



Cisco ASR 5x00 Packet Data Serving Node Administration Guide

Version 14.0

Last Updated May 31, 2013

Americas Headquarters

Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
<http://www.cisco.com>
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

Text Part Number: OL-27243-02

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Cisco and the Cisco Logo are trademarks of Cisco Systems, Inc. and/or its affiliates in the U.S. and other countries. A listing of Cisco's trademarks can be found at www.cisco.com/go/trademarks. Third party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company.

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

Cisco ASR 5x00 Packet Data Serving Node Administration Guide

© 2013 Cisco Systems, Inc. All rights reserved.

CONTENTS

About this Guide	XV
Conventions Used	xvi
Contacting Customer Support	xviii
Additional Information	xix
CDMA2000 Wireless Data Services.....	21
Product Description	22
Features and Functionality—Base Software	23
Gx and Gy Support.....	23
RADIUS Support	24
Description	24
Access Control List Support.....	25
IP Policy Forwarding	26
Description	26
AAA Server Groups.....	26
Description	26
Overlapping IP Address Pool Support	27
Routing Protocol Support	27
Description	27
Management System Overview	28
Description	28
Bulk Statistics Support	29
Description	29
Threshold Crossing Alerts (TCA) Support.....	30
Description	30
IP Header Compression - Van Jacobson.....	31
Description	31
DSCP Marking.....	31
Features and Functionality - Optional Enhanced Software Features.....	32
Session Recovery Support.....	32
Description	32
IPv6 Support.....	33
Description	33
L2TP LAC Support	34
Description	34
L2TP LNS Support	34
Description	34
Proxy Mobile IP	35
Description	35
IP Security (IPSec).....	35
Description	35
Traffic Policing and Rate Limiting.....	36
Description	36
Intelligent Traffic Control.....	37
Dynamic RADIUS Extensions (Change of Authorization).....	37
Description	37

Web Element Management System	38
Description	38
Features and Functionality - Inline Service Support	39
Content Filtering	39
Integrated Adult Content Filter	39
ICAP Interface	39
Network Address Translation (NAT)	40
Peer-to-Peer Detection	40
Personal Stateful Firewall	41
Traffic Performance Optimization (TPO)	41
Features and Functionality - External Application Support	43
Mobility Unified Reporting	43
CDMA2000 Data Network Deployment Configurations	44
Standalone PDSN/FA and HA Deployments	44
Interface Descriptions	44
Co-Located Deployments	45
Understanding Simple IP and Mobile IP	47
Simple IP	47
How Simple IP Works	47
Mobile IP	50
Mobile IP Tunneling Methods	50
How Mobile IP Works	52
Proxy Mobile IP	55
How Proxy Mobile IP Works	56
Supported Standards	61
Requests for Comments (RFCs)	61
TIA and Other Standards	64
Telecommunications Industry Association (TIA) Standards	64
Object Management Group (OMG) Standards	64
3GPP2 Standards	64
IEEE Standards	65
Understanding the Service Operation and Configuration	67
Terminology	68
Contexts	68
AAA Realms	68
Ports	69
Logical Interfaces	69
Bindings	70
Services	70
AAA Servers	72
Subscribers	72
Default Subscribers and Realm-based Subscriber Templates	73
How the System Selects Contexts	75
Context Selection for Context-level Administrative User Sessions	75
Context Selection for Subscriber Sessions	76
AAA Context Selection for Subscriber Sessions	76
Destination Context Selection For Subscriber Sessions	78
Simple IP Configuration Examples	81
Example 1: Simple IP Support Using a Single Source and Destination Context	82
Information Required	83
Source Context Configuration	83
Destination Context Configuration	85
How This Configuration Works	86

Example 2: Simple IP Using a Single Source Context and Multiple Outsourced Destination Contexts	88
Information Required	89
Source Context Configuration	89
Destination Context Configuration	91
System-Level AAA Configuration	93
How This Configuration Works	94
Mobile IP Configuration Examples	99
Example 1: Mobile IP Support Using the System as a PDSN/FA	100
Information Required	101
Source Context Configuration	101
AAA Context Configuration	102
Mobile IP Destination Context Configuration	104
System-Level AAA Configuration	105
Optional Destination Context	106
How This Configuration Works	108
Example 2: Mobile IP Support Using the System as an HA	110
Information Required	110
Source Context Configuration	110
Destination Context Configuration	114
How This Configuration Works	115
Example 3: HA Using a Single Source Context and Multiple Outsourced Destination Contexts	117
Information Required	118
Source Context Configuration	118
Destination Context Configuration	121
System-Level AAA Configuration	123
How This Configuration Works	124
Simple IP and Mobile IP in a Single System Configuration Example	127
Using the System as Both a PDSN/FA and an HA	128
Information Required	128
Source Context Configuration	129
AAA Context Configuration	130
Mobile IP Destination Context Configuration	132
Simple IP Destination Context	135
System-Level AAA Parameter Configuration	136
How This Configuration Works	137
Service Configuration Procedures	141
Creating and Configuring PDSN Services	142
Verifying the PDSN Services	143
Creating and Configuring FA Services	146
Verifying the FA Service	147
Creating and Configuring HA Services	149
Verifying the HA Service	149
Configuring IP Address Pools on the System	152
Creating IPv4 Pool	152
Creating IPv6 Pool	153
Adding Overlap-Pool Addresses to Routing	153
Verifying IP Pool Configuration	153
Monitoring the Service	155
Monitoring System Status and Performance	156
Clearing Statistics and Counters	160
Troubleshooting the System	161

Test Commands	162
Using the PPP Echo-Test Command	162
Engineering Rules.....	163
Interface and Port Rules	164
R-P Interface Rules	164
Pi Interface Rules	164
FA to HA Rules	164
HA to FA.....	165
Subscriber Rules	166
Service Rules.....	167
Supported Registration Reply Codes	169
PDSN Service Reply Codes	169
FA Service Reply Codes.....	170
Mobile-IP and Proxy-MIP Timer Considerations.....	173
Call Flow Summary.....	174
Dealing with the	176
Controlling the Mobile IP Lifetime on a Per-Domain Basis.....	177
Always-on	181
Overview	182
Configuring Always-on.....	183
Configuring Always-on.....	183
Verifying Your Configuration	184
Broadcast Multicast Service	185
Overview	186
Licensing	186
Configuring BCMCS	187
BCMCS Group Configuration	187
RADIUS Server Configuration	187
CoA, RADIUS DM, and Session Redirection (Hotlining).....	189
RADIUS Change of Authorization and Disconnect Message.....	190
CoA Overview.....	190
DM Overview	190
License Requirements.....	190
Enabling CoA and DM.....	190
Enabling CoA and DM.....	191
CoA and DM Attributes	191
CoA and DM Error-Cause Attribute	192
Viewing CoA and DM Statistics	193
Session Redirection (Hotlining)	196
Overview.....	196
License Requirements	196
Operation.....	196
ACL Rule	196
Redirecting Subscriber Sessions	196
Session Limits On Redirection	197
Stopping Redirection.....	197
Handling IP Fragments	197
Recovery	197
AAA Accounting	197
Viewing the Redirected Session Entries for a Subscriber.....	197

Gx Interface Support	203
Rel. 6 Gx Interface.....	204
Introduction.....	204
Supported Networks and Platforms	204
License Requirements	205
Supported Standards	205
How it Works	205
Configuring Rel. 6 Gx Interface.....	207
Configuring IMS Authorization Service at Context Level	208
Verifying IMS Authorization Service Configuration	209
Applying IMS Authorization Service to an APN	209
Verifying Subscriber Configuration	210
Rel. 7 Gx Interface.....	211
Introduction.....	211
Supported Networks and Platforms	213
License Requirements	213
Supported Standards	213
Terminology and Definitions.....	214
Policy Control.....	214
Charging Control.....	217
Policy and Charging Control (PCC) Rules	218
PCC Procedures over Gx Reference Point	219
Volume Reporting Over Gx.....	221
How Rel. 7 Gx Works	224
Configuring Rel. 7 Gx Interface.....	227
Configuring IMS Authorization Service at Context Level	228
Applying IMS Authorization Service to an APN	230
Configuring Volume Reporting over Gx	231
Gathering Statistics	232
Rel. 8 Gx Interface.....	233
HA/PDSN Rel. 8 Gx Interface Support.....	233
Introduction	233
Terminology and Definitions	235
How it Works.....	241
Configuring HA/PDSN Rel. 8 Gx Interface Support.....	243
Gathering Statistics.....	246
P-GW Rel. 8 Gx Interface Support.....	247
Introduction	247
Terminology and Definitions	248
Rel. 9 Gx Interface.....	252
P-GW Rel. 9 Gx Interface Support.....	252
Introduction	252
Terminology and Definitions	252
Gy Interface Support	257
Introduction	258
License Requirements.....	259
Supported Standards	259
Features and Terminology.....	260
Charging Scenarios.....	260
Session Charging with Reservation	260
Basic Operations.....	260
Re-authorization.....	261
Threshold based Re-authorization Triggers	261

Termination Action	261
Diameter Base Protocol.....	261
Diameter Credit Control Application	262
Quota Behavior	263
Supported AVPs.....	274
Unsupported AVPs.....	277
Configuring Gy Interface Support	284
Configuring GGSN / P-GW / IPSG Gy Interface Support.....	284
Configuring HA / PDSN Gy Interface Support.....	285
Gathering Statistics	287
IP Header Compression	289
Overview	290
Configuring VJ Header Compression for PPP.....	291
Enabling VJ Header Compression	291
Verifying the VJ Header Compression Configuration.....	291
Configuring RoHC Header Compression for PPP	293
Enabling RoHC Header Compression for PPP	293
Verifying the Header Compression Configuration	294
Configuring Both RoHC and VJ Header Compression.....	295
Enabling RoHC and VJ Header Compression for PPP.....	295
Verifying the Header Compression Configuration	296
Configuring RoHC for Use with SO67 in PDSN or HSGW Service.....	297
Enabling RoHC Header Compression with PDSN	297
Enabling RoHC Header Compression with HSGW	298
Verifying the Header Compression Configuration	298
Using an RoHC Profile for Subscriber Sessions	299
Creating RoHC Profile for Subscriber using Compression Mode	299
Creating RoHC Profile for Subscriber using Decompression Mode	300
Applying RoHC Profile to a Subscriber	301
Verifying the Header Compression Configuration	301
Disabling VJ Header Compression Over PPP.....	302
Disabling VJ Header Compression	302
Verifying the VJ Header Compression Configuration.....	302
Disabling RoHC Header Compression Over SO67	304
Disabling RoHC Header Compression.....	304
Verifying the Header Compression Configuration	304
Checking IP Header Compression Statistics.....	306
RADIUS Attributes for IP Header Compression	307
IP Pool Sharing Protocol.....	309
Overview	310
Primary HA Functionality	310
Secondary HA Functionality	310
Requirements, Limitations, & Behavior	311
How IPSP Works	312
IPSP Operation for New Sessions	312
IPSP Operation for Session Handoffs.....	314
Configuring IPSP Before the Software Upgrade	316
Configuring the AAA Server for IPSP	316
Enabling IPSP on the Secondary HA	317
Enabling IPSP on the Primary HA.....	317
Verifying the IPSP Configuration	318
Configuring IPSP After the Software Upgrade	319
Disabling IPSP	320

IP Security	321
Overview	323
Applicable Products and Relevant Sections	324
IPSec Terminology	327
Crypto Access Control List (ACL)	327
Transform Set	327
ISAKMP Policy	327
Crypto Map	327
Manual Crypto Maps	328
ISAKMP Crypto Maps	328
Dynamic Crypto Maps	328
Implementing IPSec for PDN Access Applications	329
How the IPSec-based PDN Access Configuration Works	329
Configuring IPSec Support for PDN Access	330
Implementing IPSec for Mobile IP Applications	332
How the IPSec-based Mobile IP Configuration Works	332
Configuring IPSec Support for Mobile IP	334
Implementing IPSec for L2TP Applications	336
How IPSec is Used for Attribute-based L2TP Configurations	336
Configuring Support for L2TP Attribute-based Tunneling with IPSec	338
How IPSec is Used for PDSN Compulsory L2TP Configurations	339
Configuring Support for L2TP PDSN Compulsory Tunneling with IPSec	340
How IPSec is Used for L2TP Configurations on the GGSN	341
Configuring GGSN Support for L2TP Tunneling with IPSec	342
Transform Set Configuration	343
Configuring Transform Set	343
Verifying the Crypto Transform Set Configuration	343
ISAKMP Policy Configuration	345
Configuring ISAKMP Policy	345
Verifying the ISAKMP Policy Configuration	346
ISAKMP Crypto Map Configuration	347
Configuring ISAKMP Crypto Maps	347
Verifying the ISAKMP Crypto Map Configuration	348
Dynamic Crypto Map Configuration	350
Configuring Dynamic Crypto Maps	350
Verifying the Dynamic Crypto Map Configuration	350
Manual Crypto Map Configuration	352
Configuring Manual Crypto Maps	352
Verifying the Manual Crypto Map Configuration	353
Crypto Map and Interface Association	355
Applying Crypto Map to an Interface	355
Verifying the Interface Configuration with Crypto Map	355
FA Services Configuration to Support IPSec	357
Modifying FA service to Support IPSec	357
Verifying the FA Service Configuration with IPSec	358
HA Service Configuration to Support IPSec	359
Modifying HA service to Support IPSec	359
Verifying the HA Service Configuration with IPSec	360
RADIUS Attributes for IPSec-based Mobile IP Applications	361
LAC Service Configuration to Support IPSec	362
Modifying LAC service to Support IPSec	362
Verifying the LAC Service Configuration with IPSec	363
Subscriber Attributes for L2TP Application IPSec Support	364
PDSN Service Configuration for L2TP Support	365

Modifying PDSN service to Support Attribute-based L2TP Tunneling	365
Modifying PDSN service to Support Compulsory L2TP Tunneling	366
Verifying the PDSN Service Configuration for L2TP	366
Redundant IPSec Tunnel Fail-Over	367
Supported Standards	367
Redundant IPSec Tunnel Fail-over Configuration	368
Configuring Crypto Group	368
Modify ISAKMP Crypto Map Configuration to Match Crypto Group	369
Verifying the Crypto Group Configuration	369
Dead Peer Detection (DPD) Configuration	371
Configuring Crypto Group	371
Verifying the DPD Configuration	372
APN Template Configuration to Support L2TP	373
Modifying APN Template to Support L2TP	373
Verifying the APN Configuration for L2TP	374
IPSec for LTE/SAE Networks	375
Encryption Algorithms	375
HMAC Functions	375
Diffie-Hellman Groups	375
Dynamic Node-to-Node IPSec Tunnels	376
ACL-based Node-to-Node IPSec Tunnels	376
Traffic Selectors	376
Authentication Methods	377
X.509 Certificate-based Peer Authentication	377
Certificate Revocation Lists	379
Child SA Rekey Support	379
IKEv2 Keep-Alive Messages (Dead Peer Detection)	379
E-UTRAN/EPC Logical Network Interfaces Supporting IPSec Tunnels	380
IPSec Tunnel Termination	381
IPSec for Femto-UMTS Networks	382
Authentication Methods	382
Crypto map Template Configuration	382
X.509 Certificate-based Peer Authentication	383
Certificate Revocation Lists	385
Child SA Rekey Support	385
IKEv2 Keep-Alive Messages (Dead Peer Detection)	385
IPSec Tunnel Termination	386
x.509 Certificate Configuration	386
Intelligent Traffic Control	389
Overview	390
ITC and EV-DO Rev A in 3GPP2 Networks	390
Bandwidth Control and Limiting	390
Licensing	391
How it Works	392
Configuring Flow-based Traffic Policing	393
Configuring Class Maps	393
Configuring Policy Maps	394
Configuring Policy Groups	395
Configuring a Subscriber for Flow-based Traffic Policing	395
Verifying Flow-based Traffic Policing Configuration	396
L2TP Access Concentrator	397
Applicable Products and Relevant Sections	398
Supported LAC Service Configurations for PDSN Simple IP	399

Attribute-based Tunneling	399
How The Attribute-based L2TP Configuration Works	400
Configuring Attribute-based L2TP Support for PDSN Simple IP	400
PDSN Service-based Compulsory Tunneling	401
How PDSN Service-based Compulsory Tunneling Works	401
Configuring L2TP Compulsory Tunneling Support for PDSN Simple IP	402
Supported LAC Service Configurations for the GGSN and P-GW	404
Transparent IP PDP Context Processing with L2TP Support	405
Non-transparent IP PDP Context Processing with L2TP Support	406
PPP PDP Context Processing with L2TP Support	407
Configuring the GGSN or P-GW to Support L2TP	408
Supported LAC Service Configuration for Mobile IP	409
How The Attribute-based L2TP Configuration for MIP Works	409
Configuring Attribute-based L2TP Support for HA Mobile IP	410
Configuring Subscriber Profiles for L2TP Support	412
RADIUS and Subscriber Profile Attributes Used	412
RADIUS Tagging Support	413
Configuring Local Subscriber Profiles for L2TP Support	413
Configuring Local Subscriber	414
Verifying the L2TP Configuration	414
Tunneling All Subscribers in a Specific Context Without Using RADIUS Attributes	415
Configuring LAC Services	416
Configuring LAC Service	416
Configuring LNS Peer	417
Verifying the LAC Service Configuration	417
Modifying PDSN Services for L2TP Support	419
Modifying PDSN Service	419
Verifying the PDSN Service for L2TP Support	420
Modifying APN Templates to Support L2TP	421
Assigning LNS Peer Address in APN Template	421
Configuring Outbound Authentication	422
Verifying the APN Configuration	422
L2TP Network Server	423
LNS Service Operation	424
Information Required	425
Source Context Configuration	425
Destination Context Configuration	427
How This Configuration Works	428
Configuring the System to Support LNS Functionality	431
Creating and Binding LNS Service	431
Configuring Authentication Parameters for LNS Service	432
Configuring Tunnel and Session Parameters for LNS Service	432
Configuring Peer LAC servers for LNS Service	433
Configuring Domain Alias for AAA Subscribers	433
Verifying the LNS Service Configuration	433
Mobile IP Registration Revocation	435
Overview	436
Configuring Registration Revocation	438
Configuring FA Services	438
Configuring HA Services	438
Policy Forwarding	441
Overview	442

IP Pool-based Next Hop Forwarding	443
Configuring IP Pool-based Next Hop Forwarding	443
Subscriber-based Next Hop Forwarding	444
Configuring Subscriber-based Next Hop Forwarding.....	444
ACL-based Policy Forwarding	445
Configuring ACL-based Policy Forwarding	445
Applying the ACL to an IP Access Group	445
Applying the ACL to a Destination Context.....	445
Applying the ACL to an Interface in a Destination Context.....	446
Pre-paid Billing.....	447
Overview	448
3GPP2 Standard Pre-paid Billing Overview	448
Custom Pre-paid Billing Overview	448
License Requirements.....	449
Configuring Standard 3GPP2 Pre-paid Billing.....	450
Configuring Pre-paid Billing With Custom Behavior	452
3GPP2 Pre-paid Attributes	454
Pre-paid Attributes	456
Proxy-Mobile IP	457
Overview	458
Proxy Mobile IP in 3GPP2 Service	459
Proxy Mobile IP in 3GPP Service	459
Proxy Mobile IP in WiMAX Service	460
How Proxy Mobile IP Works in 3GPP2 Network	461
Scenario 1: AAA server and PDSN/FA Allocate IP Address.....	461
Scenario 2: HA Allocates IP Address	463
How Proxy Mobile IP Works in 3GPP Network	466
How Proxy Mobile IP Works in WiMAX Network	470
Scenario 1: AAA server and ASN GW/FA Allocate IP Address	470
Scenario 2: HA Allocates IP Address	472
How Proxy Mobile IP Works in a WiFi Network with Multiple Authentication	475
Configuring Proxy Mobile-IP Support	480
Configuring FA Services.....	480
Verify the FA Service Configuration	481
Configuring Proxy MIP HA Failover.....	481
Configuring HA Services	482
Configuring Subscriber Profile RADIUS Attributes.....	483
RADIUS Attributes Required for Proxy Mobile IP	483
Configuring Local Subscriber Profiles for Proxy-MIP on a PDSN.....	484
Configuring Local Subscriber Profiles for Proxy-MIP on a PDIF	485
Configuring Default Subscriber Parameters in Home Agent Context.....	485
Configuring APN Parameters.....	485
Rejection/Redirection of HA Sessions on Network Failures	489
Overview	490
Configuring HA Session Redirection	491
RADIUS Attributes	495
Remote Address-based RADIUS Accounting.....	497
Overview	498
License Requirements.....	498
Configuring Remote Address-based Accounting	499
Verifying the Remote Address Lists	499
Subscriber Attribute Configuration.....	500




Supported RADIUS Attributes	500
Configuring Local Subscriber Profiles	500
Traffic Policing and Shaping	503
Overview	504
Traffic Policing	504
Traffic Shaping	504
Traffic Policing Configuration	505
Configuring Subscribers for Traffic Policing	505
Configuring APN for Traffic Policing in 3GPP Networks	506
Traffic Shaping Configuration	508
Configuring Subscribers for Traffic Shaping	508
Configuring APN for Traffic Shaping in 3GPP Networks	509
RADIUS Attributes	512
Traffic Policing for CDMA Subscribers	512
Traffic Policing for UMTS Subscribers	513

About this Guide

This document pertains to the features and functionality that run on and/or that are related to the Cisco® ASR 5000 Chassis.

Conventions Used

The following tables describe the conventions used throughout this documentation.

Icon	Notice Type	Description
	Information Note	Provides information about important features or instructions.
	Caution	Alerts you of potential damage to a program, device, or system.
	Warning	Alerts you of potential personal injury or fatality. May also alert you of potential electrical hazards.

Typeface Conventions	Description
Text represented as a <i>screen display</i>	This typeface represents displays that appear on your terminal screen, for example: <i>Login:</i>
Text represented as commands	This typeface represents commands that you enter, for example: show ip access-list This document always gives the full form of a command in lowercase letters. Commands are not case sensitive.
Text represented as a command variable	This typeface represents a variable that is part of a command, for example: show card slot_number slot_number is a variable representing the desired chassis slot number.
Text represented as menu or sub-menu names	This typeface represents menus and sub-menus that you access within a software application, for example: Click the File menu, then click New

Command Syntax Conventions	Description
{ keyword or <i>variable</i> }	Required keywords and variables are surrounded by grouped braces. Required keywords and variables are those components that are required to be entered as part of the command syntax.
[keyword or <i>variable</i>]	Optional keywords or variables, or those that a user may or may not choose to use, are surrounded by brackets.

Command Syntax Conventions	Description
	<p>Some commands support alternative variables. These options are documented within braces or brackets by separating each variable with a vertical bar.</p> <p>These variables can be used in conjunction with required or optional keywords or variables. For example:</p> <pre>{ nonce timestamp }</pre> <p>OR</p> <pre>[count <i>number_of_packets</i> size <i>number_of_bytes</i>]</pre>

Contacting Customer Support

Use the information in this section to contact customer support.

Refer to the support area of <http://www.cisco.com> for up-to-date product documentation or to submit a service request. A valid username and password are required to access this site. Please contact your Cisco sales or service representative for additional information.

Additional Information

Refer to the following guides for supplemental information about the system:

- *Cisco ASR 5000 Installation Guide*
- *Cisco ASR 5000 System Administration Guide*
- *Cisco ASR 5x00 Command Line Interface Reference*
- *Cisco ASR 5x00 Thresholding Configuration Guide*
- *Cisco ASR 5x00 SNMP MIB Reference*
- *Web Element Manager Installation and Administration Guide*
- *Cisco ASR 5x00 AAA Interface Administration and Reference*
- *Cisco ASR 5x00 GTPP Interface Administration and Reference*
- *Cisco ASR 5x00 Release Change Reference*
- *Cisco ASR 5x00 Statistics and Counters Reference*
- *Cisco ASR 5x00 Gateway GPRS Support Node Administration Guide*
- *Cisco ASR 5x00 HRPD Serving Gateway Administration Guide*
- *Cisco ASR 5000 IP Services Gateway Administration Guide*
- *Cisco ASR 5x00 Mobility Management Entity Administration Guide*
- *Cisco ASR 5x00 Packet Data Network Gateway Administration Guide*
- *Cisco ASR 5x00 Packet Data Serving Node Administration Guide*
- *Cisco ASR 5x00 System Architecture Evolution Gateway Administration Guide*
- *Cisco ASR 5x00 Serving GPRS Support Node Administration Guide*
- *Cisco ASR 5x00 Serving Gateway Administration Guide*
- *Cisco ASR 5000 Session Control Manager Administration Guide*
- *Cisco ASR 5000 Packet Data Gateway/Tunnel Termination Gateway Administration Guide*
- Release notes that accompany updates and upgrades to the StarOS for your service and platform

Chapter 1

CDMA2000 Wireless Data Services

The ASR 5x00 provides wireless carriers with a flexible solution that functions as a Packet Data Support Node (PDSN) in CDMA 2000 wireless data networks.

This overview provides general information about the PDSN including:

- [Product Description](#)
- [Features and FunctionalityBase Software](#)
- [Features and Functionality - Optional Enhanced Software Features](#)
- [CDMA2000 Data Network Deployment Configurations](#)
- [Understanding Simple IP and Mobile IP](#)
- [Supported Standards](#)

Product Description

The system provides wireless carriers with a flexible solution that can support both Simple IP and Mobile IP applications (independently or simultaneously) within a single scalable platform.

When supporting Simple IP data applications, the system is configured to perform the role of a Packet Data Serving Node (PDSN) within the carrier's 3G CDMA2000 data network. The PDSN terminates the mobile subscriber's Point-to-Point Protocol (PPP) session and then routes data to and from the Packet Data Network (PDN) on behalf of the subscriber. The PDN could consist of Wireless Application Protocol (WAP) servers or it could be the Internet.

When supporting Mobile IP and/or Proxy Mobile IP data applications, the system can be configured to perform the role of the PDSN/Foreign Agent (FA) and/or the Home Agent (HA) within the carrier's 3G CDMA2000 data network. When functioning as an HA, the system can either be located within the carrier's 3G network or in an external enterprise or ISP network. Regardless, the PDSN/FA terminates the mobile subscriber's PPP session, and then routes data to and from the appropriate HA on behalf of the subscriber.

Features and Functionality—Base Software

This section describes the features and functions supported by default in base software on PDSN service and do not require any additional licenses.



Important: To configure the basic service and functionality on the system for PDSN service, refer to the configuration examples provided in the PDSN Administration Guide.

This section describes following features:

- [Gx and Gy Support](#)
- [RADIUS Support](#)
- [Access Control List Support](#)
- [IP Policy Forwarding](#)
- [AAA Server Groups](#)
- [Overlapping IP Address Pool Support](#)
- [Routing Protocol Support](#)
- [Management System Overview](#)
- [Bulk Statistics Support](#)
- [Threshold Crossing Alerts \(TCA\) Support](#)
- [IP Header Compression - Van Jacobson](#)
- [DSCP Marking](#)

Gx and Gy Support

The PDSN supports 3GPP Release 8 standards based policy interface with the Policy and Charging Rules Function (PCRF). The policy interface is based on a subset 3GPP 29.212. based Gx interface specification. The PDSN policy interface fully supports installation/modification of dynamic and predefined rules from the PCRF.

The enforcement of dynamic and predefined PCC rules installed from the PCRF is done using Enhanced Charging Services (ECS). The full ECS functionality including the DPI and P2P detection can be enabled via predefined rules using the Gx interface.

The PDSN supports a subset of event triggers as defined in 29.212. Currently the event trigger support is limited to the following:

- RAT Change
- User location change (BSID)
- AN GW change (during inter PCF handoff)

The PDSN also supports triggering of online charging via the policy interface. 3GPP Release 8 Gy interface as defined in 32.299 is used for online charging.

The PDSN supports connectivity to multiple PCRF's. The PCRF's may be referred to by an FQDN. Load balancing of sessions across multiple servers are achieved by using a round robin algorithm. Redundancy between servers can be achieved by configuring multiple weighted sets of servers.

The configuration allows Policy support to be enabled on a per subscriber/APN basis.

The policy features supported on PDSN and GGSN will be quite similar. On PDSN the Gx will only be supported for Simple IP calls.

On PDSN additional event triggers rat type change and location change will be supported. On PDSN Gy, standard DCCA based credit control is supported; 3GPP related trigger functionality is not supported on PDSN Gy.

RADIUS Support

Provides a mechanism for performing authorization, authentication, and accounting (AAA) for subscriber PDP contexts based on the following standards:

- RFC-2618, RADIUS Authentication Client MIB, June 1999
- RFC-2620, RADIUS Accounting Client MIB, June 1999
- RFC-2865, Remote Authentication Dial In User Service (RADIUS), June 2000
- RFC-2866, RADIUS Accounting, June 2000
- RFC-2867, RADIUS Accounting Modifications for Tunnel Protocol Support, June 2000
- RFC-2868, RADIUS Attributes for Tunnel Protocol Support, June 2000
- RFC-2869, RADIUS Extensions, June 2000

Description

The Remote Authentication Dial-In User Service (RADIUS) protocol is used to provide AAA functionality for subscriber PDP contexts.

Within context contexts configured on the system, there are AAA and RADIUS protocol-specific parameters that can be configured. The RADIUS protocol-specific parameters are further differentiated between RADIUS Authentication server RADIUS Accounting server interaction.

Among the RADIUS parameters that can be configured are:

- **Priority:** Dictates the order in which the servers are used allowing for multiple servers to be configured in a single context.
- **Routing Algorithm:** Dictate the method for selecting among configured servers. The specified algorithm dictates how the system distributes AAA messages across the configured AAA servers for new sessions. Once a session is established and an AAA server has been selected, all subsequent AAA messages for the session will be delivered to the same server.

In the event that a single server becomes unreachable, the system attempts to communicate with the other servers that are configured. The system also provides configurable parameters that specify how it should behave should all of the RADIUS AAA servers become unreachable.

The system provides an additional level of flexibility by supporting the configuration RADIUS server groups. This functionality allows operators to differentiate AAA services based on the subscriber template used to facilitate their PDP context.

In general, 128 AAA Server IP address/port per context can be configured on the system and it selects servers from this list depending on the server selection algorithm (round robin, first server). Instead of having a single list of servers per context, this feature provides the ability to configure multiple server groups. Each server group, in turn, consists of a list of servers.

This feature works in following way:

- All RADIUS authentication/accounting servers configured at the context-level are treated as part of a server group named “default”. This default server group is available to all subscribers in that context through the realm (domain) without any configuration.
- It provides a facility to create “user defined” RADIUS server groups, as many as 399 (excluding “default” server group), within a context. Any of the user defined RADIUS server groups are available for assignment to a subscriber through the subscriber configuration within that context.

Since the configuration of the subscriber can specify the RADIUS server group to use as well as IP address pools from which to assign addresses, the system implements a mechanism to support some in-band RADIUS server implementations (i.e. RADIUS servers which are located in the corporate network, and not in the operator's network) where the NAS-IP address is part of the subscriber pool. In these scenarios, the PDSN supports the configuration of the first IP address of the subscriber pool for use as the RADIUS NAS-IP address.



Important: In 12.3 and earlier releases, refer to the *AAA and GTPP Interface Administration and Reference* for more information on RADIUS AAA configuration. In 14.0 and later releases, refer to the *AAA Interface Administration and Reference*.

Access Control List Support

Access Control Lists provide a mechanism for controlling (i.e permitting, denying, redirecting, etc.) packets in and out of the system.

IP access lists, or Access Control Lists (ACLs) as they are commonly referred to, are used to control the flow of packets into and out of the system. They are configured on a per-context basis and consist of “rules” (ACL rules) or filters that control the action taken on packets that match the filter criteria. Once configured, an ACL can be applied to any of the following:

- An individual interface
- All traffic facilitated by a context (known as a policy ACL)
- An individual subscriber
- All subscriber sessions facilitated by a specific context

There are two primary components of an ACL:

- **Rule:** A single ACL consists of one or more ACL rules. As discussed earlier, the rule is a filter configured to take a specific action on packets matching specific criteria. Up to 128 rules can be configured per ACL.
Each rule specifies the action to take when a packet matches the specifies criteria. This section discusses the rule actions and criteria supported by the system.
- **Rule Order:** A single ACL can consist of multiple rules. Each packet is compared against each of the ACL rules, in the order in which they were entered, until a match is found. Once a match is identified, all subsequent rules are ignored.



Important: For more information on Access Control List configuration, refer to the IP Access Control List chapter in System Administration Guide.

IP Policy Forwarding

IP Policy Forwarding enables the routing of subscriber data traffic to specific destinations based on configuration. This functionality can be implemented in support of enterprise-specific applications (i.e. routing traffic to specific enterprise domains) or for routing traffic to back-end servers for additional processing.

Description

The system can be configured to automatically forward data packets to a predetermined network destination. This can be done in one of three ways:

- **IP Pool-based Next Hop Forwarding** - Forwards data packets based on the IP pool from which a subscriber obtains an IP address.
- **ACL-based Policy Forwarding** - Forwards data packets based on policies defined in Access Control Lists (ACLs) and applied to contexts or interfaces.
- **Subscriber specific Next Hop Forwarding** - Forwards all packets for a specific subscriber.

The simplest way to forward subscriber data is to use IP Pool-based Next Hop Forwarding. An IP pool is configured with the address of a next hop gateway and data packets from all subscribers using the IP pool are forward to that gateway.

Subscriber Next Hop forwarding is also very simple. In the subscriber configuration a nexthop forwarding address is specified and all data packets for that subscriber are forwarded to the specified nexthop destination.

ACL-based Policy Forwarding gives you more control on redirecting data packets. By configuring an Access Control List (ACL) you can forward data packets from a context or an interface by different criteria, such as; source or destination IP address, ICMP type, or TCP/UDP port numbers.

ACLs are applied first. If ACL-based Policy Forwarding and Pool-based Next Hop Forwarding or Subscriber are configured, data packets are first redirected as defined in the ACL, then all remaining data packets are redirected to the next hop gateway defined by the IP pool or subscriber profile.

AAA Server Groups

Value-added feature to enable VPN service provisioning for enterprise or MVNO customers. Enables each corporate customer to maintain its own AAA servers with its own unique configurable parameters and custom dictionaries.

Description

This feature provides support for up to 800 AAA (RADIUS and Diameter) server groups and 800 NAS IP addresses that can be provisioned within a single context or across the entire chassis. A total of 128 servers can be assigned to an individual server group. Up to 1,600 accounting, authentication and/or mediation servers are supported per chassis and may be distributed across a maximum of 1,000 subscribers. This feature also enables the AAA servers to be distributed across multiple subscribers within the same context.



Important: Due to additional memory requirements, this service can only be used with 8GB Packet Accelerator Cards (PACs) or Packet Service Cards (PSCs)



Important: In 12.3 and earlier releases, refer to the *AAA and GTPP Interface Administration and Reference* for more information on AAA Server Group configuration. In 14.0 and later releases, refer to the *AAA Interface Administration and Reference*.

Overlapping IP Address Pool Support

Overlapping IP Address Pools provides a mechanism for allowing operators to more flexibly support multiple corporate VPN customers with the same private IP address space without the expensive investments in physically separate routers, or expensive configurations using virtual routers.



Important: For more information on IP pool overlapping configuration, refer to the VLANs chapter in the *System Administration Guide*.

Routing Protocol Support

The system's support for various routing protocols and routing mechanism provides an efficient mechanism for ensuring the delivery of subscriber data packets.

Description

The following routing mechanisms and protocols are supported by the system:

- **Static Routes:** The system supports the configuration of static network routes on a per context basis. Network routes are defined by specifying an IP address and mask for the route, the name of the interface in the current context that the route must use, and a next hop IP address.
- **Open Shortest Path First (OSPF) Protocol version 2:** A link-state routing protocol, OSPF is an Interior Gateway Protocol (IGP) that routes IP packets based solely on the destination IP address found in the IP packet header using the shortest path first. IP packets are routed “as is”, meaning they are not encapsulated in any further protocol headers as they transit the network.

Variable length subnetting, areas, and redistribution into and out of OSPF are supported.

OSPF routing is supported in accordance with the following standards:

- RFC-1850, OSPF Version 2 Management Information Base, November 1995
- RFC-2328, OSPF Version 2, April 1998
- RFC-3101 OSPF-NSSA Option, January 2003
- **Border Gateway Protocol version 4 (BGP-4):** The system supports a subset of BGP (RFC-1771, A Border Gateway Protocol 4 (BGP-4)), suitable for eBGP support of multi-homing typically used to support geographically redundant mobile gateways, is supported.

EBGP is supported with multi-hop, route filtering, redistribution, and route maps. The network command is support for manual route advertisement or redistribution.

BGP route policy and path selection is supported by the following means:

- Prefix match based on route access list
- AS path access-list

- Modification of AS path through path prepend
- Origin type
- MED
- Weight
- **Route Policy:** Routing policies modify and redirect routes to and from the system to satisfy specific routing needs. The following methods are used with or without active routing protocols (i.e. static or dynamic routing) to prescribe routing policy:
 - **Route Access Lists:** The basic building block of a routing policy, route access lists filter routes based upon a specified range of IP addresses.
 - **IP Prefix Lists:** A more advanced element of a routing policy. An IP Prefix list filters routes based upon IP prefixes.
 - **AS Path Access Lists:** A basic building block used for Border Gateway Protocol (BGP) routing, these lists filter Autonomous System (AS) paths.
- **Route Maps:** Route-maps are used for detailed control over the manipulation of routes during route selection or route advertisement by a routing protocol and in route redistribution between routing protocols. This detailed control is achieved using IP Prefix Lists, Route Access Lists and AS Path Access Lists to specify IP addresses, address ranges, and Autonomous System Paths.
- **Equal Cost Multiple Path (ECMP):** ECMP allows distribution of traffic across multiple routes that have the same cost to the destination. In this manner, throughput load is distributed across multiple path, typically to lessen the burden on any one route and provide redundancy. The mobile gateway supports from four to ten equal-cost paths.



Important: For more information on IP Routing configuration, refer to the Routing chapter in the *System Administration Guide*.

Management System Overview

The system's management capabilities are designed around the Telecommunications Management Network (TMN) model for management -- focusing on providing superior quality Network Element (NE) and element management system (Web Element Manager) functions. The system provides element management applications that can easily be integrated, using standards-based protocols (CORBA and SNMPv1, v2), into higher-level management systems -- giving wireless operators the ability to integrate the system into their overall network, service, and business management systems. In addition, all management is performed out-of-band for security and to maintain system performance.

Description

Cisco's O&M module offers comprehensive management capabilities to the operators and enables them to operate the system more efficiently. There are multiple ways to manage the system either locally or remotely using its out-of-band management interfaces.

These include:

- Using the Command Line Interface (CLI)
- Remote login using Telnet, and Secure Shell (SSH) access to CLI through SPIO card's Ethernet management interfaces

- Local login through the Console port on SPIO card using an RS-232 serial connection
- Using the Web Element Manager application
- Supports communications through 10 Base-T, 100 Base-TX, 1000 Base-TX, or 1000 Base-SX (optical gigabit Ethernet) Ethernet management interfaces on the SPIO
- Client-Server model supports any browser (i.e. Microsoft Internet Explorer v5.0 and above or Netscape v4.7 or above, and others)
- Supports Common Object Request Broker Architecture (CORBA) protocol and Simple Network Management Protocol version 1 (SNMPv1) for fault management
- Provides complete Fault, Configuration, Accounting, Performance, and Security (FCAPS) capabilities
- Can be easily integrated with higher-level network, service, and business layer applications using the Object Management Group's (OMG's) Interface Definition Language (IDL)



Important: For more information on command line interface based management, refer to the CDMA Command Line Interface Reference and PDSN Administration Guide.

Bulk Statistics Support

The system's support for bulk statistics allows operators to choose to view not only statistics that are of importance to them, but also to configure the format in which it is presented. This simplifies the post-processing of statistical data since it can be formatted to be parsed by external, back-end processors.

When used in conjunction with the Web Element Manager, the data can be parsed, archived, and graphed.

Description

The system can be configured to collect bulk statistics (performance data) and send them to a collection server (called a receiver). Bulk statistics are statistics that are collected in a group. The individual statistics are grouped by schema. The following schemas are supported:

- **System:** Provides system-level statistics
- **Card:** Provides card-level statistics
- **Port:** Provides port-level statistics
- **BCMCS:** Provides BCMCS service statistics
- **FA:** Provides FA service statistics
- **HA:** Provides HA service statistics
- **IP Pool:** Provides IP pool statistics
- **MIPv6HA:** Provides MIPv6HA service statistics
- **PPP:** Provides Point-to-Point Protocol statistics
- **RADIUS:** Provides per-RADIUS server statistics
- **ECS:** Provides Enhanced Charging Service Statistics

The system supports the configuration of up to 4 sets (primary/secondary) of receivers. Each set can be configured with to collect specific sets of statistics from the various schemas. Statistics can be pulled manually from the IMG or sent at configured intervals. The bulk statistics are stored on the receiver(s) in files.

The format of the bulk statistic data files can be configured by the user. Users can specify the format of the file name, file headers, and/or footers to include information such as the date, IMG host name, IMG uptime, the IP address of the system generating the statistics (available for only for headers and footers), and/or the time that the file was generated.

When the Web Element Manager is used as the receiver, it is capable of further processing the statistics data through XML parsing, archiving, and graphing.

The Bulk Statistics Server component of the Web Element Manager parses collected statistics and stores the information in the PostgreSQL database. If XML file generation and transfer is required, this element generates the XML output and can send it to a Northbound NMS or an alternate bulk statistics server for further processing.

Additionally, if archiving of the collected statistics is desired, the Bulk Statistics server writes the files to an alternative directory on the server. A specific directory can be configured by the administrative user or the default directory can be used. Regardless, the directory can be on a local file system or on an NFS-mounted file system on the Web Element Manager server.

Threshold Crossing Alerts (TCA) Support

Thresholding on the system is used to monitor the system for conditions that could potentially cause errors or outage. Typically, these conditions are temporary (i.e high CPU utilization, or packet collisions on a network) and are quickly resolved. However, continuous or large numbers of these error conditions within a specific time interval may be indicative of larger, more severe issues. The purpose of thresholding is to help identify potentially severe conditions so that immediate action can be taken to minimize and/or avoid system downtime.

The system supports Threshold Crossing Alerts for certain key resources such as CPU, memory, IP pool addresses, etc. With this capability, the operator can configure threshold on these resources whereby, should the resource depletion cross the configured threshold, a SNMP Trap would be sent.

Description

The following thresholding models are supported by the system:


- **Alert:** A value is monitored and an alert condition occurs when the value reaches or exceeds the configured high threshold within the specified polling interval. The alert is generated then generated and/or sent at the end of the polling interval.
- **Alarm:** Both high and low threshold are defined for a value. An alarm condition occurs when the value reaches or exceeds the configured high threshold within the specified polling interval. The alert is generated then generated and/or sent at the end of the polling interval.

Thresholding reports conditions using one of the following mechanisms:

- **SNMP traps:** SNMP traps have been created that indicate the condition (high threshold crossing and/or clear) of each of the monitored values.
Generation of specific traps can be enabled or disabled on the chassis. Ensuring that only important faults get displayed. SNMP traps are supported in both Alert and Alarm modes.
- **Logs:** The system provides a facility called threshold for which active and event logs can be generated. As with other system facilities, logs are generated Log messages pertaining to the condition of a monitored value are generated with a severity level of WARNING.
Logs are supported in both the Alert and the Alarm models.
- **Alarm System:** High threshold alarms generated within the specified polling interval are considered “outstanding” until a the condition no longer exists or a condition clear alarm is generated. “Outstanding”

alarms are reported to the system's alarm subsystem and are viewable through the Alarm Management menu in the Web Element Manager.

The Alarm System is used only in conjunction with the Alarm model.

 **Important:** For more information on threshold crossing alert configuration, refer to the *Thresholding Configuration Guide*.

IP Header Compression - Van Jacobson

Implementing IP header compression provides the following benefits:


- Improves interactive response time
- Allows the use of small packets for bulk data with good line efficiency
- Allows the use of small packets for delay sensitive low data-rate traffic
- Decreases header overhead
- Reduces packet loss rate over lossy links

Description

The system supports the Van Jacobson (VJ) IP header compression algorithms by default for subscriber traffic.

The VJ header compression is supported as per RFC 1144 (CTCP) header compression standard developed by V. Jacobson in 1990. It is commonly known as VJ compression. It describes a basic method for compressing the headers of IPv4/TCP packets to improve performance over low speed serial links.

By default IP header compression using the VJ algorithm is enabled for subscribers. You can also turn off IP header compression for a subscriber.

 **Important:** For more information on IP header compression support, refer to the IP Header Compression chapter.

DSCP Marking

Provides support for more granular configuration of DSCP marking.

For different Traffic class, the PDSN supports per-service and per-subscriber configurable DSCP marking for Uplink and Downlink direction based on Allocation/Retention Priority in addition to the current priorities.

Features and Functionality - Optional Enhanced Software Features

This section describes the optional enhanced features and functions for PDSN service.

Each of the following features require the purchase of an additional license to implement the functionality with the PDSN service.

This section describes following features:

- [Session Recovery Support](#)
- [IPv6 Support](#)
- [L2TP LAC Support](#)
- [L2TP LNS Support](#)
- [Proxy Mobile IP](#)
- [IP Security \(IPSec\)](#)
- [Traffic Policing and Rate Limiting](#)
- [Dynamic RADIUS Extensions \(Change of Authorization\)](#)
- [Web Element Management System](#)

Session Recovery Support

The Session Recovery feature provides seamless failover and reconstruction of subscriber session information in the event of a hardware or software fault within the system preventing a fully connected user session from being disconnected.

Description

Session recovery is performed by mirroring key software processes (e.g. session manager and AAA manager) within the system. These mirrored processes remain in an idle state (in standby-mode), wherein they perform no processing, until they may be needed in the case of a software failure (e.g. a session manager task aborts). The system spawns new instances of “standby mode” session and AAA managers for each active Control Processor (CP) being used.

Additionally, other key system-level software tasks, such as VPN manager, are performed on a physically separate Packet Services Card (PSC) to ensure that a double software fault (e.g. session manager and VPN manager fails at same time on same card) cannot occur. The PSC used to host the VPN manager process is in active mode and is reserved by the operating system for this sole use when session recovery is enabled.

The additional hardware resources required for session recovery include a standby System Processor Card (SPC) and a standby PSC.

There are two modes for Session Recovery.

- **Task recovery mode:** Wherein one or more session manager failures occur and are recovered without the need to use resources on a standby PSC. In this mode, recovery is performed by using the mirrored “standby-mode” session manager task(s) running on active PACs. The “standby-mode” task is renamed, made active, and is then populated using information from other tasks such as AAA manager.

- **Full recovery mode:** Used when a PSC hardware failure occurs, or when a PSC migration failure happens. In this mode, the standby PSC is made active and the “standby-mode” session manager and AAA manager tasks on the newly activated PSC perform session recovery.

Session/Call state information is saved in the peer AAA manager task because each AAA manager and session manager task is paired together. These pairs are started on physically different PACs to ensure task recovery.



Important: For more information on session recovery support, refer to the Session Recovery chapter in the *System Administration Guide*.

IPv6 Support

This feature allows IPv6 subscribers to connect via the CDMA 2000 infrastructure in accordance with the following standards:

- RFC 2460: Internet Protocol, Version 6 (IPv6) Specification
- RFC 2461: Neighbor Discovery for IPv6
- RFC 2462: IPv6 Stateless Address Autoconfiguration
- RFC 3314: Recommendations for IPv6 in 3GPP Standards
- RFC 3316: Internet Protocol Version 6 (IPv6) for Some Second and Third Generation Cellular Hosts
- RFC 3056: Connection of IPv6 domains via IPv4 clouds
- 3GPP TS 23.060: General Packet Radio Service (GPRS) Service description
- 3GPP TS 27.060: Mobile Station Supporting Packet Switched Services
- 3GPP TS 29.061: Interworking between the Public Land Mobile Network (PLMN) supporting Packet Based Services and Packet Data Networks (PDN)

Description

The PDSN allows a subscriber to be configured for IPv6 PDP contexts. Also, a subscriber may be configured to simultaneously allow IPv4 PDP contexts.

The PDSN supports IPv6 stateless dynamic auto-configuration. The mobile station may select any value for the interface identifier portion of the address. The link-local address is assigned by the PDSN to avoid any conflict between the mobile station link-local address and the PDSN address. The mobile station uses the interface identifier assigned by the PDSN during the stateless address auto-configuration procedure. Once this has completed, the mobile can select any interface identifier for further communication as long as it does not conflict with the PDSN's interface identifier that the mobile learned through router advertisement messages from the PDSN.

Control and configuration of the above is specified as part of the subscriber configuration on the PDSN, e.g., IPv6 address prefix and parameters for the IPv6 router advertisements. RADIUS VSAs may be used to override the subscriber configuration.

Following IPv6 PDP context establishment, the PDSN can perform either manual or automatic 6to4 tunneling, according to RFC 3056, Connection of IPv6 Domains Via IPv4 Clouds.

L2TP LAC Support

The system configured as a Layer 2 Tunneling Protocol Access Concentrator (LAC) enables communication with L2TP Network Servers (LNSs) for the establishment of secure Virtual Private Network (VPN) tunnels between the operator and a subscriber's corporate or home network.

Description

The use of L2TP in VPN networks is often used as it allows the corporation to have more control over authentication and IP address assignment. An operator may do a first level of authentication, however use PPP to exchange user name and password, and use IPCP to request an address. To support PPP negotiation between the PDSN and the corporation, an L2TP tunnel must be setup in the PDSN running a LAC service.

L2TP establishes L2TP control tunnels between LAC and LNS before tunneling the subscriber PPP connections as L2TP sessions. The LAC service is based on the same architecture as the PDSN and benefits from dynamic resource allocation and distributed message and data processing. This design allows the LAC service to support over 4000 setups per second or a maximum of over 3G of throughput. There can be a maximum up to 65535 sessions in a single tunnel and as many as 500,000 L2TP sessions using 32,000 tunnels per system.

The LAC sessions can also be configured to be redundant, thereby mitigating any impact of hardware or software issues. Tunnel state is preserved by copying the information across processor cards.



Important: For more information on L2TP Access Concentrator support, refer to the L2TP Access Concentrator chapter.

L2TP LNS Support

The system configured as a Layer 2 Tunneling Protocol Network Server (LNS) supports the termination secure Virtual Private Network (VPN) tunnels between from L2TP Access Concentrators (LACs).

Description

The LNS service takes advantage of the high performance PPP processing already supported in the system design and is a natural evolution from the LAC. The LNS can be used as a standalone, or running alongside a PDSN service in the same platform, terminating L2TP services in a cost effective and seamless manner.

L2TP establishes L2TP control tunnels between LAC and LNS before tunneling the subscriber PPP connections as L2TP sessions. There can be a maximum of up to 65535 sessions in a single tunnel and up to 500,000 sessions per LNS.

The LNS architecture is similar to the PDSN and utilizes the concept of a de-multiplexer to intelligently assign new L2TP sessions across the available software and hardware resources on the platform without operator intervention..



Important: For more information on L2TP LNS support support, refer to the L2TP Access Concentrator chapter.

Proxy Mobile IP

Mobility for subscriber sessions is provided through the Mobile IP protocol as defined in RFCs 2002-2005. However, some older Mobile Nodes (MNs) do not support the Mobile IP protocol. The Proxy Mobile IP feature provides a mobility solution for these MNs.

Description

For IP PDP contexts using Proxy Mobile IP, the MN establishes a session with the PDSN as it normally would. However, the PDSN/FA performs Mobile IP operations with an HA (identified by information stored in the subscriber's profile) on behalf of the MN (i.e. the MN is only responsible for maintaining the IP PDP context with the PDSN, no Agent Advertisement messages are communicated with the MN).

The MN is assigned an IP address by either the HA, an AAA server, or on a static-basis. The address is stored in a Mobile Binding Record (MBR) stored on the HA. Therefore, as the MN roams through the service provider's network, each time a hand-off occurs, the MN will continue to use the same IP address stored in the MBR on the HA.

Proxy Mobile IP can be performed on a per-subscriber basis based on information contained in their user profile, or for all subscribers facilitated by a specific subscriber. In the case of non-transparent IP PDP contexts, attributes returned from the subscriber's profile take precedence over the configuration of the subscriber.



Important: For more information on Proxy Mobile IP configuration, refer to the Proxy Mobile IP chapter.

IP Security (IPSec)

IP Security provides a mechanism for establishing secure tunnels from mobile subscribers to pre-defined endpoints (i.e. enterprise or home networks) in accordance with the following standards:

- RFC 2401, Security Architecture for the Internet Protocol
- RFC 2402, IP Authentication Header (AH)
- RFC 2406, IP Encapsulating Security Payload (ESP)
- RFC 2409, The Internet Key Exchange (IKE)
- RFC-3193, Securing L2TP using IPSEC, November 2001

Description

IP Security (IPSec) is a suite of protocols that interact with one another to provide secure private communications across IP networks. These protocols allow the system to establish and maintain secure tunnels with peer security gateways. IPSec can be implemented on the system for the following applications:

- PDN Access: Subscriber IP traffic is routed over an IPSec tunnel from the system to a secure gateway on the Packet Data Network (PDN) as determined by Access Control List (ACL) criteria.
- Mobile IP: Mobile IP control signals and subscriber data is encapsulated in IPSec tunnels that are established between Foreign Agents (FAs) and Home Agents (HAs) over the Pi interfaces.

Once an IPSec tunnel is established between an FA and HA for a particular subscriber, all new Mobile IP sessions using the same FA and HA are passed over the tunnel regardless of whether or not IPSec is supported for the new subscriber sessions. Data for existing Mobile IP sessions is unaffected.

- L2TP: L2TP-encapsulated packets are routed from the system to an LNS/secure gateway over an IPSec tunnel.



Important: For more information on IPSec support, refer to the IP Security chapter.

Traffic Policing and Rate Limiting

Allows the operator to proportion the network and support Service-level Agreements (SLAs) for customers

Description

The Traffic-Policing/Shaping feature enables configuring and enforcing bandwidth limitations on individual PDP contexts of a particular 3GPP traffic class. Values for traffic classes are defined in 3GPP TS 23.107 and are negotiated with the SGSN during PDP context activation using the values configured for the subscriber on the PDSN. Configuration and enforcement is done independently on the downlink and the uplink directions for each of the 3GPP traffic classes. Configuration is on a per-subscriber basis, but may be overridden for individual subscribers or subscriber tiers during RADIUS authentication/authorization.

A Token Bucket Algorithm (a modified trTCM, as specified in RFC2698) is used to implement the Traffic-Policing feature. The algorithm measures the following criteria when determining how to mark a packet.

- **Committed Data Rate (CDR):** The guaranteed rate (in bits per second) at which packets may be transmitted/received for the subscriber during the sampling interval.
- **Peak Data Rate (PDR):** The maximum rate (in bits per second) that packets may be transmitted/received for the subscriber during the sampling interval.
- **Burst-size:** The maximum number of bytes that may be transmitted/received for the subscriber during the sampling interval for both committed (CBS) and peak (PBS) rate conditions. This represents the maximum number of tokens that can be placed in the subscriber's "bucket". Note that the committed burst size (CBS) equals the peak burst size (PBS) for each subscriber.

Tokens are removed from the subscriber's bucket based on the size of the packets being transmitted/received. Every time a packet arrives, the system determines how many tokens need to be added (returned) to a subscriber's CBS (and PBS) bucket. This value is derived by computing the product of the time difference between incoming packets and the CDR (or PDR). The computed value is then added to the tokens remaining in the subscriber's CBS (or PBS) bucket. The total number of tokens can not be greater than the configured burst-size. If the total number of tokens is greater than the burst-size, the number is set to equal the burst-size. After passing through the Token Bucket Algorithm, the packet is internally classified with a color, as follows:

- There are not enough tokens in the PBS bucket to allow a packet to pass, then the packet is considered to be in violation and is marked "red" and the violation counter is incremented by one.
- There are enough tokens in the PBS bucket to allow a packet to pass, but not in the CBS "bucket", then the packet is considered to be in excess and is marked "yellow", the PBS bucket is decremented by the packet size, and the exceed counter is incremented by one.
- There are more tokens present in the CBS bucket than the size of the packet, then the packet is considered as conforming and is marked "green" and the CBS and PBS buckets are decremented by the packet size.

The subscriber on the PDSN can be configured with actions to take for red and yellow packets. Any of the following actions may be specified:

- **Drop:** The offending packet is discarded.
- **Transmit:** The offending packet is passed.
- **Lower the IP Precedence:** The packet's ToS octet is set to "0", thus downgrading it to Best Effort, prior to passing the packet.

- **Buffer the Packet:** The packet stored in buffer memory and transmitted to subscriber once traffic flow comes in allowed bandwidth.

Different actions may be specified for red and yellow, as well as for uplink and downlink directions and different 3GPP traffic classes.

Refer to the Intelligent Traffic Control section for additional policing and shaping capabilities of the PDSN.



Important: For more information on per subscriber traffic policing and shaping, refer to the Traffic Policing and Shaping section.

Intelligent Traffic Control

Enables operators to provide differentiated tiered service provisioning for native and non-native subscribers.

Description

Mobile carriers are looking for creative methods for maximizing network resources while, at the same time, enhancing their end users overall experience. These same mobile operators are beginning to examine solutions for providing preferential treatment for their native subscribers and services as compared to, for example, roaming subscribers, Mobile Virtual Network Operators (MVNOs) and/or Peer-to-Peer (P2P) applications. The overall end goal is to provide superior levels of performance for their customers/services, while ensuring that non-native users/applications do not overwhelm network resources.

ITC provides the ability to examine each subscriber session and respective flow(s) such that selective, configurable limits on a per-subscriber/per-flow basis can be applied. Initially, QoS in this context is defined as traffic policing on a per-subscriber/per-flow basis with the potential to manipulate Differentiated Services Code Points (DSCPs), queue redirection (i.e. move traffic to a Best Effort (BE) classification) and/or simply dropping out of profile traffic. ITC enables 5 tuple packet filters for individual application flows to be either manually configured via CLI or dynamically established via RSVP TFT information elements in 1xEV-DO Rev A or as a consequence of PDP context establishments in CDMA networks. Policy rules may be locally assigned or obtained from an external PCRF via push/pull policy signaling interactions. Policies may be applied on a per-subscriber, per-context and/or chassis-wide basis.



Important: For more information on intelligent traffic control support, refer to the Intelligent Traffic Control chapter.

Dynamic RADIUS Extensions (Change of Authorization)

Dynamic RADIUS extension support provide operators with greater control over subscriber PDP contexts by providing the ability to dynamically redirect data traffic, and or disconnect the PDP context.

This functionality is based on the RFC 3576, Dynamic Authorization Extensions to Remote Authentication Dial In User Service (RADIUS), July 2003 standard.

Description

The system supports the configuration and use of the following dynamic RADIUS extensions:

- **Change of Authorization:** The system supports CoA messages from the AAA server to change data filters associated with a subscriber session. The CoA request message from the AAA server must contain attributes to identify NAS and the subscriber session and a data filter ID for the data filter to apply to the subscriber session.
- **Disconnect Message:** The DM message is used to disconnect subscriber sessions in the system from a RADIUS server. The DM request message should contain necessary attributes to identify the subscriber session.

The above extensions can be used to dynamically re-direct subscriber PDP contexts to an alternate address for performing functions such as provisioning and/or account set up. This functionality is referred to as Session Redirection, or Hotlining.

Session redirection provides a means to redirect subscriber traffic to an external server by applying ACL rules to the traffic of an existing or a new subscriber session. The destination address and optionally the destination port of TCP/IP or UDP/IP packets from the subscriber are rewritten so the packet is forwarded to the designated redirected address.

Return traffic to the subscriber has the source address and port rewritten to the original values. The redirect ACL may be applied dynamically by means of the Radius Change of Authorization (CoA) extension.



Important: For more information on dynamic RADIUS extensions support, refer to the CoA, RADIUS, And Session Redirection (Hotlining) chapter.

Web Element Management System

Provides a Graphical User Interface (GUI) for performing Fault, Configuration, Accounting, Performance, and Security (FCAPS) management of the ASR 5x00 system.

Description

The Web Element Manager is a Common Object Request Broker Architecture (CORBA)-based application that provides complete Fault, Configuration, Accounting, Performance, and Security (FCAPS) management capability for the system.

For maximum flexibility and scalability, the Web Element Manager application implements a client-server architecture. This architecture allows remote clients with Java-enabled web browsers to manage one or more systems via the server component which implements the CORBA interfaces. The server component is fully compatible with the fault-tolerant Sun® Solaris® operating system.



Important: For more information on WEM support, refer to the *WEM Installation and Administration Guide*.

Features and Functionality - Inline Service Support

This section describes the features and functions of inline services supported on the PDSN. These services require additional licenses to implement the functionality.

Content Filtering

The Cisco PDSN offers two variants of network-controlled content filtering / parental control services. Each approach leverages the native DPI capabilities of the platform to detect and filter events of interest from mobile subscribers based on HTTP URL or WAP/MMS URI requests:

- **Integrated Content Filtering:** A turnkey solution featuring a policy enforcement point and category based rating database on the Cisco PDSN. An offboard AAA or PCRF provides the per-subscriber content filtering information as subscriber sessions are established. The content filtering service uses DPI to extract URL's or URI's in HTTP request messages and compares them against a static rating database to determine the category match. The provisioned policy determines whether individual subscribers are entitled to view the content.
- **Content Filtering ICAP Interface:** This solution is appropriate for mobile operators with existing installations of Active Content Filtering external servers. The service continues to harness the DPI functions of the ASR 5000 platform to extract events of interest. However in this case, the extracted requests are transferred via the Integrated Content Adaptation Protocol (ICAP) with subscriber identification information to the external ACF server which provides the category rating database and content decision functions.

Integrated Adult Content Filter

Provides a value-added service to prevent unintended viewing of objectionable content that exploits underage children. Content Filtering offers mobile operators a way to increase data ARPU and subscriber retention through a network-based solution for parental controls and content filtering. The integrated solution enables a single policy decision and enforcement point thereby streamlining the number of signaling interactions with external AAA/Policy Manager servers. When used in parallel with other services such as Enhanced Content Charging (ECS) it increases billing accuracy of charging records by insuring that mobile subscribers are only charged for visited sites they are allowed to access.

The Integrated Adult Content Filter is a subscriber-aware inline service provisioned on an ASR 5000 running PDSN services. Integrated Content Filtering utilizes the local DPI engine and harnesses a distributed software architecture that scales with the number of active PDSN sessions on the system.

Content Filtering policy enforcement is the process of deciding if a subscriber should be able to receive some content. Typical options are to allow, block, or replace/redirect the content based on the rating of the content and the policy defined for that content and subscriber. The policy definition is transferred in an authentication response from a AAA server or Diameter policy message via the Gx reference interface from an adjunct PCRF. The policy is applied to subscribers through rulebase or APN/Subscriber configuration. The policy determines the action to be taken on the content request on the basis of its category. A maximum of one policy can be associated with a rulebase.

ICAP Interface

Provides a value-added service to prevent unintended viewing of objectionable content that exploits underage children. Content Filtering offers mobile operators a way to increase data ARPU and subscriber retention through a network-based solution for parental controls and content filtering. The Content Filtering ICAP solution is appropriate for operators with existing installations of Active Content Filtering servers in their networks.

The Enhanced Charging Service (ECS) provides a streamlined Internet Content Adaptation Protocol (ICAP) interface to leverage the Deep Packet Inspection (DPI) to enable external Application Servers to provide their services without performing the DPI functionality and without being inserted in the data flow. The ICAP interface may be attractive to mobile operators that prefer to use an external Active Content Filtering (ACF) Platform. If a subscriber initiates a WAP (WAP1.x or WAP2.0) or Web session, the subsequent GET/POST request is detected by the deep packet inspection function. The URL of the GET/POST request is extracted by the local DPI engine on the ASR 5000 platform and passed, along with subscriber identification information and the subscriber request, in an ICAP message to the Application Server (AS). The AS checks the URL on the basis of its category and other classifications like, type, access level, content category and decides if the request should be authorized, blocked or redirected by answering the GET/POST message. Depending upon the response received from the ACF server, the PDSN either passes the request unmodified or discards the message and responds to the subscriber with the appropriate redirection or block message.

Network Address Translation (NAT)

NAT translates non-routable private IP address(es) to routable public IP address(es) from a pool of public IP addresses that have been designated for NAT. This enables to conserve on the number of public IP addresses required to communicate with external networks, and ensures security as the IP address scheme for the internal network is masked from external hosts, and each outgoing and incoming packet goes through the translation process.

NAT works by inspecting both incoming and outgoing IP datagrams and, as needed, modifying the source IP address and port number in the IP header to reflect the configured NAT address mapping for outgoing datagrams. The reverse NAT translation is applied to incoming datagrams.

NAT can be used to perform address translation for simple IP and mobile IP. NAT can be selectively applied/denied to different flows (5-tuple connections) originating from subscribers based on the flows' L3/L4 characteristics—Source-IP, Source-Port, Destination-IP, Destination-Port, and Protocol.

NAT supports the following mappings:

- One-to-One
- Many-to-One



Important: For more information on NAT, refer to the *Cisco ASR 5000 Series Network Address Translation Administration Guide*.

Peer-to-Peer Detection

Allows operators to identify P2P traffic in the network and applying appropriate controlling functions to ensure fair distribution of bandwidth to all subscribers.

Peer-to-Peer (P2P) is a term used in two slightly different contexts. At a functional level, it means protocols that interact in a peering manner, in contrast to client-server manner. There is no clear differentiation between the function of one node or another. Any node can function as a client, a server, or both—a protocol may not clearly differentiate between the two. For example, peering exchanges may simultaneously include client and server functionality, sending and receiving information.

Detecting peer-to-peer protocols requires recognizing, in real time, some uniquely identifying characteristic of the protocols. Typical packet classification only requires information uniquely typed in the packet header of packets of the stream(s) running the particular protocol to be identified. In fact, many peer-to-peer protocols can be detected by simple packet header inspection. However, some P2P protocols are different, preventing detection in the traditional manner. This is designed into some P2P protocols to purposely avoid detection. The creators of these protocols purposely do not

publish specifications. A small class of P2P protocols is stealthier and more challenging to detect. For some protocols no set of fixed markers can be identified with confidence as unique to the protocol.

Operators care about P2P traffic because of the behavior of some P2P applications (for example, Bittorrent, Skype, and eDonkey). Most P2P applications can hog the network bandwidth such that 20% P2P users can generate as much as traffic generated by the rest 80% non-P2P users. This can result into a situation where non-P2P users may not get enough network bandwidth for their legitimate use because of excess usage of bandwidth by the P2P users. Network operators need to have dynamic network bandwidth / traffic management functions in place to ensure fair distributions of the network bandwidth among all the users. And this would include identifying P2P traffic in the network and applying appropriate controlling functions to the same (for example, content-based premium billing, QoS modifications, and other similar treatments).

Cisco's P2P detection technology makes use of innovative and highly accurate protocol behavioral detection techniques.



Important: For more information on peer-to-peer detection, refer to the *Cisco ASR 5000 Series Application Detection and Control Administration Guide*.

Personal Stateful Firewall

The Personal Stateful Firewall is an in-line service feature that inspects subscriber traffic and performs IP session-based access control of individual subscriber sessions to protect the subscribers from malicious security attacks.

The Personal Stateful Firewall supports stateless and stateful inspection and filtering based on the configuration.

In stateless inspection, the firewall inspects a packet to determine the 5-tuple—source and destination IP addresses and ports, and protocol—information contained in the packet. This static information is then compared against configurable rules to determine whether to allow or drop the packet. In stateless inspection the firewall examines each packet individually, it is unaware of the packets that have passed through before it, and has no way of knowing if any given packet is part of an existing connection, is trying to establish a new connection, or is a rogue packet.

In stateful inspection, the firewall not only inspects packets up through the application layer / layer 7 determining a packet's header information and data content, but also monitors and keeps track of the connection's state. For all active connections traversing the firewall, the state information, which may include IP addresses and ports involved, the sequence numbers and acknowledgement numbers of the packets traversing the connection, TCP packet flags, etc. is maintained in a state table. Filtering decisions are based not only on rules but also on the connection state established by prior packets on that connection. This enables to prevent a variety of DoS, DDoS, and other security violations. Once a connection is torn down, or is timed out, its entry in the state table is discarded.

The Enhanced Charging Service (ECS) / Active Charging Service (ACS) in-line service is the primary vehicle that performs packet inspection and charging. For more information on ECS, see the *Cisco ASR 5000 Series Enhanced Charging Service Administration Guide*.



Important: For more information on Personal Stateful Firewall, refer to the *Cisco ASR 5000 Series Personal Stateful Firewall Administration Guide*.

Traffic Performance Optimization (TPO)

Though TCP is a widely accepted protocol in use today, it is optimized only for wired networks. Due to inherent reliability of wired networks, TCP implicitly assumes that any packet loss is due to network congestion and consequently invokes congestion control measures. However, wireless links are known to experience sporadic and

usually temporary losses due to several reasons, including the following, which also trigger TCP congestion control measures resulting in poor TCP performance.

Reasons for delay variability over wireless links include:

- Channel fading effect, subscriber mobility, and other transient conditions
- Link-layer retransmissions
- Handoffs between neighboring cells
- Intermediate nodes, such as SGSN and e-NodeB, implementing scheduling policies tuned to deliver better QoS for select services; resulting in variable delay in packet delivery for other services

The TPO inline service uses a combination of TCP and HTTP optimization techniques to improve TCP performance over wireless links.



Important: For more information on TPO, refer to the *Cisco ASR 5000 Series Traffic Performance Optimization Administration Guide*.

Features and Functionality - External Application Support

This section describes the features and functions of external applications supported on the PDSN. These services require additional licenses to implement the functionality.

Mobility Unified Reporting

The Cisco Mobility Unified Reporting (MUR) system is a Web-based application providing a unified reporting interface for diverse data from Cisco Systems In-line service and storage applications.

The MUR application provides comprehensive and consistent set of statistics and customized reports, report scheduling and distribution from ASR chassis / in-line service product. For example, a subscriber's Quality of Experience, top 10 sites visited, top 10 users, and so on.

The MUR application provides reporting capability for Content Filtering (CF) data, bulk statistics, Key Performance Indicators (KPIs), EDRs data from in-line service and storage applications. The MUR application facilitates and enhances the operators' ability to simply and easily determine the health and usage of the network.



Important: For more information on MUR support, refer to the *MUR Installation and Administration Guide*.

CDMA2000 Data Network Deployment Configurations

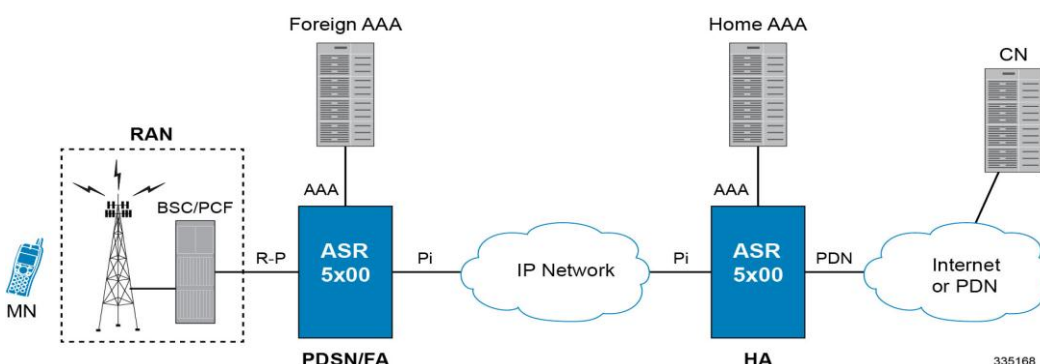
This section provides examples of how the system can be deployed within a wireless carrier's network. As noted previously in this chapter, the system can be deployed in standalone configurations, serving as a Packet Data Serving Node/Foreign Agent (PDSN/FA), a Home Agent (HA), or in a combined PDSN/FA/HA configuration providing all services from a single chassis. Although XT-2 systems are highly flexible, but XT-2 systems are pre-loaded with purchased services and operator can not add additional services through license. Operator needs to predefine the services required on a system.

Standalone PDSN/FA and HA Deployments

The PDSN/FA serves as an integral part of a CDMA2000 network by providing the packet processing and re-direction to the mobile user's home network through communications with the HA. In cases where the mobile user connects to a PDSN that serves their home network, no re-direction is required.

The following figure depicts a sample network configuration wherein the PDSN/FA and HA are separate systems.

Figure 1. PDSN/FA and HA Network Deployment Configuration Example



The HA allows mobile nodes to be reached, or served, by their home network through its home address even when the mobile node is not attached to its home network. The HA performs this function through interaction with an FA that the mobile node is communicating with using the Mobile IP protocol. Such transactions are performed through the use of virtual private networks that create Mobile IP tunnels between the HA and FA.

Interface Descriptions

This section describes the primary interfaces used in a CDMA2000 wireless data network deployment.

R-P Interface

This interface exists between the Packet Control Function (PCF) and the PDSN/FA and implements the A10 and A11 (data and bearer signaling respectively) protocols defined in 3GPP2 specifications.

The PCF can be co-located with the Base Station Controller (BSC) as part of the Radio Access Node (RAN). The PDSN/FA is connected to the RAN via Ethernet ports. These ports also support outbound IP traffic that carries user data to the HA for Mobile IP services, or to the Internet or Wireless Access Protocol (WAP) gateway for Simple IP services.

Pi Interfaces

The Pi interface provides connectivity between the HA and its corresponding FA. The Pi interface is used to establish a Mobile IP tunnels between the PDSN/FA and HA.


PDN Interfaces

PDN interface provide connectivity between the PDSN and/or HA to packet data networks such as the Internet or a corporate intranet.

AAA Interfaces

Using the LAN ports located on the Switch Processor I/O (SPIO) and Ethernet line cards, these interfaces carry AAA messages to and from RADIUS accounting and authentication servers. The SPIO supports RADIUS-capable management interfaces using either copper or fiber Ethernet connectivity through two auto-sensing 10/100/1000 Mbps Ethernet interfaces or two SFP optical gigabit Ethernet interfaces. User-based RADIUS messaging is transported using the Ethernet line cards.

While most carriers will configure separate AAA interfaces to allow for out-of-band RADIUS messaging for system administrative users and other operations personnel, it is possible to use a single AAA interface hosted on the Ethernet line cards to support a single RADIUS server that supports both management users and network users.

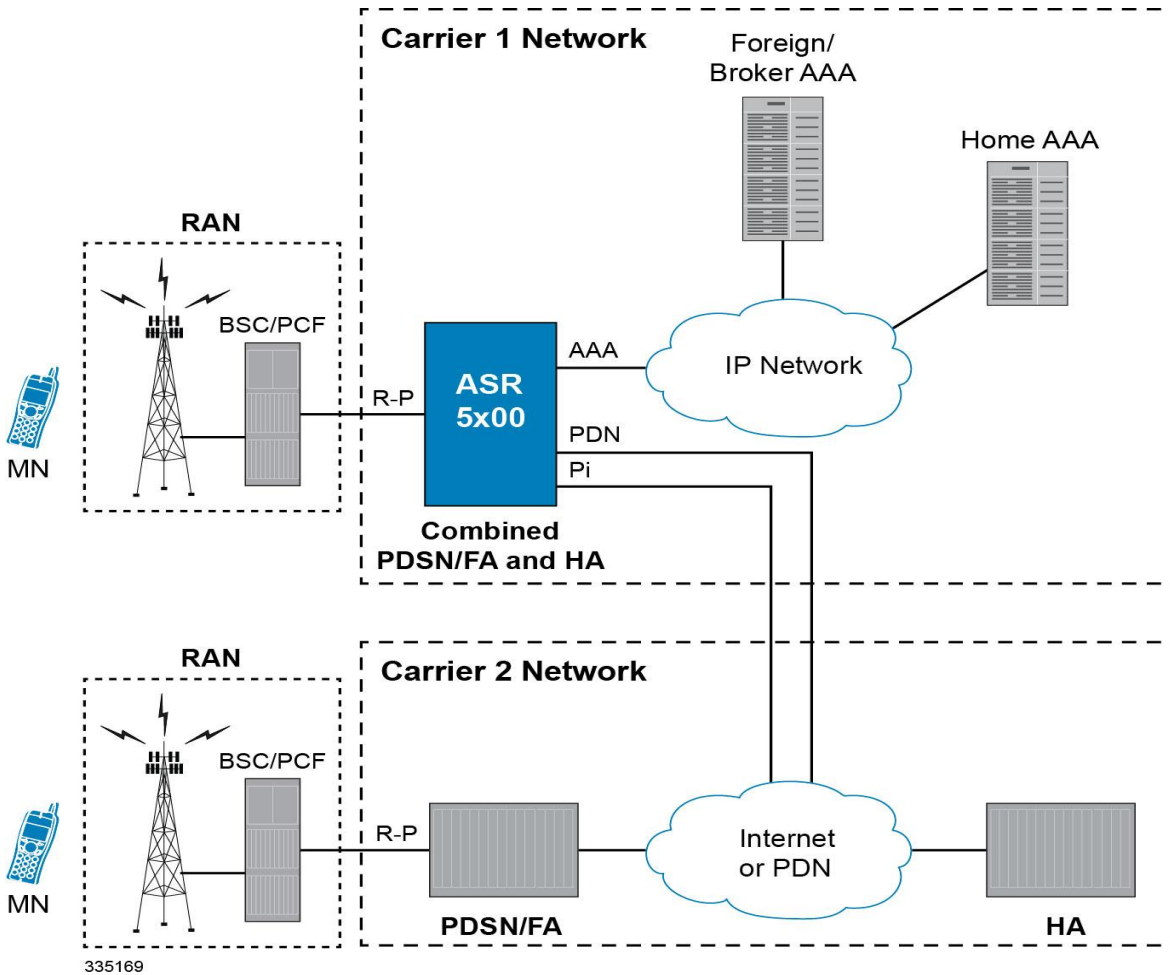
 **Important:** Subscriber AAA interfaces should always be configured using Ethernet line card interfaces for the highest performance. The out-of-band local context should not be used for service subscriber AAA functions.

Co-Located Deployments

An advantage of the system is its ability to support both high-density PDSN/FA and HA configurations within the same chassis. The economies of scale presented in this configuration example provide for both improved session handling and reduced cost in deploying a CDMA2000 data network.

The following figure depicts a sample co-located deployment.

Figure 2. Co-located PDSN/FA and HA Configuration Example



It should be noted that all interfaces defined within the 3GPP2 standards for 1x deployments exist in this configuration as they are described in the two previous sections. This configuration can support communications to external, or standalone, PDSNs/FAs and/or HAs using all prescribed standards.

Understanding Simple IP and Mobile IP

From a mobile subscriber's perspective, packet data services are delivered from the service provider network using two access methods:

- Local and public network access
- Private network access

Within the packet data network, access is similar to accessing the public Internet through any other access device. In a private network access scenario, the user must be tunneled into the private network after initial authentication has been performed.

These two methods are provided using one of the following access applications:

- **Simple IP:** The mobile user is dynamically assigned an IP address from the service provider. The user can maintain this address within a defined geographical area, but when the user moves outside of this area, their IP address will be lost. This means that whenever a mobile user moves to a new location, they will need to re-register with the service provider to obtain a new IP address.
- **Mobile IP:** The mobile subscriber uses either a static or dynamically assigned IP address that belongs to their home network. As the subscriber roams through the network, the IP address is maintained providing the subscriber with the opportunity to use IP applications that require seamless mobility such as performing file transfers.
- **Proxy Mobile IP:** Provides a mobility solution for subscribers whose Mobile Nodes (MNs) do not support the Mobile IP protocol. The PDSN/FA proxy the Mobile IP tunnel with the HA on behalf of the MS. The subscriber receives an IP address from either the service provider or from their home network. As the subscriber roams through the network, the IP address is maintained providing the subscriber with the opportunity to use IP applications that require seamless mobility such as transferring files.

The following sections outline both Simple IP, Mobile IP, and Proxy Mobile IP and how they work in a 3G network.

Simple IP

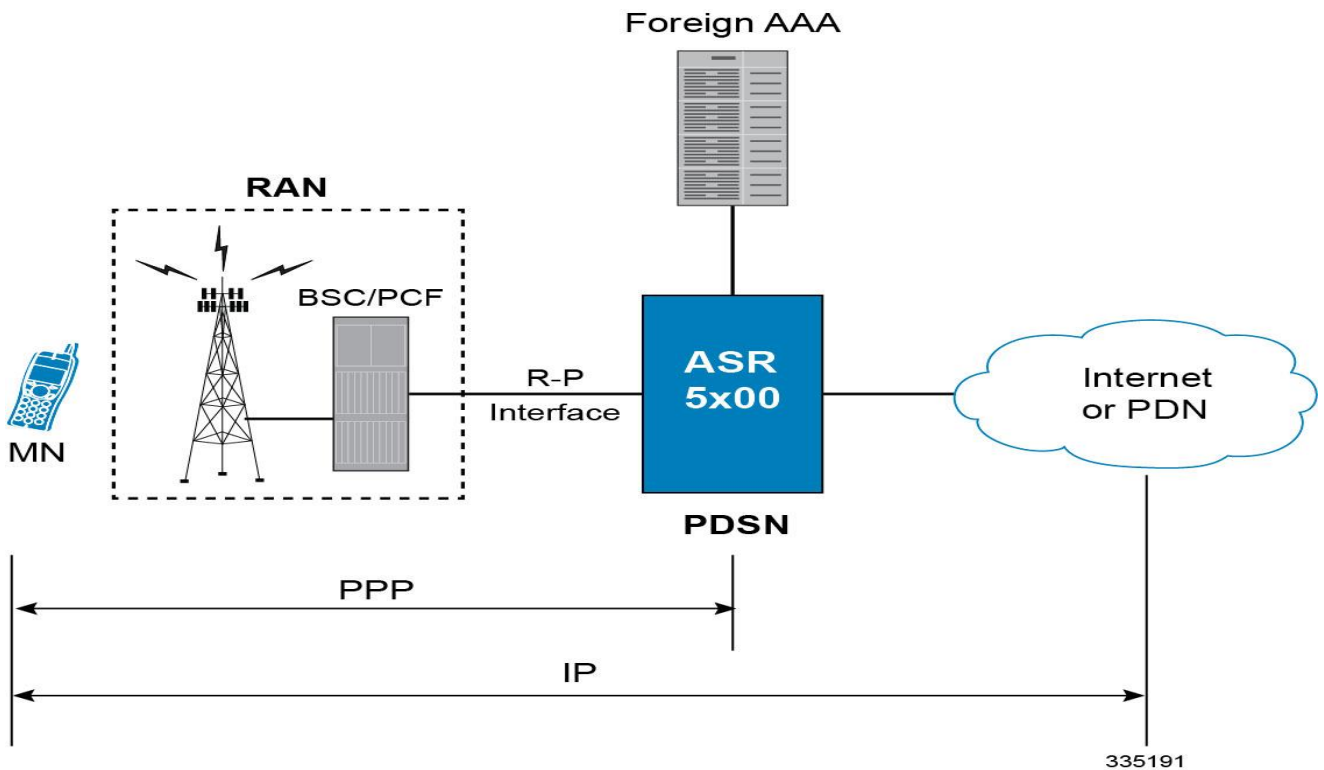
From a packet data perspective, Simple IP is similar to how a dial-up user would connect to the Internet using the Point-to-Point Protocol (PPP) and the Internet Protocol (IP) through an Internet Service Provider (ISP). With Simple IP, the mobile user is assigned a dynamic IP address from a PDSN or AAA server that is serving them locally (a specific geographic area). Once the mobile user is connected to the particular radio network that the assigning PDSN belongs to, an IP address is assigned to the mobile node. The PDSN provides IP routing services to the registered mobile user through the wireless service provider's network.

There is no mobility beyond the PDSN that assigns the dynamic IP address to the mobile user, which means that should the mobile user leave the geographic area where service was established (moves to a new radio network service area), they will need to obtain a new IP address with a new PDSN that is serving the new area. This new connection may or may not be provided by the same service provider.

How Simple IP Works

As described earlier, Simple IP uses two basic communications protocols, PPP and IP. The following figure depicts where each of these protocols are used in a Simple IP call.

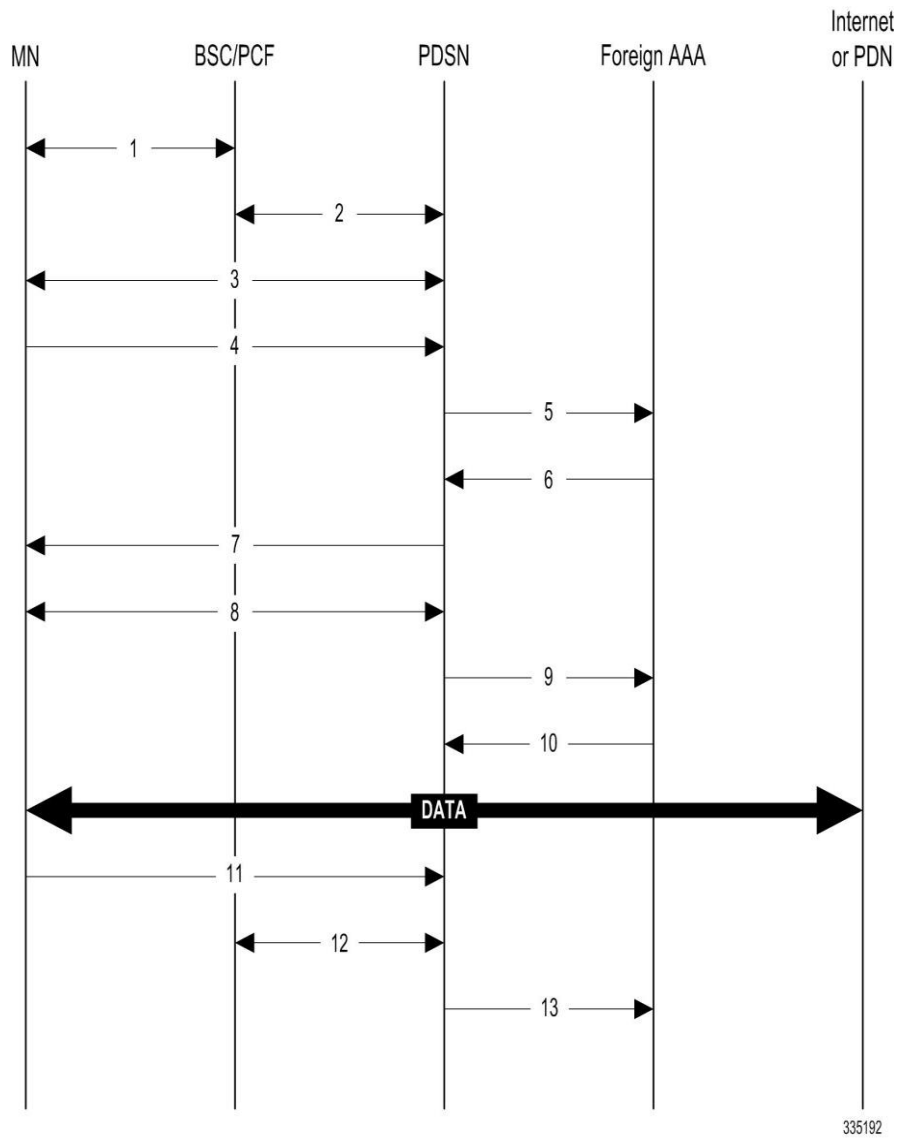
Figure 3. Simple IP Protocol Usage



As depicted in the figure above, PPP is used to establish a communications session between the MN and the PDSN. Once a PPP session is established, the Mobile Node (MN) and end host communicate using IP packets.

The following figure and table provides a high-level view of the steps required to make a Simple IP call that is initiated by the MN to an end host. Users should keep in mind that steps 2, 3, 11, and 12 in the call flow are related to the Radio Access Node (RAN) functions and are intended to show a high-level overview of radio communications iterations, and as such are outside the scope of packet-based communications presented here.

Figure 4. Simple IP Call Flow



335192

Table 1. Simple IP Call Flow Description

Step	Description
1	Mobile Node (MN) secures a traffic channel over the airlink with the RAN through the BSC/PCF.
2	The PCF and PDSN establish the R-P interface for the session.
3	The PDSN and MN negotiate Link Control Protocol (LCP).
4	Upon successful LCP negotiation, the MN sends a PPP Authentication Request message to the PDSN.
5	The PDSN sends an Access Request message to the RADIUS AAA server.

Step	Description
6	The RADIUS AAA server successfully authenticates the subscriber and returns an Access Accept message to the PDSN. The Accept message may contain various attributes to be assigned to the MN.
7	The PDSN sends a PPP Authentication Response message to the MN.
8	The MN and the PDSN negotiate the Internet Protocol Control Protocol (IPCP) that results in the MN receiving an IP address.
9	The PDSN forwards a RADIUS Accounting Start message to the AAA server fully establishing the session allowing the MN to send/receive data to/from the PDN.
10	Upon completion of the session, the MN sends an LCP Terminate Request message to the PDSN to end the PPP session.
11	The BSC closes the radio link while the PCF closes the R-P session between it and the PDSN. All PDSN resources used to facilitate the session are reclaimed (IP address, memory, etc.).
12	The PDSN sends accounting stop record to the AAA server, ending the session.

Mobile IP

Mobile IP provides a network-layer solution that allows mobile nodes (MNs, i.e. mobile phones, wireless PDAs, and other mobile devices) to receive routed IP packets from their home network while they are connected to any visitor network using their permanent or home IP address. Mobile IP allows mobility in a dynamic method that allows nodes to maintain ongoing communications while changing links as the user traverses the global Internet from various locations outside their home network.

In Mobile IP, the Mobile Node (MN) receives an IP address, either static or dynamic, called the “home address” assigned by its Home Agent (HA). A distinct advantage with Mobile IP is that MNs can hand off between different radio networks that are served by different PDSNs.

In this scenario, the PDSN in the visitor network performs as a Foreign Agent (FA), establishing a virtual session with the MN's HA. Each time the MN registers with a different PDSN/FA, the FA assigns the MN a care-of-address. Packets are then encapsulated into IP tunnels and transported between FA, HA, and the MN.

Mobile IP Tunneling Methods

Tunneling by itself is a technology that enables one network to send its data via another network's connections. Tunneling works by encapsulating a network protocol within a packet, carried by the second network. Tunneling is also called encapsulation. Service providers typically use tunneling for two purposes; first, to transport otherwise un-routable packets across the IP network and second, to provide data separation for Virtual Private Networking (VPN) services. In Mobile IP, tunnels are used to transport data packets between the FA and HA.

The system supports the following tunneling protocols, as defined in the IS-835-A specification and the relevant Request For Comments (RFCs) for Mobile IP:

IP in IP tunnels


IP in IP tunnels basically encapsulate one IP packet within another using a simple encapsulation technique. To encapsulate an IP datagram using IP in IP encapsulation, an outer IP header is inserted before the datagram's existing IP header. Between them are other headers for the path, such as security headers specific to the tunnel configuration. Each header chains to the next using IP Protocol values. The outer IP header Source and Destination identify the “endpoints” of the tunnel. The inner IP header Source and Destination identify the original sender and recipient of the datagram,

while the inner IP header is not changed by the encapsulator, except to decrement the TTL, and remains unchanged during its delivery to the tunnel exit point. No change to IP options in the inner header occurs during delivery of the encapsulated datagram through the tunnel. If needed, other protocol headers such as the IP Authentication header may be inserted between the outer IP header and the inner IP header.

The Mobile IP working group has specified the use of encapsulation as a way to deliver datagrams from an MN's HA to an FA, and conversely from an FA to an HA, that can deliver the data locally to the MN at its current location.

GRE tunnels

The Generic Routing Encapsulation (GRE) protocol performs encapsulation of IP packets for transport across disparate networks. One advantage of GRE over earlier tunneling protocols is that any transport protocol can be encapsulated in GRE. GRE is a simple, low overhead approach—the GRE protocol itself can be expressed in as few as eight octets as there is no authentication or tunnel configuration parameter negotiation. GRE is also known as IP Protocol 47.

 **Important:** The chassis simultaneously supports GRE protocols with key in accordance with RFC-1701/RFC-2784 and “Legacy” GRE protocols without key in accordance to RFC-2002.

Another advantage of GRE tunneling over IP-in-IP tunneling is that GRE tunneling can be used even when conflicting addresses are in use across multiple contexts (for the tunneled data).

Communications between the FA and HA can be done in either the forward or reverse direction using the above protocols. Additionally, another method of routing information between the FA and various content servers used by the HA exists. This method is called Triangular Routing. Each of these methods is explained below.

Forward Tunneling

In the wireless IP world, forward tunneling is a tunnel that transports packets from the packet data network towards the MN. It starts at the HA and ends at the MN's care-of address. Tunnels can be as simple as IP-in-IP tunnels, GRE tunnels, or even IP Security (IPSec) tunnels with encryption. These tunnels can be started automatically, and are selected based on the subscriber's user profile.

Reverse Tunneling

A reverse tunnel starts at the MN's care-of address, which is the FA, and terminates at the HA.

When an MN arrives at a foreign network, it listens for agent advertisements and selects an FA that supports reverse tunnels. The MN requests this service when it registers through the selected FA. At this time, the MN may also specify a delivery technique such as Direct or the Encapsulating Delivery Style.

Using the Direct Delivery Style, which is the default mode for the system, the MN designates the FA as its default router and sends packets directly to the FA without encapsulation. The FA intercepts them, and tunnels them to the HA.

Using the Encapsulating Delivery Style, the MN encapsulates all its outgoing packets to the FA. The FA then de-encapsulates and re-tunnels them to the HA, using the FA's care-of address as the entry-point for this new tunnel.

Following are some of the advantages of reverse tunneling:

- All datagrams from the mobile node seem to originate from its home network
- The FA can keep track of the HA that the mobile node is registered to and tunnel all datagrams from the mobile node to its HA

Triangular Routing

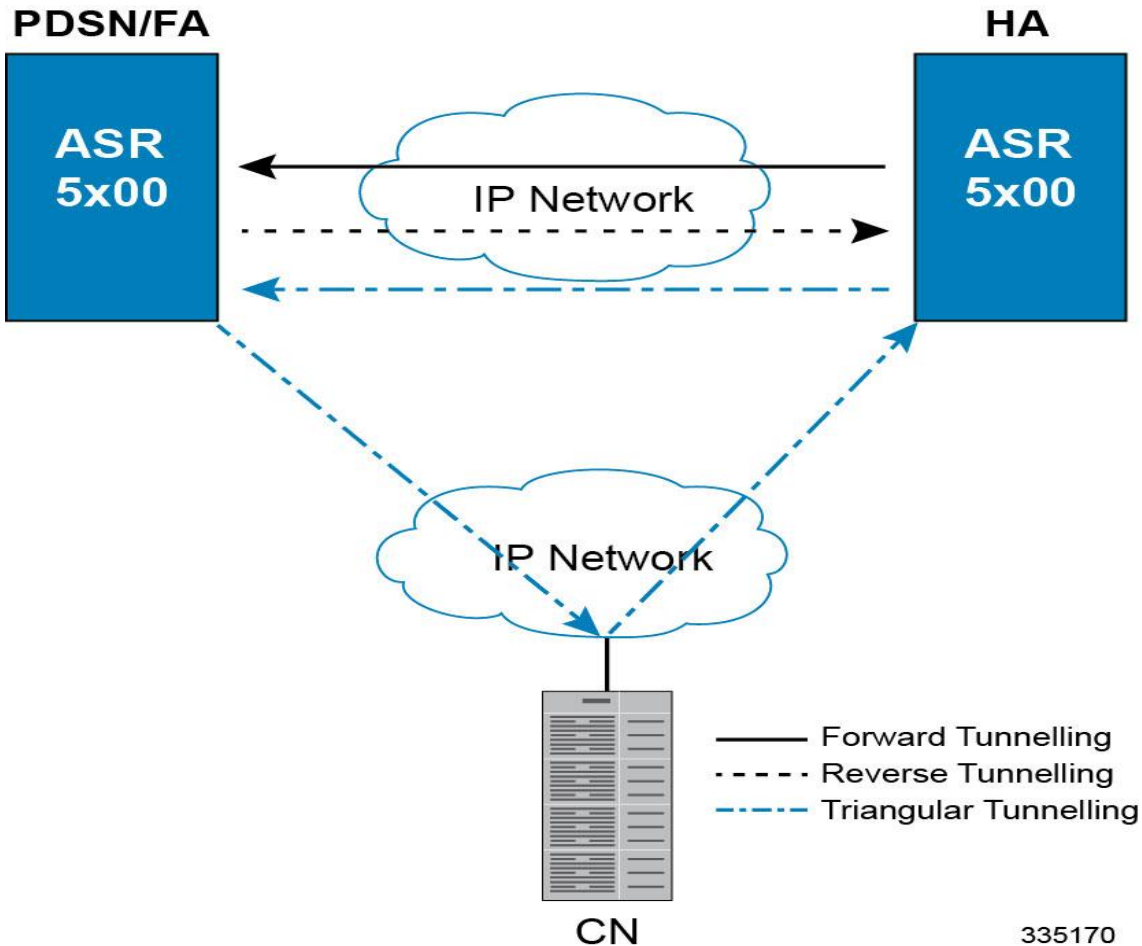
Triangular routing is the path followed by a packet from the MN to the Correspondent Node (CN) via the FA. In this routing scenario, the HA receives all the packets destined to the MN from the CN and redirects them to the MN's care-of-address by forward tunneling. In this case, the MN sends packets to the FA, which are transported using conventional IP routing methods.

A key advantage of triangular routing is that reverse tunneling is not required, eliminating the need to encapsulate and de-capsulate packets a second time during a Mobile IP session since only a forward tunnel exists between the HA and PDSN/FA.

A disadvantage of using triangular routing is that the HA is unaware of all user traffic for billing purposes. Also, both the HA and FA are required to be connected to a private network. This can be especially troublesome in large networks, serving numerous enterprise customers, as each FA would have to be connected to each private network.

The following figure shows an example of how triangular routing is performed.

Figure 5. Mobile IP, FA and HA Tunneling/Transport Methods

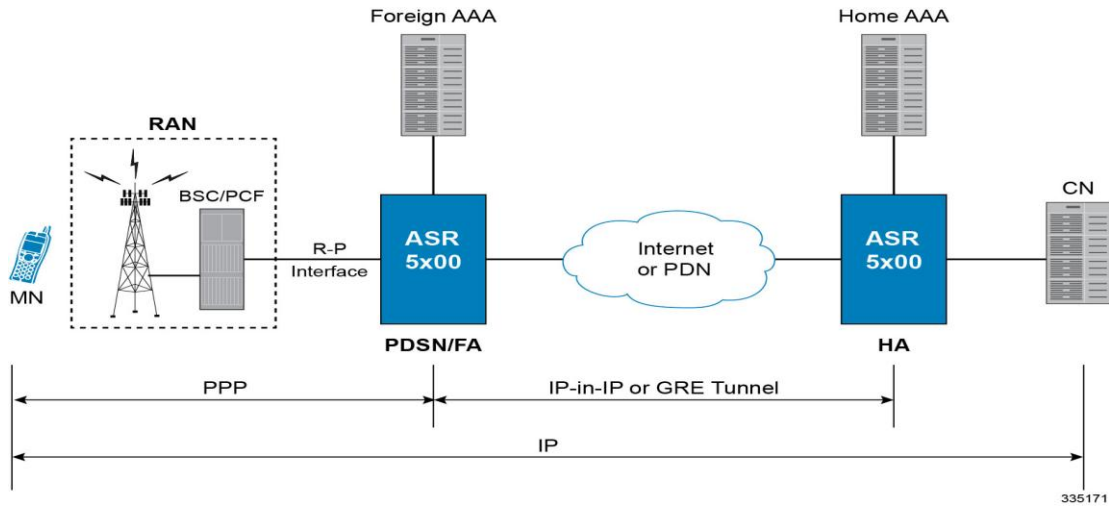


335170

How Mobile IP Works

As described earlier, Mobile IP uses three basic communications protocols; PPP, IP, and Tunneled IP in the form of IP-in-IP or GRE tunnels. The following figure depicts where each of these protocols are used in a basic Mobile IP call.

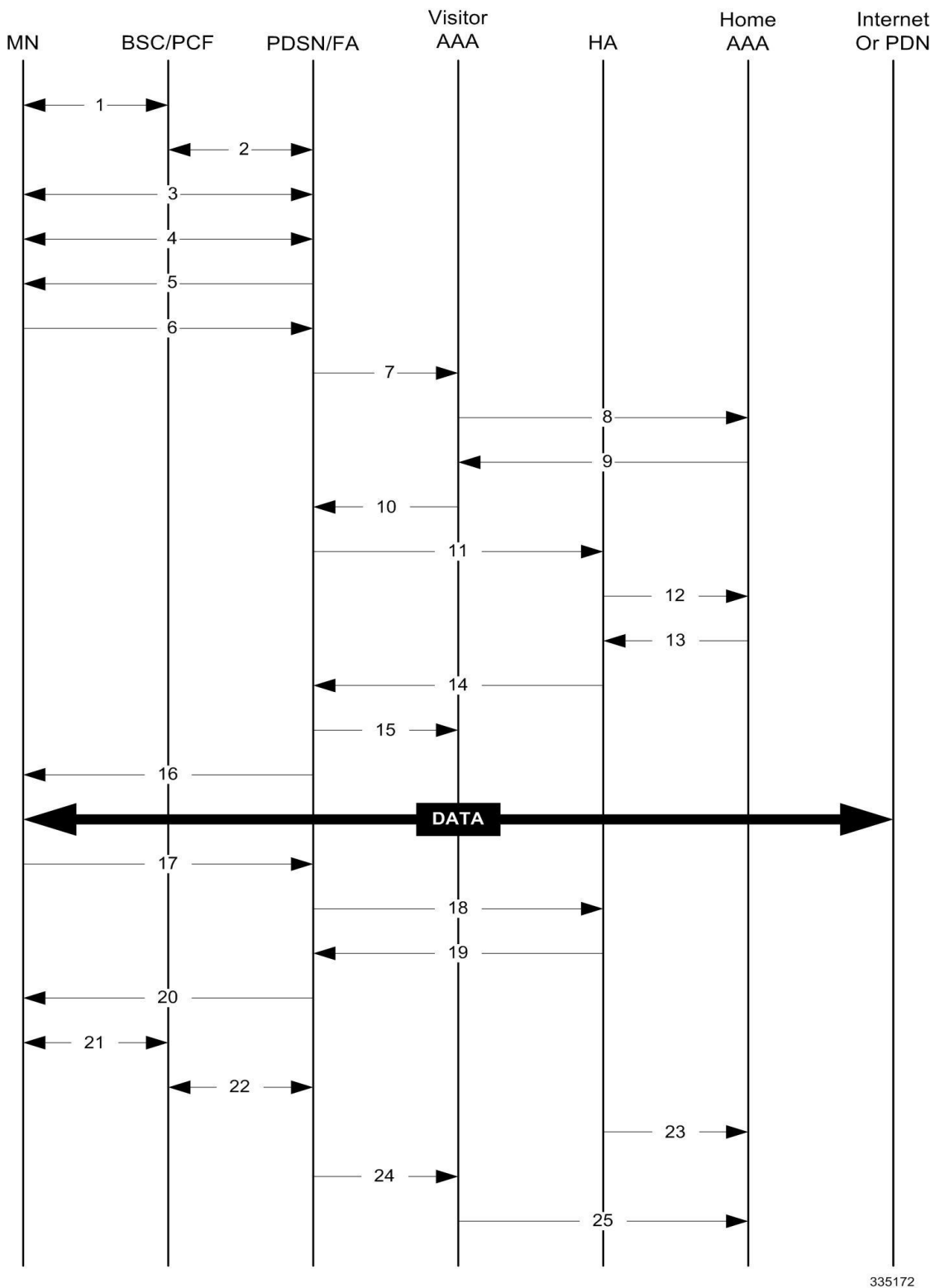
Figure 6. Mobile IP Protocol Usage



As depicted in the figure above, PPP is used to establish a communications session between the MN and the FA. Once a PPP session is established, the MN can communicate with the HA, using the FA as a mediator or broker. Data transport between the FA and HA use tunneled IP, either IP-in-IP or GRE tunneling. Communication between the HA and End Host can be achieved using the Internet or a private IP network and can use any IP protocol.

The following figure provides a high-level view of the steps required to make a Mobile IP call that is initiated by the MN to a HA and table that follows, explains each step in detail. Users should keep in mind that steps in the call flow related to the Radio Access Node (RAN) functions are intended to show a high-level overview of radio communications iterations, and as such are outside the scope of packet-based communications presented here.

Figure 7. Mobile IP Call Flow



335172

Table 2. Mobile IP Call Flow Description

Step	Description
1	Mobile Node (MN) secures a traffic channel over the airlink with the RAN through the BSC/PCF.
2	The PCF and PDSN establish the R-P interface for the session.
3	The PDSN and MN negotiate Link Control Protocol (LCP).
4	The PDSN and MN negotiate the Internet Protocol Control Protocol (IPCP).
5	The PDSN/FA sends an Agent Advertisement to the MN.
6	The MN sends a Mobile IP Registration Request to the PDSN/FA.
7	The PDSN/FA sends an Access Request message to the visitor AAA server.
8	The visitor AAA server proxies the request to the appropriate home AAA server.
9	The home AAA server sends an Access Accept message to the visitor AAA server.
10	The visitor AAA server forwards the response to the PDSN/FA.
11	Upon receipt of the response, the PDSN/FA forwards a Mobile IP Registration Request to the appropriate HA.
12	The HA sends an Access Request message to the home AAA server to authenticate the MN/subscriber.
13	The home AAA server returns an Access Accept message to the HA.
14	Upon receiving response from home AAA, the HA sends a reply to the PDSN/FA establishing a forward tunnel. Note that the reply includes a Home Address (an IP address) for the MN.
15	The PDSN/FA sends an Accounting Start message to the visitor AAA server. The visitor AAA server proxies messages to the home AAA server as needed.
16	The PDSN return a Mobile IP Registration Reply to the MN establishing the session allowing the MN to send/receive data to/from the PDN.
17	Upon session completion, the MN sends a Registration Request message to the PDSN/FA with a requested lifetime of 0.
18	The PDSN/FA forwards the request to the HA.
19	The HA sends a Registration Reply to the PDSN/FA accepting the request.
20	The PDSN/FA forwards the response to the MN.
21	The MN and PDSN/FA negotiate the termination of LCP effectively ending the PPP session.
22	The PCF and PDSN/FA close terminate the R-P session.
23	The HA sends an Accounting Stop message to the home AAA server.
24	The PDSN/FA sends an Accounting Stop message to the visitor AAA server.
25	The visitor AAA server proxies the accounting data to the home AAA server.

Proxy Mobile IP

Proxy Mobile IP provides mobility for subscribers with MNs that do not support the Mobile IP protocol stack.

For subscriber sessions using Proxy Mobile IP, R-P and PPP sessions get established as they would for a Simple IP session. However, the PDSN/FA performs Mobile IP operations with an HA (identified by information stored in the subscriber's profile) on behalf of the MN while the MN performs only Simple IP processes. The protocol details are similar to those displayed in figure earlier for Mobile IP.

The MN is assigned an IP address by either the PDSN/FA or the HA. Regardless of its source, the address is stored in a Mobile Binding Record (MBR) stored on the HA. Therefore, as the MN roams through the service provider's network, each time a hand-off occurs, the MN will receive the same IP address stored in the MBR on the HA.

Note that unlike Mobile IP-capable MNs that can perform multiple sessions over a single PPP link, Proxy Mobile IP allows only a single session over the PPP link. In addition, simultaneous Mobile and Simple IP sessions will not be supported for an MN by an FA currently facilitating a Proxy Mobile IP session for the MN.

How Proxy Mobile IP Works

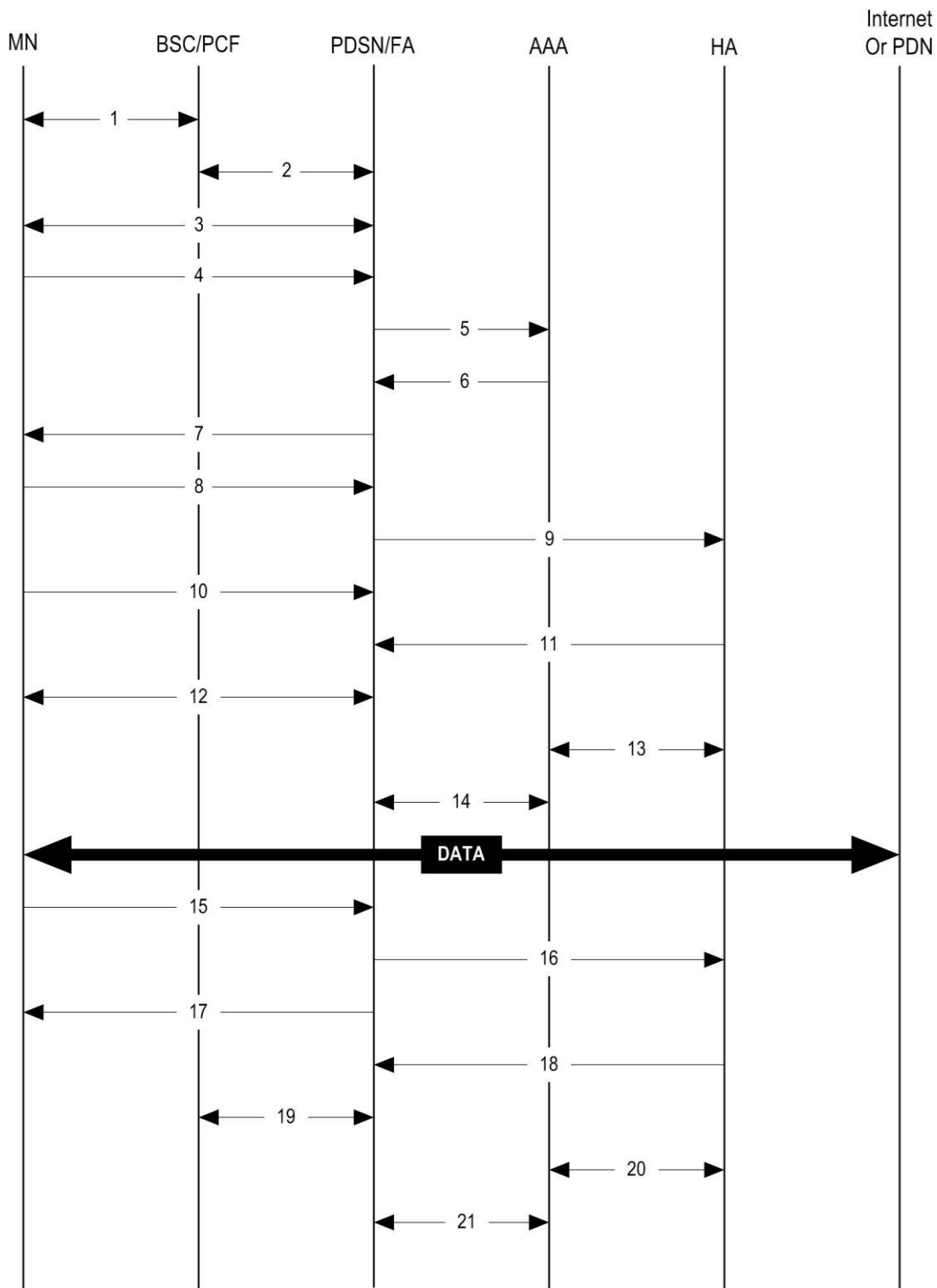
This section contains call flows displaying successful Proxy Mobile IP session setup scenarios. Two scenarios are described based on how the MN receives an IP address:

- **Scenario 1:** The AAA server specifies an IP address that the PDSN allocates to the MN from one of its locally configured static pools.
- **Scenario 2:** The HA assigns an IP address to the MN from one of its locally configured dynamic pools.

Scenario 1: AAA server and PDSN/FA Allocate IP Address

The following figure and table display and describe a call flow in which the MN receives its IP address from the AAA server and PDSN/FA.

Figure 8. AAA/PDSN Assigned IP Address Proxy Mobile IP Call Flow



335164

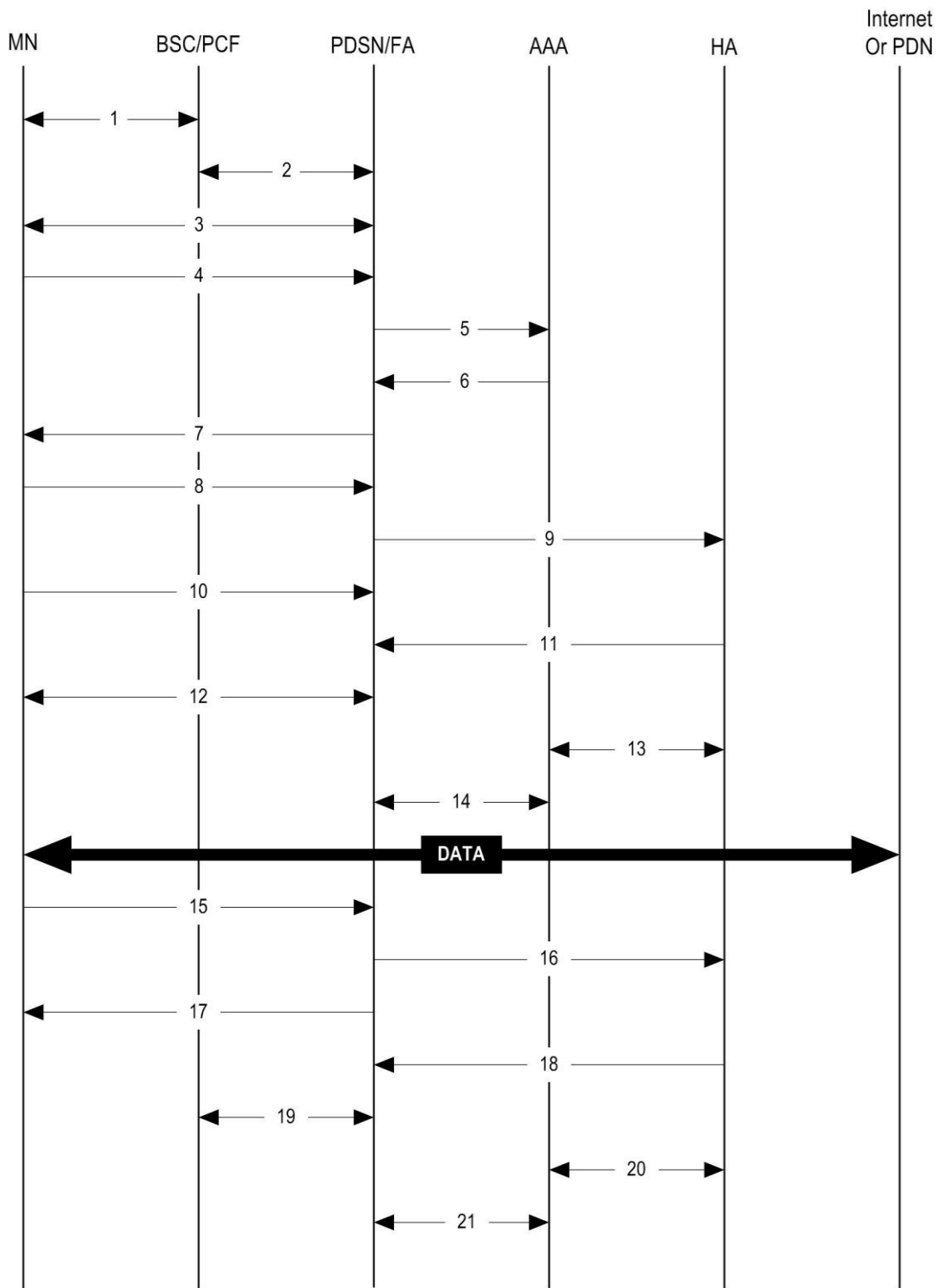
Table 3. AAA/PDSN Assigned IP Address Proxy Mobile IP Call Flow Description

Step	Description
1	Mobile Node (MN) secures a traffic channel over the airlink with the RAN through the BSC/PCF.
2	The PCF and PDSN/FA establish the R-P interface for the session.
3	The PDSN/FA and MN negotiate Link Control Protocol (LCP).
4	Upon successful LCP negotiation, the MN sends a PPP Authentication Request message to the PDSN/FA.
5	The PDSN/FA sends an Access Request message to the RADIUS AAA server.
6	The RADIUS AAA server successfully authenticates the subscriber and returns an Access Accept message to the PDSN/FA. The Accept message may contain various attributes to be assigned to the MN including the MN's Home Address (IP address) and the IP address of the HA to use.
7	The PDSN/FA sends a PPP Authentication Response message to the MN.
8	The MN sends an Internet Protocol Control Protocol (IPCP) Configuration Request message to the PDSN/FA with an MN address of 0.0.0.0.
9	The PDSN/FA forwards a Proxy Mobile IP Registration Request message to the HA. The message includes such things as the MN's home address, the IP address of the FA (the care-of-address), and the FA-HA extension (security parameter index (SPI)).
10	While the FA is communicating with the HA, the MN may send additional IPCP Configuration Request messages.
11	The HA responds with a Proxy Mobile IP Registration Response after validating the home address against its pool(s). The HA also creates a Mobile Binding Record (MBR) for the subscriber session.
12	The MN and the PDSN/FA negotiate IPCP. The result is that the MN is assigned the home address originally specified by the AAA server.
13	While the MN and PDSN/FA are negotiating IPCP, the HA and AAA server initiate accounting.
14	Upon completion of the IPCP negotiation, the PDSN/FA and AAA server initiate accounting fully establishing the session allowing the MN to send/receive data to/from the PDN.
15	Upon completion of the session, the MN sends an LCP Terminate Request message to the PDSN to end the PPP session.
16	The PDSN/FA sends a Proxy Mobile IP De-registration Request message to the HA.
17	The PDSN/FA send an LCP Terminate Acknowledge message to the MN ending the PPP session.
18	The HA sends a Proxy Mobile IP De-Registration Response message to the FA terminating the Pi interface
19	The PDSN/FA and the PCF terminate the R-P session.
20	The HA and the AAA server stop accounting for the session.
21	The PDSN and the AAA server stop accounting for the session.

Scenario 2: HA Assigns IP Address to MN from Locally Configured Dynamic Pools

The following figure and table display and describe a call flow in which the MN receives its IP address from the AAA server and PDSN/FA.

Figure 9. HA Assigned IP Address Proxy Mobile IP Call Flow



335164

Table 4. HA Assigned IP Address Proxy Mobile IP Call Flow Description

Step	Description
1	Mobile Node (MN) secures a traffic channel over the airlink with the RAN through the BSC/PCF.
2	The PCF and PDSN/FA establish the R-P interface for the session.
3	The PDSN/FA and MN negotiate Link Control Protocol (LCP).
4	Upon successful LCP negotiation, the MN sends a PPP Authentication Request message to the PDSN/FA.
5	The PDSN/FA sends an Access Request message to the RADIUS AAA server.
6	The RADIUS AAA server successfully authenticates the subscriber and returns an Access Accept message to the PDSN/FA. The Accept message may contain various attributes to be assigned to the MN including the IP address of the HA to use.
7	The PDSN/FA sends a PPP Authentication Response message to the MN.
8	The MN sends an Internet Protocol Control Protocol (IPCP) Configuration Request message to the PDSN/FA with an MN address of 0.0.0.0.
9	The PDSN/FA forwards a Proxy Mobile IP Registration Request message to the HA. The message includes such things as a Home Address indicator of 0.0.0.0, the IP address of the FA (the care-of-address), the IP address of the FA (the care-of-address), and the FA-HA extension (Security Parameter Index (SPI)).
10	While the FA is communicating with the HA, the MN may send additional IPCP Configuration Request messages.
11	The HA responds with a Proxy Mobile IP Registration Response. The response includes an IP address from one of its locally configured pools to assign to the MN (its Home Address). The HA also creates a Mobile Binding Record (MBR) for the subscriber session.
12	The MN and the PDSN/FA negotiate IPCP. The result is that the MN is assigned the home address originally specified by the AAA server.
13	While the MN and PDSN/FA are negotiating IPCP, the HA and AAA server initiate accounting.
14	Upon completion of the IPCP negotiation, the PDSN/FA and AAA server initiate accounting fully establishing the session allowing the MN to send/receive data to/from the PDN.
15	Upon completion of the session, the MN sends an LCP Terminate Request message to the PDSN to end the PPP session.
16	The PDSN/FA sends a Proxy Mobile IP De-registration Request message to the HA.
17	The PDSN/FA send an LCP Terminate Acknowledge message to the MN ending the PPP session.
18	The HA sends a Proxy Mobile IP De-Registration Response message to the FA terminating the Pi interface
19	The PDSN/FA and the PCF terminate the R-P session.
20	The HA and the AAA server stop accounting for the session.
21	The PDSN and the AAA server stop accounting for the session.

Supported Standards

The system supports the following industry standards for 1x/CDMA2000/EV-DO devices.

Requests for Comments (RFCs)

- RFC-768, User Datagram Protocol (UDP), August 1980
- RFC-791, Internet Protocol (IP), September 1982
- RFC-793, Transmission Control Protocol (TCP), September 1981
- RFC-894, A Standard for the Transmission of IP Datagrams over Ethernet Networks, April 1984
- RFC-1089, SNMP over Ethernet, February 1989
- RFC-1144, Compressing TCP/IP headers for low-speed serial links, February 1990
- RFC-1155, Structure and Identification of Management Information for TCP/IP-based Internets, May 1990
- RFC-1157, Simple Network Management Protocol (SNMP) Version 1, May 1990
- RFC-1212, Concise MIB Definitions, March 1991
- RFC-1213, Management Information Base for Network Management of TCP/IP-based Internets: MIB-II, March 1991
- RFC-1215, A Convention for Defining Traps for use with the SNMP, March 1991
- RFC-1224, Techniques for Managing Asynchronously Generated Alerts, May 1991
- RFC-1256, ICMP Router Discovery Messages, September 1991
- RFC-1305, Network Time Protocol (Version 3) Specification, Implementation and Analysis, March 1992
- RFC-1332, The PPP Internet Protocol Control Protocol (IPCP), May 1992
- RFC-1398, Definitions of Managed Objects for the Ethernet-Like Interface Types, January 1993
- RFC-1418, SNMP over OSI, March 1993
- RFC-1570, PPP LCP Extensions, January 1994
- RFC-1643, Definitions of Managed Objects for the Ethernet-like Interface Types, July 1994
- RFC-1661, The Point to Point Protocol (PPP), July 1994
- RFC-1662, PPP in HDLC-like Framing, July 1994
- RFC-1701, Generic Routing Encapsulation (GRE), October 1994
- RFC-1771, A Border Gateway Protocol 4 (BGP-4)
- RFC-1850, OSPF Version 2 Management Information Base, November 1995
- RFC-1901, Introduction to Community-based SNMPv2, January 1996
- RFC-1902, Structure of Management Information for Version 2 of the Simple Network Management Protocol (SNMPv2), January 1996
- RFC-1903, Textual Conventions for Version 2 of the Simple Network Management Protocol (SNMPv2), January 1996

- RFC-1904, Conformance Statements for Version 2 of the Simple Network Management Protocol (SNMPv2), January 1996
- RFC-1905, Protocol Operations for Version 2 of the Simple Network Management Protocol (SNMPv2), January 1996
- RFC-1906, Transport Mappings for Version 2 of the Simple Network Management Protocol (SNMPv2), January 1996
- RFC-1907, Management Information Base for Version 2 of the Simple Network Management Protocol (SNMPv2), January 1996
- RFC-1908, Coexistence between Version 1 and Version 2 of the Internet-standard Network Management Framework, January 1996
- RFC-1918, Address Allocation for Private Internets, February 1996
- RFC-1919, Classical versus Transparent IP Proxies, March 1996
- RFC-1962, The PPP Compression Control Protocol (CCP), June 1996
- RFC-1974, PPP STAC LZS Compression Protocol, August 1996
- RFC-2002, IP Mobility Support, May 1995
- RFC-2003, IP Encapsulation within IP, October 1996
- RFC-2004, Minimal Encapsulation within IP, October 1996
- RFC-2005, Applicability Statement for IP Mobility Support, October 1996
- RFC-2118, Microsoft Point-to-Point Compression (MPPC) Protocol, March 1997
- RFC-2136, Dynamic Updates in the Domain Name System (DNS UPDATE)
- RFC-2211, Specification of the Controlled-Load Network Element Service
- RFC-2246, The Transport Layer Security (TLS) Protocol Version 1.0, January 1999
- RFC-2290, Mobile IPv4 Configuration Option for PPP IPCP, February 1998
- RFC-2328, OSPF Version 2, April 1998
- RFC-2344, Reverse Tunneling for Mobile IP, May 1998
- RFC-2394, IP Payload Compression Using DEFLATE, December 1998
- RFC-2401, Security Architecture for the Internet Protocol, November 1998
- RFC-2402, IP Authentication Header (AH), November 1998
- RFC-2406, IP Encapsulating Security Payload (ESP), November 1998
- RFC-2408, Internet Security Association and Key Management Protocol (ISAKMP), November 1998
- RFC-2409, The Internet Key Exchange (IKE), November 1998
- RFC-2474, Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers, December 1998
- RFC-2475, An Architecture for Differentiated Services, December 1998
- RFC-2484, PPP LCP Internationalization Configuration Option, January 1999
- RFC-2486, The Network Access Identifier (NAI), January 1999
- RFC-2571, An Architecture for Describing SNMP Management Frameworks, April 1999
- RFC-2572, Message Processing and Dispatching for the Simple Network Management Protocol (SNMP), April 1999

- RFC-2573, SNMP Applications, April 1999
- RFC-2574, User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3), April 1999
- RFC-2597, Assured Forwarding PHB Group, June 1999
- RFC2598 - Expedited Forwarding PHB, June 1999
- RFC-2618, RADIUS Authentication Client MIB, June 1999
- RFC-2620, RADIUS Accounting Client MIB, June 1999
- RFC-2661, Layer Two Tunneling Protocol “L2TP”, August 1999
- RFC-2697, A Single Rate Three Color Marker, September 1999
- RFC-2698, A Two Rate Three Color Marker, September 1999
- RFC-2784, Generic Routing Encapsulation (GRE) - March 2000, IETF
- RFC-2794, Mobile IP Network Access Identifier Extension for IPv4, March 2000
- RFC-2809, Implementation of L2TP Compulsory Tunneling via RADIUS, April 2000
- RFC-2845, Secret Key Transaction Authentication for DNS (TSIG), May 2000
- RFC-2865, Remote Authentication Dial In User Service (RADIUS), June 2000
- RFC-2866, RADIUS Accounting, June 2000
- RFC-2867, RADIUS Accounting Modifications for Tunnel Protocol Support, June 2000
- RFC-2868, RADIUS Attributes for Tunnel Protocol Support, June 2000
- RFC-2869, RADIUS Extensions, June 2000
- RFC-3007, Secure Domain Name System (DNS) Dynamic Update, November 2000
- RFC-3012, Mobile IPv4 Challenge/Response Extensions, November 2000
- RFC-3095, Robust Header Compression (ROHC): Framework and four profiles: RTP, UDP, ESP and uncompressed, July 2001
- RFC-3101, OSPF NSSA Option, January 2003.
- RFC-3141, CDMA2000 Wireless Data Requirements for AAA, June 2001
- RFC-3143, Known HTTP Proxy/Caching Problems, June 2001
- RFC-3193, Securing L2TP using IPSEC, November 2001
- RFC-3241 Robust Header Compression (ROHC) over PPP, April 2002
- RFC-3409, Lower Layer Guidelines for Robust (RTP/UDP/IP) Header Compression, December 2002
- RFC-3519, NAT Traversal for Mobile IP, April 2003
- RFC-3543, Registration Revocation in Mobile IPv4, August 2003
- RFC 3576 - Dynamic Authorization Extensions to Remote Authentication Dial In User Service (RADIUS), July 2003
- RFC-3706, A Traffic-Based Method of Detecting Dead Internet Key Exchange (IKE) Peers, February 2004
- RFC-3759, Robust Header Compression (ROHC): Terminology and Channel Mapping Examples, April 2004
- RFC-3588, Diameter Based Protocol, September 2003
- RFC-4005, Diameter Network Access Server Application, August 2005
- RFC-4006, Diameter Credit-Control Application, August 2005

- Draft, Generalized Key Distribution Extensions for Mobile IP
- Draft, AAA Keys for Mobile IP

TIA and Other Standards

Telecommunications Industry Association (TIA) Standards

- TIA/EIA/IS-835-A, CDMA2000 Wireless IP Network Standard, April 2001
- TIA/EIA/IS-835-B, CDMA2000 Wireless IP Network Standard, October 2002
- TIA/EIA/IS-835-C, CDMA2000 Wireless IP Network Standard, August 2003
- TIA/EIA/IS-707-A-1, Data Service Options for Wideband Spread Spectrum Systems
- TIA/EIA/IS-707-A.5 Packet Data Services
- TIA/EIA/IS-707-A.9 High Speed Packet Data Services
- TIA/EIA/IS-2000.5, Upper Layer (Layer 3) Signaling for CDMA2000 Spread Spectrum Systems
- TIA/EIA/IS-2001, Interoperability Specifications (IOS) for CDMA2000 Access Network Interfaces
- TIA/EIA/TSB100, Wireless Network Reference Model
- TIA/EIA/TSB115, CDMA2000 Wireless IP Architecture Based on IETF Protocols
- TIA/EIA J-STD-025 PN4465, TR-45 Lawfully Authorized Electronic Surveillance

Object Management Group (OMG) Standards

- CORBA 2.6 Specification 01-09-35, Object Management Group

3GPP2 Standards

- 3GPP2 A.S0001-A v2: 3GPP2 Access Network Interfaces Interoperability Specification (also known as 3G-IOS v4.1.1)
- 3GPP2 P.S0001-A-3: Wireless IP Network Standard
- 3GPP2 P.S0001-B: Wireless IP Network Standard
- 3GPP2 S.R0068: Link Layer Assisted Robust Header Compression
- [9] 3GPP2 C.S0047-0: Link Layer Assisted Service Options for Voice-over-IP: Header Removal (SO60) and Robust Header Compression (SO61)
- 3GPP2 A.S0008 v3.0 Interoperability Specification (IOS) for High Rate Packet Data (HRPD) Access Network Interfaces
- 3GPP2 A.S0015-0 v2: Interoperability Specification (IOS) for CDMA2000 1X Access Network Interfaces — Part 5 (A3 and A7 12 Interfaces) (Partial Support) (also know as 3G-IOSv4.2)
- 3GPP2 P.S0001-B V1.0.0 Wireless IP Network Standard October 25, 2002 (relating to MIP interactions with IPSEC)
- 3GPP2 P.S0001 (TIA/EIA/IS-835-1) Version 1.0, Wireless IP Network Standard - December 10, 1999
- 3GPP2 P.R0001 (TSB115) Version 1.0.0, Wireless IP: Architecture Based on IETF Protocols - July 14, 2000

- 3GPP2 3GPP2 X.S0011-005-C Version: 1.0.0, CDMA2000 Wireless IP Network Standard: Accounting Services and 3GPP2 RADIUS VSAs - August 2003
- 3GPP2 X.S0011-006-C Version: 1.0.0, CDMA2000 Wireless IP Network Standard: PrePaid Packet Data Service - Date: August 2003
- 3GPP2 TSQA A.S0013-c v0.4 Interoperability Specification (IOS) for CDMA2000 June 2004
- 3GPP2 TSG-A A.S.0017-C baseline Interoperability Specification (IOS) for CDMA2000 Access Network Interfaces - Part 7(A10 and A11 Interfaces) (IOS v5.0 baseline) June 2004
- 3GPP2 A.S0012-D Segmentation for GRE January, 2005
- Inter-operability Specification (IOS) for CDMA2000 Access Network Interfaces
- 3GPP2 X.S0011-005-D Accounting Services and 3GPP2 RADIUS VSAs, February 2006
- 3GPP2 TSG-X (PSN) X.P0013-014-0, Service Based Bearer Control – Ty Interface Stage-3

IEEE Standards

- 802.1Q VLAN Standard

Chapter 2

Understanding the Service Operation and Configuration

The system provides wireless carriers with a flexible solution that can support both Simple IP and Mobile IP applications (independently or simultaneously) within a single scalable platform. Simple IP and Mobile IP applications are described in detail in the *System Overview Guide*.

When supporting Simple IP data applications, the system is configured to perform the role of a Packet Data Serving Node (PDSN) within the carrier's 3G CDMA2000 data network. The PDSN terminates the mobile subscriber's Point-to-Point Protocol (PPP) session and then routes data to and from the packet data network (PDN) on behalf of the subscriber. The PDN could consist of Wireless Application Protocol (WAP) servers or it could be the Internet.

When supporting Mobile IP data applications, the system can be configured to perform the role of the PDSN/Foreign Agent (FA) and/or the Home Agent (HA) within the carrier's 3G CDMA2000 data network. When functioning as an HA, the system can either be located within the carrier's 3G network or in an external enterprise or ISP network. Regardless, the PDSN/FA terminates the mobile subscriber's PPP session, and then routes data to and from the appropriate HA on behalf of the subscriber.

Prior to connecting to the command line interface (CLI) and beginning the system's configuration, there are important things to understand about how the system supports these applications. This chapter provides terminology and background information that must be considered before attempting to configure the system.

Terminology

This section defines some of the terms used in this manual.

Contexts

A context is a logical grouping or mapping of configuration parameters that pertain to various physical ports, logical IP interfaces, and services. A context can be thought of as a virtual private network (VPN).

The system supports the configuration of multiple contexts. Each is configured and operates independently from the others. Once a context has been created, administrative users can then configure services, logical IP interfaces, subscribers, etc. for that context. Administrative users would then bind the logical interfaces to physical ports.

Contexts can also be assigned domain aliases, wherein if a subscriber's domain name matches one of the configured alias names for that context, then that context is used.

Contexts on the system can be categorized as follows:

- **Source context:** Also referred to as the “ingress” context, this context provides the subscriber's point-of-entry in the system. It is also the context in which services are configured. For example, in a CDMA2000 network, the radio network containing the packet control functions (PCFs) would communicate with the system via R-P interfaces configured within the source context as part of the PDSN service.
- **Destination context:** Also referred to as the “egress” context, this context is where a subscriber is provided services (such as access to the Internet). Destination contexts are typically named after particular domains. For example, the system's destination context would be configured with the interfaces facilitating subscriber data traffic to/from the Internet, a VPN, or other PDN.
- **AAA context:** This context provides authorization, authentication, and accounting (AAA) functionality for subscriber and/or administrative user sessions. The AAA context contains context-specific AAA policies, the logical interfaces for communicating with AAA servers, and records for locally configured subscribers and/or administrative users.



Important: It is important to note that “source,” “destination,” and AAA functionality can optionally be configured within the same context or be configured as separate contexts. As a general rule, however, if the carrier owns and operates the AAA server, it is recommended that AAA functionality be configured within the source context. Conversely, if a home network other than the carrier's own operates the AAA server, it is recommended that AAA functionality be configured within the destination context. To ensure scalability, AAA functionality for subscriber sessions should not be configured in the local .

AAA Realms

A AAA realm is the location within the AAA context where subscriber-specific templates can be defined that are applied to subscribers who match that realm. A AAA realm is considered part of the AAA context; and the AAA context itself is also considered to be a realm. There may be many different AAA realms defined within a single AAA context.

An example of a realm would be that within a source context named ingress, there could be a domain alias of nova.com, another domain alias of bigco.com, and a single AAA configuration that is used by the entire system. In this example, the source context is also serving as a AAA context. There would be three specific AAA realms in this case; ingress, nova.com, and bigco.com, since all three could have their own subscriber templates defined.

The primary purpose of a AAA realm is to host a subscriber template for each realm that provides AAA attributes that may be used in the event that an authenticated subscriber's access-accept message from RADIUS fails to contain certain attributes. In this case, the default attributes contained in the realm-based subscriber template would be used. However, if the RADIUS authentication message contains an attribute from that subscriber's RADIUS user profile, then that information will be used to overwrite any default attribute parameters that are contained in the subscriber template.

More information about subscriber templates will be provided later in this chapter when subscribers are discussed.

Realms must be globally unique in their naming convention in that each realm name can only be used in one context in one system.

Ports

Ports are the physical interfaces that reside upon the system's line cards (Ethernet 10/100, Gigabit Ethernet 1000 Line Cards and the four-port Quad Gigabit Ethernet Line Card otherwise known as the Quad Gig-E or QGLC). Ethernet Port configuration addresses traffic profiles, data encapsulation methods, media type, and other information needed to allow physical connectivity between the system and the rest of the network. Ports are identified by the chassis slot number in which the line card resides, followed by the number of the physical connector on the line card. For example, Port 24/1 identifies connector number 1 on the card in slot 24.

Ports are associated with contexts through bindings. Additional information on bindings can be found in the Bindings section. Each physical port can be configured to support multiple logical IP interfaces each with up to 17 IP addresses (one primary and up to 16 secondaries).

Logical Interfaces

Prior to allowing the flow of user data, the port must be associated with a virtual circuit or tunnel called a logical interface. A logical interface within the system is defined as the logical assignment of a virtual router instance that provides higher-layer protocol transport, such as Layer 3 IP addressing. Interfaces are configured as part of the VPN context and are independent from the physical port that will be used to bridge the virtual interfaces to the network.

Logical interfaces are assigned to IP addresses and are bound to a specific port during the configuration process. Logical interfaces are also associated with services through bindings. Services are bound to an IP address that is configured for a particular logical interface. When associated, the interface takes on the characteristics of the functions enabled by the service. For example, if an interface is bound to a PDSN service, it will function as an R-P interface between the PDSN service and the PCF. Services are defined later in this section.

There are several types of logical interfaces that must be configured to support Simple and Mobile IP data applications as described below:

- **Management interface:** This interface provides the system's point of attachment to the management network. The interface supports remote access to the system CLI, Common Object Request Broker Architecture (CORBA)-based management via the Web Element Manager application, and event notification via the Simple Network Management Protocol (SNMP).

The system defaults to a Local context which should not be deleted. Management interfaces are defined in the Local management context and should only be bound to the ports on the Switch Processor Input/Output (SPIO) cards .
- **R-P interface:** Also referred to as the A10/A11 interface, this interface is the communications path between the Radio Node (also referred to as a PCF) and the PDSN.

The A10/A11 interface carries traffic signaling (A11) and user data traffic (A10). The A10/A11 interface is implemented in accordance with IS-835.

R-P interfaces are bound to ports on either the Ethernet 10/100 or Ethernet 1000/QGLC Line Cards.

- **Pi interface:** The packet interface (Pi) is the communications path between the PDSN/Foreign Agent (PDSN/FA) and the Home Agent (HA) for Mobile IP applications.

Pi interfaces are bound to ports on either the Ethernet 10/100 or Ethernet 1000/QGLC Line Cards.

- **PDN interface:** The interface to the packet data network (PDN). For Simple IP applications, this is the communications path between the PDSN and the PDN. For Mobile IP applications, this is the communications path between the HA and the PDN.

PDN interfaces are bound to Ethernet ports.

- **AAA interface:** The AAA interface is the connection between the PDSN and/or HA and the network servers that perform AAA functions. With this release of the system, the Remote Authentication Dial-In User Service (RADIUS) Protocol is used for communication on this interface.

AAA interfaces are bound to Ethernet ports. However, AAA interfaces can also be bound to the Local management context and to ports on the SPIO to provide AAA functions for subscribers, and for context-level administrative users.

- **ICC interface:** Inter-context communication (ICC) interfaces are only required when multiple services are configured in the same context. As mentioned previously, services are bound to interfaces. Creating an ICC interface provides a communication path between the services. For example, if an FA and HA service were configured in the same context, the FA service would need to be bound to an address assigned to the ICC interface and the HA service would need to be bound to a secondary address on the same ICC interface to provide a communications path between the two services.

The ICC interface must be configured with multiple addresses (one per service that it is facilitating) and bound to a physical port.

Bindings

A binding is an association between “elements” within the system. There are two types of bindings: static and dynamic.

Static binding is accomplished through the configuration of the system. Static bindings are used to associate:

- A specific logical interface (configured within a particular context) to a physical port. Once the interface is bound to the physical port, traffic can flow through the context just as if it were any physically defined circuit. Static bindings support any encapsulation method over any interface and port type.
- A service to an IP address assigned to a logical interface within the same context. This allows the interface to take on the characteristics (i.e., support the protocols) required by the service. For example, a PDSN service bound to a logical interface will cause the logical interface to take on the characteristics of an R-P interface within a 3G CDMA2000 network.

Dynamic binding associates a subscriber to a specific egress context based on the configuration of their profile or system parameters. This provides a higher degree of deployment flexibility as it allows a wireless carrier to support multiple services and facilitates seamless connections to multiple networks.

Services

Services are configured within a context and enable certain functionality. The following services can be configured on the system:

- **PDSN services:** Required for both Simple IP and Mobile IP applications, PDSN services define PDSN functionality for the system. The PDSN service must be bound to a logical interface within the same context.

Once bound, the interface takes on the characteristics of an R-P interface. Multiple services can be bound to the same logical interface. Therefore, a single physical port can facilitate multiple R-P interfaces.

The system treats the connection between the PCF and the PDSN service as a VPN (referred to as an RP-VPN). Individual R-P sessions are identified on this RP-VPN using the PCF address, the PDSN interface address, and the R-P Session ID.

- FA services: Currently supported only for use in CDMA 2000 networks, FA services are configured to support Mobile IP and define FA functionality on the system.

The system supports multiple Mobile IP configurations. A single system can perform the function of a FA only, an HA only, or a combined PDSN/FA/HA. Depending on your configuration, the FA service can create and maintain the Pi interface between the PDSN/FA and the HA or it can communicate with an HA service configured within the same context.

The FA service should be configured in a different context from the PDSN service. However, if the FA service will be communicating with an HA that is a separate network element, it must be configured within the same context as and be bound to the Pi interfaces that allow it to communicate with the HA.

- HA services: Currently supported only for use in CDMA 2000 networks, HA services are configured to support Mobile IP and define HA functionality on the system. Depending on your configuration, the HA service can be used to terminate the Pi interface from the FA or it can communicate with an FA service configured in the same context.

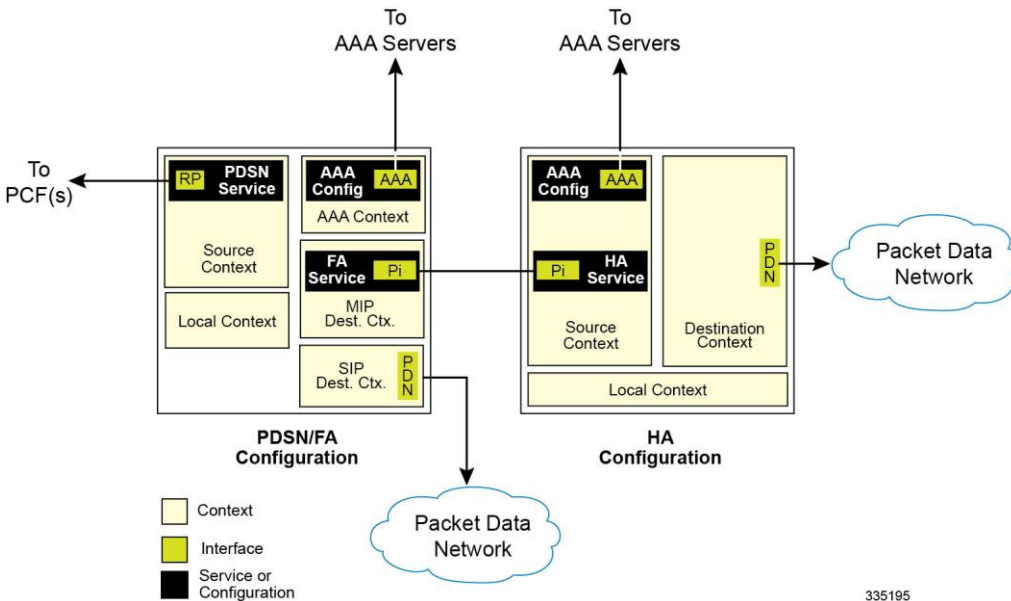
If the HA service is configured within the same system as the PDSN/FA, then it should be configured within the same context as the FA service. This context, then, would also facilitate the PDSN interfaces to the data network.

If the HA service is configured in a separate system, it should be configured in the same context as and bound to the Pi interfaces that allow it to communicate with the FA.

- LAC services: LAC services are configured on the system to provide Layer 2 Tunneling Protocol (L2TP) access concentrator (LAC) functionality. LAC services can be configured and used within either CDMA 2000 or GPRS/UMTS networks to provide secure tunneling to an L2TP network server (LNS) on a remote PDN.

The following figure diagrams the relationship between services, interfaces, and contexts within the system for CDMA 2000 networks.

Figure 10. Service, Interface, and Context Relationship Within the System for CDMA 2000 Networks



335195

AAA Servers

For most configurations, AAA servers will be used to store profiles, perform authentication, and maintain accounting records for each mobile data subscriber. The AAA servers communicate with the system over the AAA interface. The system supports the configuration of up to 128 AAA servers with which to communicate.

It is important to note that for Mobile IP, there can be foreign AAA (FAAA) and home AAA (HAAA) servers. The FAAA server(s) typically resides in the carrier's network. The HAAA server(s) could be owned and controlled by either the carrier or the home network. If the HAAA server is owned and controlled by the home network, accounting data can be transferred to the carrier via a AAA proxy server.

For most configurations, AAA servers will be used to store subscriber profiles and perform authentication. In addition, RADIUS AAA servers may be used to maintain accounting records for each mobile data subscriber as opposed to a GTPP-based Charging Gateway Function (CGF). The AAA servers communicate with the system over the AAA interface. The system supports the configuration of up to 128 AAA servers with which to communicate.

Subscribers

Subscribers are the end-users of the service who gain access to the Internet, their home network, or a public network through the system. There are three primary types of subscribers/users:

- **RADIUS-based Subscribers:** The most common type of subscriber, these users are identified by their International Mobile Subscriber Identity (IMSI) number, an Electronic Serial Number (ESN), or by their domain name or user name and are configured on and authenticated by a RADIUS AAA server.

Upon successful authentication various attributes (contained in the subscriber profile) are returned that dictate such things as session parameter settings (e.g. protocol settings, IP address assignment method, etc.), and what privileges the subscriber has (e.g. Simple IP, Mobile IP, etc.).

Attribute settings received by the system from a RADIUS AAA server take precedence over local-subscriber attributes and parameters configured on the system.

- **Local Subscribers:** These are subscribers, primarily used for testing purposes, that are configured and authenticated within a specific context. Unlike RADIUS-based subscribers, the local subscriber's user profile (containing attributes like those used by RADIUS-based subscribers) is configured within the context where they are created.

When local subscriber profiles are first created, attributes for that subscriber are set to the system's default settings. The same default settings are applied to all subscriber profiles including the subscriber named default (created automatically by the system for each system context; refer to the Default Subscribers and Realm-based Subscriber Templates section for more information). When configuring local profile attributes, the changes are made on a subscriber-by-subscriber basis.

Attributes configured for local subscribers take precedence over context-level parameters. However, they could be over-ridden by attributes returned from a RADIUS AAA server.

- **Management Subscribers:** A management user is an authorized user who can monitor, control, and configure the system through its command line interface (CLI) or Web Element Manager application. This management can be performed either locally, through the system's console port, or remotely through the use of the Telnet or secure shell (SSH) protocols. Management users are typically configured as a local subscriber within the localout-of-band management context, which is used exclusively for system management and administration. Like a local subscriber, the management subscriber's user profile is configured within the context where they are created (in this case the localout-of-band management context). However, management subscribers may also be authenticated remotely via RADIUS, if a AAA configuration exists within the localout-of-band management context.

Default Subscribers and Realm-based Subscriber Templates

Used for RADIUS-based subscribers, default subscribers – created on a per context basis, and subscriber templates – optionally created on per realm basis, contain default AAA attributes that can be used by subscribers who are remotely authenticated within a specific context or domain alias (AAA realm) when needed.

Default Subscriber

When each context is created, the system automatically creates a subscriber named default. There is only one default subscriber per context. The profile for the subscriber named default provides a configuration template of attribute values for subscribers who are remotely authenticated in that context. Any subscriber information that is not included in a RADIUS-based subscriber's user profile is configured according to the defaults defined for the default subscriber.

No matter where created all default subscribers initially have the same attributes set. The attributes for the default subscriber in each context can be changed from the CLI on a context by context basis.



Important: Local subscribers, who are authenticated locally within the context where they were created, cannot use any attributes that are defined for subscriber default. Rather, each local subscriber must have any attributes configured for them individually.

Realm-based Subscriber Templates

As defined earlier, a context can have numerous domain aliases that allows a single context to serve numerous different subscribers who have different domain names. When assigned, these domain aliases become AAA realms within the context.

Since each realm is used for a specific group of subscribers (e.g. corporate subscribers who may only have access to a specific corporate network that is protected by a virtual private network), each realm must have the ability to define

what AAA attributes should be applied to these different subscriber groups. This is achieved through the use of realm-based subscriber templates.

A subscriber template contains defined attributes that are specific to a select subscriber who belongs to that realm. Like the default subscriber (subscriber named default) who has a context-level set of configuration attributes, the subscriber template is used to provide default attribute values that may be used should a RADIUS user profile for a subscriber belonging to the specific realm fail to contain a needed attribute.



Important: If a realm-based subscriber template is not created for a specified realm, then the system will use the attributes configured for default subscriber (named default) within the context where the AAA realm exists.

Below is an example of how realm-based subscriber templates may be used.

As depicted above, a context named “ingress” contains:

- a PDSN service named “PDSN”.
- a AAA configuration that is used to communicate with an external RADIUS server. a default subscriber for the context named “default”. This default subscriber has an idle timeout attribute value of 45000 seconds.
- three additional realms, based on the following domain alias names:
 - “mega.com”, which has a realm-based subscriber template named “megauser”. This template contains an idle timeout attribute value of 36000 seconds.
 - “bigco.com”, which has a realm-based subscriber template named “bigco”. This template contains an idle timeout attribute value of 3600 seconds.
 - “smallco.com”, which has no realm-based subscriber template defined.

For this example, we will assume that all subscribers enter the system through the PDSN service defined in the [ingress] context. Configuration procedures and context selection methods will be provided in other sections in this document.

If a subscriber enters the system with a domain name that matches the context name “ingress” (example: user1@ingress), then the [ingress] context would be used for authentication. If the RADIUS server authenticates the subscriber and returns no value for the idle-timeout attribute, then this subscriber would be assigned the value contained in the subscriber default configuration. If a subscriber named user@mega.com enters the system with a domain name that matches a configured domain alias within the [ingress] context, in this case “mega.com”, then the [ingress] context would be used for authentication. However, since a realm-based subscriber template named “megauser” is defined within this AAA realm, then should the RADIUS server return no value for the idle-timeout attribute, then this subscriber would be assigned the value contained in the “megauser” subscriber template.

If a subscriber named user@bigco.com enters the system with a domain name that matches a configured domain alias within the [ingress] context, in this case “bigco.com”, then the [ingress] context would be used for authentication. However, since a realm-based subscriber template name “bigco” is defined within this AAA realm, any attributes not returned could be assigned from this subscriber template. In this example, the RADIUS server returns an idle-timeout of 18000 seconds. Because the RADIUS user profile contained a value for this attribute, the system would use that value (18000) rather than the value defined in the subscriber template.

If a subscriber name user@smallco.org enters the system with a domain name that matches a configured domain alias within the [ingress] context, in this case “smallco.org”, then the [ingress] context would be used for authentication. Note that the “smallco.org” domain alias does not have a realm-based subscriber template defined. In this case, the system would obtain any attribute values not returned from the RADIUS server from the subscriber default configuration. So if no attribute value was returned from RADIUS, user@smallco.org would be assigned an idle-timeout value of 45000 seconds.

How the System Selects Contexts

The previous section of this chapter defined what a context is and how it is used within the system. This section provides details about the process that is used to determine which context to use for context-level administrative user and/or subscriber sessions. Understanding this process allows you to better plan your configuration in terms of how many contexts and interfaces need to be configured.

Context Selection for Context-level Administrative User Sessions

The system comes configured with a context called local management context that should be used specifically for management purposes. The context selection process for context-level administrative users (those configured within a context) is simplified because the management interface(s) on the SPIO are only associated with the local out-of-band management context. Therefore, the source and destination contexts for a context-level administrative user responsible for managing the entire system should always be the local management context.

Although this is not commonly done, a context-level administrative user can also connect through other interfaces on the system and still have full system management privileges. A context-level administrative user can be created in a non-local management context. These management accounts only have privileges in the context where they are created. This type of management account can connect directly to a port in the context in which they belong, if local connectivity is enabled (SSHD for example) in that context. For all FTP or SFTP connections, you must connect through a SPIO interface. If you SFTP or FTP as a non-local management context account you must use the username syntax of `username@contextname`.

The context selection process becomes more involved depending on whether or not you will be configuring the system to provide local authentication or work with a AAA server to authenticate the context-level administrative user.

The system provides the flexibility to configure context-level administrative users locally (meaning that their profile will be configured and stored in its own memory) or remotely on an AAA server. If the user is configured locally, when he/she attempts to log onto the system, the system performs the authentication. If the user profile is configured on a AAA server, the system must determine how to contact the AAA server in order to perform authentication. It does this by determining the AAA context for the session.

The following table and figure describe the process that the system uses to select an AAA context for a context-level administrative user.

Table 5. Context-level Administrative User AAA Context Selection

Item	Description
1	During authentication, the system determines if local authentication is enabled in the local management context. If it is, the system attempts to authenticate the administrative user in the local out-of-band management context. If it is not, proceed to item 2 in this table. If the administrative user's username is configured, authentication is performed using the AAA configuration within the local management context. If not, proceed to item 2 in this table.
2	If local authentication is disabled on the system or if the administrative user's username is not configured in the local management context, then the system determines if a domain was received as part of the username. If there is a domain and it matches the name of a configured context or domain, then the AAA configuration within that context is used. If there is a domain and it does not match the name of a configured context or domain, go to item 4 in this table. If there is no domain as part of the username, go to item 3 in this table.

Item	Description
3	<p>If there was no domain specified in the username or the domain is not recognized, the system determines if an AAA Administrator Default Domain is configured.</p> <p>If the default domain is configured and it matches a configured context, then the AAA configuration within the AAA Administrator Default Domain context is used.</p> <p>If the default domain is not configured or does not match a configured context or domain, go to item 4 item this table</p>
4	<p>If a domain was specified as part of the username but it did not match a configured context, or if a domain was not specified as part of the username, the system determines if the AAA Administrator Last Resort context parameter is configured.</p> <p>If a last resort context is configured and it matches a configured context, then the AAA configuration within that context is used.</p> <p>If a last resort context is not configured or does not match a configured context or domain, then the AAA configuration within the local management context is used.</p>

Context Selection for Subscriber Sessions

The context selection process for a subscriber session is more involved than that for the administrative users.

The source context used to service a subscriber session is mostly dependant on the mapping of PCFs to PDSNs. Depending on this mapping and the subscribers' location in the network, the same subscriber may initiate several different data sessions throughout the day and have their session serviced by several different source contexts.

The AAA and destination context selection is determined based on what services are provided to the subscriber. For example, a carrier may only offer wireless Internet access and therefore be responsible for performing AAA functions for a subscriber session and for providing the network interfaces to the Internet. In this example, the carrier may choose to combine the source and AAA contexts into one and provide a separate destination context. Another carrier may choose to provide both wireless Internet access and VPN service to a corporate or Internet Service Provider (ISP) network. The system is flexible enough to simultaneously support these services because of the unique way in which it determines how to provide AAA functionality and route the session to the appropriate destination.

The following two sections provide details on the system's process in determining the correct AAA and destination contexts for a subscriber session.

AAA Context Selection for Subscriber Sessions

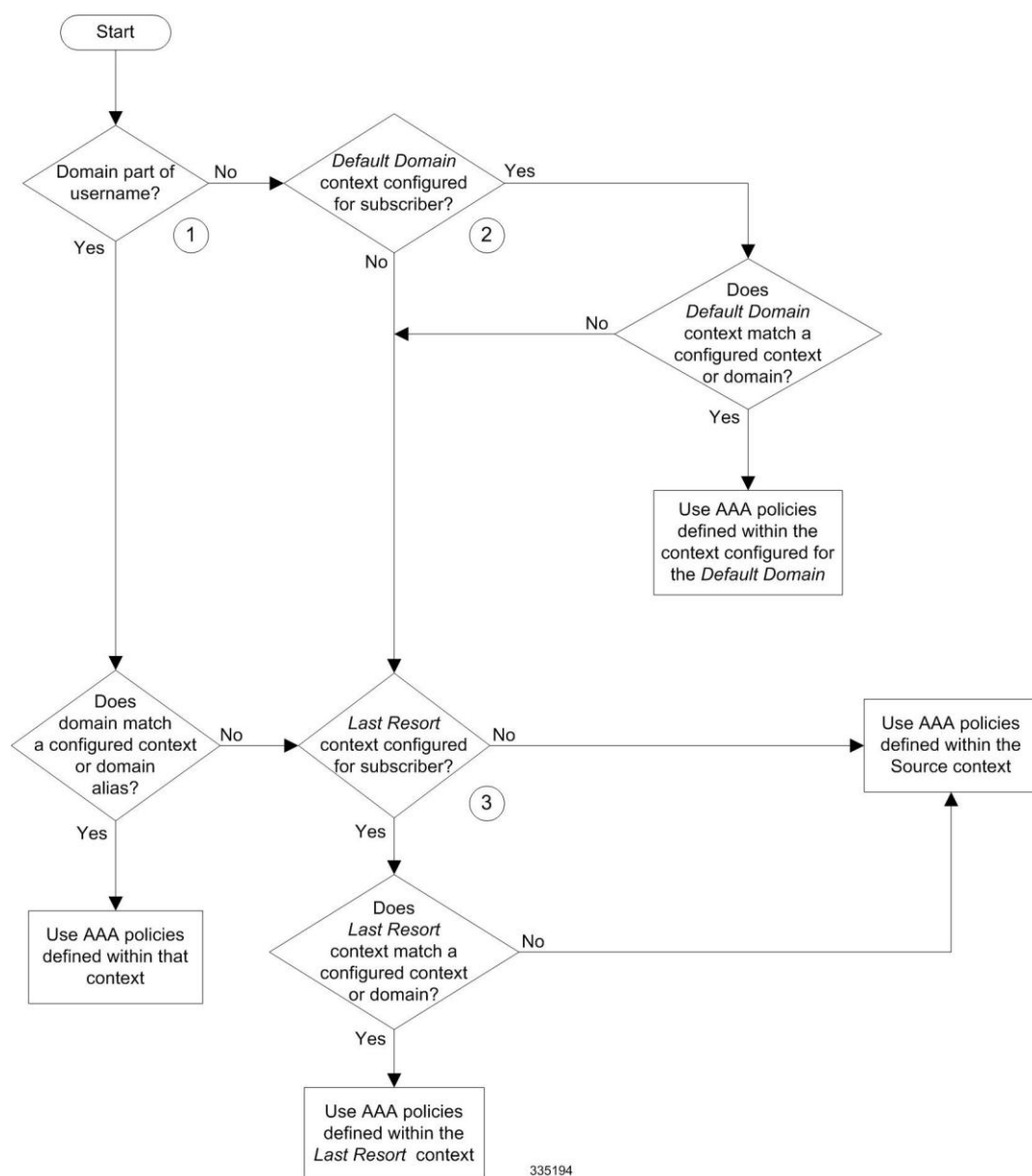
The following table and figure describe the process that the system uses to select an AAA context for a subscriber.

Table 6. Subscriber AAA Context Selection

Item	Description
1	<p>During authentication, the system determines if a domain was received as part of the username.</p> <p>If there is a domain and it matches the name of a configured context or domain alias, then the AAA configuration within that context is used.</p>

Item	Description
2	<p>If there was no domain specified in the username, the system determines if an AAA Subscriber Default Domain was configured. The AAA Subscriber Default Domain parameter is a system-wide AAA parameter that provides the system with the name of a context or domain that can provide AAA functions.</p> <p>If the AAA Subscriber Default Domain is configured and it matches a configured context or domain, then the AAA configuration within the AAA Subscriber Default Domain context is used.</p> <p>If the AAA Subscriber Default Domain is not configured or does not match a configured context or domain, then the system determines if an AAA Subscriber Last Resort is configured.</p>

Figure 11. Subscriber AAA Context Selection



335194

Destination Context Selection For Subscriber Sessions

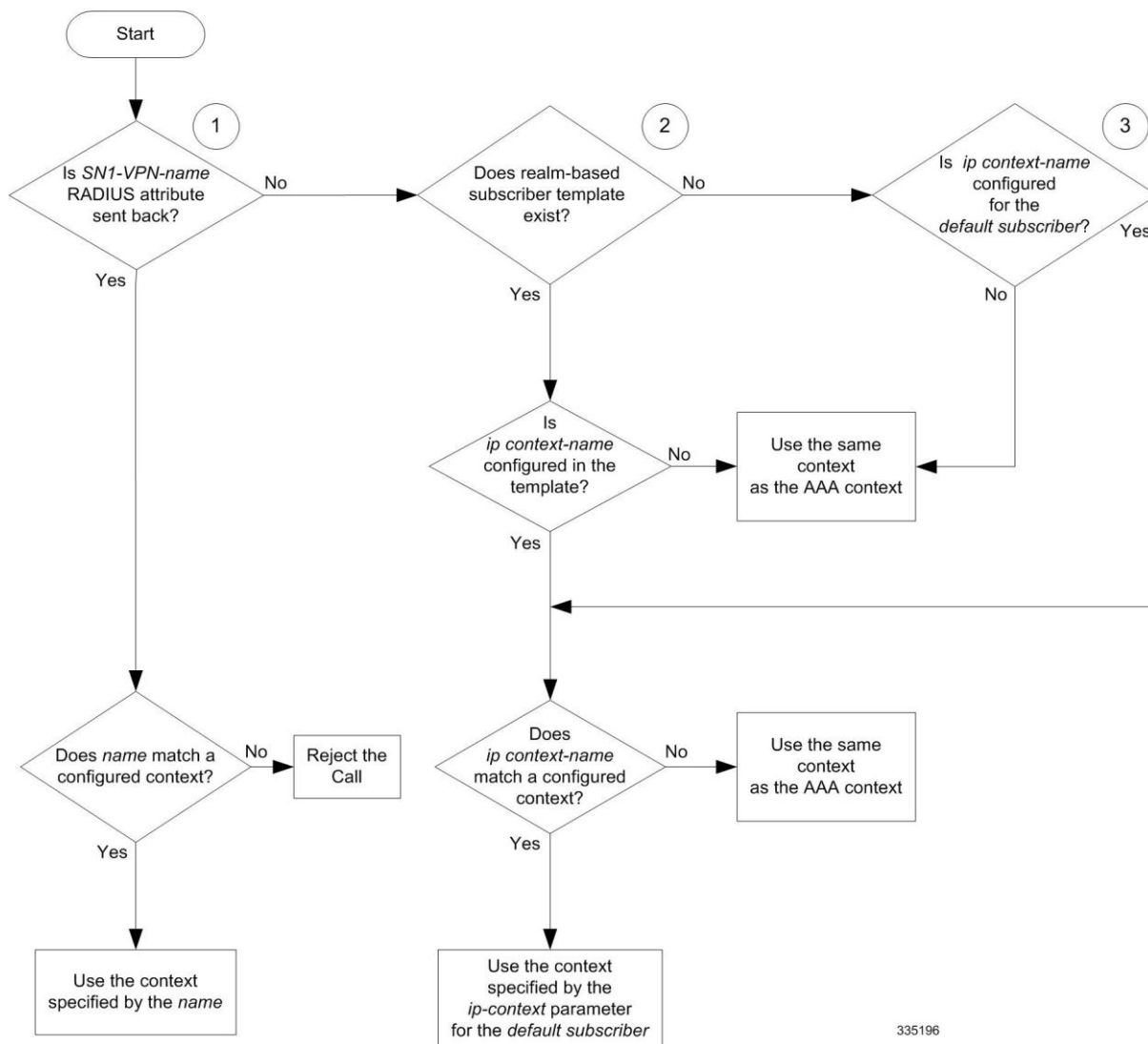
This section provides information on how a destination context is selected for subscribers whose profiles are configured on a RADIUS AAA server and for those whose profiles are locally configured. Note that the destination context for context-level administrative users is always the local management context.

The following table describes the process that the system uses to select a destination context for a RADIUS-based subscriber whose profile is configured on a RADIUS AAA server and for a subscriber whose profile is configured within a specific context.

Table 7. Subscriber Destination Context Selection

Item	Description
1	<p>The system supports a RADIUS attribute called SN1-VPN-name (or SN-VPN-name in some dictionaries). This attribute specifies the name of the subscriber's destination context. If configured in the subscriber's RADIUS user profile, it will be returned as part of the Access Accept message. If the SN1-VPN-Name attribute is returned, and it matches a configured context, then that context is used as the destination context.</p> <p>If the SN1-VPN-Name attribute is returned, and it does not match a configured context, the call is rejected.</p> <p>If the SN1-VPN-Name attribute is not returned with a value, go to item 2 in this table.</p>
2	<p>The system attempts to use the ip context name parameter configuration for the realm-based subscriber template or context-level default subscriber configured within the AAA context. If a realm-based subscriber template does not exist, go to item 3 in this table. If a realm-based subscriber template exists, the system checks to see if ip context-name is configured in the template.</p> <p>If ip context-name is not configured in the template, the AAA context is used for the destination context.</p> <p>If ip context-name is configured in the template, a check is made to see if it matches the name of a configured context.</p> <p>If ip context-name is configured in the template, but does not match the name of a configured context, the call is rejected.</p> <p>If ip context-name is configured in the template, and matches the name of a configured context, the destination context is set to the ip name-context for the default subscriber.</p>
3	<p>The local default subscriber profile contains an attribute called ip context-name. This attribute specifies the destination context to use for a local subscriber.</p> <p>If ip context-name is not configured, the AAA context is used for the destination context. If ip context-name is configured, a check is made to see if it matches the name of a configured context.</p> <p>If ip context-name is configured, but does not match the name of a configured context, the AAA context is used for the destination context.</p> <p>If ip context-name is configured, and matches the name of a configured context, the destination context is set to the ip name-context for the default subscriber.</p>

Figure 12. Subscriber Destination Context Selection



335196

Chapter 3

Simple IP Configuration Examples

This chapter provides information for several configuration examples that can be implemented on the system to support Simple IP data services.

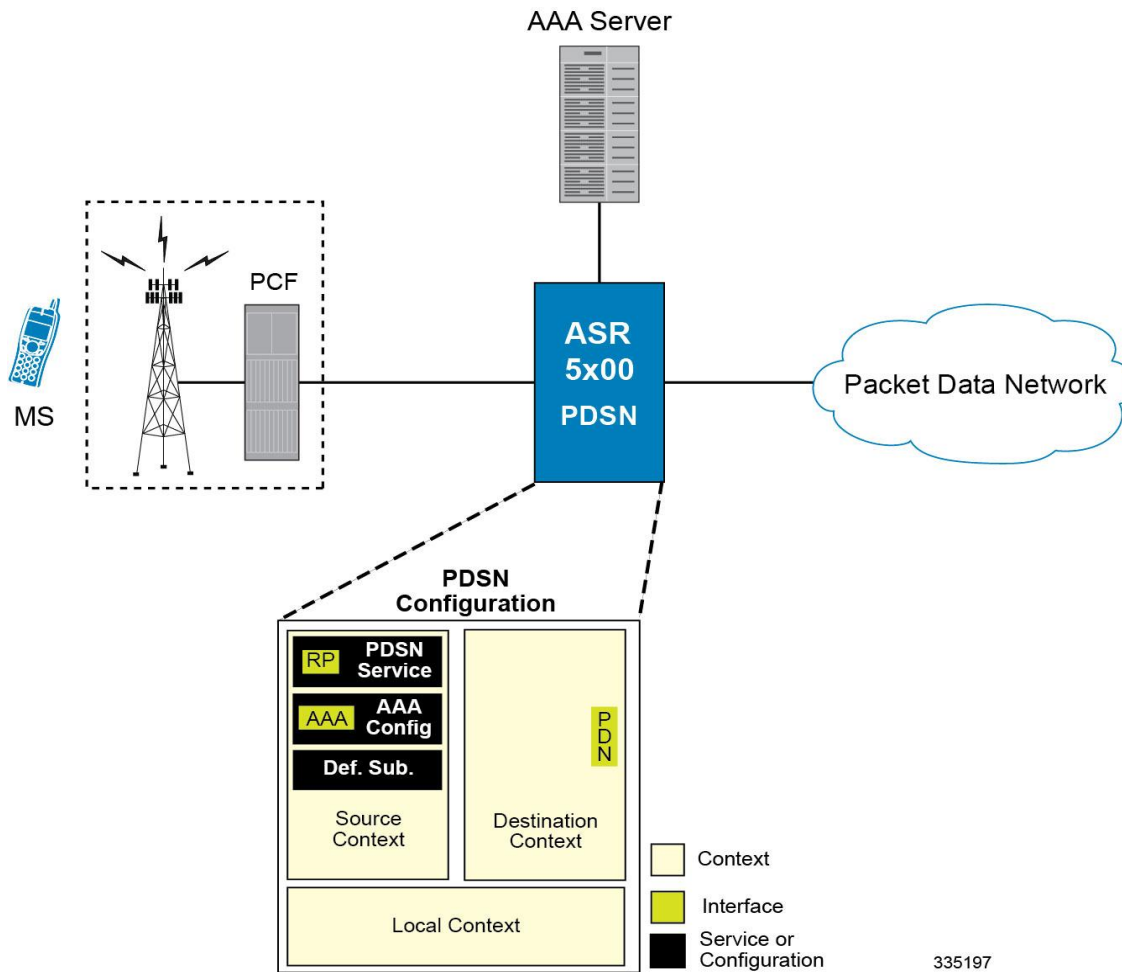


Important: This chapter does not discuss the configuration of the local context. Information about the local management context can be found in the *Command Line Interface Reference* guide.

Example 1: Simple IP Support Using a Single Source and Destination Context

The most simple configuration that can be implemented on the system to support Simple IP data applications requires that two contexts (one source and one destination) be configured on the system as shown below.

Figure 13. Simple IP Support Using a Single Source and Destination Context



The source context will facilitate the packet data serving node (PDSN) service(s) and the R-P and AAA interfaces. The source context will also be configured to provide AAA functionality for subscriber sessions. The destination context will facilitate the packet data network interface(s).

In this configuration, the wireless carrier provides the function of an Internet Service Provider (ISP) to their subscribers. The PDSN service in the source context terminates subscriber point-to-point protocol (PPP) sessions and routes their data traffic through the destination context to and from a packet data network such as the Internet.

Information Required

Prior to configuring the system as shown in this example, there is a minimum amount of information required. The following sections describe the information required to configure the source and destination contexts.

Source Context Configuration

The following table lists the information that is required to configure the source context.

Table 8. Required Information for Source Context Configuration

Required Information	Description
Source context name	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the source context will be recognized by the system.
R-P Interface Configuration	
R-P interface name	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the interface will be recognized by the system. Multiple names are needed if multiple interfaces will be configured. R-P interfaces are configured in the source context.
IP address and subnet	These will be assigned to the R-P interface. Multiple addresses and/or subnets are needed if multiple interfaces will be configured.
Physical port number	This specifies the physical port to which the interface will be bound. Ports are identified by the chassis slot number where the line card resides in, followed by the number of the physical connector on the line card. For example, port 17/1 identifies connector number 1 on the card in slot 17. A single physical port can facilitate multiple interfaces.
Physical port description	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the physical port will be recognized by the system. Multiple descriptions are needed if multiple ports will be used. Physical ports are configured within the source context and are used to bind logical R-P interfaces.
Gateway IP address	Used when configuring static routes from the R-P interface(s) to a specific network.
PDSN service Configuration	
PDSN service name	This is an identification string between 1 and 63 characters (alpha and/or numeric) by which the PDSN service will be recognized by the system. Multiple names are needed if multiple PDSN services will be used. PDSN services are configured in the source context.
UDP port number for R-P traffic	Specifies the port used by the PDSN service and the PCF for communications. The UDP port number can be any integer value between 1 and 65535. The default value is 699.
Authentication protocols used	Specifies how the system handles authentication: using a protocol (such as CHAP, PAP, or MSCHAP), or not requiring any authentication.
Domain alias for NAI-construction	Specifies a context name for the system to use to provide accounting functionality for a subscriber session. This parameter is needed only if the system is configured to support no authentication.

■ Example 1: Simple IP Support Using a Single Source and Destination Context

Required Information	Description
Security Parameter Index Information	PCF IP address: Specifies the IP address of the PCF that the PDSN service will be communicating with. The PDSN service allows the creation of a security profile that can be associated with a particular PCF. Multiple IP addresses are needed if the PDSN service will be communicating with multiple PCFs.
	Index: Specifies the shared SPI between the PDSN service and a particular PCF. The SPI can be configured to any integer value between 256 and 4294967295. Multiple SPIs can be configured if the PDSN service is to communicate with multiple PCFs.
	Secret: Specifies the shared SPI secret between the PDSN service and the PCF. The secret can be between 1 and 127 characters (alpha and/or numeric). An SPI secret is required for each SPI configured.
	Hash-algorithm: Specifies the algorithm used to hash the SPI and SPI secret. The possible algorithms that can be configured are MD5 per RFC 1321 and keyed-MD5 per RFC 2002. The default is MD5. A hash-algorithm is required for each SPI configured.
	Replay-protection process: Specifies how protection against replay-attacks is implemented. The possible processes are nonce and timestamp. The default is timestamp with a tolerance of 60 seconds. A replay-protection process is required for each SPI configured.
Subscriber session lifetime	Specifies the time in seconds that an A10 connection can exist before its registration is considered expired. The time is expressed in seconds and can be configured to any integer value between 1 and 65534, or the timer can be disabled to set an infinite lifetime. The default value is 1800 seconds.
AAA Interface Configuration	
AAA interface name	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the interface will be recognized by the system. Multiple names are needed if multiple interfaces will be configured. AAA interfaces will be configured in the source context.
IP address and subnet	These will be assigned to the AAA interface. Multiple addresses and/or subnets are needed if multiple interfaces will be configured.
Physical port number	This specifies the physical port to which the interface will be bound. Ports are identified by the chassis slot number where the line card resides in, followed by the number of the physical connector on the line card. For example, port 17/1 identifies connector number 1 on the card in slot 17. A single physical port can facilitate multiple interfaces.
Physical port description	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the physical port will be recognized by the system. Multiple descriptions are needed if multiple ports will be used. Physical ports are configured within the source context and are used to bind logical AAA interfaces.
Gateway IP address	Used when configuring static routes from the AAA interface(s) to a specific network.
RADIUS Server Configuration	

Required Information	Description
RADIUS Authentication server	IP Address: Specifies the IP address of the RADIUS authentication server the source context will communicate with to provide subscriber authentication functions. Multiple addresses are needed if multiple RADIUS servers will be configured. RADIUS authentication servers are configured within the source context. Multiple servers can be configured and each assigned a priority.
	Shared Secret: The shared secret is a string between 1 and 15 characters (alpha and/or numeric) that specifies the key that is exchanged between the RADIUS authentication server and the source context. A shared secret is needed for each configured RADIUS server.
	UDP Port Number: Specifies the port used by the source context and the RADIUS authentication server for communications. The UDP port number can be any integer value between 1 and 65535. The default value is 1812.
RADIUS Accounting server	IP Address: Specifies the IP address of the RADIUS accounting server that the source context will communicate with to provide subscriber accounting functions. Multiple addresses are needed if multiple RADIUS servers will be configured. RADIUS accounting servers are configured within the source context. Multiple servers can be configured and each assigned a priority.
	Shared Secret: The shared secret is a string between 1 and 15 characters (alpha and/or numeric) that specifies the key that is exchanged between the RADIUS accounting server and the source context. A shared secret is needed for each configured RADIUS server.
	UDP Port Number: Specifies the port used by the source context and the RADIUS Accounting server for communications. The UDP port number can be any integer value between 1 and 65535. The default value is 1813.
RADIUS attribute NAS Identifier	Specifies the name by which the source context will be identified in the Access-Request message(s) it sends to the RADIUS server. The name must be between 1 and 32 alpha and/or numeric characters and is case sensitive.
RADIUS NAS IP address	Specifies the IP address of the source context's AAA interface. A secondary IP address interface can optionally be configured.
Default Subscriber Configuration	
"Default" subscriber's IP context name	Specifies the name of the egress context on the system that facilitates the PDN ports. NOTE: For this configuration, the IP context name should be identical to the name of the destination context.

Destination Context Configuration

The following table lists the information that is required to configure the destination context.

Table 9. Required Information for Destination Context Configuration

Required Information	Description
----------------------	-------------

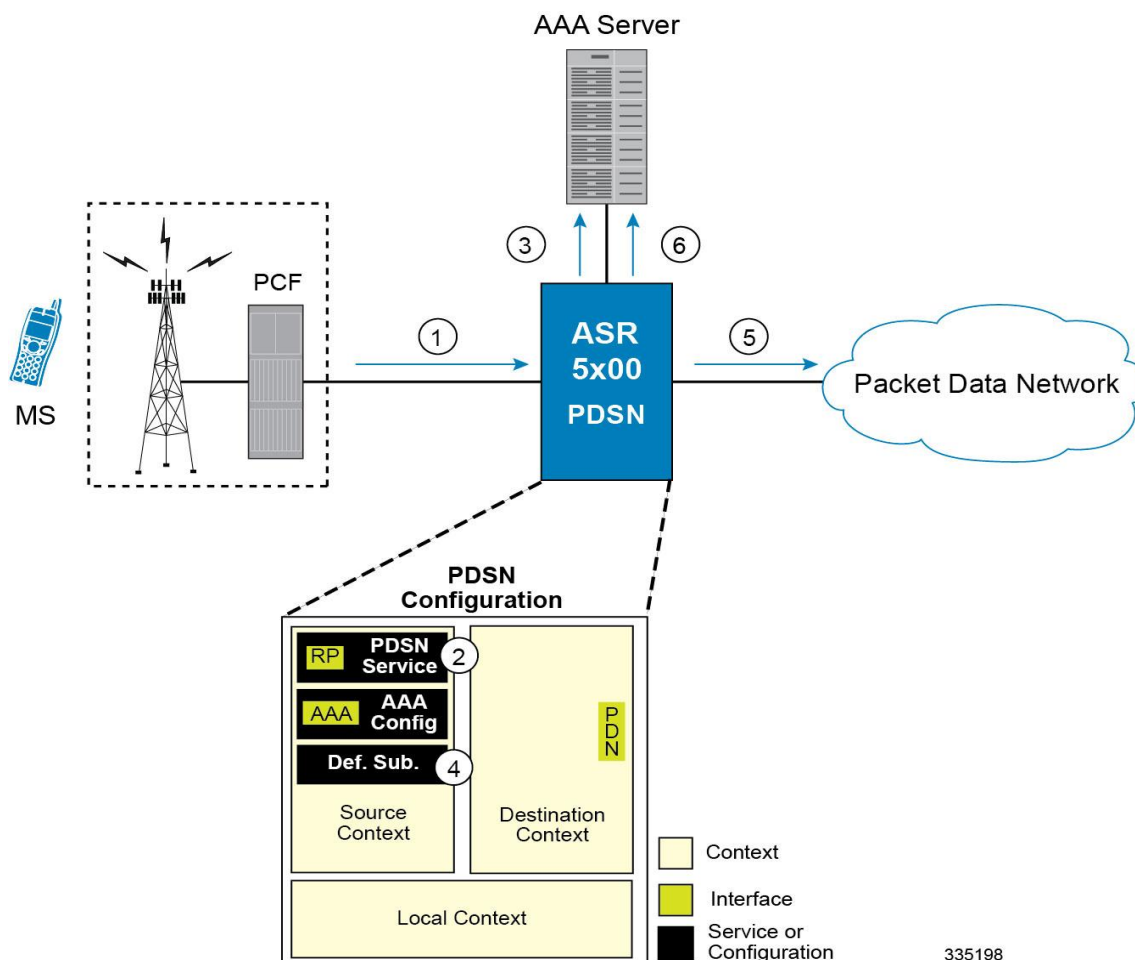
■ Example 1: Simple IP Support Using a Single Source and Destination Context

Required Information	Description
Destination context name	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the destination context will be recognized by the system. NOTE: For this configuration, the destination context name should not match the domain name of a specific domain.
PDN Interface Configuration	
PDN interface name	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the interface will be recognized by the system. Multiple names are needed if multiple interfaces will be configured. PDN interfaces are configured in the destination context.
IP address and subnet	These will be assigned to the PDN interface. Multiple addresses and/or subnets are needed if multiple interfaces will be configured.
Physical port number	This specifies the physical port to which the interface will be bound. Ports are identified by the chassis slot number where the line card resides in, followed by the number of the physical connector on the line card. For example, port 17/1 identifies connector number 1 on the card in slot 17. A single physical port can facilitate multiple interfaces.
Physical port description(s)	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the physical port will be recognized by the system. Multiple descriptions will be needed if multiple ports will be used. Physical ports are configured within the destination context and are used to bind logical PDN interfaces.
Gateway IP address(es)	Used when configuring static routes from the PDN interface(s) to a specific network.
IP Address Pool Configuration (optional)	
IP address pool name(s)	If IP address pools will be configured in the destination context(s), names or identifiers will be needed for them. The pool name can be between 1 and 31 alpha and/or numeric characters and is case sensitive.
IP pool addresses	An initial address and a subnet, or a starting address and an ending address, are required for each configured pool. The pool will then consist of every possible address within the subnet, or all addresses from the starting address to the ending address. The pool can be configured as public, private, or static.

How This Configuration Works

The following figure and the text that follows describe how this configuration with a single source and destination context would be used by the system to process a Simple IP data call.

Figure 14. Call Processing Using a Single Source and Destination Context



1. A subscriber session from the PCF is received by the PDSN service over the R-P interface.
2. The PDSN service determines which context to use in providing AAA functionality for the session. This process is described in the How the System Selects Contexts section located in the Understanding the System Operation and Configuration chapter of the *System Administration Guide*.

For this example, the result of this process is that PDSN service determined that AAA functionality should be provided by the *Source* context.

3. The system communicates with the AAA server specified in the *Source* context's AAA configuration to authenticate the subscriber.
4. Upon successful authentication, the system determines which egress context to use for the subscriber session. This process is described in the How the System Selects Contexts section located in the Understanding the System Operation and Configuration chapter of the *System Administration Guide*.

The system determines that the egress context is the destination context based on the configuration of either the *Default* subscriber's *ip-context* name or from the *SN-VPN-NAME* or *SN1-VPN-NAME* attributes that is configured in the subscriber's RADIUS profile.

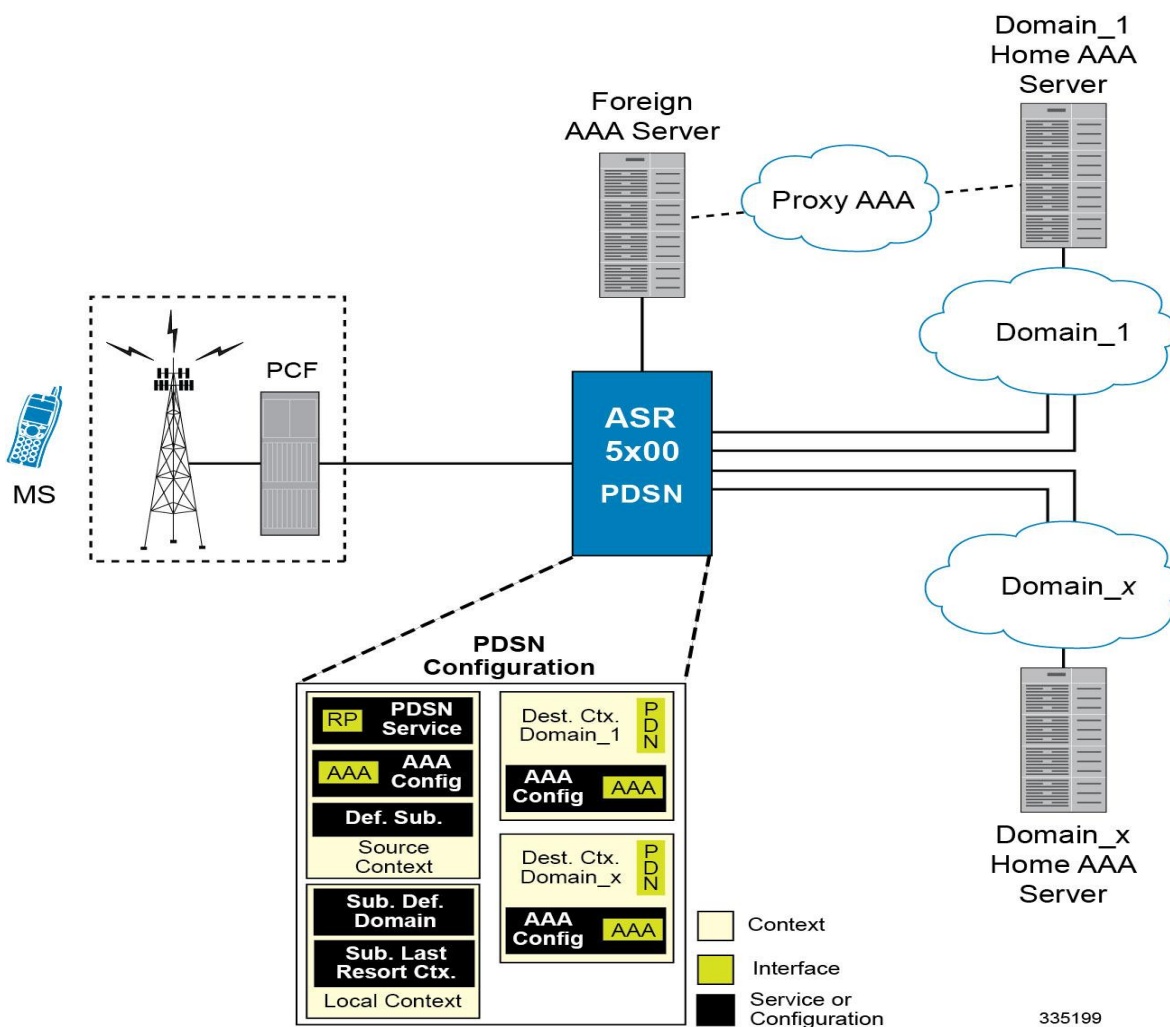
5. Data traffic for the subscriber session is routed through the PDN interface in the *Destination* context.
6. Accounting information for the session is sent to the AAA server over the AAA interface.

Example 2: Simple IP Using a Single Source Context and Multiple Outsourced Destination Contexts

The system allows the wireless carrier to easily generate additional revenue by providing the ability to configure separate contexts that can then be leased or outsourced to various enterprises or ISPs, each having a specific domain.

In order to support multiple outsourced domains, the system must first be configured with at least one source context and multiple destination contexts as shown in the following figure. The AAA servers could be owned/maintained by either the carrier or the domain. If they are owned by the domain, the carrier will have to receive the AAA information via proxy.

Figure 15. Simple IP Support Using a Single Source Context and Multiple Outsourced Destination Contexts



The source context will facilitate the PDSN service(s), and the R-P interface(s). The source context will also be configured with AAA interface(s) to provide AAA functionality for subscriber sessions. The destination contexts will

each be configured to facilitate PDN interfaces. In addition, because each of the destination contexts can be outsourced to different domains, they will also be configured with AAA interface(s) to provide AAA functionality for that domain.

In addition to the source and destination contexts, there are additional system-level AAA parameters that must be configured.

Information Required

Prior to configuring the system as shown in this example, there is a minimum amount of information required. The following sections describe the information required to configure the source and destination contexts.

Source Context Configuration

The following table lists the information that is required to configure the source context.

Table 10. Required Information for Source Context Configuration

Required Information	Description
Source context name	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the source context will be recognized by the system.
R-P Interface Configuration	
R-P interface name	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the interface will be recognized by the system. Multiple names are needed if multiple interfaces will be configured. R-P interfaces are configured in the source context.
IP address and subnet	These will be assigned to the R-P interface. Multiple addresses and/or subnets are needed if multiple interfaces will be configured.
Physical port number	This specifies the physical port to which the interface will be bound. Ports are identified by the chassis slot number where the line card resides in, followed by the number of the physical connector on the line card. For example, port 17/1 identifies connector number 1 on the card in slot 17. A single physical port can facilitate multiple interfaces.
Physical port description	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the physical port will be recognized by the system. Multiple descriptions are needed if multiple ports will be used. Physical ports are configured within the source context and are used to bind logical R-P interfaces.
Gateway IP address	Used when configuring static routes from the R-P interface(s) to a specific network.
PDSN service Configuration	
PDSN service name	This is an identification string between 1 and 63 characters (alpha and/or numeric) by which the PDSN service will be recognized by the system. Multiple names are needed if multiple PDSN services will be used. PDSN services are configured in the source context.
UDP port number for R-P traffic	Specifies the port used by the PDSN service and the PCF for communications. The UDP port number and can be any integer value between 1 and 65535. The default value is 699.

■ Example 2: Simple IP Using a Single Source Context and Multiple Outsourced Destination Contexts

Required Information	Description
Authentication protocols used	Specifies how the system handles authentication: using a protocol (such as CHAP, PAP, or MSCHAP), or not requiring any authentication.
Domain alias for NAI-construction	Specifies a context name for the system to use to provide accounting functionality for a subscriber session. This parameter is needed only if the system is configured to support no authentication.
Security Parameter Index Information	PCF IP address: Specifies the IP address of the PCF that the PDSN service will be communicating with. The PDSN service allows the creation of a security profile that can be associated with a particular PCF. Multiple IP addresses are needed if the PDSN service will be communicating with multiple PCFs.
	Index: Specifies the shared SPI between the PDSN service and a particular PCF. The SPI can be configured to any integer value between 256 and 4294967295. Multiple SPIs can be configured if the PDSN service is to communicate with multiple PCFs.
	Secret: Specifies the shared SPI secret between the PDSN service and the PCF. The secret can be between 1 and 127 characters (alpha and/or numeric). An SPI secret is required for each SPI configured.
	Hash-algorithm: Specifies the algorithm used to hash the SPI and SPI secret. The possible algorithms that can be configured are MD5 per RFC 1321 and keyed-MD5 per RFC 2002. The default is MD5. A hash-algorithm is required for each SPI configured.
	Replay-protection process: Specifies how protection against replay-attacks is implemented. The possible processes are nonce and timestamp. The default is timestamp with a tolerance of 60 seconds. A replay-protection process is required for each SPI configured.
Subscriber session lifetime	Specifies the time in seconds that an A10 connection can exist before its registration is considered expired. The time is expressed in seconds and can be configured to any integer value between 1 and 65534, or the timer can be disabled to set an infinite lifetime. The default value is 1800 seconds.
AAA Interface Configuration	
AAA interface name	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the interface will be recognized by the system. Multiple names are needed if multiple interfaces will be configured. AAA interfaces will be configured in the source context.
IP address and subnet	These will be assigned to the AAA interface. Multiple addresses and/or subnets are needed if multiple interfaces will be configured.
Physical port number	This specifies the physical port to which the interface will be bound. Ports are identified by the chassis slot number where the line card resides in, followed by the number of the physical connector on the line card. For example, port 17/1 identifies connector number 1 on the card in slot 17. A single physical port can facilitate multiple interfaces.
Physical port description	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the physical port will be recognized by the system. Multiple descriptions are needed if multiple ports will be used. Physical ports are configured within the source context and are used to bind logical AAA interfaces.
Gateway IP address	Used when configuring static routes from the AAA interface(s) to a specific network.

Required Information	Description
RADIUS Server Configuration	
RADIUS Authentication server	IP Address: Specifies the IP address of the RADIUS authentication server the source context will communicate with to provide subscriber authentication functions. Multiple addresses are needed if multiple RADIUS servers will be configured. RADIUS authentication servers are configured within the source context. Multiple servers can be configured and each assigned a priority.
	Shared Secret: The shared secret is a string between 1 and 15 characters (alpha and/or numeric) that specifies the key that is exchanged between the RADIUS authentication server and the source context. A shared secret is needed for each configured RADIUS server.
	UDP Port Number: Specifies the port used by the source context and the RADIUS authentication server for communications. The UDP port number can be any integer value between 1 and 65535. The default value is 1812.
RADIUS Accounting server	IP Address: Specifies the IP address of the RADIUS accounting server that the source context will communicate with to provide subscriber accounting functions. Multiple addresses are needed if multiple RADIUS servers will be configured. RADIUS accounting servers are configured within the source context. Multiple servers can be configured and each assigned a priority.
	Shared Secret: The shared secret is a string between 1 and 15 characters (alpha and/or numeric) that specifies the key that is exchanged between the RADIUS accounting server and the source context. A shared secret is needed for each configured RADIUS server.
	UDP Port Number: Specifies the port used by the source context and the RADIUS Accounting server for communications. The UDP port number can be any integer value between 1 and 65535. The default value is 1813.
RADIUS attribute NAS Identifier	Specifies the name by which the source context will be identified in the Access-Request message(s) it sends to the RADIUS server. The name must be between 1 and 32 alpha and/or numeric characters and is case sensitive.
RADIUS NAS IP address	Specifies the IP address of the source context's AAA interface. A secondary IP address interface can optionally be configured.
Default Subscriber Configuration	
"Default" subscriber's IP context name	Specifies the name of the egress context on the system that facilitates the PDN ports. NOTE: For this configuration, the IP context name should be identical to the name of the destination context.

Destination Context Configuration

The following table lists the information that is required to configure the destination context.

Table 11. Required Information for Destination Context Configuration

Required Information	Description
Destination context name	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the destination context will be recognized by the system. NOTE: For this configuration, the destination context name should not match the domain name of a specific domain.
PDN Interface Configuration	
PDN interface name	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the interface will be recognized by the system. Multiple names are needed if multiple interfaces will be configured. PDN interfaces are configured in the destination context.
IP address and subnet	These will be assigned to the PDN interface. Multiple addresses and/or subnets are needed if multiple interfaces will be configured.
Physical port number	This specifies the physical port to which the interface will be bound. Ports are identified by the chassis slot number where the line card resides in, followed by the number of the physical connector on the line card. For example, port 17/1 identifies connector number 1 on the card in slot 17. A single physical port can facilitate multiple interfaces.
Physical port description(s)	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the physical port will be recognized by the system. Multiple descriptions will be needed if multiple ports will be used. Physical ports are configured within the destination context and are used to bind logical PDN interfaces.
Gateway IP address(es)	Used when configuring static routes from the PDN interface(s) to a specific network.
IP Address Pool Configuration (optional)	
IP address pool name(s)	If IP address pools will be configured in the destination context(s), names or identifiers will be needed for them. The pool name can be between 1 and 31 alpha and/or numeric characters and is case sensitive.
IP pool addresses	An initial address and a subnet, or a starting address and an ending address, are required for each configured pool. The pool will then consist of every possible address within the subnet, or all addresses from the starting address to the ending address. The pool can be configured as public, private, or static.
AAA Interface Configuration	
AAA interface name	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the interface will be recognized by the system. Multiple names are needed if multiple interfaces will be configured. AAA interfaces will be configured in the source context.
IP address and subnet	These will be assigned to the AAA interface. Multiple addresses and/or subnets are needed if multiple interfaces will be configured.
Physical port number	This specifies the physical port to which the interface will be bound. Ports are identified by the chassis slot number where the line card resides in, followed by the number of the physical connector on the line card. For example, port 17/1 identifies connector number 1 on the card in slot 17. A single physical port can facilitate multiple interfaces.

Required Information	Description
Physical port description	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the physical port will be recognized by the system. Multiple descriptions are needed if multiple ports will be used. Physical ports are configured within the source context and are used to bind logical AAA interfaces.
Gateway IP address	Used when configuring static routes from the AAA interface(s) to a specific network.
RADIUS Server Configuration	
RADIUS Authentication server	IP Address: Specifies the IP address of the RADIUS authentication server the source context will communicate with to provide subscriber authentication functions. Multiple addresses are needed if multiple RADIUS servers will be configured. RADIUS authentication servers are configured within the source context. Multiple servers can be configured and each assigned a priority.
	Shared Secret: The shared secret is a string between 1 and 15 characters (alpha and/or numeric) that specifies the key that is exchanged between the RADIUS authentication server and the source context. A shared secret is needed for each configured RADIUS server.
	UDP Port Number: Specifies the port used by the source context and the RADIUS authentication server for communications. The UDP port number can be any integer value between 1 and 65535. The default value is 1812.
RADIUS Accounting server	IP Address: Specifies the IP address of the RADIUS accounting server that the source context will communicate with to provide subscriber accounting functions. Multiple addresses are needed if multiple RADIUS servers will be configured. RADIUS accounting servers are configured within the source context. Multiple servers can be configured and each assigned a priority.
	Shared Secret: The shared secret is a string between 1 and 15 characters (alpha and/or numeric) that specifies the key that is exchanged between the RADIUS accounting server and the source context. A shared secret is needed for each configured RADIUS server.
	UDP Port Number: Specifies the port used by the source context and the RADIUS Accounting server for communications. The UDP port number can be any integer value between 1 and 65535. The default value is 1813.
RADIUS attribute NAS Identifier	Specifies the name by which the source context will be identified in the Access-Request message(s) it sends to the RADIUS server. The name must be between 1 and 32 alpha and/or numeric characters and is case sensitive.
RADIUS NAS IP address	Specifies the IP address of the source context's AAA interface. A secondary IP address interface can optionally be configured.

System-Level AAA Configuration

The following table lists the information required to configure the system-level AAA parameters.

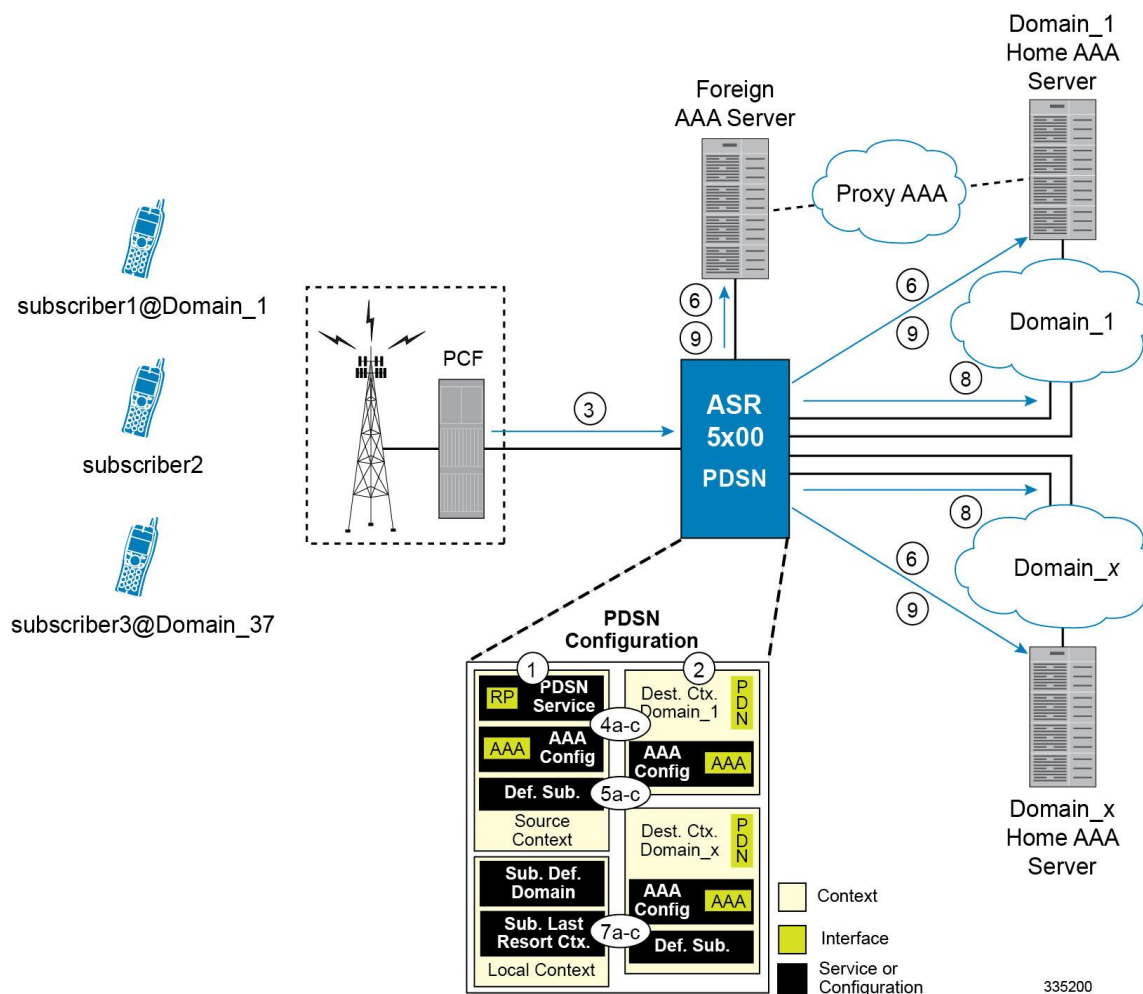
Table 12. Required Information for System-Level AAA Configuration

Required Information	Description
Subscriber default domain name	<p>Specifies the name of a context that can provide AAA functions in the event that the domain-part of the username is missing or poorly formed.</p> <p>This parameter will be applied to all subscribers if their domain can not be determined from their username regardless of what domain they are trying to access.</p> <p>NOTE: The default domain name can be the same as the source context.</p>
Subscriber Last-resort context	<p>Specifies the name of a context that can provide AAA functions in the event that the domain-part of the username was present but does not match the name of a configured destination context.</p> <p>This parameter will be applied to all subscribers if their specified domain does not match a configured destination context regardless of what domain they are trying to access.</p> <p>NOTE: The last-resort context name can be the same as the source context.</p>
Subscriber username format	<p>Specifies the format of subscriber usernames as to whether or not the username or domain is specified first and the character that separates them. The possible separator characters are:</p> <ul style="list-style-type: none"> • @ • % • - • \ • # • / <p>Up to six username formats can be specified. The default is <i>username @</i>.</p> <p>NOTE: The username string is searched from right to left for the separator character. Therefore, if there is one or more separator characters in the string, only the first one that is recognized is considered the actual separator. For example, if the default username format was used, then for the username string <i>user1@enterprise@isp1</i>, the system resolves to the username <i>user1@enterprise</i> with domain <i>isp1</i>.</p>

How This Configuration Works

The following figure and the text that follows describe how this configuration with a single source and destination context would be used by the system to process a Simple IP data call.

Figure 16. Call Processing Using a Single Source and Destination Context



1. The system-level AAA settings were configured as follows: Default subscriber domain name = DomainxSubscriber username format = username @No subscriber last-resort context name was configured. The IP context names for the Default subscriber were configured as follows: Within the Source context, the IP context name was configured as Domainx. Within the Domainx context, the IP context name was configured as Domainx. Sessions are received by the PDSN service from the PCF over the R-P interface for subscriber1@Domain1, subscriber2, and subscriber3@Domain37. The PDSN service attempts to determine the domain names for each session. For subscriber1, the PDSN service determines that a domain name is present and is Domain1. For subscriber2, the PDSN service determines that no domain name is present. For subscriber3, the PDSN service determines that a domain name is present and is Domain37. The PDSN service determines which context to use to provide AAA functionality for the session. This process is described in the How the System Selects Contexts section located in the Understanding the System Operation and Configuration chapter of the System Administration Guide. For subscriber1, the PDSN service determines that a context is configured with a name that matches the domain name specified in the username string (Domain1). Therefore, Domain1 is used. For subscriber2, the PDSN service determines that Domainx was configured as the subscriber default domain name. Therefore, Domainx was used. For subscriber3, the PDSN service determines that no context was configured that matched the domain name specified in the username string (Domain37). Because no subscriber last-resort context name is configured, the source context is used. The system then communicates with the AAA servers specified in each of the selected context's AAA configuration to authenticate the

■ Example 2: Simple IP Using a Single Source Context and Multiple Outsourced Destination Contexts

subscriber. Upon successful authentication of all three subscribers, the PDSN service determines which destination context to use for each of the subscriber sessions. This process is described in the How the System Selects Contexts section located in the Understanding the System Operation and Configuration chapter of the System Administration Guide. For subscriber1, the PDSN service receives the SN-VPN-NAME or SN1-VPN-NAME attribute equal to Domain1 as part of the authentication accept message from the AAA server on Domain1's network. Therefore, Domain1 is used as the destination context. For subscriber2, the PDSN service determined that the SN-VPN-NAME or SN1-VPN-NAME attribute was not returned with the Authentication Accept response, and determines the subscriber IP context name configured for the Default subscriber within the Domainx context. Because this parameter is configured to Domainx, the Domainx context will be used as the destination context. For subscriber3, the PDSN service determines that the SN-VPN-NAME or SN1-VPN-NAME attribute was not returned with the Authentication Accept response, and determined the Default subscriber IP context name configured within the Source context. Because this parameter is configured to Domainx, the Domainx context is used as the destination context. Data traffic for the subscriber session is routed through the PDN interface in each subscriber's destination context. Accounting messages for the session are sent to the AAA servers over the AAA interfaces.

A subscriber session from the PCF is received by the PDSN service over the R-P interface.

2. The PDSN service determines which context to use in providing AAA functionality for the session. This process is described in the How the System Selects Contexts section located in the Understanding the System Operation and Configuration chapter of the *System Administration Guide*.

For this example, the result of this process is that PDSN service determined that AAA functionality should be provided by the *Source* context.

3. The system communicates with the AAA server specified in the *Source* context's AAA configuration to authenticate the subscriber.
4. Upon successful authentication, the system determines which egress context to use for the subscriber session. This process is described in the How the System Selects Contexts section located in the Understanding the System Operation and Configuration chapter of the *System Administration Guide*.

The system determines that the egress context is the destination context based on the configuration of either the *Default* subscriber's *ip-context* name or from the *SN-VPN-NAME* or *SN1-VPN-NAME* attributes that is configured in the subscriber's RADIUS profile.

5. Data traffic for the subscriber session is routed through the PDN interface in the *Destination* context.
6. Accounting information for the session is sent to the AAA server over the AAA interface.

1. The system-level AAA settings were configured as follows:
 - Default subscriber domain name = Domainx
 - Subscriber username format = username @
 - No subscriber last-resort context name was configured.
2. The IP context names for the Default subscriber were configured as follows:
 - Within the Source context, the IP context name was configured as Domainx.
 - Within the Domainx context, the IP context name was configured as Domainx.
3. Sessions are received by the PDSN service from the PCF over the R-P interface for subscriber1@Domain1, subscriber2, and subscriber3@Domain37.
4. The PDSN service attempts to determine the domain names for each session.
 - For subscriber1, the PDSN service determines that a domain name is present and is Domain1.
 - For subscriber2, the PDSN service determines that no domain name is present.
 - For subscriber3, the PDSN service determines that a domain name is present and is Domain37.

5. The PDSN service determines which context to use to provide AAA functionality for the session. This process is described in the How the System Selects Contexts section located in the Understanding the System Operation and Configuration chapter of the System Administration Guide.
 - For subscriber1, the PDSN service determines that a context is configured with a name that matches the domain name specified in the username string (Domain1). Therefore, Domain1 is used.
 - For subscriber2, the PDSN service determines that Domainx was configured as the subscriber default domain name. Therefore, Domainx was used.
 - For subscriber3, the PDSN service determines that no context was configured that matched the domain name specified in the username string (Domain37). Because no subscriber last-resort context name is configured, the source context is used.
6. The system then communicates with the AAA servers specified in each of the selected context's AAA configuration to authenticate the subscriber.
7. Upon successful authentication of all three subscribers, the PDSN service determines which destination context to use for each of the subscriber sessions. This process is described in the How the System Selects Contexts section located in the Understanding the System Operation and Configuration chapter of the System Administration Guide.
 - For subscriber1, the PDSN service receives the SN-VPN-NAME or SN1-VPN-NAME attribute equal to Domain1 as part of the authentication accept message from the AAA server on Domain1's network. Therefore, Domain1 is used as the destination context.
 - For subscriber2, the PDSN service determined that the SN-VPN-NAME or SN1-VPN-NAME attribute was not returned with the Authentication Accept response, and determines the subscriber IP context name configured for the Default subscriber within the Domainx context. Because this parameter is configured to Domainx, the Domainx context will be used as the destination context.
 - For subscriber3, the PDSN service determines that the SN-VPN-NAME or SN1-VPN-NAME attribute was not returned with the Authentication Accept response, and determined the Default subscriber IP context name configured within the Source context. Because this parameter is configured to Domainx, the Domainx context is used as the destination context.
8. Data traffic for the subscriber session is routed through the PDN interface in each subscriber's destination context.
9. Accounting messages for the session are sent to the AAA servers over the AAA interfaces

Chapter 4

Mobile IP Configuration Examples

This chapter provides information for several configuration examples that can be implemented on the system to support Mobile IP (MIP) data services.



Important: This chapter does not discuss the configuration of the local management context. Information about the local management context can be found in Chapter 1 of Command Line Reference. Additionally, when configuring Mobile IP take into account the MIP timing considerations discussed in *MIP Timer Considerations*.

Example 1: Mobile IP Support Using the System as a PDSN/FA

The system supports both Simple and Mobile IP. For Mobile IP applications, the system can be configured to perform the function of a Packet Data Serving Node/Foreign Agent (PDSN/FA) and/or a Home Agent (HA). This example describes what is needed for and how the system performs the role of the PDSN/FA. Examples 2 and 3 provide information on using the system to provide HA functionality.

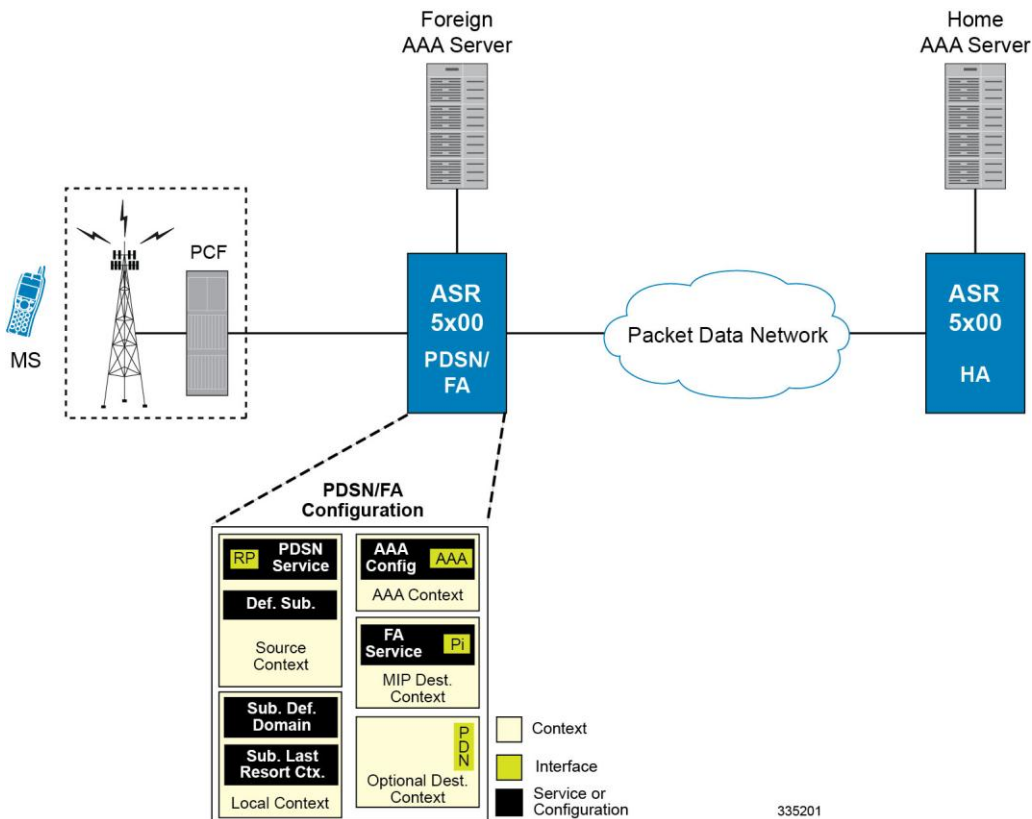
The system's PDSN/FA configuration for Mobile IP applications is best addressed with three contexts (one source, one AAA, and one Mobile IP destination) configured as shown in the figure below.

Important: A fourth context that serves as a destination context must also be configured if Reverse Tunneling is disabled in the FA service configuration. Reverse Tunneling is enabled by default.

The source context will facilitate the PDSN service(s), and the R-P interfaces. The AAA context will be configured to provide foreign AAA functionality for subscriber sessions and facilitate the AAA interfaces. The MIP destination context will facilitate the FA service(s) and the Pi interface(s) from the PDSN/FA to the HA.

The optional destination context will allow the routing of data from the mobile node to the packet data network by facilitating a packet data network (PDN) interface. This context will be used only if reverse tunneling was disabled.

Figure 17. Mobile IP Support using the system as a PDSN/FA



Information Required

Prior to configuring the system as shown in this example, there is a minimum amount of information required. The following sections describe the information required to configure the source and destination contexts.

Source Context Configuration

The following table lists the information that is required to configure the source context.

Table 13. Required Information for Source Context Configuration

Required Information	Description
Source context name	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the source context will be recognized by the system.
R-P Interface Configuration	
R-P interface name	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the interface will be recognized by the system. Multiple names are needed if multiple interfaces will be configured. R-P interfaces are configured in the source context.
IP address and subnet	These will be assigned to the R-P interface. Multiple addresses and/or subnets are needed if multiple interfaces will be configured.
Physical port number	This specifies the physical port to which the interface will be bound. Ports are identified by the chassis slot number where the line card resides in, followed by the number of the physical connector on the line card. For example, port 17/1 identifies connector number 1 on the card in slot 17. A single physical port can facilitate multiple interfaces.
Physical port description	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the physical port will be recognized by the system. Multiple descriptions are needed if using multiple ports. Physical ports are configured within the source context and are used to bind logical R-P interfaces.
Gateway IP address	Used when configuring static routes from the R-P interface(s) to a specific network.
PDSN service Configuration	
PDSN service name	This is an identification string between 1 and 63 characters (alpha and/or numeric) by which the PDSN service will be recognized by the system. Multiple names are needed if using multiple PDSN services. PDSN services are configured in the source context.
UDP port number for R-P traffic	Specifies the port used by the PDSN service and the PCF for communications. The UDP port number and can be any integer value between 1 and 65535. The default value is 699.
Authentication protocols used	Specifies how the system handles authentication: using a protocol (such as CHAP, PAP, or MSCHAP), or not requiring any authentication.
Domain alias for NAI-construction	Specifies a context name for the system to use to provide accounting functionality for a subscriber session. This parameter is needed only if the system is configured to support no authentication.


■ Example 1: Mobile IP Support Using the System as a PDSN/FA

Required Information	Description
Security Parameter Index Information	PCF IP address: Specifies the IP address of the PCF that the PDSN service will be communicating with. The PDSN service allows the creation of a security profile that can be associated with a particular PCF. Multiple IP addresses are needed if the PDSN service is to communicate with multiple PCFs.
	Index: Specifies the shared SPI between the PDSN service and a particular PCF. The SPI can be configured to any integer value between 256 and 4294967295. Configure multiple SPIs if the PDSN service is to communicate with multiple PCFs.
	Secret: Specifies the shared SPI secret between the PDSN service and the PCF. The secret can be between 1 and 127 characters (alpha and/or numeric). An SPI secret is required for each SPI configured.
	Hash-algorithm: Specifies the algorithm used to hash the SPI and SPI secret. The possible algorithms that can be configured are MD5 per RFC 1321 and keyed-MD5 per RFC 2002. The default is MD5. A hash-algorithm is required for each SPI configured.
	Replay-protection process: Specifies how protection against replay-attacks is implemented. The possible processes are nonce and timestamp. The default is timestamp with a tolerance of 60 seconds. A replay-protection process is required for each SPI configured.
Subscriber session lifetime	Specifies the time in seconds that an A10 connection can exist before its registration is considered expired. The time is expressed in seconds and can be configured to any integer value between 1 and 65534, or the timer can be disabled to set an infinite lifetime. The default value is 1800 seconds.
Mobile IP FA context name	Specifies the name of the context in which the FA service is configured.

AAA Context Configuration

The following table lists the information that is required to configure the AAA context.

Table 14. Required Information for AAA Context Configuration

Required Information	Description
AAA context name	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the AAA context will be recognized by the system. <div style="border: 1px solid black; padding: 5px; margin-top: 10px;">  Important: If a separate system is used to provide HA functionality, the AAA context name should match the name of the context in which the AAA functionality is configured on the HA machine. </div>
AAA Interface Configuration	

Required Information	Description
AAA interface name	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the interface will be recognized by the system. Multiple names are needed if multiple interfaces will be configured. AAA interfaces will be configured in the source context.
IP address and subnet	These will be assigned to the AAA interface. Multiple addresses and/or subnets are needed if multiple interfaces will be configured.
Physical port number	This specifies the physical port to which the interface will be bound. Ports are identified by the chassis slot number where the line card resides in, followed by the number of the physical connector on the line card. For example, port 17/1 identifies connector number 1 on the card in slot 17. A single physical port can facilitate multiple interfaces.
Physical port description	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the physical port will be recognized by the system. Multiple descriptions are needed if multiple ports will be used. Physical ports are configured within the destination context and are used to bind logical AAA interfaces.
Gateway IP address(es)	Used when configuring static routes from the AAA interface(s) to a specific network.
Foreign RADIUS Server Configuration	
Foreign RADIUS Authentication server	IP Address: Specifies the IP address of the foreign RADIUS accounting server that the source context will communicate with to provide subscriber accounting functions. Multiple addresses are needed if configuring multiple RADIUS servers. Foreign RADIUS accounting servers are configured within the source context. Multiple servers can be configured and each assigned a priority.
	Shared Secret: The shared secret is a string between 1 and 15 characters (alpha and/or numeric) that specifies the key exchanged between the RADIUS accounting server and the source context. A shared secret is needed for each configured RADIUS server.
	UDP Port Number: Specifies the port used by the source context and the foreign RADIUS Accounting server for communications. The UDP port number can be any integer value between 1 and 65535. The default value is 1812.
Foreign RADIUS Accounting server	IP Address: Specifies the IP address of the foreign RADIUS accounting server that the source context will communicate with to provide subscriber accounting functions. Multiple addresses are needed if configuring multiple RADIUS servers. Foreign RADIUS accounting servers are configured within the source context. Multiple servers can be configured and each assigned a priority.
	Shared Secret: The shared secret is a string between 1 and 15 characters (alpha and/or numeric) that specifies the key exchanged between the RADIUS accounting server and the source context. A shared secret is needed for each configured RADIUS server.
	UDP Port Number: Specifies the port used by the source context and the foreign RADIUS Accounting server for communications. The UDP port number can be any integer value between 1 and 65535. The default value is 1813.


■ Example 1: Mobile IP Support Using the System as a PDSN/FA


Required Information	Description
RADIUS attribute NAS Identifier	Specifies the name by which the source context will be identified in the Access-Request message(s) it sends to the foreign RADIUS server. The name must be between 1 and 32 alpha and/or numeric characters and is case sensitive.
RADIUS NAS IP address	Specifies the IP address of the source context's AAA interface. A secondary address can be optionally configured.

Mobile IP Destination Context Configuration

The following table lists the information required to configure the destination context.

Table 15. Required Information for Destination Context Configuration




Required Information	Description
Mobile IP destination context name	<p>This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the destination context will be recognized by the system.</p> <div>  Important: For this configuration, the destination context name should not match the domain name of a specific domain. It should, however, match the name of the context in which the HA service is configured if a separate system is used to provide HA functionality. </div>
Pi Interface Configuration	
Pi interface name	<p>This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the interface will be recognized by the system.</p> <p>Multiple names are needed if multiple interfaces will be configured.</p> <p>Pi interfaces are configured in the destination context.</p>
IP address and subnet	<p>These will be assigned to the Pi interface.</p> <p>Multiple addresses and/or subnets are needed if multiple interfaces will be configured.</p>
Physical port number	<p>This specifies the physical port to which the interface will be bound. Ports are identified by the chassis slot number where the line card resides in, followed by the number of the physical connector on the line card. For example, port 17/1 identifies connector number 1 on the card in slot 17.</p> <p>A single physical port can facilitate multiple interfaces.</p>
Physical port description	<p>This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the physical port will be recognized by the system.</p> <p>Multiple descriptions are needed if multiple ports will be used.</p> <p>Physical ports are configured within the destination context and are used to bind logical Pi interfaces.</p>
Gateway IP address(es)	Used when configuring static routes from the Pi interface(s) to a specific network.
FA Service Configuration	
FA service name	<p>This is an identification string between 1 and 63 characters (alpha and/or numeric) by which the FA service will be recognized by the system.</p> <p>Multiple names are needed if multiple FA services will be used.</p> <p>FA services are configured in the destination context.</p>

Required Information	Description
UDP port number for Mobile IP traffic	Specifies the port used by the FA service and the HA for communications. The UDP port number can be any integer value between 1 and 65535. The default value is 434.
Security Parameter Index (indices) Information	HA IP address: Specifies the IP address of the HAs with which the FA service communicates. The FA service allows the creation of a security profile that can be associated with a particular HA.
	Index: Specifies the shared SPI between the FA service and a particular HA. The SPI can be configured to any integer value between 256 and 4294967295. Multiple SPIs can be configured if the FA service is to communicate with multiple HAs.
	Secrets: Specifies the shared SPI secret between the FA service and the HA. The secret can be between 1 and 127 characters (alpha and/or numeric). An SPI secret is required for each SPI configured.
	Hash-algorithm: Specifies the algorithm used to hash the SPI and SPI secret. The possible algorithms that can be configured are MD5 per RFC 1321 and keyed-MD5 per RFC 2002. The default is hmac-md5. A hash-algorithm is required for each SPI configured.
FA agent advertisement lifetime	Specifies the time (in seconds) that an FA agent advertisement remains valid in the absence of further advertisements. The time can be configured to any integer value between 1 and 65535. The default is 9000.
Number of allowable unanswered FA advertisements	Specifies the number of unanswered agent advertisements that the FA service will allow during call setup before it will reject the session. The number can be any integer value between 1 and 65535. The default is 5.
Maximum mobile-requested registration lifetime allowed	Specifies the longest registration lifetime that the FA service will allow in any Registration Request message from the mobile node. The lifetime is expressed in seconds and can be configured between 1 and 65534. An infinite registration lifetime can be configured by disabling the timer. The default is 600 seconds.
Registration reply timeout	Specifies the amount of time that the FA service will wait for a Registration Reply from an HA. The time is measured in seconds and can be configured to any integer value between 1 and 65535. The default is 7.
Number of simultaneous registrations	Specifies the number of simultaneous Mobile IP sessions that will be supported for a single subscriber. The maximum number of sessions is 3. The default is 1. <div>  Important: The system will only support multiple Mobile IP sessions per subscriber if the subscriber's mobile node has a static IP address. </div>
Mobile node re-registration requirements	Specifies how the system should handle authentication for mobile node re-registrations. The FA service can be configured to always require authentication or not. If not, the initial registration and de-registration will still be handled normally.

System-Level AAA Configuration

The following table lists the information that is required to configure the system-level AAA parameters.

Table 16. Required Information for System-Level AAA Configuration

Required Information	Description
Subscriber default domain name	<p>Specifies the name of a context that can provide AAA functions in the event that the domain-part of the username is missing or poorly formed. This parameter will be applied to all subscribers if their domain can not be determined from their username regardless of what domain they are trying to access.</p> <hr/> <p> Important: The default domain name can be the same as the source context.</p>
Subscriber Last-resort context	<p>Specifies the name of a context that can provide AAA functions in the event that the domain-part of the username was present but does not match the name of a configured destination context. This parameter will be applied to all subscribers if their specified domain does not match a configured destination context regardless of what domain they are trying to access.</p> <hr/> <p> Important: The last-resort context name can be the same as the source context.</p>
Subscriber username format	<p>Specifies the format of subscriber usernames as to whether or not the username or domain is specified first and the character that separates them. The possible separator characters are:</p> <ul style="list-style-type: none"> • @ • % • - • \ • # • / <p>Up to six username formats can be specified. The default is <i>username @</i>.</p> <hr/> <p> Important: The username string is searched from right to left for the separator character. Therefore, if there is one or more separator characters in the string, only the first one that is recognized is considered the actual separator. For example, if the default username format was used, then for the username string <i>user1@enterprise@isp1</i>, the system resolves to the username <i>user1@enterprise</i> with domain <i>isp1</i>.</p>


Optional Destination Context

The following table lists the information required to configure the optional destination context. As discussed previously, This context is required if: 1) reverse tunneling is disabled in the FA service, or 2) if access control lists (ACLs) are used.



Important: If ACLs are used, the destination context would only consist of the ACL configuration. Interface configuration would not be required.

Table 17. Required Information for Destination Context Configuration

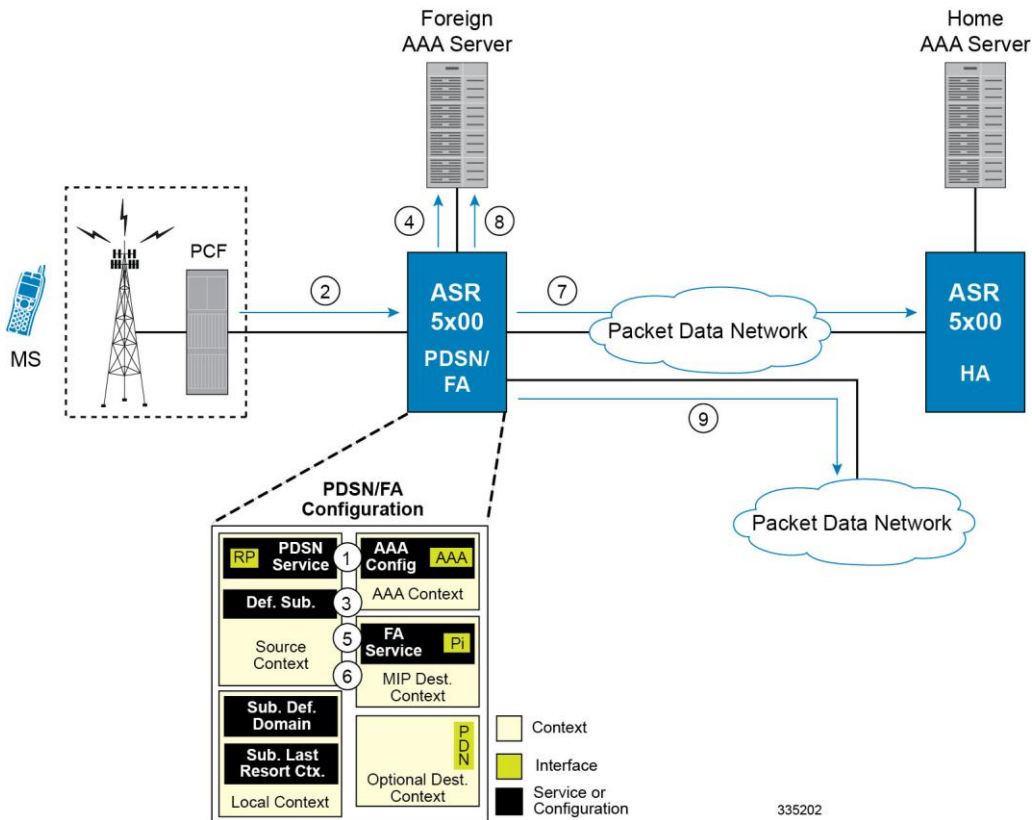
Required Information	Description
Destination context name	<p>This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the destination context will be recognized by the system.</p> <p> Important: For this configuration, the destination context name should not match the domain name of a specific domain.</p>
PDN Interface Configuration	
PDN interface name	<p>This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the interface will be recognized by the system.</p> <p>Multiple names are needed if multiple interfaces will be configured.</p> <p>PDN interfaces are configured in the destination context.</p>
IP address and subnet	<p>These will be assigned to the PDN interface.</p> <p>Multiple addresses and/or subnets are needed if multiple interfaces will be configured.</p>
Physical port number	<p>This specifies the physical port to which the interface will be bound. Ports are identified by the chassis slot number where the line card resides in, followed by the number of the physical connector on the line card. For example, port 17/1 identifies connector number 1 on the card in slot 17.</p> <p>A single physical port can facilitate multiple interfaces.</p>
Physical port description	<p>This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the physical port will be recognized by the system.</p> <p>Multiple descriptions are needed if multiple ports will be used.</p> <p>Physical ports are configured within the destination context and are used to bind logical PDN interfaces.</p>
Gateway IP address(es)	Used when configuring static routes from the PDN interface(s) to a specific network.
IP Address Pool Configuration	
IP address pool name	<p>Each IP address pool is identified by a name. The pool name can be between 1 and 31 alpha and/or numeric characters and is case sensitive.</p> <p>IP address pools are configured in the destination context(s). Multiple address pools can be configured within a single context.</p>
IP pool addresses	<p>An initial address and a subnet, or a starting address and an ending address, are required for each configured pool. The pool will then consist of every possible address within the subnet, or all addresses from the starting address to the ending address.</p> <p>The pool can be configured as public, private, or static.</p> <p>If this IP pool is being used for Interchassis Session Recovery, it must be a static and srp-activated.</p>

Example 1: Mobile IP Support Using the System as a PDSN/FA

How This Configuration Works

The following figure and the text that follows describe how this configuration with a single source and destination context would be used by the system to process a Mobile IP data call.

Figure 18. Call Processing When Using the system as a PDSN/FA



- The system-level AAA settings were configured as follows:
 - Subscriber default domain name = *AAA context*
 - Subscriber username format = *username @*
 - Subscriber last-resort context name = *AAA context*
- A subscriber session from the PCF is received by the PDSN service over the R-P interface.
- The PDSN service determines which context to use to provide foreign AAA functionality for the session. This process is described in the How the System Selects Contexts section located in the Understanding the System Operation and Configuration chapter of the *System Administration Guide*.
For this example, the result of this process is that PDSN service determined that foreign AAA functionality should be provided by the *AAA context*.
- The system then communicates with the foreign AAA server specified in the AAA context's AAA configuration to authenticate the subscriber.
- Upon successful authentication, the PDSN service determines the IP address of the subscriber's HA using either an attribute returned in the Access Accept message, or the address specified by the mobile.

The PDSN service uses the Mobile IP FA context name to determine what destination context is facilitating the FA service. In this example, it determines that it must use the *MIP Destination* context.

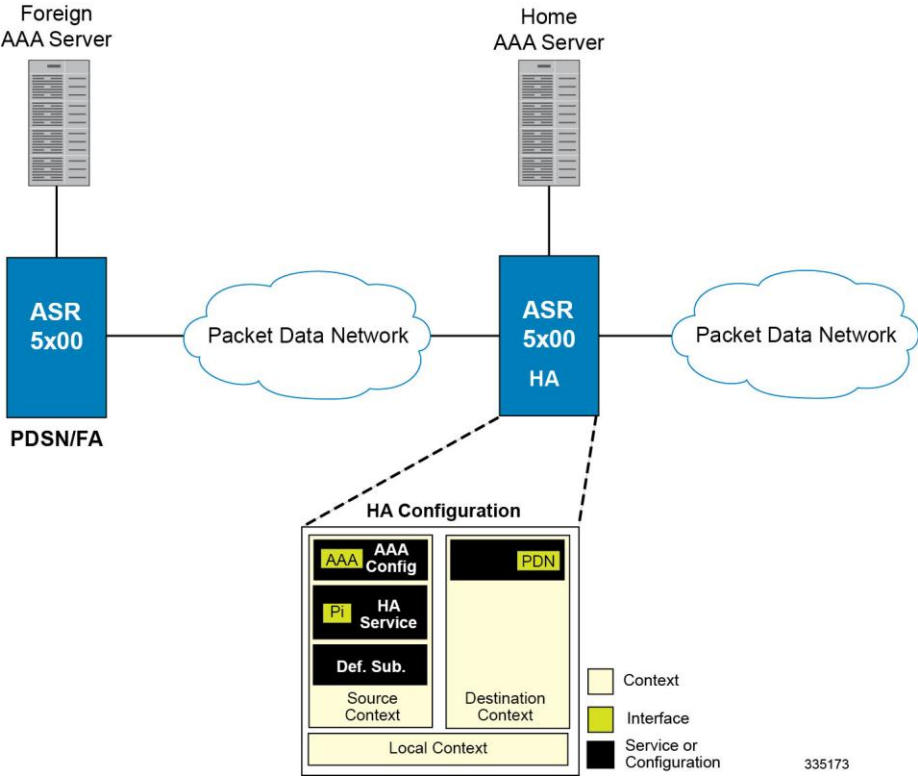
6. The PDSN service passes the HA IP address to the FA service.
7. The FA service then establishes a connection to the specified HA over the Pi interface.
8. Accounting messages for the session are sent to the Foreign AAA server over the AAA interface.
9. If reverse tunneling is disabled, then subscriber data traffic would have been routed over the PDN interface configured in the *Optional Destination* context.

Example 2: Mobile IP Support Using the System as an HA

The system supports both Simple and Mobile IP. For Mobile IP applications, the system can be configured to perform the function of a PDSN/FA and/or a HA. This example describes what is needed for and how the system performs the role of the HA. Example number 1 provides information on using the system to provide PDSN/FA functionality.

The system’s HA configuration for Mobile IP applications requires that at least two contexts (one source and one destination) be configured as shown in the following figure .

Figure 19. Mobile IP Support Using the system as an HA



The source context will facilitate the HA service(s), the Pi interfaces from the FA, and the AAA interfaces. The source context will also be configured to provide Home AAA functionality for subscriber sessions. The destination context will facilitate the PDN interface(s).


Information Required

Prior to configuring the system as shown in this example, there is a minimum amount of information required. The following sections describe the information required to configure the source and destination contexts.

Source Context Configuration

The following table lists the information that is required to configure the source context.

Table 18. Required Information for Source Context Configuration


Required Information	Description
Source context name	<p>This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the source context will be recognized by the system.</p> <hr/> <p> Important: The name of the source context should be the same as the name of the context in which the FA-context is configured if a separate system is being used to provide PDSN/FA functionality.</p> <hr/>
Pi Interface Configuration	
Pi interface name	<p>This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the interface will be recognized by the system.</p> <p>Multiple names are needed if multiple interfaces will be configured.</p> <p>Pi interfaces are configured in the source context.</p> <p>If this interface is being used for Interchassis Session Recovery, you must specify a loopback interface type after the interface_name.</p>
IP address and subnet	<p>These will be assigned to the Pi interface.</p> <p>Multiple addresses and/or subnets are needed if multiple interfaces will be configured.</p>
Physical port number	<p>This specifies the physical port to which the interface will be bound. Ports are identified by the chassis slot number where the line card resides in, followed by the number of the physical connector on the line card. For example, port 17/1 identifies connector number 1 on the card in slot 17.</p> <p>A single physical port can facilitate multiple interfaces.</p>
Physical port description	<p>This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the physical port will be recognized by the system.</p> <p>Multiple descriptions are needed if using multiple ports.</p> <p>Physical ports are configured within the source context and are used to bind logical Pi interfaces.</p>
Gateway IP address	Used when configuring static routes from the R-P interface(s) to a specific network.
HA service Configuration	
HA service name	<p>This is an identification string between 1 and 63 characters (alpha and/or numeric) by which the HA service will be recognized by the system.</p> <p>Multiple names are needed if multiple HA services will be used.</p> <p>HA services are configured in the destination context.</p>
UDP port number for Mobile IP traffic	Specifies the port used by the HA service and the FA for communications. The UDP port number can be any integer value between 1 and 65535. The default value is 434.
Mobile node re-registration requirements	<p>Specifies how the system should handle authentication for mobile node re-registrations.</p> <p>The HA service can be configured as follows:</p> <ul style="list-style-type: none"> Always require authentication Never require authentication (NOTE: the initial registration and de-registration will still be handled normally) Never look for mn-aaa extension Not require authentication but will authenticate if mn-aaa extension present

■ Example 2: Mobile IP Support Using the System as an HA

Required Information	Description
FA-to-HA Security Parameter Index Information	<p>FA IP address: The HA service allows the creation of a security profile that can be associated with a particular FA. This specifies the IP address of the FA that the HA service will be communicating with. Multiple FA addresses are needed if the HA will be communicating with multiple FAs.</p>
	<p>Index: Specifies the shared SPI between the HA service and a particular FA. The SPI can be configured to any integer value between 256 and 4294967295. Multiple SPIs can be configured if the HA service is to communicate with multiple FAs.</p>
	<p>Secret: Specifies the shared SPI secret between the HA service and the FA. The secret can be between 1 and 127 characters (alpha and/or numeric). An SPI secret is required for each SPI configured.</p>
	<p>Hash-algorithm: Specifies the algorithm used to hash the SPI and SPI secret. The possible algorithms that can be configured are MD5 per RFC 1321 and keyed-MD5 per RFC 2002. The default algorithm is hmac-md5. A hash-algorithm is required for each SPI configured.</p>
Mobile Node Security Parameter Index Information	<p>Index: Specifies the shared SPI between the HA service and the mobile node(s). The SPI can be configured to any integer value between 256 and 4294967295. Multiple SPIs can be configured if the HA service is to communicate with multiple mobile nodes.</p>
	<p>Secret(s): Specifies the shared SPI secret between the HA service and the mobile node. The secret can be between 1 and 127 characters (alpha and/or numeric). An SPI secret is required for each SPI configured.</p>
	<p>Hash-algorithm: Specifies the algorithm used to hash the SPI and SPI secret. The possible algorithms that can be configured are MD5 per RFC 1321 and keyed-MD5 per RFC 2002. The default algorithm is hmac-md5. A hash-algorithm is required for each SPI configured.</p>
	<p>Replay-protection process: Specifies how protection against replay-attacks is implemented. The possible processes are nonce and timestamp. The default is timestamp with a tolerance of 60 seconds. A replay-protection process is required for each mobile node-to-HA SPI configured.</p>
Maximum registration lifetime	<p>Specifies the longest registration lifetime that the HA service will allow in any Registration Request message from the mobile node. The time is measured in seconds and can be configured to any integer value between 1 and 65534. An infinite registration lifetime can also be configured by disabling the timer. The default is 600.</p>
Maximum number of simultaneous bindings	<p>Specifies the maximum number of “care-of” addresses that can simultaneously be bound for the same user as identified by NAI and Home address. The number can be configured to any integer value between 1 and 5. The default is 3.</p>
AAA Interface Configuration	
AAA interface name	<p>This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the interface will be recognized by the system. Multiple names are needed if multiple interfaces will be configured. AAA interfaces will be configured in the source context.</p>

Required Information	Description
IP address and subnet	These will be assigned to the AAA interface. Multiple addresses and/or subnets are needed if multiple interfaces will be configured.
Physical port number	This specifies the physical port to which the interface will be bound. Ports are identified by the chassis slot number where the line card resides in, followed by the number of the physical connector on the line card. For example, port 17/1 identifies connector number 1 on the card in slot 17. A single physical port can facilitate multiple interfaces.
Physical port description	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the physical port will be recognized by the system. Multiple descriptions are needed if multiple ports will be used. Physical ports are configured within the source context and are used to bind logical AAA interfaces.
Gateway IP address	Used when configuring static routes from the AAA interface(s) to a specific network.
Home RADIUS Server Configuration	
Home RADIUS Authentication server	IP Address: Specifies the IP address of the home RADIUS authentication server the source context will communicate with to provide subscriber authentication functions. Multiple addresses are needed if multiple RADIUS servers will be configured. Home RADIUS authentication servers are configured within the source context. Multiple servers can be configured and each assigned a priority.
	Shared Secret: The shared secret is a string between 1 and 15 characters (alpha and/or numeric) that specifies the key that is exchanged between the RADIUS authentication server and the source context. A shared secret is needed for each configured RADIUS server.
	UDP Port Number: Specifies the port used by the source context and the home RADIUS authentication server for communications. The UDP port number can be any integer value between 1 and 65535. The default value is 1812.
Home RADIUS Accounting server	IP Address: Specifies the IP address of the home RADIUS accounting server that the source context will communicate with to provide subscriber accounting functions. Multiple addresses are needed if multiple RADIUS servers will be configured. Home RADIUS accounting servers are configured within the source context. Multiple servers can be configured and each assigned a priority.
	Shared Secret: The shared secret is a string between 1 and 15 characters (alpha and/or numeric) that specifies the key that is exchanged between the RADIUS accounting server and the source context. A shared secret is needed for each configured RADIUS server.
	UDP Port Number: Specifies the port used by the source context and the home RADIUS Accounting server for communications. The UDP port number can be any integer value between 1 and 65535. The default value is 1813.
RADIUS attribute NAS Identifier	Specifies the name by which the source context will be identified in the Access-Request message(s) it sends to the home RADIUS server. The name must be between 1 and 32 alpha and/or numeric characters and is case sensitive.
RADIUS NAS IP address	Specifies the IP address of the source context's AAA interface. A secondary address can be optionally configured.
Default Subscriber Configuration	


■ Example 2: Mobile IP Support Using the System as an HA

Required Information	Description
“Default” subscriber’s IP context name	<p>Specifies the name of the egress context on the system that facilitates the PDN ports.</p> <hr/> <p> Important: For this configuration, the IP context name should be identical to the name of the destination context.</p> <hr/>

Destination Context Configuration

The following table lists the information required to configure the destination context.

Table 19. Required Information for Destination Context Configuration

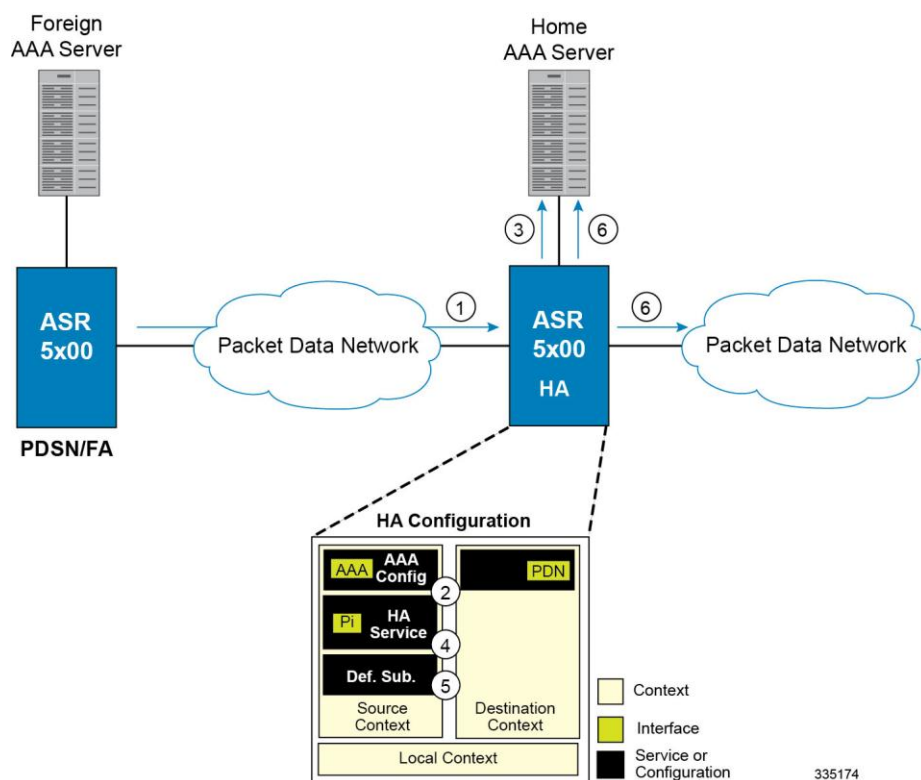
Required Information	Description
Destination context name	<p>This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the destination context will be recognized by the system.</p> <hr/> <p> Important: For this configuration, the destination context name should not match the domain name of a specific domain.</p> <hr/>
PDN Interface Configuration	
PDN interface name	<p>This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the interface will be recognized by the system.</p> <p>Multiple names are needed if multiple interfaces will be configured.</p> <p>PDN interfaces are configured in the destination context.</p>
IP address and subnet	<p>These will be assigned to the PDN interface.</p> <p>Multiple addresses and/or subnets are needed if multiple interfaces will be configured.</p>
Physical port number	<p>This specifies the physical port to which the interface will be bound. Ports are identified by the chassis slot number where the line card resides in, followed by the number of the physical connector on the line card. For example, port 17/1 identifies connector number 1 on the card in slot 17.</p> <p>A single physical port can facilitate multiple interfaces.</p>
Physical port description	<p>This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the physical port will be recognized by the system.</p> <p>Multiple descriptions are needed if multiple ports will be used.</p> <p>Physical ports are configured within the destination context and are used to bind logical PDN interfaces.</p>
Gateway IP address(es)	<p>Used when configuring static routes from the PDN interface(s) to a specific network.</p>
IP Address Pool Configuration	

Required Information	Description
IP address pool name	Each IP address pool is identified by a name. The pool name can be between 1 and 31 alpha and/or numeric characters and is case sensitive. IP address pools are configured in the destination context(s). Multiple address pools can be configured within a single context.
IP pool addresses	An initial address and a subnet, or a starting address and an ending address, are required for each configured pool. The pool will then consist of every possible address within the subnet, or all addresses from the starting address to the ending address. The pool can be configured as public, private, or static. If this IP pool is being used for Interchassis Session Recovery, it must be a static and srp-activated.

How This Configuration Works

The following figure and the text that follows describe how this configuration with a single source and destination context would be used by the system to process a Mobile IP data call.

Figure 20. Call Processing When Using the system as a PDSN/FA



1. The system-level AAA settings were configured as follows:

- Subscriber default domain name = *AAA context*
- Subscriber username format = *username @*

■ Example 2: Mobile IP Support Using the System as an HA

- Subscriber last-resort context name = *AAA context*
2. A subscriber session from the PCF is received by the PDSN service over the R-P interface.
 3. The PDSN service determines which context to use to provide foreign AAA functionality for the session. This process is described in the How the System Selects Contexts section located in the Understanding the System Operation and Configuration chapter of the *System Administration Guide*.

For this example, the result of this process is that PDSN service determined that foreign AAA functionality should be provided by the *AAA context*.
 4. The system then communicates with the foreign AAA server specified in the AAA context's AAA configuration to authenticate the subscriber.
 5. Upon successful authentication, the PDSN service determines the IP address of the subscriber's HA using either an attribute returned in the Access Accept message, or the address specified by the mobile.

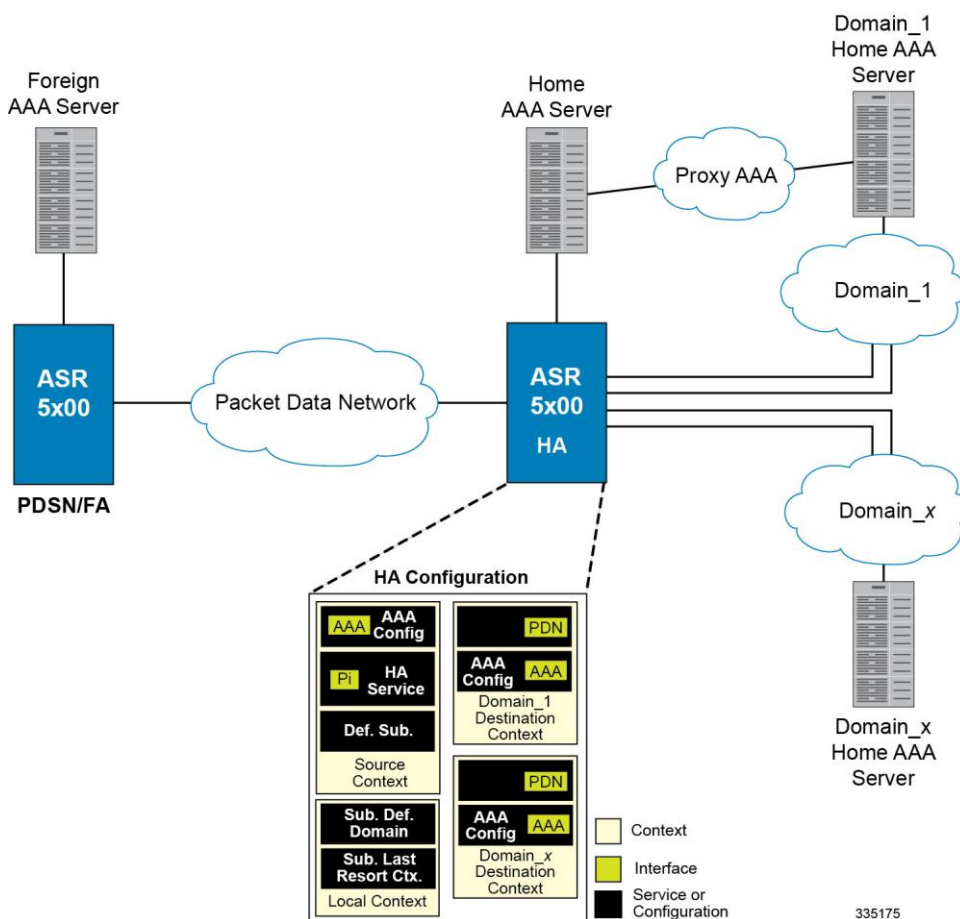
The PDSN service uses the Mobile IP FA context name to determine what destination context is facilitating the FA service. In this example, it determines that it must use the *MIP Destination* context.
 6. The PDSN service passes the HA IP address to the FA service.
 7. The FA service then establishes a connection to the specified HA over the Pi interface.
 8. Accounting messages for the session are sent to the Foreign AAA server over the AAA interface.
 9. If reverse tunneling is disabled, then subscriber data traffic would have been routed over the PDN interface configured in the *Optional Destination* context.

Example 3: HA Using a Single Source Context and Multiple Outsourced Destination Contexts

The system allows the wireless carrier to easily generate additional revenue by providing the ability to configure separate contexts that can then be leased or outsourced to various enterprises or ISPs, each having a specific domain.

In order to perform the role of an HA and support multiple outsourced domains, the system must be configured with at least one source context and multiple destination contexts as shown in the following figure. The AAA servers could be owned/maintained by either the carrier or the domain. If they are owned by the domain, the carrier will have to receive the AAA information via proxy.

Figure 21. The system as an HA Using a Single Source Context and Multiple Outsourced Destination Contexts



The source context will facilitate the HA service(s), and the Pi interface(s) to the FA(s). The source context will also be configured with AAA interface(s) and to provide Home AAA functionality for subscriber sessions. The destination contexts will each be configured to facilitate PDN interfaces. In addition, because each of the destination contexts can be outsourced to different domains, they will also be configured with AAA interface(s) and to provide AAA functionality for that domain.

■ Example 3: HA Using a Single Source Context and Multiple Outsourced Destination Contexts

In addition to the source and destination contexts, there are additional system-level AAA parameters that must be configured.

Information Required

Prior to configuring the system as shown in this example, there is a minimum amount of information required. The following sections describe the information required to configure the source and destination contexts.

Source Context Configuration

The following table lists the information that is required to configure the source context.


Table 20. Required Information for Source Context Configuration

Required Information	Description
Source context name	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the source context will be recognized by the system.
Pi Interface Configuration	
Piinterface name	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the interface will be recognized by the system. Multiple names are needed if multiple interfaces will be configured. Pi interfaces are configured in the source context.
IP address and subnet	These will be assigned to the Pi interface. Multiple addresses and/or subnets are needed if multiple interfaces will be configured.
Physical port number	This specifies the physical port to which the interface will be bound. Ports are identified by the chassis slot number where the line card resides in, followed by the number of the physical connector on the line card. For example, port 17/1 identifies connector number 1 on the card in slot 17. A single physical port can facilitate multiple interfaces.
Physical port description	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the physical port will be recognized by the system. Multiple descriptions are needed if using multiple ports. Physical ports are configured within the source context and are used to bind logical Pi interfaces.
Gateway IP address	Used when configuring static routes from the Pi interface(s) to a specific network.
HA service Configuration	
HA service name	This is an identification string between 1 and 63 characters (alpha and/or numeric) by which the PDSN service will be recognized by the system. Multiple names are needed if using multiple HA services. HA services are configured in the source context.
UDP port number for Mobile IP traffic	Specifies the port used by the HA service and the FA for communications. The UDP port number can be any integer value between 1 and 65535. The default value is 434.

Required Information	Description
Mobile node re-registration requirements	<p>Specifies how the system should handle authentication for mobile node re-registrations. The HA service can be configured as follows:</p> <ul style="list-style-type: none"> Always require authentication Never require authentication (NOTE: the initial registration and de-registration will still be handled normally) Never look for mn-aaa extension Not require authentication but will authenticate if mn-aaa extension present
FA-to-HA Security Parameter Index Information	<p>FA IP address: The HA service allows the creation of a security profile that can be associated with a particular FA. This specifies the IP address of the FA that the HA service will be communicating with. Multiple FA addresses are needed if the HA will be communicating with multiple FAs.</p>
	<p>Index: Specifies the shared SPI between the HA service and a particular FA. The SPI can be configured to any integer value between 256 and 4294967295. Multiple SPIs can be configured if the HA service is to communicate with multiple FAs.</p>
	<p>Secret: Specifies the shared SPI secret between the HA service and the FA. The secret can be between 1 and 127 characters (alpha and/or numeric). An SPI secret is required for each SPI configured.</p>
	<p>Hash-algorithm: Specifies the algorithm used to hash the SPI and SPI secret. The possible algorithms that can be configured are MD5 per RFC 1321 and keyed-MD5 per RFC 2002. The default algorithm is hmac-md5. A hash-algorithm is required for each SPI configured.</p>
Mobile Node Security Parameter Index Information	<p>Index: Specifies the shared SPI between the HA service and the mobile node(s). The SPI can be configured to any integer value between 256 and 4294967295. Multiple SPIs can be configured if the HA service is to communicate with multiple mobile nodes.</p>
	<p>Secret(s): Specifies the shared SPI secret between the HA service and the mobile node. The secret can be between 1 and 127 characters (alpha and/or numeric). An SPI secret is required for each SPI configured.</p>
	<p>Hash-algorithm: Specifies the algorithm used to hash the SPI and SPI secret. The possible algorithms that can be configured are MD5 per RFC 1321 and keyed-MD5 per RFC 2002. The default algorithm is hmac-md5. A hash-algorithm is required for each SPI configured.</p>
	<p>Replay-protection process: Specifies how protection against replay-attacks is implemented. The possible processes are nonce and timestamp. The default is timestamp with a tolerance of 60 seconds. A replay-protection process is required for each mobile node-to-HA SPI configured.</p>
Maximum registration lifetime	<p>Specifies the longest registration lifetime that the HA service will allow in any Registration Request message from the mobile node. The time is measured in seconds and can be configured to any integer value between 1 and 65535. An infinite registration lifetime can also be configured by disabling the timer. The default is 600.</p>

■ Example 3: HA Using a Single Source Context and Multiple Outsourced Destination Contexts


Required Information	Description
Maximum number of simultaneous bindings	Specifies the maximum number of “care-of” addresses that can simultaneously be bound for the same user as identified by NAI and Home address. The number can be configured to any integer value between 1 and 5. The default is 3.
AAA Interface Configuration	
AAA interface name	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the interface will be recognized by the system. Multiple names are needed if multiple interfaces will be configured. AAA interfaces will be configured in the source context.
IP address and subnet	These will be assigned to the AAA interface. Multiple addresses and/or subnets are needed if multiple interfaces will be configured.
Physical port number	This specifies the physical port to which the interface will be bound. Ports are identified by the chassis slot number where the line card resides in, followed by the number of the physical connector on the line card. For example, port 17/1 identifies connector number 1 on the card in slot 17. A single physical port can facilitate multiple interfaces.
Physical port description	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the physical port will be recognized by the system. Multiple descriptions are needed if multiple ports will be used. Physical ports are configured within the source context and are used to bind logical AAA interfaces.
Gateway IP address	Used when configuring static routes from the AAA interface(s) to a specific network.
Home RADIUS Server Configuration	
Home RADIUS Authentication server	IP Address: Specifies the IP address of the home RADIUS authentication server the source context will communicate with to provide subscriber authentication functions. Multiple addresses are needed if multiple RADIUS servers will be configured. Home RADIUS authentication servers are configured within the source context. Multiple servers can be configured and each assigned a priority.
	Shared Secret: The shared secret is a string between 1 and 15 characters (alpha and/or numeric) that specifies the key that is exchanged between the RADIUS authentication server and the source context. A shared secret is needed for each configured RADIUS server.
	UDP Port Number: Specifies the port used by the source context and the home RADIUS authentication server for communications. The UDP port number can be any integer value between 1 and 65535. The default value is 1812.
Home RADIUS Accounting server	IP Address: Specifies the IP address of the home RADIUS accounting server that the source context will communicate with to provide subscriber accounting functions. Multiple addresses are needed if multiple RADIUS servers will be configured. Home RADIUS accounting servers are configured within the source context. Multiple servers can be configured and each assigned a priority.
	Shared Secret: The shared secret is a string between 1 and 15 characters (alpha and/or numeric) that specifies the key that is exchanged between the RADIUS accounting server and the source context. A shared secret is needed for each configured RADIUS server.

Required Information	Description
	<p>UDP Port Number:</p> <p>Specifies the port used by the source context and the home RADIUS Accounting server for communications. The UDP port number can be any integer value between 1 and 65535. The default value is 1813.</p>
RADIUS attribute NAS Identifier	Specifies the name by which the source context will be identified in the Access-Request message(s) it sends to the home RADIUS server. The name must be between 1 and 32 alpha and/or numeric characters and is case sensitive.
RADIUS NAS IP address	Specifies the IP address of the source context's AAA interface. A secondary address can be optionally configured.
Default Subscriber Configuration	
"Default" subscriber's IP context name	<p>Specifies the name of the egress context on the system that facilitates the PDN ports.</p> <hr/> <p> Important: For this configuration, the IP context name should be identical to the name of the destination context.</p> <hr/>

Destination Context Configuration

The following table lists the information required to configure the destination context.

Table 21. Required Information for Destination Context Configuration

Required Information	Description
Destination context name	<p>This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the destination context will be recognized by the system.</p> <hr/> <p> Important: For this configuration, the destination context name should not match the domain name of a specific domain.</p> <hr/>
PDN Interface Configuration	
PDN interface name	<p>This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the interface will be recognized by the system.</p> <p>Multiple names are needed if multiple interfaces will be configured.</p> <p>PDN interfaces are configured in the destination context.</p>
IP address and subnet	<p>These will be assigned to the PDN interface.</p> <p>Multiple addresses and/or subnets are needed if multiple interfaces will be configured.</p>
Physical port number	<p>This specifies the physical port to which the interface will be bound. Ports are identified by the chassis slot number where the line card resides in, followed by the number of the physical connector on the line card. For example, port 17/1 identifies connector number 1 on the card in slot 17.</p> <p>A single physical port can facilitate multiple interfaces.</p>

■ Example 3: HA Using a Single Source Context and Multiple Outsourced Destination Contexts


Required Information	Description
Physical port description	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the physical port will be recognized by the system. Multiple descriptions are needed if multiple ports will be used. Physical ports are configured within the destination context and are used to bind logical PDN interfaces.
Gateway IP address(es)	Used when configuring static routes from the PDN interface(s) to a specific network.
IP Address Pool Configuration	
IP address pool name	Each IP address pool is identified by a name. The pool name can be between 1 and 31 alpha and/or numeric characters and is case sensitive. IP address pools are configured in the destination context(s). Multiple address pools can be configured within a single context.
IP pool addresses	An initial address and a subnet, or a starting address and an ending address, are required for each configured pool. The pool will then consist of every possible address within the subnet, or all addresses from the starting address to the ending address. The pool can be configured as public, private, or static. If this IP pool is being used for Interchassis Session Recovery, it must be a static and srp-activated.
AAA Interface Configuration	
AAA interface name	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the interface will be recognized by the system. Multiple names are needed if multiple interfaces will be configured. AAA interfaces will be configured in the source context.
IP address and subnet	These will be assigned to the AAA interface. Multiple addresses and/or subnets are needed if multiple interfaces will be configured.
Physical port number	This specifies the physical port to which the interface will be bound. Ports are identified by the chassis slot number where the line card resides in, followed by the number of the physical connector on the line card. For example, port 17/1 identifies connector number 1 on the card in slot 17. A single physical port can facilitate multiple interfaces.
Physical port description	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the physical port will be recognized by the system. Multiple descriptions are needed if multiple ports will be used. Physical ports are configured within the source context and are used to bind logical AAA interfaces.
Gateway IP address	Used when configuring static routes from the AAA interface(s) to a specific network.
Home RADIUS Server Configuration	
Home RADIUS Authentication server	IP Address: Specifies the IP address of the home RADIUS authentication server the source context will communicate with to provide subscriber authentication functions. Multiple addresses are needed if multiple RADIUS servers will be configured. Home RADIUS authentication servers are configured within the source context. Multiple servers can be configured and each assigned a priority.
	Shared Secret: The shared secret is a string between 1 and 15 characters (alpha and/or numeric) that specifies the key that is exchanged between the RADIUS authentication server and the source context. A shared secret is needed for each configured RADIUS server.

Required Information	Description
	<p>UDP Port Number: Specifies the port used by the source context and the home RADIUS authentication server for communications. The UDP port number can be any integer value between 1 and 65535. The default value is 1812.</p>
Home RADIUS Accounting server	<p>IP Address: Specifies the IP address of the home RADIUS accounting server that the source context will communicate with to provide subscriber accounting functions. Multiple addresses are needed if multiple RADIUS servers will be configured. Home RADIUS accounting servers are configured within the source context. Multiple servers can be configured and each assigned a priority.</p>
	<p>Shared Secret: The shared secret is a string between 1 and 15 characters (alpha and/or numeric) that specifies the key that is exchanged between the RADIUS accounting server and the source context. A shared secret is needed for each configured RADIUS server.</p>
	<p>UDP Port Number: Specifies the port used by the source context and the home RADIUS Accounting server for communications. The UDP port number can be any integer value between 1 and 65535. The default value is 1813.</p>
RADIUS attribute NAS Identifier	Specifies the name by which the source context will be identified in the Access-Request message(s) it sends to the home RADIUS server. The name must be between 1 and 32 alpha and/or numeric characters and is case sensitive.
RADIUS NAS IP address	Specifies the IP address of the source context's AAA interface. A secondary address can be optionally configured.



System-Level AAA Configuration

The following table lists the information that is required to configure the system-level AAA parameters.

Table 22. Required Information for System-Level AAA Configuration

Required Information	Description
Subscriber default domain name	<p>Specifies the name of a context that can provide AAA functions in the event that the domain-part of the username is missing or poorly formed. This parameter will be applied to all subscribers if their domain can not be determined from their username regardless of what domain they are trying to access.</p> <hr/> <p> Important: The default domain name can be the same as the source context.</p>

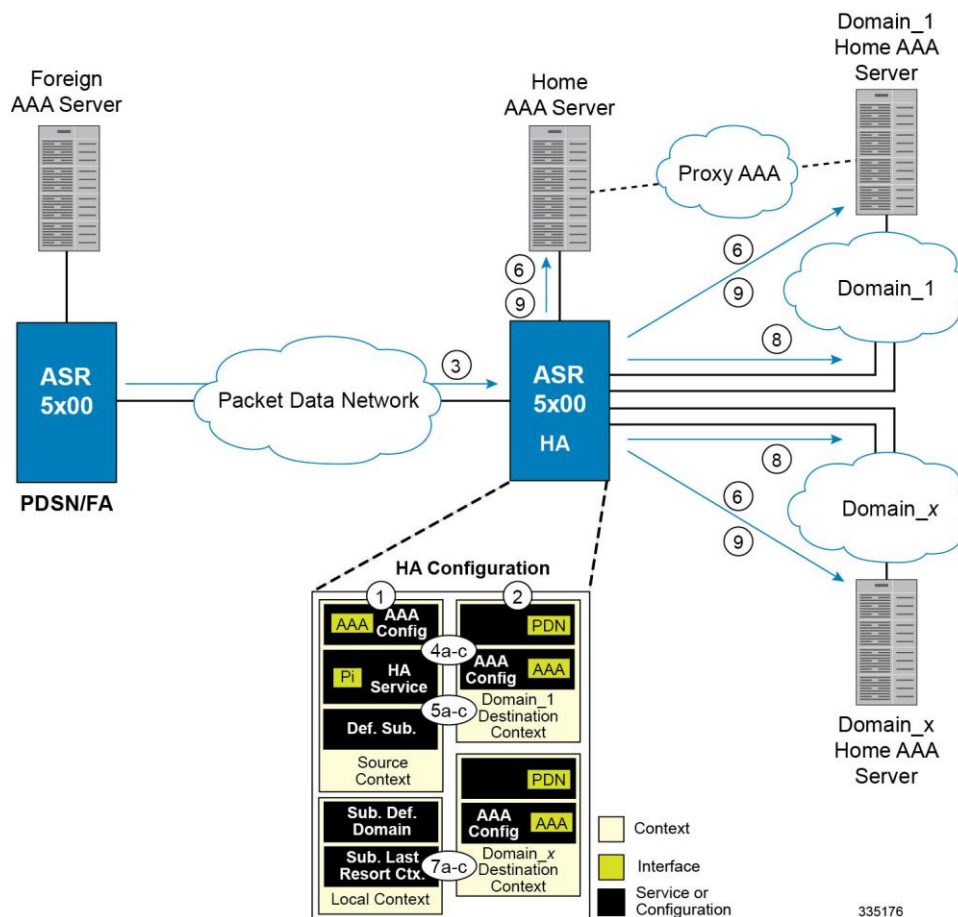
■ Example 3: HA Using a Single Source Context and Multiple Outsourced Destination Contexts

Required Information	Description
Subscriber Last-resort context	<p>Specifies the name of a context that can provide AAA functions in the event that the domain-part of the username was present but does not match the name of a configured destination context. This parameter will be applied to all subscribers if their specified domain does not match a configured destination context regardless of what domain they are trying to access.</p> <hr/> <p> Important: The last-resort context name can be the same as the source context.</p>
Subscriber username format	<p>Specifies the format of subscriber usernames as to whether or not the username or domain is specified first and the character that separates them. The possible separator characters are:</p> <ul style="list-style-type: none"> • @ • % • - • \ • # • / <p>Up to six username formats can be specified. The default is <i>username @</i>.</p> <hr/> <p> Important: The username string is searched from right to left for the separator character. Therefore, if there is one or more separator characters in the string, only the first one that is recognized is considered the actual separator. For example, if the default username format was used, then for the username string <i>user1@enterprise@isp1</i>, the system resolves to the username <i>user1@enterprise</i> with domain <i>isp1</i>.</p>

How This Configuration Works

The following figure and the text that follows describe how this configuration with a single source and destination context would be used by the system to process a Mobile IP data call.

Figure 22. Call Processing When Using the system as a PDSN/FA



1. The system-level AAA settings were configured as follows:
 - Subscriber default domain name = *AAA context*
 - Subscriber username format = *username @*
 - Subscriber last-resort context name = *AAA context*
2. A subscriber session from the PCF is received by the PDSN service over the R-P interface.
3. The PDSN service determines which context to use to provide foreign AAA functionality for the session. This process is described in the How the System Selects Contexts section located in the Understanding the System Operation and Configuration chapter of the *System Administration Guide*.

For this example, the result of this process is that PDSN service determined that foreign AAA functionality should be provided by the *AAA context*.
4. The system then communicates with the foreign AAA server specified in the AAA context's AAA configuration to authenticate the subscriber.
5. Upon successful authentication, the PDSN service determines the IP address of the subscriber's HA using either an attribute returned in the Access Accept message, or the address specified by the mobile.

The PDSN service uses the Mobile IP FA context name to determine what destination context is facilitating the FA service. In this example, it determines that it must use the *MIP Destination* context.

■ Example 3: HA Using a Single Source Context and Multiple Outsourced Destination Contexts

6. The PDSN service passes the HA IP address to the FA service.
7. The FA service then establishes a connection to the specified HA over the Pi interface.
8. Accounting messages for the session are sent to the Foreign AAA server over the AAA interface.

Chapter 5

Simple IP and Mobile IP in a Single System Configuration Example

This chapter provides information for several configuration examples that can be implemented on the system to support Simple IP and Mobile IP data services in a single system.



Important: This chapter does not discuss the configuration of the localout-of-band management context. Information about the localout-of-band management context can be found in Chapter 1 of *Command Line Reference*. Additionally, when configuring Mobile IP take into account the MIP timing considerations discussed in the section MIP Timer Considerations.

Using the System as Both a PDSN/FA and an HA

The system supports both Simple and Mobile IP. For Mobile IP applications, the system can be configured to perform the function of a Packet Data Service Node/Foreign Agent (PDSN/FA) and/or a Home Agent (HA). This example describes what is needed and how a single system simultaneously supports both of these functions.

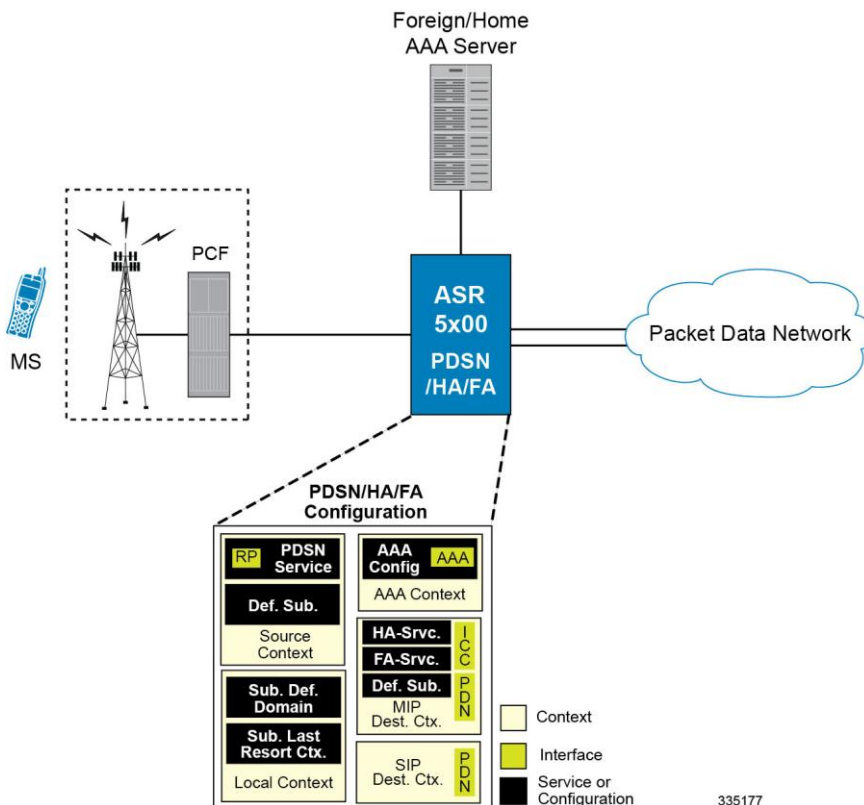
In order to support PDSN, FA, and HA functionality, the system must be configured with at least one source context and at least two destination contexts as shown in the following figure.

The source context will facilitate the PDSN service(s), and the R-P interfaces. The AAA context will be configured to provide foreign/home AAA functionality for subscriber sessions and facilitate the AAA interfaces.

The Mobile IP destination context will be configured to facilitate the FA service, the HA service and the PDN interfaces for Mobile IP data services. The Simple IP destination context will facilitate the PDN interfaces for Simple IP data Services.

In addition to the source and destination contexts, there are additional system-level AAA parameters that must be configured.

Figure 23. Simple and Mobile IP Support Within a Single System



Information Required

Prior to configuring the system as shown in this example, there is a minimum amount of information required. The following sections describe the required information to configure the source and destination contexts.

Source Context Configuration

The following table lists the information that is required to configure the source context.

Table 23. Required Information for Source Context Configuration

Required Information	Description
Source context name	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the source context will be recognized by the system.
R-P Interface Configuration	
R-P interface name	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the interface will be recognized by the system. Multiple names are needed if multiple interfaces will be configured. R-P interfaces are configured in the source context.
IP address and subnet	These will be assigned to the R-P interface. Multiple addresses and/or subnets are needed if multiple interfaces will be configured.
Physical port number	This specifies the physical port to which the interface will be bound. Ports are identified by the chassis slot number where the line card resides in, followed by the number of the physical connector on the line card. For example, port 17/1 identifies connector number 1 on the card in slot 17. A single physical port can facilitate multiple interfaces.
Physical port description	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the physical port will be recognized by the system. Multiple descriptions are needed if multiple ports will be used. Physical ports are configured within the source context and are used to bind logical R-P interfaces.
Gateway IP address	Used when configuring static routes from the R-P interface(s) to a specific network.
PDSN service Configuration	
PDSN service name	This is an identification string between 1 and 63 characters (alpha and/or numeric) by which the PDSN service will be recognized by the system. Multiple names are needed if multiple PDSN services will be used. PDSN services are configured in the source context.
UDP port number for R-P traffic	Specifies the port used by the PDSN service and the PCF for communications. The UDP port number and can be any integer value between 1 and 65535. The default value is 699.
Authentication protocols used	Specifies how the system handles authentication: using a protocol (such as CHAP, PAP, or MSCHAP), or not requiring any authentication.
Domain alias for NAI-construction	Specifies a context name for the system to use to provide accounting functionality for a subscriber session. This parameter is needed only if the system is configured to support no authentication.
Security Parameter Index Information	PCF IP address: Specifies the IP address of the PCF that the PDSN service will be communicating with. The PDSN service allows the creation of a security profile that can be associated with a particular PCF. Multiple IP addresses are needed if the PDSN service will be communicating with multiple PCFs.
	Index: Specifies the shared SPI between the PDSN service and a particular PCF. The SPI can be configured to any integer value between 256 and 4294967295. Multiple SPIs can be configured if the PDSN service is to communicate with multiple PCFs.

Required Information	Description
	Secret: Specifies the shared SPI secret between the PDSN service and the PCF. The secret can be between 1 and 127 characters (alpha and/or numeric). An SPI secret is required for each SPI configured.
	Hash-algorithm: Specifies the algorithm used to hash the SPI and SPI secret. The possible algorithms that can be configured are MD5 per RFC 1321 and keyed-MD5 per RFC 2002. The default is MD5. A hash-algorithm is required for each SPI configured.
	Replay-protection process: Specifies how protection against replay-attacks is implemented. The possible processes are nonce and timestamp. The default is timestamp with a tolerance of 60 seconds. A replay-protection process is required for each SPI configured.
Subscriber session lifetime	Specifies the time in seconds that an A10 connection can exist before its registration is considered expired. The time is expressed in seconds and can be configured to any integer value between 1 and 65534, or the timer can be disabled to set an infinite lifetime. The default value is 1800 seconds.
Mobile IP FA context name	Specifies the name of the context in which the FA service is configured.
Default Subscriber Configuration	
“Default” subscriber’s IP context name	Specifies the name of the egress context on the system that facilitates the PDN ports. NOTE: For this configuration, the IP context name should be identical to the name of the destination context.

AAA Context Configuration

The following table lists the information that is required to configure the AAA context.

Table 24. Required Information for AAA Context Configuration

Required Information	Description
AAA context name	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the AAA context will be recognized by the system.
AAA Interface Configuration	
AAA interface name	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the interface will be recognized by the system. Multiple names are needed if multiple interfaces will be configured. AAA interfaces will be configured in the source context.
IP address and subnet	These will be assigned to the AAA interface. Multiple addresses and/or subnets are needed if multiple interfaces will be configured.

Required Information	Description
Physical port number	This specifies the physical port to which the interface will be bound. Ports are identified by the chassis slot number where the line card resides in, followed by the number of the physical connector on the line card. For example, port 17/1 identifies connector number 1 on the card in slot 17. A single physical port can facilitate multiple interfaces.
Physical port description	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the physical port will be recognized by the system. Multiple descriptions are needed if multiple ports will be used. Physical ports are configured within the source context and are used to bind logical AAA interfaces.
Gateway IP address	Used when configuring static routes from the AAA interface(s) to a specific network.
Foreign/Home RADIUS Server Configuration	
Foreign/Home RADIUS Authentication server	IP Address: Specifies the IP address of the foreign/home RADIUS authentication server the source context will communicate with to provide subscriber authentication functions. Multiple addresses are needed if multiple RADIUS servers will be configured. Foreign/home RADIUS authentication servers are configured within the source context. Multiple servers can be configured and each assigned a priority.
	Shared Secret: The shared secret is a string between 1 and 15 characters (alpha and/or numeric) that specifies the key that is exchanged between the RADIUS authentication server and the source context. A shared secret is needed for each configured RADIUS server.
	UDP Port Number: Specifies the port used by the source context and the foreign/home RADIUS authentication server for communications. The UDP port number can be any integer value between 1 and 65535. The default value is 1812.
Foreign/Home RADIUS Accounting server	IP Address: Specifies the IP address of the foreign/home RADIUS accounting server that the source context will communicate with to provide subscriber accounting functions. Multiple addresses are needed if multiple RADIUS servers will be configured. Foreign/home RADIUS accounting servers are configured within the source context. Multiple servers can be configured and each assigned a priority.
	Shared Secret: The shared secret is a string between 1 and 15 characters (alpha and/or numeric) that specifies the key that is exchanged between the RADIUS accounting server and the source context. A shared secret is needed for each configured RADIUS server.
	UDP Port Number: Specifies the port used by the source context and the foreign/home RADIUS Accounting server for communications. The UDP port number can be any integer value between 1 and 65535. The default value is 1813.
RADIUS attribute NAS Identifier	Specifies the name by which the source context will be identified in the Access-Request message(s) it sends to the foreign/home RADIUS server. The name must be between 1 and 32 alpha and/or numeric characters and is case sensitive.
RADIUS NAS IP address	Specifies the IP address of the source context's AAA interface. A secondary IP address interface can optionally be configured.

Mobile IP Destination Context Configuration

The following table lists the information that is required to configure the destination context.

Table 25. Required Information for Destination Context Configuration

Required Information	Description
Mobile IP Destination context name	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the Mobile IP destination context will be recognized by the system. NOTE: For this configuration, the destination context name should not match the domain name of a specific domain. It should, however, match the name of the context in which the HA service is configured if a separate system is used to provide HA functionality.
ICC Interface Configuration	
ICC interface name	The intra-context communication (ICC) interface is configured to allow FA and HA services configured within the same context to communicate with each other. The ICC interface name is an identification string between 1 and 79 characters (alpha and/or numeric) by which the interface will be recognized by the system. Multiple names are needed if multiple interfaces will be configured. ICC interface(s) are configured in the same destination context as the FA and HA services.
IP address and subnet	These will be assigned to the ICC interface(s). Multiple addresses (at least one per service) on the same subnet will be needed to assign to the same ICC interface.
Physical port number	This specifies the physical port to which the interface will be bound. Ports are identified by the chassis slot number where the line card resides in, followed by the number of the physical connector on the line card. For example, port 17/1 identifies connector number 1 on the card in slot 17. A single physical port can facilitate multiple interfaces.
Physical port description	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the physical port will be recognized by the system. Multiple descriptions are needed if multiple ports will be used. Physical ports are configured within the destination context and are used to bind logical ICC interfaces.
PDN Interface Configuration	
PDN interface name	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the interface will be recognized by the system. Multiple names are needed if multiple interfaces will be configured. PDN interfaces are configured in the destination context.
IP address and subnet	These will be assigned to the PDN interface. Multiple addresses and/or subnets are needed if multiple interfaces will be configured.
Physical port number	This specifies the physical port to which the interface will be bound. Ports are identified by the chassis slot number where the line card resides in, followed by the number of the physical connector on the line card. For example, port 17/1 identifies connector number 1 on the card in slot 17. A single physical port can facilitate multiple interfaces.
Physical port description(s)	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the physical port will be recognized by the system. Multiple descriptions will be needed if multiple ports will be used. Physical ports are configured within the destination context and are used to bind logical PDN interfaces.

Required Information	Description
Gateway IP address(es)	Used when configuring static routes from the PDN interface(s) to a specific network.
IP Address Pool Configuration (optional)	
IP address pool name(s)	If IP address pools will be configured in the destination context(s), names or identifiers will be needed for them. The pool name can be between 1 and 31 alpha and/or numeric characters and is case sensitive.
IP pool addresses	An initial address and a subnet, or a starting address and an ending address, are required for each configured pool. The pool will then consist of every possible address within the subnet, or all addresses from the starting address to the ending address. The pool can be configured as public, private, or static.
FA Service Configuration	
FA service name	This is an identification string between 1 and 63 characters (alpha and/or numeric) by which the FA service will be recognized by the system. Multiple names are needed if multiple FA services will be used. FA services are configured in the destination context.
UDP port number for Mobile IP traffic	Specifies the port used by the FA service and the HA for communications. The UDP port number can be any integer value between 1 and 65535. The default value is 434.
Security Parameter Index (indices) Information	HA IP address: Specifies the IP address of the HAs with which the FA service communicates. The FA service allows the creation of a security profile that can be associated with a particular HA.
	Index: Specifies the shared SPI between the FA service and a particular HA. The SPI can be configured to any integer value between 256 and 4294967295. Multiple SPIs can be configured if the FA service is to communicate with multiple HAs.
	Secrets: Specifies the shared SPI secret between the FA service and the HA. The secret can be between 1 and 127 characters (alpha and/or numeric). An SPI secret is required for each SPI configured.
	Hash-algorithm: Specifies the algorithm used to hash the SPI and SPI secret. The possible algorithms that can be configured are MD5 per RFC 1321 and keyed-MD5 per RFC 2002. The default is hmac-md5. A hash-algorithm is required for each SPI configured.
FA agent advertisement lifetime	Specifies the time (in seconds) that an FA agent advertisement remains valid in the absence of further advertisements. The time can be configured to any integer value between 1 and 65535. The default is 9000.
Number of allowable unanswered FA advertisements	Specifies the number of unanswered agent advertisements that the FA service will allow during call setup before it will reject the session. The number can be any integer value between 1 and 65535. The default is 5.
Maximum mobile-requested registration lifetime allowed	Specifies the longest registration lifetime that the FA service will allow in any Registration Request message from the mobile node. The lifetime is expressed in seconds and can be configured between 1 and 65534. An infinite registration lifetime can be configured by disabling the timer. The default is 600 seconds.
Registration reply timeout	Specifies the amount of time that the FA service will wait for a Registration Reply from an HA. The time is measured in seconds and can be configured to any integer value between 1 and 65535. The default is 7.

■ Using the System as Both a PDSN/FA and an HA

Required Information	Description
Number of simultaneous registrations	Specifies the number of simultaneous Mobile IP sessions that will be supported for a single subscriber. The maximum number of sessions is 3. The default is 1. NOTE: The system will only support multiple Mobile IP sessions per subscriber if the subscriber's mobile node has a static IP address.
Mobile node re-registration requirements	Specifies how the system should handle authentication for mobile node re-registrations. The FA service can be configured to always require authentication or not. If not, the initial registration and de-registration will still be handled normally.
HA service Configuration	
HA service name	This is an identification string between 1 and 63 characters (alpha and/or numeric) by which the HA service will be recognized by the system. Multiple names are needed if multiple HA services will be used. HA services are configured in the destination context.
UDP port number for Mobile IP traffic	Specifies the port used by the HA service and the FA for communications. The UDP port number can be any integer value between 1 and 65535. The default value is 434.
Mobile node re-registration requirements	Specifies how the system should handle authentication for mobile node re-registrations. The HA service can be configured as follows: <ul style="list-style-type: none"> • Always require authentication • Never require authentication (NOTE: the initial registration and de-registration will still be handled normally) • Never look for mn-aaa extension • Not require authentication but will authenticate if mn-aaa extension present
FA-to-HA Security Parameter Index Information	FA IP address: The HA service allows the creation of a security profile that can be associated with a particular FA. This specifies the IP address of the FA that the HA service will be communicating with. Multiple FA addresses are needed if the HA will be communicating with multiple FAs.
	Index: Specifies the shared SPI between the HA service and a particular FA. The SPI can be configured to any integer value between 256 and 4294967295. Multiple SPIs can be configured if the HA service is to communicate with multiple FAs.
	Secret: Specifies the shared SPI secret between the HA service and the FA. The secret can be between 1 and 127 characters (alpha and/or numeric). An SPI secret is required for each SPI configured.
	Hash-algorithm: Specifies the algorithm used to hash the SPI and SPI secret. The possible algorithms that can be configured are MD5 per RFC 1321 and keyed-MD5 per RFC 2002. The default algorithm is hmac-md5. A hash-algorithm is required for each SPI configured.
Mobile Node Security Parameter Index Information	Index: Specifies the shared SPI between the HA service and the mobile node(s). The SPI can be configured to any integer value between 256 and 4294967295. Multiple SPIs can be configured if the HA service is to communicate with multiple mobile nodes.

Required Information	Description
	Secret(s): Specifies the shared SPI secret between the HA service and the mobile node. The secret can be between 1 and 127 characters (alpha and/or numeric). An SPI secret is required for each SPI configured.
	Hash-algorithm: Specifies the algorithm used to hash the SPI and SPI secret. The possible algorithms that can be configured are MD5 per RFC 1321 and keyed-MD5 per RFC 2002. The default algorithm is hmac-md5. A hash-algorithm is required for each SPI configured.
	Replay-protection process: Specifies how protection against replay-attacks is implemented. The possible processes are nonce and timestamp. The default is timestamp with a tolerance of 60 seconds. A replay-protection process is required for each mobile node-to-HA SPI configured.
Maximum registration lifetime	Specifies the longest registration lifetime that the HA service will allow in any Registration Request message from the mobile node. The time is measured in seconds and can be configured to any integer value between 1 and 65535. An infinite registration lifetime can also be configured by disabling the timer. The default is 600.
Maximum number of simultaneous bindings	Specifies the maximum number of “care-of” addresses that can simultaneously be bound for the same user as identified by NAI and Home address. The number can be configured to any integer value between 1 and 5. The default is 3.
Default Subscriber Configuration	
“Default” subscriber’s IP context name	Specifies the name of the egress context on the system that facilitates the PDN ports. NOTE: For this configuration, the IP context name should be identical to the name of the destination context.

Simple IP Destination Context

The following table lists the information that is required to configure the optional destination context. As discussed previously, This context is only required if Reverse Tunneling is disabled in the FA service.

Table 26. Required Information for Destination Context Configuration

Required Information	Description
Destination context name	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the destination context will be recognized by the system. NOTE: For this configuration, the destination context name should not match the domain name of a specific domain.
PDN Interface Configuration	
PDN interface name	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the interface will be recognized by the system. Multiple names are needed if multiple interfaces will be configured. PDN interfaces are configured in the destination context.

Required Information	Description
IP address and subnet	These will be assigned to the PDN interface. Multiple addresses and/or subnets are needed if multiple interfaces will be configured.
Physical port number	This specifies the physical port to which the interface will be bound. Ports are identified by the chassis slot number where the line card resides in, followed by the number of the physical connector on the line card. For example, port 17/1 identifies connector number 1 on the card in slot 17. A single physical port can facilitate multiple interfaces.
Physical port description	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the physical port will be recognized by the system. Multiple descriptions are needed if multiple ports will be used. Physical ports are configured within the destination context and are used to bind logical PDN interfaces.
Gateway IP address(es)	Used when configuring static routes from the PDN interface(s) to a specific network.
IP Address Pool Configuration (optional)	
IP address pool name	Each IP address pool is identified by a name. The pool name can be between 1 and 31 alpha and/or numeric characters and is case sensitive. IP address pools are configured in the destination context(s). Multiple address pools can be configured within a single context.
IP pool addresses	An initial address and a subnet, or a starting address and an ending address, are required for each configured pool. The pool will then consist of every possible address within the subnet, or all addresses from the starting address to the ending address. The pool can be configured as public, private, or static.

System-Level AAA Parameter Configuration

The following table lists the information that is required to configure the system-level AAA parameters.

Table 27. Required Information for System-Level AAA Configuration

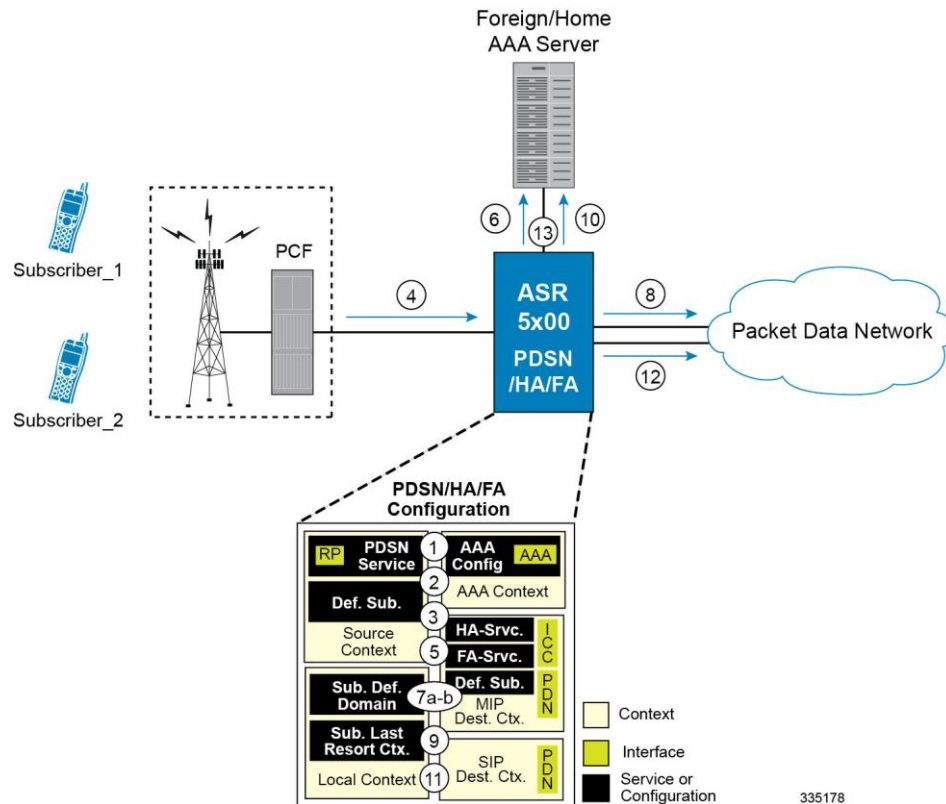
Required Information	Description
Subscriber default domain name	Specifies the name of a context that can provide AAA functions in the event that the domain-part of the username is missing or poorly formed. This parameter will be applied to all subscribers if their domain can not be determined from their username regardless of what domain they are trying to access. NOTE: The default domain name can be the same as the source context.
Subscriber Last-resort context	Specifies the name of a context that can provide AAA functions in the event that the domain-part of the username was present but does not match the name of a configured destination context. This parameter will be applied to all subscribers if their specified domain does not match a configured destination context regardless of what domain they are trying to access. NOTE: The last-resort context name can be the same as the source context.

Required Information	Description
Subscriber username format	<p>Specifies the format of subscriber usernames as to whether or not the username or domain is specified first and the character that separates them. The possible separator characters are:</p> <ul style="list-style-type: none">• @• %• -• \• #• / <p>Up to six username formats can be specified. The default is <i>username @</i>.</p> <p>NOTE: The username string is searched from right to left for the separator character. Therefore, if there is one or more separator characters in the string, only the first one that is recognized is considered the actual separator. For example, if the default username format was used, then for the username string <i>user1@enterprise@isp1</i>, the system resolves to the username <i>user1@enterprise</i> with domain <i>isp1</i>.</p>

How This Configuration Works

The following figure and the text that follows describe how this configuration with a single source and destination context would be used by the system to process a Simple IP data call.

Figure 24. Call Processing When Using the System as a PDSN, FA, and HA



In this example, *Subscriber1* is establishing a Simple IP data session, while *Subscriber2* is establishing a Mobile IP data session.

- The system-level AAA settings were configured as follows:
 - Default domain name = *AAA*
 - Subscriber username format = *username @*
 - Last-resort context name = *AAA*
- The Default Subscriber was configured with an IP context name of *SIP Destination*.
- The Mobile IP FA context name parameter within the PDSN service was configured to the *MIP Destination* context.
- Sessions for *Subscriber1* and *Subscriber2* are received by the PDSN service over the R-P interface from the PCF.
- The PDSN service determines which context to use to provide foreign AAA functionality for each session. This process is described in the How the System Selects Contexts section located in the Understanding the System Operation and Configuration chapter of the *System Administration Guide*.
 For this configuration, the result of this process for both *Subscriber1* and *Subscriber2* would be that the system determines that AAA functionality should be provided by the *AAA* context.
- The system would then communicate with the AAA server specified in the *AAA* context's AAA configuration to authenticate the subscribers.

7. Upon successful authentication, the PDSN service will take the following actions for *Subscriber1* and *Subscriber2*:
 - *Subscriber1*: The system will go through the process of determining which destination context to use for the subscriber session. This process is described in the How the System Selects Contexts section located in the Understanding the System Operation and Configuration chapter of the *System Administration Guide*. For this configuration, the system determines that the egress context is the *SIP Destination* context based on the configuration of the *Default* subscriber in the *Source* context.
 - *Subscriber2*: The system uses the Mobile IP FA context name configured within the PDSN service to determine what destination context facilitates the FA service. In this example, it determines that it must use the *MIP Destination* context and it passes the HA IP address to the FA service.
8. For *Subscriber1*'s session, data traffic would then be routed through the PDN interface in the *SIP Destination* context.
9. For *Subscriber2*, the FA service then establishes a connection to the specified HA service through the ICC interface.
10. For *Subscriber2*, the system would then communicate with the AAA server specified in the *AAA* context's AAA configuration to authenticate the subscriber.
11. For *Subscriber2*, upon successful authentication, the *MIP Destination* context determines which destination context to use for the session and Mobile IP registration would be completed. This process is described in the How the System Selects Contexts section located in the Understanding the System Operation and Configuration chapter of the *System Administration Guide*.

For this example, the *Source* context determines that the egress context is the *MIP Destination* context based on the configuration of the *Default* subscriber.
12. For *Subscriber2*'s session, data traffic would then be routed through the PDN interface in the *MIP Destination* context.
13. Accounting messages for both sessions would be sent to the AAA server over the AAA interface in the *AAA* context.

Chapter 6


Service Configuration Procedures


This chapter is meant to be used in conjunction with the previous chapters that provide examples for configuring the system to support Simple IP services, Mobile IP services, or both. It provides procedures for configuring the various elements to support these services.

It is recommended that you first select the configuration example that best meets your service model, and then use the procedures in this chapter to configure the required elements for that model.

This section includes the following topics:

- [Creating and Configuring PDSN Services](#)
- [Creating and Configuring FA Services](#)
- [Creating and Configuring HA Services](#)
- [Configuring IP Address Pools on the System](#)

 **Important:** This manual is valid for configuring PDSN on multiple platforms. Consequently not all sections, descriptions, features and commands are supported on all platforms. Others are activated by license only.

 **Important:** For hardware supporting them, at least onepacket processing card must be made active prior to service configuration. Information and instructions for configuring active cards can be found in the “Configuring System Settings” chapter of the *System Administration Guide*.

Creating and Configuring PDSN Services

PDSN services are configured within contexts and allow the system to function as a PDSN in the 3G wireless data network.



Important: This section provides the minimum instruction set for configuring a PDSN service that allows the system to process data sessions. Commands that configure additional PDSN service properties are provided in the *Command Line Interface Reference*.

Use this example to configure PDSN services:

configure

```

context <name>

    pdsn-service <name>

        ip local-port <port#>

        authentication allow-noauth

        authentication chap 1 mschap 2 pap 3 allow-noauth

        nai-construct domain <alias>

        spi remote-address <pcf_ipv4_address/pcf_ipv6_address/mask> spi-number <number>
{ secret <secret> }

        lifetime <time>

        gre protocol-type { any | byte-stream | ppp }

        bind address address

        exit

    ppp lcp-start-delay <seconds>

    no ppp renegotiation retain-ip-address

end

```

Notes:

- Optional: If you are implementing Mobile IP data services, configure the name of the context in which the FA service is configured by entering the **mobile-ip foreign-agent context fa_context_name [fa-service <name>]** command.
- Optionally configure the PDSN service to monitor all PCFs that it is associated with, enter the **pcf-monitor** command.
- Optionally configure the PDSN behavior for A11 RRQ related parameters. **airlink bad-sequence-number deny** can be used to deny A11 RRQ messages that have an unsupported Vendor Id or invalid Airlink Sequence number (less than or equal to a previously received sequence number). Keywords and options that

configure additional PDSN service behavior for A11 RRQs with this command are provided in the *Command Line Interface Reference*.

- Optionally use the **no dormant-transition initial-session-setup** command to configure the PDSN behavior to terminate A10 session, when the PDSN receives the A11-RRQ (Type 4) before the session for the original MN is established completely.
- Optionally use the **no pcf-session-id-change restart-ppp** command to configure the PDSN behavior to disable the ppp renegotiation, when the PDSN receives the A11 RRQ (Type 4) with a change in GRE key or PCF session Id, from current PCF and no change in PCF/PANID/CANID.
- Optionally use the **setup-timeout<seconds>** command to change the maximum amount of time, in seconds, allowed to set up a session. The default setting is 60 seconds.
- Optionally configure a delay before starting LCP to avoid the first LCP Configuration Request being lost because the RP link may not be ready even if it has indicated it is active. Losing an LCP Config Request increases the total session setup time.
- Optional: You can configure the system whether to retain the currently allocated IP address for the session or to release the current IP address, and a new IP address is to allocate after PPP renegotiation.
- To retain the allocated IP during PPP renegotiation use the **[default] ppp renegotiation retain-ip-address** command



Important: By default it will use the same IP address, allocated during renegotiation, after renegotiation also. Detailed informations are provided in the *Command Line Interface Reference*.

- Optionally configure the MSID length to reject the A11-RRQs with illegal IMSI value by entering the **[default] msid length [min min_length] max max_length** command:
By default it will use the default MSID length as per standard. Detailed informations are provided in the *Command Line Interface Reference*.
- The nai-construct domain command should only be used if the PDSN service is configured to allow no authentication using the authentication allow-noauth command.
- Multiple SPIs can be configured within the PDSN service in order to accommodate a single PDSN interface communicating with multiple PCFs.
- An infinite lifetime can be configured using the no lifetime command.
- Multiple addresses on the same IP interface can be bound to different PDSN services. However, each address can be bound to only one PDSN service.
- The hardware configuration and features installed can affect the maximum subscriber sessions that can be supported.
- Repeat this configuration as needed to create and bind additional PDSN services to any other interfaces.
- Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Verifying the PDSN Services

Step 1 Use the following command to verify that the PDSN service was created and configured properly:

```
show pdsn-service { name service_name | all }
```

The output is a concise listing of PDSN service parameter settings as shown in the sample output below. In this example, a PDSN service called pdsn1 was configured.

```
Service name: pdsn1

Context: test1

Bind: Not Done

Local IP Address: 0.0.0.0 Local IP Port: 699

Lifetime: 00h30m00s Retransmission Timeout: 3 (secs)

Max Retransmissions: 5 Setup Timeout : 60 (secs)

No MIP FA Context defined

No NAI construct domain defined

GRE Sequence Numbers: Enabled GRE Protocol Type: Any

GRE Reorder Timeout: 100 msec GRE Sequence Mode: None

GRE Checksum: Disabled GRE Checksum Verification: Disabled

Enable Data Available Indicator: Yes Inter-PDSN handoffs have MEI: No

Reg discard on bad extension: No Reg discard on GRE key change: No

Reg ack deny terminates session: No Reg update wait timeout: No

Deny newcall if no rev. tunnel: No

Terminate session on R-P errors: No Max retried replies on reg deny: 3

Deny using zero GRE key: No Deny if session already closed: No

Deny if session already dormant: No Deny if session already active: No

Deny if CoA & src addr mismatch: No

Deny newcall if no conn setup: No (Deny code: Reason Unspecified)

RRQ with bad airlink seq num: Accept(Deny code: Poorly Formed Request)

Deny if CRP to RP H/O in progress:No

Handoff with no conn setup: Accept

Accept H/O if sess being disc: No

PPP Authentication: CHAP 1 PAP 2

Allow Noauthentication: Disabled MSID Authentication: Disabled

Fragment PPP Data: Enabled
```


```
GRE Flow Control: Disabled
GRE Flow Control Timeout: 10000 msec
GRE Flow Control Timeout Action: disconnect-session
Max sessions: 500000
Alt-PPP: Disabled
PPP Tunnel Type: None No PPP Tunnel Context defined
No Default Subscriber defined
IP SRC-Violation Reneg Limit: 5 IP SRC-Violation Drop Limit: 10
IP SRC-Violation Clear-on-ValidPDU: No IP SRC-Violation Period: 120 secs
Always-On-Indication: Disabled SDB Indication for Echo Req: Disabled
SPI(s):
Service Status: Not started
Overload Policy: Reject (Reject code: Admin Prohibited)
Newcall Policy: None
Service Option Policy: Enforce
Service Options: 7,15,22,23,24,25,33,59
PCF Monitor Config: Disabled
```

Step 2 Verify configuration for errors by entering the following command:

```
show configuration errors section pdsn-service verbose | more
```

Creating and Configuring FA Services

FA services are configured within contexts and allow the system to function as an FA in the 3G wireless data network.

 **Important:** This section provides the minimum instruction set for configuring an FA service that allows the system to process data sessions. Commands that configure additional FA service properties are provided in the *Command Line Interface Reference*. Additionally, when configuring Mobile IP take into account the MIP timing considerations discussed in the Appendix *MIP Timer Considerations*.

Use this example to create and/or configure FA services:

```
configure

context <name>

    fa-service <name>

        ip local-port <port#>

        fa-ha-spi remote-address<ipv4_address/ipv6_address>|<ipv4/ipv6_address/mask spi-
number number

{ encrypted secret secret | secret secret }

    advertise adv-lifetime <time>

    advertise num-adv-sent <number>

    advertise reg-lifetime <reg_time>

    multiple-reg <number>

    authentication mn-aaa { always | ignore-after-handoff | init-reg | init-reg-
except-handoff | renew-and-dereg-noauth | renew-reg-noauth }

    reg-timeout time

    bind address ipv4_address max-subscribers max#

end
```

Following are a few things to be aware of:

- The **ip local-port** command configures the User Datagram Protocol (UDP) port for the Pi interfaces' IP socket.
- A maximum of 2048 FA-HA SPIs can be configured for a single FA service.
- The agent advertisement lifetime is the amount of time that an FA agent advertisement remains valid in the absence of further advertisements.
- An infinite registration lifetime can be configured using the **no advertise reg-lifetime** command.
- The system only supports multiple Mobile IP sessions per subscriber if the subscriber's mobile node has a static IP address. The system only allows a single Mobile IP session for mobile nodes that receive a dynamically

assigned home IP address. The hardware configuration and features installed can affect the maximum subscriber sessions that can be supported.

- Optionally configure the FA service for controlling the negotiation and sending of the I-bit in revocation messages by adding the **revocation negotiate-i-bit** command. By default, it will not send I-bit in revocation message.
- Repeat the configuration as needed to create and bind additional FA services to any other interfaces.

Verifying the FA Service

Step 1 Verify that your FA services were created and configured properly by entering the following command:

```
show fa-service { name service_name | all }
```

The output is a concise listing of FA service parameter settings similar to the sample displayed below. In this example, a FA service called `fa1` was configured.

```
Service name: fa1

Context: xxx

Bind: Done Max Subscribers: 500000

Local IP Address: 195.20.20.3 Local IP Port: 434

Lifetime: 00h10m00s Registration Timeout: 45 (secs)

Advt Lifetime: 02h30m00s Advt Interval: 5000 (msecs)

Num Advt: 5

Advt Prefix Length Extn: NO

Reverse Tunnel: Enabled GRE Encapsulation: Enabled

Optimize Tunnel Reassembly: Disabled Allow Priv Addr w/o Rev Tunnel: Disabled

Dynamic MIP Key Update: Enabled Ignore Dynamic MIP Key: Disabled

Remove MN-AAA/MN-FAC extns: Disabled

Proxy MIP: Enabled Proxy MIP Max Retransmissions: 5

Proxy MIP Retrans Timeout: 3 (secs) Proxy MIP Renew Percent Time: 75%

SPI(s):

FAHA: Remote Addr: 195.30.30.3/32

Hash Algorithm: HMAC_MD5 SPI Num: 1000 Replay Protection: Timestamp Timestamp
Tolerance: 60 FAHA: Remote Addr: 195.30.30.2/32 Hash Algorithm: HMAC_MD5 SPI
Num: 1000 Replay Protection: Timestamp Timestamp Tolerance: 60 FAHA: Remote Addr:
```

```

195.30.30.1/32 Hash Algorithm: HMAC_MD5 SPI Num: 1000 Replay Protection:
Timestamp Timestamp Tolerance: 60

FAHA: Remote Addr: 195.20.20.4/32

Hash Algorithm: HMAC_MD5 SPI Num: 1000

Replay Protection: Timestamp Timestamp Tolerance: 60

IPSEC Crypto Map(s):

Peer HA Addr: 195.30.30.2

Crypto Map: test

GRE Sequence Numbers: Disabled GRE Sequence Mode: None

GRE Reorder Timeout: 100 msec

GRE Checksum: Disabled GRE Checksum Verification: Disabled

Registration Revocation: Enabled Reg-Revocation I bit: Enabled

Reg-Revocation Max Retries: 3 Reg-Revocation Timeout: 3 (secs)

Reg-Rev on InternalFailure: Enabled

Default Subscriber: None

Max sessions: 500000

Max challenge len: 16

Challenge Window: 2

Service Status: Started

MN-AAA Auth Policy: Always

MN-HA Auth Policy: Always

Newcall Policy: None

Idle Timeout Mode: Normal

Ignore Stale Challenge: Disabled

```

- Step 2** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Creating and Configuring HA Services

HA services are configured within contexts and allow the system to function as an HA in the 3G wireless data network.



Important: This section provides the minimum instruction set for configuring an HA service that allows the system to process data sessions. Commands that configure additional HA service properties are provided in the *Command Line Interface Reference*. Additionally, when configuring Mobile IP take into account the MIP timing considerations discussed in *MIP Timer Considerations*.

Use this example to create and/or configure HA services:

configure

```
context <name>

    ha-service <name>

        ip local-port <port#>

        authentication mn-aaa { allow-noauth | always | noauth | renew-reg-noauth }

        fa-ha-spi remote-address <ipv4/ipv6_address > | <ipv4/ipv6_address/mask> spi-
number <number> { [encrypted] secret <secret> }

        mn-ha-spi spi-number <number> { encrypted secret <secret> | secret <secret>
}
        reg-lifetime <time>          simultaneous-bindings <number>          bind address
<ipv4_address> max-subscribers <max#>          end
```

Following are a few things to be aware of:

- The **ip local-port** command configures the User Datagram Protocol (UDP) port for the Pi interfaces' IP socket.
- A maximum of 2048 FA-HA SPIs can be configured for each HA service.
- An infinite registration lifetime can be configured using the **no reg-lifetime** command.
- The hardware configuration and features installed can affect the maximum subscriber sessions that can be supported.
- Optionally configure the HA service for controlling the negotiation and sending of the I-bit in revocation messages by adding the **revocation negotiate-i-bit** command. By default it will not send I-bit in revocation message.
- Optionally change the maximum amount of time, in seconds, allowed to set up a session. The default setting is 60 seconds. To change this value add the **setup-timeout seconds** command.
- Repeat the configuration as needed to create and bind additional HA services to any other interfaces.

Verifying the HA Service

Step 1 Verify that your HA services were created and configured properly by entering the following command:

```
show ha-service { name service_name | all }
```

The output is a concise listing of HA service parameter settings. In this example, a HA service called `hal` was configured.

```
Service name: hal

Context: ha

Bind: Done Max Subscribers: 500000

Local IP Address: 192.168.4.10 Local IP Port: 434

Lifetime: 00h10m00s Simul Bindings: 3

Reverse Tunnel: Enabled GRE Encapsulation: Enabled

Optimize Tunnel Reassembly: Enabled Setup Timeout: 60 sec

SPI(s) :

MNHA: Remote Addr: 0.0.0.0

Hash Algorithm: MD5 SPI Num: 1000

Replay Protection: Timestamp Timestamp Tolerance: 60

Permit Any Hash Algorithm: Disabled

FAHA: Remote Addr: 195.20.20.6/32

Hash Algorithm: HMAC_MD5 SPI Num: 1000

Replay Protection: Timestamp Timestamp Tolerance: 60

FAHA: Remote Addr: 195.20.20.5/32

Hash Algorithm: HMAC_MD5 SPI Num: 1000

Replay Protection: Timestamp Timestamp Tolerance: 60

FAHA: Remote Addr: 195.20.20.3/32

Hash Algorithm: HMAC_MD5 SPI Num: 1000

Replay Protection: Timestamp Timestamp Tolerance: 60

FAHA: Remote Addr: 195.20.20.2/32

Hash Algorithm: HMAC_MD5 SPI Num: 1000

Replay Protection: Timestamp Timestamp Tolerance: 60

IPSEC Crypto Map(s) :

Peer FA Addr: 192.168.4.1
```

```
Crypto Map: test

'S' Key expires at: No Valid S-Key

'S' Lifetime Skew: 00h00m10s

IPSEC AAA Context: xxx

GRE Sequence Numbers: Disabled GRE Sequence Mode: None

GRE Reorder Timeout: 100 msec

GRE Checksum: Disabled GRE Checksum Verification: Disabled

Registration Revocation: Enabled Reg-Revocation I bit: Enabled

Reg-Revocation Max Retries: 3 Reg-Revocation Timeout: 3 (secs)

Reg-Rev Handoff old-FA: Enabled Reg-Rev Idle-Timeout: Enabled

Default Subscriber: None

Max Sessions: 500000

Service Status: Started

MN-AAA Auth Policy: Always

MN-HA Auth Policy: Always

IMSI Auth: Disabled

AAA accounting: Enabled

Idle Timeout Mode: Aggressive

Newcall Policy: None

Overload Policy: Reject (Reject code: Admin Prohibited)

NW-Reachability Policy: Reject (Reject code: Admin Prohibited)
```

- Step 2** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Configuring IP Address Pools on the System

One of the steps in establishing a PPP session between the mobile and the PDSN service running on the system is that upon successful authentication, the subscriber's mobile node is assigned an IP address. The IP address could be dynamically assigned from a pool that is configured on the system or on the AAA server. It may also be an address that is statically configured in the user profile or even one that is requested by the subscriber.

IP addresses can be dynamically assigned from a single pool/a group of IP pools/a group of IP pool groups. The addresses/IP pools/ IP pool groups are placed into a queue in each pool or pool group. An address is assigned from the head of the queue and, when released, returned to the end. This method is known as least recently used (LRU).

When a group of pools have the same priority, an algorithm is used to determine a probability for each pool based on the number of available addresses, then a pool is chosen based on the probability. This method, over time, allocates addresses evenly from the group of pools.



Important: Note that setting different priorities on each individual pool can cause addresses in some pools to be used more frequently.

To configure the IP pool:

- Create the IP pool for IPv4 addresses in system context by applying the example configuration.
- Optional. Configure the IP pool for IPv6 addresses in system context by applying the example.
- Optional. Configure the overlap-pool addresses to routing by applying the example configuration.
- Verify your IP pool configuration.
- Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Creating IPv4 Pool

Use the following example to create the IPv4 address pool:

configure

context <dest_ctxt_name>

ip pool <pool_name> <ipv4/ipv6_address|ipv4/ipv6_address/mask>

end

Following are a few things to be aware of:

- To ensure proper operation, IP pools should be configured within a destination context.
- Each address in the pool requires approximately 24 bytes of memory. Therefore, in order to conserve available memory, the number of pools may need to be limited depending on the number of addresses to be configured and the number of PACs/PSCs installed.
- Setting different priorities on individual pools can cause addresses in some pools to be used more frequently.

- For more information on commands/keywords that configure additional parameters and options, refer `ipv6 pool` command section in the “Context Configuration Mode Commands” chapter of the *Command Line Interface Reference*.

Creating IPv6 Pool

Use the following example to create the IPv6 address pool:

```
configure
```

```
context <dest_ctxt_name>

  ipv6 pool <pool_name> 6to4 local-endpoint <ipv4/ipv6_address>

end
```

Following are a few things to be aware of:

- To ensure proper operation, IP pools should be configured within a destination context.
- Each address in the pool requires approximately 24 bytes of memory. Therefore, in order to conserve available memory, the number of pools may need to be limited depending on the number of addresses to be configured and the number of PACs/PSCs installed.
- Setting different priorities on individual pools can cause addresses in some pools to be used more frequently.
- For more information on commands/keywords that configure additional parameters and options, refer `ipv6 pool` command section in the “Context Configuration Mode Commands” chapter of the *Command Line Interface Reference*.

Adding Overlap-Pool Addresses to Routing

Use the following configuration to advertise overlap-pool addresses in dynamic routing protocols.

```
configure
```

```
context <context_name>

  [ no | default ] ip routing overlap-pool
```

If `ip routing overlap-pool` is configured, then the overlap addresses are added as interface addresses in the routing stack and a route is added in the kernel. The intf-address in the routing stack and the route in the kernel for the overlap address are removed when all the overlap-pools are deleted. The default is `no ip routing overlap-pool`.

Verifying IP Pool Configuration

Step 1 Verify that your IPv4 address pool configured properly by entering the following command in Exec Mode:

```
show ip pool
```

The output from this command should look similar to the sample shown below. In this example all IP pools were configured in the `isp1` context.

■ Configuring IP Address Pools on the System

```

context : ispl:
+-----Type: (P) - Public (R) - Private
| (S) - Static (E) - Resource
|
|+-----State: (G) - Good (D) - Pending Delete (R)-Resizing
||
||+---Priority: 0..10 (Highest (0) .. Lowest (10))
||||
||||+--Busyout: (B) - Busyout configured
|||||
|||||

vvvvv Pool Name Start Address Mask/End Address Used Avail
-----
PG00 ipsec      12.12.12.0 255.255.255.0 0 254
RG00 pool3      30.30.0.0 255.255.0.0 0 65534
SG00 pool2      20.20.0.0 255.255.0.0 10 65524
PG00 pool1      10.10.0.0 255.255.0.0 0 65534
SG00 vpnpool    192.168.1.250 192.168.1.254 0 5

Total Pool Count: 5

```

Step 2 Verify that your IPv6 address pools configured properly by entering the following command in Exec Mode:

```
show ipv6 pools
```

The output from this command should look similar to the sample shown above except IPv6 addresses.

Chapter 7

Monitoring the Service

This chapter provides information for monitoring service status and performance using the **show** commands found in the Command Line Interface (CLI). These command have many related keywords that allow them to provide useful information on all aspects of the system ranging from current software configuration through call activity and status.


The selection of keywords described in this chapter is intended to provided the most useful and in-depth information for monitoring the system. For additional information on these and other **show** command keywords, refer to the *Command Line Interface Reference*.

In addition to the CLI, the system supports the sending of Simple Network Management Protocol (SNMP) traps that indicate status and alarm conditions. Refer to the *SNMP MIB Reference Guide* for a detailed listing of these traps.


Monitoring System Status and Performance



This section contains commands used to monitor the status of tasks, managers, applications and other software components in the system. Output descriptions for most of the commands are located in the *Statistics and Counters Reference*.

Table 28. System Status and Performance Monitoring Commands

To do this:	Enter this command:
View Congestion-Control Statistics	
View Congestion-Control Statistics	<code>show congestion-control statistics { allmgr gtpcmgr hamgr l2tpmgr }</code>
View Subscriber Information	
View session resource status	<code>show resources session</code>
Display Subscriber Configuration Information	
View locally configured subscriber profile settings (must be in context where subscriber resides)	<code>show subscribers configuration username subscriber_name</code>
View remotely configured subscriber profile settings	<code>show subscribers aaa-configuration username subscriber_name</code>
View Subscribers Currently Accessing the System	
View a listing of subscribers currently accessing the system	<code>show subscribers all</code>
Display PCF-Summary Session Counters	
View PCF-summary session counters	<code>show session counters pcf-summary</code>
Display Session State Statistics	
View session state statistics	<code>show session progress</code>
Display Session State PCF Statistics	
View session state PCF statistics	<code>show session progress pcf all</code>
Display Session Subsystem and Task Statistics	
 Important: Refer to the System Software Task and Subsystem Descriptions appendix in the System Administration Guide for additional information on the Session subsystem and its various manager tasks.	

To do this:	Enter this command:
View A11 Manager statistics	<code>show session subsystem facility allmgr all</code>
View AAA Manager statistics	<code>show session subsystem facility aaamgr all</code>
View FA Manager statistics	<code>show session subsystem facility famgr all</code>
View L2TP demux manager statistics	<code>show session subsystem facility l2tpdemux all</code>
View L2TP Manager statistics	<code>show session subsystem facility l2tpmgr all</code>
View Session Manager statistics	<code>show session subsystem facility sessmgr all</code>
View Session Recovery Status	
View session recovery status	<code>show session recovery status [verbose]</code>
Display Session Disconnect Reasons	
View session disconnect reasons with verbose output	<code>show session disconnect-reasons</code>
View Point-to-Point Protocol Statistics	
Display a Summary of PPP Counter Status	
View cumulative subscriber session PPP counters	<code>show ppp</code>
Display PPP Counters for a Specific Subscriber	
View individual subscriber session PPP counters	<code>show ppp username subscriber_name</code>
View individual subscriber session PPP error and data counters	<code>show ppp counters username subscriber_name</code>
View individual subscriber session detailed PPP counters	<code>show ppp full username subscriber_name</code>
Display PPP Statistics for PDSN Services	
Views PPP statistics for a all PDSN services	<code>show ppp statistics pdsn-service</code>
Views PPP statistics for a specific PDSN service	<code>show ppp statistics pdsn-service service_name</code>
View R-P Interface Statistics	
Display a Summary of R-P Interface Counter Status	
View cumulative R-P interface counters for every subscriber session currently in progress	<code>show rp</code>
Display R-P Interface Counters for a Specific Subscriber	

To do this:	Enter this command:
View R-P interface counters for a specific subscriber	<code>show rp full username subscriber_name</code>
Display R-P Interface Statistics for PDSN Services	
View R-P interface statistics for all PDSN services	<code>show rp statistics pdsn- service</code>
View R-P interface statistics for a specific PDSN service	<code>show rp statistics pdsn- service service_name</code>
View Mobile IP Foreign Agent Statistics	
Display Mobile IP FA Information for a Specific Subscriber	
View Mobile IP FA counters for a specific subscriber	<code>show mipfa full username subscriber_name</code>
Display Mobile IP Statistics for FA Services	
View statistics for a specific FA service	<code>show mipfa statistics fa- service service_name</code>
Display Mobile IP FA Counters	
View Mobile IP FA counters for individual subscriber sessions	<code>show mipfa counters</code>
Display RADIUS Server States  Important: These commands can display 10 state transition histories of RADIUS accounting and authentication servers (Active/Not responding/Down States). For a complete explanation of RADIUS server states, refer to the RADIUS Server State Behavior appendix in the AAA Administration and Reference.	
View RADIUS authentication server group server states for a specific group	<code>show radius authentication servers radius group group_name detail</code>
View RADIUS accounting server group server states for a specific group	<code>show radius accounting servers radius group group_name detail</code>
Display RADIUS Protocol Counters	
View cumulative RADIUS protocol counters	<code>show radius counters all</code>
View RADIUS protocol counter summary of RADIUS authentication and accounting	<code>show radius counters summary</code>
View L2TP Information	
Display L2TP Session Information	

To do this:	Enter this command:
View cumulative statistics for all sessions processed within the current context  Important: If this command is executed from within the localout-of-band context, cumulative session information is displayed for all contexts.	<code>show l2tp sessions</code>
View all information pertaining to the L2TP session of a specific subscriber	<code>show l2tp session full username subscriber_name</code>
Display L2TP Statistics	
View statistics for a specific LAC service  Important: If this command is executed from within the localout-of-band context, cumulative session information is displayed for all contexts.	<code>show l2tp statistics lac-service service_name</code>
Display L2TP Tunnel Information	
View all tunnels currently being facilitated by LAC services within a specific context	<code>show l2tp tunnels all</code>
Display IPSec Security Association Statistics	
View IPSec security association statistics for crypto maps in the current context	<code>show crypto ipsec security-associations statistics</code>
Display Pre-shared ISAKMP Keys	
View pre-shared keys received from peer security gateways as part of the Diffie-Hellman exchange	<code>show crypto isakmp keys</code>
Display IPSec Statistics	
View cumulative IPSec statistics for the current context	<code>show crypto statistics</code>

Clearing Statistics and Counters

It may be necessary to periodically clear statistics and counters in order to gather new information. The system provides the ability to clear statistics and counters based on their grouping (PPP, MIPHA, MIPFA, etc.).

Statistics and counters can be cleared using the CLI **clear** command. Refer to the *Command Line Interface Reference* for detailed information on using this command.

Chapter 8

Troubleshooting the System

This chapter provides information and instructions for using the system command line interface (CLI) for troubleshooting any issues that may arise during system operation.

Test Commands

In the event that an issue was discovered with an installed application or line card, depending on the severity, it may be necessary to take corrective action.

The system provides several redundancy and fail-over mechanisms to address issues with application and line cards in order to minimize system downtime and data loss. These mechanisms are described below.

Using the PPP Echo-Test Command

The system provides a mechanism to verify the Point-to-Point Protocol session of a particular subscriber by sending Link Control Protocol (LCP) packets to the mobile node. This functionality can be extremely useful in determining the quality of the air link and delays that may occur.

The command has the following syntax:

```
ppp echo-test { callid call_id | ipaddr ipv4_address | msid ms_id | username
subscriber_name }
```

Keyword/Variable	Description
callid <i>call_id</i>	Specifies that the test is executed for a subscriber with a specific call identification number (callid). <i>call_id</i> is the specific call identification number that you wish to test.
ipaddr <i>ip_address</i>	Specifies that the test is executed for a subscriber with a specific IP address. <i>ipv4_address</i> is the specific IPV4 address that you wish to test.
msid <i>ms_id</i>	Specifies that the test is executed for a subscriber with a specific mobile station identification (MSID) number. <i>ms_id</i> is the specific mobile station identification number that you wish to test.
username <i>subscriber_name</i>	Specifies that the test is executed for a subscriber with a specific username. <i>subscriber_name</i> is the specific username that you wish to test.

The following displays a sample of this command's output showing a successful PPP echo-test to a subscriber named *user2@aaa*.

```
USERNAME: user2@aaa MSID: 0000012345 CALLID: 001e8481

Tx/Rx 1/0 RTT(min/max/avg) 0/0/0

USERNAME: user2@aaa MSID: 0000012345 CALLID: 001e8481

Tx/Rx 1/1 RTT(min/max/avg) 77/77/77 (COMPLETE)
```

Appendix A

Engineering Rules

This section provides engineering rules or guidelines that must be considered prior to configuring the system for your network deployment.

Interface and Port Rules

The rules discussed in this section pertain to the Ethernet 10/100 line card, the Ethernet 1000 line card and the four-port Quad Gig-E line card and the type of interfaces they facilitate, regardless of the application.

R-P Interface Rules

The following engineering rules apply to the R-P interface:

- An R-P interface is created once the IP address of a logical interface is bound to a PDSN service.
- The logical interface(s) that will be used to facilitate the R-P interface(s) must be configured within an “ingress” context.
- PDSN services must be configured within an “ingress” context.
- At least one PDSN service must be bound to each interface; however, multiple PDSN services can be bound to a single interface if secondary addresses are assigned to the interface.
- Each PDSN service must be configured with the Security Parameter Index (SPI) of the Packet Control Function (PCF) that it will be communicating with over the R-P interface.
- Multiple SPIs can be configured within the PDSN service to allow communications with multiple PCFs over the R-P interface. It is best to define SPIs using a netmask to specify a range of addresses rather than entering separate SPIs. This assumes that the network is physically designed to allow this communication.
- Depending on the services offered to the subscriber, the number of sessions facilitated by the R-P interface can be limited.

Pi Interface Rules

FA to HA Rules

When supporting Mobile IP, the system can be configured to perform the role of a FA, an HA, or both. This section describes the engineering rules for the Pi interface when using the system as a FA.

The following engineering rules apply to the Pi interface between the FA and HA:

- A Pi interface is created once the IP address of a logical interface is bound to an FA service.
- The logical interface(s) that will be used to facilitate the Pi interface(s) must be configured within the egress context.
- FA services must be configured within the egress context.
- If the system is configured as a FA is communicating with a system configured as an HA, then it is recommended that the name of the context in which the FA service is configured is identical to the name of the context that the HA service is configured in on the other system.
- Each FA service may be configured with the Security Parameter Index (SPI) of the HA that it will be communicating with over the Pi interface.
- Multiple SPIs can be configured within the FA service to allow communications with multiple HAs over the Pi interface. It is best to define SPIs using a netmask to specify a range of addresses rather than entering separate SPIs. This assumes that the network is physically designed to allow this communication.

- Depending on the services offered to the subscriber, the number of sessions facilitated by the Pi interface can be limited.

HA to FA

The following engineering rules apply to the Pi interface between the HA and FA:

- When supporting Mobile IP, the system can be configured to perform the role of a FA, an HA or both. This section describes the engineering rules for the Pi interface when using the system as an HA.
- A Pi interface is created once the IP address of a logical interface is bound to an HA service.
- The logical interface(s) that will be used to facilitate the Pi interface(s) must be configured within an ingress context.
- HA services must be configured within an ingress context.
- If the system configured as an HA is communicating with a system configured as a FA, then it is recommended that the name of the context in which the HA service is configured is identical to the name of the context that the FA service is configured in on the other system.
- Each HA service may be configured with the Security Parameter Index (SPI) of the FA that it will be communicating with over the Pi interface.
- Multiple SPIs can be configured within the HA service to allow communications with multiple FAs over the Pi interface. It is best to define SPIs using a netmask to specify a range of addresses rather than entering separate SPIs. This assumes that the network is physically designed to allow this communication.
- Each HA service must be configured with a Security Parameter Index (SPI) that it will share with mobile nodes.
- Depending on the services offered to the subscriber, the number of sessions facilitated by the Pi interface can be limited in order to allow higher bandwidth per subscriber.

Subscriber Rules

The following engineering rule applies to subscribers configured within the system:

- A maximum of 2,048 local subscribers can be configured per context.
- Default subscriber templates may be configured on a per PDSN or FA service.

Service Rules

The following engineering rules apply to services configured within the system:

- A maximum of 256 services (regardless of type) can be configured per system.



Caution: Large numbers of services greatly increase the complexity of management and may impact overall system performance (i.e. resulting from such things as system handoffs). Therefore, it is recommended that a large number of services only be configured if your application absolutely requires it. Please contact your local service representative for more information.

- Up to 2,048 Security Parameter Indices (SPIs) can be configured for a single PDSN service.
- Up to 2,048 MN-HA and 2048 FA-HA SPIs can be supported for a single HA service.
- Up to 2,048 FA-HA SPIs can be supported for a single FA service.
- The system supports unlimited peer FA addresses per HA.
 - The system maintains statistics for a maximum of 8192 peer FAs per HA service.
 - If more than 8192 FAs are attached, older statistics are identified and overwritten.
- The system maintains statistics for a maximum of 4096 peer HAs per FA service.
- There are a maximum of 8 HA assignment tables per context and per chassis.
- The total number of entries per table and per chassis is limited to 256.
- Up to 10,000 LAC addresses can be configured per LNS service.
- Even though service names can be identical to those configured in different contexts on the same system, this is not a good practice. Having services with the same name can lead to confusion, difficulty troubleshooting problems, and make it difficult understanding outputs of show commands.

Appendix B

Supported Registration Reply Codes

Each of the three sections that follow describe the registration reply codes supported by the system for the PDSN and FA services.

PDSN Service Reply Codes

The following registration reply codes are supported by the system's PDSN service in accordance with the *3GPP2 A.S0001-A v2: 3GPP2 Access Network Interfaces Interoperability Specification* (also known as *3G-IOS v4.1.1*).

Table 29. Supported PDSN Service Registration Reply Codes

Reply Code (Hex / Base 10)	Description	Notes
00H / 0	Registration Accepted	Sent when the subscriber session is successfully set up.
80H / 128	Registration Denied - reason unspecified	Sent when internal errors are encountered when processing the Request packet.
81H / 129	Registration Denied - administratively prohibited	Sent when a newcall policy is set to reject.
82H / 130	Registration Denied - insufficient resources	Sent when no memory or session managers are available to process the session.
83H / 131	Registration Denied - mobile node failed authentication	Sent when the mobile node failed authentication. When multiple errors occur on authentication this message takes precedence and is listed first.
85H / 133	Registration Denied - identification mismatch	Sent when the PCF's timestamp does not match the PDSN. The PDSN sends a corrected timestamp to be used as the ID by the PCF in subsequent requests.
86H / 134	Registration Denied - poorly formed request	Sent when an unsupported Service option is received or the packet is malformed in any way.
88H / 136	Registration Denied - unknown PDSN address	Sent when PDSN redirect policy is invoked.
89H / 137	Registration Denied - requested reverse tunnel unavailable	Sent when reverse tunneling is requested by the mobile node but it is not enabled on the system.
8AH / 138	Registration Denied - reverse tunnel is mandatory and 'T' bit not set	Sent when reverse tunneling is enabled on the system but is not supported by the mobile node.
8DH / 139	Registration Denied - unsupported vendor ID or unable to interpret data in the CVSE	Sent when the Airlink records from the PCF contain a Vendor ID value other than 0x159F or the Accounting VSE contains an Application Sub Type other than RADIUS or the Request contains a Critical Vendor Specific Extension Type that is not recognized by the PDSN.

FA Service Reply Codes

The following registration reply codes are supported by the system's FA service in accordance with the following Request For Comments (RFCs):

- RFC-2002, IPv4 Mobility, May 1995
- RFC-2344, Reverse Tunneling for Mobile IP, May 1998
- RFC-2794, Mobile IP Network Access Identifier Extension for IPv4, March 2000
- RFC-3012, Mobile IPv4 Challenge/Response Extensions, November 2000

Table 30. Supported FA Service Registration Reply Codes

Reply Code (Hex / Base 10)	Description	Notes
40H / 64	Registration Denied - reason unspecified	Sent when internal errors are encountered when processing the Request packet.
41H / 65	Registration Denied - administratively prohibited	Sent when a newcall policy is set to reject calls or the subscriber is not permitted to use Mobile IP FA services.
42H / 66	Registration Denied - insufficient resources	Sent when no memory or session managers are available to process the session.
43H / 67	Registration Denied - mobile node failed authentication	Sent when the mobile node failed authentication.
44H / 68	Registration Denied - home agent failed authentication	Sent when an HA attempted to communicate with the FA service using an incorrect security parameter index (SPI).
45H / 69	Registration Denied - requested lifetime too long	Sent when the mobile node requests a registration lifetime longer than the maximum supported by the FA.
46H / 70	Registration Denied - poorly formed request	Sent when the registration request is poorly formed (i.e. missing an Authentication extension).
47H / 71	Registration Denied - poorly formed reply	Sent when the registration reply is poorly formed (i.e. missing an Authentication extension).
48H / 72	Registration Denied - requested encapsulation unavailable	Sent when requested encapsulation type is unavailable (GRE or minimal IP encapsulation).
4AH / 74	Registration Denied - reverse tunneling unavailable	Sent when reverse tunneling is requested by the mobile node but it is not enabled on the system.
4BH / 75	Registration Denied - reverse tunneling mandatory	Sent when reverse tunneling is enabled on the system but is not supported by the mobile node.
4CH / 76	Registration Denied - reverse tunneling mobile node too distant	Sent when IP TTL is not set to 255 in Reg Request with T bit set
4DH / 77	Registration Denied - invalid care-of address	Sent when D bit is set in the Registration Request.
4EH / 78	Registration Denied - registration timeout	Sent when FA reg-timeout is exceeded.

Reply Code (Hex / Base 10)	Description	Notes
4FH / 79	Registration Denied - reverse tunneling delivery style unavailable	Sent if the Encapsulating Delivery Style Extension sent by the mobile is not supported by the FA service.
50H / 80	Registration Denied - home network unreachable (ICMP error received)	Sent when the FA service can not contact the home network due to an Internet Control Message Protocol (ICMP) error.
51H / 81	Registration Denied - home agent host unreachable (ICMP error received)	Sent when the FA service can not contact the HA host due to an Internet Control Message Protocol (ICMP) error.
52H / 82	Registration Denied - home agent port unreachable (ICMP error received)	Sent when the FA service can not contact the HA port due to an Internet Control Message Protocol (ICMP) error.
58H / 88	Registration Denied - home agent unreachable (other ICMP error received)	Sent when the FA service can not contact the HA due to an Internet Control Message Protocol (ICMP) error.
60H / 96	Registration Denied - missing home address	Sent when the FA service could not determine the IP address of the mobile node.
61H / 97	Registration Denied - missing NAI	Sent when the FA service could not determine the subscriber's network access identifier.
62H / 98	Registration Denied - missing home agent	Sent when the FA service could not determine the IP address of the mobile node's home agent.
68H / 104	Registration Denied - unknown challenge	Sent if the FA cannot validate the Mobile IP mobile-to-foreign agent advertisement challenge extension provided in the Registration Request.
69H / 105	Registration Denied - missing challenge	Sent if the mobile node's Registration Request does not include a mobile-to-foreign agent advertisement challenge extension.
6AH / 106	Registration Denied - stale challenge	Sent when the mobile node has sent a Registration Request with a challenge value that was already used before.

Appendix C

Mobile-IP and Proxy-MIP Timer Considerations

This appendix is intended to provide a brief explanation of the considerations for lifetime, idle, and absolute timer settings that must be understood when setting up a system in a mobile IP or proxy mobile IP environment. In the Cisco ASR5x00 platform, there is not an explicitly defined MIP lifetime. The MIP lifetime is determined through various timers settings in the configuration and through radius attributes returned in an Access-Accept message.

Call Flow Summary

The following steps describe the call flow as regards the timers that affect a call initiated by the Mobile Node (MN).

1. **PPP Negotiation:** A data call is initiated by beginning PPP. Once PPP is successfully established, the system will understand if the call is a mobile IP call or simple IP call. At this point, the system is not aware of the subscriber username and will use settings from the default subscriber template in the source context or the context defined by the “aaa default-domain subscriber” setting in the global configuration.
2. **FA Agent Advertisement:** Once the system has determined the call is a Mobile IP call, the FA will send a Router Advertisement message with a Mobility Agent Advertisement extension. The Mobility Agent Advertisement includes a Registration Lifetime field. The value of this field will come from one of two places. The FA service has a configurable setting named “advertise reg-lifetime”. The default value for this setting is 600. A setting in the default subscriber template called “timeout idle” is also a candidate. The default value for this setting is 0 (null). The smaller of these two configurable parameters is used as the Registration Lifetime value. Leaving the settings at the defaults will result in an advertised lifetime of 600.

Advertise Reg-Lifetime in FA Service	Timeout Idle in Subscriber Template	Resulting Advertised Registration Lifetime
600	0	600
600	900	600
3600	1200	1200

The device will receive the agent advertisement and send a MIP Registration Request. The device uses the advertised registration lifetime value as the requested MIP lifetime.

3. **AAA Authentication and MIP Registration Request:** The next step in the MIP process will be to authenticate the user at the FA. It is at this stage where a failure condition can be introduced.

If the Access-Accept message does not return any values related to timers, the subscribers MIP Registration Request is sent on to the HA.

If the Access-Accept message does include an attribute relating to Idle or Absolute timer the FA will evaluate the requested lifetime from the device to the value returned by the AAA. The FA will treat any Idle or Absolute timer value returned by the AAA as a maximum value and as such:

- If the requested MIP lifetime from the device is less-than than the returned radius attribute, the lifetime value is considered valid and the MIP Registration Request is forwarded on to the HA.
- If the requested MIP lifetime from the device is greater-than the returned radius attribute, the requested lifetime value is considered to be too long. The FA will send a MIP Registration Reply to the device with a response code of `Error 69 - Requested Lifetime Too Long`. In the reply message, the FA will populate the Lifetime value with the maximum acceptable lifetime. The device may send a new MIP request with this new lifetime value.

MIP Lifetime Requested by Device	Idle-Timer Value in Access-Accept	Resulting MIP Lifetime Request in MIP Request to HA
3600	(Not Returned)	3600
3600	7200	3600
3600	1800	Failure - Error 69

4. **HA Process MIP Request:** The HA has now received a Mobile IP Registration request forwarded by the FA on behalf of the device. The MIP request contains the username and the requested lifetime (as well as other

parameters). The HA will take this lifetime request and compare it to the configurable parameters associated with the HA service and associated configurations. The HA will use the username to determine which subscriber template to use for subscriber specific settings.

The parameters the HA uses to determine the MIP lifetime are the requested lifetime, the “reg-lifetime” setting in the HA service and the “timeout idle” setting in the subscriber template. If the requested MIP lifetime is lower it is sent back to the mobile; if the MIP lifetime is higher the system sends back an RRQ accept with the lifetime set to 5 seconds less than the lower of the idle or absolute timeout for the user.

MIP Lifetime Requested by Device	Timeout Idle/Absolute in Subscriber Template	Reg-Lifetime Value in HA Service	MIP Lifetime Returned to Mobile Device
3600	0(default)	7200	3600
3600	7200	1805	1800
3600	1705	3600	1700

Timer tables combined


PDSN/FA			HA		
Advertise Reg-Lifetime in FA Service	Timeout Idle/Absolute in Subsc. Template (Source Context)	Idle-Timer Value in Access-Accept	Timeout Idle/Absolute in Subscriber Template(HA Context)	Reg-Lifetime Value in HA Service	Resulting Lifetime Value sent to Mobile Device
600	0(default)	(not returned)	0(default)	7200	600
1800	900	7200	7200	1805	900
3600	1200	3600	1705	3600	1200
1500	3600	1500	0(default)	3600	1500
3600	0(default)	(not returned)	0(default)	2405	2400
3600	0(default)	(not returned)	2005	3600	2000
65534	0(default)	7200	0(default)	3600	Lifetime Too Long

Dealing with the "Requested Lifetime Too Long" Error Code

In some configurations, a roaming partner may return an "Idler-Timer" attribute in an access-accept whose value is smaller than what a carrier may have configured for its own subscribers. This will result in a "Requested Lifetime Too Long" error message being returned to the device. There are several ways to correct this. One is to use a setting in the FA service configuration. Using the "no limit-reg-lifetime" in the FA service configuration will tell the FA service to allow the MIP lifetime to be greater than the Idle or Absolute timers. The FA will not send Error 69 and continue to process the call. The lifetime value in the MIP Request sent to the HA will still be what was determined in Phase 2.


Controlling the Mobile IP Lifetime on a Per-Domain Basis

The system does not support the configuration of the MIP lifetime timer on per-domain (context) basis. However, a domain-wide lifetime timer can be achieved by configuring the idle-timeout attribute for the default subscriber for each domain.

 **Important:** Mobile IP lifetime settings can be controlled on a per-domain basis **only** in deployments for which the idle timeout attribute for individual subscriber profiles is **not** used during operation.

In this configuration, the value of the registration lifetime sent by the system in Agent Advertisements is selected by comparing the configured FA Agent Advertisement lifetime setting, and the idle and/or absolute timeout settings configured for the domain's default subscriber. If the value of the idle and/or absolute timeout parameter is less than the Agent Advertisement lifetime, then the system provides a registration lifetime equal to 5 seconds less than the lowest timer value.

If the idle timeout attribute is configured in individual subscriber profiles, per-domain lifetime control is not possible. In this case, the registration lifetime configured for the FA must be the lower of the two values.

 **Important:** Commands used in the examples in this section provide base functionality to the extent that the most common or likely commands and/or keyword options are presented. In many cases, other optional commands and/or keyword options are available. Refer to the *Command Line Interface Reference* for complete information regarding all commands.

The following is an example CLI command sequence used to configure the Mobile IP lifetime on a per-domain basis.

```
configure

context <aaa_context_name>

  subscriber default

    ip context-name <abc>

  exit

subscriber name <ptt.bigco.com>

  timeout idle <3605>

  ip context-name <abc>

  exit

subscriber name <bigco.com>

  timeout idle <7205>

  ip context-name <abc>

  exit
```

```

domain <ptt.bigco.com> default subscriber <ptt.bigco.com>

domain <bigco.com> default subscriber <bigco.com>

end

configure

context <ha_context_name>

subscriber default

exit    ha-service <ha>

idle-timeout-mode normal      reg-lifetime <7200>

end

configure

context <fa_context_name>

fa-service <fa>

advertise reg-lifetime <7200>

end

```

In the example above, two domains (ptt.bigco.com and bigco.com) are configured. The default subscribers are defined for the two domains respectively. The desired operation requires a Mobile IP lifetime of 1 hour (3600 secs) for the ptt.bigco.com domain, and a lifetime of 2 hours (7200 secs) for the bigco.com domain.

Whenever a subscriber session belonging to the ptt.bigco.com domain arrives, the system uses a Mobile IP lifetime timer value equal to 5 seconds less than the idle timeout configured for the default subscriber because the configured value is less than the registration lifetime value configured for the Agent Advertisement. 5 seconds less than the configured value of 3605 seconds equals 3600 seconds which meets the desired operation.

Whenever a subscriber session belonging to the bigco.com domain arrives, the system uses the configured registration lifetime value as the Mobile IP lifetime in Agent Advertisements because it is less than the configured idle timeout in the default subscriber's profile.

As a general rule, the registration lifetime value on the agent **must** be configured as the highest Mobile IP lifetime that is desired for a subscriber. (In the above example, it would be the subscriber bigco.com.)

Another important factor to consider is that the idle timeout value should be reset on receipt of a renewal request. To support this operation, the system provides the **idle-timeout-mode** configurable in the HA service. The following modes are supported:

- **normal**: Resets the idle timeout value on receipt of Mobile IP user data and control signaling
- **aggressive**: Resets the idle timeout value on receipt of Mobile IP user data only (this is the default behavior)
- **handoff**: Resets the idle timeout value on receipt of Mobile IP user data and upon inter-AGW handoff or inter access technologies

The following optional modifier is also supported:

- **upstream-only**: Only upstream user data (data from the mobile node) resets the idle timer for the session. This is disabled by default.

Appendix D

Always-on

This chapter provides information on configuring an enhanced, or extended, service. The product Administration Guides provide examples and procedures for configuration of basic services on the system. It is recommended that you select the configuration example that best meets your service model, and configure the required elements for that model, as described in this Administration Guide, before using the procedures in this chapter.

This chapter contains the following sections:

- [Overview](#)
- [Configuring Always-on](#)

Overview

Always-on is enabled for each subscriber individually through a local subscriber profile or a RADIUS profile. Always-on is disabled for all subscribers by default.

If Always-on is enabled for a subscriber, when the idle time-out limit is reached the subscribers IP/PPP session remains connected as long as the subscriber is reachable. This is true even if the airlink between the mobile device and the RN (Radio Node) is moved from active to dormant (inactive) status. When the idle timeout limit is reached, the PDSN determines Mobile Node availability using LCP keepalive messages. A response to these messages indicates that the “always-on” status should be maintained. Failure to respond to a predetermined number of LCP keepalive messages causes the PDSN to tear-down (disconnect) the subscriber session.



Caution: When always-on is enabled, the subscriber must have an idle time-out period configured (default is 0, no time-out). Failure to configure an idle time-out results in the LCP keepalive messages never being sent and the subscriber session stays up indefinitely.

The RADIUS attribute **3GPP2-Always-On** defined in a subscriber profile stored remotely on a RADIUS server can be used to enable Always-on for the subscriber. The attribute has two possible values, **inactive** and **active**. To enable Always-on, set the attribute to **active**.

For more information on the attributes, if you are using StarOS 12.3 or an earlier release, refer to the *AAA and GTPP Interface Administration and Reference*. If you are using StarOS 14.0 or a later release, refer to the *AAA Interface Administration and Reference*.

Configuring Always-on

To configure Always-on for a subscriber:

- Step 1** Configure Always-on as described in the [Configuring Always-on](#) section.
- Step 2** Verify your configuration as described in the [Verifying Your Configuration](#) section.
- Step 3** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.



Important: Commands used in the configuration examples in this section provide base functionality to the extent that the most common or likely commands and/or keyword options are presented. In many cases, other optional commands and/or keyword options are available. Refer to the *Command Line Interface Reference* for complete information regarding all commands.

Configuring Always-on

Use the following example to configure Always-on:

configure

```
context <context_name>

  subscriber name <subscriber_name>

  timeout idle <seconds>

  always-on

end
```

Notes:

- *<context_name>* must be the name of the destination context where the subscriber that you want to enable always-on is configured.
- *Option:* To configure the echo-retransmit-timeout setting to wait before sending a keepalive message to an always-on subscriber, in the Context Configuration Mode, enter the following command:
ppp echo-retransmit-timeout <milliseconds>
- *Option:* To configure the echo-max-retransmissions setting to retransmit a Keepalive message to a subscriber, in the Context Configuration Mode use the following command:
ppp echo-max-retransmissions <num_retries>
- The optional echo-retransmit-timeout and echo-max-retransmissions settings apply to all subscriber sessions within the current context that have always-on enabled.
- *Option:* To configure the long duration timer for the subscriber, in the Subscriber Configuration Mode, enter the following command:

```
timeout long-duration <ld_timeout> [ inactivity-time <inact_timeout>]
```

- *Option:* To configure the long duration timer detection to trigger long duration timer action for the subscriber, in the Subscriber Configuration Mode enter the following command:

```
long-duration-action detection
```

- *Option:* To configure the long duration timer action for sessions exceeding the long duration timer timeout or the idle timeout durations for the subscriber, in the Subscriber Configuration Mode enter the following command:

```
long-duration-action disconnection [ suppress-notification ] [ dormant-only ] +
```

Verifying Your Configuration

To verify your configuration:

- Step 1** Change to the context where Always-on was configured by entering the following command:

```
context <context_name>
```

- Step 2** View the subscriber's configuration by entering the following command:

```
show subscriber configuration username <name>
```

Output of the command displays the subscriber's configurations. Examine the output for the idle timeout and always-on fields.

Appendix E

Broadcast Multicast Service

This chapter provides information on Broadcast Multicast Service (BCMCS) functionality. The product Administration Guides provide examples and procedures for configuration of basic services on the system. It is recommended that you select the configuration example that best meets your service model, and configure the required elements for that model, as described in this Administration Guide, before using the procedures in this chapter.



Important: The features described in this chapter are only available if you have purchased and installed a feature license for Broadcast & Multicast Services.

This chapter contains the following sections:

- [Overview](#)
- [Configuring BCMCS](#)

Overview

BCMCS eliminates unnecessary replication of data on CDMA2000 wireless networks by transmitting a single stream of data to multiple users. By delivering a single, unidirectional data stream to many subscribers, BCMCS makes more efficient use of wireless network resources than traditional point to point connections.

BCMCS functionality on the system is provided by an existing PDSN service and is enabled by a valid Broadcast & Multicast Services license. In the absence of a valid license, the system functions as a standard unicast PDSN. When a PDSN is functioning in a BCMCS environment, it is designated as a Broadcast Serving Node (BSN). The main features supported by the Broadcast & Multicast Services license are:

- Multicast proxy-host functionality.
- Support for BCMCS-specific A11 messages.
- Authentication of BCMCS flow-IDs using a BCMCS controller.
- Establishment and teardown of BCMCS bearer paths using the multicast framework.
- Support for framing HDLC-like and segment based framing.
- Accounting for the BCMCS flows to charge the originator of the content.

Licensing

The BCMCS is a licensed Cisco feature. A separate feature license may be required. Contact your Cisco account representative for detailed information on specific licensing requirements. For information on installing and verifying licenses, refer to the *Managing License Keys* section of the *Software Management Operations* chapter in the *System Administration Guide*.

Configuring BCMCS

To configure the system for BCMCS:

- Step 1** Configure the system for PDSN functionality as described in the this Administration Guide.
- Step 2** Set the BCMCS group user name and password for RADIUS access as described in the [BCMCS Group Configuration](#) section.
- Step 3** Create a multicast group profile on your RADIUS server to achieve BCMCS functionality as described in the [RADIUS Server Configuration](#) section.
- Step 4** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.



Important: Commands used in the configuration examples in this section provide base functionality to the extent that the most common or likely commands and/or keyword options are presented. In many cases, other optional commands and/or keyword options are available. Refer to the *Command Line Interface Reference* for complete information regarding all commands.

BCMCS Group Configuration

Use the following example to set the BCMCS group user name and password for RADIUS access:

configure

```
context context_name

    pdsn-service pdsn_service_name

    bcmcs grpusrname group_name

    bcmcs grppasswd group_password

end
```

RADIUS Server Configuration

You must create a multicast group profile on your RADIUS server to achieve BCMCS functionality. The group name and password must be the same as configured in [BCMCS Group Configuration](#) section.

For information about the supported BCMCS attributes, refer to the *AAA and GTP Interface Administration and Reference*.

Appendix F

CoA, RADIUS DM, and Session Redirection (Hotlining)

This chapter describes Change of Authorization (CoA), Disconnect Message (DM), and Session Redirect (Hotlining) support in the system. RADIUS attributes, Access Control Lists (ACLs) and filters that are used to implement these features are discussed. The product administration guides provide examples and procedures for configuration of basic services on the system. It is recommended that you select the configuration example that best meets your service model, and configure the required elements for that model, as described in this Administration Guide, before using the procedures in this chapter.



Important: Not all functions, commands, and keywords/variables are available or supported for all network function or services. This depends on the platform type and the installed license(s).

RADIUS Change of Authorization and Disconnect Message

This section describes how the system implements CoA and DM RADIUS messages and how to configure the system to use and respond to CoA and DM messages.

CoA Overview

The system supports CoA messages from the AAA server to change data filters associated with a subscriber session. The CoA request message from the AAA server must contain attributes to identify NAS and the subscriber session and a data filter ID for the data filter to apply to the subscriber session. The filter-id attribute (attribute ID 11) contains the name of an Access Control List (ACL). For detailed information on configuring ACLs, refer to the *IP Access Control Lists* chapter in the *System Administration Guide*.

If the system successfully executes a CoA request, a CoA-ACK message is sent back to the RADIUS server and the data filter is applied to the subscriber session. Otherwise, a CoA-NAK message is sent with an error-cause attribute without making any changes to the subscriber session.



Important: Changing ACL and rulebase together in a single CoA is not supported. For this, two separate CoA requests can be sent through AAA server requesting for one attribute change per request.

DM Overview

The DM message is used to disconnect subscriber sessions in the system from a RADIUS server. The DM request message should contain necessary attributes to identify the subscriber session. If the system successfully disconnects the subscriber session, a DM-ACK message is sent back to the RADIUS server, otherwise, a DM-NAK message is sent with proper error reasons.

License Requirements

The RADIUS Change of Authorization (CoA) and Disconnect Message (DM) are licensed Cisco features. A separate feature license may be required. Contact your Cisco account representative for detailed information on specific licensing requirements. For information on installing and verifying licenses, refer to the *Managing License Keys* section of the *Software Management Operations* chapter in the *System Administration Guide*.

Enabling CoA and DM

To enable RADIUS Change of Authorization and Disconnect Message:

- Step 1** Enable the system to listen for and respond to CoA and DM messages from the RADIUS server as described in the [Enabling CoA and DM](#) section.
- Step 2** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Step 3 View CoA and DM message statistics as described in the [Viewing CoA and DM Statistics](#) section.



Important: Commands used in the configuration examples in this section provide base functionality to the extent that the most common or likely commands and/or keyword options are presented. In many cases, other optional commands and/or keyword options are available. Refer to the *Command Line Interface Reference* for complete information regarding all commands. Not all commands and keywords/variables are available or supported. This depends on the platform type and the installed license(s).

Enabling CoA and DM

Use the following example to enable the system to listen for and respond to CoA and DM messages from the RADIUS server:

configure

```
context <context_name>

    radius change-authorize-nas-ip <ipv4/ipv6_address>

end
```

Notes:

- `<context_name>` must be the name of the AAA context where you want to enable CoA and DM.
For more information on configuring the AAA context, if you are using StarOS 12.3 or an earlier release, refer to the *Configuring Context-Level AAA Functionality* section of the *AAA and GTPP Interface Administration and Reference*. If you are using StarOS 14.0 or a later release, refer to the *AAA Interface Administration and Reference*.
- A number of optional keywords and variables are available for the **radius change-authorize-nas-ip** command. For more information regarding this command please refer to the *Command Line Interface Reference*.

CoA and DM Attributes

For CoA and DM messages to be accepted and acted upon, the system and subscriber session to be affected must be identified correctly.

To identify the system, use any one of the following attributes:

- NAS-IP-Address: NAS IP address if present in the CoA/DM request should match with the NAS IP address.
- NAS-Identifier: If this attribute is present, its value should match to the nas-identifier generated for the subscriber session

To identify the subscriber session, use any one of the following attributes.

- If 3GPP2 service is configured the following attribute is used for correlation identifier:
 - 3GPP2-Correlation-ID: The values should exactly match the 3GPP2-correlation-id of the subscriber session. This is one of the preferred methods of subscriber session identification.
- If 3GPP service is configured the following attributes are used for different identifiers:
 - 3GPP-IMSI: International Mobile Subscriber Identification (IMSI) number should be validated and matched with the specified IMSI for specific PDP context.

- 3GPP-NSAPI: Network Service Access Point Identifier (NSAPI) should match to the NSAPI specified for specific PDP context.
- User-Name: The value should exactly match the subscriber name of the session. This is one of the preferred methods of subscriber session identification.
- Framed-IP-Address: The values should exactly match the framed IP address of the session.
- Calling-station-id: The value should match the Mobile Station ID.

To specify the ACL to apply to the subscriber session, use the following attribute:

- Filter-ID: CoA only. This must be the name of an existing Access Control List. If this is present in a CoA request, the specified ACL is immediately applied to the specified subscriber session. The Context Configuration mode command, **radius attribute filter-id direction**, controls in which direction filters are applied.

The following attributes are also supported:

- Event-Timestamp: This attribute is a timestamp of when the event being logged occurred.
- If 3GPP2 service is configured following additional attributes are supported:
 - 3GPP2-Disconnect-Reason: This attribute indicates the reason for disconnecting the user. This attribute may be present in the RADIUS Disconnect-request Message from the Home Radius server to the PDSN.
 - 3GPP2-Session-Termination-Capability: When CoA and DM are enabled by issuing the radius change-authorize-nas-ip command, this attribute is included in a RADIUS Access-request message to the Home RADIUS server and contains the value 3 to indicate that the system supports both Dynamic authorization with RADIUS and Registration Revocation for Mobile IPv4. The attribute is also included in the RADIUS Access-Accept message and contains the preferred resource management mechanism by the home network, which is used for the session and may include values 1 through 3.

CoA and DM Error-Cause Attribute

The Error-Cause attribute is used to convey the results of requests to the system. This attribute is present when a CoA or DM NAK or ACK message is sent back to the RADIUS server.

The value classes of error causes are as follows:

- 0-199, 300-399 reserved
- 200-299 - successful completion
- 400-499 - errors in RADIUS server
- 500-599 - errors in NAS/Proxy

The following error cause is sent in ACK messages upon successful completion of a CoA or DM request:

- 201- Residual Session Context Removed

The following error causes are sent in NAK messages when a CoA or DM request fails:

- 401 - Unsupported Attribute
- 402 - Missing Attribute
- 403 - NAS Identification Mismatch
- 404 - Invalid Request
- 405 - Unsupported Service

- 406 - Unsupported Extension
- 501 - Administratively Prohibited
- 503 - Session Context Not Found
- 504 - Session Context Not Removable
- 506 - Resources Unavailable

Viewing CoA and DM Statistics

View CoA and DM message statistics by entering the following command:

```
show session subsystem facility aaamgr
```

The following is a sample output of this command.

```

1 AAA Managers

807 Total aaa requests                0 Current aaa requests
379 Total aaa auth requests           0 Current aaa auth requests
    0 Total aaa auth probes            0 Current aaa auth probes
    0 Total aaa auth keepalive          0 Current aaa auth keepalive
426 Total aaa acct requests           0 Current aaa acct requests
    0 Total aaa acct keepalive          0 Current aaa acct keepalive
379 Total aaa auth success             0 Total aaa auth failure
    0 Total aaa auth purged            0 Total aaa auth cancelled
    0 Total auth keepalive success      0 Total auth keepalive failure
    0 Total auth keepalive purged
    0 Total aaa auth DMU challenged

367 Total radius auth requests        0 Current radius auth requests
    2 Total radius auth requests retried
    0 Total radius auth responses dropped
    0 Total local auth requests         0 Current local auth requests
12 Total pseudo auth requests         0 Current pseudo auth requests
    0 Total null-username auth requests (rejected)
    0 Total aaa acct completed          0 Total aaa acct purged
    0 Total acct keepalive success      0 Total acct keepalive timeout

```

■ RADIUS Change of Authorization and Disconnect Message

```

0 Total acct keepalive purged
0 Total aaa acct cancelled
426 Total radius acct requests          0 Current radius acct requests
0 Total radius acct requests retried
0 Total radius acct responses dropped
0 Total gtpa acct requests              0 Current gtpa acct requests
0 Total gtpa acct cancelled             0 Total gtpa acct purged
0 Total null acct requests              0 Current null acct requests
54 Total aaa acct sessions              5 Current aaa acct sessions
3 Total aaa acct archived               0 Current aaa acct archived
0 Current recovery archives             0 Current valid recovery records
2 Total aaa sockets opened              2 Current aaa sockets open
0 Total aaa requests pend socket open
0 Current aaa requests pend socket open
0 Total radius requests pend server max-outstanding
0 Current radius requests pend server max-outstanding
0 Total aaa radius coa requests          0 Total aaa radius dm requests
0 Total aaa radius coa acks             0 Total aaa radius dm acks
0 Total aaa radius coa naks             0 Total aaa radius dm naks
2 Total radius charg auth               0 Current radius charg auth
0 Total radius charg auth succ          0 Total radius charg auth fail
0 Total radius charg auth purg          0 Total radius charg auth cancel
0 Total radius charg acct              0 Current radius charg acct
0 Total radius charg acct succ          0 Total radius charg acct purg
0 Total radius charg acct cancel
357 Total gtpa charg                   0 Current gtpa charg
357 Total gtpa charg success            0 Total gtpa charg failure
0 Total gtpa charg cancel               0 Total gtpa charg purg
0 Total prepaid online requests         0 Current prepaid online requests

```

0 Total prepaid online success	0 Current prepaid online failure
0 Total prepaid online retried	0 Total prepaid online cancelled
0 Current prepaid online purged	
0 Total aaamgr purged requests	
0 SGSN: Total db records	
0 SGSN: Total sub db records	
0 SGSN: Total mm records	
0 SGSN: Total pdp records	
0 SGSN: Total auth records	

Session Redirection (Hotlining)



Important: Functionality described for this feature in this segment is not applicable for HNB-GW sessions.

Overview

Session redirection provides a means to redirect subscriber traffic to an external server by applying ACL rules to the traffic of an existing or a new subscriber session. The destination address and optionally the destination port of TCP/IP or UDP/IP packets from the subscriber are rewritten so the packet is forwarded to the designated redirected address. Return traffic to the subscriber has the source address and port rewritten to the original values. The redirect ACL may be applied dynamically by means of the RADIUS Change of Authorization (CoA) feature.

Note that the session redirection feature is only intended to redirect a very small subset of subscribers at any given time. The data structures allocated for this feature are kept to the minimum to avoid large memory overhead in the session managers.

License Requirements

The Session Redirection (Hotlining) is a licensed Cisco feature. A separate feature license may be required. Contact your Cisco account representative for detailed information on specific licensing requirements. For information on installing and verifying licenses, refer to the *Managing License Keys* section of the *Software Management Operations* chapter in the *System Administration Guide*.

Operation

ACL Rule

An ACL rule named **readdress server** supports redirection of subscriber sessions. The ACL containing this rule must be configured in the destination context of the user. Only TCP and UDP protocol packets are supported. The ACL rule allows specifying the redirected address and an optional port. The source and destination address and ports (with respect to the traffic originating from the subscriber) may be wildcarded. If the redirected port is not specified, the traffic will be redirected to the same port as the original destination port in the datagrams. For detailed information on configuring ACLs, refer to the *IP Access Control Lists* chapter in the *System Administration Guide*. For more information on **readdress server**, refer to the *ACL Configuration Mode Commands* chapter of the *Command Line Interface Reference*.

Redirecting Subscriber Sessions

An ACL with the **readdress server** rule is applied to an existing subscriber session through CoA messages from the RADIUS server. The CoA message contains the 3GPP2-Correlation-ID, User-Name, Acct-Session-ID, or Framed-IP-Address attributes to identify the subscriber session. The CoA message also contains the Filter-Id attribute which specifies the name of the ACL with the **readdress server** rule. This enables applying the ACL dynamically to existing subscriber sessions. By default, the ACL is applied as both the input and output filter for the matching subscriber unless the Filter-Id in the CoA message bears the prefix **in:** or **out:**.

For information on CoA messages and how they are implemented in the system, refer to the [RADIUS Change of Authorization and Disconnect Message](#) section.



Important: Changing ACL and rulebase together in a single CoA is not supported. For this, two separate CoA requests can be sent through AAA server requesting for one attribute change per request.

Session Limits On Redirection

To limit the amount of memory consumed by a session manager a limit of 2000 redirected session entries per session manager is allocated. This limit is equally shared by the set of subscribers who are currently being redirected. Whenever a redirected session entry is subject to revocation from a subscriber due to an insufficient number of available session entries, the least recently used entry is revoked.

Stopping Redirection

The redirected session entries for a subscriber remain active until a CoA message issued from the RADIUS server specifies a filter that does not contain the readdress server ACL rule. When this happens, the redirected session entries for the subscriber are deleted.

All redirected session entries are also deleted when the subscriber disconnects.

Handling IP Fragments

Since TCP/UDP port numbers are part of the redirection mechanism, fragmented IP datagrams must be reassembled before being redirected. Reassembly is particularly necessary when fragments are sent out of order. The session manager performs reassembly of datagrams and reassembly is attempted only when a datagram matches the redirect server ACL rule. To limit memory usage, only up to 10 different datagrams may be concurrently reassembled for a subscriber. Any additional requests cause the oldest datagram being reassembled to be discarded. The reassembly timeout is set to 2 seconds. In addition, the limit on the total number of fragments being reassembled by a session manager is set to 1000. If this limit is reached, the oldest datagram being reassembled in the session manager and its fragment list are discarded. These limits are not configurable.

Recovery

When a session manager dies, the ACL rules are recovered. The session redirect entries have to be re-created when the MN initiates new traffic for the session. Therefore when a crash occurs, traffic from the Internet side is not redirected to the MN.

AAA Accounting

Where destination-based accounting is implemented, traffic from the subscriber is accounted for using the original destination address and not the redirected address.

Viewing the Redirected Session Entries for a Subscriber

View the redirected session entries for a subscriber by entering the following command:

```
show subscribers debug-info { callid <id> | msid <id> | username <name> }
```

The following command displays debug information for a subscriber with the MSID 0000012345:

```
show subscribers debug-info msid 0000012345
```

The following is a sample output of this command:

```
username: user1 callid: 01callb1 msid: 0000100003
```

```
Card/Cpu: 4/2
```

```
Sessmgr Instance: 7
```

```
Primary callline:
```

```
Redundancy Status: Original Session
```

```
Checkpoints Attempts Success Last-Attempt Last-Success
```

```
Full: 27 26 15700ms 15700ms
```

```
Micro: 76 76 4200ms 4200ms
```

```
Current state: SMGR_STATE_CONNECTED
```

```
FSM Event trace:
```

```
State Event
```

```
SMGR_STATE_OPEN SMGR_EVT_NEWCALL SMGR_STATE_NEWCALL_ARRIVED SMGR_EVT_ANSWER_CALL
SMGR_STATE_NEWCALL_ANSWERED SMGR_EVT_LINE_CONNECTED SMGR_STATE_LINE_CONNECTED
SMGR_EVT_LINK_CONTROL_UP SMGR_STATE_LINE_CONNECTED SMGR_EVT_AUTH_REQ
```

```
SMGR_STATE_LINE_CONNECTED SMGR_EVT_IPADDR_ALLOC_SUCCESS
```

```
SMGR_STATE_LINE_CONNECTED SMGR_EVT_AUTH_SUCCESS
```

```
SMGR_STATE_LINE_CONNECTED SMGR_EVT_UPDATE_SESS_CONFIG
```

```
SMGR_STATE_LINE_CONNECTED SMGR_EVT_LOWER_LAYER_UP
```

```
Data Reorder statistics
```

```
Total timer expiry: 0 Total flush (tmr expiry): 0
```

```
Total no buffers: 0 Total flush (no buffers): 0
```

```
Total flush (queue full): 0 Total flush (out of range): 0
```

```
Total flush (svc change): 0 Total out-of-seq pkt drop: 0
```

```
Total out-of-seq arrived: 0
```

```
IPv4 Reassembly Statistics:
```

```
Success: 0 In Progress: 0
```

```
Failure (timeout): 0 Failure (no buffers): 0
```

```
Failure (other reasons): 0
```

Redirected Session Entries:

Allowed: 2000 Current: 0

Added: 0 Deleted: 0

Revoked for use by different subscriber: 0

Peer callline:

Redundancy Status: Original Session

Checkpoints Attempts Success Last-Attempt Last-Success

Full: 0 0 0ms 0ms

Micro: 0 0 0ms 0ms

Current state: SMGR_STATE_CONNECTED

FSM Event trace:

State Event

SMGR_STATE_OPEN SMGR_EVT_MAKECALL

SMGR_STATE_MAKECALL_PENDING SMGR_EVT_LINE_CONNECTED

SMGR_STATE_LINE_CONNECTED SMGR_EVT_LOWER_LAYER_UP

SMGR_STATE_CONNECTED SMGR_EVT_AUTH_REQ

SMGR_STATE_CONNECTED SMGR_EVT_AUTH_SUCCESS

SMGR_STATE_CONNECTED SMGR_EVT_REQ_SUB_SESSION

SMGR_STATE_CONNECTED SMGR_EVT_RSP_SUB_SESSION

username: user1 callid: 01callb1 msid: 0000100003

Card/Cpu: 4/2

Sessmgr Instance: 7

Primary callline:

Redundancy Status: Original Session

Checkpoints Attempts Success Last-Attempt Last-Success

Full: 27 26 15700ms 15700ms

Micro: 76 76 4200ms 4200ms

Current state: SMGR_STATE_CONNECTED

FSM Event trace:

State Event

```

SMGR_STATE_OPEN SMGR_EVT_NEWCALL

SMGR_STATE_NEWCALL_ARRIVED SMGR_EVT_ANSWER_CALL

SMGR_STATE_NEWCALL_ANSWERED SMGR_EVT_LINE_CONNECTED

SMGR_STATE_LINE_CONNECTED SMGR_EVT_LINK_CONTROL_UP

SMGR_STATE_LINE_CONNECTED SMGR_EVT_AUTH_REQ

SMGR_STATE_LINE_CONNECTED SMGR_EVT_IPADDR_ALLOC_SUCCESS

SMGR_STATE_LINE_CONNECTED SMGR_EVT_AUTH_SUCCESS

SMGR_STATE_LINE_CONNECTED SMGR_EVT_UPDATE_SESS_CONFIG

SMGR_STATE_LINE_CONNECTED SMGR_EVT_LOWER_LAYER_UP

```

Data Reorder statistics

```

Total timer expiry: 0 Total flush (tmr expiry): 0

Total no buffers: 0 Total flush (no buffers): 0

Total flush (queue full): 0 Total flush (out of range):0

Total flush (svc change): 0 Total out-of-seq pkt drop: 0

    Total out-of-seq arrived: 0

```

IPv4 Reassembly Statistics:

```

Success: 0 In Progress: 0

Failure (timeout): 0 Failure (no buffers): 0

Failure (other reasons): 0

```

Redirected Session Entries:

```

Allowed: 2000 Current: 0

Added: 0 Deleted: 0

Revoked for use by different subscriber: 0

```

Peer callline:

```

Redundancy Status: Original Session

Checkpoints Attempts Success Last-Attempt Last-Success

Full: 0 0 0ms 0ms

Micro: 0 0 0ms 0ms

```

```
Current state: SMGR_STATE_CONNECTED

FSM Event trace:

State Event

SMGR_STATE_OPEN SMGR_EVT_MAKECALL

SMGR_STATE_MAKECALL_PENDING SMGR_EVT_LINE_CONNECTED

SMGR_STATE_LINE_CONNECTED SMGR_EVT_LOWER_LAYER_UP

SMGR_STATE_CONNECTED SMGR_EVT_AUTH_REQ

SMGR_STATE_CONNECTED SMGR_EVT_AUTH_SUCCESS

SMGR_STATE_CONNECTED SMGR_EVT_REQ_SUB_SESSION

SMGR_STATE_CONNECTED SMGR_EVT_RSP_SUB_SESSION

SMGR_STATE_CONNECTED SMGR_EVT_ADD_SUB_SESSION

SMGR_STATE_CONNECTED SMGR_EVT_AUTH_REQ

SMGR_STATE_CONNECTED SMGR_EVT_AUTH_SUCCESS

Data Reorder statistics

Total timer expiry: 0 Total flush (tmr expiry): 0

Total no buffers: 0 Total flush (no buffers): 0

Total flush (queue full): 0 Total flush (out of range):0

Total flush (svc change): 0 Total out-of-seq pkt drop: 0

Total out-of-seq arrived: 0

IPv4 Reassembly Statistics:

Success: 0 In Progress: 0

Failure (timeout): 0 Failure (no buffers): 0

Failure (other reasons): 0

Redirected Session Entries:

Allowed: 2000 Current: 0

Added: 0 Deleted: 0

Revoked for use by different subscriber: 0
```


Appendix G

Gx Interface Support

This chapter provides information on configuring Gx interface to support policy and charging control for subscribers.

The IMS service provides application support for transport of voice, video, and data independent of access support. Roaming IMS subscribers require apart from other functionality sufficient, uninterrupted, consistent, and seamless user experience during an application session. It is also important that a subscriber gets charged only for the resources consumed by the particular IMS application used.

It is recommended that before using the procedures in this chapter you select the configuration example that best meets your service model, and configure the required elements for that model as described in this Administration Guide.

The following topics are covered in this chapter:

- [Rel. 6 Gx Interface](#)
- [Rel. 7 Gx Interface](#)
- [Rel. 8 Gx Interface](#)
- [Rel. 9 Gx Interface](#)

Rel. 6 Gx Interface

Rel. 6 Gx interface support is available on the Cisco ASR chassis running StarOS 8.0 and later releases for the following products:

- GGSN
- IPSG



Important: In 14.0 and later releases, Rel. 6 Gx interface functionality is not supported on the chassis.

This section describes the following topics:

- [Introduction](#)
- [How it Works](#)
- [Configuring Rel. 6 Gx Interface](#)

Introduction

In GPRS/UMTS networks, the client functionality lies with the GGSN/IPSG, therefore in the IMS authorization scenario it is also called Access Gateway (AGW).

The provisioning of charging rules that are based on the dynamic analysis of flows used for the IMS session is carried out over the Gx interface. In 3GPP, Rel. 6 the Gx is an interface between Access Gateway functioning as Traffic Plane Function (TPF) and the Charging Rule Function (CRF). It is based on the Diameter Base Protocol (DIABASE) and the Diameter Credit Control Application (DCCA) standard. The GGSN/TPF acts as the client where as the CRF contains the Diameter server functionality.

The AGW is required to perform query, in reply to which the servers provision certain policy or rules that are enforced at the AGW for that particular subscriber session. The CRF analyzes the IP flow data, which in turn has been retrieved from the Session Description Protocol (SDP) data exchanged during IMS session establishment.



Important: In addition to standard Gx interface functionality, the Gx interface implemented here provides support of SBLP with additional AVPs in custom DPCA dictionaries. For more information on customer-specific support contact your Cisco account representative. In view of required flow bandwidth and QoS, the system provides enhanced support for use of Service Based Local Policy (SBLP) to provision and control the resources used by the IMS subscriber. SBLP is based on the dynamic parameters such as the media/traffic flows for data transport, network conditions and static parameters, such as subscriber configuration and category. It also provides Flow-based Charging (FBC) mechanism to charge the subscriber dynamically based on content usage. With this additional functionality, the Cisco Systems Gateway can act as an Enhanced Policy Decision Function (E-PDF).

Supported Networks and Platforms

This feature is supported on all chassis with StarOS Release 8.0 or later running GGSN service for the core network services.

License Requirements

The Rel. 6 Gx interface support is a licensed Cisco feature. A separate feature license may be required. Contact your Cisco account representative for detailed information on specific licensing requirements. For information on installing and verifying licenses, refer to the *Managing License Keys* section of the *Software Management Operations* chapter in the *System Administration Guide*.

Supported Standards

The Rel 6. Gx interface support is based on the following standards and request for comments (RFCs):

- 3GPP TS 29.210, Charging rule provisioning over Gx interface



Important: Note that Charging rule provisioning over Gx interface functionality is not supported in 14.0 and later releases.

- RFC 3588, Diameter Base Protocol; September 2003
- RFC 4006, Diameter Credit-Control Application; August 2005

In addition to the above RFCs and standards, IMS Authorization partially supports 3GPP TS 29.212 for Policy and Charging Control over Gx reference point functionality.

How it Works

This section describes the IMS authorization and dynamic policy support in GPRS/UMTS networks.

The following figure and table explain the IMS authorization process between a system and IMS components that is initiated by the MN.

In the case of GGSN, the DPCA is the Gx interface to the Control and Charging Rule Function (CRF). In this context CRF will act as Enhanced Policy Decision Function (E-PDF). The CRF may reside in Proxy-Call Session Control Function (P-CSCF) or on stand-alone system.

The interface between IMSA with CRF is the Gx interface, and between Session Manager and Online Charging Service (OCS) is the Gy interface.

Note that the IMS Authorization (IMSA) service and Diameter Policy Control Application (DPCA) are part of Session Manager on the system, and separated in the following figure for illustration purpose only.

Figure 25. Rel. 6 Gx IMS Authorization Call Flow

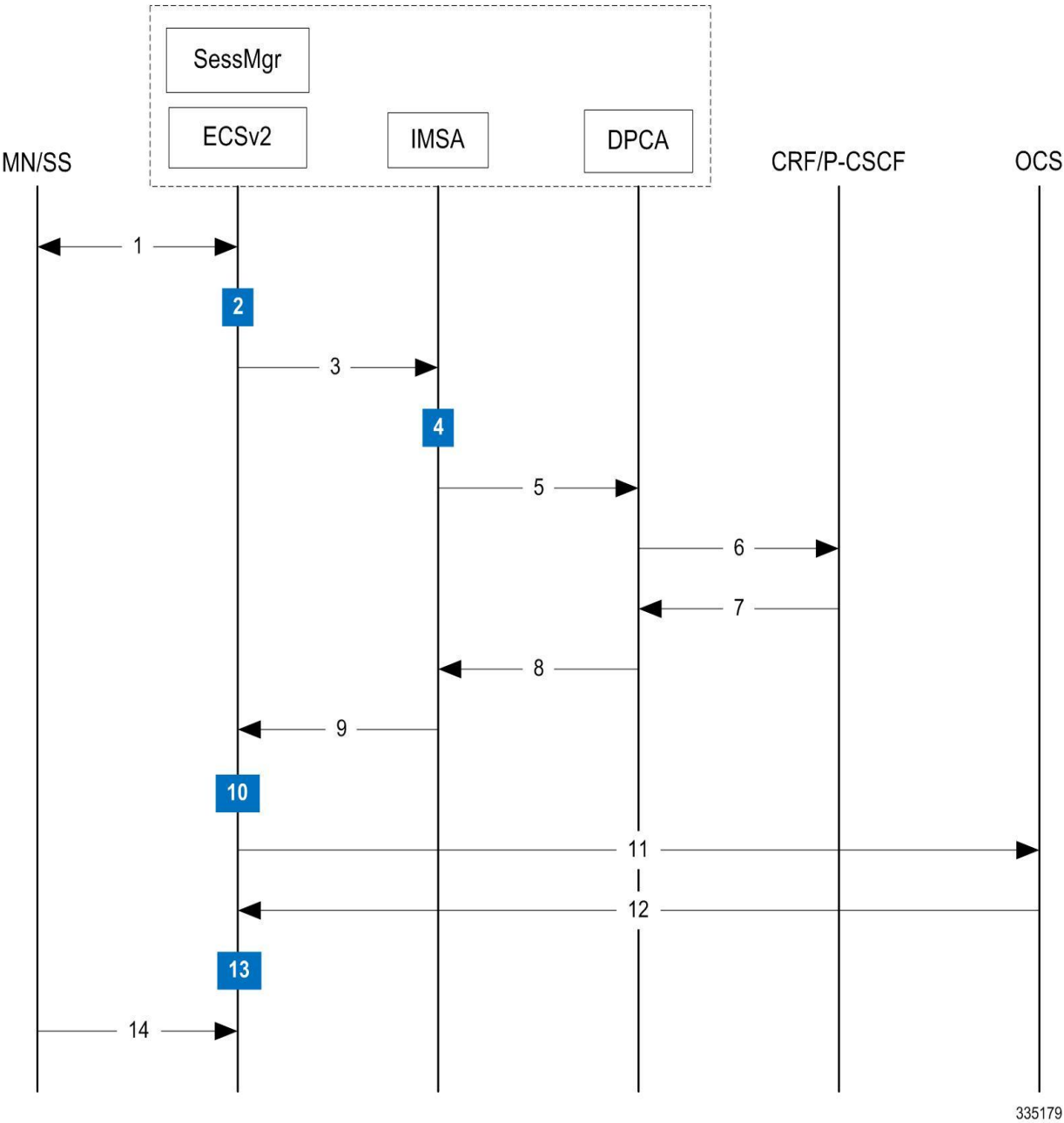


Table 31. Rel. 6 Gx IMS Authorization Call flow Description

Step	Description
1	IMS subscriber (MN) sends request for primary PDP context activation/creation.
2	Session manager allocates IP address to MN.

Step	Description
3	Session manager sends IMS authorization request to IMS Authorization service (IMSA).
4	IMSA creates a session with the CRF on the basis of CRF configuration.
5	IMSA sends request to DPCA module to issue the authorization request to selected CRF.
6	DPCA sends a CCR-initial message to the selected CRF. This message includes the IP address allocated to MN.
7	CCA message sent to DPCA. If a preconfigured rule set for the PDP context is provided in CRF, it sends that charging rules to DPCA in CCA message.
8	DPCA module calls the callback function registered with it by IMSA.
9	After processing the charging rules, IMSA sends Policy Authorization Complete message to session manager.
10	The rules received in CCA message are used for dynamic rule configuration structure and session manager sends the message to ECS.
11	ECS installs the rules and performs credit authorization by sending CCR-Initial to Online Charging System (OCS) with CC-Request-Type set to INITIAL_REQUEST to open the credit control session. This request includes the active rule base ID and 3GPP specific attributes (for example, APN, QoS and so on).
12	OCS returns a CCA-Initial message to activate the statically configured rulebase and includes preemptive credit quotas.
13	ECS responds to session manager with the response message for dynamic rule configuration.
14	On the basis of response for the PDP context authorization, Session Manager sends the response to the MN and activates/rejects the call.

Configuring Rel. 6 Gx Interface

To configure Rel. 6 Gx interface functionality:

- Step 1** Configure the IMS Authorization Service at the context level for an IMS subscriber in GPRS/UMTS network as described in the [Configuring IMS Authorization Service at Context Level](#) section.
- Step 2** Verify your configuration, as described in the [Verifying IMS Authorization Service Configuration](#) section.
- Step 3** Configure an APN within the same context to use the IMS Authorization service for an IMS subscriber as described in the [Applying IMS Authorization Service to an APN](#) section.
- Step 4** Verify your configuration as described in the [Verifying Subscriber Configuration](#) section.
- Step 5** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.



Important: Commands used in the configuration examples in this section provide base functionality to the extent that the most common or likely commands and/or keyword options are presented. In many cases, other optional commands and/or keyword options are available. Refer to the *Command Line Interface Reference* for complete information regarding all commands.

Configuring IMS Authorization Service at Context Level

Use the following example to configure IMS Authorization Service at context level for IMS subscribers in GPRS/UMTS networks:

```
configure

context <context_name>

    ims-auth-service <imsa_service_name>

        p-cscf table { 1 | 2 } row-precedence <precedence_value> { address <ip_address>
| ipv6-address <ipv6_address> }

        p-cscf discovery { table { 1 | 2 } [ algorithm { ip-address-modulus | msisd-
modulus | round-robin } ] | diameter-configured }

        policy-control

            diameter origin endpoint <endpoint_name>

            diameter dictionary <dictionary>

            failure-handling cc-request-type { any-request | initial-request | terminate-
request | update-request } { diameter-result-code { any-error | <result_code> [ to
<end_result_code> ] } } { continue | retry-and-terminate | terminate }

            diameter host-select row-precedence <precedence_value> table { 1 | 2 } host
<host_name> [ realm <realm_name> ] [ secondary host <host_name> [ realm <realm_name> ] ]

            diameter host-select reselect subscriber-limit <subscriber_limit> time-
interval <duration>

            diameter host-select table { 1 | 2 } algorithm { ip-address-modulus | msisd-
modulus | round-robin }

        end
```

Notes:

- <context_name> must be the name of the context where you want to enable IMS Authorization Service.
- <imsa_service_name> must be the name of the IMS Authorization Service to be configured for the Gx interface authentication.
- A maximum of 16 authorization services can be configured globally in a system. There is also a system limit for maximum number of total configured services.
- Secondary P-CSCF IP address can be configured in the P-CSCF table. Refer to the *Command Line Interface Reference* for more information on the **p-cscf table** command.
- To enable Rel. 6 Gx interface support, specific Diameter dictionary must be configured. For information on the Diameter dictionary to use, contact your Cisco account representative.
- *Optional:* To configure the quality of service (QoS) update timeout for a subscriber, in the IMS Authorization Service Configuration Mode, enter the following command:

```
qos-update-timeout <timeout_duration>
```



Important: This command is obsolete in release 11.0 and later releases.

- *Optional:* To configure signalling restrictions, in the IMS Authorization Service Configuration Mode, enter the following commands:

```
signaling-flag { deny | permit }

signaling-flow permit server-address <ip_address> [ server-port { <port_number> |
range <start_number> to <end_number> } ] [ description <string> ]
```
- *Optional:* To configure action on packets that do not match any policy gates in the general purpose PDP context, in the IMS Authorization Service Configuration Mode, enter the following command:

```
traffic-policy general-pdp-context no-matching-gates direction { downlink | uplink
} { forward | discard }
```
- *Optional:* To configure the algorithm to select Diameter host table, in the Policy Control Configuration Mode, enter the following command:

```
diameter host-select table { 1 | 2 } algorithm { ip-address-modulus | msisd-
modulus | round-robin }
```

Verifying IMS Authorization Service Configuration

To verify the IMS Authorization Service configuration:

- Step 1** Change to the context where you enabled IMS Authorization Service by entering the following command:

```
context <context_name>
```

- Step 2** Verify the IMS Authorization Service's configurations by entering the following command:

```
show ims-authorization service name <imsa_service_name>
```

Applying IMS Authorization Service to an APN

After configuring IMS Authorization service at the context-level, an APN must be configured to use the IMS Authorization service for an IMS subscriber.

Use the following example to apply IMS Authorization service functionality to a previously configured APN within the context configured in the [Configuring IMS Authorization Service at Context Level](#) section.

configure

```
context <context_name>

apn <apn_name>

ims-auth-service <imsa_service_name>

end
```

Notes:

- <context_name> must be the name of the context in which the IMS Authorization service was configured.

- `<imsa_service_name>` must be the name of the IMS Authorization Service configured for IMS authentication in the context.

Verifying Subscriber Configuration

Verify the IMS Authorization Service configuration for subscriber(s) by entering the following command:

```
show subscribers ims-auth-service <imsa_service_name>
```

`<imsa_service_name>` must be the name of the IMS Authorization Service configured for IMS authentication.

Rel. 7 Gx Interface

Rel. 7 Gx interface support is available on the Cisco ASR chassis running StarOS 8.1 or StarOS 9.0 and later releases for the following products:

- GGSN
- IPSG

This section describes the following topics:

- [Introduction](#)
- [Terminology and Definitions](#)
- [How it Works](#)
- [Configuring Rel. 7 Gx Interface](#)
- [Gathering Statistics](#)

Introduction

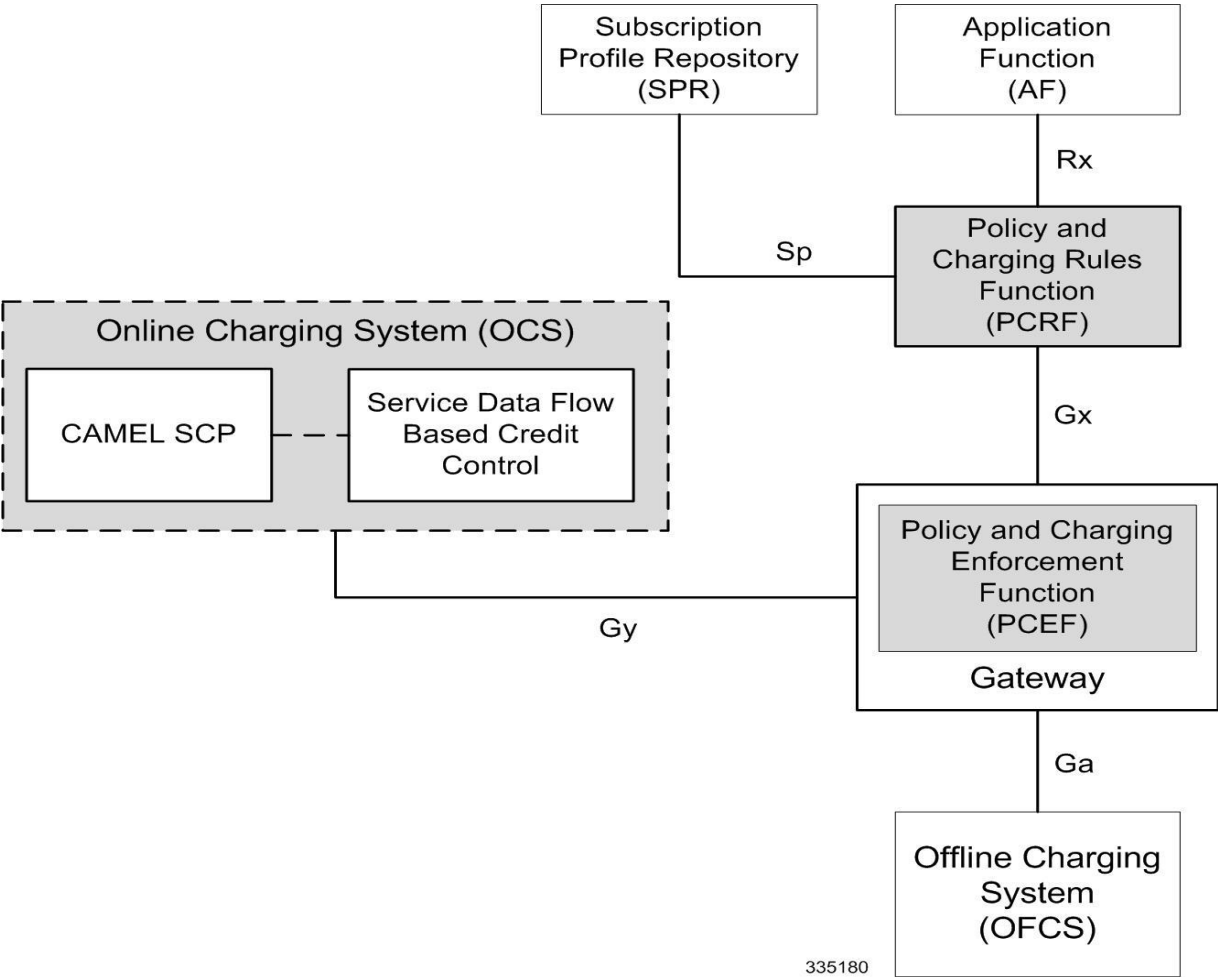
For IMS deployment in GPRS/UMTS networks the system uses Rel. 7 Gx interface for policy-based admission control support and flow-based charging. The Rel. 7 Gx interface supports enforcing policy control features like gating, bandwidth limiting, and so on, and also supports flow-based charging. This is accomplished via dynamically provisioned Policy Control and Charging (PCC) rules. These PCC rules are used to identify Service Data Flows (SDF) and do charging. Other parameters associated with the rules are used to enforce policy control.

The PCC architecture allows operators to perform service-based QoS policy, and flow-based charging control. In the PCC architecture, this is accomplished mainly by the Policy and Charging Enforcement Function (PCEF)/Cisco Systems GGSN and the Policy and Charging Rules Function (PCRF).

In GPRS/UMTS networks, the client functionality lies with the GGSN, therefore in the IMS authorization scenario it is also called the Gateway. In the following figure, Gateway is the Cisco Systems GGSN, and the PCEF function is provided by Enhanced Charging Service (ECS). The Rel 7. Gx interface is implemented as a Diameter connection. The Gx messages mostly involve installing/modifying/removing dynamic rules and activating/deactivating predefined rules.

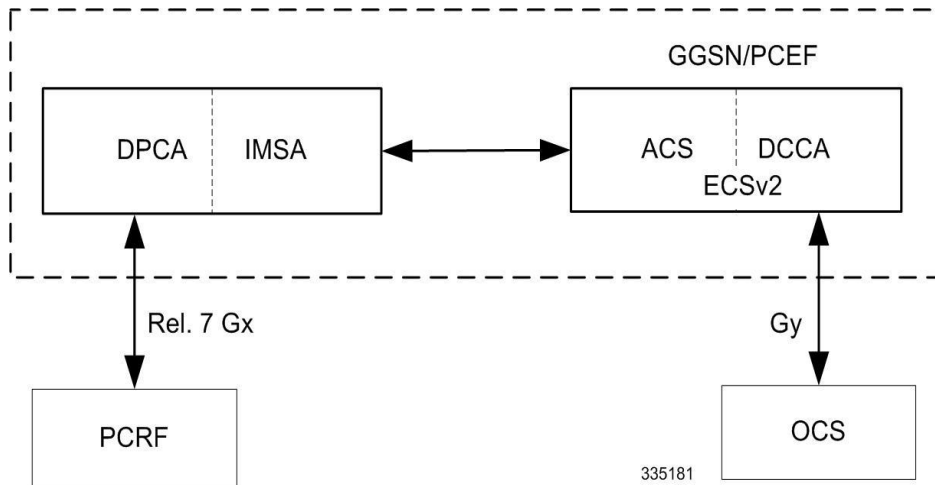
The Rel. 7 Gx reference point is located between the Gateway and the PCRF. This reference point is used for provisioning and removal of PCC rules from the PCRF to the Gateway, and the transmission of traffic plane events from the Gateway to the PCRF. The Gx reference point can be used for charging control, policy control, or both by applying AVPs relevant to the application. The following figure shows the reference points between various elements involved in the policy and charging architecture.

Figure 26. PCC Logical Architecture



Within the Gateway, the IMSA and DPCA modules handle the Gx protocol related functions (at the SessMgr) and the policy enforcement and charging happens at ECS. The Gy protocol related functions are handled within the DCCA module (at the ECS). The following figure shows the interaction between components within the Gateway.

Figure 27. PCC Architecture within Cisco PCEF



Supported Networks and Platforms

This feature is supported on all chassis with StarOS Release 8.1 and later running GGSN service for the core network services.

License Requirements

The Rel. 7 Gx interface support is a licensed Cisco feature. A separate feature license may be required. Contact your Cisco account representative for detailed information on specific licensing requirements. For information on installing and verifying licenses, refer to the *Managing License Keys* section of the *Software Management Operations* chapter in the *System Administration Guide*.

Supported Standards

The Rel 7. Gx interface support is based on the following standards and RFCs:

- 3GPP TS 23.203 V7.6.0 (2008-03): 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Policy and charging control architecture (Release 7)
- 3GPP TS 29.212 V7.8.0 (2009-03): 3rd Generation Partnership Project; Technical Specification Group Core Network and Terminals; Policy and Charging Control over Gx reference point (Release 7)
- 3GPP TS 29.213 V7.4.0 (2008-03): 3rd Generation Partnership Project; Technical Specification Group Core Network and Terminals; Policy and Charging Control signalling flows and QoS parameter mapping; (Release 7)
- RFC 3588, Diameter Base Protocol; September 2003
- RFC 4006, Diameter Credit-Control Application; August 2005

Terminology and Definitions

This section describes features and terminology pertaining to Rel. 7 Gx functionality.

Policy Control

The process whereby the PCRF indicates to the PCEF how to control the IP-CAN bearer.

Policy control comprises the following functions:

- **Binding:** Binding is the generation of an association between a Service Data Flow (SDF) and the IP CAN bearer (for GPRS a PDP context) transporting that SDF.

The QoS demand in the PCC rule, as well as the SDF template are input for the bearer binding. The selected bearer will have the same QoS Class as the one indicated by the PCC rule.

Depending on the type of IP-CAN and bearer control mode, bearer binding can be executed either by the PCRF, or both PCRF and PCEF.

- For UE-only IP-CAN bearer establishment mode, the PCRF performs bearer binding. When the PCRF performs bearer binding, it indicates the bearer (PDP context) by means of Bearer ID. The Bearer ID uniquely identifies the bearer within the PDP session.
- For UE/NW IP-CAN bearer establishment mode, the PCRF performs the binding of the PCC rules for user controlled services, while the PCEF performs the binding of the PCC rules for the network-controlled services.
- **Gating Control:** Gating control is the blocking or allowing of packets, belonging to an SDF, to pass through to the desired endpoint. A gate is described within a PCC rule and gating control is applied on a per SDF basis. The commands to open or close the gate leads to the enabling or disabling of the passage for corresponding IP packets. If the gate is closed, all packets of the related IP flows are dropped. If the gate is opened, the packets of the related IP flows are allowed to be forwarded.
- **Event Reporting:** Event reporting is the notification of and reaction to application events to trigger new behavior in the user plane as well as the reporting of events related to the resources in the Gateway (PCEF).
 - Event triggers may be used to determine which IP-CAN session modification or specific event causes the PCEF to re-request PCC rules. Although event trigger reporting from PCEF to PCRF can apply for an IP CAN session or bearer depending on the particular event, provisioning of event triggers will be done at session level.

Note that in 11.0 and later releases, RAR with unknown event triggers are silently ignored and responded with DIAMETER_SUCCESS. In earlier releases, when unknown event triggers were received in the RAR command from PCRF, invalid AVP result code was set in the RAA command.

- The Event Reporting Function (ERF) receives event triggers from PCRF during the Provision of PCC Rules procedure and performs event trigger detection. When an event matching the received event trigger occurs, the ERF reports the occurred event to the PCRF. If the provided event triggers are associated with certain parameter values then the ERF includes those values in the response back to the PCRF. The Event Reporting Function is located in the PCEF.

In StarOS releases prior to 14.0, SUCCESSFUL_RESOURCE_ALLOCATION (22) event trigger was sent for rules irrespective of successful installation. In 14.0 and later releases, SUCCESSFUL_RESOURCE_ALLOCATION (22) event trigger will be sent under the following conditions:

- When a rule is installed successfully (and the event trigger is armed by PCRF and resource-allocation-notification is enabled).

- On partial failure, i.e., when two or more rules are installed and at least one of the rules were successfully installed. (and the event trigger is armed by PCRF and resource-allocation-notification is enabled).

On complete failure, i.e., none of the rules were installed, the event-trigger SUCCESSFUL_RESOURCE_ALLOCATION (22) will not be sent.



Important: In this release, event triggers “IP-CAN_CHANGE” and “MAX_NR_BEARERS_REACHED” are not supported.

- **QoS Control:** QoS control is the authorization and enforcement of the maximum QoS that is authorized for a SDF or an IP-CAN bearer or a QoS Class Identifier (QCI). In case of an aggregation of multiple SDFs (for GPRS a PDP context), the combination of the authorized QoS information of the individual SDFs is provided as the authorized QoS for this aggregate.
 - QoS control per SDF allows the PCC architecture to provide the PCEF with the authorized QoS to be enforced for each specific SDF.
 - The enforcement of the authorized QoS of the IP-CAN bearer may lead to a downgrading or upgrading of the requested bearer QoS by the Gateway (PCEF) as part of a UE-initiated IP-CAN bearer establishment or modification. Alternatively, the enforcement of the authorized QoS may, depending on operator policy and network capabilities, lead to network-initiated IP-CAN bearer establishment or modification. If the PCRF provides authorized QoS for both, the IP-CAN bearer and PCC rule(s), the enforcement of authorized QoS of the individual PCC rules takes place first.
 - QoS authorization information may be dynamically provisioned by the PCRF, or it can be a predefined PCC rule in the PCEF. In case the PCRF provides PCC rules dynamically, authorized QoS information for the IP-CAN bearer (combined QoS) may be provided. For a predefined PCC rule within the PCEF, the authorized QoS information takes affect when the PCC rule is activated. The PCEF combines the different sets of authorized QoS information, that is the information received from the PCRF and the information corresponding to the predefined PCC rules. The PCRF knows the authorized QoS information of the predefined PCC rules and takes this information into account when activating them. This ensures that the combined authorized QoS of a set of PCC rules that are activated by the PCRF is within the limitations given by the subscription and operator policies regardless of whether these PCC rules are dynamically provided, predefined, or both.



Important: In this release, QoS Resource Reservation is not supported.

Supported Features:

- Provisioning and Policy Enforcement of Authorized QoS: The PCRF may provide authorized QoS to the PCEF. The authorized QoS provides appropriate values for resources to be enforced.
- Provisioning of “Authorized QoS” Per IP CAN Bearer: The authorized QoS per IP-CAN bearer is used if the bearer binding is performed by the PCRF.
- Policy Enforcement for “Authorized QoS” per IP CAN Bearer: The PCEF is responsible for enforcing the policy-based authorization, that is to ensure that the requested QoS is in-line with the “Authorized QoS” per IP CAN Bearer.
- Policy Provisioning for Authorized QoS Per SDF: The provisioning of authorized QoS per SDF is a part of PCC rule provisioning procedure.

- Policy Enforcement for Authorized QoS Per SDF: If an authorized QoS is defined for a PCC rule, the PCEF limits the data rate of the SDF corresponding to that PCC rule not to exceed the maximum authorized bandwidth for the PCC rule by discarding packets exceeding the limit.
- Upon deactivation or removal of a PCC rule, the PCEF frees the resources reserved for that PCC rule. If the PCRF provides authorized QoS for both the IP-CAN bearer and PCC rule(s), the enforcement of authorized QoS of the individual PCC rules takes place first.



Important: In this release, coordination of authorized QoS scopes in mixed mode (BCM = UE_NW) is not supported.

- Provisioning of Authorized QoS Per QCI: If the PCEF performs the bearer binding, the PCRF may provision an authorized QoS per QCI for non-GBR bearer QCI values. If the PCRF performs the bearer binding the PCRF does not provision an authorized QoS per QCI. The PCRF does not provision an authorized QoS per QCI for GBR bearer QCI values.
- Policy Enforcement for Authorized QoS per QCI: The PCEF can receive an authorized QoS per QCI for non GBR-bearer QCI values.
- Other Features:
 - Bearer Control Mode Selection: The PCEF may indicate, via the Gx reference point, a request for Bearer Control Mode (BCM) selection at IP-CAN session establishment or IP-CAN session modification (as a consequence of an SGSN change). It will be done using the “PCC Rule Request” procedure.

If the Bearer-Control-Mode AVP is not received from PCRF, the IP-CAN session is not terminated. The value negotiated between UE/SGSN/GGSN is considered as the BCM. The following values are considered for each of the service types:

- GGSN: The negotiated value between UE/SGSN/GGSN is considered.

In the following scenarios UE_ONLY is chosen as the BCM:

Scenario 1:

- UE-> UE_ONLY
- SGSN-> UE_ONLY
- GGSN-> UE_ONLY
- PCRF-> NO BCM

Scenario 2:

- UE-> UE_ONLY
- SGSN-> UE_ONLY
- GGSN-> Mixed
- PCRF-> NO BCM
- GTP-PGW: BCM of UE_NW is considered.
- IPSG: BCM of UE_ONLY is considered.
- HSGW/SGW/PDIF/FA/PDSN/HA/MIPv6HA: BCM of NONE is considered.

- PCC Rule Error Handling: If the installation/activation of one or more PCC rules fails, the PCEF includes one or more Charging-Rule-Report AVP(s) in either a CCR or an RAA command for the affected PCC rules. Within each Charging-Rule-Report AVP, the PCEF identifies the failed PCC

rule(s) by including the Charging-Rule-Name AVP(s) or Charging-Rule-Base-Name AVP(s), identifies the failed reason code by including a Rule-Failure-Code AVP, and includes the PCC-Rule-Status AVP.

If the installation/activation of one or more new PCC rules (that is, rules that were not previously successfully installed) fails, the PCEF sets the PCC-Rule-Status to INACTIVE for both the PUSH and the PULL modes.

If a PCC rule was successfully installed/activated, but can no longer be enforced by the PCEF, the PCEF shall send the PCRF a new CCR command and include a Charging-Rule-Report AVP. The PCEF shall include the Rule-Failure-Code AVP within the Charging-Rule-Report AVP and shall set the PCC-Rule-Status to INACTIVE.

- Time of the Day Procedures: PCEF performs PCC rule request as instructed by the PCRF. Revalidation-Time when set by the PCRF, causes the PCEF to trigger a PCRF interaction to request PCC rules from the PCRF for an established IP CAN session. The PCEF stops the timer once the PCEF triggers a REVALIDATION_TIMEOUT event.



Important: In 11.0 and later releases, Rule-Activation-Time / Rule-Deactivation-Time / Revalidation-Time AVP is successfully parsed only if its value corresponds to current time or a later time than the current IPSPG time, else the AVP and entire message is rejected. In earlier releases the AVP is successfully parsed only if its value corresponds to a later time than the current IPSPG time, else the AVP and entire message is rejected.

Charging Control

Charging Control is the process of associating packets belonging to a SDF to a charging key, and applying online charging and/or offline charging, as appropriate. Flow-based charging handles differentiated charging of the bearer usage based on real time analysis of the SDFs. In order to allow for charging control, the information in the PCC rule identifies the SDF and specifies the parameters for charging control. The PCC rule information may depend on subscription data.

In the case of online charging, it is possible to apply an online charging action upon PCEF events (for example, re-authorization upon QoS change).

It is possible to indicate to the PCEF that interactions with the charging systems are not required for a PCC rule, that is to perform neither accounting nor credit control for this SDF, and then no offline charging information is generated.

Supported Features:

- Provisioning of Charging-related Information for the IP-CAN Session.
- Provisioning of Charging Addresses: Primary or secondary event charging function name (Online Charging Server (OCS) addresses or the peer names).



Important: In this release, provisioning of primary or secondary charging collection function name (Offline Charging Server (OFCS) addresses) over Gx is not supported.

- Provisioning of Default Charging Method: In this release, the default charging method is sent in CCR-I message. For this, new AVPs Online/Offline are sent in CCR-I message based on the configuration.

Charging Correlation

For the purpose of charging correlation between SDF level and application level (for example, IMS) as well as on-line charging support at the application level, applicable charging identifiers and IP-CAN type identifiers are passed from the PCRF to the AF, if such identifiers are available.

For IMS bearer charging, the IP Multimedia Core Network (IM CN) subsystem and the Packet Switched (PS) domain entities are required to generate correlated charging data.

In order to achieve this, the Gateway provides the GGSN Charging Identifier (GCID) associated with the PDP context along with its address to the PCRF. The PCRF in turn sends the IMS Charging Identifier (ICID), which is provided by the P-CSCF, to the Gateway. The Gateway generates the charging records including the GCID as well as the ICID if received from PCRF, so that the correlation of charging data can be done with the billing system.

PCRF also provides the flow identifier, which uniquely identifies an IP flow in an IMS session.

Policy and Charging Control (PCC) Rules

A PCC rule enables the detection of an SDF and provides parameters for policy control and/or charging control. The purpose of the PCC rule is to:

- Detect a packet belonging to an SDF.
 - Select downlink IP CAN bearers based on SDF filters in the PCC rule.
 - Enforce uplink IP flows are transported in the correct IP CAN bearer using the SDF filters within the PCC rule.
- Identify the service that the SDF contributes to.
- Provide applicable charging parameters for an SDF.
- Provide policy control for an SDF.

The PCEF selects a PCC rule for each packet received by evaluating received packets against SDF filters of PCC rules in the order of precedence of the PCC rules. When a packet matches a SDF filter, the packet matching process for that packet is completed, and the PCC rule for that filter is applied.

There are two types of PCC rules:

- **Dynamic PCC Rules:** Rules dynamically provisioned by the PCRF to the PCEF via the Gx interface. These PCC rules may be either predefined or dynamically generated in the PCRF. Dynamic PCC rules can be installed, modified, and removed at any time.
- **Predefined PCC Rule:** Rules preconfigured in the PCEF by the operators. Predefined PCC rules can be activated or deactivated by the PCRF at any time. Predefined PCC rules within the PCEF may be grouped allowing the PCRF to dynamically activate a set of PCC rules over the Gx reference point.



Important: A third type of rule, the static PCC rule can be preconfigured in the chassis by the operators. Static PCC rules are not explicitly known in the PCRF, and are not under control of the PCRF. Static PCC rules are bound to general purpose bearer with no Gx control.

A PCC rule consists of:

- **Rule Name:** The rule name is used to reference a PCC rule in the communication between the PCEF and PCRF.
- **Service Identifier:** The service identifier is used to identify the service or the service component the SDF relates to.
- **Service Data Flow Filter(s):** The service flow filter(s) is used to select the traffic for which the rule applies.

- **Precedence:** For different PCC rules with overlapping SDF filter, the precedence of the rule determines which of these rules is applicable. When a dynamic PCC rule and a predefined PCC rule have the same priority, the dynamic PCC rule takes precedence.
- **Gate Status:** The gate status indicates whether the SDF, detected by the SDF filter(s), may pass (gate is open) or will be discarded (gate is closed) in uplink and/or in downlink direction.
- **QoS Parameters:** The QoS information includes the QoS class identifier (authorized QoS class for the SDF), the Allocation and Retention Priority (ARP), and authorized bitrates for uplink and downlink.



Important: In earlier releases, ECS used only the Priority-Level part of ARP byte for bearer binding, (along with QCI). Now the entire ARP byte is used for bearer binding (along with QCI). Since the capability and vulnerability bits are optional in a dynamic rule, if a dynamic rule is received without these flags, it is assumed that the capability bit is set to 1 (disabled) and vulnerability bit is set to 0 (enabled). For predefined rules, currently configuring these two flags is not supported, so as of now all predefined rules are assumed to have capability bit set to 1 (disabled) and vulnerability bit set to 0 (enabled).

- **Charging key (rating group)**
- **Other charging parameters:** The charging parameters define whether online and offline charging interfaces are used, what is to be metered in offline charging, on what level the PCEF will report the usage related to the rule, and so on.



Important: In this release, configuring the Metering Method and Reporting Level for dynamic PCC rules is not supported.

PCC rules also include Application Function (AF) record information for enabling charging correlation between the application and bearer layer if the AF has provided this information via the Rx interface. For IMS, this includes the IMS Charging Identifier (ICID) and flow identifiers.

PCC Procedures over Gx Reference Point

Request for PCC rules

The PCEF, via the Gx reference point, requests for PCC rules in the following instances:

- At IP-CAN session establishment.
- At IP-CAN session modification.

PCC rules can also be requested as a consequence of a failure in the PCC rule installation/activation or enforcement without requiring an event trigger.

Provisioning of PCC rules


The PCRF indicates, via the Rel. 7 Gx reference point, the PCC rules to be applied at the PCEF. This may be using one of the following procedures:

- **PULL (provisioning solicited by the PCEF):** In response to a request for PCC rules being made by the PCEF, the PCRF provisions PCC rules in the CC-Answer.
- **PUSH (unsolicited provisioning):** The PCRF may decide to provision PCC rules without obtaining a request from the PCEF. For example, in response to information provided to the PCRF via the Rx reference point, or in response to an internal trigger within the PCRF. To provision PCC rules without a request from the PCEF, the

PCRF includes these PCC rules in an RA-Request message. No CCR/CCA messages are triggered by this RA-Request.


For each request from the PCEF or upon unsolicited provision the PCRF provisions zero or more PCC rules. The PCRF may perform an operation on a single PCC rule by one of the following means:


- To activate or deactivate a PCC rule that is predefined at the PCEF, the PCRF provisions a reference to this PCC rule within a Charging-Rule-Name AVP and indicates the required action by choosing either the Charging-Rule-Install AVP or the Charging-Rule-Remove AVP.
- To install or modify a PCRF-provisioned PCC rule, the PCRF provisions a corresponding Charging-Rule-Definition AVP within a Charging-Rule-Install AVP.
- To remove a PCC rule which has previously been provisioned by the PCRF, the PCRF provisions the name of this rule as value of a Charging-Rule-Name AVP within a Charging-Rule-Remove AVP.
- If the PCRF performs the bearer binding, the PCRF may move previously installed or activated PCC rules from one IP CAN bearer to another IP CAN bearer.

 **Important:** In 11.0 and later releases, the maximum valid length for a charging rule name is 63 bytes. When the length of the charging rule name is greater than 63 bytes, a charging rule report with RESOURCES_LIMITATION as Rule-Failure-Code is sent. This charging rule report is sent only when the length of the rule name is lesser than 128 characters. When the charging rule name length is greater than or equal to 128 characters no charging rule report will be sent. In earlier releases, the length of the charging rule name constructed by PCRF was limited to 32 bytes.

Selecting a PCC Rule for Uplink IP Packets

If PCC is enabled, the PCEF selects the applicable PCC rule for each received uplink IP packet within an IP CAN bearer by evaluating the packet against uplink SDF filters of PCRF-provided or predefined active PCC rules of this IP CAN bearer in the order of the precedence of the PCC rules.


 **Important:** When a PCRF-provided PCC rule and a predefined PCC rule have the same precedence, the uplink SDF filters of the PCRF-provided PCC rule is applied first.

 **Important:** In 11.0 and later releases, IMSA and ECS allow the PCRF to install two (or more) dynamic rules with the same precedence value. In earlier releases, for two distinct dynamic rules having the same precedence the second rule used to be rejected.

When a packet matches an SDF filter, the packet matching process for that packet is completed, and the PCC rule for that filter is applied. Uplink IP packets which do not match any PCC rule of the corresponding IP CAN bearer are discarded.

Selecting a PCC Rule and IP CAN Bearer for Downlink IP Packets

If PCC is enabled, the PCEF selects a PCC rule for each received downlink IP packet within an IP CAN session by evaluating the packet against downlink SDF filters of PCRF-provided or predefined active PCC rules of all IP CAN bearers of the IP CAN session in the order of the precedence of the PCC rules.

 **Important:** When a PCRF-provided PCC rule and a predefined PCC rule have the same precedence, the downlink SDF filters of the PCRF-provided PCC rule are applied first.

When a packet matches a SDF filter, the packet matching process for that packet is completed, and the PCC rule for that filter is applied. The Downlink IP Packet is transported within the IP CAN bearer where the selected PCC rule is mapped. Downlink IP packets that do not match any PCC rule of the IP CAN session are discarded.

The following procedures are also supported:

- Indication of IP-CAN Bearer Termination Implications
- Indication of IP-CAN Session Termination: When the IP-CAN session is being terminated (for example, for GPRS when the last PDP Context within the IP-CAN session is being terminated) the PCEF contacts the PCRF.
- Request of IP-CAN Bearer Termination: If the termination of the last IP CAN bearer within an IP CAN session is requested, the PCRF and PCEF apply the “Request of IP-CAN Session Termination” procedure.
- Request of IP-CAN Session Termination: If the PCRF decides to terminate an IP CAN session due to an internal trigger or trigger from the SPR, the PCRF informs the PCEF. The PCEF acknowledges to the PCRF and instantly removes/deactivates all the PCC rules that have been previously installed or activated on that IP-CAN session.

The PCEF applies IP CAN specific procedures to terminate the IP CAN session. For GPRS, the GGSN send a PDP context deactivation request with the teardown indicator set to indicate that the termination of the entire IP-CAN session is requested. Furthermore, the PCEF applies the “Indication of IP CAN Session Termination” procedure.


In 12.0 and later releases, volume or rule information obtained from PCRF is discarded if the subscriber is going down.

Volume Reporting Over Gx

This section describes the 3GPP Rel. 9 Volume Reporting over Gx feature, which is supported by all products supporting Rel. 7 Gx interface.

License Requirements

The Volume Reporting over Gx is a licensed Cisco feature. A separate feature license may be required. Contact your Cisco account representative for detailed information on specific licensing requirements. For information on installing and verifying licenses, refer to the *Managing License Keys* section of the *Software Management Operations* chapter in the *System Administration Guide*.

 **Important:** In 12.0 and later releases, no separate license is required for Charging over Gx / Volume Reporting over Gx feature. This feature can be enabled as part of "Policy Interface" license.


Supported Standards

The Volume Reporting over Gx feature is based on the following standard:

3GPP TS 29.212 V9.5.0 (2010-06): 3rd Generation Partnership Project; Technical Specification Group Core Network and Terminals; Policy and Charging Control over Gx reference point (Release 9).

Feature Overview

The Volume Reporting over Gx feature provides PCRF the capability to make real-time decisions based on the data usage by subscribers.

 **Important:** Volume Reporting over Gx is applicable only for volume quota.



Important: In release 10.0, only total data usage reporting is supported, uplink/downlink level reporting is not supported. In 10.2 and later releases, it is supported.



Important: The PCEF only reports the accumulated usage since the last report for usage monitoring and not from the beginning.



Important: If the usage threshold is set to zero (infinite threshold), no further threshold events will be generated by PCEF, but monitoring of usage will continue and be reported at the end of the session.



Important: In 12.2 and later releases, usage reporting on bearer termination is supported.

The following steps explain how Volume Reporting over Gx works:

1. PCEF after receiving the message from PCRF parses the usage monitoring related AVPs, and sends the information to IMSA.
2. IMSA updates the information to ECS.
3. Once the ECS is updated with the usage monitoring information from PCRF, the PCEF (ECS) starts tracking the data usage.
4. For session-level monitoring, the ECS maintains the amount of data usage.
5. For PCC rule monitoring, usage is monitored with the monitoring key as the unique identifier. Each node maintains the usage information per monitoring key. When the data traffic is passed, the usage is checked against the usage threshold values and reported as described in the *Usage Reporting* section.
6. The PCEF continues to track data usage after the threshold is reached and before a new threshold is provided by the PCRF. If a new usage threshold is not provided by the PCRF in the acknowledgement of an IP-CAN Session modification where its usage was reported, then usage monitoring does not continue in the PCEF for that IP CAN session.

Usage Monitoring

- Usage Monitoring at Session Level: PCRF subscribes to the session-level volume reporting over Gx by sending the Usage-Monitoring-Information AVP with the usage threshold level set in Granted-Service-Unit AVP and Usage-Monitoring-Level AVP set to SESSION_LEVEL(0). After the AVPs are parsed by DPCA, IMSA updates the information to ECS. Once ECS is updated usage monitoring is started and constantly checked with the usage threshold whenever the data traffic is present. In 11.0 and later releases, Monitoring Key at session level is supported.

In 12.0 and later releases, enabling and disabling session usage in a single message from PCRF is supported. This is supported only if the monitoring key is associated at session level.

In 12.0 and later releases, monitoring of usage based on input/output octet threshold levels is supported. Usage is reported based on the enabled threshold level. If multiple levels are enabled, usage will be reported on all the enabled levels even if only one of the levels is breached. Monitoring will be stopped on the missing threshold levels in the response for the usage report from PCRF (expected to provide the complete set again if PCRF wants to continue monitoring on the multiple levels enabled earlier).

Total threshold level along with UL/DL threshold level in the GSU AVP is treated as an error and only total threshold level is accepted.

- **Usage Monitoring at Flow Level:** PCRF subscribes to the flow-level volume reporting over Gx by sending the Usage-Monitoring-Information AVP with the usage threshold level set in Granted-Service-Unit AVP and Usage-Monitoring-Level AVP set to PCC_RULE_LEVEL(1). Monitoring Key is mandatory in case of a flow-level monitoring since the rules are associated with the monitoring key and enabling/disabling of usage monitoring at flow level can be controlled by PCRF using it. After the AVPs are parsed by DPCA, IMSA updates the information to ECS. Once ECS is updated usage monitoring is started and constantly checked with the usage threshold whenever the data traffic is present.

Usage monitoring is supported for static, predefined rules, and dynamic rule definitions.

- **Usage Monitoring for Static Rules:** In the case of static rules, the usage reporting on last rule removal associated with the monitoring key is not applicable. In this case only the usage monitoring information is received from the PCRF.
- **Usage Monitoring for Predefined Rules:** If the usage monitoring needs to be enabled for the predefined rules, PCRF sends the rule and the usage monitoring information containing the monitoring key and the usage threshold. The Monitoring key should be same as the one pre-configured in PCEF for that predefined rule. There can be multiple rules associated with the same monitoring key. Hence enabling a particular monitoring key would result in the data being tracked for multiple rules having the same monitoring key. After DPCA parses the AVPs IMSA updates the information to ECS. Once ECS is updated usage monitoring is started and constantly checked with the usage threshold whenever the data traffic is present.
- **Usage Monitoring for Dynamic Rules:** If the usage monitoring needs to be enabled for dynamic ruledefs, PCRF provides the monitoring key along with a charging rule definition and the usage monitoring information containing the monitoring key and the usage threshold. This would result in the usage monitoring being done for all the rules associated with that monitoring key. After DPCA parses the AVPs, IMSA updates the information to ECS. Once ECS is updated, the usage monitoring is started and constantly checked with the usage threshold whenever the data traffic is present. Monitoring key for dynamic ruledef is dynamically assigned by PCRF which is the only difference with predefined rules in case of usage monitoring.

Usage Reporting

Usage at subscriber/flow level is reported to PCRF under the following conditions:

- **Usage Threshold Reached:** PCEF records the subscriber data usage and checks if the usage threshold provided by PCRF is reached. This is done for both session and rule level reporting.

For session-level reporting, the actual usage volume is compared with the usage volume threshold.

For rule-level reporting the rule that hits the data traffic is used to find out if the monitoring key is associated with it, and based on the monitoring key the data usage is checked. Once the condition is met, it reports the usage information to IMSA and continues monitoring. IMSA then triggers the CCR-U if “USAGE_REPORT” trigger is enabled by the PCRF. The Usage-Monitoring-Information AVP is sent in this CCR with the “Used-Service-Unit” set to the amount of data usage by subscriber.

If PCRF does not provide a new usage threshold in the usage monitoring information as a result of CCR from PCEF when the usage threshold is reached, the usage monitoring is stopped at PCEF and no usage status is reported.

In the non-standard Volume Reporting over Gx implementation, usage monitoring will be stopped once the threshold is breached, else the monitoring will continue. There will be no further usage reporting until the CCA is received.

- **Usage Monitoring Disabled:** If the PCRF explicitly disables the usage monitoring with Usage-Monitoring-Support AVP set to USAGE_MONITORING_DISABLED, the PCEF stops monitoring and reports the usage information (when the monitoring was enabled) to PCRF if the usage monitoring is disabled by PCRF as a

result of CCR from PCEF which is not related to reporting usage, other external triggers, or a PCRF internal trigger. If the PCRF does not provide a new usage threshold as a result of CCR from PCEF when the usage threshold is reached, the usage monitoring is stopped at PCEF and no further usage status is reported.

- **IP CAN Session Termination:** When the IP CAN session is terminated, the accumulated subscriber usage information is reported to PCRF in the CCR-T from PCEF. If PCC usage level information is enabled by PCRF, the PCC usage will also be reported.
- **PCC Rule Removal:** When the PCRF deactivates the last PCC rule associated with a usage monitoring key, the PCEF sends a CCR with the data usage for that monitoring key. If the PCEF reports the last PCC rule associated with a usage monitoring key is inactive, the PCEF reports the accumulated usage for that monitoring key within the same CCR command if the Charging-Rule-Report AVP was included in a CCR command; otherwise, if the Charging-Rule-Report AVP was included in an RAA command, the PCEF sends a new CCR command to report accumulated usage for the usage monitoring key. In 12.0 and later releases, usage reporting on last rule deactivation using rule deactivation time set by PCRF is supported.
- **PCRF Requested Usage Report:** In 10.2 and later releases, the accumulated usage since the last report is sent even in case of immediate reporting, the usage is reset after immediate reporting and usage monitoring continued so that the subsequent usage report will have the usage since the current report. In earlier releases the behavior was to accumulate the so far usage in the next report.
- **Release 12.2 onwards,** usage reporting on bearer termination can be added. When a bearer is deleted due to some reason, the rules associated with the bearer will also be removed. So, the usage will be reported on the monitoring key(s) whose associated rule is the last one that is removed because of bearer termination.
- **Revalidation Timeout:** In the non-standard implementation, if usage monitoring and reporting is enabled and a revalidation timeout occurs, the PCEF sends a CCR to request PCC rules and reports all accumulated usage for all enabled monitoring keys since the last report (or since usage reporting was enabled if the usage was not yet reported) with the accumulated usage at IP-CAN session level (if enabled) and at service data flow level (if enabled) This is the default behavior.

In the case of standard implementation, this must be enabled by CLI configuration.



Important: The Usage Reporting on Revalidation Timeout feature is available by default in non-standard implementation of Volume Reporting over Gx. In 10.2 and later releases, this is configurable in the standard implementation. This is not supported in 10.0 release for standard based volume reporting.

Once the usage is reported, the usage counter is reset to zero. The PCEF continues to track data usage from the zero value after the threshold is reached and before a new threshold is provided by the PCRF. If a new usage threshold is not provided by the PCRF in the acknowledgement of an IP-CAN Session modification where its usage was reported, then usage monitoring does not continue in the PCEF for that IP CAN session and the usage accumulated between the CCR-CCA will be discarded.

For information on how to configure the Volume Reporting over Gx feature, see the [Configuring Volume Reporting over Gx](#) section.

How Rel. 7 Gx Works

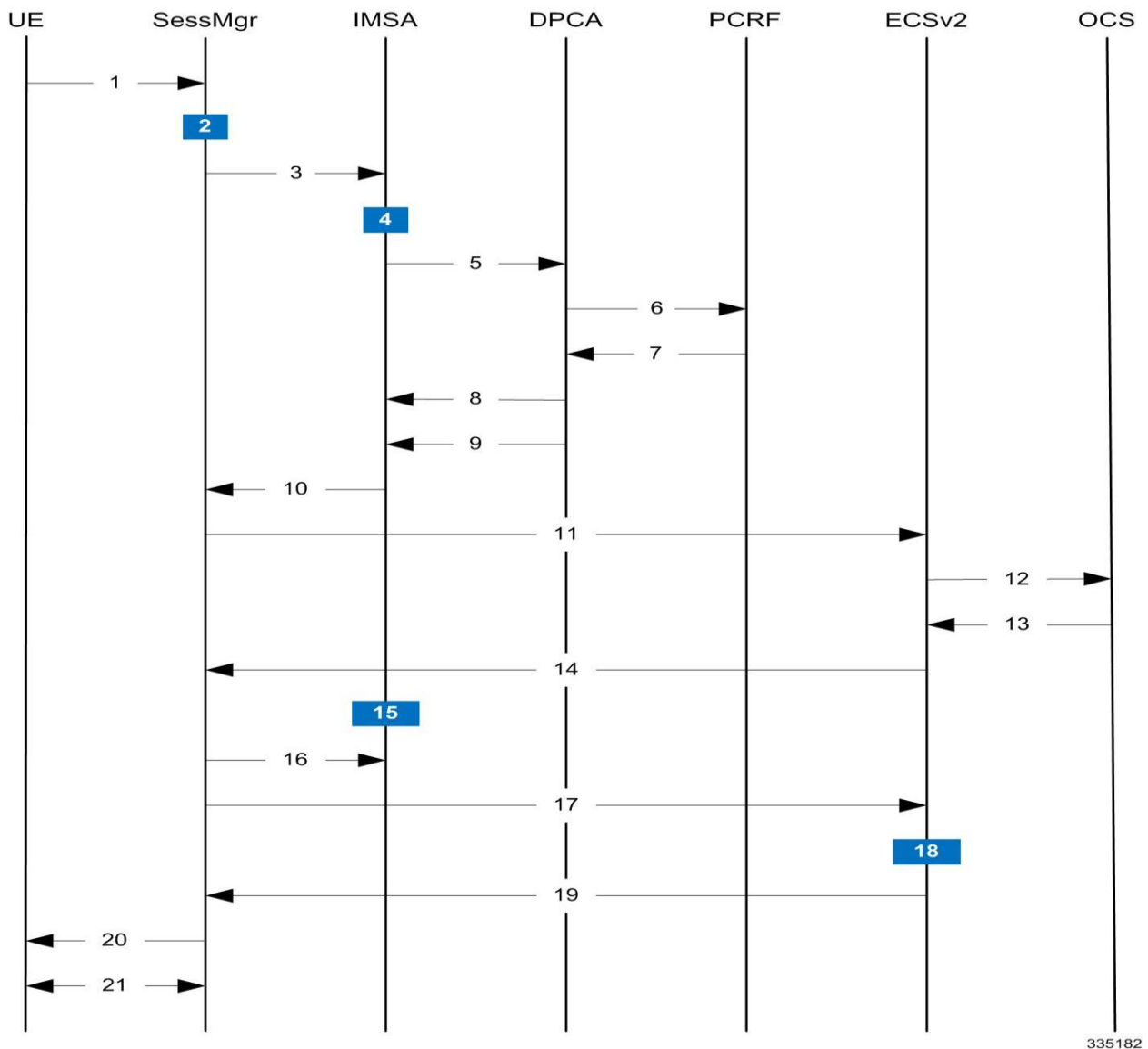
This section describes how dynamic policy and charging control for subscribers works with Rel. 7 Gx interface support in GPRS/UMTS networks.

The following figure and table explain the IMSA process between a system and IMS components that is initiated by the UE.

In this example, the Diameter Policy Control Application (DPCA) is the Gx interface to the PCRF. The interface between IMSA with PCRF is the Gx interface, and the interface between Session Manager (SessMgr) and Online Charging Service (OCS) is the Gy interface. Note that the IMSA service and DPCA are part of SessMgr on the system and separated in the figure for illustration purpose only.

Important: In 14.0 and later releases, the DPCA and the IMSA will be acting as one module within the Policy Server interface application.

Figure 28. Rel. 7 Gx IMS Authorization Call Flow



335182

Table 32. Rel. 7 Gx IMS Authorization Call flow Description

Step	Description
1	UE (IMS subscriber) requests for primary PDP context activation/creation.
2	SessMgr allocates an IP address to the UE.
3	SessMgr requests IMS Authorization, if IMSA is enabled for the APN.
4	IMSA allocates resources for the IP CAN session and the bearer, and selects the PCRF to contact based on the user's selection key (for example, msisdn).
5	IMSA requests the DPCA module to issue an auth request to the PCRF.
6	DPCA sends a CCR initial message to the selected PCRF. This message includes the Context-Type AVP set to PRIMARY and the IP address allocated to the UE. The message may include the Bearer-Usage AVP set to GENERAL. The Bearer-Operation is set to Establishment. The Bearer ID is included if the PCRF does the bearer binding.
7	PCRF may send preconfigured charging rules in CCA, if a preconfigured rule set for general purpose PDP context is provided in PCRF. The dynamic rules and the authorized QoS parameters could also be included by the PCRF.
8	DPCA passes the charging rule definition, charging rule install, QoS information received from the PCRF, event triggers, and so on, along with the Bearer ID that corresponds to the rules received from the PCRF to IMSA. IMSA stores the information. If the Bearer ID is absent, and PCRF does the bearer binding, the rule is skipped. Whereas, if the Bearer ID is absent and the PCEF does the bearer binding, the rule is passed onto the ECS to perform bearer binding.
9	DPCA calls the callback function registered with it by IMSA.
10	IMSA stores the bearer authorized QoS information and notifies the SessMgr. Other PCRF provided information common to the entire PDP session (event trigger, primary/secondary OCS address, and so on) is stored within the IMSA. After processing the information, IMSA notifies the SessMgr about the policy authorization complete.
11	If the validation of the rules fails in IMSA/DPCA, a failure is notified to PCRF containing the Charging-Rule-Report AVP. Else, IMSA initiates creation of ECS session. The APN name, primary/secondary OCS server address, and so on are sent to the ECS from the SessMgr.
12	ECS performs credit authorization by sending CCR(I) to OCS with CC-Request-Type set to INITIAL_REQUEST to open the credit control session. This request includes the active Rulebase-Id (default rulebase ID from the APN/AAA) and GPRS specific attributes (for example, APN, UMTS QoS, and so on).
13	OCS returns a CCA initial message that may activate a statically configured Rulebase and may include preemptive quotas.
14	ECS responds to SessMgr with the response message.
15	SessMgr requests IMSA for the dynamic rules.
16	IMSA sends the dynamic rules to SessMgr. Note that, in 14.0 and later releases, the RAR messages are allowed before the session is established. In earlier releases, until the primary PDP context is established, all RAR messages from the PCRF were rejected. Also note that, in 14.0 and later releases, the RAR message is rejected and RAA is sent with 3002 result code when the recovery of dynamic rule information and audit of Session Manager are in progress. Earlier, the RAR messages were processed by DPCA even when the recovery audit was in progress.
17	SessMgr sends the dynamic rule information to the ECS. The gate flow status information and the QoS per flow (charging rule) information are also sent in the message.

Step	Description
18	ECS activates the predefined rules received, and installs the dynamic rules received. Also, the gate flow status and the QoS parameters are updated by ECS as per the dynamic charging rules. The Gx rulebase is treated as an ECS group-of-ruledefs. The response message contains the Charging Rule Report conveying the status of the rule provisioning at the ECS. ECS performs PCEF bearer binding for rules without bearer ID.
19	If the provisioning of rules fails partially, the context setup is accepted, and a new CCR-U is sent to the PCRF with the Charging-Rule-Report containing the PCC rule status for the failed rules. If the provisioning of rules fails completely, the context setup is rejected.
20	Depending on the response for the PDP Context Authorization, SessMgr sends the response to the UE and activates/rejects the call. If the Charging-Rule-Report contains partial failure for any of the rules, the PCRF is notified, and the call is activated. If the Charging-Rule-Report contains complete failure, the call is rejected.
21	Based on the PCEF bearer binding for the PCC rules at Step 18, the outcome could be one or more network-initiated PDP context procedures with the UE (Network Requested Update PDP Context (NRUPC) / Network Requested Secondary PDP Context Activation (NRSPCA)).

Configuring Rel. 7 Gx Interface

To configure Rel. 7 Gx interface functionality, the IMS Authorization service must be configured at the context level, and then the APN configured to use the IMS Authorization service.

To configure Rel. 7 Gx interface functionality:

- Step 1** Configure IMS Authorization service at the context level for IMS subscriber in GPRS/UMTS network as described in the [Configuring IMS Authorization Service at Context Level](#) section.
- Step 2** Verify your configuration as described in the [Verifying the Configuration](#) section.
- Step 3** Configure an APN within the same context to use the IMS Authorization service for IMS subscriber as described in the [Applying IMS Authorization Service to an APN](#) section.
- Step 4** Verify your configuration as described in the [Verifying Subscriber Configuration](#) section.
- Step 5** *Optional:* Configure the Volume Reporting over Gx feature as described in the [Configuring Volume Reporting over Gx](#) section.
- Step 6** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.



Important: Commands used in the configuration examples in this section provide base functionality to the extent that the most common or likely commands and/or keyword options are presented. In many cases, other optional commands and/or keyword options are available. Refer to the *Command Line Interface Reference* for complete information regarding all commands.

Configuring IMS Authorization Service at Context Level

Use the following example to configure IMS Authorization service at context level for IMS subscribers in GPRS/UMTS networks:

configure

```

context <context_name>

    ims-auth-service <imsa_service_name>

        p-cscf discovery table { 1 | 2 } algorithm { ip-address-modulus | msisdn-modulus
| round-robin }

        p-cscf table { 1 | 2 } row-precedence <precedence_value> { address <ip_address>
| ipv6-address <ipv6_address> } [ secondary { address <ip_address> | ipv6-address
<ipv6_address> } ]

        policy-control

            diameter origin endpoint <endpoint_name>

            diameter dictionary <dictionary>

            diameter request-timeout <timeout_duration>

            diameter host-select table { { { 1 | 2 } algorithm { ip-address-modulus |
msisdn-modulus | round-robin } } | prefix-table { 1 | 2 } }

            diameter host-select row-precedence <precedence_value> table { { { 1 | 2 }
host <host_name> [ realm <realm_id> ] [ secondary host <host_name> [ realm <realm_id> ] ]
} | { prefix-table { 1 | 2 } msisdn-prefix-from <msisdn_prefix_from> msisdn-prefix-to
<msisdn_prefix_to> host <host_name> [ realm <realm_id> ] [ secondary host <sec_host_name>
[ realm <sec_realm_id> ] algorithm { active-standby | round-robin } ] } } [ -noconfirm ]

            diameter host-select reselect subscriber-limit <subscriber_limit> time-
interval <duration>

            failure-handling cc-request-type { any-request | initial-request | terminate-
request | update-request } { diameter-result-code { any-error | <result_code> [ to
<end_result_code> ] } } { continue | retry-and-terminate | terminate }

        end

```

Notes:

- <context_name> must be the name of the context where you want to enable IMS Authorization service.
- <imsa_service_name> must be the name of the IMS Authorization service to be configured for Rel. 7 Gx interface authentication.
- A maximum of 16 authorization services can be configured globally in a system. There is also a system limit for the maximum number of total configured services.
- To enable Rel. 7 Gx interface support, pertinent Diameter dictionary must be configured. For information on the specific Diameter dictionary to use, contact your Cisco account representative.

- When configuring the MSISDN prefix range based PCRF selection mechanism:

To enable the Gx interface to connect to a specific PCRF for a range of subscribers configure **msisdn-prefix-from** *<msisdn_prefix_from>* and **msisdn-prefix-to** *<msisdn_prefix_to>* with the starting and ending MSISDNs respectively.

To enable the Gx interface to connect to a specific PCRF for a specific subscriber, configure both **msisdn-prefix-from** *<msisdn_prefix_from>* and **msisdn-prefix-to** *<msisdn_prefix_to>* with the same MSISDN.

In StarOS 8.1 and later releases, per MSISDN prefix range table a maximum of 128 rows can be added. In StarOS 8.0 and earlier releases, a maximum of 100 rows can be added.

The MSISDN ranges must not overlap between rows.

- The Round Robin algorithm for PCRF selection is effective only over a large number of PCRF selections, and not at a granular level.
- *Optional:* To configure the Quality of Service (QoS) update timeout for a subscriber, in the IMS Authorization Service Configuration Mode, enter the following command:

```
qos-update-timeout <timeout_duration>
```



Important: This command is obsolete in release 11.0 and later releases.

- *Optional:* To configure signalling restrictions, in the IMS Authorization Service Configuration Mode, enter the following commands:


```
signaling-flag { deny | permit }

signaling-flow permit server-address <ip_address> [ server-port { <port_number> |
range <start_number> to <end_number> } ] [ description <string> ]
```
- *Optional:* To configure action on packets that do not match any policy gates in the general purpose PDP context, in the IMS Authorization Service Configuration Mode, enter the following command:


```
traffic-policy general-pdp-context no-matching-gates direction { downlink | uplink
} { forward | discard }
```
- To configure the PCRF host destinations configured in the GGSN/PCEF, use the **diameter host-select** CLI commands.
- To configure the GGSN/PCEF to use a pre-defined rule when the Gx fails, set the **failure-handling cc-request-type** CLI to **continue**. Policies available/in use will continue to be used and there will be no further interaction with the PCRF.
- For provisioning of default charging method, use the following configurations. For this, the AVPs Online and Offline will be sent in CCR-I message based on the configuration.

- To send Enable Online:

```
configure
  active-charging service <ecs_service_name>
  charging-action <charging_action_name>
  cca charging credit
exit
```

- To send Enable Offline:

```
configure
```

```

active-charging service <ecs_service_name>
    rulebase <rulebase_name>
    billing-records rf
exit

```

Verifying the Configuration

To verify the IMS Authorization service configuration:

Step 1 Change to the context where you enabled IMS Authorization service by entering the following command:

```
context <context_name>
```

Step 2 Verify the IMS Authorization service's configurations by entering the following command:

```
show ims-authorization service name <imsa_service_name>
```

Applying IMS Authorization Service to an APN

After configuring IMS Authorization service at the context-level, an APN must be configured to use the IMS Authorization service for an IMS subscriber.

Use the following example to apply IMS Authorization service functionality to a previously configured APN within the context configured in the [Configuring Rel. 7 Gx Interface](#) section.

configure

```

context <context_name>

    apn <apn_name>

    ims-auth-service <imsa_service_name>

    active-charging rulebase <rulebase_name>

end

```

Notes:

- <context_name> must be the name of the context in which the IMS Authorization service was configured.
- <imsa_service_name> must be the name of the IMS Authorization service configured for IMS authentication in the context.
- For Rel. 7 Gx, the ECS rulebase must be configured in the APN.
- ECS allows change of rulebase via Gx for PCEF binding scenarios. When the old rulebase goes away, all the rules that were installed from that rulebase are removed. This may lead to termination of a few bearers (PDP contexts) if they are left without any rules. If there is a Gx message that changes the rulebase, and also activates some predefined rules, the rulebase change is made first, and the rules are activated from the new rulebase. Also, the rulebase applies to the entire call. All PDP contexts (bearers) in one call use the same ECS rulebase.
- For predefined rules configured in the ECS, MBR/GBR of a dynamic/predefined rule is checked before it is used for PCEF binding. All rules (dynamic as well as predefined) have to have an MBR associated with them and all rules with GBR QCI should have GBR also configured. So for predefined rules, one needs to configure

appropriate peak-data-rate, committed-data-rate as per the QCI being GBR QCI or non-GBR QCI. For more information, in the ACS Charging Action Configuration Mode, see the **flow limit-for-bandwidth** CLI command.

- Provided interpretation of the Gx rulebase is chosen to be ECS group-of-ruledefs, in the Active Charging Service Configuration Mode configure the following command:

```
policy-control charging-rule-base-name active-charging-group-of-ruledefs
```

Verifying Subscriber Configuration

Verify the IMS Authorization service configuration for subscriber(s) by entering the following command:

```
show subscribers ims-auth-service <imsa_service_name>
```

<imsa_service_name> must be the name of the IMS Authorization service configured for IMS authentication.

Configuring Volume Reporting over Gx

This section describes the configuration required to enable Volume Reporting over Gx.

To enable Volume Reporting over Gx, use the following configuration:

```
configure
```

```
    active-charging service <ecs_service_name>
```

```
        rulebase <rulebase_name>
```

```
            action priority <priority> dynamic-only ruledef <ruledef_name> charging-action
            <charging_action_name> monitoring-key <monitoring_key>
```

```
        exit
```

```
    exit
```

```
context <context_name>
```

```
    ims-auth-service <imsa_service_name>
```

```
        policy-control
```

```
            event-update send-usage-report [ reset-usage ]
```

```
        end
```

Notes:

- The maximum accepted monitoring key value by the PCEF is 4294967295. If the PCEF sends a greater value, the value is converted to an Unsigned Integer value.
- The **event-update** CLI which enables volume usage report to be sent in event updates is available only in 10.2 and later releases. The optional keyword **reset-usage** enables to support delta reporting wherein the usage is reported and reset at PCEF. If this option is not configured, the behavior is to send the usage information as part of event update but not reset at PCEF.

Gathering Statistics

This section explains how to gather Rel. 7 Gx statistics and configuration information.

In the following table, the first column lists what statistics to gather, and the second column lists the action to perform.

Table 33. Gathering Rel. 7 Gx Statistics and Information

Statistics/Information	Action to perform
Information and statistics specific to policy control in IMS Authorization service.	<code>show ims-authorization policy-control statistics</code>
Information and statistics specific to the authorization servers used for IMS Authorization service.	<code>show ims-authorization servers ims-auth-service</code>
Information of all IMS Authorization service.	<code>show ims-authorization service all</code>
Statistics of IMS Authorization service.	<code>show ims-authorization service statistics</code>
Information, configuration, and statistics of sessions active in IMS Authorization service.	<code>show ims-authorization sessions all</code>
Complete information, configuration, and statistics of sessions active in IMS Authorization service.	<code>show ims-authorization sessions full</code>
Summarized information of sessions active in IMS Authorization service.	<code>show ims-authorization sessions summary</code>
Complete statistics for active charging service sessions.	<code>show active-charging sessions full</code>
Information for all rule definitions configured in the service.	<code>show active-charging ruledef all</code>
Information for all rulebases configured in the system.	<code>show active-charging rulebase all</code>
Information on all group of ruledefs configured in the system.	<code>show active-charging group-of-ruledefs all</code>
Information on policy gate counters and status.	<code>show ims-authorization policy-gate { counters status }</code> This command is no longer an option in StarOS release 11.0 and beyond.

Rel. 8 Gx Interface

Rel. 8 Gx interface support is available on the Cisco ASR chassis running StarOS 10.0 or StarOS 11.0 and later releases.

This section describes the following topics:

- [HA/PDSN Rel. 8 Gx Interface Support](#)
- [P-GW Rel. 8 Gx Interface Support](#)

HA/PDSN Rel. 8 Gx Interface Support

This section provides information on configuring Rel. 8 Gx interface for HA and PDSN to support policy and charging control for subscribers in CDMA networks.

The IMS service provides application support for transport of voice, video, and data independent of access support. Roaming IMS subscribers in CDMA networks require apart from other functionality sufficient, uninterrupted, consistent, and seamless user experience during an application session. It is also important that a subscriber gets charged only for the resources consumed by the particular IMS application used.

It is recommended that before using the procedures in this section you select the configuration example that best meets your service model, and configure the required elements for that model as described in this Administration Guide.

This section describes the following topics:

- [Introduction](#)
- [Terminology and Definitions](#)
- [How it Works](#)
- [Configuring HA/PDSN Rel. 8 Gx Interface Support](#)
- [Gathering Statistics](#)

Introduction

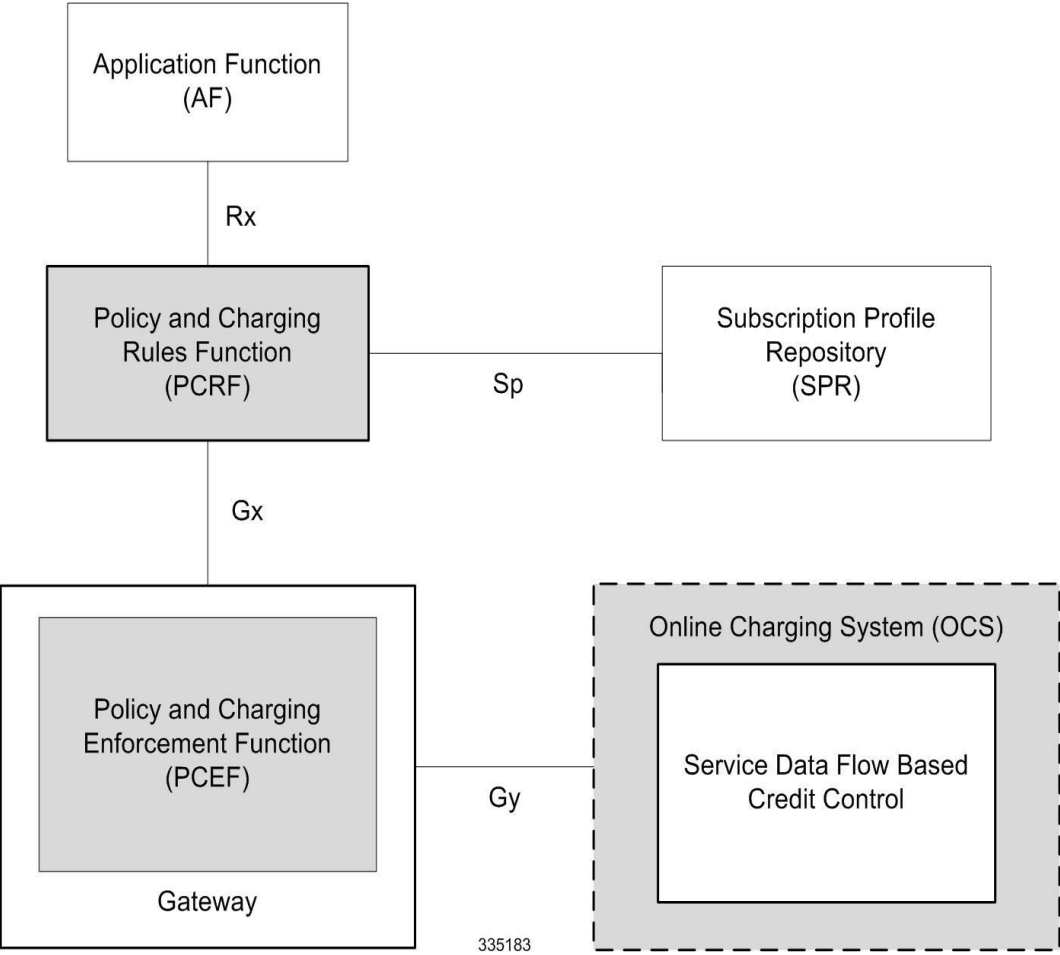
For IMS deployment in CDMA networks the system uses Rel. 8 Gx interface for policy-based admission control support and flow-based charging (FBC). The Rel. 8 Gx interface supports enforcing policy control features like gating, bandwidth limiting, and so on, and also supports FBC. This is accomplished via dynamically provisioned Policy Control and Charging (PCC) rules. These PCC rules are used to identify Service Data Flows (SDF) and to do charging. Other parameters associated with the rules are used to enforce policy control.

The PCC architecture allows operators to perform service-based QoS policy and FBC control. In the PCC architecture, this is accomplished mainly by the Policy and Charging Enforcement Function (PCEF)/HA/PDSN and the Policy and Charging Rules Function (PCRF). The client functionality lies with the HA/PDSN, therefore in the IMS Authorization (IMSA) scenario it is also called the Gateway. The PCEF function is provided by the Enhanced Charging Service (ECS). The Gx interface is implemented as a Diameter connection. The Gx messaging mostly involves installing/modifying/removing dynamic rules and activating/deactivating predefined rules.

The Gx reference point is located between the Gateway/PCEF and the PCRF. This reference point is used for provisioning and removal of PCC rules from the PCRF to the Gateway/PCEF, and the transmission of traffic plane events from the Gateway/PCEF to the PCRF. The Gx reference point can be used for charging control, policy control, or both by applying AVPs relevant to the application.

The following figure shows the reference points between elements involved in the policy and charging architecture.

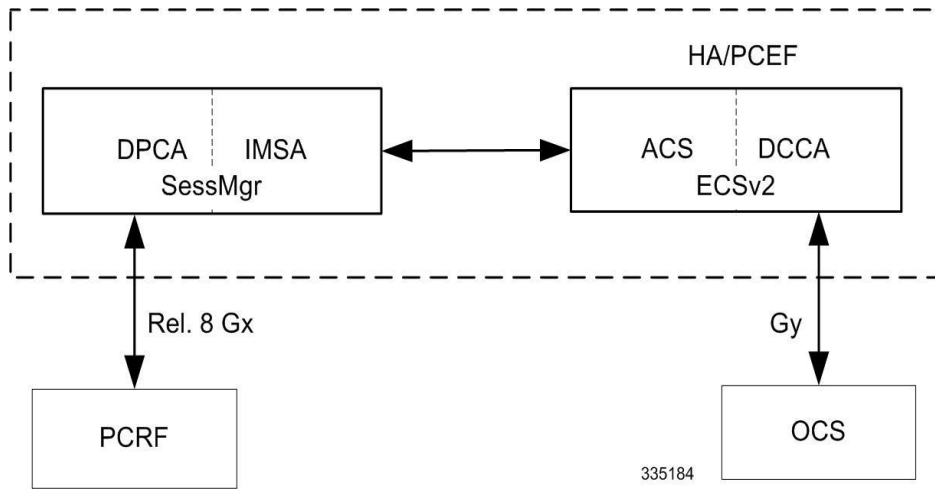
Figure 29. HA/PDSN Rel. 8 Gx PCC Logical Architecture



Within the Gateway, the IMSA and DPCA modules handle the Gx protocol related functions (at the SessMgr) and the policy enforcement and charging happens at ECS. The Gy protocol related functions are handled within the DCCA module (at the ECS).

The following figure shows the interaction between components within the Gateway.

Figure 30. HA/PDSN Rel. 8 Gx PCC Architecture within PCEF



License Requirements

The HA/PDSN Rel. 8 Gx interface support is a licensed Cisco feature. A separate feature license may be required. Contact your Cisco account representative for detailed information on specific licensing requirements. For information on installing and verifying licenses, refer to the *Managing License Keys* section of the *Software Management Operations* chapter in the *System Administration Guide*.

Supported Standards

HA/PDSN Rel 8. Gx interface support is based on the following standards and RFCs:

- 3GPP TS 23.203 V8.3.0 (2008-09) 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Policy and charging control architecture (Release 8)
- 3GPP TS 29.212 V8.6.0 (2009-12) 3rd Generation Partnership Project; Technical Specification Group Core Network and Terminals; Policy and Charging Control over Gx reference point (Release 8)
- 3GPP TS 29.213 V8.1.1 (2008-10) 3rd Generation Partnership Project; Technical Specification Group Core Network and Terminals; Policy and Charging Control signalling flows and QoS parameter mapping; (Release 8)
- RFC 3588, Diameter Base Protocol; September 2003
- RFC 4006, Diameter Credit-Control Application; August 2005

Terminology and Definitions

This section describes features and terminology pertaining to HA/PDSN Rel. 8 Gx functionality.

Policy Control

The process whereby the PCRF indicates to the PCEF how to control the IP-CAN session.

Policy control comprises the following functions:

- Binding
- Gating Control

- Event Reporting
- QoS Control
- Other Features


Binding


In the HA/PDSN Rel. 8 Gx implementation, since there are no bearers within a MIP session the IP-CAN Bearer concept does not apply. Only authorized IP-CAN session is applicable.

Gating Control

Gating control is the blocking or allowing of packets belonging to an SDF, to pass through to the desired endpoint. A gate is described within a PCC rule and gating control is applied on a per SDF basis. The commands to open or close the gate leads to the enabling or disabling of the passage for corresponding IP packets. If the gate is closed, all packets of the related IP flows are dropped. If the gate is open, the packets of the related IP flows are allowed to be forwarded.

Event Reporting


 **Important:** Unconditional reporting of event triggers from PCRF to PCEF when PCEF has not requested for is not supported.

 **Important:** In the HA/PDSN Rel. 8 Gx implementation, only the AN_GW_CHANGE (21) event trigger is supported.

Event reporting is the notification of and reaction to application events to trigger new behavior in the user plane as well as the reporting of events related to the resources in the Gateway (PCEF). Event triggers may be used to determine which IP-CAN session modification or specific event causes the PCEF to re-request PCC rules. Event trigger reporting from PCEF to PCRF, and provisioning of event triggers happens at IP-CAN session level.

The Event Reporting Function (ERF) located in the PCEF, receives event triggers from PCRF during the Provision of PCC Rules procedure and performs event trigger detection. When an event matching the received event trigger occurs, the ERF reports the occurred event to the PCRF. If the provided event triggers are associated with certain parameter values then the ERF includes those values in the response to the PCRF.

QoS Control

 **Important:** In the HA/PDSN Rel. 8 Gx implementation, only authorized IP-CAN Session is supported. Provisioning of authorized QoS per IP-CAN bearer, policy enforcement for authorized QoS per QCI, and coordination of authorized QoS scopes in mixed mode are not applicable.

QoS control is the authorization and enforcement of the maximum QoS that is authorized for an SDF. In case of an aggregation of multiple SDFs, the combination of the authorized QoS information of the individual SDFs is provided as the authorized QoS for this aggregate. QoS control per SDF allows the PCC architecture to provide the PCEF with the authorized QoS to be enforced for each specific SDF.

QoS authorization information may be dynamically provisioned by the PCRF, or it can be a predefined PCC rule in the PCEF. For a predefined PCC rule within the PCEF, the authorized QoS information takes affect when the PCC rule is activated. The PCEF combines the different sets of authorized QoS information, that is the information received from the PCRF and the information corresponding to the predefined PCC rules. The PCRF knows the authorized QoS

information of the predefined PCC rules and takes this information into account when activating them. This ensures that the combined authorized QoS of a set of PCC rules that are activated by the PCRF is within the limitations given by the subscription and operator policies regardless of whether these PCC rules are dynamically provided, predefined, or both.

Supported features include:

- Provisioning and Policy Enforcement of Authorized QoS: The PCRF may provide authorized QoS to the PCEF. The authorized QoS provides appropriate values for resources to be enforced.
- Policy Provisioning for Authorized QoS Per SDF: The provisioning of authorized QoS per SDF is a part of PCC rule provisioning procedure.
- Policy Enforcement for Authorized QoS Per SDF: If an authorized QoS is defined for a PCC rule, the PCEF limits the data rate of the SDF corresponding to that PCC rule not to exceed the maximum authorized bandwidth for the PCC rule by discarding packets exceeding the limit.
- Upon deactivation or removal of a PCC rule, the PCEF frees the resources reserved for that PCC rule.

Other Features

This section describes some of the other features.

PCC Rule Error Handling

If the installation/activation of one or more PCC rules fails, the PCEF communicates the failure to the PCRF by including one or more Charging-Rule-Report AVP(s) in either a CCR or an RAA command for the affected PCC rules. Within each Charging-Rule-Report AVP, the PCEF identifies the failed PCC rule(s) by including the Charging-Rule-Name AVP(s) or Charging-Rule-Base-Name AVP(s), identifies the failed reason code by including a Rule-Failure-Code AVP, and includes the PCC-Rule-Status AVP.

If the installation/activation of one or more new PCC rules (that is, rules that were not previously successfully installed) fail, the PCEF sets the PCC-Rule-Status to INACTIVE for both the PUSH and the PULL modes.

If a PCC rule was successfully installed/activated, but can no longer be enforced by the PCEF, the PCEF sends the PCRF a new CCR command and includes the Charging-Rule-Report AVP. The PCEF includes the Rule-Failure-Code AVP within the Charging-Rule-Report AVP and sets the PCC-Rule-Status to INACTIVE.

In the HA/PDSN Gx implementation, the following rule failure codes are supported:

- RATING_GROUP_ERROR (2)
- SERVICE_IDENTIFIER_ERROR (3)
- GW/PCEF_MALFUNCTION (4)
- RESOURCES_LIMITATION (5)

If the installation/activation of one or more PCC rules fails during RAR procedure, the RAA command is sent with the Experimental-Result-Code AVP set to DIAMETER_PCC_RULE_EVENT (5142).

Time of the Day Procedures

PCEF performs PCC rule request as instructed by the PCRF. Revalidation-Time when set by the PCRF, causes the PCEF to trigger a PCRF interaction to request PCC rules from the PCRF for an established IP-CAN session. The PCEF stops the timer once the PCEF triggers a REVALIDATION_TIMEOUT event.

When installed, the PCC rule is inactive. If Rule-Activation-Time / Rule-Deactivation-Time is specified, then the PCEF sets the rule active / inactive after that time.

Charging Control



Important: In the HA/PDSN Rel. 8 Gx implementation, offline charging is not supported.

Charging Control is the process of associating packets belonging to an SDF to a charging key, and applying online charging as appropriate. FBC handles differentiated charging of the bearer usage based on real-time analysis of the SDFs. In order to allow for charging control, the information in the PCC rule identifies the SDF and specifies the parameters for charging control. The PCC rule information may depend on subscription data.

Online charging is supported via the Gy interface. In the case of online charging, it is possible to apply an online charging action upon PCEF events (for example, re-authorization upon QoS change).

It is possible to indicate to the PCEF that interactions with the charging systems are not required for a PCC rule, that is to perform neither accounting nor credit control for this SDF, then neither online nor offline charging is performed.

Supported Features:

- Provisioning of charging-related information for the IP-CAN Session
- Provisioning of charging addresses: Primary or secondary event charging function name (Online Charging Server (OCS) addresses)



Important: In the HA/PDSN Rel. 8 Gx implementation, provisioning of primary or secondary charging collection function name (Offline Charging Server (OFCS) addresses) over Gx is not supported.

- Provisioning of Default Charging Method

Charging Correlation

In the HA/PDSN Rel. 8 Gx implementation, Charging Correlation is not supported. PCRF provides the flow identifier, which uniquely identifies an IP flow in an IMS session.

Policy and Charging Control (PCC) Rules

A PCC rule enables the detection of an SDF and provides parameters for policy control and/or charging control. The purpose of the PCC rule is to:

- Detect a packet belonging to an SDF in case of both uplink and downlink IP flows based on SDF filters in the PCC rule (packet rule matching).

If no PCC rule matches the packet, the packet is dropped.

- Identify the service that the SDF contributes to.
- Provide applicable charging parameters for an SDF.
- Provide policy control for an SDF.

The PCEF selects a PCC rule for each packet received by evaluating received packets against SDF filters of PCC rules in the order of precedence of the PCC rules. When a packet matches an SDF filter, the packet matching process for that packet is completed, and the PCC rule for that filter is applied.

There are two types of PCC rules:

- **Dynamic PCC Rules:** Rules dynamically provisioned by the PCRF to the PCEF via the Gx interface. These PCC rules may be either predefined or dynamically generated in the PCRF. Dynamic PCC rules can be activated, modified, and deactivated at any time.

- **Predefined PCC Rule:** Rules preconfigured in the PCEF by the operators. Predefined PCC rules can be activated or deactivated by the PCRF at any time. Predefined PCC rules within the PCEF may be grouped allowing the PCRF to dynamically activate a set of PCC rules over the Gx reference point.



Important: A third kind of rule, the static PCC rule can be preconfigured in the chassis by the operators. Static PCC rules are not explicitly known in the PCRF, and are not under control of the PCRF. Static PCC rules are bound to general purpose bearer with no Gx control.

A PCC rule consists of:

- **Rule Name:** The rule name is used to reference a PCC rule in the communication between the PCEF and PCRF.
- **Service Identifier:** The service identifier is used to identify the service or the service component the SDF relates to.
- **Service Data Flow Filter(s):** The service flow filter(s) is used to select the traffic for which the rule applies.
- **Precedence:** For different PCC rules with overlapping SDF filter, the precedence of the rule determines which of these rules is applicable. When a dynamic PCC rule and a predefined PCC rule have the same priority, the dynamic PCC rule takes precedence.
- **Gate Status:** The gate status indicates whether the SDF, detected by the SDF filter(s), may pass (gate is open) or will be discarded (gate is closed) in uplink and/or in downlink direction.
- **QoS Parameters:** The QoS information includes the QoS class identifier (authorized QoS class for the SDF), and authorized bitrates for uplink and downlink.
- **Charging Key (rating group)**
- **Other charging parameters:** The charging parameters define whether online charging interfaces are used, on what level the PCEF will report the usage related to the rule, etc.



Important: Configuring the Metering Method and Reporting Level for dynamic PCC rules is not supported.

PCC rules also include Application Function (AF) record information for enabling charging correlation between the application and bearer layer if the AF has provided this information via the Rx interface. For IMS, this includes the IMS Charging Identifier (ICID) and flow identifiers.

PCC Procedures over Gx Reference Point

Request for PCC Rules

The PCEF, via the Gx reference point, requests for PCC rules in the following instances:

- At IP-CAN session establishment
- At IP-CAN session modification

PCC rules can also be requested as a consequence of a failure in the PCC rule installation/activation or enforcement without requiring an event trigger.

Provisioning of PCC Rules

The PCRF indicates, via the Rel. 8 Gx reference point, the PCC rules to be applied at the PCEF. This may be using one of the following procedures:


- **PULL** (provisioning solicited by the PCEF): In response to a request for PCC rules being made by the PCEF, the PCRF provisions PCC rules in the CC-Answer.
- **PUSH** (unsolicited provisioning): The PCRF may decide to provision PCC rules without obtaining a request from the PCEF. For example, in response to information provided to the PCRF via the Rx reference point, or in response to an internal trigger within the PCRF. To provision PCC rules without a request from the PCEF, the PCRF includes these PCC rules in an RA-Request message. No CCR/CCA messages are triggered by this RA-Request.

For each request from the PCEF or upon unsolicited provisioning, the PCRF provisions zero or more PCC rules. The PCRF may perform an operation on a single PCC rule by one of the following means:

- To activate or deactivate a PCC rule that is predefined at the PCEF, the PCRF provisions a reference to this PCC rule within a Charging-Rule-Name AVP and indicates the required action by choosing either the Charging-Rule-Install AVP or the Charging-Rule-Remove AVP.
- To install or modify a PCRF-provisioned PCC rule, the PCRF provisions a corresponding Charging-Rule-Definition AVP within a Charging-Rule-Install AVP.
- To remove a PCC rule which has previously been provisioned by the PCRF, the PCRF provisions the name of this rule as value of a Charging-Rule-Name AVP within a Charging-Rule-Remove AVP.

Selecting a PCC Rule for Uplink IP Packets


If PCC is enabled, the PCEF selects the applicable PCC rule for each received uplink IP packet within an IP-CAN session by evaluating the packet against uplink SDF filters of PCRF-provided or predefined active PCC rules of this IP-CAN session in the order of the precedence of the PCC rules.

 **Important:** When a PCRF-provided PCC rule and a predefined PCC rule have the same precedence, the uplink SDF filters of the PCRF-provided PCC rule is applied first.

When a packet matches an SDF filter, the packet matching process for that packet is completed, and the PCC rule for that filter is applied. Uplink IP packets which do not match any PCC rule of the corresponding IP-CAN session are discarded.

Selecting a PCC Rule for Downlink IP Packets

If PCC is enabled, the PCEF selects a PCC rule for each received downlink IP packet within an IP-CAN session by evaluating the packet against downlink SDF filters of PCRF-provided or predefined active PCC rules of the IP-CAN session in the order of precedence of the PCC rules.

 **Important:** When a PCRF-provided PCC rule and a predefined PCC rule have the same precedence, the downlink SDF filters of the PCRF-provided PCC rule are applied first.

When a packet matches an SDF filter, the packet matching process for that packet is completed, and the PCC rule for that filter is applied. Downlink IP packets that do not match any PCC rule of the IP-CAN session are discarded.

The following procedures are also supported:

- **Indication of IP-CAN Session Termination:** When the IP-CAN session is being terminated the PCEF contacts the PCRF.
- **Request of IP-CAN Session Termination:** If the PCRF decides to terminate an IP-CAN session due to an internal trigger or trigger from the SPR, the PCRF informs the PCEF. The PCEF acknowledges to the PCRF and

instantly removes/deactivates all the PCC rules that have been previously installed or activated on that IP-CAN session.

The PCEF applies IP-CAN specific procedures to terminate the IP-CAN session. The HA/PDSN sends a MIP Revocation Request with the teardown indicator set to indicate that the termination of the entire IP-CAN session is requested. Furthermore, the PCEF applies the “Indication of IP-CAN Session Termination” procedure.

- Use of the Supported-Features AVP during session establishment to inform the destination host about the required and optional features that the origin host supports.

How it Works

This section describes how HA/PDSN Rel. 8 Gx Interface support works.

The following figure and table explain the IMS Authorization process between a system and IMS components that is initiated by the UE.

In this example, the Diameter Policy Control Application (DPCA) is the Gx interface to the PCRF. The interface between IMSA with PCRF is the Gx interface, and the interface between Session Manager (SessMgr) and Online Charging Service (OCS) is the Gy interface. Note that the IMSA service and DPCA are part of SessMgr on the system and separated in the figure for illustration purpose only.



Important: In 14.0 and later releases, the DPCA and the IMSA will be acting as one module within the Policy Server interface application.

Figure 31. HA/PDSN Rel. 8 Gx IMS Authorization Call Flow

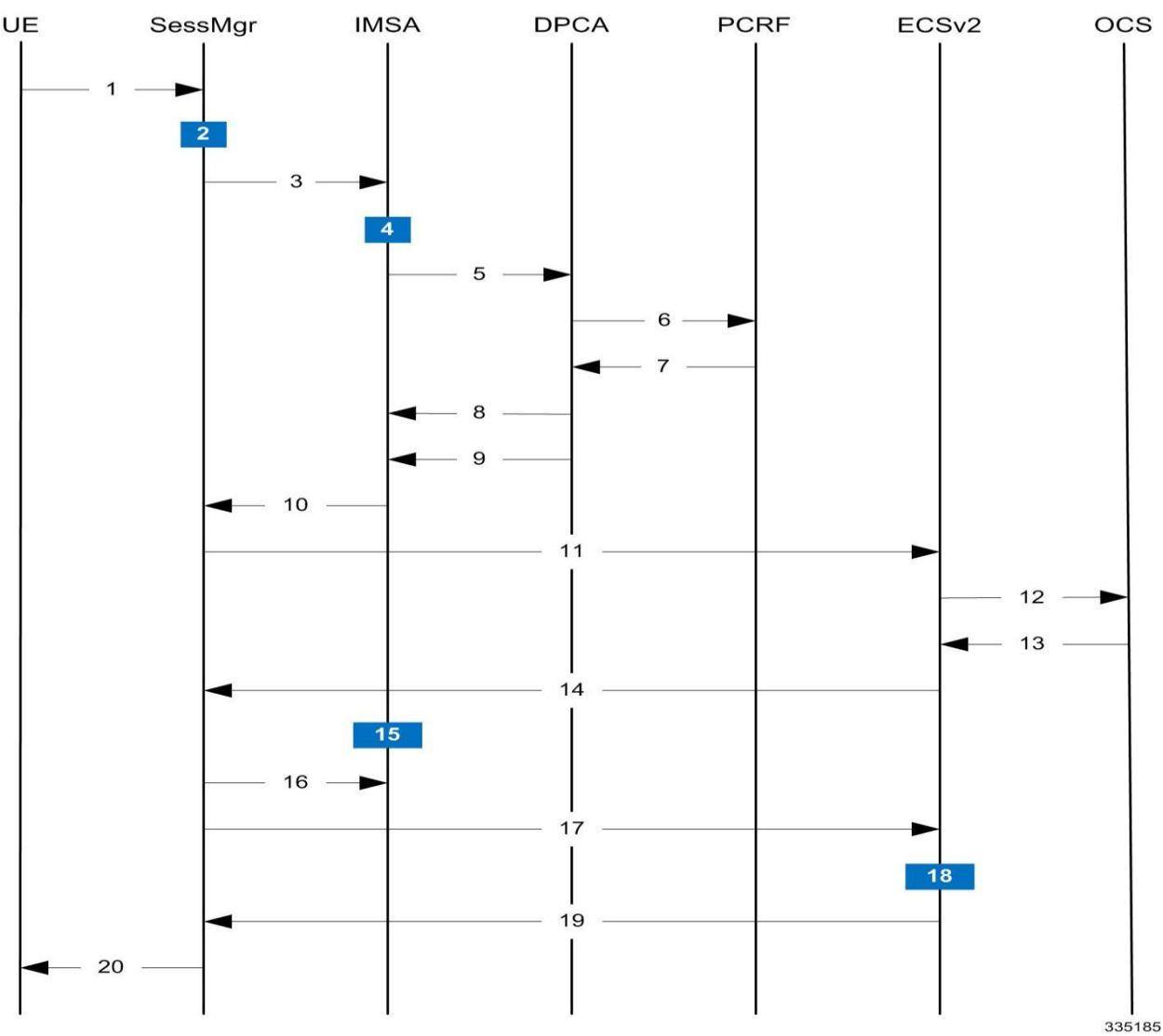


Table 34. HA/PDSN Rel. 8 Gx IMS Authorization Call flow Description

Step	Description
1	UE (IMS subscriber) requests for MIP Registration Request.
2	SessMgr allocates an IP address to the UE.
3	SessMgr requests IMS Authorization, if IMSA is enabled for the subscriber. IMSA service can either be configured in the subscriber template, or can be received from the AAA.
4	IMSA allocates resources for the IP-CAN session, and selects the PCRF to contact based on the user's selection key (for example, round-robin).
5	IMSA requests the DPCA module to issue an auth request to the PCRF.

Step	Description
6	DPCA sends a CCR initial message to the selected PCRF.
7	PCRF may send preconfigured charging rules in CCA. The dynamic rules and the authorized QoS parameters could also be included by the PCRF.
8	DPCA passes the charging rule definition, charging rule install, QoS information received from the PCRF, event triggers, etc. IMSA stores the information.
9	DPCA calls the callback function registered with it by IMSA.
10	PCRF-provided information common to the entire IP-CAN session (event trigger, primary/secondary OCS address, etc.) is stored within the IMSA. After processing the information, IMSA notifies the SessMgr about the policy authorization complete.
11	If the validation of the rules fails in IMSA/DPCA, a failure is notified to PCRF containing the Charging-Rule-Report AVP. Else, IMSA initiates creation of ECS session. The primary/secondary OCS server address, etc. are sent to the ECS from the SessMgr.
12	ECS performs credit authorization by sending CCR(I) to OCS with CC-Request-Type set to INITIAL_REQUEST to open the credit control session. This request includes the active Rulebase-Id (default rulebase ID from the AAA).
13	OCS returns a CCA initial message that may activate a statically configured Rulebase and may include preemptive quotas.
14	ECS responds to SessMgr with the response message.
15	SessMgr requests IMSA for the dynamic rules.
16	IMSA sends the dynamic rules to SessMgr. Note that, in 14.0 and later releases, the RAR messages are allowed before the session is established. In earlier releases, until the MIP session is established, all RAR messages from the PCRF were rejected. Also note that, in 14.0 and later releases, the RAR message is rejected and RAA is sent with 3002 result code when the recovery of dynamic rule information and audit of Session Manager are in progress. Earlier, the RAR messages were processed by DPCA even when the recovery audit was in progress.
17	SessMgr sends the dynamic rule information to the ECS. The gate flow status information and the QoS per flow (charging rule) information are also sent in the message.
18	ECS activates the predefined rules received, and installs the dynamic rules received. Also, the gate flow status and the QoS parameters are updated by ECS as per the dynamic charging rules. The Gx rulebase is treated as an ECS group-of-ruldefs. The response message contains the Charging Rule Report conveying the status of the rule provisioning at the ECS.
19	If the provisioning of rules fails partially, the context setup is accepted, and a new CCR-U is sent to the PCRF with the Charging-Rule-Report containing the PCC rule status for the failed rules. If the provisioning of rules fails completely, the context setup is rejected.
20	Depending on the response for the MIP Session Authorization, SessMgr sends the response to the UE and activates/rejects the call. If the Charging-Rule-Report contains partial failure for any of the rules, the PCRF is notified, and the call is activated. If the Charging-Rule-Report contains complete failure, the call is rejected.

Configuring HA/PDSN Rel. 8 Gx Interface Support

To configure HA/PDSN Rel. 8 Gx Interface functionality:

1. At the context level, configure IMSA service for IMS subscribers as described in the [Configuring IMS Authorization Service at Context Level](#) section.

2. Within the same context, configure the subscriber template to use the IMSA service as described in the Applying IMS Authorization Service to Subscriber Template section.
3. Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.



Important: Commands used in the configuration examples in this section provide base functionality to the extent that the most common or likely commands and/or keyword options are presented. In many cases, other optional commands and/or keyword options are available. Refer to the *Command Line Interface Reference* for complete information regarding all commands.

Configuring IMS Authorization Service at Context Level

Use the following example to configure IMSA service at context level for IMS subscribers:

configure

```

context <context_name>

    ims-auth-service <imsa_service_name>

        policy-control

            diameter origin endpoint <endpoint_name>

            diameter dictionary <dictionary>

            diameter request-timeout <timeout_duration>

            diameter host-select table { 1 | 2 } algorithm round-robin

            diameter host-select row-precedence <precedence_value> table { 1 | 2 } host
<primary_host_name> [ realm <primary_realm_id> ] [ secondary host <secondary_host_name> [
realm <secondary_realm_id> ] ] [ -noconfirm ]

            failure-handling cc-request-type { any-request | initial-request | terminate-
request | update-request } { diameter-result-code { any-error | <result_code> [ to
<end_result_code> ] } } { continue | retry-and-terminate | terminate }

            exit

        exit

    diameter endpoint <endpoint_name> [ -noconfirm ]

        origin realm <realm_name>

        use-proxy

        origin host <host_name> address <ip_address>

        no watchdog-timeout

        response-timeout <timeout_duration>

```



```

connection timeout <timeout_duration>

connection retry-timeout <timeout_duration>

peer <primary_peer_name> [ realm <primary_realm_name> ] address <ip_address> [
port <port_number> ]

    peer <secondary_peer_name> [ realm <secondary_realm_name> ] address <ip_address>
[ port <port_number> ]

end

```

Notes:

- <context_name> must be the name of the context where you want to enable IMSA service.
- <imsa_service_name> must be the name of the IMSA service to be configured for Rel. 8 Gx interface authentication.
- A maximum of 16 authorization services can be configured globally in a system. There is also a system limit for the maximum number of total configured services.
- To enable Rel. 8 Gx interface support, pertinent Diameter dictionary must be configured. For information on the specific Diameter dictionary to use, contact your Cisco account representative.
- The Round Robin algorithm for PCRF selection is effective only over a large number of PCRF selections, and not at a granular level.
- To configure the PCRF host destinations configured in the PCEF, use the diameter host-select CLI commands.
- To configure the PCEF to use a pre-defined rule when the Gx fails, set the **failure-handling cc-request-type** CLI to **continue**. Policies available/in use will continue to be used and there will be no further interaction with the PCRF.

Verifying the IMSA Service Configuration

To verify the IMSA service configuration:

- Change to the context where you enabled IMSA service by entering the following command:

```
context <context_name>
```
- Verify the IMSA service's configuration by entering the following command:

```
show ims-authorization service name <imsa_service_name>
```

Applying IMS Authorization Service to Subscriber Template

After configuring IMSA service at the context-level, within the same context subscriber template must be configured to use the IMSA service for IMS subscribers.

Use the following example to apply IMSA service functionality to subscriber template within the context previously configured in the [Configuring IMS Authorization Service at Context Level](#) section.

```

configure

context <context_name>

    subscriber default

        encrypted password <encrypted_password>

```

```
ims-auth-service <imsa_service_name>

ip access-group <access_group_name> in

ip access-group <access_group_name> out

ip context-name <context_name>

mobile-ip home-agent <ip_address>

active-charging rulebase <rulebase_name>

end
```

Notes:

- <context_name> must be the name of the context in which the IMSA service was configured.
- <imsa_service_name> must be the name of the IMSA service configured for IMS authentication in the context.
- The ECS rulebase must be configured in the subscriber template.
- Provided interpretation of the Gx rulebase (Charging-Rule-Base-Name AVP) from PCRF is chosen to be ECS group-of-ruledefs, configure the following command in the Active Charging Service Configuration Mode:
policy-control charging-rule-base-name active-charging-group-of- ruledefs

Verifying the Subscriber Configuration

Verify the IMSA service configuration for subscriber(s) by entering the following command in the Exec CLI configuration mode:

```
show subscribers ims-auth-service <imsa_service_name>
```

Notes:

<imsa_service_name> must be the name of the IMSA service configured for IMS authentication.

Gathering Statistics

This section explains how to gather Rel. 8 Gx statistics and configuration information.
In the following table, the first column lists what statistics to gather, and the second column lists the action to perform.

Table 35. Gathering HA/PDSN Rel. 8 Gx Statistics and Information

Statistics/Information	Action to perform
Information and statistics specific to policy control in IMS Authorization service.	show ims-authorization policy-control statistics
Information and statistics specific to the authorization servers used for IMS Authorization service.	show ims-authorization servers ims-auth-service
Information of all IMS Authorization service.	show ims-authorization service all
Statistics of IMS Authorization service.	show ims-authorization service statistics

Statistics/Information	Action to perform
Information, configuration, and statistics of sessions active in IMS Authorization service.	show ims-authorization sessions all
Complete information, configuration, and statistics of sessions active in IMS Authorization service.	show ims-authorization sessions full
Summarized information of sessions active in IMS Authorization service.	show ims-authorization sessions summary
Complete statistics for active charging service sessions.	show active-charging sessions full
Information for all rule definitions configured in the service.	show active-charging ruledef all
Information for all rulebases configured in the system.	show active-charging rulebase all
Information on all group of ruledefs configured in the system.	show active-charging group-of-ruledefs all
Information on policy gate counters and status.	show ims-authorization policy-gate { counters status } This command is no longer an option in StarOS release 11.0 and beyond.

P-GW Rel. 8 Gx Interface Support

Introduction

The Gx reference point is located between the Policy and Charging Rules Function (PCRF) and the Policy and Charging Enforcement Function (PCEF) on the Packet Data Network (PDN) Gateway (P-GW). The Gx reference point is used for provisioning and removal of PCC rules from the PCRF to the PCEF and the transmission of traffic plane events from the PCEF to the PCRF. The Gx reference point can be used for charging control, policy control, or both, by applying AVPs relevant to the application.

The PCEF is the functional element that encompasses policy enforcement and flow based charging functionality. This functional entity is located at the P-GW. The main functions include:

- Control over the user plane traffic handling at the gateway and its QoS.
- Service data flow detection and counting, as well as online and offline charging interactions.
- For a service data flow that is under policy control, the PCEF shall allow the service data flow to pass through the gateway if and only if the corresponding gate is open.
- For a service data flow that is under charging control, the PCEF shall allow the service data flow to pass through the gateway if and only if there is a corresponding active PCC rule and, for online charging, the OCS has authorized the applicable credit with that charging key.
- If requested by the PCRF, the PCEF shall report to the PCRF when the status of the related service data flow changes.
- In case the SDF is tunnelled at the BBERF, the PCEF shall inform the PCRF about the mobility protocol tunnelling header of the service data flows at IP-CAN session establishment.

Terminology and Definitions


This section describes features and terminology pertaining to Rel. 8 Gx functionality.

Volume Reporting Over Gx

This section describes the 3GPP Rel. 9 Volume Reporting over Gx feature.

License Requirements

The Volume Reporting over Gx is a licensed Cisco feature. A separate feature license may be required. Contact your Cisco account representative for detailed information on specific licensing requirements. For information on installing and verifying licenses, refer to the *Managing License Keys* section of the *Software Management Operations* chapter in the *System Administration Guide*.

 **Important:** In 12.0 and later releases, no separate license is required for Charging over Gx / Volume Reporting over Gx feature. This feature can be enabled as part of "Policy Interface" license.


Supported Standards


The Volume Reporting over Gx feature is based on the following standard:


3GPP TS 29.212 V9.5.0 (2010-06): 3rd Generation Partnership Project; Technical Specification Group Core Network and Terminals; Policy and Charging Control over Gx reference point (Release 9).


Feature Overview


The Volume Reporting over Gx feature provides PCRF the capability to make real-time decisions based on the data usage by subscribers.

 **Important:** Volume Reporting over Gx is applicable only for volume quota.

 **Important:** In release 10.0, only total data usage reporting is supported, uplink/downlink level reporting is not supported. In 10.2 and later releases, it is supported.

 **Important:** The PCEF only reports the accumulated usage since the last report for usage monitoring and not from the beginning.

 **Important:** If the usage threshold is set to zero (infinite threshold), no further threshold events will be generated by PCEF, but monitoring of usage will continue and be reported at the end of the session.

 **Important:** In 12.2 and later releases, usage reporting on bearer termination is supported.

The following steps explain how Volume Reporting over Gx works:

1. PCEF after receiving the message from PCRF parses the usage monitoring related AVPs, and sends the information to IMSA.

2. IMSA updates the information to ECS.
3. Once the ECS is updated with the usage monitoring information from PCRF, the PCEF (ECS) starts tracking the data usage.
4. For session-level monitoring, the ECS maintains the amount of data usage.
5. For PCC rule monitoring, usage is monitored with the monitoring key as the unique identifier. Each node maintains the usage information per monitoring key. When the data traffic is passed, the usage is checked against the usage threshold values and reported as described in the *Usage Reporting* section.
6. The PCEF continues to track data usage after the threshold is reached and before a new threshold is provided by the PCRF. If a new usage threshold is not provided by the PCRF in the acknowledgement of an IP-CAN Session modification where its usage was reported, then usage monitoring does not continue in the PCEF for that IP CAN session.

Usage Monitoring

- **Usage Monitoring at Session Level:** PCRF subscribes to the session-level volume reporting over Gx by sending the Usage-Monitoring-Information AVP with the usage threshold level set in Granted-Service-Unit AVP and Usage-Monitoring-Level AVP set to SESSION_LEVEL(0). After the AVPs are parsed by DPCA, IMSA updates the information to ECS. Once ECS is updated usage monitoring is started and constantly checked with the usage threshold whenever the data traffic is present. In 11.0 and later releases, Monitoring Key at session level is supported.

In 12.0 and later releases, enabling and disabling session usage in a single message from PCRF is supported. This is supported only if the monitoring key is associated at session level.

In 12.0 and later releases, monitoring of usage based on input/output octet threshold levels is supported. Usage is reported based on the enabled threshold level. If multiple levels are enabled, usage will be reported on all the enabled levels even if only one of the levels is breached. Monitoring will be stopped on the missing threshold levels in the response for the usage report from PCRF (expected to provide the complete set again if PCRF wants to continue monitoring on the multiple levels enabled earlier).

Total threshold level along with UL/DL threshold level in the GSU AVP is treated as an error and only total threshold level is accepted.

- **Usage Monitoring at Flow Level:** PCRF subscribes to the flow-level volume reporting over Gx by sending the Usage-Monitoring-Information AVP with the usage threshold level set in Granted-Service-Unit AVP and Usage-Monitoring-Level AVP set to PCC_RULE_LEVEL(1). Monitoring Key is mandatory in case of a flow-level monitoring since the rules are associated with the monitoring key and enabling/disabling of usage monitoring at flow level can be controlled by PCRF using it. After the AVPs are parsed by DPCA, IMSA updates the information to ECS. Once ECS is updated usage monitoring is started and constantly checked with the usage threshold whenever the data traffic is present.

Usage monitoring is supported for static, predefined rules, and dynamic rule definitions.

- **Usage Monitoring for Static Rules:** In the case of static rules, the usage reporting on last rule removal associated with the monitoring key is not applicable. In this case only the usage monitoring information is received from the PCRF.
- **Usage Monitoring for Predefined Rules:** If the usage monitoring needs to be enabled for the predefined rules, PCRF sends the rule and the usage monitoring information containing the monitoring key and the usage threshold. The Monitoring key should be same as the one pre-configured in PCEF for that predefined rule. There can be multiple rules associated with the same monitoring key. Hence enabling a particular monitoring key would result in the data being tracked for multiple rules having the same monitoring key. After DPCA parses the AVPs IMSA updates the information to ECS. Once ECS is

updated usage monitoring is started and constantly checked with the usage threshold whenever the data traffic is present.

- **Usage Monitoring for Dynamic Rules:** If the usage monitoring needs to be enabled for dynamic ruledefs, PCRF provides the monitoring key along with a charging rule definition and the usage monitoring information containing the monitoring key and the usage threshold. This would result in the usage monitoring being done for all the rules associated with that monitoring key. After DPCA parses the AVPs, IMSA updates the information to ECS. Once ECS is updated, the usage monitoring is started and constantly checked with the usage threshold whenever the data traffic is present. Monitoring key for dynamic ruledef is dynamically assigned by PCRF which is the only difference with predefined rules in case of usage monitoring.

Usage Reporting

Usage at subscriber/flow level is reported to PCRF under the following conditions:

- **Usage Threshold Reached:** PCEF records the subscriber data usage and checks if the usage threshold provided by PCRF is reached. This is done for both session and rule level reporting.

For session-level reporting, the actual usage volume is compared with the usage volume threshold.

For rule-level reporting the rule that hits the data traffic is used to find out if the monitoring key is associated with it, and based on the monitoring key the data usage is checked. Once the condition is met, it reports the usage information to IMSA and continues monitoring. IMSA then triggers the CCR-U if “USAGE_REPORT” trigger is enabled by the PCRF. The Usage-Monitoring-Information AVP is sent in this CCR with the “Used-Service-Unit” set to the amount of data usage by subscriber.

If PCRF does not provide a new usage threshold in the usage monitoring information as a result of CCR from PCEF when the usage threshold is reached, the usage monitoring is stopped at PCEF and no usage status is reported.

In the non-standard Volume Reporting over Gx implementation, usage monitoring will be stopped once the threshold is breached, else the monitoring will continue. There will be no further usage reporting until the CCA is received.

- **Usage Monitoring Disabled:** If the PCRF explicitly disables the usage monitoring with Usage-Monitoring-Support AVP set to USAGE_MONITORING_DISABLED, the PCEF stops monitoring and reports the usage information (when the monitoring was enabled) to PCRF if the usage monitoring is disabled by PCRF as a result of CCR from PCEF which is not related to reporting usage, other external triggers, or a PCRF internal trigger. If the PCRF does not provide a new usage threshold as a result of CCR from PCEF when the usage threshold is reached, the usage monitoring is stopped at PCEF and no further usage status is reported.
- **IP CAN Session Termination:** When the IP CAN session is terminated, the accumulated subscriber usage information is reported to PCRF in the CCR-T from PCEF. If PCC usage level information is enabled by PCRF, the PCC usage will also be reported.
- **PCC Rule Removal:** When the PCRF deactivates the last PCC rule associated with a usage monitoring key, the PCEF sends a CCR with the data usage for that monitoring key. If the PCEF reports the last PCC rule associated with a usage monitoring key is inactive, the PCEF reports the accumulated usage for that monitoring key within the same CCR command if the Charging-Rule-Report AVP was included in a CCR command; otherwise, if the Charging-Rule-Report AVP was included in an RAA command, the PCEF sends a new CCR command to report accumulated usage for the usage monitoring key. In 12.0 and later releases, usage reporting on last rule deactivation using rule deactivation time set by PCRF is supported.
- **PCRF Requested Usage Report:** In 10.2 and later releases, the accumulated usage since the last report is sent even in case of immediate reporting, the usage is reset after immediate reporting and usage monitoring continued so that the subsequent usage report will have the usage since the current report. In earlier releases the behavior was to accumulate the so far usage in the next report.

- Release 12.2 onwards, usage reporting on bearer termination can be added. When a bearer is deleted due to some reason, the rules associated with the bearer will also be removed. So, the usage will be reported on the monitoring key(s) whose associated rule is the last one that is removed because of bearer termination.
- Revalidation Timeout: In the non-standard implementation, if usage monitoring and reporting is enabled and a revalidation timeout occurs, the PCEF sends a CCR to request PCC rules and reports all accumulated usage for all enabled monitoring keys since the last report (or since usage reporting was enabled if the usage was not yet reported) with the accumulated usage at IP-CAN session level (if enabled) and at service data flow level (if enabled) This is the default behavior.

In the case of standard implementation, this must be enabled by CLI configuration.



Important: The Usage Reporting on Revalidation Timeout feature is available by default in non-standard implementation of Volume Reporting over Gx. In 10.2 and later releases, this is configurable in the standard implementation. This is not supported in 10.0 release for standard based volume reporting.

Once the usage is reported, the usage counter is reset to zero. The PCEF continues to track data usage from the zero value after the threshold is reached and before a new threshold is provided by the PCRF. If a new usage threshold is not provided by the PCRF in the acknowledgement of an IP-CAN Session modification where its usage was reported, then usage monitoring does not continue in the PCEF for that IP CAN session and the usage accumulated between the CCR-CCA will be discarded.

For information on how to configure the Volume Reporting over Gx feature, see the [Configuring Volume Reporting over Gx](#) section.

Rel. 9 Gx Interface

Rel. 9 Gx interface support is available on the Cisco ASR chassis running StarOS 12.2 and later releases.

P-GW Rel. 9 Gx Interface Support

Introduction

The Gx reference point is located between the Policy and Charging Rules Function (PCRF) and the Policy and Charging Enforcement Function (PCEF) on the Packet Data Network (PDN) Gateway (P-GW). The Gx reference point is used for provisioning and removal of PCC rules from the PCRF to the PCEF and the transmission of traffic plane events from the PCEF to the PCRF. The Gx reference point can be used for charging control, policy control, or both, by applying AVPs relevant to the application.

The PCEF is the functional element that encompasses policy enforcement and flow based charging functionality. This functional entity is located at the P-GW. The main functions include:

- Control over the user plane traffic handling at the gateway and its QoS.
- Service data flow detection and counting, as well as online and offline charging interactions.
- For a service data flow that is under policy control, the PCEF shall allow the service data flow to pass through the gateway if and only if the corresponding gate is open.
- For a service data flow that is under charging control, the PCEF shall allow the service data flow to pass through the gateway if and only if there is a corresponding active PCC rule and, for online charging, the OCS has authorized the applicable credit with that charging key.
- If requested by the PCRF, the PCEF shall report to the PCRF when the status of the related service data flow changes.
- In case the SDF is tunnelled at the BBERF, the PCEF shall inform the PCRF about the mobility protocol tunnelling header of the service data flows at IP-CAN session establishment.

Terminology and Definitions


This section describes features and terminology pertaining to Rel. 9 Gx functionality.

Volume Reporting Over Gx

This section describes the 3GPP Rel. 9 Volume Reporting over Gx feature.

License Requirements

The Volume Reporting over Gx is a licensed Cisco feature. A separate feature license may be required. Contact your Cisco account representative for detailed information on specific licensing requirements. For information on installing and verifying licenses, refer to the *Managing License Keys* section of the *Software Management Operations* chapter in the *System Administration Guide*.

 **Important:** In 12.0 and later releases, no separate license is required for Charging over Gx / Volume Reporting over Gx feature. This feature can be enabled as part of "Policy Interface" license.


Supported Standards


The Volume Reporting over Gx feature is based on the following standard:


3GPP TS 29.212 V9.5.0 (2010-06): 3rd Generation Partnership Project; Technical Specification Group Core Network and Terminals; Policy and Charging Control over Gx reference point (Release 9).


Feature Overview


The Volume Reporting over Gx feature provides PCRF the capability to make real-time decisions based on the data usage by subscribers.

 **Important:** Volume Reporting over Gx is applicable only for volume quota.

 **Important:** In release 10.0, only total data usage reporting is supported, uplink/downlink level reporting is not supported. In 10.2 and later releases, it is supported.

 **Important:** The PCEF only reports the accumulated usage since the last report for usage monitoring and not from the beginning.

 **Important:** If the usage threshold is set to zero (infinite threshold), no further threshold events will be generated by PCEF, but monitoring of usage will continue and be reported at the end of the session.

 **Important:** In 12.2 and later releases, usage reporting on bearer termination is supported.

The following steps explain how Volume Reporting over Gx works:

1. PCEF after receiving the message from PCRF parses the usage monitoring related AVPs, and sends the information to IMSA.
2. IMSA updates the information to ECS.
3. Once the ECS is updated with the usage monitoring information from PCRF, the PCEF (ECS) starts tracking the data usage.
4. For session-level monitoring, the ECS maintains the amount of data usage.
5. For PCC rule monitoring, usage is monitored with the monitoring key as the unique identifier. Each node maintains the usage information per monitoring key. When the data traffic is passed, the usage is checked against the usage threshold values and reported as described in the *Usage Reporting* section.
6. The PCEF continues to track data usage after the threshold is reached and before a new threshold is provided by the PCRF. If a new usage threshold is not provided by the PCRF in the acknowledgement of an IP-CAN Session modification where its usage was reported, then usage monitoring does not continue in the PCEF for that IP CAN session.

Usage Monitoring

- **Usage Monitoring at Session Level:** PCRF subscribes to the session-level volume reporting over Gx by sending the Usage-Monitoring-Information AVP with the usage threshold level set in Granted-Service-Unit AVP and Usage-Monitoring-Level AVP set to SESSION_LEVEL(0). After the AVPs are parsed by DPCA, IMSA updates the information to ECS. Once ECS is updated usage monitoring is started and constantly checked with the usage threshold whenever the data traffic is present. In 11.0 and later releases, Monitoring Key at session level is supported.

In 12.0 and later releases, enabling and disabling session usage in a single message from PCRF is supported. This is supported only if the monitoring key is associated at session level.

In 12.0 and later releases, monitoring of usage based on input/output octet threshold levels is supported. Usage is reported based on the enabled threshold level. If multiple levels are enabled, usage will be reported on all the enabled levels even if only one of the levels is breached. Monitoring will be stopped on the missing threshold levels in the response for the usage report from PCRF (expected to provide the complete set again if PCRF wants to continue monitoring on the multiple levels enabled earlier).

Total threshold level along with UL/DL threshold level in the GSU AVP is treated as an error and only total threshold level is accepted.

- **Usage Monitoring at Flow Level:** PCRF subscribes to the flow-level volume reporting over Gx by sending the Usage-Monitoring-Information AVP with the usage threshold level set in Granted-Service-Unit AVP and Usage-Monitoring-Level AVP set to PCC_RULE_LEVEL(1). Monitoring Key is mandatory in case of a flow-level monitoring since the rules are associated with the monitoring key and enabling/disabling of usage monitoring at flow level can be controlled by PCRF using it. After the AVPs are parsed by DPCA, IMSA updates the information to ECS. Once ECS is updated usage monitoring is started and constantly checked with the usage threshold whenever the data traffic is present.

Usage monitoring is supported for static, predefined rules, and dynamic rule definitions.

- **Usage Monitoring for Static Rules:** In the case of static rules, the usage reporting on last rule removal associated with the monitoring key is not applicable. In this case only the usage monitoring information is received from the PCRF.
- **Usage Monitoring for Predefined Rules:** If the usage monitoring needs to be enabled for the predefined rules, PCRF sends the rule and the usage monitoring information containing the monitoring key and the usage threshold. The Monitoring key should be same as the one pre-configured in PCEF for that predefined rule. There can be multiple rules associated with the same monitoring key. Hence enabling a particular monitoring key would result in the data being tracked for multiple rules having the same monitoring key. After DPCA parses the AVPs IMSA updates the information to ECS. Once ECS is updated usage monitoring is started and constantly checked with the usage threshold whenever the data traffic is present.
- **Usage Monitoring for Dynamic Rules:** If the usage monitoring needs to be enabled for dynamic ruledefs, PCRF provides the monitoring key along with a charging rule definition and the usage monitoring information containing the monitoring key and the usage threshold. This would result in the usage monitoring being done for all the rules associated with that monitoring key. After DPCA parses the AVPs, IMSA updates the information to ECS. Once ECS is updated, the usage monitoring is started and constantly checked with the usage threshold whenever the data traffic is present. Monitoring key for dynamic ruledef is dynamically assigned by PCRF which is the only difference with predefined rules in case of usage monitoring.

Usage Reporting

Usage at subscriber/flow level is reported to PCRF under the following conditions:

- **Usage Threshold Reached:** PCEF records the subscriber data usage and checks if the usage threshold provided by PCRF is reached. This is done for both session and rule level reporting.

For session-level reporting, the actual usage volume is compared with the usage volume threshold.

For rule-level reporting the rule that hits the data traffic is used to find out if the monitoring key is associated with it, and based on the monitoring key the data usage is checked. Once the condition is met, it reports the usage information to IMSA and continues monitoring. IMSA then triggers the CCR-U if “USAGE_REPORT” trigger is enabled by the PCRF. The Usage-Monitoring-Information AVP is sent in this CCR with the “Used-Service-Unit” set to the amount of data usage by subscriber.

If PCRF does not provide a new usage threshold in the usage monitoring information as a result of CCR from PCEF when the usage threshold is reached, the usage monitoring is stopped at PCEF and no usage status is reported.

In the non-standard Volume Reporting over Gx implementation, usage monitoring will be stopped once the threshold is breached, else the monitoring will continue. There will be no further usage reporting until the CCA is received.

- **Usage Monitoring Disabled:** If the PCRF explicitly disables the usage monitoring with Usage-Monitoring-Support AVP set to USAGE_MONITORING_DISABLED, the PCEF stops monitoring and reports the usage information (when the monitoring was enabled) to PCRF if the usage monitoring is disabled by PCRF as a result of CCR from PCEF which is not related to reporting usage, other external triggers, or a PCRF internal trigger. If the PCRF does not provide a new usage threshold as a result of CCR from PCEF when the usage threshold is reached, the usage monitoring is stopped at PCEF and no further usage status is reported.
- **IP CAN Session Termination:** When the IP CAN session is terminated, the accumulated subscriber usage information is reported to PCRF in the CCR-T from PCEF. If PCC usage level information is enabled by PCRF, the PCC usage will also be reported.
- **PCC Rule Removal:** When the PCRF deactivates the last PCC rule associated with a usage monitoring key, the PCEF sends a CCR with the data usage for that monitoring key. If the PCEF reports the last PCC rule associated with a usage monitoring key is inactive, the PCEF reports the accumulated usage for that monitoring key within the same CCR command if the Charging-Rule-Report AVP was included in a CCR command; otherwise, if the Charging-Rule-Report AVP was included in an RAA command, the PCEF sends a new CCR command to report accumulated usage for the usage monitoring key. In 12.0 and later releases, usage reporting on last rule deactivation using rule deactivation time set by PCRF is supported.
- **PCRF Requested Usage Report:** In 10.2 and later releases, the accumulated usage since the last report is sent even in case of immediate reporting, the usage is reset after immediate reporting and usage monitoring continued so that the subsequent usage report will have the usage since the current report. In earlier releases the behavior was to accumulate the so far usage in the next report.
- **Release 12.2 onwards,** usage reporting on bearer termination can be added. When a bearer is deleted due to some reason, the rules associated with the bearer will also be removed. So, the usage will be reported on the monitoring key(s) whose associated rule is the last one that is removed because of bearer termination.
- **Revalidation Timeout:** In the non-standard implementation, if usage monitoring and reporting is enabled and a revalidation timeout occurs, the PCEF sends a CCR to request PCC rules and reports all accumulated usage for all enabled monitoring keys since the last report (or since usage reporting was enabled if the usage was not yet reported) with the accumulated usage at IP-CAN session level (if enabled) and at service data flow level (if enabled) This is the default behavior.

In the case of standard implementation, this must be enabled by CLI configuration.



Important: The Usage Reporting on Revalidation Timeout feature is available by default in non-standard implementation of Volume Reporting over Gx. In 10.2 and later releases, this is configurable in the standard implementation. This is not supported in 10.0 release for standard based volume reporting.

Once the usage is reported, the usage counter is reset to zero. The PCEF continues to track data usage from the zero value after the threshold is reached and before a new threshold is provided by the PCRF. If a new usage threshold is not provided by the PCRF in the acknowledgement of an IP-CAN Session modification where its usage was reported, then usage monitoring does not continue in the PCEF for that IP CAN session and the usage accumulated between the CCR-CCA will be discarded.

For information on how to configure the Volume Reporting over Gx feature, see the [Configuring Volume Reporting over Gx](#) section.

Appendix H

Gy Interface Support

This chapter provides an overview of the Gy interface and describes how to configure the Gy interface.

Gy interface support is available on the Cisco system running StarOS 9.0 or later releases for the following products:

- GGSN
- HA
- IPSG
- PDSN
- P-GW

It is recommended that before using the procedures in this chapter you select the configuration example that best meets your service model, and configure the required elements for that model as described in the administration guide for the product that you are deploying.

This chapter describes the following topics:

- [Introduction](#)
- [Features and Terminology](#)
- [Configuring Gy Interface Support](#)

Introduction

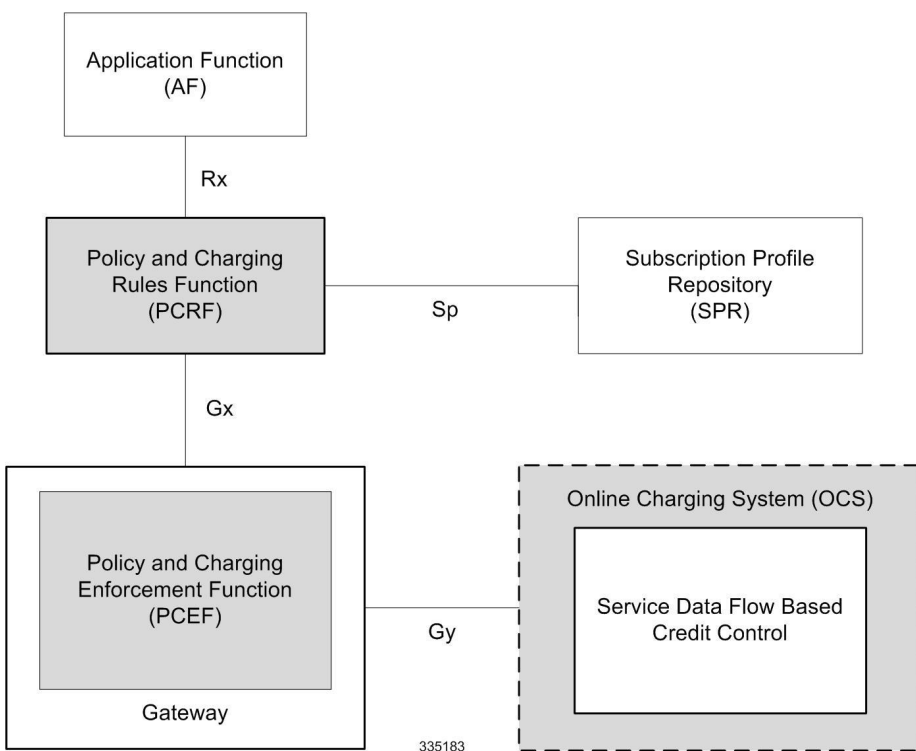
The Gy interface is the online charging interface between the PCEF/GW (Charging Trigger Function (CTF)) and the Online Charging System (Charging-Data-Function (CDF)).

The Gy interface makes use of the Active Charging Service (ACS) / Enhanced Charging Service (ECS) for real-time content-based charging of data services. It is based on the 3GPP standards and relies on quota allocation. The Online Charging System (OCS) is the Diameter Credit Control server, which provides the online charging data to the PCEF/GW. With Gy, customer traffic can be gated and billed in an online or prepaid style. Both time- and volume-based charging models are supported. In these models differentiated rates can be applied to different services based on ECS shallow- or deep-packet inspection.

In the simplest possible installation, the system will exchange Gy Diameter messages over Diameter TCP links between itself and one prepaid server. For a more robust installation, multiple servers would be used. These servers may optionally share or mirror a single quota database so as to support Gy session failover from one server to the other. For a more scalable installation, a layer of proxies or other Diameter agents can be introduced to provide features such as multi-path message routing or message and session redirection features.

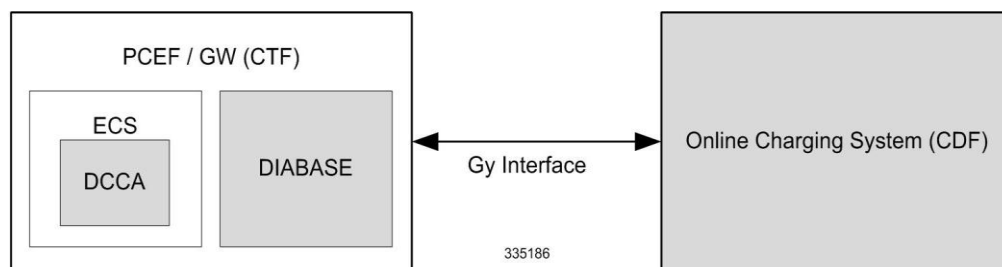
The following figure shows the Gy reference point in the policy and charging architecture.

Figure 32. PCC Logical Architecture



The following figure shows the Gy interface between CTF/Gateway/PCEF/Client running ECS and OCS (CDF/Server). Within the PCEF/GW, the Gy protocol functionality is handled in the DCCA module (at the ECS).

Figure 33. Gy Architecture



License Requirements

The Gy interface support is a licensed Cisco feature. A separate feature license may be required. Contact your Cisco account representative for detailed information on specific licensing requirements. For information on installing and verifying licenses, refer to the *Managing License Keys* section of the *Software Management Operations* chapter in the *System Administration Guide*.

Supported Standards

Gy interface support is based on the following standards:

- IETF RFC 4006: Diameter Credit Control Application; August 2005
- 3GPP TS 32.299 V9.6.0 (2010-12) 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Telecommunication management; Charging management; Diameter charging applications (Release 9)

Features and Terminology

This section describes features and terminology pertaining to Gy functionality.

Charging Scenarios



Important: Online charging for events (“Immediate Event Charging” and “Event Charging with Reservation”) is not supported. Only “Session Charging with Reservation” is supported.

Session Charging with Reservation

Session Charging with Unit Reservation is used for credit control of sessions.

Decentralized Unit Determination and Centralized Rating

In this scenario, the CTF requests the reservation of units prior to session supervision. An account debit operation is carried out following the conclusion of session termination.

Centralized Unit Determination and Centralized Rating

In this scenario, the CTF requests the OCS to reserve units based on the session identifiers specified by the CTF. An account debit operation is carried out following the conclusion of session.

Decentralized Unit Determination and Decentralized Rating



Important: Decentralized Rating is not supported in this release. Decentralized Unit determination is done using CLI configuration.

In this scenario, the CTF requests the OCS to assure the reservation of an amount of the specified number of monetary units from the subscriber's account. An account debit operation that triggers the deduction of the amount from the subscriber's account is carried out following the conclusion of session establishment.

Basic Operations



Important: Immediate Event Charging is not supported in this release. “Reserve Units Request” and “Reserve Units Response” are done for Session Charging and not for Event Charging.

Online credit control uses the basic logical operations “Debit Units” and “Reserve Units”.

- **Debit Units Request;** sent from CTF to OCS: After receiving a service request from the subscriber, the CTF sends a Debit Units Request to the OCS. The CTF may either specify a service identifier (centralised unit determination) or the number of units requested (decentralised unit determination). For refund purpose, the CTF sends a Debit Units Request to the OCS as well.

- Debit Units Response; sent from OCS to CTF: The OCS replies with a Debit Units Response, which informs the CTF of the number of units granted as a result of the Debit Units Request. This includes the case where the number of units granted indicates the permission to render the requested service. For refund purpose, the OCS replies with a Debit Units Response.
- Reserve Units Request; sent from CTF to OCS: Request to reserve a number of units for the service to be provided by an CTF. In case of centralised unit determination, the CTF specifies a service identifier in the Reserve Unit Request, and the OCS determines the number of units requested. In case of decentralised unit determination, the number of units requested is specified by the CTF.
- Reserve Units Response; sent from OCS to CTF: Response from the OCS which informs the CTF of the number of units that were reserved as a result of the “Reserve Units Request”.

Session Charging with Unit Reservation (SCUR) use both the “Debit Units” and “Reserve Units” operations. SCUR uses the Session Based Credit Control procedure specified in RFC 4006. In session charging with unit reservation, when the “Debit Units” and “Reserve Units” operations are both needed, they are combined in one message.



Important: Cost-Information, Remaining-Balance, and Low-Balance-Indication AVPs are not supported.

The consumed units are deducted from the subscriber's account after service delivery. Thus, the reserved and consumed units are not necessarily the same. Using this operation, it is also possible for the CTF to modify the current reservation, including the return of previously reserved units.

Re-authorization

The server may specify an idle timeout associated with a granted quota. Alternatively, the client may have a configurable default value. The expiry of that timer triggers a re-authorization request.

Mid-session service events (re-authorization triggers) may affect the rating of the current service usage. The server may instruct the credit control client to re-authorize the quota upon a number of different session related triggers that can affect the rating conditions.

When a re-authorization is trigger, the client reports quota usage. The reason for the quota being reported is notified to the server.

Threshold based Re-authorization Triggers

The server may optionally include an indication to the client of the remaining quota threshold that triggers a quota re-authorization.

Termination Action

The server may specify to the client the behavior on consumption of the final granted units; this is known as termination action.

Diameter Base Protocol

The Diameter Base Protocol maintains the underlying connection between the Diameter Client and the Diameter Server. The connection between the client and server is TCP based. There are a series of message exchanges to check the status of the connection and the capabilities.

- **Capabilities Exchange Messages:** Capabilities Exchange Messages are exchanged between the diameter peers to know the capabilities of each other and identity of each other.
 - **Capabilities Exchange Request (CER):** This message is sent from the client to the server to know the capabilities of the server.
 - **Capabilities Exchange Answer (CEA):** This message is sent from the server to the client in response to the CER message.



Important: Acct-Application-Id is not parsed and if sent will be ignored by the PCEF/GW. In case the Result-Code is not DIAMETER_SUCCESS, the connection to the peer is closed.

- **Device Watchdog Request (DWR):** After the CER/CEA messages are exchanged, if there is no more traffic between peers for a while, to monitor the health of the connection, DWR message is sent from the client. The Device Watchdog timer (Tw) is configurable in PCEF/GW and can vary from 6 through 30 seconds. A very low value will result in duplication of messages. The default value is 30 seconds. On two consecutive expiries of Tw without a DWA, the peer is taken to be down.



Important: DWR is sent only after Tw expiry after the last message that came from the server. Say if there is continuous exchange of messages between the peers, DWR might not be sent if (Current Time - Last message received time from server) is less than Tw.

- **Device Watchdog Answer (DWA):** This is the response to the DWR message from the server. This is used to monitor the connection state.
- **Disconnect Peer Request (DPR):** This message is sent to the peer to inform to shutdown the connection. PCEF/GW only receives this message. There is no capability currently to send the message to the diameter server.
- **Disconnect Peer Answer (DPA):** This message is the response to the DPR request from the peer. On receiving the DPR, the peer sends DPA and puts the connection state to “DO NOT WANT TO TALK TO YOU” state and there is no way to get the connection back except for reconfiguring the peer again.
A timeout value for retrying the disconnected peer must be provided.
- **Tw Timer Expiry Behavior:** The connection between the client and the server is taken care by the DIABASE application. When two consecutive Tw timers are expired, the peer state is set to idle and the connection is retried to be established. All the active sessions on the connection are then transferred to the secondary connection if one is configured. All new session activations are also tried on the secondary connection.

There is a connection timeout interval, which is also equivalent to Tw timer, wherein after a CER has been sent to the server, if there is no response received while trying to reestablish connection, the connection is closed and the state set to idle.

Diameter Credit Control Application

The Diameter Credit Control Application (DCCA) is a part of the ECS subsystem. For every prepaid customer with Diameter Credit Control enabled, whenever a session comes up, the Diameter server is contacted and quota for the subscriber is fetched.

Quota Behavior

Various forms of quotas are present that can be used to charge the subscriber in an efficient way. Various quota mechanisms provide the end user with a variety of options to choose from and better handling of quotas for the service provider.

Time Quotas

The Credit-Control server can send the CC-Time quota for the subscriber during any of the interrogation of client with it. There are also various mechanisms as discussed below which can be used in conjunction with time quota to derive variety of methods for customer satisfaction.

- **Quota Consumption Time:** The server can optionally indicate to the client that the quota consumption must be stopped after a period equal to the “Quota Consumption Time” in which no packets are received or at session termination, whichever is sooner. The idle period equal to the Quota Consumption Time is included in the reported usage. The quota is consumed normally during gaps in traffic of duration less than or equal to the Quota-Consumption-Time. Quota consumption resumes on receipt of a further packet belonging to the service data flow.

If packets are allowed to flow during a CCR (Update)/CCA exchange, and the Quota-Consumption-Time AVP value in the provided quota is the same as in the previously provided quota, then the Quota-Consumption-Time runs normally through this procedure. For example, if 5 seconds of a 10 second QCT timer have passed when a CCR(U) is triggered, and the CCA(U) returns 2 seconds later, then the QCT timer will expire 3 seconds after the receipt of the CCA and the remaining unaccounted 5 seconds of usage will be recorded against the new quota even though no packets were transmitted with the new quota.

A locally configurable default value in the client can be used if the server doesn't send the QCT in the CCA.

- **Combinational Quota:** Discrete-Time-Period (DTP) and Continuous-Time-Period (CTP) defines mechanisms that extends and generalize the Quota-Consumption-Time for consuming time-quota.
 - Both DTP and CTP uses a “base-time-interval” that is used to create time-envelopes of quota used.
 - Instead of consuming the quota linearly, DTP and CTP consumes the granted quota discretely in chunks of base-time-interval at the start of the each base-time-interval.
 - Selection of one of this algorithm is based on the “Time-Quota-Mechanism” AVP sent by the server in CCA.
 - Reporting usage can also be controlled by Envelope-Reporting AVP sent by the server in CCA during the quota grant. Based on the value of this AVP, the usage can be reported either as the usage per envelope or as usual cumulative usage for that grant.
- **Discrete-Time-Period:** The base-time-interval defines the length of the Discrete-Time-Period. So each time-envelope corresponds to exactly one Discrete-Time-Period. So when a traffic is detected, an envelope of size equal to Base-Time-Interval is created. The traffic is allowed to pass through the time-envelope. Once the traffic exceeds the base-time-interval another new envelope equal to the base-time-interval is created. This continues till the quota used exceeds the quota grant or reaches the threshold limit for that quota.
- **Continuous-Time-Period:** Continuous time period mechanism constructs time envelope out of consecutive base-time intervals in which the traffic occurred up to and including a base time interval which contains no traffic. Therefore the quota consumption continues within the time envelope, if there was traffic in the previous base time interval. After an envelope has closed, then the quota consumption resumes only on the first traffic following the closure of the envelope. The envelope for CTP includes the last base time interval which contains no traffic.

The size of the envelope is not constant as it was in Parking meter. The end of the envelope can only be determined retrospectively.

- **Quota Hold Time:** The server can specify an idle timeout associated with a granted quota using the Quota-Holding-Time AVP. If no traffic associated with the quota is observed for this time, the client understands that the traffic has stopped and the quota is returned to the server. The client starts the quota holding timer when quota consumption ceases. This is always when traffic ceases, i.e. the timer is re-started at the end of each packet. It applies equally to the granted time quota and to the granted volume quota. The timer is stopped on sending a CCR and re-initialized on receiving a CCA with the previous used value or a new value of Quota-Holding-Time if received.

Alternatively, if this AVP is not present, a locally configurable default value in the client is used. A Quota-Holding-Time value of zero indicates that this mechanism is not used.

- **Quota Validity Time:** The server can optionally send the validity time for the quota during the interrogation with the client. The Validity-Time AVP is present at the MSCC level and applies equally to the entire quota that is present in that category. The quota gets invalidated at the end of the validity time and a CCR-Update is sent to the server with the Used-Service-Units AVP and the reporting reason as VALIDITY_TIME. The entire quota present in that category will be invalidated upon Quota-Validity-Time expiry and traffic in that category will be passed or dropped depending on the configuration, till a CCA-Update is received with quota for that category.

Validity-Time of zero is invalid. Validity-Time is relative and not absolute.

Volume Quota

The server sends the CC-Total-Octets AVP to provide volume quota to the subscriber. DCCA currently supports only CC-Total-Octets AVP, which applies equally to uplink and downlink packets. If the total of uplink and downlink packets exceeds the CC-Total-Octets granted, the quota is assumed to be exhausted.

If CC-Input-Octets and/or CC-Output-Octets is provided, the quota is counted against CC-Input-Octets and/or CC-Output-Octets respectively.



Important: Restricting usages based on CC-Input-Octets and CC_Output-Octets is not supported in this release.

Units Quota

The server can also send a CC-Service-Specific-Units quota which is used to have packets counted as units. The number of units per packet is a configurable option.

Granting Quota

Gy implementation assumes that whenever the CC-Total-Octets AVP is present, volume quota has been granted for both uplink and downlink.

If the Granted-Service-Unit contains no data, Gy treats it as an invalid CCA.

If the values are zero, it is assumed that no quota was granted.

If the AVP contains the sub AVPs without any data, it is assumed to be infinite quota.

Additional parameters relating to a category like QHT, QCT is set for the category after receiving a valid volume or time grant.

If a default quota is configured for the subscriber, and subscriber traffic is received it is counted against the default quota. The default quota is applicable only to the initial request and is not regranted during the course of the session. If subscriber disconnects and reconnects, the default quota will be applied again for the initial request.

Requesting Quota

Quotas for a particular category type can be requested using the Requested-Service-Unit AVP in the CCR. The MSCC is filled with the Rating-Group AVP which corresponds to the category of the traffic and Requested-Service-Unit (RSU) AVP without any data.

The Requested-Service-Unit can contain the CC AVPs used for requesting specific quantity of time or volume grant. Gy CLI can be used to request quota for a category type.

Alternatively quota can also be requested from the server preemptively for a particular category in CCR-I. When the server grants preemptive quota through the Credit control answer response, the quota will be used only when traffic is hit for that category. Quota can be preemptively requested from the Credit Control server from the CLI.

In 12.3 and earlier releases, when no pre-emptive quota request is present in CCR-I, on hitting server unreachable state for initial request, MSCC AVP with RSU is present in the CCR-I on server retries. Release 14.0 onwards, the MSCC AVP is skipped in the CCR-I on server retries. Corresponding quota usage will be reported in the next CCR-U (MSCC AVP with USU and RSU).

Reporting Quota

Quotas are reported to the server for number of reasons including:

- Threshold
- QHT Expiry
- Quota Exhaustion
- Rating Condition Change
- Forced Reauthorization
- Validity Time Expiry
- Final during Termination of Category Instance from Server

For the above cases except for QHT and Final, the Requested-Service-Unit AVP is present in the CCR.

Reporting Reason is present in CCR to let the server know the reason for the reporting of Quota. The Reporting-Reason AVP can be present either in MSCC level or at Used Service Unit (USU) level depending on whether the reason applies to all quotas or to single quota.

When one of these conditions is met, a CCR Update is sent to the server containing a Multiple-Services-Credit-Control AVP(s) indicating the reason for reporting usage in the Reporting-Reason and the appropriate value(s) for Trigger, where appropriate. Where a threshold was reached, the DCCA still has the amount of quota available to it defined by the threshold.

For all other reporting reasons the client discards any remaining quota and either discards future user traffic matching this category or allows user traffic to pass, or buffers traffic according to configuration.

For Reporting-Reason of Rating Condition Change, Gy requires the Trigger Type AVP to be present as part of the CCR to indicate which trigger event caused the reporting and re-authorization request.

For Reporting-Reason of end user service denied, this happens when a category is blacklisted by the credit control server, in this case a CCR-U is sent with used service unit even if the values as zero. When more quota is received from the server for that particular category, the blacklisting is removed.

If a default quota has been set for the subscriber then the usage from the default quota is deducted from the initial GSU received for the subscriber for the Rating Group or Rating Group and Service ID combination.

Default Quota Handling

- If default quota is set to 0, no data is passed/reported.

- If default quota is configured and default quota is not exhausted before OCS responds with quota, traffic is passed. Initial default quota used is counted against initial quota allocated. If quota allocated is less than the actual usage then actual usage is reported and additional quota requested. If no additional quota is available then traffic is denied.
- If default quota is not exhausted before OCS responds with denial of quota, gateway blocks traffic after OCS response. Gateway will report usage on default quota even in this case in CCR-U (FINAL) or CCR-T.
- if default quota is consumed before OCS responds, if OCS is not declared dead (see definition in use case 1 above) then traffic is blocked until OCS responds.

Thresholds

The Gy client supports the following threshold types:

- Volume-Quota-Threshold
- Time-Quota-Threshold
- Units-Quota-Threshold

A threshold is always associated with a particular quota and a particular quota type. in the Multiple-Services-Credit-Control AVP, the Time-Quota-Threshold, Volume-Quota-Threshold, and Unit-Quota-Threshold are optional AVPs.

They are expressed as unsigned numbers and the units are seconds for time quota, octets for volume quota and units for service specific quota. Once the quota has reached its threshold, a request for more quotas is triggered toward the server. User traffic is still allowed to flow. There is no disruption of traffic as the user still has valid quota.

The Gy sends a CCR Update with a Multiple-Services-Credit-Control AVP containing usage reported in one or more User-Service-Unit AVPs, the Reporting-Reason set to THRESHOLD and the Requested-Service-Unit AVP without data.

When quota of more than one type has been assigned to a category, each with its own threshold, then the threshold is considered to be reached once one of the unit types has reached its threshold even if the other unit type has not been consumed.

When reporting volume quota, the DCCA always reports uplink and downlink separately using the CC-Input-Octets AVP and the CC-Output-Octets AVP, respectively.

On receipt of more quotas in the CCA the Gy discard any quota not yet consumed since sending the CCR. Thus the amount of quota now available for consumption is the new amount received less any quota that may have been consumed since last sending the CCR.

Conditions for Reauthorization of Quota

Quota is re-authorized/requested from the server in case of the following scenarios:

- Threshold is hit
- Quota is exhausted
- Validity time expiry
- Rating condition change:
 - Cellid change: Applicable only to GGSN and P-GW implementations.
 - LAC change: Applicable only to GGSN and P-GW implementations.
 - QoS change
 - RAT change

- SGSN/Serving-Node change: Applicable only to GGSN and P-GW implementations.

Discarding or Allowing or Buffering Traffic to Flow

Whenever Gy is waiting for CCA from the server, there is a possibility of traffic for that particular traffic type to be encountered in the Gy. The behavior of what needs to be done to the packet is determined by the configuration. Based on the configuration, the traffic is either allowed to pass or discarded or buffered while waiting for CCA from the server.

This behavior applies to all interrogation of client with server in the following cases:

- No quota present for that particular category
- Validity timer expiry for that category
- Quota exhausted for that category
- Forced Reauthorization from the server

In addition to allowing or discarding user traffic, there is an option available in case of quota exhausted or no quota circumstances to buffer the traffic. This typically happens when the server has been requested for more quota, but a valid quota response has not been received from the server, in this case the user traffic is buffered and on reception of valid quota response from the server the buffered traffic is allowed to pass through.

Procedures for Consumption of Time Quota

- QCT is zero: When QCT is deactivated, the consumption is on a wall-clock basis. The consumption is continuous even if there is no packet flow.
- QCT is active: When QCT is present in the CCA or locally configured for the session, then the consumption of quota is started only at the time of first packet arrival. The quota is consumed normally till last packet arrival plus QCT time and is passed till the next packet arrival.

If the QCT value is changed during intermediate interrogations, then the new QCT comes into effect from the time the CCA is received. For instance, if the QCT is deactivated in the CCA, then quota consumptions resume normally even without any packet flow. Or if the QCT is activated from deactivation, then the quota consumption resume only after receiving the first packet after CCA.

- QHT is zero: When QHT is deactivated, the user holds the quota indefinitely in case there is no further usage (for volume quota and with QCT for time quota). QHT is active between the CCA and the next CCR.
- QHT is non-zero: When QHT is present in CCA or locally configured for the session, then after a idle time of QHT, the quota is returned to the server by sending a CCR-Update and reporting usage of the quota. On receipt of CCR-U, the server does not grant quota. QHT timer is stopped on sending the CCR and is restarted only if QHT is present in the CCA.

QHT timer is reset every time a packet arrives.

Envelope Reporting

The server may determine the need for additional detailed reports identifying start time and end times of specific activity in addition to the standard quota management. The server controls this by sending a CCA with Envelope-Reporting AVP with the appropriate values. The DCCA client, on receiving the command, will monitor for traffic for a period of time controlled by the Quota-Consumption-Time AVP and report each period as a single envelope for each Quota-Consumption-Time expiry where there was traffic. The server may request envelope reports for just time or time and volume. Reporting the quota back to the server, is controlled by Envelope AVP with Envelope-Start-Time and Envelope-End-Time along with usage information.

Credit Control Request

Credit Control Request (CCR) is the message that is sent from the client to the server to request quota and authorization. CCR is sent before the establishment of MIP session, and at the termination of the MIP session. It can be sent during service delivery to request more quotas.

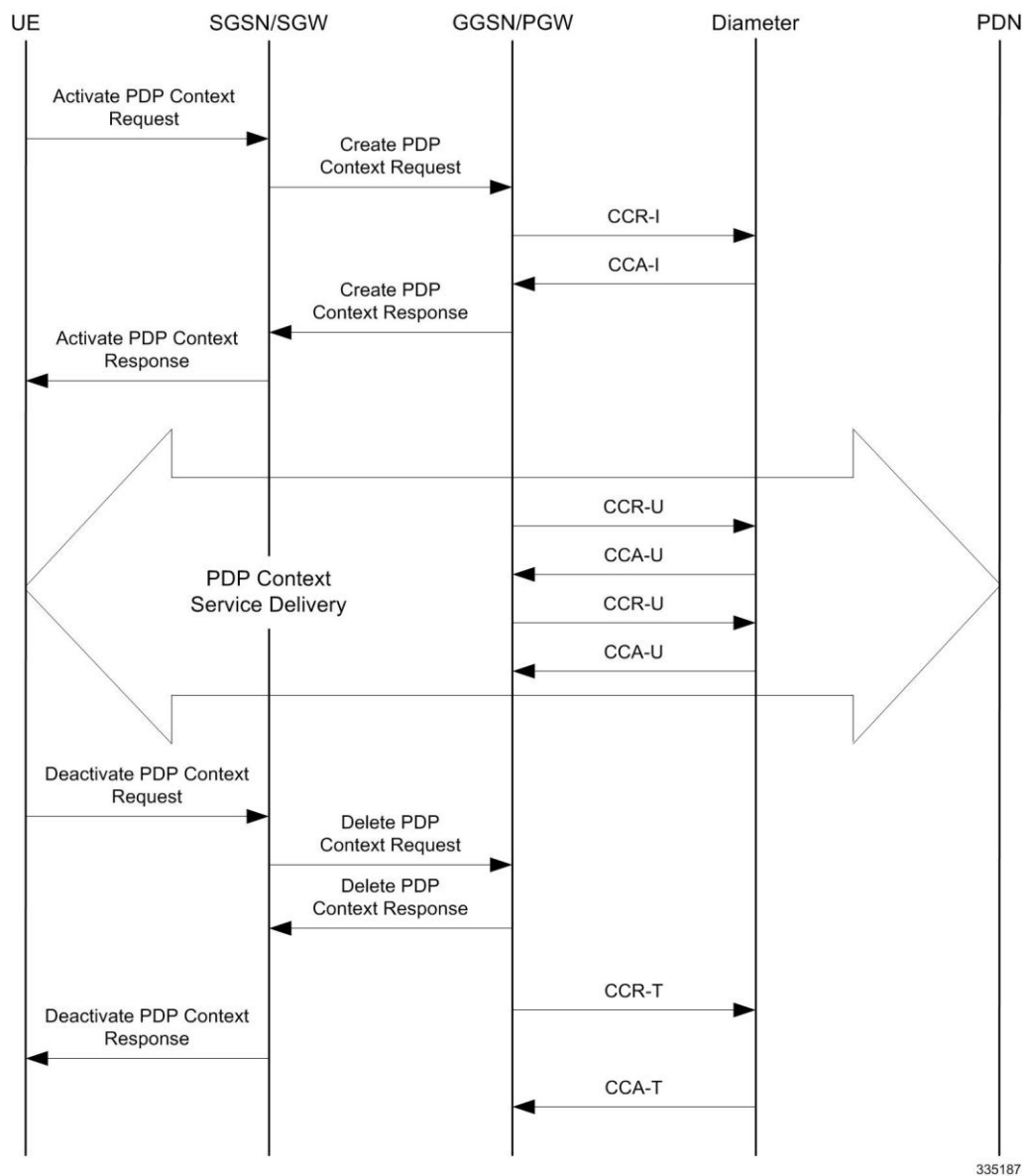
- Credit Control Request - Initial (CCR-I)
- Credit Control Request - Update (CCR-U)
- Credit Control Request - Terminate (CCR-T)
- Credit Control Answer (CCA)
- Credit Control Answer - Initial (CCA-I)
- Credit Control Answer - Update (CCA-U)

If the MSCC AVP is missing in CCA-Update it is treated as invalid CCA and the session is terminated.

- Credit Control Answer - Terminate (CCA-T)

The following figure depicts the call flow for a simple call request in the GGSN/P-GW/IPSG Gy implementation.

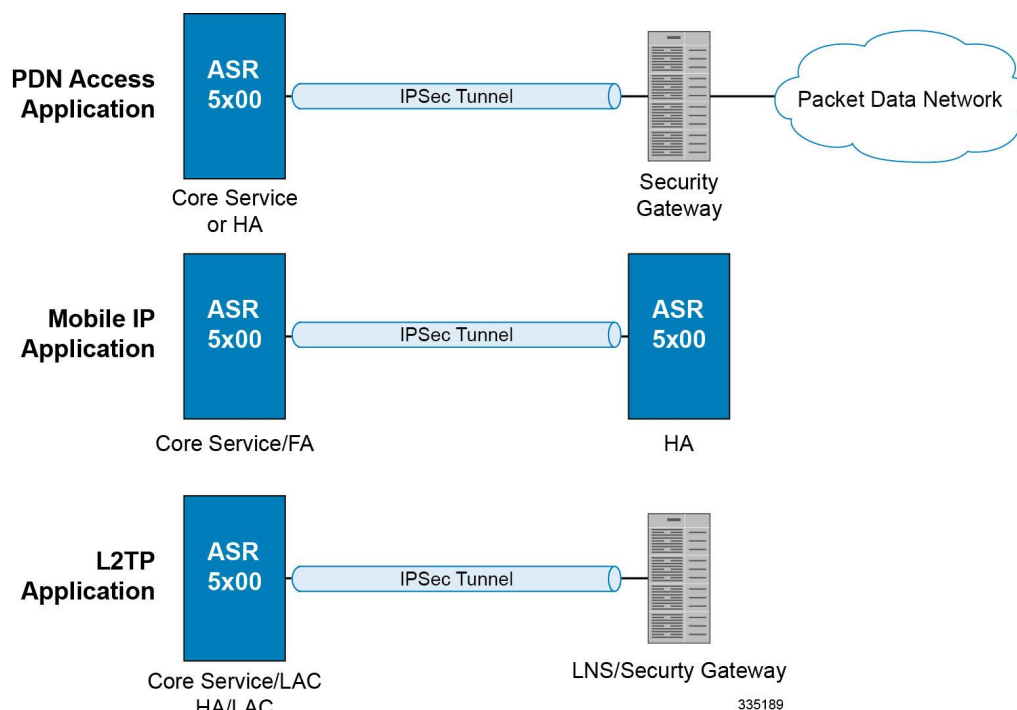
Figure 34. Gy Call Flow for Simple Call Request for GGSN/P-GW/IPSG



335187

The following figure depicts the call flow for a simple call request in the HA Gy implementation.

Figure 35. Gy Call Flow for Simple Call Request for HA



Tx Timer Expiry Behavior

A timer is started each time a CCR is sent out from the system, and the response has to arrive within Tx time. The timeout value is configurable in the Diameter Credit Control Configuration mode.

In case there is no response from the Diameter server for a particular CCR, within Tx time period, and if there is an alternate server configured, the CCR is sent to the alternate server after Tw expiry as described in “Tw Timer expiry behavior” section.

It also depends on the Credit-Control-Session-Failover AVP value for the earlier requests. If this AVP is present and is coded to `FAILOVER_SUPPORTED` then the credit-control message stream is moved to the secondary server, in case it is configured. If the AVP value is `FAILOVER_NOT_SUPPORTED`, then the call is dropped in case of failures, even if a secondary server is configured.

Redirection


In the Final-Unit-Indication AVP, if the Final-Action is `REDIRECT` or Redirect-Server AVP is present at command level, redirection is performed.

The redirection takes place at the end of consumption of quota of the specified category. The GY sends a CCR-Update without any RSU or Rating-Group AVP so that the server does not give any more quotas.

If the Final-Action AVP is `RESTRICT_ACCESS`, then according to the settings in Restriction-Filter-Rule AVP or Filter-Id AVP. GY sends CCR-Update to the server with used quota.

Triggers

The Diameter server can provide with the triggers for which the client should reauthorize a particular category. The triggers can be configured locally as well but whatever trigger is present in the CCA from the server will have precedence.

 **Important:** In this release, Gy triggers are not supported for HA.

The trigger types that are supported are:

- SGSN/Serving-Node Change
- QoS Change - Any
- RAT Change
- LAC Change
- CellID Change

On any event as described in the Trigger type happens, the client reauthorizes quota with the server. The reporting reason is set as `RATING_CONDITION_CHANGE`.

Tariff Time Change


The tariff change mechanism applies to each category instance active at the time of the tariff change whenever the server indicated it should apply for this category.

The concept of dual coupon is supported. Here the server grants two quotas, which is accompanied by a Tariff-Time-Change, in this case the first granted service unit is used until the tariff change time, once the tariff change time is reached the usage is reported up to the point and any additional usage is not accumulated, and then the second granted service unit is used.

If the server expects a tariff change to occur within the validity time of the quota it is granting, then it includes the Tariff-Time-Change AVP in the CCA. The DCCA report usage, which straddles the change time by sending two instances of the Used-Service-Unit AVP, one with Tariff-Change-Usage set to `UNIT_BEFORE_TARIFF_CHANGE`, and one with Tariff-Change-Usage set to `UNIT_AFTER_TARIFF_CHANGE`, and this independently of the type of units used by application. Both Volume and Time quota are reported in this way.

The Tariff time change functionality can as well be done using Validity-Time AVP, where in the Validity-Time is set to Tariff Time change and the client will reauthorize and get quota at Validity-Time expiry. This will trigger a lot of reauthorize request to the server at a particular time and hence is not advised.

Tariff-Time-Usage AVP along with the Tariff-Time-Change AVP in the answer message to the client indicates that the quotas defined in Multiple-Services-Credit-Control are to be used before or after the Tariff Time change. Two separate quotas are allocated one for before Tariff-Time-Change and one for after Tariff-Time-Change. This gives the flexibility to the operators to allocate different quotas to the users for different periods of time. In this case, the DCCA should not send the Before-Usage and After-Usage counts in the update messages to the server. When Tariff-Time-Change AVP is present without Tariff-Time-Usage AVP in the answer message, then the quota is used as in single quota mechanism and the client has to send before usage and after usage quotas in the updates to the server.

 **Important:** In this release, Gy does not support `UNIT_INDETERMINATE` value.

Final Unit Indication

The Final-Unit-Indication AVP can be present in the CCA from the server to indicate that the given quota is the final quota from the server and the corresponding action as specified in the AVP needs to be taken.

Final Unit Indication at Command Level

Gy currently does not support FUI AVP at command level. If this AVP is present at command level it is ignored. If the FUI AVP is present at command level and the Final-Unit-Action AVP set to TERMINATE, Gy sends a CCR-Terminate at the expiry of the quota, with all quotas in the USU AVP.



Important: FUI AVP at command level is only supported for Terminate action.

Final Unit Indication at MSCC Level

If the Final-Unit-Indication AVP is present at MSCC level, and if the Final-Unit-Action AVP is set to TERMINATE, a CCR-Update is sent at the expiry of the allotted quota and report the usage of the category that is terminated.

For information on redirection cases refer to Redirection section.

Credit Control Failure Handling

CCFH AVP defines what needs to be done in case of failure of any type between the client and the server. The CCFH functionality can be defined in configuration but if the CCFH AVP is present in the CCA, it takes precedence. CCFH AVP gives flexibility to have different failure handling.

Gy supports the following Failure Handling options:

- TERMINATE
- CONTINUE
- RETRY AND TERMINATE

CCFH with Failover Supported

In case there is a secondary server is configured and if the CC-Session-Failover AVP is set to FAILOVER_SUPPORTED, the following behavior takes place:

- Terminate: On any Tx expiry for the CCR-I the message is discarded and the session is torn down. In case of CCR-Updates and Terminates the message is sent to the secondary server after response timeout and the session is proceeded with the secondary server. In case there is a failure with the secondary server too, the session is torn down.
- Continue: On any Tx expiry, the message is sent to the secondary server after response timeout and the session is proceeded with the secondary server. In case there is a failure with the secondary server too, the session is still established, but without quota management.
- Retry and Terminate: On any Tx expiry, the message is sent to the secondary server after the response timeout. In case there is a failure with secondary server too, the session is taken down.

CCFH with Failover Not Supported

In case there is a secondary server configured and if the CC-Session-Failover AVP is set to FAILOVER_NOT_SUPPORTED, the following behavior takes place as listed below. Same is the case if there is no secondary server configured on the system.

- Terminate: On any Tx expiry, the session is taken down.
- Continue: On any Tx expiry, the session is still established, but without quota management.
- Retry and Terminate: On any Tx expiry, the session is taken down.

Failover Support

The CC-Session-Failover AVP and the Credit-Control-Failure-Handling (CCFH) AVP may be returned by the CC server in the CCA-I, and are used by the DCCA to manage the failover procedure. If they are present in the CCA they override the default values that are locally configured in the system.

If the CC-Session-Failover is set to `FAILOVER_NOT_SUPPORTED`, a CC session will never be moved to an alternative Diameter Server.

If the value of CC-Session-Failover is set to `FAILOVER_SUPPORTED`, then the Gy attempts to move the CC session to the alternative server when it considers a request to have failed, i.e:

- On receipt of result code “`DIAMETER_UNABLE_TO_DELIVER`”, “`DIAMETER_TOO_BUSY`”, or “`DIAMETER_LOOP_DETECTED`”.
- On expiry of the request timeout.
- On expiry of Tw without receipt of DWA, if the server is connected directly to the client.

The CCFH determines the behavior of the client in fault situations. If the Tx timer expires then based on the CCFH value the following actions are taken:

- **CONTINUE**: Allow the MIP session and user traffic for the relevant category or categories to continue, regardless of the interruption (delayed answer). Note that quota management of other categories is not affected.
- **TERMINATE**: Terminate the MIP session, which affects all categories.
- **RETRY_AND_TERMINATE**: Allow the MIP session and user traffic for the relevant category or categories to continue, regardless of the interruption (delayed answer). The client retries to send the CCR when it determines a failure-to-send condition and if this also fails, the MIP session is then terminated.

After the failover action has been attempted, and if there is still a failure to send or temporary error, depending on the CCFH action, the following action is taken:

- **CONTINUE**: Allow the MIP session to continue.
- **TERMINATE**: Terminate the MIP session.
- **RETRY_AND_TERMINATE**: Terminate the MIP session.

Recovery Mechanisms

DCCA supports a recovery mechanism that is used to recover sessions without much loss of data in case of Session Manager failures. There is a constant check pointing of Gy data at regular intervals and at important events like update, etc.

For more information on recovery mechanisms, please refer to the *System Administration Guide*.

Error Mechanisms

Unsupported AVPs

All unsupported AVPs from the server with “M” bit set are ignored.

Invalid Answer from Server

If there is an invalid answer from the server, Gy action is dependent on the CCFH setting:

- In case of continue, the MIP session context is continued without further control from Gy.

- In case of terminate and retry-and-terminate, the MIP session is terminated and a CCR-T is sent to the diameter server.

Result Code Behavior

- **DIAMETER_RATING_FAILED**: On reception of this code, Gy discards all traffic for that category and does not request any more quota from the server. This is supported at the MSCC level and not at the command level.
- **DIAMETER_END_USER_SERVICE_DENIED**: On reception of this code, Gy temporarily blacklists the category and further traffic results in requesting new quota from the server. This is supported at the MSCC level and not at the command level.
- **DIAMETER_CREDIT_LIMIT_REACHED**: On reception of this code, Gy discards all traffic for that category and waits for a configured time, after which if there is traffic for the same category requests quota from the server. This is supported at the MSCC level and not at the command level.
- **DIAMETER_CREDIT_CONTROL_NOT_APPLICABLE**: On reception of this code, Gy allows the session to establish, but without quota management. This is supported only at the command level and not at the MSCC level.
- **DIAMETER_USER_UNKNOWN**: On reception of this code, DCCA does not allow the credit control session to get established, the session is terminated. This result code is supported only at the command level and not at the MSCC level.

For all other permanent/transient failures, Gy action is dependent on the CCFH setting.

Supported AVPs

The Gy functionality supports the following AVPs:

- Supported Diameter Credit Control AVPs specified in RFC 4006:
 - **CC-Input-Octets (AVP Code: 412)**:
Gy supports this AVP only in USU.
 - **CC-Output-Octets (AVP Code: 414)**:
Gy supports this AVP only in USU.
 - **CC-Request-Number (AVP Code: 415)**
 - **CC-Request-Type (AVP Code: 416)**:
Gy currently does not support EVENT_REQUEST value.
 - **CC-Service-Specific-Units (AVP Code: 417)**
 - **CC-Session-Failover (AVP Code: 418)**
 - **CC-Time (AVP Code: 420)**:
Gy does not support this AVP in RSU.
 - **CC-Total-Octets (AVP Code: 421)**:
Gy does not support this AVP in RSU.
 - **Credit-Control-Failure-Handling (AVP Code: 427)**
 - **Final-Unit-Action (AVP Code: 449)**:
Supported at Multiple-Services-Credit-Control grouped AVP level and not at command level.

- Final-Unit-Indication (AVP Code: 430):
Fully supported at Multiple-Services-Credit-Control grouped AVP level and partially supported (TERMINATE) at command level.
- Granted-Service-Unit (AVP Code: 431)
- Multiple-Services-Credit-Control (AVP Code: 456)
- Multiple-Services-Indicator (AVP Code: 455)
- Rating-Group (AVP Code: 432)
- Redirect-Address-Type (AVP Code: 433):
Gy currently supports only URL (2) value.
- Redirect-Server (AVP Code: 434)
- Redirect-Server-Address (AVP Code: 435)
- Requested-Service-Unit (AVP Code: 437)
- Result-Code (AVP Code: 268)
- Service-Context-Id (AVP Code: 461)
- Service-Identifier (AVP Code: 439)
- Subscription-Id (AVP Code: 443)
- Subscription-Id-Data (AVP Code: 444)
- Subscription-Id-Type (AVP Code: 450)
- Tariff-Change-Usage (AVP Code: 452):
Gy does NOT support UNIT_INDETERMINATE (2) value.
- Tariff-Time-Change (AVP Code: 451)
- Used-Service-Unit (AVP Code: 446):
Gy sends only incremental counts for all the AVPs from the last CCA-U.
- User-Equipment-Info (AVP Code: 458)
- User-Equipment-Info-Type (AVP Code: 459):
Gy currently supports only IMEISV value.
Cisco GGSN and P-GW support IMEISV by default.
- User-Equipment-Info-Value (AVP Code: 460)
- Validity-Time (AVP Code: 448)
- Supported 3GPP specific AVPs specified in 3GPP TS 32.299:
 - 3GPP-Charging-Characteristics (AVP Code: 13)
 - 3GPP-Charging-Id (AVP Code: 2)
 - 3GPP-GGSN-MCC-MNC (AVP Code: 9)
 - 3GPP-GPRS-QoS-Negotiated-Profile (AVP Code: 5)
 - 3GPP-IMSI-MCC-MNC (AVP Code: 8)
 - 3GPP-NSAPI (AVP Code: 10)
 - 3GPP-PDP-Type (AVP Code: 3)

- 3GPP-RAT-Type (AVP Code: 21)
- 3GPP-Selection-Mode (AVP Code: 12)
- 3GPP-Session-Stop-Indicator (AVP Code: 11)
- 3GPP-SGSN-MCC-MNC (AVP Code: 18)
- 3GPP-User-Location-Info (AVP Code: 22)
- Base-Time-Interval (AVP Code: 1265)
- Charging-Rule-Base-Name (AVP Code: 1004)
- Envelope (AVP Code: 1266)
- Envelope-End-Time (AVP Code: 1267)
- Envelope-Reporting (AVP Code: 1268)
- Envelope-Start-Time (AVP Code: 1269)
- GGSN-Address (AVP Code: 847)
- Offline-Charging (AVP Code: 1278)
- PDP-Address (AVP Code: 1227)
- PDP-Context-Type (AVP Code: 1247)

This AVP is present only in CCR-I.

- PS-Information (AVP Code: 874)
- Quota-Consumption-Time (AVP Code: 881):

This optional AVP is present only in CCA.

- Quota-Holding-Time (AVP Code: 871):

This optional AVP is present only in the CCA command. It is contained in the Multiple-Services-Credit-Control AVP. It applies equally to the granted time quota and to the granted volume quota.

- Reporting-Reason (AVP Code: 872):

Gy currently does not support the POOL_EXHAUSTED (8) value. It is used in case of credit-pooling which is currently not supported.

- Service-Information (AVP Code: 873):

Only PS-Information is supported.

- SGSN-Address (AVP Code: 1228)
- Time-Quota-Mechanism (AVP Code: 1270):

The Gy server may include this AVP in an Multiple-Services-Credit-Control AVP when granting time quota.

- Time-Quota-Threshold (AVP Code: 868)
- Time-Quota-Type (AVP Code: 1271)
- Trigger (AVP Code: 1264)
- Trigger-Type (AVP Code: 870)
- Unit-Quota-Threshold (AVP Code: 1226)
- Volume-Quota-Threshold (AVP Code: 869)

- Supported Diameter AVPs specified in 3GPP TS 32.299 V8.1.0:

- Auth-Application-Id (AVP Code: 258)
- Destination-Host (AVP Code: 293)
- Destination-Realm (AVP Code: 283)
- Disconnect-Cause (AVP Code: 273)
- Error-Message (AVP Code: 281)
- Event-Timestamp (AVP Code: 55)
- Failed-AVP (AVP Code: 279)
- Multiple-Services-Credit-Control (AVP Code: 456)
- Origin-Host (AVP Code: 264)
- Origin-Realm (AVP Code: 296)
- Origin-State-Id (AVP Code: 278)
- Redirect-Host (AVP Code: 292)
- Redirect-Host-Usage (AVP Code: 261)
- Redirect-Max-Cache-Time (AVP Code: 262)
- Rating-Group (AVP Code: 432)
- Result-Code (AVP Code: 268)
- Route-Record (AVP Code: 282)
- Session-Id (AVP Code: 263)
- Service-Context-Id (AVP Code: 461)
- Service-Identifier (AVP Code: 439)
- Supported-Vendor-Id (AVP Code: 265)
- Termination-Cause (AVP Code: 295)
- Used-Service-Unit (AVP Code: 446)
- User-Name (AVP Code: 1)

Unsupported AVPs

This section lists the AVPs that are NOT supported.

- NOT Supported Credit Control AVPs specified in RFC 4006:
 - CC-Correlation-Id
 - CC-Money
 - CC-Sub-Session-Id
 - CC-Unit-Type (AVP Code: 454)
 - Check-Balance-Result
 - Cost-Information (AVP Code: 423)
 - Cost-Unit (AVP Code: 445)
 - Credit-Control
 - Currency-Code (AVP Code: 425)

- Direct-Debiting-Failure-Handling (AVP Code: 428)
- Exponent (AVP Code: 429)
- G-S-U-Pool-Identifier (AVP Code: 453)
- G-S-U-Pool-Reference (AVP Code: 457)
- Requested-Action (AVP Code: 436)
- Service-Parameter-Info (AVP Code: 440)
- Service-Parameter-Type (AVP Code: 441)
- Service-Parameter-Value (AVP Code: 442)
- Unit-Value (AVP Code: 424)
- Value-Digits (AVP Code: 447)
- NOT supported Diameter AVPs specified in 3GPP TS 32.299 V8.1.0:
 - Acct-Application-Id (AVP Code: 259)
 - Error-Reporting-Host (AVP Code: 294)
 - Experimental-Result (AVP Code: 297)
 - Experimental-Result-Code (AVP Code: 298)
 - Proxy-Host
 - Proxy-Info
 - Proxy-State
- NOT supported 3GPP-specific AVPs specified in 3GPP TS 32.299 V8.1.0:
 - 3GPP-CAMEL-Charging-Info (AVP Code: 24)
 - 3GPP-MS-TimeZone (AVP Code: 23)
 - 3GPP-PDSN-MCC-MNC
 - Authorised-QoS
 - Access-Network-Information
 - Adaptations
 - Additional-Content-Information
 - Additional-Type-Information
 - Address-Data
 - Address-Domain
 - Addressee-Type
 - Address-Type
 - AF-Correlation-Information
 - Alternate-Charged-Party-Address
 - Application-provided-Called-Party-Address
 - Application-Server

- Application-Server-Information
- Applic-ID
- Associated-URI
- Aux-Applic-Info
- Bearer-Service
- Called-Asserted-Identity
- Called-Party-Address
- Calling-Party-Address
- Cause-Code
- Charged-Party
- Class-Identifier
- Content-Class
- Content-Disposition
- Content-Length
- Content-Size
- Content-Type
- Data-Coding-Scheme
- Deferred-Location-Event-Type
- Delivery-Report-Requested
- Destination-Interface
- Domain-Name
- DRM-Content
- Early-Media-Description
- Event
- Event-Type
- Expires
- File-Repair-Supported
- IM-Information
- IMS-Charging-Identifier (ICID)
- IMS-Communication-Service-Identifier
- IMS-Information
- Incoming-Trunk-Group-ID
- Interface-Id
- Interface-Port

- Interface-Text
- Interface-Type
- Inter-Operator-Identifier
- LCS-APN
- LCS-Client-Dialed-By-MS
- LCS-Client-External-ID
- LCS-Client-ID
- LCS-Client-Name
- LCS-Client-Type
- LCS-Data-Coding-Scheme
- LCS-Format-Indicator
- LCS-Information
- LCS-Name-String
- LCS-Requestor-ID
- LCS-Requestor-ID-String
- Location-Estimate
- Location-Estimate-Type
- Location-Type
- Low-Balance-Indication
- MBMS-Information
- MBMS-User-Service-Type
- Media-Initiator-Flag
- Media-Initiator-Party
- Message-Body
- Message-Class
- Message-ID
- Message-Size
- Message-Type
- MMBox-Storage-Requested
- MM-Content-Type
- MMS-Information
- Node-Functionality
- Number-Of-Participants
- Number-Of-Received-Talk-Bursts

- Number-Of-Talk-Bursts
- Originating-IOI
- Originator
- Originator-Address
- Originator-Interface
- Originator-SCCP-Address
- Outgoing-Trunk-Group-ID
- Participant-Access-Priority
- Participants-Group
- Participants-Involved
- PDG-Address
- PDG-Charging-Id
- PoC-Change-Condition
- PoC-Change-Time
- PoC-Controlling-Address
- PoC-Group-Name
- PoC-Information
- PoC-Server-Role
- PoC-Session-Id
- PoC-Session-Initiation-Type
- PoC-Session-Type
- PoC-User-Role
- PoC-User-Role-IDs
- PoC-User-Role-info-Units
- Positioning-Data
- Priority
- PS-Append-Free-Format-Data (AVP Code: 867):
 - The PCEF/GW ignores this AVP if no PS free format data is stored for the online charging session.
- PS-Free-Format-Data (AVP Code: 866)
- PS-Furnish-Charging-Information (AVP Code: 865)
- RAI (AVP Code: 909)
- Read-Reply-Report-Requested
- Received-Talk-Burst-Time
- Received-Talk-Burst-Volume
- Recipient-Address

- Recipient-SCCP-Address
- Refund-Information
- Remaining-Balance
- Reply-Applic-ID
- Reply-Path-Requested
- Requested-Party-Address
- Role-of-node
- SDP-Answer-Timestamp
- SDP-Media-Component
- SDP-Media-Description
- SDP-Media-Name
- SDP-Offer-Timestamp
- SDP-Session-Description
- SDP-TimeStamp
- Served-Party-IP-Address
- Service-Generic-Information
- Service-ID
- Service-Specific-Data
- Service-Specific-Info
- Service-Specific-Type
- SIP-Method
- SIP-Request-Timestamp
- SIP-Response-Timestamp
- SM-Discharge-Time
- SM-Message-Type
- SM-Protocol-Id
- SMSC-Address
- SMS-Information
- SMS-Node
- SM-Status
- SM-User-Data-Header
- Submission-Time
- Talk-Burst-Exchange
- Talk-Burst-Time

- Talk-Burst-Volume
- Terminating-IOI
- Time-Stamps
- Token-Text
- Trunk-Group-ID
- Type-Number
- User-Participating-Type
- User-Session-ID
- WAG-Address
- WAG-PLMN-Id
- WLAN-Information
- WLAN-Radio-Container
- WLAN-Session-Id
- WLAN-Technology
- WLAN-UE-Local-IPAddress

Configuring Gy Interface Support

To configure Gy interface support:

1. Configure the core network service as described in this Administration Guide.
2. Configure Gy interface support as described in the relevant section:
 - [Configuring GGSN / P-GW / IPSG Gy Interface Support](#)
 - [Configuring HA / PDSN Gy Interface Support](#)
3. Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.



Important: Commands used in the configuration examples in this section provide base functionality to the extent that the most common or likely commands and/or keyword options are presented. In many cases, other optional commands and/or keyword options are available. Refer to the *Command Line Interface Reference* for complete information regarding all commands.

Configuring GGSN / P-GW / IPSG Gy Interface Support

To configure the standard Gy interface support for GGSN/P-GW/IPSG, use the following configuration:

```
configure

context <context_name>

    diameter endpoint <endpoint_name>

        origin realm <realm>

        origin host <diameter_host> address <ip_address>

        peer <peer> realm <realm> address <ip_address>

    exit

exit

active-charging service <ecs_service_name>

    credit-control [ group <cc_group_name> ]

        diameter origin endpoint <endpoint_name>

        diameter peer-select peer <peer> realm <realm>

        diameter pending-timeout <timeout_period>

        diameter session failover
```



```

diameter dictionary <dictionary>

failure-handling initial-request continue

failure-handling update-request continue

failure-handling terminate-request continue

exit

exit

context <context_name>

  apn <apn_name>

    selection-mode sent-by-ms

    ims-auth-service <service>

    ip access-group <access_list_name> in

    ip access-group <access_list_name> out

    ip context-name <context_name>

    active-charging rulebase <rulebase_name>

    credit-control-group <cc_group_name>

  end

```

Notes:

- For information on configuring IP access lists, refer to the *Access Control Lists* chapter in the *System Administration Guide*.
- For more information on configuring ECS ruledefs, refer to the *ACS Ruledef Configuration Mode Commands* chapter in the *Command Line Interface Reference*.
- For more information on configuring ECS charging actions, refer to the *ACS Charging Action Configuration Mode Commands* chapter in the *Command Line Interface Reference*.
- For more information on configuring ECS rulebases, refer to the *ACS Rulebase Configuration Mode Commands* chapter in the *Command Line Interface Reference*.

Configuring HA / PDSN Gy Interface Support

To configure HA / PDSN Gy interface support, use the following configuration:

```

configure

context <context_name>

  diameter endpoint <endpoint_name>

```

```

    origin realm <realm>

    origin host <diameter_host> address <ip_address>

    peer <peer> realm <realm> address <ip_address>

    exit

exit

active-charging service <ecs_service_name>

    ruledef <ruledef_name>

        ip any-match = TRUE

        exit

    charging-action <charging_action_name>

        content-id <content_id>

        cca charging credit rating-group <rating_group>

        exit

    rulebase <rulebase_name>

        action priority <action_priority> ruledef <ruledef_name> charging-action
        <charging_action_name>

        exit

    credit-control [ group <cc_group_name> ]

        diameter origin endpoint <endpoint_name>

        diameter peer-select peer <peer> realm <realm>

        diameter pending-timeout <timeout>

        diameter session failover

        diameter dictionary <dictionary>

        failure-handling initial-request continue

        failure-handling update-request continue

        failure-handling terminate-request continue

        pending-traffic-treatment noquota buffer

        pending-traffic-treatment quota-exhausted buffer

        exit

```

```

exit

context <context_name>

    subscriber default

        ip access-group <acl_name> in

        ip access-group <acl_name> out

        ip context-name <context_name>

        active-charging rulebase <rulebase_name>

        credit-control-group <cc_group_name>

    end

```

Notes:

- For information on configuring IP access lists, refer to the *Access Control Lists* chapter in the *Systems Administration Guide*.
- For more information on configuring ECS ruledefs, refer to the *ACS Ruledef Configuration Mode Commands* chapter in the *Command Line Interface Reference*.
- For more information on configuring ECS charging actions, refer to the *ACS Charging Action Configuration Mode Commands* chapter in the *Command Line Interface Reference*.
- For more information on configuring ECS rulebases, refer to the *ACS Rulebase Configuration Mode Commands* chapter in the *Command Line Interface Reference*.

Gathering Statistics

This section explains how to gather Gy related statistics and configuration information.

In the following table, the first column lists what statistics to gather, and the second column lists the action to perform.

Statistics/Information	Action to perform
Complete statistics for ECS sessions.	show active-charging sessions full
Detailed information for the Active Charging Service (ACS)	show active-charging service all
Information on all rule definitions configured in the service.	show active-charging ruledef all
Information on all charging actions configured in the service.	show active-charging charging-action all
Information on all rulebases configured in the service.	show active-charging rulebase all
Statistics of the Credit Control application, DCCA.	show active-charging credit-control statistics
States of the Credit Control application's sessions, DCCA.	show active-charging credit-control session-states [rulebase <rulebase_name>] [content-id <content_id>]

Appendix I

IP Header Compression

This chapter provides information on configuring an enhanced, or extended, service. The product administration guides provide examples and procedures for configuration of basic services on the system. It is recommended that you select the configuration example that best meets your service model, and configure the required elements for that model, as described in the respective product administration guide, before using the procedures in this chapter.



Important: RoHC header compression is not applicable for SGSN and GGSN services.

This chapter includes the following procedures:

- [Configuring VJ Header Compression for PPP](#)
- [Configuring RoHC Header Compression for PPP](#)
- [Configuring Both RoHC and VJ Header Compression](#)
- [Configuring RoHC for Use with SO67 in PDSN or HSGW Service](#)
- [Using an RoHC Profile for Subscriber Sessions](#)
- [Disabling VJ Header Compression Over PPP](#)
- [Disabling RoHC Header Compression Over SO67](#)
- [Checking IP Header Compression Statistics](#)
- [RADIUS Attributes for IP Header Compression](#)

Overview

The system supports IP header compression on the PPP tunnels established over the EVDO-RevA A10 links and also over the GRE tunnel that is connected to the PCF to support EVDO-RevA Service Option 67 (SO67).

By default IP header compression using the VJ algorithm is enabled for subscribers using PPP.

Note that you can use the default VJ header compression algorithm alone, configure the use of RoHC header compression only, or use both VJ and RoHC IP header compression.

- **Van Jacobsen (VJ)** - The RFC 1144 (CTCP) header compression standard was developed by V. Jacobson in 1990. It is commonly known as VJ compression. It describes a basic method for compressing the headers of IPv4/TCP packets to improve performance over low speed serial links.
- **RObust Header Compression (RoHC)** - The RFC 3095 (RoHC) standard was developed in 2001. This standard can compress IP/UDP/RTP headers to just over one byte, even in the presence of severe channel impairments. This compression scheme can also compress IP/UDP and IP/ESP packet flows. RoHC is intended for use in wireless radio network equipment and mobile terminals to decrease header overhead, reduce packet loss, improve interactive response, and increase security over low-speed, noisy wireless links.



Important: The RoHC is a licensed Cisco feature. A separate feature license may be required. Contact your Cisco account representative for detailed information on specific licensing requirements. For information on installing and verifying licenses, refer to the *Managing License Keys* section of the *Software Management Operations* chapter in the *System Administration Guide*.

In addition, you can configure RoHC profiles that define RoHC Compressor and Decompressor parameters. These RoHC profiles can be applied to subscribers.

You can also turn off all IP header compression for a subscriber.

The procedures in this chapter describe how to configure the IP header compression methods used, but for RoHC over PPP the Internet Protocol Control Protocol (IPCP) negotiations determine when they are used.


Implementing IP header compression provides the following benefits:

- Improves interactive response time
- Allows the use of small packets for bulk data with good line efficiency
- Allows the use of small packets for delay sensitive low data-rate traffic
- Decreases header overhead.
- Reduces packet loss rate over lossy links.

Configuring VJ Header Compression for PPP

By default, VJ IP header compression is enabled for subscriber sessions. When VJ header compression is configured all IP headers are compressed using the VJ compression algorithm.

Note that procedure described in this section is applicable only when VJ header compression is disabled.

 **Important:** This section provides the minimum instruction set for configuring subscriber profile for header compression. For more information on commands that configure additional parameters and options, refer *Subscriber Configuration Mode Commands* chapter in *Command Line Interface Reference* .

To configure the system to enable VJ header compression to IP headers:

- Step 1** Enable VJ header compression by applying the example configuration in the [Enabling VJ Header Compression](#) section.
- Step 2** Verify your VJ header compression configuration by following the steps in the [Verifying the VJ Header Compression Configuration](#) section.
- Step 3** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Enabling VJ Header Compression

Use the following example to enable the VJ header compression over PPP:

```
configure

context <ctxt_name>

    subscriber name <subs_name>

        ip header-compression vj

    end
```

Notes:

- *<ctxt_name>* is the system context in which you wish to configure the subscriber profile. Typically this is an AAA context.
- *<subs_name>* is the name of the subscriber in the current context that you want to enable VJ IP header compression for.

Verifying the VJ Header Compression Configuration

These instructions are used to verify the VJ header compression configuration.


- Step 1** Verify that your header compression configurations for subscriber by entering the following command in Exec Mode in specific context:

```
show subscriber configuration username subs_name
```

The output of this command is a concise listing of subscriber parameter settings as configured.

Configuring RoHC Header Compression for PPP

RoHC IP header compression can be configured for all IP traffic, uplink traffic only, or downlink traffic only. When RoHC is configured for all traffic, you can specify the mode in which RoHC is applied.

 **Important:** This section provides the minimum instruction set for configuring subscriber profile for header compression. For more information on commands that configure additional parameters and options, refer *Subscriber Configuration Mode Commands* chapter in the *Command Line Interface Reference*.

To configure the system to enable RoHC header compression to IP headers:

- Enable RoHC header compression by applying the example configuration in the [Enabling RoHC Header Compression for PPP](#) section.
- Verify your RoHC header compression configuration by following the steps in the [Verifying the Header Compression Configuration](#) section.
- Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Enabling RoHC Header Compression for PPP

Use the following example to enable the RoHC over PPP:

```
configure

context <ctxt_name>

    subscriber name <subs_name>

        ip header-compression RoHC [ any [ mode { optimistic | reliable | unidirectional
} ] | cid-mode { { large | small } [ marked-flows-only | max-cid | max-hdr <value> | mrru
<value> ] } | marked flows-only | max-hdr <value> | mrru <value> | downlink | uplink ] ]+

    end
```

Notes:

- *<ctxt_name>* is the system context in which you wish to configure the subscriber profile. Typically this is an AAA context.
- *<subs_name>* is the name of the subscriber in the current context that you want to enable RoHC header compression for.
- Refer to the *Subscriber Configuration Mode Commands* chapter in *Command Line Interface Reference* for more details on this command and its options.

Verifying the Header Compression Configuration

These instructions are used to verify the header compression configuration.


- Step 1** Verify that your header compression configurations for subscriber by entering the following command in Exec Mode in specific context:


```
show subscriber configuration username subs_name
```

The output of this command is a concise listing of subscriber parameter settings as configured.

Configuring Both RoHC and VJ Header Compression

You can configure the system to use both VJ and RoHC IP header compression. When both VJ and RoHC are specified, the optimum header compression algorithm for the type of data being transferred is used for data in the downlink direction.

 **Important:** If both RoHC and VJ header compression are specified, the optimum header compression algorithm for the type of data being transferred is used for data in the downlink direction.

 **Important:** This section provides the minimum instruction set for configuring subscriber profile for header compression. For more information on commands that configure additional parameters and options, refer *Subscriber Configuration Mode Commands* chapter in the *Command Line Interface Reference*.

To configure the system to enable both RoHC and VJ header compression to IP headers:

- Enable the RoHC and VJ header compression by applying the example configuration in the [Enabling RoHC and VJ Header Compression for PPP](#) section.
- Verify your RoHC and VJ header compression configuration by following the steps in the [Verifying the Header Compression Configuration](#) section.
- Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Enabling RoHC and VJ Header Compression for PPP

Use the following example to enable the header compression over PPP:

```
configure

context <ctxt_name>

    subscriber name <subs_name>

        ip header-compression vj RoHC [ any [ mode { optimistic | reliable |
unidirectional } ] | cid-mode { { large | small } [ marked-flows-only | max-cid | max-hdr
<value> | mrru <value> ] } | marked flows-only | max-hdr <value> | mrru <value> |
downlink | uplink ] ]+

    end
```

Notes:

- *<ctxt_name>* is the system context in which you wish to configure the subscriber profile. Typically this is an AAA context.
- *<subs_name>* is the name of the subscriber in the current context that you want to enable RoHC header compression for.

- Refer to the Subscriber Configuration Mode Commands chapter in Command Line Interface Reference for more details on this command and its options.

Verifying the Header Compression Configuration

These instructions are used to verify the header compression configuration.


- Step 1** Verify that your header compression configurations for subscriber by entering the following command in Exec Mode in specific context:

```
show subscriber configuration username subs_name
```

The output of this command is a concise listing of subscriber parameter settings as configured.

Configuring RoHC for Use with SO67 in PDSN or HSGW Service

This section explains how to set RoHC settings in the PDSN or HSGW Service configuration mode. These settings are transferred to the PCF during the initial A11 setup and are used for the GRE tunnel that is connected to the PCF to support EVDO-RevA Service Option 67 (SO67). RoHC is enabled through an auxiliary SO67 A10 connection and the PCF signals this information when the auxiliary A10 is connected.

 **Important:** This section provides the minimum instruction set for configuring subscriber profile for header compression. For more information on commands that configure additional parameters and options, refer *PDSN Service Configuration Mode Commands* or *HSGW Service Configuration Mode Commands* chapter in *Command Line Interface Reference*.

To configure the system to enable the RoHC header compression feature at the PDSN or HSGW Service over SO67:

- Step 1** Enable header compression by applying the example configuration in the [Enabling ROHC Header Compression with PDSN](#) or [Enabling ROHC Header Compression with HSGW](#) section.
- Step 2** Verify your RoHC configuration by following the steps in the [Verifying the Header Compression Configuration](#) section.
- Step 3** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Enabling RoHC Header Compression with PDSN

Use the following example to enable the RoHC header compression with PDSN over SO67:

```
configure

context <ctxt_name>

    pdsn-service <svc_name>

        ip header-compression rohc

        cid-mode {large | small} max-cid integer

        mrru <num_octets>

        profile { [esp-ip] [rtp-udp] [udp-ip] [uncompressed-ip] }          end
```

Notes:

- *<ctxt_name>* is the system context in which PDSN service is configured and you wish to configure the service profile.
- *<svc_name>* is the name of the PDSN service in which you want to enable RoHC over SO67.
- Refer to the *PDSN Service RoHC Configuration Mode Commands* chapter in *Command Line Interface Reference* for more details on this command and its options.

Enabling RoHC Header Compression with HSGW

Use the following example to enable the RoHC header compression with HSGW over SO67:

```
configure

context <ctxt_name>

    hsgw-service <svc_name>

        ip header-compression rohc

            cid-mode {large | small} max-cid integer

            mrru <num_octets>

            profile { [esp-ip] [rtp-udp] [udp-ip] [uncompressed-ip] }

        end
```

Notes:

- <ctxt_name> is the system context in which HSGW service is configured and you wish to configure the service profile.
- <svc_name> is the name of the HSGW service in which you want to enable RoHC over SO67.
- Refer to the *HSGW Service RoHC Configuration Mode Commands* chapter in *Command Line Interface Reference* for more details on this command and its options.

Verifying the Header Compression Configuration

These instructions are used to verify the header compression configuration.


- Step 1** Verify that your header compression configurations for subscriber by entering the following command in Exec Mode in specific context:

```
show configuration context ctxt_name
```

The output of this command is a concise listing of subscriber parameter settings as configured.

Using an RoHC Profile for Subscriber Sessions

You can configure RoHC profiles that specify numerous compressor and decompressor settings. These profiles can in turn be applied to a specific subscriber or the default subscriber. RoHC profiles are used for both RoHC over PPP and for RoHC over SO67.

 **Important:** This section provides the minimum instruction set for configuring subscriber profile for header compression. For more information on commands that configure additional parameters and options, refer *Subscriber Configuration Mode Commands* chapter in *Command Line Interface Reference*.

To configure the system to apply RoHC profile to a subscriber session:

- Step 1** Create RoHC profile using decompression mode or decompression mode. If you want to use compression mode go to step a else follow step b:
- Step a** Configure RoHC profile by applying the example configuration in the [Creating ROHC Profile for Subscriber using Compression Mode](#) section using compression mode.
 - Step b** Alternatively configure RoHC profile by applying the example configuration in the [Creating ROHC Profile for Subscriber using Decompression Mode](#) section using compression mode.
- Step 2** Apply existing RoHC profile to a subscriber by applying the example configuration in the [Applying ROHC Profile to a Subscriber](#) section.
- Step 3** Verify your RoHC header compression configuration by following the steps in the [Verifying the Header Compression Configuration](#) section.
- Step 4** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Creating RoHC Profile for Subscriber using Compression Mode

Use the following example to create RoHC profile for a subscriber using compression mode:

```
configure

RoHC-profile profile-name <RoHC_comp_profile_name>

  decompression-options

    [no] multiple-ts-stride

    rtp-sn-p <p_value>

    [no] use-ipid-override

    [no] use-optimized-talkspurt

    [no] use-optimized-transience
```

```
[no] use-timer-based-compression

end
```

Notes:

- `<RoHC_comp_profile_name>` is the name of the RoHC profile with compression mode which you want to apply to a subscriber.
- System configured most of the parameters by default. For more information on other options and parameters and details, refer to the *RoHC Profile Compression Configuration Mode Commands* chapter in *Command Line Interface Reference*.

Creating RoHC Profile for Subscriber using Decompression Mode

Use the following example to create RoHC profile for a subscriber using decompression mode:

```
configure
```

```
RoHC-profile profile-name <RoHC_decomp_profile_name>

  decompression-options

    context-timeout <dur>

    max-jitter-cd <dur_ms>

    nak-limit <limit>

    optimistic-mode-ack

    optimistic-mode-ack-limit <num_pkts>

    piggyback-wait-time <dur_ms>

    preferred-feedback-mode { bidirectional-optimistic | bidirectional-reliable |
unidirectional }

    rtp-sn-p <p_value>

    [no] rtp-sn-p-override

    [no] use-clock-option

    [no] use-crc-option

    [no] use-feedback

    [no] use-jitter-option

    [no] use-reject-option

    [no] use-sn-option

  end
```


Notes:

- `<RoHC_profile_name>` is the name of the RoHC profile with decompression mode which you want to apply to a subscriber.
- System configured most of the parameters by default. For more information on other options and parameters and details, refer to the *RoHC Profile Decompression Configuration Mode Commands* chapter in *Command Line Interface Reference*.

Applying RoHC Profile to a Subscriber

Once an RoHC profile has been created that profile can be specified to be used for a specific subscribers. Use the following example to apply the RoHC profile to a subscriber:

```
configure
```

```
context <ctxt_name>

    subscriber name <subs_name>

        RoHC-profile-name <RoHC_profile_name>

    end
```

Notes:

- `<ctxt_name>` is the system context in which you wish to configure the subscriber profile. Typically this is an AAA context.
- `<subs_name>` is the name of the subscriber in the current context that you want to enable RoHC header compression for.
- `<RoHC_profile_name>` is the name of the existing RoHC profile (created with compressed or decompressed mode) which you want to apply to a subscriber in the current context.
- Refer to the *Subscriber Configuration Mode Commands* chapter in *Command Line Interface Reference* for more details on this command and its options.

Verifying the Header Compression Configuration

These instructions are used to verify the header compression configuration.

Step 1 Verify that your header compression configurations for subscriber by entering the following command in Exec Mode in specific context:

```
show subscriber configuration username subs_name
```

The output of this command is a concise listing of subscriber parameter settings as configured.

Disabling VJ Header Compression Over PPP

By default, VJ IP header compression is enabled for subscriber sessions. When VJ header compression is configured all IP headers are compressed using the VJ compression algorithm.

If you do not want to apply compression to any IP headers for a subscriber session you can disable the IP header compression feature.



Important: This section provides the minimum instruction set for configuring subscriber profile for header compression. For more information on commands that configure additional parameters and options, refer *Subscriber Configuration Mode Commands* chapter in *Command Line Interface Reference*.

To configure the system to disable VJ header compression to IP headers:

- Step 1** Disable header compression by applying the example configuration in the [Disabling VJ Header Compression](#) section.
- Step 2** Verify your VJ header compression configuration by following the steps in the [Verifying the VJ Header Compression Configuration](#) section.
- Step 3** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Disabling VJ Header Compression

Use the following example to disable the VJ header compression over PPP:

```
configure

context <ctxt_name>

    subscriber name <subs_name>

    no ip header-compression

end
```

Notes:

- <ctxt_name> is the system context in which you wish to configure the subscriber profile. Typically this is an AAA context.
- <subs_name> is the name of the subscriber in the current context that you want to disable IP header compression for.

Verifying the VJ Header Compression Configuration

These instructions are used to verify the VJ header compression configuration.

Step 1 Verify that your header compression configurations for subscriber by entering the following command in Exec Mode in specific context:

```
show subscriber configuration username <subs_name>
```

The output of this command is a concise listing of subscriber parameter settings as configured.

Disabling RoHC Header Compression Over SO67

If you do not want to apply compression to any IP headers for a subscriber sessions using the EVDO-RevA SO67 feature, you can disable the IP header compression feature at the PDSN or HSGW Service.



Important: This section provides the minimum instruction set for configuring subscriber profile for header compression. For more information on commands that configure additional parameters and options, refer *PDSN Service Configuration Mode Commands* or *HSGW Service Configuration Mode Commands* chapter in *Command Line Interface Reference*.

To configure the system to disable the IP header compression feature at the PDSN or HSGW Service:

- Step 1** Disable header compression by applying the example configuration in the [Disabling ROHC Header Compression](#) section.
- Step 2** Verify your RoHC configuration by following the steps in the [Verifying the Header Compression Configuration](#) section.
- Step 3** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Disabling RoHC Header Compression

Use the following example to disable the header compression over SO67:

```
configure

context <ctxt_name>

    pdsn/hsgw-service <svc_name>

        no ip header-compression RoHC

    end
```

Notes:

- <ctxt_name> is the system context in which PDSN or HSGW service is configured and you wish to configure the service profile.
- <svc_name> is the name of the PDSN or HSGW service in which you want to disable RoHC over SO67.

Verifying the Header Compression Configuration

These instructions are used to verify the header compression configuration.

- Step 1** Verify that your header compression configurations for subscriber by entering the following command in Exec Mode in specific context:

```
show configuration context <ctxt_name>
```

The output of this command is a concise listing of subscriber parameter settings as configured.

Checking IP Header Compression Statistics

This section contains commands to use to retrieve statistics that include IP header compression information.

The following Exec mode commands can be used to retrieve IP header compression statistics:

- `monitor protocol ppp`
- `show ppp`
- `show ppp statistics`
- `show RoHC statistics`
- `show RoHC statistics pdsn-service`
- `show subscriber full username`

For more information on these commands, refer to the *Command Line Interface Reference*.

RADIUS Attributes for IP Header Compression

This section lists the names of the RADIUS attributes to use for RoHC header compression. For more information on these attributes, refer to the AAA Interface Administration and Reference.

One of the following attributes can be used to specify the name of the RoHC profile to use for the subscriber session:

- SN-RoHC-Profile-Name
- SN1-RoHC-Profile-Name

Any RoHC parameters not specified in the RoHC profile are set to their default values.

Appendix J

IP Pool Sharing Protocol

This chapter provides information on configuring an enhanced, or extended, service. The product administration guides provide examples and procedures for configuration of basic services on the system. It is recommended that you select the configuration example that best meets your service model, and configure the required elements for that model before using the procedures in this chapter.

Sections in this chapter include:

- [Overview](#)
- [How IPSP Works](#)
- [Configuring IPSP Before the Software Upgrade](#)
- [Configuring IPSP After the Software Upgrade](#)
- [Disabling IPSP](#)

Overview

The IP Pool Sharing Protocol (IPSP) is a protocol that system-based HA services can use during an offline-software upgrade to avoid the assignment of duplicate IP addresses to sessions while allowing them to maintain the same address, and to preserve network capacity.

In order for IPSP to be used, at least two system-based HAs with identical configurations must be present on the same LAN. IPSP uses a primary & secondary model to manage the IP pools between the HAs. When used, this protocol ensures the following:

- In-progress sessions can be handed-off to the secondary HA when an offline-software upgrade is being performed on the primary and receive the same IP address that it was originally assigned.
- New sessions can be redirected to the secondary HA when an offline-software upgrade is being performed on the primary and receive a non-duplicate IP address.

The protocol is enabled at the interface level. Each system-based HA must have an IPSP-enabled interface configured in the same context as the HA service for this protocol to function properly.

Primary HA Functionality

The primary HA is the system that is to be upgraded. It performs the following functions for IPSP:

- Queries the pool information from the secondary HA; the pool configurations on both HAs must be identical
- Assigns an IP address or address block to the secondary HA when requested by the secondary HA; the primary HA releases sessions if they have an IP address requested by the secondary
- For graceful termination conditions (e.g. an administrative user issues the **reload** command), sends a termination message to the secondary HA causing it to assume the responsibilities of the primary HA until the primary is available again.
- Sends a trap when the number of calls drops to zero after starting IPSP

Secondary HA Functionality

The secondary HA is the system that takes over Mobile IP sessions from the primary HA that is being upgraded. It performs the following functions for IPSP:

- Locks the IP pools until it receives an address or address block assignment from the primary HA; it unlocks the IP pools after busying out the addresses that are not assigned to it
- Processes address requests for sessions that are within the address block assigned to it
- Communicates with the primary HA, as needed, to request IP addresses that are not currently assigned to it; it does not assign the address until the primary HA approves it
- For graceful termination conditions (e.g. an administrative user issues the **reload** command), it notifies the primary HA that it is going out of service
- Assumes the responsibility of the primary HA when requested to
- In the event that it determines that primary HA is not available, it assumes the responsibility of the primary HA if there is at least one address allocated to verify that the AAA server is re-configured to direct the calls

Requirements, Limitations, & Behavior

- One IPSP interface can be configured per system context.
- The IPSP interfaces for both the primary and secondary HAs must be configured to communicate on the same network.
- If IP pool busyout is enabled on any configured address pool, IPSP can not be configured.
- The IP pool configuration (pool name, addresses, priority, pool group, etc.) on both the HAs must be identical.
- IP pools cannot be modified on either the primary or the secondary HAs once IPSP is enabled.
- Sessions are dropped during the IPSP setup process if:
 - the primary HA has not yet approved an IP address or address block.
 - the primary HA is not known to the secondary HA.
- Once an address is assigned to the secondary HA, all the information about that address is erased on the primary HA and that address becomes unusable by the primary HA.
- LRU is not supported across the systems. Although, LRU continues to be supported within the system.
- If the IPSP configuration is not disabled before removing the HA from the IPSP network link, sessions may be rejected if the system's VPN Manager is rebooted or restarts.
- IPSP does not control static IP pools. An external application (AAA, etc.) must be responsible for ensuring that duplicate addresses are not assigned.
- IPSP ignores interface failures allowing the configured dead-interval timer to determine when the HA should become the primary and control the pool addresses. Before the dead-interval timer starts, the secondary HA maintains its state and any busied out addresses remain busied out. After the dead-interval timer starts, IPSP marks the neighboring peer HA as down, becomes primary, and will unbusy out all pool addresses.

How IPSP Works

IPSP operation requires special configuration in both the primary and secondary HAs. As mentioned previously, both HAs must have identical configurations. This allows the secondary HA to process sessions identically to the primary when the primary is taken offline for upgrade.

Configuration must also be performed on the AAA server. Whereas subscriber profiles on the AAA server originally directed sessions to the primary HA, prior to using IPSP, subscriber profiles must be re-configured to direct sessions to the secondary HA.

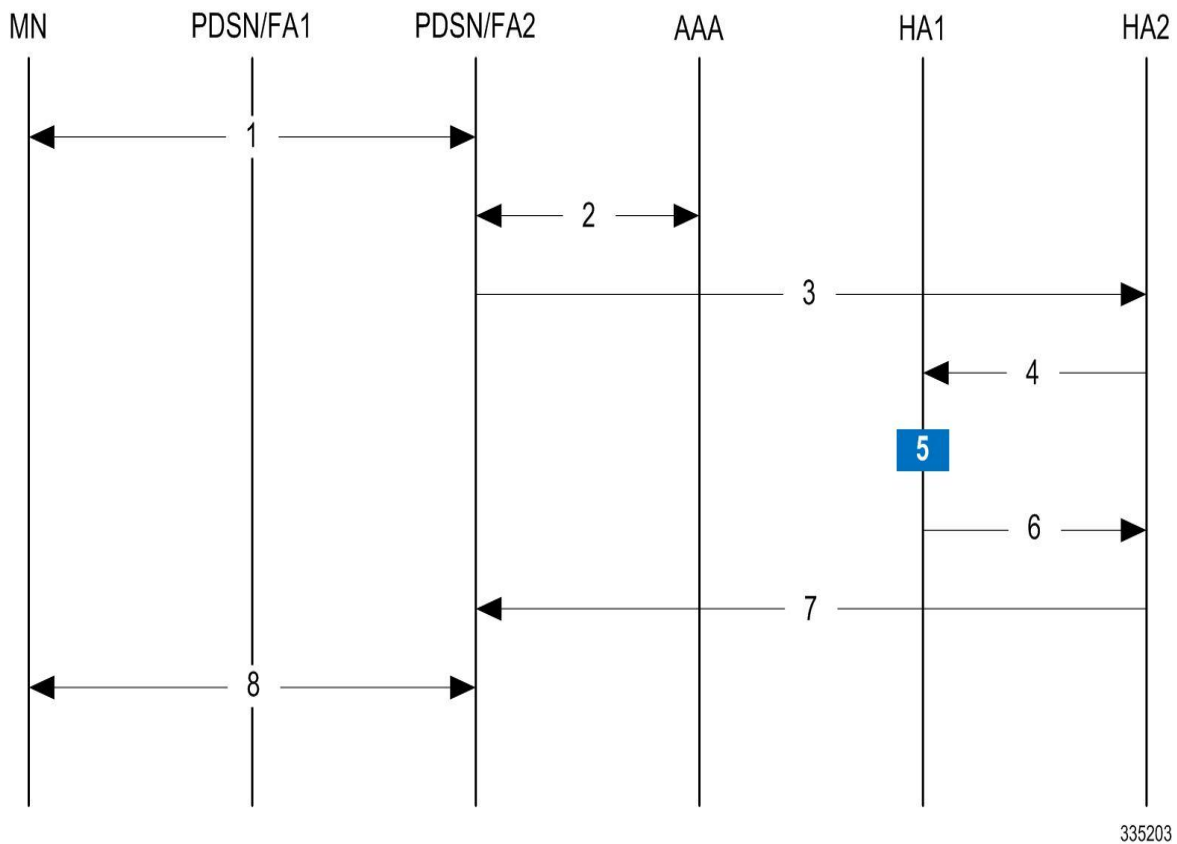
There are two scenarios in which IPSP takes effect:

- **New sessions:** Once IPSP is configured, new sessions are directed to a secondary HA (HA2) allowing the primary HA to go through a software upgrade without degrading network capacity. The secondary HA requests addresses from the primary HA's (HA1) pools as needed. As the addresses are allocated, they are busied out on the primary HA. This procedure is displayed below.
- **Session handoffs:** Once IPSP is configured, sessions originally registered with the primary HA (HA1) are re-registered with the secondary HA (HA2). To ensure the session is assigned the same IP address, the secondary HA requests the address from the primary HA. The primary HA verifies the binding and releases it to the secondary HA which, in turn, re-assigns it to the session. As the addresses are allocated, they are busied out on the primary HA. This procedure is displayed below.

IPSP Operation for New Sessions

The following figure and text describe how new sessions are handled when IPSP is enabled.

Figure 36. IPSP Operation for New Sessions



335203

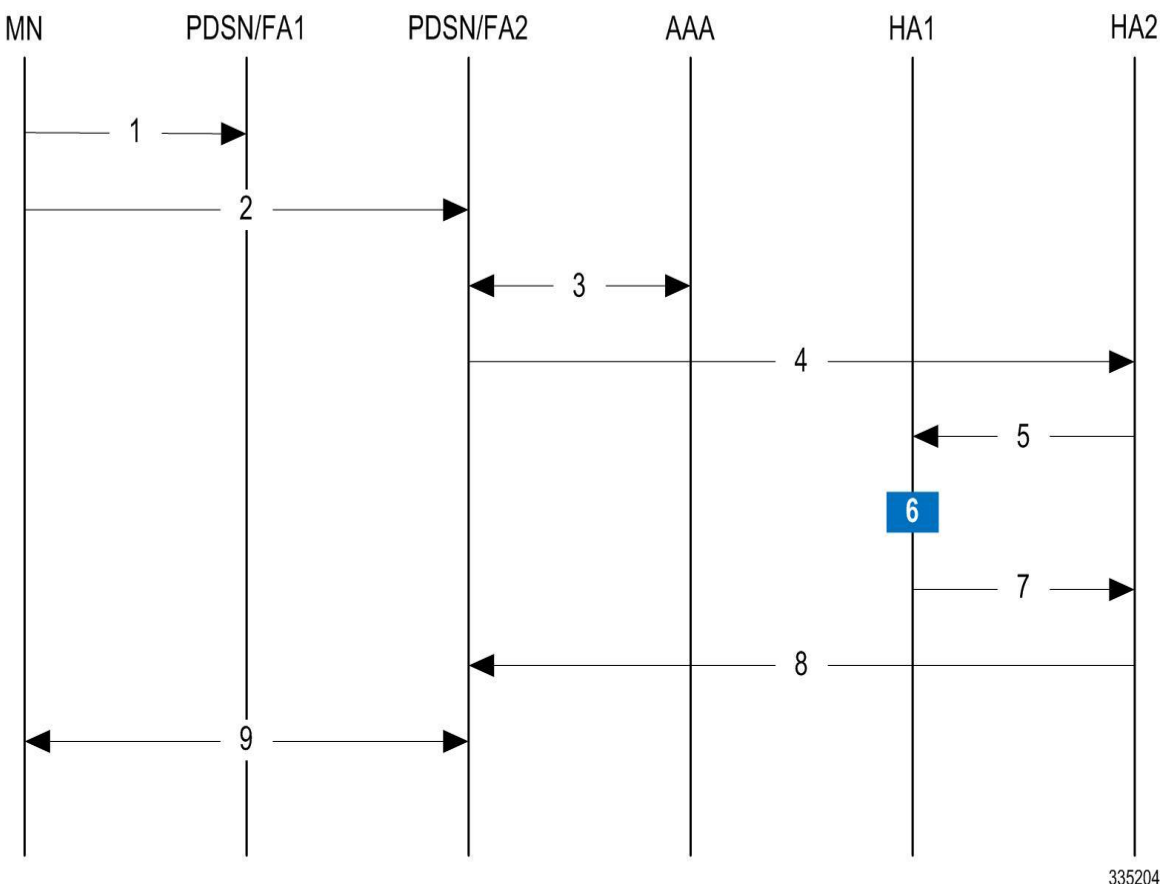
Table 36. IPSP Operation for New Sessions Description

Step	Description
1	A mobile node (MN) attempting to establish a data session is connected to PDSN/FA 2.
2	PDSNFA 2 authenticates the subscriber with the AAA server. One of the attributes returned by the AAA server as part of a successful authentication is the IP address of the secondary HA.
3	PDSN/FA 2 forwards the session request to HA2 for processing. HA2 processes the session as it would for any Mobile IP session.
4	With IPSP enabled, prior to assigning an IP address, HA2 sends a request to HA1 for an IP address.
5	HA1 allocates the address to HA2 and busies it out so it can not be re-assigned.
6	HA1 responds to HA2 with the IP address for the session.
7	HA2 proceeds with session processing and provides PDSN/FA 2 with the IP address for the MN.
8	The MN and PDSN/FA 2 complete session processing.

IPSP Operation for Session Handoffs

The following figure and text describe how session handoffs are handled when IPSP is enabled.

Figure 37. IPSP Operation for Session Handoffs



335204

Table 37. IPSP Operation for Session Handoffs Description

Step	Description
1	A mobile node (MN) is connected to PDSN/FA 1.
2	The MN's session is handed-off to PDSN/FA2 and goes through the re-registration process.
3	PDSN/FA 2 authenticates the subscriber with the AAA server as part of the re-registration process. One of the attributes returned by the AAA server as part of a successful authentication is the IP address of the secondary HA.
4	PDSN/FA 2 forwards the session request to HA2 for processing. Included in the request is the MN's current IP address.
5	With IPSP enabled, prior to assigning an IP address, HA2 sends a request to HA1 for an IP address.
6	HA1 verifies the MN's information and releases the binding. It then busies out the address so it can not be re-assigned.
7	HA1 allocates the original IP address to HA2 for the session.

Step	Description
8	HA2 proceeds with session processing and provides PDSN/FA 2 with the IP address for the mobile node.
9	The mobile node and PDSN/FA 2 complete session processing.

Configuring IPSP Before the Software Upgrade

Configuring IPSP requires changes to the primary HA (the HA on which the software upgrade is to occur), the secondary HA (the HA to which subscribers sessions are to be directed), and the AAA server.

This section provides information and instructions for configuring IPSP before the software upgrade.



Important: This section provides the minimum instruction set for configuring IPSP on the system. For more information on commands that configure additional parameters and options, refer to the *IPSP Configuration Mode Commands* chapter in the *Command Line Interface Reference*.

To enable the IP pool sharing during software upgrade:

- Step 1** Configure the AAA servers by applying the example configuration in the [Configuring the AAA Server for IPSP](#) section.
- Step 2** Configure an interface on the system for use by IPSP according to the instructions found in the *Creating and Configuring Ethernet Interfaces and Ports* section of the *System Administration Guide*.
- Step 3** Enable the IPSP on secondary HA by applying the example configuration in the [Enabling IPSP on the Secondary HA](#) section.
- Step 4** Perform the boot system priority and SPC/SMC card synchronization as described in *Off-line Software Upgrade* section in the *System Administration Guide*.
- Step 5** Enable the IPSP on primary HA by applying the example configuration in the [Enabling IPSP on the Primary HA](#) section.
- Step 6** Verify your ACL configuration by following the steps in the *Verifying the IPSP Configuration* section.
- Step 7** Proceed for software upgrade as described in *Off-line Software Upgrade* section in the *System Administration Guide*.
- Step 8** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Configuring the AAA Server for IPSP

For subscriber session establishment, the AAA server provides the IP address of the HA that is to service the session. This information exists in the 3GPP2_MIP_HA_Address RADIUS attribute configured for the subscriber.


Because the primary HA has been responsible for facilitating subscriber sessions, its IP address is the one configured via this attribute. For IPSP however, the attribute configuration must change in order to direct sessions to the secondary HA.

To do this, reconfigure the 3GPP2_MIP_HA_Address RADIUS attribute for each subscriber on the AAA server with the IP address of the secondary HA.

The precise instructions for performing this operation vary depending on the AAA server vendor. Refer to the documentation for your AAA server for more information.

Enabling IPSP on the Secondary HA

The secondary HA is the alternate HA that is to take responsibility while the primary HA is upgraded.

 **Important:** This section provides the minimum instruction set for configuring IPSP on the system. For more information on commands that configure additional parameters and options, refer to the *IPSP Configuration Mode Commands* chapter in the *Command Line Interface Reference*.

Use the following example to enable the IPSP on secondary HA:

```
configure

context <ipsp_ctxt_name> [ -noconfirm ]

    interface <ipsp_if_name>

        pool-share-protocol primary <pri_ha_address> [ mode {active | inactive |
check-config } ]

        dead-interval <dur_sec>


    end
```

Notes:

- The interface must be configured in the same context as the HA service and must be on the same network as the primary HA's IPSP interface.
- *ipsp_if_name* is the name of the interface on which you want to enable IPSP.
- *dead-interval* is an optional command to configure time to wait before retrying the primary HA for the IP Pool Sharing Protocol.

Enabling IPSP on the Primary HA

The primary HA is the HA that is to be upgraded.

 **Important:** This section provides the minimum instruction set for configuring IPSP on the system. For more information on commands that configure additional parameters and options, refer to the *IPSP Configuration Mode Commands* chapter in the *Command Line Interface Reference*.

Use the following example to enable the IPSP on primary HA:

```
configure

context <ipsp_ctxt_name> [ -noconfirm ]

    interface <ipsp_if_name>

        pool-share-protocol secondary <sec_ha_address> [ mode {active | inactive
| check-config } ]
```

```
dead-interval <dur_sec>

end
```

Notes:

- The interface must be configured in the same context as the HA service and must be on the same network as the secondary HA's IPSP interface.
- *ipsp_if_name* is the name of the interface on which you want to enable IPSP.
- *dead-interval* is an optional command to configure time to wait before retrying the secondary HA for the IP Pool Sharing Protocol.



Important: Once this configuration is done, the primary HA begins to hand responsibility for sessions and release IP addresses to the secondary HA. Prior to performing the software upgrade, all IP addresses must be released. When IPSP has released all IP pool addresses from the primary HA an SNMP trap (**starIPSPAllAddrsFree**) is triggered.

Verifying the IPSP Configuration

These instructions are used to verify the IPSP configuration.


Verify that IPSP has released all IP addresses by entering the following command in Exec Mode with in specific context:

```
show ip ipsp
```

The output of this command provides the list of used addresses and released addresses. The system will send the **starIPSPAllAddrsFree** trap once all IP addresses are released. When the value in the *Used Addresses* column reaches 0 for all IP pools listed, then the primary HA sends the SNMP trap and notifies the secondary HA to take over as the primary HA.


Configuring IPSP After the Software Upgrade

If desired, IP pool addresses can be migrated from the original secondary HA back to the original primary HA once the upgrade process is complete.

 **Important:** It is important to note that the HA that was originally designated as the secondary is now functioning as the primary HA. Conversely, the HA that was originally designated as the primary is now functioning as the secondary.

In order to migrate the addresses, both HAs and the AAA server must be configured according to the instructions in this section.

This section provides information and instructions for configuring IPSP after the software upgrade.

 **Important:** This section provides the minimum instruction set for configuring IPSP on the system. For more information on commands that configure additional parameters and options, refer *IPSP Configuration Mode Commands* chapter in the *Command Line Interface Reference*.

To enable the IP pool sharing after software upgrade:

- Step 1** Configure the AAA servers by applying the example configuration in the [Configuring the AAA Server for IPSP](#) section.
- Step 2** Configure an interface on the system for use by IPSP according to the instructions found in the Creating and Configuring Ethernet Interfaces and Ports section of *System Administration Guide*.
- Step 3** Enable the IPSP on secondary HA by applying the example configuration in the [Enabling IPSP on the Secondary HA](#) section.
- Step 4** Enable the IPSP on primary HA by applying the example configuration in the [Enabling IPSP on the Primary HA](#) section.
- Step 5** Verify your ACL configuration by following the steps in the *Verifying the IPSP Configuration* section.
- Step 6** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Disabling IPSP

Once all IP addresses on the primary HA have been released, IPSP must be disabled on both the primary and secondary HAs.



Caution: Prior to disabling IPSP, ensure that the primary HA has released all IP addresses to secondary HA.

Follow the instructions in this section to disable IPSP on primary and secondary HA after migration of all IP addresses.



Important: This section provides the minimum instruction set for disabling IPSP on the HAs. For more information on commands, refer to the *IPSP Configuration Mode Commands* chapter in the *Command Line Interface Reference*.

Use the following example to enable the IPSP on primary/secondary HA:

```
configure

context <ipsp_ctxt_name> [ -noconfirm ]

    interface <ipsp_if_name>

        no pool-share-protocol

    end
```

Notes:

- The interface must be configured in the same context as the primary/secondary HA service and must be on the same network as the primary/secondary HA's IPSP interface.
- *ipsp_if_name* is the name of the interface on which you want to disable IPSP.
- IPSP must be disabled on both the HAs.

Appendix K

IP Security

This chapter provides information on configuring an enhanced or extended service. The product Administration Guides provide examples and procedures for configuration of basic services on the system. It is recommended that you select the configuration example that best meets your service model, and configure the required elements for that model, as described in the respective product Administration Guide, before using the procedures in this chapter.



Important: The IP Security is a licensed Cisco feature. A separate feature license may be required. Contact your Cisco account representative for detailed information on specific licensing requirements. For information on installing and verifying licenses, refer to the *Managing License Keys* section of the *Software Management Operations* chapter in the *System Administration Guide*.



Caution: IPSec parameter configurations saved using this release may not function properly with older software releases.

This chapter contains the following sections:

- [Overview](#)
- [IPSec Terminology](#)
- [Implementing IPSec for PDN Access Applications](#)
- [Implementing IPSec for Mobile IP Applications](#)
- [Implementing IPSec for L2TP Applications](#)
- [Transform Set Configuration](#)
- [ISAKMP Policy Configuration](#)
- [ISAKMP Crypto Map Configuration](#)
- [Dynamic Crypto Map Configuration](#)
- [Manual Crypto Map Configuration](#)
- [Crypto Map and Interface Association](#)
- [FA Services Configuration to Support IPSec](#)
- [HA Service Configuration to Support IPSec](#)
- [RADIUS Attributes for IPSec-based Mobile IP Applications](#)
- [LAC Service Configuration to Support IPSec](#)
- [Subscriber Attributes for L2TP Application IPSec Support](#)
- [PDSN Service Configuration for L2TP Support](#)
- [Redundant IPSec Tunnel Fail-Over](#)
- [Redundant IPSec Tunnel Fail-over Configuration](#)
- [Dead Peer Detection \(DPD\) Configuration](#)

■ Disabling IPSP

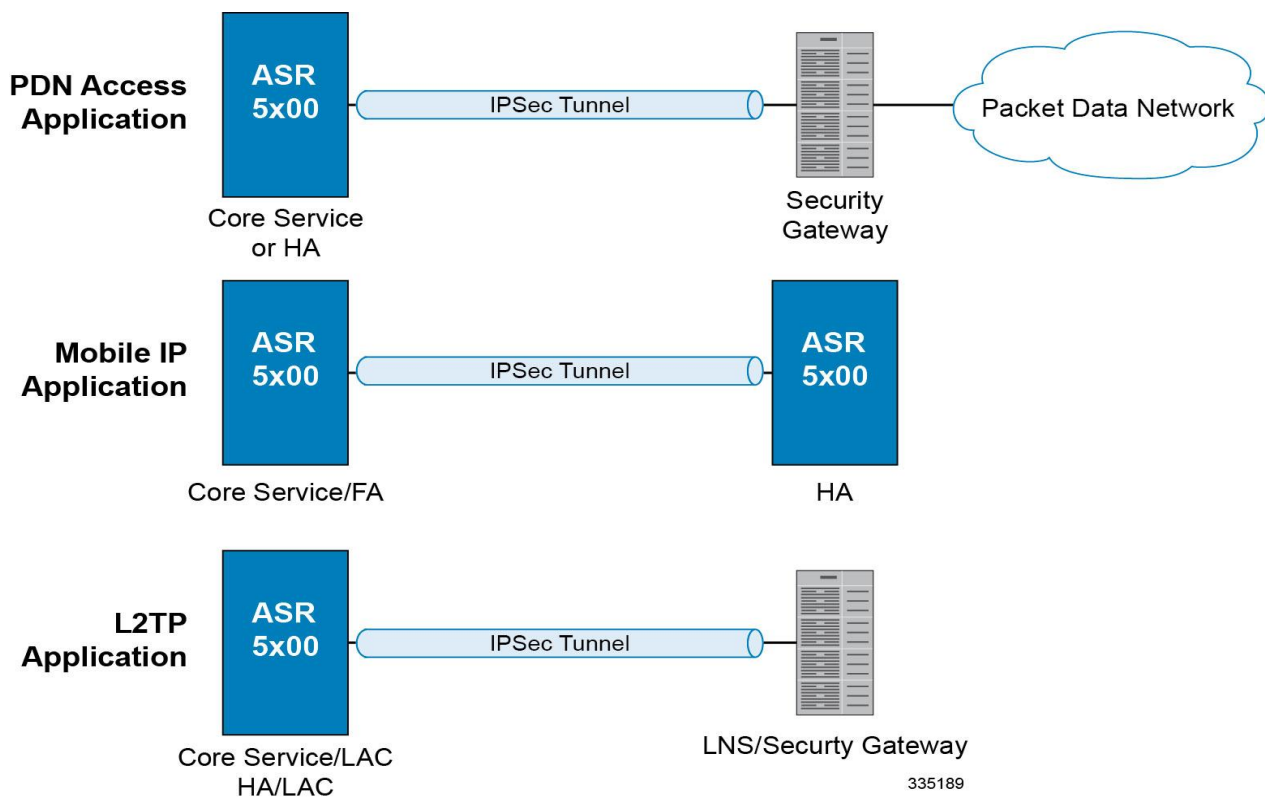
- [APN Template Configuration to Support L2TP](#)
- [IPSec for LTE/SAE Networks](#)

Overview

IP Security (IPSec) is a suite of protocols that interact with one another to provide secure private communications across IP networks. These protocols allow the system to establish and maintain secure tunnels with peer security gateways. IPSec can be implemented on the system for the following applications:

- **PDN Access:** Subscriber IP traffic is routed over an IPSec tunnel from the system to a secure gateway on the packet data network (PDN) as determined by access control list (ACL) criteria. This application can be implemented for both core network service and HA-based systems. The following figure shows IPSec configurations.

Figure 38. IPSec Applications



- **Mobile IP:** Mobile IP control signals and subscriber data is encapsulated in IPSec tunnels that are established between foreign agents (FAs) and home agents (HAs) over the Pi interfaces.



Important: Once an IPSec tunnel is established between an FA and HA for a particular subscriber, all new Mobile IP sessions using the same FA and HA are passed over the tunnel regardless of whether or not IPSec is supported for the new subscriber sessions. Data for existing Mobile IP sessions is unaffected.

- **L2TP:** L2TP-encapsulated packets are routed from the system to an LNS/secure gateway over an IPSec tunnel.

Note that: IPSec can be implemented for both attribute-based and compulsory tunneling applications for 3GPP2 services.

Applicable Products and Relevant Sections

The IPSec feature is supported for various products. The following table indicates the products on which the feature is supported and the relevant sections within the chapter that pertain to that product.

Applicable Product(s)	Refer to Sections
PDSN/FA/HA	<ul style="list-style-type: none"> • Implementing IPSec for PDN Access Applications • Implementing IPSec for Mobile IP Applications • Transform Set Configuration • ISAKMP Policy Configuration • ISAKMP Crypto Map Configuration • Dynamic Crypto Map Configuration • Manual Crypto Map Configuration • Crypto Map and Interface Association • FA Services Configuration to Support IPSec • HA Service Configuration to Support IPSec • RADIUS Attributes for IPSec-based Mobile IP Applications • LAC Service Configuration to Support IPSec • Subscriber Attributes for L2TP Application IPSec Support • PDSN Service Configuration for L2TP Support • Redundant IPSec Tunnel Fail-Over • Dead Peer Detection (DPD) Configuration

Applicable Product(s)	Refer to Sections
GGSN/FA/HA	<ul style="list-style-type: none">• Implementing IPSec for PDN Access Applications• Implementing IPSec for Mobile IP Applications• Implementing IPSec for L2TP Applications• Transform Set Configuration• ISAKMP Policy Configuration• ISAKMP Crypto Map Configuration• Dynamic Crypto Map Configuration• Manual Crypto Map Configuration• Crypto Map and Interface Association• FA Services Configuration to Support IPSec• HA Service Configuration to Support IPSec• RADIUS Attributes for IPSec-based Mobile IP Applications• LAC Service Configuration to Support IPSec• Redundant IPSec Tunnel Fail-Over• Dead Peer Detection (DPD) Configuration• TAPN Template Configuration to Support L2TP

Applicable Product(s)	Refer to Sections
ASN GW	<ul style="list-style-type: none">• Implementing IPsec for PDN Access Applications• Implementing IPsec for Mobile IP Applications• Implementing IPsec for L2TP Applications• Transform Set Configuration• ISAKMP Policy Configuration• ISAKMP Crypto Map Configuration• Dynamic Crypto Map Configuration• Manual Crypto Map Configuration• Crypto Map and Interface Association• FA Services Configuration to Support IPsec• HA Service Configuration to Support IPsec• RADIUS Attributes for IPsec-based Mobile IP Applications• LAC Service Configuration to Support IPsec• Subscriber Attributes for L2TP Application IPsec Support• Redundant IPsec Tunnel Fail-Over• Dead Peer Detection (DPD) Configuration

IPSec Terminology

There are four items related to IPSec support on the system that must be understood prior to beginning configuration. They are:

- Crypto Access Control List (ACL)
- Transform Set
- ISAKMP Policy
- Crypto Map

Crypto Access Control List (ACL)

As described in the *IP Access Control Lists* chapter of this guide, ACLs on the system define rules, usually permissions, for handling subscriber data packets that meet certain criteria. Crypto ACLs, however, define the criteria that must be met in order for a subscriber data packet to be routed over an IPSec tunnel.

Unlike other ACLs that are applied to interfaces, contexts, or one or more subscribers, crypto ACLs are matched with crypto maps. In addition, crypto ACLs contain only a single rule while other ACL types can consist of multiple rules.

Prior to routing, the system examines the properties of each subscriber data packet. If the packet properties match the criteria specified in the crypto ACL, the system will initiate the IPSec policy dictated by the crypto map.

Transform Set

Transform Sets are used to define IPSec security associations (SAs). IPSec SAs specify the IPSec protocols to use to protect packets.

Transform sets are used during Phase 2 of IPSec establishment. In this phase, the system and a peer security gateway negotiate one or more transform sets (IPSec SAs) containing the rules for protecting packets. This negotiation ensures that both peers can properly protect and process the packets.

ISAKMP Policy

Internet Security Association Key Management Protocol (ISAKMP) policies are used to define Internet Key Exchange (IKE) SAs. The IKE SAs dictate the shared security parameters (i.e. which encryption parameters to use, how to authenticate the remote peer, etc.) between the system and a peer security gateway.

During Phase 1 of IPSec establishment, the system and a peer security gateway negotiate IKE SAs. These SAs are used to protect subsequent communications between the peers including the IPSec SA negotiation process.

Crypto Map

Crypto Maps define the tunnel policies that determine how IPSec is implemented for subscriber data packets.

There are three types of crypto maps supported by the system. They are:

- Manual crypto maps

- ISAKMP crypto maps
- Dynamic crypto maps

Manual Crypto Maps

These are static tunnels that use pre-configured information (including security keys) for establishment. Because they rely on statically configured information, once created, the tunnels never expire; they exist until their configuration is deleted.

Manual crypto maps define the peer security gateway to establish a tunnel with, the security keys to use to establish the tunnel, and the IPSec SA to be used to protect data sent/received over the tunnel. Additionally, manual crypto maps are applied to specific system interfaces.



Important: Because manual crypto map configurations require the use of static security keys (associations), they are not as secure as crypto maps that rely on dynamically configured keys. Therefore, it is recommended that they only be configured and used for testing purposes.

ISAKMP Crypto Maps

These tunnels are similar to manual crypto maps in that they require some statically configured information such as the IP address of a peer security gateway and that they are applied to specific system interfaces.

However, ISAKMP crypto maps offer greater security because they rely on dynamically generated security associations through the use of the Internet Key Exchange (IKE) protocol.

When ISAKMP crypto maps are used, the system uses the pre-shared key configured for map as part of the Diffie-Hellman (D-H) exchange with the peer security gateway to initiate Phase 1 of the establishment process. Once the exchange is complete, the system and the security gateway dynamically negotiate IKE SAs to complete Phase 1. In Phase 2, the two peers dynamically negotiate the IPSec SAs used to determine how data traversing the tunnel will be protected.

Dynamic Crypto Maps

These tunnels are used for protecting L2TP-encapsulated data between the system and an LNS/security gateway or Mobile IP data between an FA service configured on one system and an HA service configured on another.

The system determines when to implement IPSec for L2TP-encapsulated data either through attributes returned upon successful authentication for attribute based tunneling, or through the configuration of the LAC service used for compulsory tunneling.

The system determines when to implement IPSec for Mobile IP based on RADIUS attribute values as well as the configurations of the FA and HA service(s).

Implementing IPsec for PDN Access Applications

This section provides information on the following topics:

- How the IPSec-based PDN Access Configuration Works
- Configuring IPSec Support for PDN Access

In covering these topics, this section assumes that ISAKMP crypto maps are configured/used as opposed to manual crypto maps.

How the IPSec-based PDN Access Configuration Works

The following figure and the text that follows describe how sessions accessing a PDN using IPSec are processed by the system.

Figure 39. IPSec PDN Access Processing

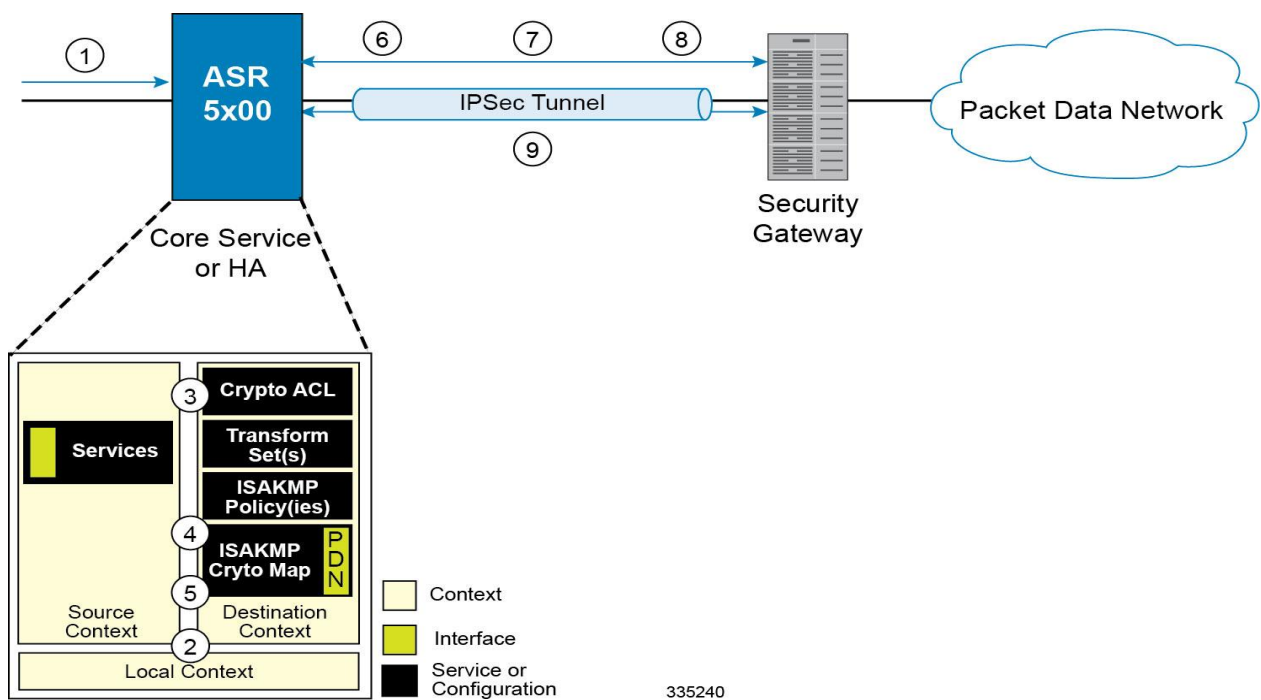


Table 38. IPSec PDN Access Processing

Step	Description
1.	A subscriber session or PDP context Request, in GGSN service, arrives at the system.
2.	The system processes the subscriber session or request as it would typically.
3.	Prior to routing the session packets, the system compares them against configured Access Control Lists (ACLs).

Step	Description
4.	The system determines that the packet matches the criteria of an ACL that is associated with a configured crypto map.
5.	From the crypto map, the system determines the following: <ul style="list-style-type: none"> • The map type, in this case ISAKMP • The pre-shared key used to initiate the Internet Key Exchange (IKE) and the IKE negotiation mode • The IP address of the security gateway • Whether perfect forward secrecy (PFS) should be enabled for the IPsec SA and if so, what group should be used • IPsec SA lifetime parameters • The name of a configured transform set defining the IPsec SA
6.	To initiate the IKE SA negotiation, the system performs a Diffie-Hellman exchange of the pre-shared key specified in the crypto map with the specified peer security gateway.
7.	The system and the security gateway negotiate an ISAKMP policy (IKE SA) to use to protect further communications.
8.	Once the IKE SA has been negotiated, the system negotiates an IPsec SA with the security gateway using the transform method specified in the transform sets.
9.	Once the IPsec SA has been negotiated, the system protects the data according to the IPsec SAs established during step 8 and sends it over the IPsec tunnel.

Configuring IPsec Support for PDN Access

This section provides a list of the steps required to configure IPsec functionality on the system in support of PDN access. Each step listed refers to a different section containing the specific instructions for completing the required procedure.



Important: These instructions assume that the system was previously configured to support subscriber data sessions either as a core service or an HA. In addition, parameters configured using this procedure must be configured in the same destination context on the system.

- Step 1** Configure one or more IP access control lists (ACLs) according to the information and instructions located in *IP Access Control Lists* chapter of this guide.
- Step 2** Configure one or more transform sets according to the instructions located in the [Transform Set Configuration](#) section of this chapter.
- Step 3** Configure one or more ISAKMP policies according to the instructions located in the [ISAKMP Policy Configuration](#) section of this chapter.
- Step 4** Configure an ipsec-isakmp crypto map according to the instructions located in the [ISAKMP Crypto Map Configuration](#) section of this chapter.
- Step 5** Apply the crypto map to an interface on the system according to the instructions located in the [Crypto Map and Interface Association](#) section of this chapter.

- Step 6** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Implementing IPSec for Mobile IP Applications

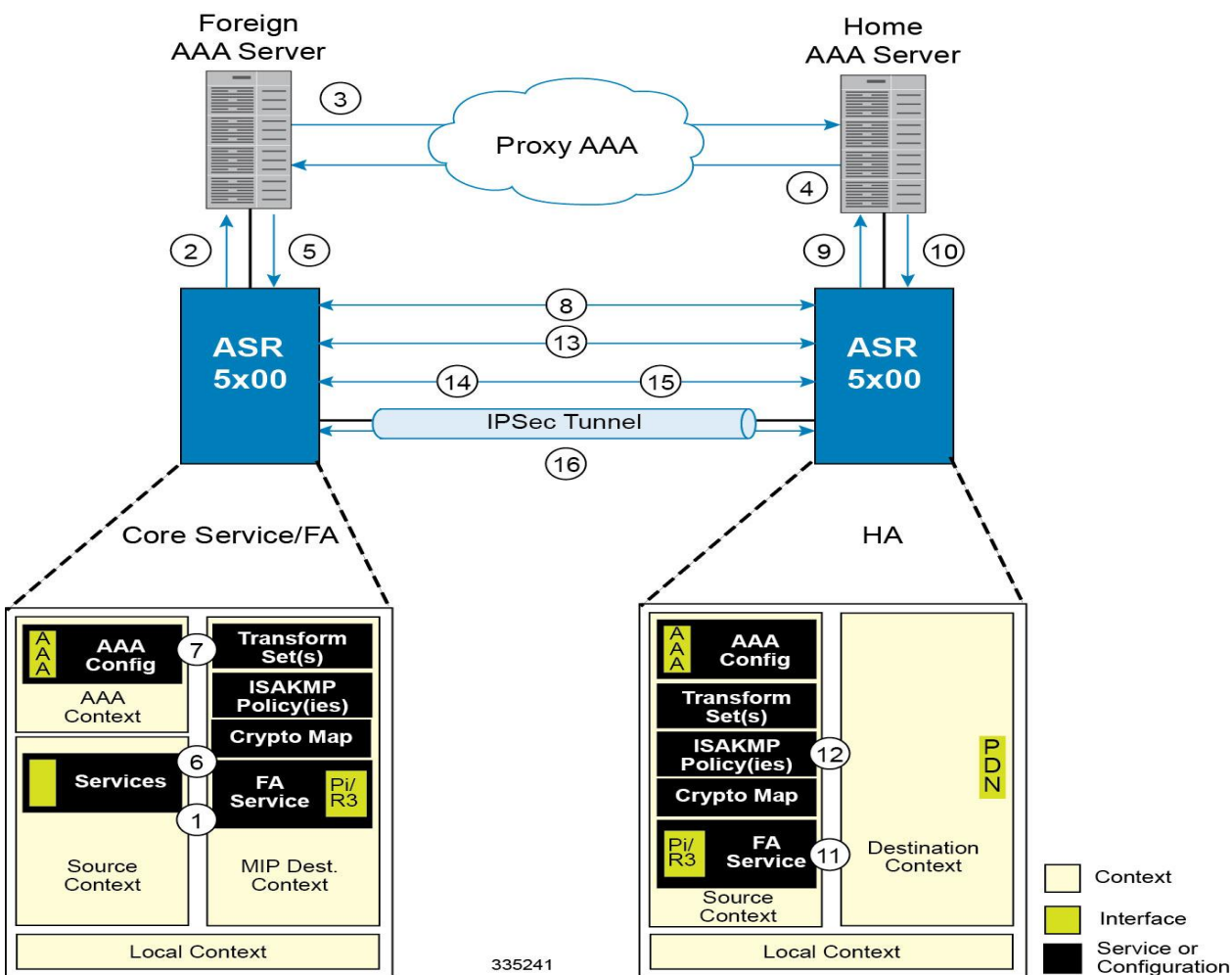
This section provides information on the following topics:

- [How the IPSec-based Mobile IP Configuration Works](#)
- [Configuring IPSec Support for Mobile IP](#)

How the IPSec-based Mobile IP Configuration Works

The following figure and the text that follows describe how Mobile IP sessions using IPSec are processed by the system.

Figure 40. IPSec-based Mobile IP Session Processing



335241

Table 39. IPSec-based Mobile IP Session Processing

Step	Description
1.	FA service receives a Mobile IP registration request from the mobile node.
2.	FA sends an Access-Request to the FAAA server with the 3GPP2-IKE-Secret-Request attribute equal to yes.
3.	The FAAA proxies the request to the HAAA.
4.	The HAAA returns an Access-Accept message including the following attributes: <ul style="list-style-type: none"> • 3GPP2-Security-Level set to 3 for IPSec tunnels and registration messages • 3GPP2-MIP-HA-Address indicating the IP address of the HA that the FA is to communicate with. • 3GPP2-KeyId providing an identification number for the IKE secret (alternatively, the keys may be statically configured for the FA and/or HA) • 3GPP2-IKE-Secret indicating the pre-shared secret to use to negotiate the IKE SA
5.	The FAAA passes the accept message to the FA with all of the attributes.
6.	The FA determines if an IPSec SA already exists based on the HA address supplied. If so, that SA will be used. If not, a new IPSec SA will be negotiated.
7.	The FA determines the appropriate crypto map to use for IPSec protection based on the HA address attribute. It does this by comparing the address received to those configured using the isakmp peer-ha command. From the crypto map, the system determines the following: <ul style="list-style-type: none"> • The map type, in this case dynamic • Whether perfect forward secrecy (PFS) should be enabled for the IPSec SA and if so, what group should be used • IPSec SA lifetime parameters • The name of one or more configured transform set defining the IPSec SA
8.	To initiate the IKE SA negotiation, the FA performs a Diffie-Hellman (D-H) exchange of the ISAKMP secret specified in the IKE secret attribute with the peer HA dictated by the HA address attribute. Included in the exchange is the Key ID received from the HAAA.
9.	Upon receiving the exchange, the HA sends an access request to the HAAA with the following attributes: <ul style="list-style-type: none"> • 3GPP2-S-Request (note that this attribute is not used if the IPSec keys are statically configured) • 3GPP2-User-name (the username specified is the IP addresses of the FA and HA). The password used in the access request is the RADIUS shared secret.
10.	The HAAA returns an Access-Accept message to the HA with the following attributes: <ul style="list-style-type: none"> • 3GPP2-S indicating the “S” secret used to generate the HA’s response to the D-H exchange • 3GPP2-S-Lifetime indicating the length of time that the “S” secret is valid • 3GPP2-Security-Level set to 3 for IPSec tunnels and registration messages (optional)

Step	Description
11.	<p>The HA determines the appropriate crypto map to use for IPsec protection based on the FA's address. It does this by comparing the address received to those configured using the isakmp peer-fa command. From the crypto map, the system determines the following:</p> <ul style="list-style-type: none"> • The map type, in this case dynamic • Whether perfect forward secrecy (PFS) should be enabled for the IPsec SA and if so, what group should be used • IPsec SA lifetime parameters • The name of one or more configured transform set defining the IPsec SA
12.	The HA creates a response to the D-H exchange using the "S" secret and the Key ID sent by the FA.
13.	The HA sends IKE SA negotiation D-H exchange response to the FA.
14.	The FA and the HA negotiate an ISAKMP (IKE) policy to use to protect further communications.
15.	Once the IKE SA has been negotiated, the system negotiates an IPsec SA with the security gateway using the transform method specified in the transform sets.
16.	Once the IPsec SA has been negotiated, the system protects the data according to the IPsec SAs established during step 15 and sends it over the IPsec tunnel.



Important: Once an IPsec tunnel is established between an FA and HA for a particular subscriber, all new Mobile IP sessions using the same FA and HA are passed over the tunnel regardless of whether or not IPsec is supported for the new subscriber sessions. Data for existing Mobile IP sessions is unaffected.

Configuring IPsec Support for Mobile IP

This section provides a list of the steps required to configure IPsec functionality on the system in support of Mobile IP. Each step listed refers to a different section containing the specific instructions for completing the required procedure.



Important: These instructions assume that the systems were previously configured to support subscriber data sessions either as an FA or an HA.

- Step 1** Configure one or more transform sets for the FA system according to the instructions located in the [Transform Set Configuration](#) section of this chapter.
- The transform set(s) must be configured in the same context as the FA service.
- Step 2** Configure one or more ISAKMP policies for the FA system according to the instructions located in the [ISAKMP Policy Configuration](#) section of this chapter.
- The ISAKMP policy(ies) must be configured in the same context as the FA service.
- Step 3** Configure an ipsec-isakmp crypto map for the FA system according to the instructions located in the [Dynamic Crypto Map Configuration](#) section of this chapter.
- The crypto map(s) must be configured in the same context as the FA service.

- Step 4** Optional. Configure DPD for the FA to help prevent IPSec tunnel state mismatches between the FA and HA according to the instructions located in the [Dead Peer Detection \(DPD\) Configuration](#) section of this chapter.



Important: Though the use of DPD is optional, it is recommended in order to ensure service availability.

- Step 5** Configure the FA Service or the FA system according to the instructions located in the [FA Services Configuration to Support IPSec](#) section of this chapter.

- Step 6** Configure one or more transform sets for the HA system according to the instructions located in the [Transform Set Configuration](#) section of this chapter.

The transform set(s) must be configured in the same context as the HA service.

- Step 7** Configure one or more ISAKMP policies or the HA system according to the instructions located in the [ISAKMP Policy Configuration](#) section of this chapter.

The ISAKMP policy(ies) must be configured in the same context as the HA service.

- Step 8** Configure an ipsec-isakmp crypto map or the HA system according to the instructions located in the [Dynamic Crypto Map Configuration](#) section of this chapter.

The crypto map(s) must be configured in the same context as the HA service.

- Step 9** Optional. Configure DPD for the HA to help prevent IPSec tunnel state mismatches between the FA and HA according to the instructions located in the [Dead Peer Detection \(DPD\) Configuration](#) section of this chapter.



Important: Though the use of DPD is optional, it is recommended in order to ensure service availability.

- Step 10** Configure the HA Service or the HA system according to the instructions located in the section of this chapter.

- Step 11** Configure the required attributes for RADIUS-based subscribers according to the information located in the [RADIUS Attributes for IPSec-based Mobile IP Applications](#) section of this chapter.

- Step 12** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Implementing IPsec for L2TP Applications

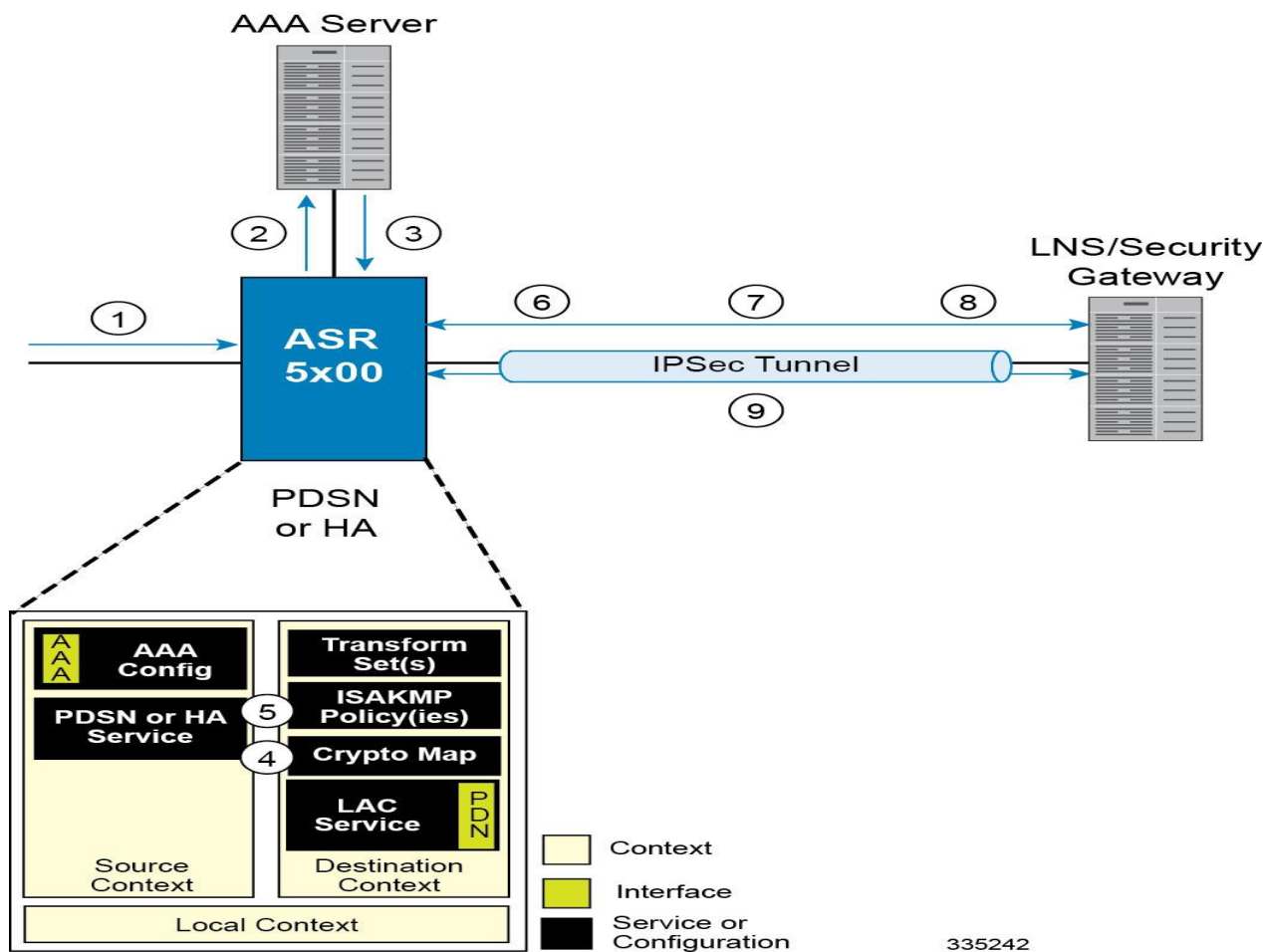
This section provides information on the following topics:

- [How IPsec is Used for Attribute-based L2TP Configurations](#)
- [Configuring Support for L2TP Attribute-based Tunneling with IPsec](#)
- [How IPsec is Used for PDSN Compulsory L2TP Configurations](#)
- [Configuring Support for L2TP PDSN Compulsory Tunneling with IPsec](#)
- [How IPsec is Used for L2TP Configurations on the GGSN](#)
- [Configuring GGSN Support for L2TP Tunneling with IPsec](#)

How IPsec is Used for Attribute-based L2TP Configurations

The following figure and the text that follows describe how IPsec-encrypted attribute-based L2TP sessions are processed by the system.

Figure 41. Attribute-based L2TP, IPsec-Encrypted Session Processing



335242

Table 40. Attribute-based L2TP, IPsec-Encrypted Session Processing

Step	Description
1.	A subscriber session arrives at the system.
2.	The system attempts to authenticate the subscriber with the AAA server.
3.	The profile attributes returned upon successful authentication by the AAA server indicate that session data is to be tunneled using L2TP. In addition, attributes specifying a crypto map name and ISAKMP secret are also supplied indicating that IP security is also required.
4.	The system determines that the crypto map name supplied matches a configured crypto map.

Step	Description
5.	From the crypto map, the system determines the following: <ul style="list-style-type: none"> • The map type, in this case dynamic • Whether perfect forward secrecy (PFS) should be enabled for the IPsec SA and if so, what group should be used • IPsec SA lifetime parameters • The name of one or more configured transform set defining the IPsec SA
6.	To initiate the IKE SA negotiation, the system performs a Diffie-Hellman exchange of the ISAKMP secret specified in the profile attribute with the specified peer LNS/security gateway.
7.	The system and the LNS/security gateway negotiate an ISAKMP (IKE) policy to use to protect further communications.
8.	Once the IKE SA has been negotiated, the system negotiates an IPsec SA with the LNS/security gateway using the transform method specified in the transform sets.
9.	Once the IPsec SA has been negotiated, the system protects the L2TP encapsulated data according to the IPsec SAs established during step 9 and sends it over the IPsec tunnel.

Configuring Support for L2TP Attribute-based Tunneling with IPsec

This section provides a list of the steps required to configure IPsec functionality on the system in support of attribute-based L2TP tunneling. Each step listed refers to a different section containing the specific instructions for completing the required procedure.



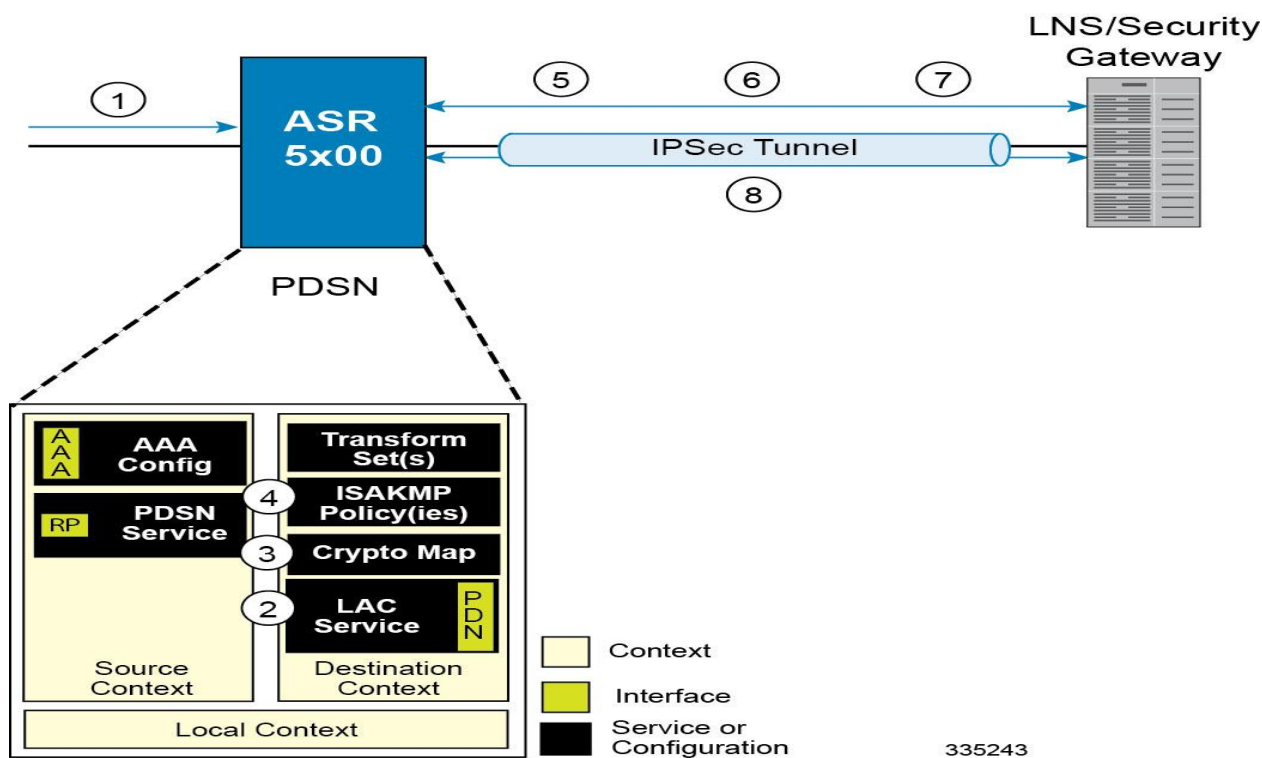
Important: These instructions assume that the system was previously configured to support subscriber data sessions and L2TP tunneling either as a PDSN or an HA. In addition, with the exception of subscriber attributes, all other parameters configured using this procedure must be configured in the same destination context on the system as the LAC service.

- Step 1** Configure one or more transform sets according to the instructions located in the [Transform Set Configuration](#) section of this chapter.
- Step 2** Configure one or more ISAKMP policies according to the instructions located in the [ISAKMP Policy Configuration](#) section of this chapter.
- Step 3** Configure an ipsec-isakmp crypto map according to the instructions located in the [Dynamic Crypto Map Configuration](#) section of this chapter.
- Step 4** Configure the subscriber profile attributes according to the instructions located in the [Subscriber Attributes for L2TP Application IPsec Support](#) section of this chapter.
- Step 5** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

How IPsec is Used for PDSN Compulsory L2TP Configurations

The following figure and the text that follows describe how IPsec-encrypted PDSN compulsory L2TP sessions are processed by the system.

Figure 42. PDSN Compulsory L2TP, IPsec-Encrypted Session Processing



335243

Table 41. PDSN Compulsory L2TP, IPsec-Encrypted Session Processing

Step	Description
1.	A subscriber session arrives at a PDSN service on the system that is configured to perform compulsory tunneling. The system uses the LAC service specified in the PDSN service's configuration.
2.	The LAC service dictates the peer LNS to use and also specifies the following parameters indicating that IP security is also required: <ul style="list-style-type: none"> • Crypto map name • ISAKMP secret
3.	The system determines that the crypto map name supplied matches a configured crypto map.

Step	Description
4.	From the crypto map, the system determines the following: <ul style="list-style-type: none"> • The map type, in this case dynamic • Whether perfect forward secrecy (PFS) should be enabled for the IPsec SA and if so, what group should be used • IPsec SA lifetime parameters • The name of one or more configured transform set defining the IPsec SA
5.	To initiate the IKE SA negotiation, the system performs a Diffie-Hellman exchange of the ISAKMP secret specified by the attribute with the specified peer LNS/security gateway.
6.	The system and the LNS/security gateway negotiate an ISAKMP policy (IKE SA) to use to protect further communications.
7.	Once the IKE SA has been negotiated, the system negotiates an IPsec SA with the LNS/security gateway.
8.	Once the IPsec SA has been negotiated, the system protects the L2TP encapsulated data according to the rules specified in the transform set and sends it over the IPsec tunnel.

Configuring Support for L2TP PDSN Compulsory Tunneling with IPsec

This section provides a list of the steps required to configure IPsec functionality on the system in support of PDSN compulsory L2TP tunneling. Each step listed refers to a different section containing the specific instructions for completing the required procedure.



Important: These instructions assume that the system was previously configured to support PDSN compulsory tunneling subscriber data sessions. In addition, all parameters configured using this procedure must be configured in the same destination context on the system as the LAC service.

- Step 1** Configure one or more transform sets according to the instructions located in the [Transform Set Configuration](#) section of this chapter.
- Step 2** Configure one or more ISAKMP policies according to the instructions located in the [ISAKMP Policy Configuration](#) section of this chapter.
- Step 3** Configure an ipsec-isakmp crypto map according to the instructions located in the [Dynamic Crypto Map Configuration](#) section of this chapter.
- Step 4** Configure the subscriber profile attributes according to the instructions located in the [Subscriber Attributes for L2TP Application IPsec Support](#) section of this chapter.
- Step 5** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

How IPSec is Used for L2TP Configurations on the GGSN

The following figure and the text that follows describe how IPSec-encrypted attribute-based L2TP sessions are processed by the system.

Figure 43. GGSN PDP Context Processing with IPSec-Encrypted L2TP

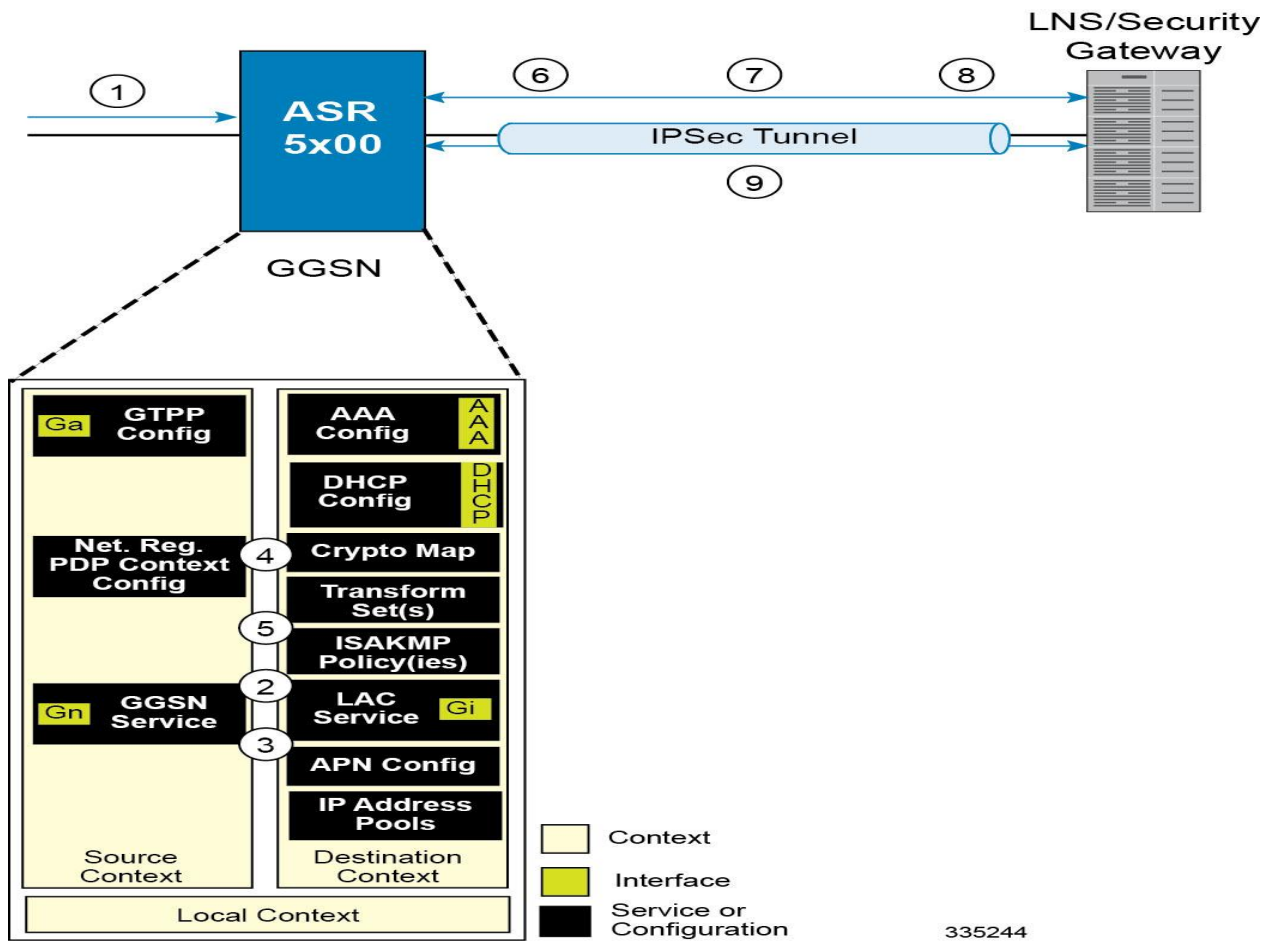


Table 42. GGSN PDP Context Processing with IPSec-Encrypted L2TP

Step	Description
1.	A subscriber session/PDP Context Request arrives at the system.
2.	The configuration of the APN accessed by the subscriber indicates that session data is to be tunneled using L2TP. In addition, attributes specifying a crypto map name and ISAKMP secret are also supplied indicating that IP security is also required.
3.	The system determines that the crypto map name supplied matches a configured crypto map.

Step	Description
4.	From the crypto map, the system determines the following: <ul style="list-style-type: none"> • The map type, in this case dynamic • Whether perfect forward secrecy (PFS) should be enabled for the IPsec SA and if so, what group should be used • IPsec SA lifetime parameters • The name of one or more configured transform set defining the IPsec SA
5.	To initiate the IKE SA negotiation, the system performs a Diffie-Hellman exchange of the ISAKMP secret specified in the profile attribute with the specified peer LNS/security gateway.
6.	The system and the LNS/security gateway negotiate an ISAKMP (IKE) policy to use to protect further communications.
7.	Once the IKE SA has been negotiated, the system negotiates an IPsec SA with the LNS/security gateway using the transform method specified in the transform sets.
8.	Once the IPsec SA has been negotiated, the system protects the L2TP encapsulated data according to the IPsec SAs established during step 9 and sends it over the IPsec tunnel.

Configuring GGSN Support for L2TP Tunneling with IPsec

This section provides a list of the steps required to configure the GGSN to encrypt L2TP tunnels using IPSEC. Each step listed refers to a different section containing the specific instructions for completing the required procedure.




Important: These instructions assume that the system was previously configured to support subscriber PDP contexts and L2TP tunneling either as a GGSN. In addition, all parameters configured using this procedure must be configured in the same destination context on the system as the LAC service.

- Step 1** Configure one or more transform sets according to the instructions located in the [Transform Set Configuration](#) section of this chapter.
- Step 2** Configure one or more ISAKMP policies according to the instructions located in the [ISAKMP Policy Configuration](#) section of this chapter.
- Step 3** Configure an ipsec-isakmp crypto map according to the instructions located in the [Dynamic Crypto Map Configuration](#) section of this chapter.
- Step 4** Configure APN support for encrypting L2TP tunnels using IPsec according to the instructions located in the [APN Template Configuration to Support L2TP](#) section of this chapter.
- Step 5** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Transform Set Configuration

This section provides instructions for configuring transform sets on the system.

 **Important:** This section provides the minimum instruction set for configuring transform set on your system. For more information on commands that configure additional parameters and options, refer to the *Context Configuration Mode Commands* and *Crypto Transform Configuration Mode* chapters in the *Command Line Interface Reference*.

To configure the crypto transform set for IPSec:

- Step 1** Configure crypto transform set by applying the example configuration in the [Configuring Transform Set](#) section.
- Step 2** Verify your Crypto Transform Set configuration by following the steps in the [Verifying the Crypto Transform Set Configuration](#) section.
- Step 3** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Configuring Transform Set

Use the following example to create the crypto transform set on your system:

```
configure

context <ctxt_name>

    crypto ipsec transform-set <transform_name> ah hmac { md5-96 | none | sha1-96 } esp
hmac { { md5-96 | none | sha1-96 } { cipher {des-cbc | 3des-cbc | aes-cbc } | none }

    mode { transport | tunnel }

end
```

Notes:

- *<ctxt_name>* is the system context in which you wish to create and configure the crypto transform set(s).
- *<transform_name>* is the name of the crypto transform set in the current context that you want to configure for IPSec configuration.
- For more information on parameters, refer to the *IPSec Transform Configuration Mode Commands* chapter in the *Command Line Interface Reference*.

Verifying the Crypto Transform Set Configuration

These instructions are used to verify the crypto transform set(s) was/were configured.

- Step 1** Verify that your header crypto transform set configurations by entering the following command in Exec Mode in specific context:

```
show crypto transform-set transform_name
```

This command produces an output similar to that displayed below using the configuration of a transform set named test1.

```
Transform-Set test1 :  
  
AH : none  
  
ESP : hmac md5-96, 3des-cbc  
  
Encaps Mode: TUNNEL
```

ISAKMP Policy Configuration

This section provides instructions for configuring ISAKMP policies on the system. ISAKMP policy configuration is only required if the crypto map type is either ISAKMP or Dynamic.



Important: This section provides the minimum instruction set for configuring ISAKMP policies on the system. For more information on commands that configure additional parameters and options, refer to the *Context Configuration Mode Commands* and *ISAKMP Configuration Mode Commands* chapters in the *Command Line Interface Reference*.

To configure the ISAKMP policy for IPSec:

- Step 1** Configure crypto transform set by applying the example configuration in the [Configuring ISAKMP Policy](#) section.
- Step 2** Verify your ISAKMP policy configuration by following the steps in the [Verifying the ISAKMP Policy Configuration](#) section.
- Step 3** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Configuring ISAKMP Policy

Use the following example to create the ISAKMP policy on your system:

```
configure

context <ctxt_name>

    ikev1 policy <priority>

        encryption { 3des-cbc | des-cbc }

        hash { md5 | sha1 }

        group { 1 | 2 | 3 | 4 | 5 }

        lifetime <time>

    end
```

Notes:

- <ctxt_name> is the system context in which you wish to create and configure the ISAKMP policy.
- <priority> dictates the order in which the ISAKMP policies are proposed when negotiating IKE SAs.
- For more information on parameters, refer to the *ISAKMP Configuration Mode Commands* chapter in the *Command Line Interface Reference*.

Verifying the ISAKMP Policy Configuration

These instructions are used to verify the ISAKMP policy configuration.

Step 1 Verify that your ISAKMP policy configuration by entering the following command in Exec Mode in specific context:

```
show crypto isakmp policy priority
```

This command produces an output similar to that displayed below that displays the configuration of an ISAKMP policy with priority 1.

```
1 ISAKMP Policies are configured

Priority : 1

Authentication Method : preshared-key

Lifetime : 120 seconds

IKE group : 5

hash : md5

encryption : 3des-cbc
```



Caution: Modification(s) to an existing ISAKMP policy configuration will not take effect until the related security association has been cleared. Refer to the **clear crypto security-association** command located in the *Exec Mode Commands* chapter of the *Command Line Interface Reference* for more information.

ISAKMP Crypto Map Configuration

This section provides instructions for configuring ISAKMP crypto maps.



Important: This section provides the minimum instruction set for configuring ISAKMP crypto maps on the system. For more information on commands that configure additional parameters and options, refer to the *Context Configuration Mode Commands* and *Crypto Map ISAKMP Configuration Mode* chapters in the *Command Line Interface Reference*.

To configure the ISAKMP crypto maps for IPSec:

- Step 1** Configure ISAKMP crypto map by applying the example configuration in the [Configuring ISAKMP Crypto Maps](#) section.
- Step 2** Verify your ISAKMP crypto map configuration by following the steps in the [Verifying the ISAKMP Crypto Map Configuration](#) section.
- Step 3** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Configuring ISAKMP Crypto Maps

Use the following example to create the ISAKMP crypto map on your system:

```
configure

context <ctxt_name>

    crypto map <map_name> ipsec-isakmp

        set peer <agw_address>

        set isakmp preshared-key <isakmp_key>

        set mode { aggressive | main }

        set pfs { group1 | group2 | group5 }

        set transform-set <transform_name>

        match address <acl_name> [ preference ]

        match crypto-group <group_name> { primary | secondary }

    end
```

Notes:

- <ctxt_name> is the system context in which you wish to create and configure the ISAKMP crypto maps.
- <map_name> is name by which the ISAKMP crypto map will be recognized by the system.

- `<acl_name>` is name of the pre-configured ACL. It is used for configurations not implementing the IPsec Tunnel Failover feature and match the crypto map to a previously defined crypto ACL. This is an optional parameter.
- `<group_name>` is name of the Crypto group configured in the same context. It is used for configurations using the IPsec Tunnel Failover feature. This is an optional parameter. For more information, refer to the [Redundant IPsec Tunnel Fail-Over](#) section of this chapter.
- For more information on parameters, refer to the *Crypto Map ISAKMP Configuration Mode Commands* chapter in the *Command Line Interface Reference*.

Verifying the ISAKMP Crypto Map Configuration

These instructions are used to verify the ISAKMP crypto map configuration.

- Step 1** Verify that your ISAKMP crypto map configurations by entering the following command in Exec Mode in specific context:

```
show crypto map [ tag map_name | type ipsec-isakmp ]
```

This command produces an output similar to that displayed below that displays the configuration of a crypto map named test_map2.

```
Map Name : test_map2

=====

Payload :

crypto_acl2: permit tcp host 10.10.2.12 neq 35 any

Crypto map Type : ISAKMP

IKE Mode : MAIN

IKE pre-shared key : 3fd32rf09svc

Perfect Forward Secrecy : Group2

Hard Lifetime :

28800 seconds

4608000 kilobytes

Number of Transforms: 1

Transform : test1


AH : none

ESP: md5 3des-cbc

Encaps mode: TUNNEL
```


Local Gateway: Not Set

Remote Gateway: 192.168.1.1

 **Caution:** Modification(s) to an existing ISAKMP crypto map configuration will not take effect until the related security association has been cleared. Refer to the **clear crypto security-association** command located in the *Exec Mode Commands* chapter of the *Command Line Interface Reference* for more information.

Dynamic Crypto Map Configuration

This section provides instructions for configuring dynamic crypto maps. Dynamic crypto maps should only be configured in support of L2TP or Mobile IP applications.



Important: This section provides the minimum instruction set for configuring dynamic crypto maps on the system. For more information on commands that configure additional parameters and options, refer to the *Context Configuration Mode Commands* and *Crypto Map Dynamic Configuration Mode* chapters in the *Command Line Interface Reference*.

To configure the dynamic crypto maps for IPsec:

- Step 1** Configure dynamic crypto maps by applying the example configuration in the [Configuring Dynamic Crypto Maps](#) section.
- Step 2** Verify your dynamic crypto map configuration by following the steps in the [Verifying the Dynamic Crypto Map Configuration](#) section.
- Step 3** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Configuring Dynamic Crypto Maps

Use the following example to create the crypto transform set on your system:

```
configure

context <ctxt_name>

    crypto map <map_name> ipsec-dynamic

        set pfs { group1 | group2 | group5 }

        set transform-set <transform_name>

    end
```

Notes:

- <ctxt_name> is the system context in which you wish to create and configure the dynamic crypto maps.
- <map_name> is name by which the dynamic crypto map will be recognized by the system.
- For more information on parameters, refer to the *Crypto Map Dynamic Configuration Mode Commands* chapter in the *Command Line Interface Reference*.

Verifying the Dynamic Crypto Map Configuration

These instructions are used to verify the dynamic crypto map configuration.

- Step 1** Verify that your dynamic crypto map configurations by entering the following command in Exec Mode in specific context:

```
show crypto map [ tag map_name | type ipsec-dynamic ]
```

This command produces an output similar to that displayed below using the configuration of a dynamic crypto map named test_map3.

```
Map Name : test_map3

=====

Crypto map Type : ISAKMP (Dynamic)

IKE Mode : MAIN

IKE pre-shared key :

Perfect Forward Secrecy : Group2

Hard Lifetime :

28800 seconds

4608000 kilobytes

Number of Transforms: 1

Transform : test1

AH : none

ESP: md5 3des-cbc

Encaps mode: TUNNEL

Local Gateway: Not Set

Remote Gateway: Not Set
```



Caution: Modification(s) to an existing dynamic crypto map configuration will not take effect until the related security association has been cleared. Refer to the **clear crypto security-association** command located in the *Exec Mode Commands* chapter of the *Command Line Interface Reference* for more information.

Manual Crypto Map Configuration

This section provides instructions for configuring manual crypto maps on the system.



Important: Because manual crypto map configurations require the use of static security keys (associations), they are not as secure as crypto maps that rely on dynamically configured keys. Therefore, it is recommended that they only be configured and used for testing purposes.



Important: This section provides the minimum instruction set for configuring manual crypto maps on the system. For more information on commands that configure additional parameters and options, refer to the *Context Configuration Mode Commands* and *Crypto Map Manual Configuration Mode* chapters in the *Command Line Interface Reference*.

To configure the manual crypto maps for IPSec:

- Step 1** Configure manual crypto map by applying the example configuration in the [Configuring Manual Crypto Maps](#) section.
- Step 2** Verify your manual crypto map configuration by following the steps in the [Verifying the Manual Crypto Map Configuration](#) section.
- Step 3** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Configuring Manual Crypto Maps

Use the following example to create the manual crypto map on your system:

configure

```
context <ctxt_name>

  crypto map <map_name> ipsec-manual

    set peer <agw_address>

    match address <acl_name> [ preference ]

    set transform-set <transform_name>

    set session-key { inbound | outbound } { ah <ah_spi> [ encrypted ] key <ah_key>
| esp <esp_spi> [ encrypted ] cipher <encryption_key> [ encrypted ] authenticator
<auth_key> }

  end
```

Notes:

- <ctxt_name> is the system context in which you wish to create and configure the manual crypto maps.

- `<map_name>` is name by which the manual crypto map will be recognized by the system.
- `<acl_name>` is name of the pre-configured ACL. It is used for configurations not implementing the IPsec Tunnel Failover feature and match the crypto map to a previously defined crypto ACL. This is an optional parameter.
- The length of the configured key must match the configured algorithm.
- `<group_name>` is name of the Crypto group configured in the same context. It is used for configurations using the IPsec Tunnel Failover feature. This is an optional parameter.
- For more information on parameters, refer to the *Crypto Map Manual Configuration Mode Commands* chapter in the *Command Line Interface Reference*.

Verifying the Manual Crypto Map Configuration

These instructions are used to verify the manual crypto map configuration.

- Step 1** Verify that your manual crypto map configurations by entering the following command in Exec Mode in specific context:

```
show crypto map [ tag map_name | type ipsec-manual ]
```

This command produces an output similar to that displayed below that displays the configuration of a crypto map named `test_map`.

```
Map Name : test_map

=====

Payload :

crypto_acl1: permit tcp host 1.2.3.4 gt 30 any

Crypto map Type : manual(static)

Transform : test1

Encaps mode: TUNNEL

Transmit Flow

Protocol : ESP

SPI : 0x102 (258)

Hmac : md5, key: 23d32d23cs89

Cipher : 3des-cbc, key: 1234asd3c3d

Receive Flow

Protocol : ESP

SPI : 0x101 (257) Hmac : md5, key: 008j90u3rjp
```

Cipher : 3des-cbc, key: sdfsdffasdf342d32

Local Gateway: Not Set

Remote Gateway: 192.168.1.40



Caution: Modification(s) to an existing manual crypto map configuration will not take effect until the related security association has been cleared. Refer to the **clear crypto security-association** command located in the *Exec Mode Commands* chapter of the *Command Line Interface Reference* for more information.

Crypto Map and Interface Association

This section provides instructions for applying manual or ISAKMP crypto maps to interfaces configured on the system. Dynamic crypto maps should not be applied to interfaces.



Important: This section provides the minimum instruction set for applying manual or ISAKMP crypto maps to an interface on the system. For more information on commands that configure additional parameters and options, refer to the *Command Line Interface Reference*.

To apply the crypto maps to an interface:

- Step 1** Configure a manual or ISAKMP crypto map by applying the example configuration in any of the following sections:
- Step 2** Apply desired crypto map to system interface by following the steps in the [Applying Crypto Map to an Interface](#) section
- Step 3** Verify your manual crypto map configuration by following the steps in the [Verifying the Interface Configuration with Crypto Map](#) section.
- Step 4** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Applying Crypto Map to an Interface

Use the following example to apply an existing crypto map to an interface on your system:

configure

```
context <ctxt_name>

  interface <interface_name>

    crypto-map <map_name>

  end
```

Notes:

- <ctxt_name> is the system context in which the interface is configured to apply crypto map.
- <interface_name> is the name of a specific interface configured in the context to which the crypto map will be applied.
- <map_name> is name of the preconfigured ISAKMP or a manual crypto map.

Verifying the Interface Configuration with Crypto Map

These instructions are used to verify the interface configuration with crypto map.

- Step 1** Verify that your interface is configured properly with crypto map by entering the following command in Exec Mode in specific context:

```
show configuration context ctxt_name | grep interface
```

The interface configuration aspect of the display should look similar to that shown below. In this example an interface named 20/6 was configured with a crypto map called isakmp_map1.

```
interface 20/6


ip address 192.168.4.10 255.255.255.0

crypto-map isakmp_map1
```


FA Services Configuration to Support IPSec

This section provides instructions for configuring FA services to support IPSec.

These instructions assume that the FA service was previously configured and system is ready to serve as an FA.

 **Important:** This section provides the minimum instruction set for configuring an FA service to support IPSec on the system. For more information on commands that configure additional parameters and options, refer to the *Command Line Interface Reference*.

To configure the FA service to support IPSec:

- Step 1** Modify FA service configuration by following the steps in the [Modifying FA service to Support IPSec](#) section
- Step 2** Verify your FA service configuration by following the steps in the [Verifying the FA Service Configuration with IPSec](#) section.
- Step 3** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Modifying FA service to Support IPSec

Use the following example to modify FA service to support IPSec on your system:

configure

```
context <ctxt_name>

    fa-service <fa_svc_name>

        isakmp peer-ha <ha_address> crypto-map <map_name> [ secret <presared_secret> ]

        isakmp default crypto-map <map_name> [ secret <presared_secret> ]

    end
```

Notes:

- <ctxt_name> is the system context in which the FA service is configured to support IPSec.
- <fa_svc_name> is name of the FA service for which you are configuring IPSec.
- <ha_address> is IP address of the HA service to which FA service will communicate on IPSec.
- <map_name> is name of the preconfigured ISAKMP or a manual crypto map.
- A default crypto map for the FA service to be used in the event that the AAA server returns an HA address that is not configured as an ISAKMP peer HA.
- For maximum security, the default crypto map should be configured in addition to peer-ha crypto maps instead of being used to provide IPSec SAs to all HAs. Note that once an IPSec tunnel is established between the FA and HA for a particular subscriber, all new Mobile IP sessions using the same FA and HA are passed over the

tunnel regardless of whether or not IPSec is supported for the new subscriber sessions. Data for existing Mobile IP sessions is unaffected.

Verifying the FA Service Configuration with IPSec

These instructions are used to verify the FA service to support IPSec.

- Step 1** Verify that your FA service is configured properly with IPSec by entering the following command in Exec Mode in specific context:

```
show fa-service { name service_name | all }
```

The output of this command is a concise listing of FA service parameter settings configured on the system.

HA Service Configuration to Support IPSec

This section provides instructions for configuring HA services to support IPSec.

These instructions assume that the HA service was previously configured and system is ready to serve as an HA.



Important: This section provides the minimum instruction set for configuring an HA service to support IPSec on the system. For more information on commands that configure additional parameters and options, refer to the *Command Line Interface Reference*.

To configure the HA service to support IPSec:

- Step 1** Modify HA service configuration by following the steps in the [Modifying HA service to Support IPSec](#) section
- Step 2** Verify your HA service configuration by following the steps in the [Verifying the HA Service Configuration with IPSec](#) section.
- Step 3** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Modifying HA service to Support IPSec

Use the following example to modify an existing HA service to support IPSec on your system:

configure

```
context <ctxt_name>

  ha-service <ha_svc_name>

    isakmp aaa-context <aaa_ctxt_name>

    isakmp peer-fa <fa_address> crypto-map <map_name> [ secret <presared_secret> ]

  end
```

Notes:

- <ctxt_name> is the system context in which the FA service is configured to support IPSec.
- <ha_svc_name> is name of the HA service for which you are configuring IPSec.
- <fa_address> is IP address of the FA service to which HA service will communicate on IPSec.
- <aaa_ctxt_name> name of the context through which the HA service accesses the HAAA server to fetch the IKE S Key and S Lifetime parameters.
- <map_name> is name of the preconfigured ISAKMP or a manual cryptot map.

Verifying the HA Service Configuration with IPSec

These instructions are used to verify the HA service to support IPSec.

- Step 1** Verify that your HA service is configured properly with IPSec by entering the following command in Exec Mode in specific context:

```
show ha-service { name service_name | all }
```

The output of this command is a concise listing of HA service parameter settings configured on the system.

RADIUS Attributes for IPSec-based Mobile IP Applications

As described in the [How the IPSec-based Mobile IP Configuration Works](#) section of this chapter, the system uses attributes stored in a subscriber's RADIUS profile to determine how IPSec should be implemented.

The table below lists the attributes that must be configured in the subscriber's RADIUS attributes to support IPSec for Mobile IP. These attributes are contained in the following dictionaries:


- 3GPP2
- 3GPP2-835
- Starent
- Starent-835
- Starent-VSA1
- Starent-VSA1-835

Table 43. Attributes Used for Mobile IP IPSec Support


Attribute	Description	Variable
3GPP2-Security-Level	This attribute indicates the type of security that the home network mandates on the visited network.	Integer value: 3 : Enables IPSec for tunnels and registration messages 4 : Disables IPSec
3GPP2 - KeyId	This attribute contains the opaque IKE Key Identifier for the FA/HA shared IKE secret.	Supported value for the first eight bytes is the network-order FA IP address in hexadecimal characters. Supported value for the next eight bytes is the network-order HA IP address in hexadecimal characters. Supported value for the final four bytes is a timestamp in network order, indicating when the key was created, and is the number of seconds since January 1, 1970, UTC.
3GPP2-IKE-Secret	This attribute contains the FA/HA shared secret for the IKE protocol. This attribute is salt-encrypted.	A binary string of 1 to 127 bytes.
3GPP2-S	This attribute contains the 'S' secret parameter used to make the IKE pre-shared secret.	A binary string of the value of 'S' consisting of 1 to 127 characters.
3GPP2- S-Lifetime	This attribute contains the lifetime of the 'S' secret parameter used to make the IKE pre-shared secret.	An integer in network order, indicating the time in seconds since January 1, 1970 00:00 UTC. Note that this is equivalent to the Unix operating system expression of time.

LAC Service Configuration to Support IPSec

This section provides instructions for configuring LAC services to support IPSec.

 **Important:** These instructions are required for compulsory tunneling. They should only be performed for attribute-based tunneling if the Tunnel-Service-Endpoint, the SN1-Tunnel-ISAKMP-Crypto-Map, or the SN1 -Tunnel-ISAKMP-Secret are not configured in the subscriber profile.

These instructions assume that the LAC service was previously configured and system is ready to serve as an LAC server.

 **Important:** This section provides the minimum instruction set for configuring an LAC service to support IPSec on the system. For more information on commands that configure additional parameters and options, refer to the *Command Line Interface Reference*.

To configure the LAC service to support IPSec:

- Step 1** Modify LAC service configuration by following the steps in the [Modifying LAC service to Support IPSec](#) section.
- Step 2** Verify your LAC service configuration by following the steps in the [Verifying the LAC Service Configuration with IPSec](#) section.
- Step 3** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Modifying LAC service to Support IPSec

Use the following example to modify an existing LAC service to support IPSec on your system:

configure

```

context <ctxt_name>

    lac-service <lac_svc_name>

        peer-lns <ip_address> [encrypted] secret <secret> [crypto-map <map_name> {
[encrypted] isakmp-secret <secret> } ] [ description <text> ] [ preference <integer>]

        isakmp aaa-context <aaa_ctxt_name>

        isakmp peer-fa <fa_address> crypto-map <map_name> [ secret <preshared_secret> ]

    end

```

Notes:

- <ctxt_name> is the destination context where the LAC service is configured to support IPSec.

- *<lac_svc_name>* is name of the LAC service for which you are configuring IPSec.
- *<lns_address>* is IP address of the LNS node to which LAC service will communicate on IPSec.
- *<aaa_ctxt_name>* name of the context through which the HA service accesses the HAAA server to fetch the IKE S Key and S Lifetime parameters.
- *<map_name>* is name of the preconfigured ISAKMP or a manual cryptot map.

Verifying the LAC Service Configuration with IPSec

These instructions are used to verify the LAC service to support IPSec.

- Step 1** Verify that your LAC service is configured properly with IPSec by entering the following command in Exec Mode in specific context:

```
show lac-service nameservice_name
```

The output of this command is a concise listing of LAC service parameter settings configured on the system.

Subscriber Attributes for L2TP Application IPsec Support

In addition to the subscriber profile attributes listed in the *RADIUS and Subscriber Profile Attributes Used* section of the *L2TP Access Concentrator* chapter in this guide, the table below lists the attributes required to support IPsec for use with attribute-based L2TP tunneling.

These attributes are contained in the following dictionaries:

- Starent
- Starent-835

Table 44. Subscriber Attributes for IPsec encrypted L2TP Support

RADIUS Attribute	Local SubscriberAttribute	Description	Variable
SN1-Tunnel- ISAKMP- Crypto-Map	tunnel l2tp crypto-map	The name of a crypto map configured on the system.	A salt-encrypted ascii string specifying the crypto-map to use for this subscriber. It can be tagged, in which case it is treated as part of a tunnel group.
SN1 -Tunnel- ISAKMP- Secret	tunnel l2tp crypto-map isakmp-secret	The pre-shared secret that will be used as part of the D-H exchange to negotiate an IKE SA.	A salt-encrypted string specifying the IKE secret. It can be tagged, in which case it is treated as part of a tunnel group.

PDSN Service Configuration for L2TP Support

PDSN service configuration is required for compulsory tunneling and optional for attribute-based tunneling.

For attribute-based tunneling, a configuration error could occur such that upon successful authentication, the system determines that the subscriber session requires L2TP but can not determine the name of the context in which the appropriate LAC service is configured from the attributes supplied. As a precautionary, a parameter has been added to the PDSN service configuration options that will dictate the name of the context to use. It is strongly recommended that this parameter be configured.

This section contains instructions for modifying the PDSN service configuration for either compulsory or attribute-based tunneling.

These instructions assume that the PDSN service was previously configured and system is ready to serve as a PDSN.

This section provides the minimum instruction set for configuring an L2TP service on the PDSN system. For more information on commands that configure additional parameters and options, refer to the *Command Line Interface Reference*.

To configure the PDSN service to support L2TP:

- Step 1** Modify PDSN service to configure compulsory tunneling or attribute-based tunneling by applying the example configuration in any of the following sections:
- [Modifying PDSN service to Support Attribute-based L2TP Tunneling](#)
 - [Modifying PDSN service to Support Compulsory L2TP Tunneling](#)
- Step 2** Verify your LAC service configuration by following the steps in the [Verifying the PDSN Service Configuration for L2TP](#) section.
- Step 3** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Modifying PDSN service to Support Attribute-based L2TP Tunneling

Use the following example to modify an existing PDSN service to support attribute-based L2TP tunneling on your system:

```
configure

context <ctxt_name>

    pdsn-service <pdsn_svc_name>

    ppp tunnel-context <lac_ctxt_name>

end
```

Notes:

- <ctxt_name> is the destination context where the PDSN service is configured.

- `<pdsn_svc_name>` is name of the PDSN service for which you are configuring attribute-based L2TP tunneling.
- `<lac_ctxt_name>` is the name of the destination context where the LAC service is located.

Modifying PDSN service to Support Compulsory L2TP Tunneling

Use the following example to modify an existing PDSN service to support compulsory L2TP tunneling on your system:

configure

```
context <ctxt_name>

  pdsn-service <pdsn_svc_name>

    ppp tunnel-context <lac_ctxt_name>

    ppp tunnel-type l2tp

  end
```

Notes:

- `<ctxt_name>` is the destination context where the PDSN service is configured.
- `<pdsn_svc_name>` is name of the PDSN service for which you are configuring attribute-based L2TP tunneling.
- `<lac_ctxt_name>` is name of the destination context where the LAC service is located.

Verifying the PDSN Service Configuration for L2TP

These instructions are used to verify the PDSN service to support L2TP.

Step 1 Verify that your PDSN service is configured properly with L2TP by entering the following command in Exec Mode in specific context:

```
show pdsn-service name service_name
```

The output of this command is a concise listing of PDSN service parameter settings configured on the system.

Redundant IPSec Tunnel Fail-Over

The Redundant IPSec Tunnel Fail-Over functionality is included with the IPSec feature license and allows the configuration of a secondary ISAKMP crypto map-based IPSec tunnel over which traffic is routed in the event that the primary ISAKMP crypto map-based tunnel cannot be used.

This feature introduces the concept of crypto (tunnel) groups when using IPSec tunnels for access to packet data networks (PDNs). A crypto group consists of two configured ISAKMP crypto maps. Each crypto map defines the IPSec policy for a tunnel. In the crypto group, one tunnel serves as the primary, the other as the secondary (redundant). Note that the method in which the system determines to encrypt user data in an IPSec tunnel remains unchanged.

Group tunnels are perpetually maintained with IPSec Dead Peer Detection (DPD) packets exchanged with the peer security gateway.



Important: The peer security gateway must support RFC 3706 in order for this functionality to function properly.

When the system determines that incoming user data traffic must be routed over one of the tunnels in a group, the system automatically uses the primary tunnel until either the peer is unreachable (the IPSec DPD packets cease), or the IPSec tunnel fails to re-key. If the primary peer becomes unreachable, the system automatically begins to switch user traffic to the secondary tunnel. The system can be configured to either automatically switch user traffic back to the primary tunnel once the corresponding peer security gateway is reachable and the tunnel is configured, or require manual intervention to do so.

This functionality also supports the generation of Simple network Management Protocol (SNMP) notifications indicating the following conditions:

- **Primary Tunnel is down:** A primary tunnel that was previously "up" is now "down" representing an error condition.
- **Primary Tunnel is up:** A primary tunnel that was previously "down" is now "up".
- **Secondary tunnel is down:** A secondary tunnel that was previously "up" is now "down" representing an error condition.
- **Secondary Tunnel is up:** A secondary tunnel that was previously "down" is now "up".
- **Fail-over successful:** The switchover of user traffic was successful. This is generated for both primary-to-secondary and secondary-to-primary switchovers.
- **Unsuccessful fail-over:** An error occurred when switching user traffic from either the primary to secondary tunnel or the secondary to primary tunnel.


Supported Standards


Support for the following standards and requests for comments (RFCs) has been added with the Redundant IPSec Tunnel Fail-over functionality:


- RFC 3706, A Traffic-Based Method of Detecting Dead Internet Key Exchange (IKE) Peers, February 2004

Redundant IPSec Tunnel Fail-over Configuration

This section provides information and instructions for configuring the Redundant IPSec Tunnel Fail-over feature. These instructions assume that the system was previously configured to support subscriber data sessions either as a core service or an HA.

 **Important:** Parameters configured using this procedure must be configured in the same context on the system.

 **Important:** The system supports a maximum of 32 crypto groups per context. However, configuring crypto groups to use the same loopback interface for secondary IPSec tunnels is not recommended and may compromise redundancy on the chassis.

 **Important:** This section provides the minimum instruction set for configuring crypto groups on the system. For more information on commands that configure additional parameters and options, refer Command Line Interface Reference.

To configure the Crypto group to support IPSec:

- Step 1** Configure a crypto group by following the steps in the [Configuring Crypto Group](#) section
- Step 2** Configure one or more ISAKMP policies according to the instructions provided in the [ISAKMP Policy Configuration](#) section of this chapter.
- Step 3** Configure IPSec DPD settings using the instructions provided in the [Dead Peer Detection \(DPD\) Configuration](#) section of this chapter.
- Step 4** Configure an ISAKMP crypto map for the primary and secondary tunnel according to the instructions provided in the [ISAKMP Crypto Map Configuration](#) section of this chapter.
- Step 5** Match the existing ISAKMP crypto map to Crypto group by following the steps in the [Modify ISAKMP Crypto Map Configuration to Match Crypto Group](#) section
- Step 6** Verify your Crypto Group configuration by following the steps in the [Verifying the Crypto Group Configuration](#) section.
- Step 7** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Configuring Crypto Group

Use the following example to configure a crypto group on your system for redundant IPSec tunnel fail-over support:

```
configure
```

```
context <ctxt_name>
```

```
ikev1 keepalive dpd interval <dur> timeout <dur> num-retry <retries>
```

```

crypto-group <group_name>

    match address <acl_name> [ <preference> ]

    switchover auto [ do-not-revert ]

end

```

Notes:

- <ctxt_name> is the destination context where the Crypto Group is to be configured.
- <group_name> is name of the Crypto group you want to configure for IPSec tunnel failover support.
- <acl_name> is name of the pre-configured crypto ACL. It is used for configurations not implementing the IPSec Tunnel Failover feature and match the crypto map to a previously defined crypto ACL. For more information on crypto ACL, refer [Crypto Access Control List \(ACL\)](#) section of this chapter.

Modify ISAKMP Crypto Map Configuration to Match Crypto Group

Use the following example to match the crypto group with ISAKMP crypto map on your system:

configure

```

context <ctxt_name>

    crypto map <map_name1> ipsec-isakmp

    match crypto-group <group_name> primary

end

```

configure

```

context <ctxt_name>

    crypto map <map_name> ipsec-isakmp

    match crypto-group <group_name> secondary

end

```

Notes:

- <ctxt_name> is the system context in which you wish to create and configure the ISAKMP crypto maps.
- <group_name> is name of the Crypto group configured in the same context for IPSec Tunnel Failover feature.
- <map_name1> is name of the preconfigured ISAKMP crypto map to match with crypto group as primary.
- <map_name2> is name of the preconfigured ISAKMP crypto map to match with crypto group as secondary.

Verifying the Crypto Group Configuration

These instructions are used to verify the crypto group configuration.

Step 1 Verify that your system is configured properly with crypto group by entering the following command in Exec Mode in specific context:

```
show crypto group [ summary | name group_name ]
```

The output of this command is a concise listing of crypto group parameter settings configured on the system.


Dead Peer Detection (DPD) Configuration


This section provides instructions for configuring the Dead Peer Detection (DPD).

Defined by RFC 3706, Dead Peer Detection (DPD) is used to simplify the messaging required to verify communication between peers and tunnel availability.

DPD is configured at the context level and is used in support of the IPsec Tunnel Failover feature (refer to the [Redundant IPsec Tunnel Fail-Over](#) section) and/or to help prevent tunnel state mismatches between an FA and HA when IPsec is used for Mobile IP applications. When used with Mobile IP applications, DPD ensures the availability of tunnels between the FA and HA. (Note that the starIPSECDynTunUp and starIPSECDynTunDown SNMP traps are triggered to indicate tunnel state for the Mobile IP scenario.)

Regardless of the application, DPD must be supported/configured on both security peers. If the system is configured with DPD but it is communicating with a peer that does not have DPD configured, IPsec tunnels still come up. However, the only indication that the remote peer does not support DPD exists in the output of the **show crypto isakmp security-associations summary** command.

 **Important:** If DPD is enabled while IPsec tunnels are up, it will not take affect until all of the tunnels are cleared.

 **Important:** DPD must be configured in the same context on the system as other IPsec Parameters.

To configure the Crypto group to support IPsec:

- Step 1** Enable dead peer detection on system in support of the IPsec Tunnel Failover feature by following the steps in the [Configuring Crypto Group](#) section
- Step 2** Verify your Crypto Group configuration by following the steps in the [Verifying the DPD Configuration](#) section.
- Step 3** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Configuring Crypto Group

Use the following example to configure a crypto group on your system for redundant IPsec tunnel fail-over support:

configure

```
context <ctxt_name>

  ikev1 keepalive dpd interval <dur> timeout <dur> num-retry <retries>

end
```

Notes:

- <ctxt_name> is the destination context where the Crypto Group is to be configured.

Verifying the DPD Configuration

These instructions are used to verify the dead peer detection configuration.

- Step 1** Verify that your system is configured properly with crypto group with DPD by entering the following command in Exec Mode in specific context:

```
show crypto group [ summary | name group_name ]
```

The output of this command is a concise listing of crypto group parameter settings configured on the system.

APN Template Configuration to Support L2TP

This section provides instructions for adding L2TP support for APN templates configured on the system.

These instructions assume that the APN template was previously configured on this system.



Important: This section provides the minimum instruction set for configuring an APN template to support L2TP for APN. For more information on commands that configure additional parameters and options, refer to the *Command Line Interface Reference*. To configure the APN to support L2TP:

- Step 1** Modify preconfigured APN template by following the steps in the [Modifying APN Template to Support L2TP](#) section
- Step 2** Verify your APN configuration by following the steps in the [Verifying the APN Configuration for L2TP](#) section.
- Step 3** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Modifying APN Template to Support L2TP

Use the following example to modify APN template to support L2TP:

configure

```
context <ctxt_name>

    apn <apn_name>

        tunnel l2tp [ peer-address <lns_address> [ [ encrypted ] secret <l2tp_secret> ]
        [ preference <num> ] [ tunnel-context <tunnel_ctxt_name> ] [ local-address
        <agw_ip_address> ] [ crypto-map <map_name> { [ encrypted ] isakmp-secret <crypto_secret>
        } ]

    end
```

Notes:

- <ctxt_name> is the system context in which the APN template is configured.
- <apn_name> is name of the preconfigured APN template in which you want to configure L2TP support.
- <lns_address> is IP address of the LNS node to which this APN will communicate.
- <tunnel_ctxt_name> is the L2TP context in which the L2TP tunnel is configured.
- <agw_ip_address> is the local IP address of the GGSN in which this APN template is configured.
- <map_name> is the preconfigured crypto map (ISAKMP or manual) which is to use for L2TP.

Verifying the APN Configuration for L2TP

These instructions are used to verify the APN template configuration for L2TP.

IPSec for LTE/SAE Networks

The Cisco MME (Mobility Management Entity), S-GW (Serving Gateway), and P-GW (Packet Data Network Gateway) support IPSec and IKEv2 encryption using IPv4 and IPv6 addressing in LTE/SAE (Long Term Evolution/System Architecture Evolution) networks. IPSec and IKEv2 encryption enables network domain security for all IP packet-switched networks, providing confidentiality, integrity, authentication, and anti-replay protection via secure IPSec tunnels.

Encryption Algorithms

IPSec for LTE/SAE supports the following control and data path encryption algorithms:

- AES-CBC-128 (Advanced Encryption Standard-Cipher Block Chaining-128)
- AES-CBC-256 (Advanced Encryption Standard-Cipher Block Chaining-256)
- DES-CBC (Data Encryption Standard-Cipher Block Chaining)
- 3DES-CBC (Triple Data Encryption Standard-Cipher Block Chaining)

HMAC Functions

IPSec for LTE/SAE supports the following data path HMAC (Hash-based Message Authentication Code) functions:

- AES-XCBC-MAC-96 (Advanced Encryption Standard-X Cipher Block Chaining-Message Authentication Code-96)
- MD5-96 (Message Digest 5-96)
- SHA1-96 (Secure Hash Algorithm 1-96)

IPSec for LTE/SAE supports the following control path HMAC (Hash-based Message Authentication Code) functions:

- AES-XCBC-MAC-96 (Advanced Encryption Standard-X Cipher Block Chaining-Message Authentication Code-96)
- MD5-96 (Message Digest 5-96)
- SHA1-96 (Secure Hash Algorithm 1-96)
- SHA2-256-128 (Secure Hash Algorithm 2-256-128)
- SHA2-384-192 (Secure Hash Algorithm 2-384-192)
- SHA2-512-256 (Secure Hash Algorithm 2-512-256)

Diffie-Hellman Groups

IPSec for LTE/SAE supports the following Diffie-Hellman groups for IKE and Child SAs (Security Associations):

- Diffie-Hellman Group 1: 768-bit MODP (Modular Exponential) Group
- Diffie-Hellman Group 2: 1024-bit MODP Group

- Diffie-Hellman Group 5: 1536-bit MODP Group
- Diffie-Hellman Group 14: 2048-bit MODP Group
- None: No Diffie-Hellman Group (no perfect forward secrecy)

Dynamic Node-to-Node IPSec Tunnels

IPSec for LTE/SAE enables network nodes to initiate an IPSec tunnel with another node for secure signaling and data traffic between the nodes, enabling up to 64K dynamic, service-integrated IPSec tunnels per chassis. Once established, a dynamic node-to-node IPSec tunnel continues to carry all of the signaling and/or bearer traffic between the nodes. Dynamic node-to-node IPSec for LTE/SAE is supported on the S1-MME interface for signaling traffic between the eNodeB and the MME, on the S1-U interface for data traffic between the eNodeB and the S-GW, and on the S5 interface for data traffic between the S-GW and the P-GW.

Dynamic node-to-node IPSec gets configured using dynamic IKEv2 crypto templates, which are used to specify common cryptographic parameters for the IPSec tunnels such as the encryption algorithm, HMAC function, and Diffie-Hellman group. Additional information necessary for creating node-to-node IPSec tunnels such as revocation lists are fetched dynamically from the IPSec tunnel requests.

For configuration instructions for dynamic node-to-node IPSec, see the configuration chapter in the administration guides for the MME, S-GW, and P-GW.

ACL-based Node-to-Node IPSec Tunnels

Node-to-node IPSec for LTE/SAE can also be configured using crypto ACLs (Access Control Lists), which define the matching criteria used for routing subscriber data packets over an IPSec tunnel. ACL-based node-to-node IPSec tunnels are supported on the S1-MME interface for signaling traffic between the eNodeB and the MME, on the S1-U interface for data traffic between the eNodeB and the S-GW, and on the S5 interface for data traffic between the S-GW and the P-GW.

Unlike other ACLs that are applied to interfaces, contexts, or to one or more subscribers, crypto ACLs are applied via matching criteria to crypto maps, which define tunnel policies that determine how IPSec is implemented for subscriber data packets. Prior to routing, the system examines the properties of each subscriber data packet. If the packet properties match the criteria specified in the crypto ACL, the system initiates the IPSec policy dictated by the crypto map. ACL-based node-to-node IPSec tunnels are configured using either IKEv2-IPv4 or IKEv2-IPv6 crypto maps for IPv4 or IPv6 addressing.

Up to 150 ACL-based node-to-node IPSec tunnels are supported on the system, each with one SA bundle that includes one Tx and one Rx endpoint. However, to avoid significant performance degradation, dynamic node-to-node IPSec tunnels are recommended. If ACL-based node-to-node IPSec tunnels are used, a limit of about ten ACL-based node-to-node IPSec tunnels per system is recommended.

For configuration instructions for ACL-based node-to-node IPSec, see the configuration chapter in the administration guides for the MME, S-GW, and P-GW.

For more information on ACLs, see the *System Administration Guide*.

Traffic Selectors

Per RFC 4306, when a packet arrives at an IPSec subsystem and matches a 'protect' selector in its Security Policy Database (SPD), the subsystem must protect the packet via IPSec tunneling. Traffic selectors enable an IPSec subsystem to accomplish this by allowing two endpoints to share information from their SPDs. Traffic selector payloads contain

the selection criteria for packets being sent over IPSec security associations (SAs). Traffic selectors can be created on the P-GW, S-GW, and MME for dynamic node-to-node IPSec tunnels during crypto template configuration by specifying a range of peer IPv4 or IPV6 addresses from which to carry traffic over IPSec tunnels.

For example, consider an eNodeB with an IP address of 1.1.1.1 and an S-GW with a service address of 2.2.2.2. The S-GW is registered to listen for IKE requests from the eNodeBs in the network using the following information:

- Local Address: 2.2.2.2
- Peer Address Network: 1.1.0.0 Mask: 255.255.0.0
- Payload ACL (Access Control List): udp host 2.2.2.2 eq 2123 1.1.0.0 0.0.255.255

When an IKE request arrives the S-GW from eNodeB address 1.1.1.1, the IPSec subsystem converts the payload ACL to: udp host 2.2.2.2 eq 2123 host 1.1.1.1, and this payload becomes the traffic selector for the IPSec tunnel being negotiated.

To properly accommodate control traffic between IPSec nodes, each child SA must include at least two traffic selectors: one with a well-known port in the source address, and one with a well-known port in the destination address. Continuing the example above, the final traffic selectors would be:

- Destination port as well-known port: udp host 2.2.2.2 1.1.0.0 0.0.255.255 eq 2123
- Source port as well-known port: udp host 2.2.2.2 eq 2123 1.1.0.0 0.0.255.255

Note that for ACL-based node-to-node IPSec tunnels, the configured crypto ACL becomes the traffic selector with no modification.

Authentication Methods

IPSec for LTE/SAE includes the following authentication methods:

- **PSK (Pre-Shared Key) Authentication:** A pre-shared key is a shared secret that was previously shared between two network nodes. IPSec for LTE/SAE supports PSK such that both IPSec nodes must be configured to use the same shared secret.
- **X.509 Certificate-based Peer Authentication:** IPSec for LTE/SAE supports X.509 certificate-based peer authentication and CA (Certificate Authority) certificate authentication as described below.

X.509 Certificate-based Peer Authentication

X.509 specifies standard formats for public key certificates, certificate revocation lists, attribute certificates, and a certification path validation algorithm. X.509 certificates are configured on each IPSec node so that it can send the certificate as part of its IKE_AUTH_REQ for the remote node to authenticate it. These certificates can be in PEM (Privacy Enhanced Mail) or DER (Distinguished Encoding Rules) format, and can be fetched from a repository via HTTP or FTP.

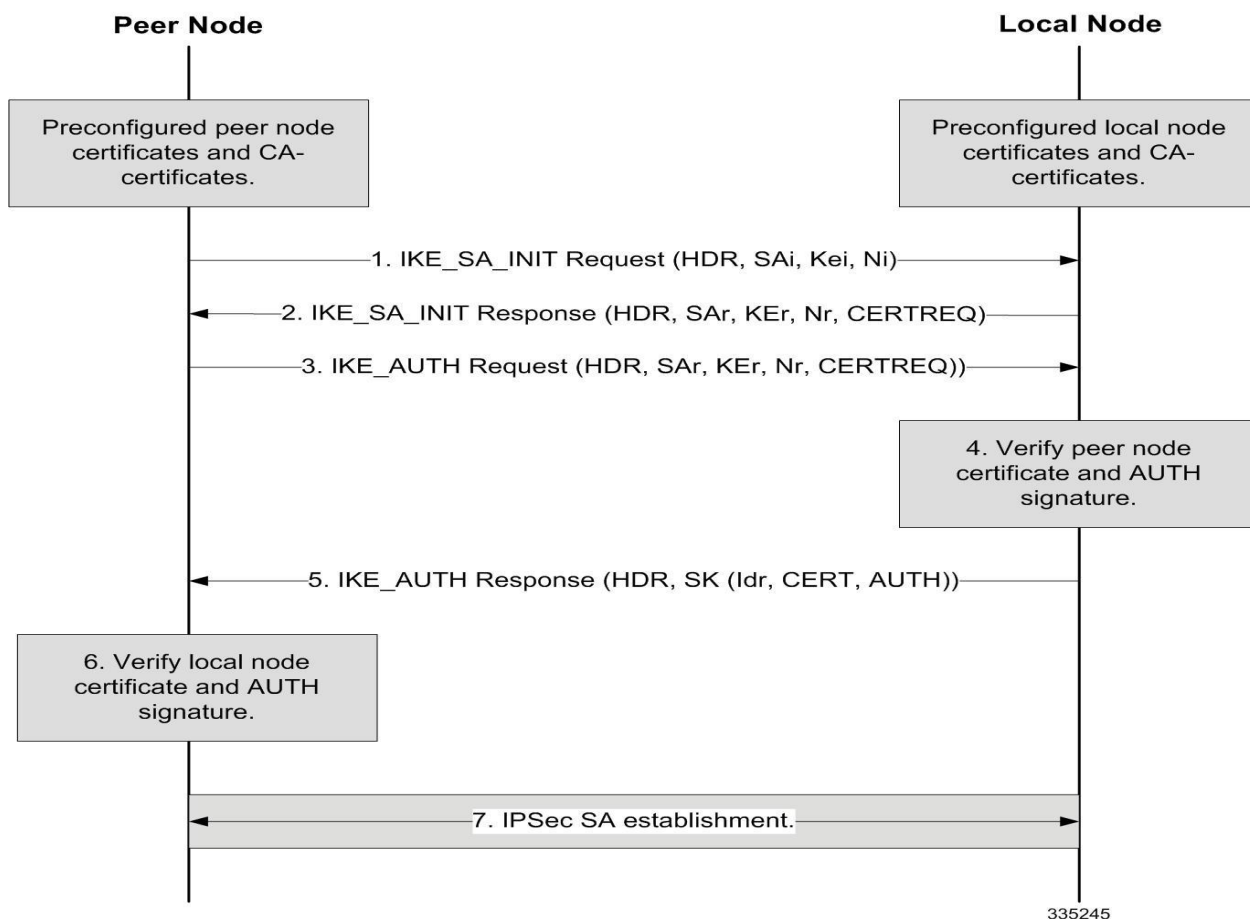
CA certificate authentication is used to validate the certificate that the local node receives from a remote node during an IKE_AUTH exchange.

A maximum of sixteen certificates and sixteen CA certificates are supported per system. One certificate is supported per service, and a maximum of four CA certificates can be bound to one crypto template.

For configuration instructions for X.509 certificate-based peer authentication, see the configuration chapter in the administration guides for the MME, S-GW, and P-GW.

The figure below shows the message flow during X.509 certificate-based peer authentication. The table that follows the figure describes each step in the message flow.

Figure 44. X.509 Certificate-based Peer Authentication



335245

Table 45. X.509 Certificate-based Peer Authentication

Step	Description
1.	The peer node initiates an IKEv2 exchange with the local node, known as the IKE_SA_INIT exchange, by issuing an IKE_SA_INIT Request to negotiate cryptographic algorithms, exchange nonces, and perform a Diffie-Hellman exchange with the local node.
2.	The local node responds with an IKE_SA_INIT Response by choosing a cryptographic suite from the initiator's offered choices, completing the Diffie-Hellman and nonce exchanges with the peer node. In addition, the local node includes the list of CA certificates that it will accept in its CERTREQ payload. For successful peer authentication, the CERTREQ payload must contain at least one CA certificate that is in the trust chain of the peer certificate. At this point in the negotiation, the IKE_SA_INIT exchange is complete and all but the headers of all the messages that follow are encrypted and integrity-protected.
3.	The peer node initiates an IKE_AUTH exchange with the local node by including the IDi payload, setting the CERT payload to the peer certificate, and including the AUTH payload containing the signature of the previous IKE_SA_INIT Request message (in step 1) generated using the private key of the peer certificate. The authentication algorithm used to generate the AUTH payload is also included in the AUTH payload. The peer node also includes the CERTREQ payload containing the list of SHA-1 hash algorithms for local node authentication. For successful server authentication, the CERTREQ payload must contain at least one CA certificate that is in the trust chain of the peer certificate.

Step	Description
4.	Using the CA certificate corresponding to the peer certificate, the local node first verifies that the peer certificate in the CERT payload has not been modified and the identity included in the IDi corresponds to the identity in the peer certificate. If the verification is successful, using the public key of the peer certificate, the local node generates the expected AUTH payload and compares it with the received AUTH payload. If they match, the authentication of the peer node is successful. Otherwise, the local node sends an IKEv2 Notification message indicating authentication failure.
5.	The local node responds with the IKE_AUTH Response, including the IDr payload, setting the CERT payload to the local node certificate, and including the AUTH payload containing the signature of the IKE_SA_INIT Response message (in step 2) generated using the private key of the local node certificate. The authentication algorithm used to generate the AUTH payload is also included in the AUTH payload.
6.	Using the CA certificate corresponding to the local node certificate, the peer node first verifies that the local node certificate in the CERT payload has not been modified. If the verification is successful, using the public key of the local node certificate, the peer generates the expected AUTH payload and compares it with the received AUTH payload. If they match, the local node authentication is successful. This completes the IKE_AUTH exchange.
7.	An IPSec SA gets established between the peer node and the local node. If more IPSec SAs are needed, either the peer or local node can initiate the creation of additional Child SAs using a CREATE_CHILD_SA exchange.

Certificate Revocation Lists

Certificate revocation lists track certificates that have been revoked by the CA (Certificate Authority) and are no longer valid. Per RFC 3280, during certificate validation, IPSec for LTE/SAE checks the certificate revocation list to verify that the certificate the local node receives from the remote node has not expired and hence is still valid.

During configuration via the system CLI, one certificate revocation list is bound to each crypto template and can be fetched from its repository via HTTP or FTP.

Child SA Rekey Support

Rekeying of an IKEv2 Child Security Association (SA) occurs for an already established Child SA whose lifetime (either time-based or data-based) is about to exceed a maximum limit. The IPSec subsystem initiates rekeying to replace the existing Child SA. During rekeying, two Child SAs exist momentarily (500ms or less) to ensure that transient packets from the original Child SA are processed by the IPSec node and not dropped.

Child SA rekeying is disabled by default, and rekey requests are ignored. This feature gets enabled in the Crypto Configuration Payload Mode of the system's CLI.

IKEv2 Keep-Alive Messages (Dead Peer Detection)

IPSec for LTE/SAE supports IKEv2 keep-alive messages, also known as Dead Peer Detection (DPD), originating from both ends of an IPSec tunnel. Per RFC 3706, DPD is used to simplify the messaging required to verify communication between peers and tunnel availability. You configure DPD on each IPSec node. You can also disable DPD, and the node will not initiate DPD exchanges with other nodes. However, the node always responds to DPD availability checks initiated by another node regardless of its DPD configuration.

E-UTRAN/EPC Logical Network Interfaces Supporting IPsec Tunnels

The figure below shows the logical network interfaces over which secure IPsec tunnels can be created in an E-UTRAN/EPC (Evolved UMTS Terrestrial Radio Access Network/Evolved Packet Core) network. The table that follows the figure provides a description of each logical network interface.

Figure 45. E-UTRAN/EPC Logical Network Interfaces Supporting IPsec Tunnels

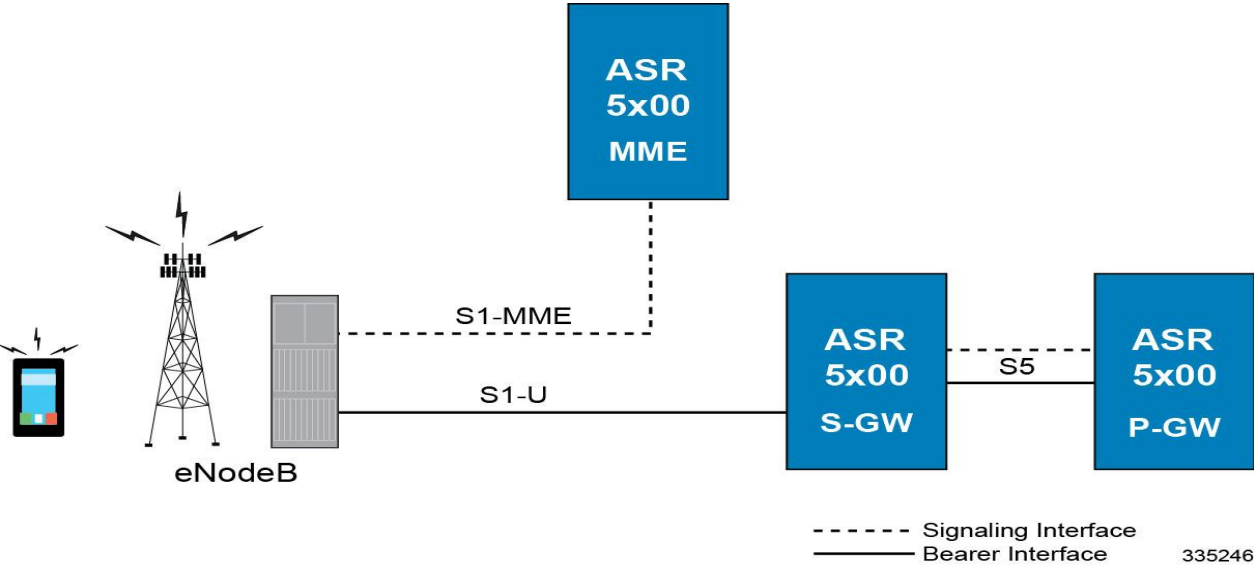


Table 46. E-UTRAN/EPC Logical Network Interfaces Supporting IPsec Tunnels

Interface	Description
S1-MME Interface	<p>This interface is the reference point for the control plane protocol between the eNodeB and the MME. The S1-MME interface uses S1-AP (S1- Application Protocol) over SCTP (Stream Control Transmission Protocol) as the transport layer protocol for guaranteed delivery of signaling messages between the MME and the eNodeB (S1). When configured, the S1-AP over SCTP signaling traffic gets carried over an IPsec tunnel.</p> <p>When a subscriber UE initiates a connection with the eNodeB, the eNodeB initiates an IPsec tunnel with the MME, and SCTP signaling for all subsequent subscriber UEs served by this MME gets carried over the same IPsec tunnel. The MME can also initiate an IPsec tunnel with the eNodeB when the following conditions exist:</p> <ul style="list-style-type: none">• The first tunnel setup is always triggered by the eNodeB. This is the tunnel over which initial SCTP exchanges occur.• The MME initiates additional tunnels to the eNodeB after an SCTP connection is set up if the MME is multi-homed: a tunnel is initiated from MME's second address to the eNodeB.• The eNodeB is multi-homed: tunnels are initiated from the MME's primary address to each secondary address of the eNodeB.• Both of the prior two conditions: a tunnel is initiated from each of MME's addresses to each address of the eNodeB.

Interface	Description
S1-U Interface	This interface is the reference point for bearer channel tunneling between the eNodeB and the S-GW. Typically, the eNodeB initiates an IPSec tunnel with the S-GW over this interface for subscriber data traffic. But the S-GW may also initiate an IPSec tunnel with the eNodeB, if required.
S5 Interface	This interface is the reference point for tunneling between the S-GW and the P-GW. Based on the requested APN from a subscriber UE, the MME selects both the S-GW and the P-GW that the S-GW connects to. GTP-U data traffic is carried over the IPSec tunnel between the S-GW and P-GW for the current and all subsequent subscriber UEs.

IPSec Tunnel Termination

IPSec tunnel termination occurs during the following scenarios:

- **Idle Tunnel Termination:** When a session manager for a service detects that all subscriber sessions using a given IPSec tunnel have terminated, the IPSec tunnel also gets terminated after a timeout period.
- **Service Termination:** When a service running on a network node is brought down for any reason, all corresponding IPSec tunnels get terminated. This may be caused by the interface for a service going down, a service being stopped manually, or a task handling an IPSec tunnel restarting.
- **Unreachable Peer:** If a network node detects an unreachable peer via Dead Peer Detection (DPD), the IPSec tunnel between the nodes gets terminated. DPD can be enabled per P-GW, S-GW, and MME service via the system CLI during crypto template configuration.
- **E-UTRAN Handover Handling:** Any IPSec tunnel that becomes unusable due to an E-UTRAN network handover gets terminated, while the network node to which the session is handed initiates a new IPSec tunnel for the session.

IPSec for Femto-UMTS Networks

The Cisco HNB-GW (Home-NodeB Gateway) supports IPSec and IKEv2 encryption using IPv4 addressing in Femto-UMTS. IPSec and IKEv2 encryption enables network domain security for all IP packet-switched networks, providing confidentiality, integrity, authentication, and anti-replay protection via secure IPSec tunnels.

Authentication Methods

IPSec for Femto-UMTS includes the following authentication methods:

- **PSK (Pre-Shared Key) Authentication:** A pre-shared key is a shared secret that was previously shared between two network nodes. IPSec for Femto-UMTS supports PSK such that both IPSec nodes must be configured to use the same shared secret.
- **X.509 Certificate-based Peer Authentication:** IPSec for Femto-UMTS supports X.509 certificate-based peer authentication and CA (Certificate Authority) certificate authentication as described below.

Crypto map Template Configuration

Use the following example to configure the IPsec profile and Crypto map template to associate with SeGW and enabling IPsec tunneling.

```
configure

context <vpn_ctxt_name>

    eap-profile <eap_prof_name>

    mode authentication-pass-through

    exit

    ip pool ipsec <ip_address> <subnetmask>

    ipsec transform-set <ipsec_trans_set>

    exit

    ikev2 transform-set <ikev2_trans_set>

    exit

    crypto template <crypto_template>

        authentication eap-profile <eap_prof_name>

        exit

        ikev2-ikesa transform-set list <ikev2_trans_set>

        payload <crypto_payload_name> match childsa [match {ipv4 | ipv6}]
```

```

        ip-address-alloc dynamic

        ipsec transform-setlist <ipsec_trans_set>

        exit

    ikev2-ikesa keepalive-user-activity

end

configure

    context <vpn_ctxt_name>

        hnbgw-service <hnbgw_svc_name>

            security-gateway bind address <segw_ip_address> crypto-template <crypto_template>
        context <segw_ctxt_name>

    end

```

Notes:

- <vpn_ctxt_name> is name of the source context in which HNB-GW service is configured.
- <segw_ctxt_name> is name of the context in which Se-GW service is configured. By default it takes context where HNB-GW service is configured.
- <hnbgw_svc_name> is name of the HNB-GW service which is to be configured for used for Iuh reference between HNB-GW and HNB.

X.509 Certificate-based Peer Authentication

X.509 specifies standard formats for public key certificates, certificate revocation lists, attribute certificates, and a certification path validation algorithm. X.509 certificates are configured on each IPSec node so that it can send the certificate as part of its IKE_AUTH_REQ for the remote node to authenticate it. These certificates can be in PEM (Privacy Enhanced Mail) or DER (Distinguished Encoding Rules) format, and can be fetched from a repository via HTTP or FTP.

CA certificate authentication is used to validate the certificate that the local node receives from a remote node during an IKE_AUTH exchange.

A maximum of sixteen certificates and sixteen CA certificates are supported per system. One certificate is supported per service, and a maximum of four CA certificates can be bound to one crypto template.

The figure below shows the message flow during X.509 certificate-based peer authentication. The table that follows the figure describes each step in the message flow.

Figure 46. X.509 Certificate-based Peer Authentication

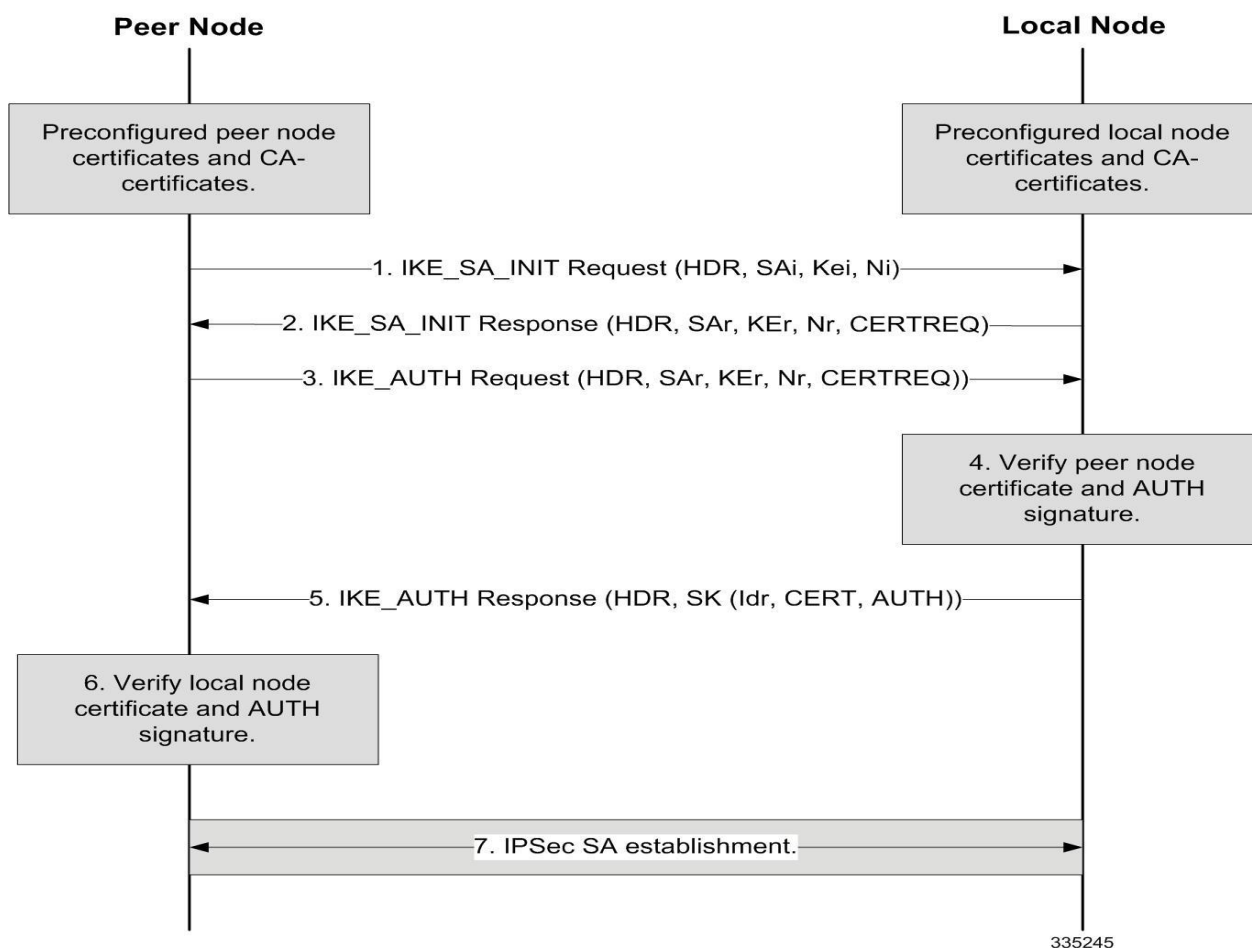


Table 47. X.509 Certificate-based Peer Authentication

Step	Description
1.	The peer node initiates an IKEv2 exchange with the local node, known as the IKE_SA_INIT exchange, by issuing an IKE_SA_INIT Request to negotiate cryptographic algorithms, exchange nonces, and perform a Diffie-Hellman exchange with the local node.
2.	The local node responds with an IKE_SA_INIT Response by choosing a cryptographic suite from the initiator's offered choices, completing the Diffie-Hellman and nonce exchanges with the peer node. In addition, the local node includes the list of CA certificates that it will accept in its CERTREQ payload. For successful peer authentication, the CERTREQ payload must contain at least one CA certificate that is in the trust chain of the peer certificate. At this point in the negotiation, the IKE_SA_INIT exchange is complete and all but the headers of all the messages that follow are encrypted and integrity-protected.

Step	Description
3.	The peer node initiates an IKE_AUTH exchange with the local node by including the IDi payload, setting the CERT payload to the peer certificate, and including the AUTH payload containing the signature of the previous IKE_SA_INIT Request message (in step 1) generated using the private key of the peer certificate. The authentication algorithm used to generate the AUTH payload is also included in the AUTH payload. The peer node also includes the CERTREQ payload containing the list of SHA-1 hash algorithms for local node authentication. For successful server authentication, the CERTREQ payload must contain at least one CA certificate that is in the trust chain of the peer certificate.
4.	Using the CA certificate corresponding to the peer certificate, the local node first verifies that the peer certificate in the CERT payload has not been modified and the identity included in the IDi corresponds to the identity in the peer certificate. If the verification is successful, using the public key of the peer certificate, the local node generates the expected AUTH payload and compares it with the received AUTH payload. If they match, the authentication of the peer node is successful. Otherwise, the local node sends an IKEv2 Notification message indicating authentication failure.
5.	The local node responds with the IKE_AUTH Response, including the IDr payload, setting the CERT payload to the local node certificate, and including the AUTH payload containing the signature of the IKE_SA_INIT Response message (in step 2) generated using the private key of the local node certificate. The authentication algorithm used to generate the AUTH payload is also included in the AUTH payload.
6.	Using the CA certificate corresponding to the local node certificate, the peer node first verifies that the local node certificate in the CERT payload has not been modified. If the verification is successful, using the public key of the local node certificate, the peer generates the expected AUTH payload and compares it with the received AUTH payload. If they match, the local node authentication is successful. This completes the IKE_AUTH exchange.
7.	An IPSec SA gets established between the peer node and the local node. If more IPSec SAs are needed, either the peer or local node can initiate the creation of additional Child SAs using a CREATE_CHILD_SA exchange.

Certificate Revocation Lists

Certificate revocation lists track certificates that have been revoked by the CA (Certificate Authority) and are no longer valid. Per RFC 3280, during certificate validation, IPSec for Femto-UMTS checks the certificate revocation list to verify that the certificate the local node receives from the remote node has not expired and hence is still valid.

During configuration via the system CLI, one certificate revocation list is bound to each crypto template and can be fetched from its repository via HTTP or FTP.

Child SA Rekey Support

Rekeying of an IKEv2 Child Security Association (SA) occurs for an already established Child SA whose lifetime (either time-based or data-based) is about to exceed a maximum limit. The IPSec subsystem initiates rekeying to replace the existing Child SA. During rekeying, two Child SAs exist momentarily (500ms or less) to ensure that transient packets from the original Child SA are processed by the IPSec node and not dropped.

Child SA rekeying is disabled by default, and rekey requests are ignored. This feature gets enabled in the Crypto Configuration Payload Mode of the system's CLI.

IKEv2 Keep-Alive Messages (Dead Peer Detection)

IPSec for Femto-UMTS supports IKEv2 keep-alive messages, also known as Dead Peer Detection (DPD), originating from both ends of an IPSec tunnel. Per RFC 3706, DPD is used to simplify the messaging required to verify communication between peers and tunnel availability. You configure DPD on each IPSec node. You can also disable

DPD, and the node will not initiate DPD exchanges with other nodes. However, the node always responds to DPD availability checks initiated by another node regardless of its DPD configuration.

IPSec Tunnel Termination

IPSec tunnel termination occurs during the following scenarios:

- **Idle Tunnel Termination:** When a session manager for a service detects that all subscriber sessions using a given IPSec tunnel have terminated, the IPSec tunnel also gets terminated after a timeout period.
- **Service Termination:** When a service running on a network node is brought down for any reason, all corresponding IPSec tunnels get terminated. This may be caused by the interface for a service going down, a service being stopped manually, or a task handling an IPSec tunnel restarting.
- **Unreachable Peer:** If a network node detects an unreachable peer via Dead Peer Detection (DPD), the IPSec tunnel between the nodes gets terminated. DPD can be enabled per HNB-GW service via the system CLI during crypto template configuration.
- **Network Handover Handling:** Any IPSec tunnel that becomes unusable due to a network handover gets terminated, while the network node to which the session is handed initiates a new IPSec tunnel for the session.

x.509 Certificate Configuration

Use the following example to configure the x.509 certificates on the system to provide security certification between FAP and SeGW in Femto-UMTS network.

configure

```
certificate name <x.509_cert_name> pem { data <pem_data_string> | url <pem_data_url>}
private-key pem { [encrypted] data <PKI_pem_data_string> | url <PKI_pem_data_url>}

ca-certificate name <ca_root_cert_name> pem { data <pem_data_string> | url
<pem_data_url>}

exit

crypto template <segw_crypto_template> ikev2-dynamic

authentication local certificate

authentication remote certificate

keepalive interval <dur> timeout <dur_timeout>

certificate <x.509_cert_name>

ca-certificate list ca-cert-name <ca_root_cert_name>

payload <crypto_payload_name> match childsa [match {ipv4 | ipv6}]

ip-address-alloc dynamic

ipsec transform-setlist <ipsec_trans_set>

end
```

```
configure
context <vpn_ctxt_name>
  subscriber default
    ip context-name <vpn_ctxt_name>
    ip address pool name <ip_pool_name>
  end
```

Notes:

- <vpn_ctxt_name> is name of the source context in which HNB-GW service is configured.
- <x.509_cert_name> is name of the x.509 certificate where PEM data <pem_data_string> and PKI <PKI_pem_data_string> is configured.
- <ca_root_cert_name> is name of the CA root certificate where PEM data <pem_data_string> is configured for CPE.

Appendix L

Intelligent Traffic Control

Before using the procedures in this chapter, it is recommended that you select the configuration example that best meets your service model, and configure the required elements as per that model.

This chapter covers the following topics:

- [Overview](#)
- [How it Works](#)
- [Configuring Flow-based Traffic Policing](#)

Overview

Intelligent Traffic Control (ITC) enables you to configure a set of customizable policy definitions that enforce and manage service level agreements for a subscriber profile, thus enabling you to provide differentiated levels of services for native and roaming subscribers.

In 3GPP2 service ITC uses a local policy look-up table and permits either static EV-DO Rev 0 or dynamic EV-DO Rev A policy configuration.



Important: ITC includes the class-map, policy-map and policy-group commands. Currently ITC does not include an external policy server interface.

ITC provides per-subscriber/per-flow traffic policing to control bandwidth and session quotas. Flow-based traffic policing enables the configuring and enforcing bandwidth limitations on individual subscribers, which can be enforced on a per-flow basis on the downlink and the uplink directions.

Flow-based traffic policies are used to support various policy functions like Quality of Service (QoS), and bandwidth, and admission control. It provides the management facility to allocate network resources based on defined traffic-flow, QoS, and security policies.

ITC and EV-DO Rev A in 3GPP2 Networks



Important: The Ev-Do Rev is a licensed Cisco feature. A separate feature license may be required. Contact your Cisco account representative for detailed information on specific licensing requirements. For information on installing and verifying licenses, refer to the *Managing License Keys* section of the *Software Management Operations* chapter in the *System Administration Guide*.

You can configure your system to support both EV-DO Rev A and ITC. ITC uses flow-based traffic policing to configure and enforce bandwidth limitations per subscriber. Enabling EV-DO Rev A with ITC allows you to control the actual level of bandwidth that is allocated to individual subscriber sessions and the application flows within the sessions.

For more information on EV-DO Rev A, refer to the *Policy-Based Management and EV-DO Rev A* chapter. For setting the DSCP parameters to control ITC functionality, refer to the *Traffic Policy-Map Configuration Mode Commands* chapter in the *Command Line Reference*.

Bandwidth Control and Limiting

Bandwidth control in ITC controls the bandwidth limit, flow action, and charging action for a subscriber, application, and source/destination IP addresses. This is important to help limit bandwidth intensive applications on a network. You can configure ITC to trigger an action to drop, lower-ip-precedence, or allow the flow when the subscriber exceeds the bandwidth usage they have been allotted by their policy.

Licensing

The Intelligent Traffic Control is a licensed Cisco feature. A separate feature license may be required. Contact your Cisco account representative for detailed information on specific licensing requirements. For information on installing and verifying licenses, refer to the *Managing License Keys* section of the *Software Management Operations* chapter in the *System Administration Guide*.

How it Works

ITC enables you to configure traffic policing on a per-subscriber/per-flow basis with the potential to manipulate Differentiated Services Code Points (DSCPs), queue redirection (for example, move traffic to a Best Effort (BE) classification), or drop profile traffic.

In flow-based traffic policies, policy modules interact with the system through a set of well defined entry points, provide access to a stream of system events, and permit the defined policies to implement functions such as access control decisions, QoS enforcement decisions, etc.

Traffic policing can be generally defined as

policy: condition >> action

- **condition:** Specifies the flow-parameters like source-address, destination-address, source-port, destination-port, protocol, etc. for ingress and/or egress packet.
- **action:** Specifies a set of treatments for flow/packet when condition matches. Broadly these actions are based on:
 - **Flow Classification:** Each flow is classified separately on the basis of source-address, destination-address, source-port, destination-port, protocol, etc. for ingress and/or egress packet. After classification access-control allowed or denied by the system.
 - **QoS Processing for individual flow and DSCP marking:** Flow-based traffic policing is implemented by each flow separately for the traffic-policing algorithm. Each flow has its own bucket (burst-size) along with committed data rate and peak data rate. A Token Bucket Algorithm (a modified trTCM) [RFC2698] is used to implement this flow-based QoS traffic policing feature.

Refer to the *Traffic Policing and Shaping* chapter for more information on Token Bucket Algorithm.

Configuring Flow-based Traffic Policing


Traffic Policing is configured on a per-subscriber basis for either locally configured subscribers on the system or subscriber profiles configured on a remote RADIUS server.

Flow-based traffic policy is configured on the system with the following building blocks:

- **Class Maps:** The basic building block of a flow-based traffic policing. It is used to control over the packet classification.
- **Policy Maps:** A more advanced building block for a flow-based traffic policing. It manages admission control based on the Class Maps and the corresponding flow treatment based on QoS traffic-police or QoS DSCP marking.
- **Policy Group:** This is a set of one or more Policy Maps applied to a subscriber. it also resolves the conflict if a flow matches to multiple policies.

This section provides instructions for configuring traffic policies and assigning to local subscriber profiles on the system.

For information on how to configure subscriber profiles on a remote RADIUS server, refer to the *StarentVSA* and *StarentVSA1* dictionary descriptions in the *AAA and GTP Interface Administration and Reference*.

 **Important:** This section provides the minimum instruction set for configuring flow-based traffic policing on an AGW service. Commands that configure additional properties are provided in the *Command Line Interface Reference*.

These instructions assume that you have already configured the system-level configuration as described in product administration guide.

To configure the flow-based traffic policing on an AGW service:

1. Configure the traffic class maps on the system to support flow-based traffic policing by applying the example configuration in the [Configuring Class Maps](#) section.
2. Configure the policy maps with traffic class maps on the system to support flow-based traffic policing by applying the example configuration in the [Configuring Policy Maps](#) section.
3. Configure the policy group with policy maps on the system to support flow-based traffic policing by applying the example configuration in the [Configuring Policy Groups](#) section.
4. Associate the subscriber profile with policy group to enable flow-based traffic policing for subscriber by applying the example configuration in the [Configuring a Subscriber for Flow-based Traffic Policing](#) section.
5. Verify your flow-based traffic policing configuration by following the steps in the [Verifying Flow-based Traffic Policing Configuration](#) section.
6. Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command `save configuration`. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Configuring Class Maps

This section describes how to configure Class Maps on the system to support Flow-based Traffic Policing.



Important: In this mode classification match rules added sequentially with **match** command to form a Class-Map. To change and/or delete or re-add a particular rule user must delete specific Class-Map and re-define it.

configure

```
context <vpn_context_name> [ -noconfirm ]

  class-map name <class_name> [ match-all | match-any ]

    match src-ip-address <src_ip_address> [ <subnet_mask> ]

    match dst-ip-address <dst_ip_address> [ <subnet_mask> ]

    match source-port-range <initial_port_number> [ to <last_port_number> ]

    match dst-port-range <initial_port_number> [ to <last_port_number> ]

    match protocol [ tcp | udp | gre | ip-in-ip ]

    match ip-tos <service_value>

    match ipsec-spi <index_value>

    match packet-size [ gt | lt ] <size>

  end
```

Notes:

- <vpn_context_name> is the name of the destination context in which you want to configure the flow-based traffic policing.
- <class_name> is the name of the traffic class to map with the flow for the flow-based traffic policing. A maximum of 32 class-maps can be configured in one context.
- For description and variable values of these commands and keywords, refer to the *Class-Map Configuration Mode Commands* chapter of the *Command Line Interface Reference*.

Configuring Policy Maps

This section provides information and instructions for configuring the policy maps on the system to support flow-based traffic policing.

configure

```
context <vpn_context_name>

  policy-map name <policy_name>

    class <class_name>

      type { static | dynamic }

      access-control { allow | discard }
```

```

    qos traffic-police committed <bps> peak <bps> burst-size <byte> exceed-action {
drop | lower-ip-precedence | allow } violate-action { drop | lower-ip-precedence | allow
}

    qos encaps-header dscp-marking [ copy-from-user-datagram | <dscp_code> ]

end

```

Notes:

- *<vpn_context_name>* is the name of the destination context in which is configured during Class-Map configuration for flow-based traffic policing.
- *<policy_name>* is the name of the traffic policy map you want to configure for the flow-based traffic policing. A maximum of 32 policy maps can be configured in one context.
- *<class_name>* is the name of the traffic class to map that you configured in *Configuring Class Maps* section for the flow-based traffic policing.
- For description and variable values of these commands and keywords, refer to the *Traffic Policy-Map Configuration Mode Commands* chapter of the *Command Line Interface Reference*.

Configuring Policy Groups

This section provides information and instructions for configuring the policy group in a context to support flow-based traffic policing.

configure

```

context <vpn_context_name>

    policy-group name <policy_group>

        policy <policy_map_name> precedence <value>

    end

```

Notes:

- *<vpn_context_name>* is the name of the destination context which is configured during Class-Map configuration for flow-based traffic policing.
- *<policy_group>* is name of the traffic policy group of policy maps you want to configure for the flow-based traffic policing. A maximum of 32 policy groups can be configured in one context.
- *<policy_map_name>* is name of the traffic policy you configured in *Configuring Policy Maps* section for the flow-based traffic policing. A maximum of 16 Policy Maps can be assigned in a Policy Group.
- For description and variable values of these commands and keywords, refer to the *Traffic Policy-Map Configuration Mode Commands* chapter of the *Command Line Interface Reference*.

Configuring a Subscriber for Flow-based Traffic Policing

This section provides information and instructions for configuring the subscriber for Flow-based Traffic Policing.

configure

```
context <vpn_context_name>

  subscriber name <user_name>

    policy-group <policy_group> direction [ in | out ]

  end
```

Notes:

- <vpn_context_name> is the name of the destination context configured during Class-Map configuration for flow-based traffic policing.
- <user_name> is the name of the subscriber profile you want to configure for the flow-based traffic policing.
- <policy_group> is name of the traffic policy group you configured in *Configuring Policy Groups* section for the flow-based traffic policing. A maximum of 16 Policy groups can be assigned to a subscriber profile.
- For description and variable values of these commands and keywords, refer to the *Traffic Policy-Group Configuration Mode Commands* chapter of the *Command Line Interface Reference*.

Verifying Flow-based Traffic Policing Configuration

Step 1 Verify that your flow-based traffic policing is configured properly by entering the following command in Exec Mode:

```
show subscribers access-flows full
```

The output of this command displays flow-based information for a subscriber session.

Appendix M

L2TP Access Concentrator

This chapter describes the Layer 2 Tunneling Protocol (L2TP) Access Concentrator (LAC) functionality support on Cisco® ASR 5x00 chassis and explains how it is configured.

The product Administration Guides provide examples and procedures for configuration of basic services on the system. It is recommended that you select the configuration example that best meets your service model, and configure the required elements for that model, as described in the respective product Administration Guide, before using the procedures in this chapter.



Important: The L2TP Access Concentrator is a licensed Cisco feature. A separate feature license may be required. Contact your Cisco account representative for detailed information on specific licensing requirements. For information on installing and verifying licenses, refer to the *Managing License Keys* section of the *Software Management Operations* chapter in the *System Administration Guide*.

When enabled through the session license and feature use key, the system supports L2TP for encapsulation of data packets between it and one or more L2TP Network Server (LNS) nodes. In the system, this optional packet encapsulation, or tunneling, is performed by configuring L2TP Access Concentrator (LAC) services within contexts.



Important: The LAC service uses UDP ports 13660 through 13668 as the source port for sending packets to the LNS.

Applicable Products and Relevant Sections

The LAC feature is supported for various products. The following table indicates the products on which the feature is supported and the relevant sections within the chapter that pertain to that product.

Applicable Product(s)	Refer to Sections
PDSN/FA/HA	<ul style="list-style-type: none"> • <i>Supported LAC Service Configurations for PDSN Simple IP</i> • <i>Supported LAC Service Configuration for Mobile IP</i> • <i>Configuring Subscriber Profiles for L2TP Support</i> <ul style="list-style-type: none"> • <i>RADIUS and Subscriber Profile Attributes Used</i> • <i>Configuring Local Subscriber Profiles for L2TP Support</i> • <i>Tunneling All Subscribers in a Specific Context Without Using RADIUS Attributes</i> • <i>Configuring LAC Services</i> • <i>Modifying PDSN Services for L2TP Support</i>
GGSN/SGSN/FA/P-GW	<ul style="list-style-type: none"> • <i>Supported LAC Service Configurations for the GGSN</i> • <i>Supported LAC Service Configuration for Mobile IP</i> • <i>Configuring Subscriber Profiles for L2TP Support</i> <ul style="list-style-type: none"> • <i>RADIUS and Subscriber Profile Attributes Used</i> • <i>Configuring Local Subscriber Profiles for L2TP Support</i> • <i>Configuring LAC Services</i> • <i>Modifying APN Templates to Support L2TP</i>
ASN GW	<ul style="list-style-type: none"> • <i>Supported LAC Service Configuration for Mobile IP</i> • <i>Configuring Subscriber Profiles for L2TP Support</i> <ul style="list-style-type: none"> • <i>RADIUS and Subscriber Profile Attributes Used</i> • <i>Configuring Local Subscriber Profiles for L2TP Support</i> • <i>Tunneling All Subscribers in a Specific Context Without Using RADIUS Attributes</i> • <i>Configuring LAC Services</i>

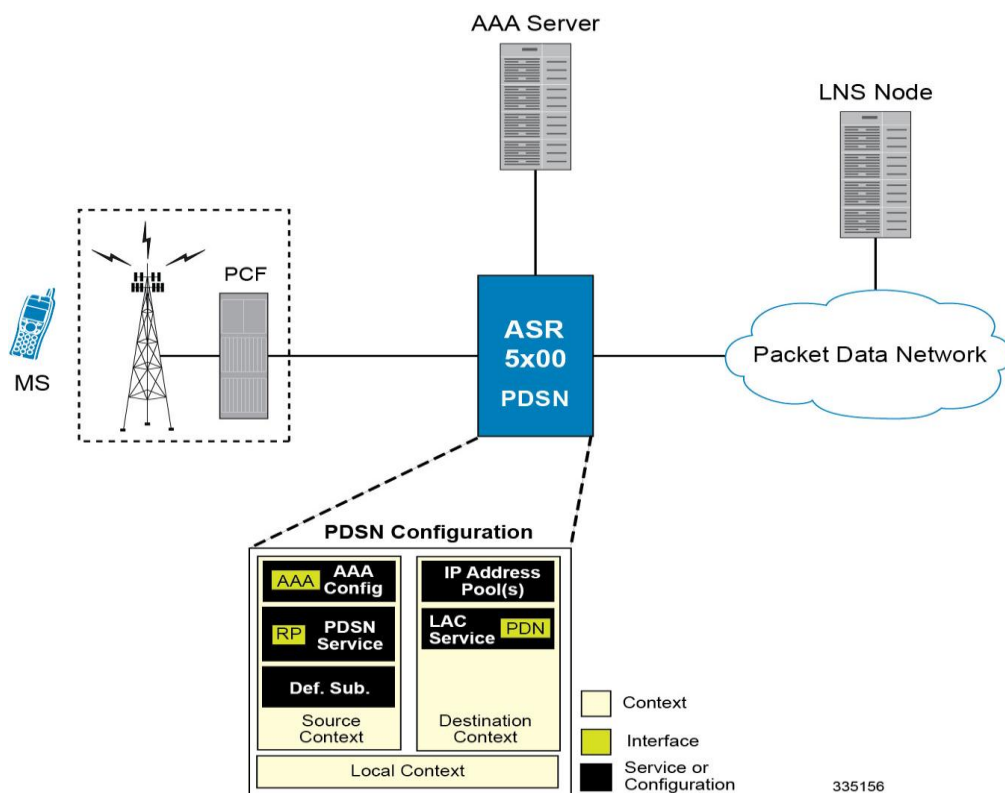
Supported LAC Service Configurations for PDSN Simple IP

LAC services can be applied to incoming PPP sessions using one of the following methods:

- **Attribute-based tunneling:** This method is used to encapsulate PPP packets for only specific users, identified during authentication. In this method, the LAC service parameters and allowed LNS nodes that may be communicated with are controlled by the user profile for the particular subscriber. The user profile can be configured locally on the system or remotely on a RADIUS server.
- **PDSN Service-based compulsory tunneling:** This method of tunneling is used to encapsulate all incoming PPP traffic from the R-P interface coming into a PDSN service, and tunnel it to an LNS peer for authentication. It should be noted that this method does not consider subscriber configurations, since all authentication is performed by the peer LNS.

Each LAC service is bound to a single system interface configured within the same system context. It is recommended that this context be a destination context as displayed in the following figure.

Figure 47. LAC Service Configuration for SIP



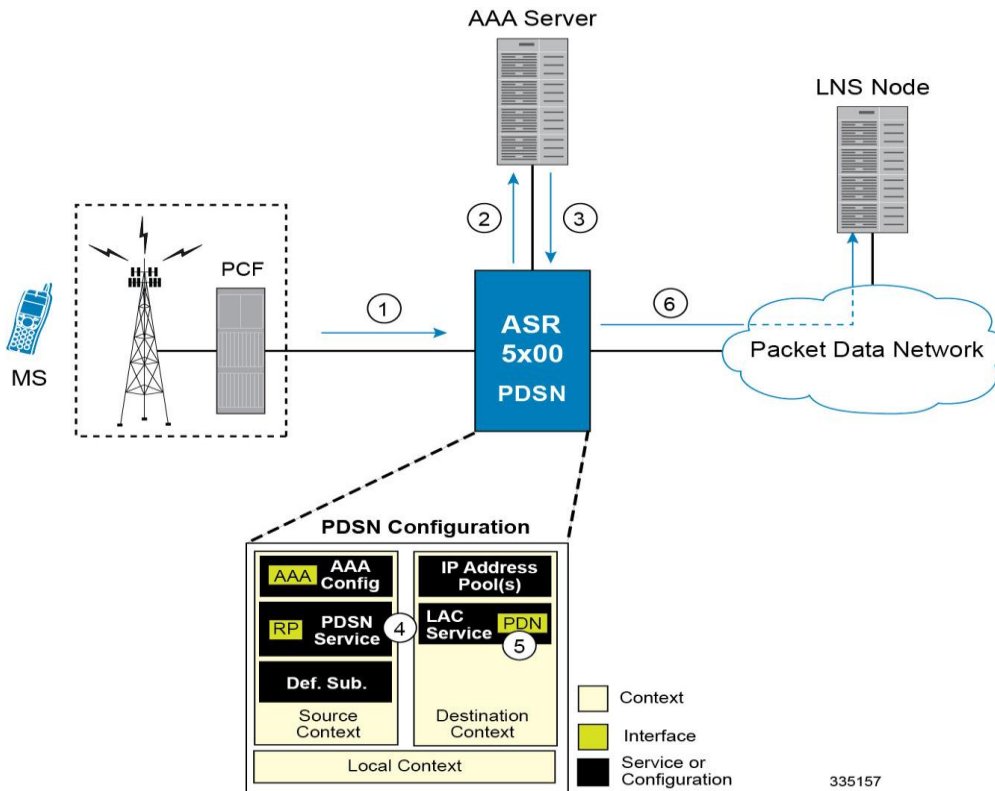
Attribute-based Tunneling

This section describes the working of attribute-based tunneling and its configuration.

How The Attribute-based L2TP Configuration Works

The following figure and the text that follows describe how Attribute-based tunneling is performed using the system.

Figure 48. Attribute-based L2TP Session Processing for SIP



1. A subscriber session from the PCF is received by the PDSN service over the R-P interface.
2. The PDSN service attempts to authenticate the subscriber. The subscriber could be configured either locally or remotely on a RADIUS server. Figure above shows subscriber authentication using a RADIUS AAA server.
3. The RADIUS server returns an Access-Accept message, which includes attributes indicating that session data is to be tunneled using L2TP, and the name and location of the LAC service to use. An attribute could also be provided indicating the LNS peer to connect to.
4. The PDSN service receives the information and then forwards the packets to the LAC service, configured within the Destination context.
5. The LAC service, upon receiving the packets, encapsulates the information and forwards it to the appropriate PDN interface for delivery to the LNS.
6. The encapsulated packets are sent to the peer LNS through the packet data network where they will be un-encapsulated.

Configuring Attribute-based L2TP Support for PDSN Simple IP

This section provides a list of the steps required to configure attribute-based L2TP support for use with PDSN Simple IP applications. Each step listed refers to a different section containing the specific instructions for completing the required procedure.



Important: These instructions assume that the system was previously configured to support subscriber data sessions as a PDSN.

- Step 1** Configure the subscriber profiles according to the information and instructions located in the *Configuring Subscriber Profiles for L2TP Support* section of this chapter.
- Step 2** Configure one or more LAC services according to the information and instructions located in the *Configuring LAC Services* section of this chapter.
- Step 3** Configure the PDSN service(s) with the tunnel context location according to the instructions located in the *Modifying PDSN Services for L2TP Support* section of this chapter.
- Step 4** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

PDSN Service-based Compulsory Tunneling

This section describes the working of service-based compulsory tunneling and its configuration.

How PDSN Service-based Compulsory Tunneling Works

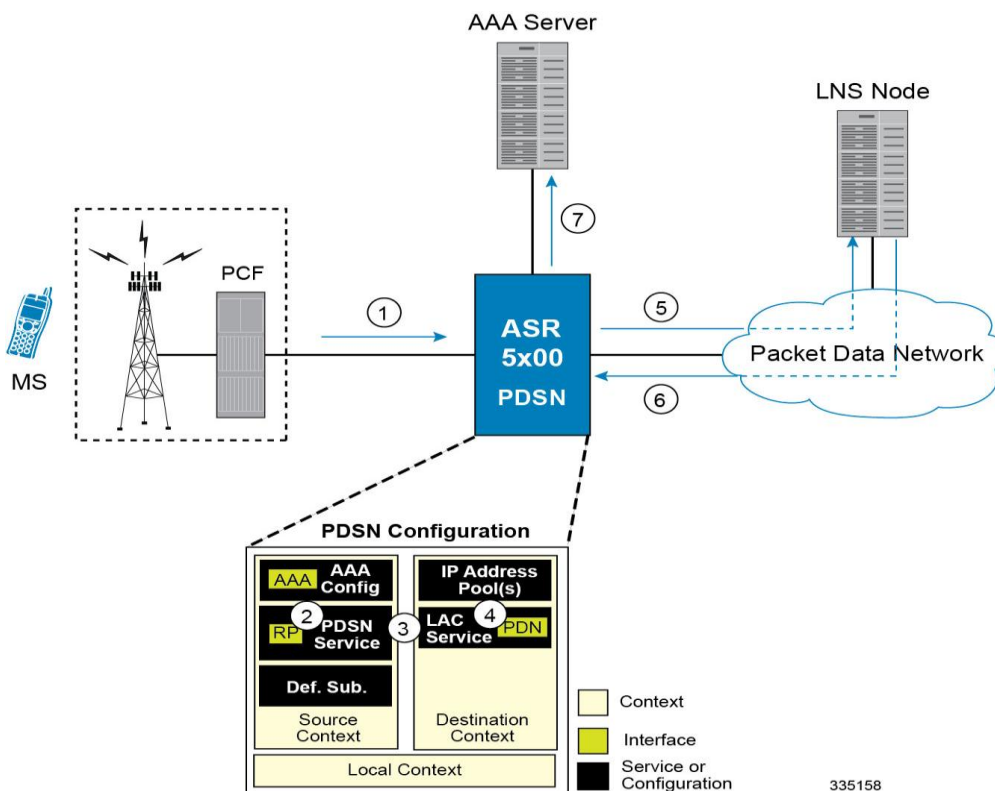
PDSN Service-based compulsory tunneling enables wireless operators to send all PPP traffic to remote LNS peers over an L2TP tunnel for authentication. This means that no PPP authentication is performed by the system.

Accounting start and interim accounting records are still sent to the local RADIUS server configured in the system's AAA Service configuration. When the L2TP session setup is complete, the system starts its call counters and signals the RADIUS server to begin accounting. The subscriber name for accounting records is based on the NAI-constructed name created for each session.

PDSN service-based compulsory tunneling requires the modification of one or more PDSN services and the configuration of one or more LAC services.

The following figure and the text that follows describe how PDSN service-based compulsory tunneling is performed using the system.

Figure 49. PDSN Service-based Compulsory Tunneling Session Processing



1. A subscriber session from the PCF is received by the PDSN service over the R-P interface.
2. The PDSN service detects its **tunnel-type** parameter is configured to L2TP and its **tunnel-context** parameter is configured to the Destination context.
3. The PDSN forwards all packets for the session to a LAC service configured in the Destination context. If multiple LAC services are configured, session traffic will be routed to each using a round-robin algorithm.
4. The LAC service initiates an L2TP tunnel to one of the LNS peers listed as part of its configuration.
5. Session packets are passed to the LNS over a packet data network for authentication.
6. The LNS authenticates the session and returns an Access-Accept to the PDSN.
7. The PDSN service initiates accounting for the session using a constructed NAI.

Session data traffic is passed over the L2TP tunnel established in step 4.

Configuring L2TP Compulsory Tunneling Support for PDSN Simple IP

This section provides a list of the steps required to configure L2TP compulsory tunneling support for use with PDSN Simple IP applications. Each step listed refers to a different section containing the specific instructions for completing the required procedure.



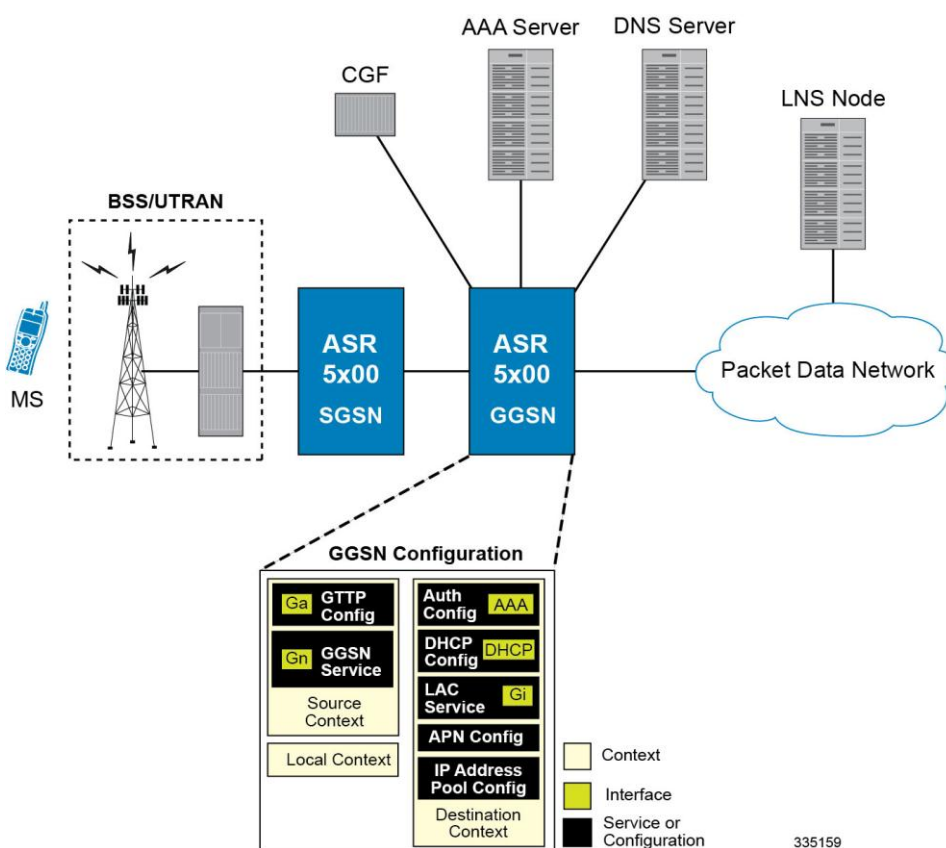
Important: These instructions assume that the system was previously configured to support subscriber data sessions as a PDSN.

- Step 1** Configure one or more LAC services according to the information and instructions located in the *Configuring LAC Services* section of this chapter.
- Step 2** Configure the PDSN service(s) according to the instructions located in the *Modifying PDSN Services for L2TP Support* section of this chapter.
- Step 3** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Supported LAC Service Configurations for the GGSN and P-GW

As mentioned previously, L2TP is supported through the configuration of LAC services on the system. Each LAC service is bound to a single system interface configured within the same system destination context as displayed in following figure.

Figure 50. GGSN LAC Service Configuration



LAC services are applied to incoming subscriber PDP contexts based on the configuration of attributes either in the GGSN's Access Point Name (APN) templates or in the subscriber's profile. Subscriber profiles can be configured locally on the system or remotely on a RADIUS server.

LAC service also supports domain-based L2TP tunneling with LNS. This method is used to create multiple tunnels between LAC and LNS on the basis of values received in "Tunnel-Client-Auth-ID" or "Tunnel-Server-Auth-ID" attribute received from AAA Server in Access-Accept as a key for tunnel selection and creation. When the LAC needs to establish a new L2TP session, it first checks if there is any existing L2TP tunnel with the peer LNS based on the value of key "Tunnel-Client-Auth-ID" or "Tunnel-Server-Auth-ID" attribute. If no such tunnel exists for the key, it will create a new Tunnel with the LNS.

If LAC service needs to establish a new tunnel for new L2TP session with LNS and the tunnel create request fails because maximum tunnel creation limit is reached, LAC will try other LNS addresses received from AAA server in Access-Accept message. If all available peer-LNS are exhausted, LAC service will reject the call

L2TP tunnel parameters are configured within the APN template and are applied to all subscribers accessing the APN. However, L2TP operation will differ depending on the subscriber's PDP context type as described below:

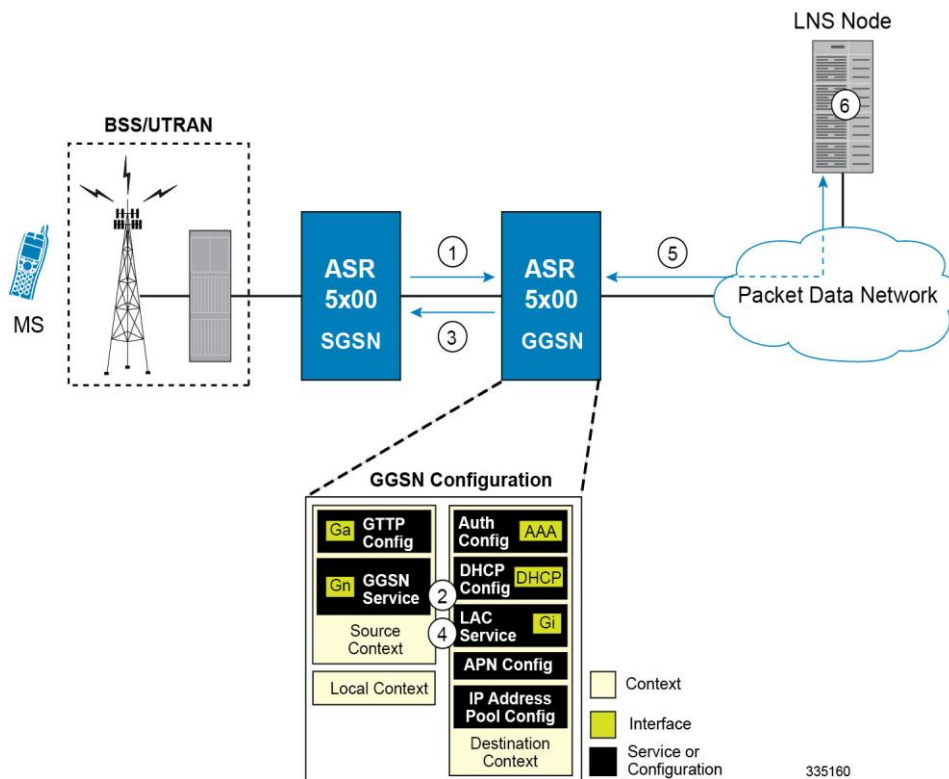
- **Transparent IP:** The APN template's L2TP parameter settings will be applied to the session.
- **Non-transparent IP:** Since authentication is required, L2TP parameter attributes in the subscriber profile (if configured) will take precedence over the settings in the APN template.
- **PPP:** The APN template's L2TP parameter settings will be applied and all of the subscriber's PPP packets will be forwarded to the specified LNS.

More detailed information is located in the sections that follow.

Transparent IP PDP Context Processing with L2TP Support

The following figure and the text that follows describe how transparent IP PDP contexts are processed when L2TP tunneling is enabled.

Figure 51. Transparent IP PDP Context Call Processing with L2TP Tunneling



1. A Create PDP Context Request message for a subscriber session is sent from the SGSN to the GGSN service over the Gn interface. The message contains information such as the PDP Type, APN, and charging characteristics.
2. The GGSN determines whether or not it is configured with an APN identical to the one specified in the message. If so, it determines how to process the session based on the configuration of the APN.

The APN configuration indicates such things as the IP address of the LNS, the system destination context in which a LAC service is configured, and the outbound username and password that will be used by the LNS to authenticate incoming

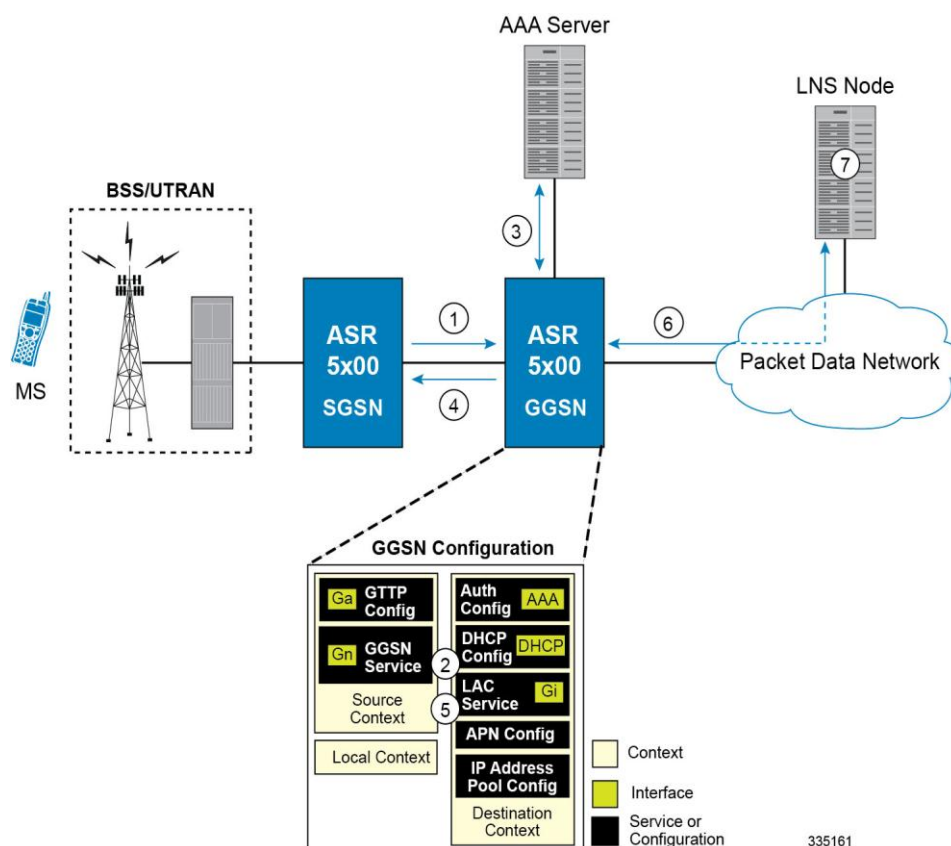
sessions. If no outbound information is configured, the subscriber's International Mobile Subscriber Identity (IMSI) is used as the username at the peer LNS.

1. The GGSN returns an affirmative Create PDP Context Response to the SGSN over the Gn interface.
2. The GGSN passes data received from the MS to a LAC service.
3. The LAC service encapsulates the IP packets and forwards it to the appropriate Gi interface for delivery to the LNS.
4. The LNS un-encapsulates the packets and processes them as needed. The processing includes IP address allocation.

Non-transparent IP PDP Context Processing with L2TP Support

The following figure and the text that follows describe how non-transparent IP PDP contexts are processed when L2TP tunneling is enabled.

Figure 52. Non-transparent IP PDP Context Call Processing with L2TP Tunneling



1. A Create PDP Context Request message for a subscriber session is sent from the SGSN to the GGSN service over the Gn interface. The message contains information such as the PDP Type, APN, and charging characteristics.
2. The GGSN determines whether or not it is configured with an APN identical to the one specified in the message. If so, it determines how to process the session based on the configuration of the APN.

The APN configuration indicates such things as the IP address of the LNS, the system destination context in which a LAC service is configured, and the outbound username and password that will be used by the LNS to authenticate incoming sessions. If no outbound information is configured, the subscriber's username is sent to the peer LNS.

3. The GGSN service authenticates the subscriber. The subscriber could be configured either locally or remotely on a RADIUS server. Figure above shows subscriber authentication using a RADIUS AAA server. As part of the authentication, the RADIUS server returns an Access-Accept message.

The message may include attributes indicating that session data is to be tunneled using L2TP, and the name and location of the LAC service to use. An attribute could also be provided indicating the LNS peer to connect to.

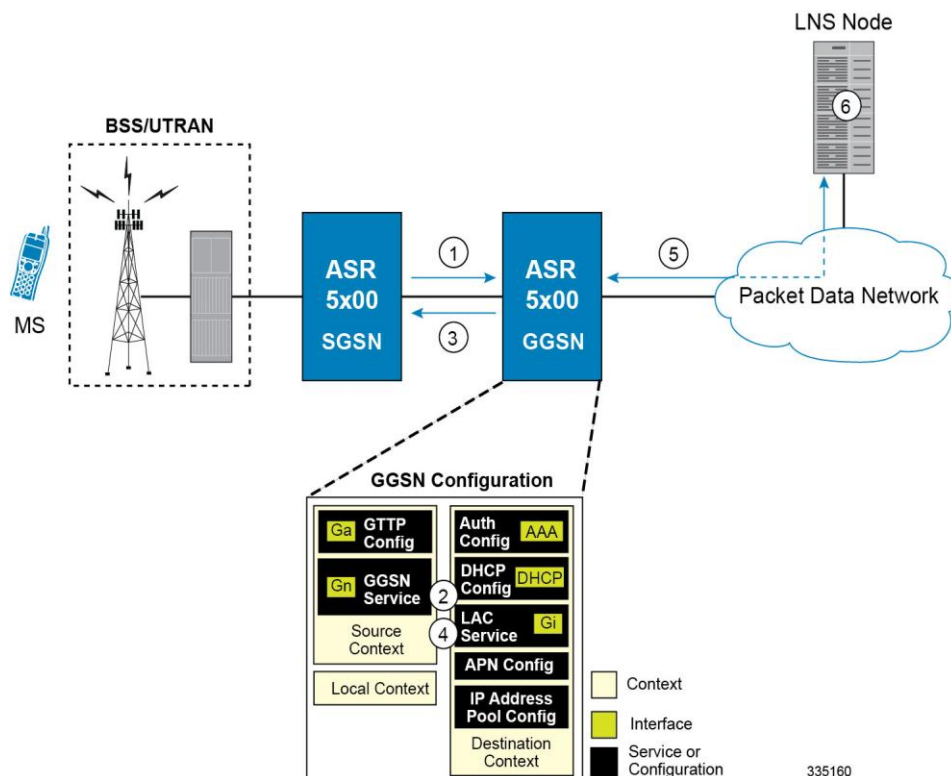
If these attributes are supplied, they take precedence over those specified in the APN template.

4. The GGSN returns an affirmative Create PDP Context Response to the SGSN over the Gn interface.
5. The GGSN passes data received from the MS to a LAC service.
6. The LAC service encapsulates the IP packets and forwards it to the appropriate Gi interface for delivery to the LNS.
7. The LNS un-encapsulates the packets and processes them as needed. The processing includes authentication and IP address allocation.

PPP PDP Context Processing with L2TP Support

The following figure and the text that follows describe how non-transparent IP PDP contexts are processed when L2TP tunneling is enabled.

Figure 53. PPP PDP Context Call Processing with L2TP Tunneling



1. A Create PDP Context Request message for a subscriber session is sent from the SGSN to the GGSN service over the Gn interface. The message contains information such as the PDP Type, APN, and charging characteristics.
2. The GGSN determines whether or not it is configured with an APN identical to the one specified in the message. If so, it determines how to process the session based on the configuration of the APN. The APN configuration indicates such things as the IP address of the LNS, the system destination context in which a LAC service is configured.

Note that L2TP support could also be configured in the subscriber's profile. If the APN is not configured for L2TP tunneling, the system will attempt to authenticate the subscriber. The tunneling parameters in the subscriber's profile would then be used to determine the peer LNS.

3. The GGSN returns an affirmative Create PDP Context Response to the SGSN over the Gn interface.
4. The GGSN passes the PPP packets received from the MS to a LAC service.
5. The LAC service encapsulates the PPP packets and forwards it to the appropriate Gi interface for delivery to the LNS.
6. The LNS un-encapsulates the packets and processes them as needed. The processing includes PPP termination, authentication (using the username/password provided by the subscriber), and IP address allocation.

Configuring the GGSN or P-GW to Support L2TP

This section provides a list of the steps required to configure the GGSN or P-GW to support L2TP. Each step listed refers to a different section containing the specific instructions for completing the required procedure.



Important: These instructions assume that the system was previously configured to support subscriber data sessions as a GGSN or P-GW.

1. Configure the APN template to support L2TP tunneling according to the information and instructions located in the *Modifying APN Templates to Support L2TP* section of this chapter.



Important: L2TP tunneling can be configured within individual subscriber profiles as opposed/or in addition to configuring support with an APN template. Subscriber profile configuration is described in the *Configuring Subscriber Profiles for L2TP Support* section of this chapter.

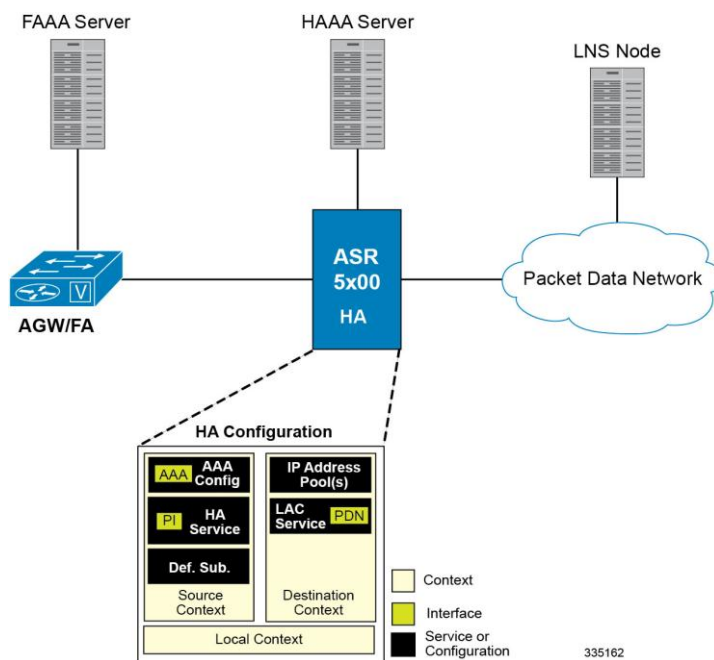
2. Configure one or more LAC services according to the information and instructions located in the *Configuring LAC Services* section of this chapter.
3. Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Supported LAC Service Configuration for Mobile IP

LAC services can be applied to incoming MIP sessions using attribute-based tunneling. Attribute-based tunneling is used to encapsulate PPP packets for specific users, identified during authentication. In this method, the LAC service parameters and allowed LNS nodes that may be communicated with are controlled by the user profile for the particular subscriber. The user profile can be configured locally on the system or remotely on a RADIUS server.

Each LAC service is bound to a single system interface within the same system context. It is recommended that this context be a destination context as displayed in figure below.

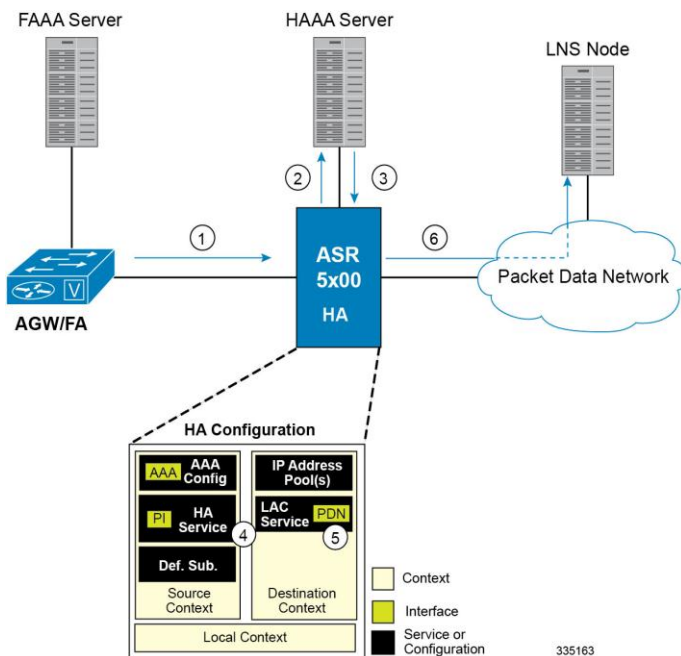
Figure 54. LAC Service Configuration for MIP



How The Attribute-based L2TP Configuration for MIP Works

The following figure and the text that follows describe how Attribute-based tunneling for MIP is performed using the system.

Figure 55. Attribute-based L2TP Session Processing for MIP



1. A subscriber session from the FA is received by the HA service over the Pi interface.
2. The HA service attempts to authenticate the subscriber. The subscriber could be configured either locally or remotely on a RADIUS server. Figure above shows subscriber authentication using a RADIUS AAA server.
3. The RADIUS server returns an Access-Accept message, which includes attributes indicating that session data is to be tunneled using L2TP, and the name and location of the LAC service to use. An attribute could also be provided indicating the LNS peer to connect to.
4. The HA service receives the information and then forwards the packets to the LAC service, configured within the Destination context.
5. The LAC service, upon receiving the packets, encapsulates the information and forwards it to the appropriate PDN interface for delivery to the LNS.
6. The encapsulated packets are sent to the peer LNS through the packet data network where they will be un-encapsulated.

Configuring Attribute-based L2TP Support for HA Mobile IP

This section provides a list of the steps required to configure attribute-based L2TP support for use with HA Mobile IP applications. Each step listed refers to a different section containing the specific instructions for completing the required procedure.



Important: These instructions assume that the system was previously configured to support subscriber data sessions as an HA.

- Step 1** Configure the subscriber profiles according to the information and instructions located in the *Configuring Subscriber Profiles for L2TP Support* section of this chapter.
- Step 2** Configure one or more LAC services according to the information and instructions located in the *Configuring LAC Services* section of this chapter.

- Step 3** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Configuring Subscriber Profiles for L2TP Support

This section provides information and instructions on the following procedures:

- [RADIUS and Subscriber Profile Attributes Used](#)
- [Configuring Local Subscriber Profiles for L2TP Support](#)
- [Configuring Local Subscriber](#)
- [Verifying the L2TP Configuration](#)





Important: Since the instructions for configuring subscribers differ between RADIUS server applications, this section only provides the individual attributes that can be added to the subscriber profile. Refer to the documentation that shipped with your RADIUS server for instructions on configuring subscribers.

RADIUS and Subscriber Profile Attributes Used

Attribute-based L2TP tunneling is supported through the use of attributes configured in subscriber profiles stored either locally on the system or remotely on a RADIUS server. The following table describes the attributes used in support of LAC services. These attributes are contained in the standard and VSA dictionaries.

Table 48. Subscriber Attributes for L2TP Support

RADIUS Attribute	Local Subscriber Attribute	Description	Variable
Tunnel-Type	tunnel l2tp	Specifies the type of tunnel to be used for the subscriber session	L2TP
Tunnel-Server-Endpoint	tunnel l2tp peer-address	Specifies the IP address of the peer LNS to connect tunnel to.	IPv4 address in dotted-decimal format, enclosed in quotation marks
Tunnel-Password	tunnel l2tp secret	Specifies the shared secret between the LAC and LNS.	Alpha and or numeric string from 1 to 63 characters, enclosed in quotation marks
Tunnel-Private-Group-ID	tunnel l2tp tunnel-context	Specifies the name of the destination context configured on the system in which the LAC service(s) to be used are located.  Important: If the LAC service and egress interface are configured in the same context as the core service or HA service, this attribute is not needed.	Alpha and or numeric string from 1 to 63 characters, enclosed in quotation marks

RADIUS Attribute	Local Subscriber Attribute	Description	Variable
Tunnel-Preference	tunnel l2tp preference	Configures the priority of each peer LNS when multiple LNS nodes are configured.  Important: This attribute is only used when the loadbalance-tunnel-peers parameter or SN-Tunnel-Load-Balancing attribute configured to prioritized.	Integer from 1 to 65535
SN-Tunnel-Load-Balancing	loadbalance-tunnel- peer	A vendor-specific attribute (VSA) used to provides a selection algorithm defining how an LNS node is selected by the RADIUS server when multiple LNS peers are configured within the subscriber profile.	<ul style="list-style-type: none"> • Random - Random LNS selection order, the Tunnel-Preference attribute is not used in determining which LNS to select. • Balanced - LNS selection is sequential balancing the load across all configured LNS nodes, the Tunnel-Preference attribute is not used in determining which LNS to select. • Prioritized - LNS selection is made based on the priority assigned in the Tunnel-Preference attribute.
Client-Endpoint	local-address	Specifies the IP address of a specific LAC service configured on the system that to use to facilitate the subscriber's L2TP session. This attribute is used when multiple LAC services are configured.	IPv4 address in dotted decimal notation. (xxx.xxx.xxx.xxx)

RADIUS Tagging Support

The system supports RADIUS attribute tagging for tunnel attributes. These “tags” organize together multiple attributes into different groups when multiple LNS nodes are defined in the user profile. Tagging is useful to ensure that the system groups all the attributes used for a specific server. If attribute tagging is not supported by your specific RADIUS server, the system implicitly organizes the attributes in the order that they are listed in the access accept packet.

Configuring Local Subscriber Profiles for L2TP Support

This section provides information and instructions for configuring local subscriber profiles on the system to support L2TP.



Important: The configuration of RADIUS-based subscriber profiles is not discussed in this document. Please refer to the documentation supplied with your RADIUS server for further information.



Important: This section provides the minimum instruction set for configuring local subscriber profile for L2TP support on the system. For more information on commands that configure additional parameters and options, refer to the *LAC Service Configuration Mode Commands* chapter in the *Command Line Interface Reference*.

To configure the system to provide L2TP support to subscribers:

- Step 1** Configure the “Local” subscriber with L2TP tunnel parameters and the load balancing parameters with action by applying the example configuration in the *Configuring Local Subscriber* section.
- Step 2** Verify your L2TP configuration by following the steps in the *Verifying the L2TP Configuration* section.
- Step 3** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Configuring Local Subscriber

Use the following example to configure the Local subscriber with L2TP tunnel parameters. Optionally you can configure load balancing between multiple LNS servers:

```
configure

context <ctxt_name> [-noconfirm]

    subscriber name <subs_name>

        tunnel l2tp peer-address <lns_ip_address> [ preference <integer> | [ encrypted ]
secret <secret_string> | tunnel-context <context_name> | local-address <local_ip_address>
    }

    load-balancing { random | balanced | prioritized }

end
```

Notes:

- <ctxt_name> is the system context in which you wish to configure the subscriber profile.
- <lns_ip_address> is the IP address of LNS server node and <local_ip_address> is the IP address of system which is bound to LAC service.

Verifying the L2TP Configuration

These instructions are used to verify the L2TP configuration.

- Step 1** Verify that your L2TP configurations were configured properly by entering the following command in Exec Mode in specific context:

```
show subscriber configuration username user_name
```

The output of this command is a concise listing of subscriber parameter settings as configured.

Tunneling All Subscribers in a Specific Context Without Using RADIUS Attributes

As with other services supported by the system, values for subscriber profile attributes not returned as part of a RADIUS Access-Accept message can be obtained using the locally configured profile for the subscriber named default. The subscriber profile for default must be configured in the AAA context (i.e. the context in which AAA functionality is configured).

As a time saving feature, L2TP support can be configured for the subscriber named default with no additional configuration for RADIUS-based subscribers. This is especially useful when you have separate source/AAA contexts for specific subscribers.

To configure the profile for the subscriber named default, follow the instructions above for configuring a local subscriber and enter the name default.

Configuring LAC Services



Important: Not all commands, keywords and functions may be available. Functionality is dependent on platform and license(s).

This section provides information and instructions for configuring LAC services on the system allowing it to communicate with peer LNS nodes.



Important: This section provides the minimum instruction set for configuring LAC service support on the system. For more information on commands that configure additional parameters and options, refer to the *LAC Service Configuration Mode Commands* chapter in the *Command Line Interface Reference*.

To configure the LAC services on system:

- Step 1** Configure the LAC service on system and bind it to an IP address by applying the example configuration in the *Configuring LAC Service* section.
- Step 2** *Optional.* Configure LNS peer information if the Tunnel-Service-Endpoint attribute is not configured in the subscriber profile or PDSN compulsory tunneling is supported by applying the example configuration in the *Configuring LNS Peer* section.
- Step 3** Verify your LAC configuration by following the steps in the Verifying the LAC Service Configuration section.
- Step 4** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Configuring LAC Service

Use the following example to create the LAC service and bind the service to an IP address:

```
configure
  context <dst_ctxt_name> [-noconfirm]
    lac-service <service_name>
      bind address <ip_address>
    end
```

Notes:

- <dst_ctxt_name> is the destination context where you want to configure the LAC service.

Configuring LNS Peer

Use the following example to configure the LNS peers and load balancing between multiple LNS peers:

```
configure

context <dst_ctxt_name> [ -noconfirm ]

lac-service <service_name>

    tunnel selection-key tunnel-server-auth-id

    peer-lns <ip_address> [encrypted] secret <secret> [crypto-map <map_name>
{[encrypted] isakmp-secret <secret> }] [description <text>] [ preference <integer>]

    load-balancing { random | balanced | prioritized }

end
```

Notes:

- <dst_ctxt_name> is the destination context where the LAC service is configured.

Verifying the LAC Service Configuration

These instructions are used to verify the LAC service configuration.

- Step 1** Verify that your LAC service configurations were configured properly by entering the following command in Exec Mode in specific context:

```
show lac-service name service_name
```

The output given below is a concise listing of LAC service parameter settings as configured.

```
Service name: vpn1

Context:                               isp1

Bind:                                  Done

Local IP Address:                      192.168.2.1

First Retransmission Timeout: 1 (secs)

Max Retransmission Timeout: 8 (secs)

Max Retransmissions: 5

Max Sessions: 500000                  Max Tunnels: 32000

Max Sessions Per Tunnel: 512

Data Sequence Numbers: Enabled      Tunnel Authentication: Enabled
```

■ Configuring LAC Services


Keep-alive interval:	60	Control receive window:	16
Max Tunnel Challenge Length:	16		
Proxy LCP Authentication:	Enabled		
Load Balancing:	Random		
Service Status:	Started		
Newcall Policy:	None		

Modifying PDSN Services for L2TP Support

PDSN service modification is required for compulsory tunneling and optional for attribute-based tunneling.

For attribute-based tunneling, a configuration error could occur such that upon successful authentication, the system determines that the subscriber session requires L2TP but can not determine the name of the context in which the appropriate LAC service is configured from the attributes supplied. As a precautionary, a parameter has been added to the PDSN service configuration options that will dictate the name of the context to use. It is strongly recommended that this parameter be configured.

This section contains instructions for modifying the PDSN service configuration for either compulsory or attribute-based tunneling.

 **Important:** This section provides the minimum instruction set for modifying PDSN service for L2TP support on the system. For more information on commands that configure additional parameters and options, refer to the *LAC Service Configuration Mode Commands* chapter in the *Command Line Interface Reference*.

To configure the LAC services on system:

- Step 1** Modify the PDSN service to support L2TP by associating LAC context and defining tunnel type by applying the example configuration in the *Modifying PDSN Service* section.
- Step 2** Verify your configuration to modify PDSN service by following the steps in the *Verifying the PDSN Service for L2TP Support* section.
- Step 3** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Modifying PDSN Service

Use the following example to modify the PDSN service to support L2TP by associating LAC context and defining tunnel type:

```
configure

context <source_ctxt_name> [ -noconfirm ]

pdsn-service <pdsn_service_name>

ppp tunnel-context <lac_context_name>

ppp tunnel-type { l2tp | none }

end
```

Notes:

- *<source_ctxt_name>* is the name of the source context containing the PDSN service, which you want to modify for L2TP support.

- *<pdsn_service_name>* is the name of the pre-configured PDSN service, which you want to modify for L2TP support.
- *<lac_context_name>* is typically the destination context where the LAC service is configured.

Verifying the PDSN Service for L2TP Support

These instructions are used to verify the PDSN service configuration.


Step 1 Verify that your PDSN is configured properly by entering the following command in Exec Mode in specific context:

```
show pdsn-service name pdsn_service_name
```

The output of this command is a concise listing of PDSN service parameter settings as configured.

Modifying APN Templates to Support L2TP

This section provides instructions for adding L2TP support for APN templates configured on the system.

 **Important:** This section provides the minimum instruction set for configuring LAC service support on the system. For more information on commands that configure additional parameters and options, refer to the *LAC Service Configuration Mode Commands* chapter in the *Command Line Interface Reference*.

To configure the LAC services on system:

- Step 1** Modify the APN template to support L2TP with LNS server address and other parameters by applying the example configuration in the *Assigning LNS Peer Address in APN Template* section.
- Step 2** Optional. If L2TP will be used to tunnel transparent IP PDP contexts, configure the APN's outbound username and password by applying the example configuration in the *Configuring Outbound Authentication* section.
- Step 3** Verify your APN configuration by following the steps in the *Verifying the APN Configuration* section.
- Step 4** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Assigning LNS Peer Address in APN Template

Use following example to assign LNS server address with APN template:

```
configure

context <dst_ctxt_name> [-noconfirm]

    apn <apn_name>

        tunnel l2tp [ peer-address <lns_address> [ [ encrypted ] secret <l2tp_secret> ]
        [ preference <integer> ] [ tunnel-context <l2tp_context_name> ] [ local-address
        <local_ip_address> ] [ crypto-map <map_name> { [ encrypted ] isakmp-secret
        <crypto_secret> } ]

    end
```

Notes:

- <dst_ctxt_name> is the name of system destination context in which the APN is configured.
- <apn_name> is the name of the pre-configured APN template which you want to modify for the L2TP support.
- <lns_address> is the IP address of LNS server node and <local_ip_address> is the IP address of system which is bound to LAC service.

Configuring Outbound Authentication

Use the following example to configure the LNS peers and load balancing between multiple LNS peers:

```
configure
  context <dst_ctxt_name> [ -noconfirm ]
    apn <apn_name>
      outbound { [ encrypted ] password <pwd> | username <name> }
    end
```

Notes:

- <dst_ctxt_name> is the destination context where APN template is configured.
- <apn_name> is the name of the pre-configured APN template which you want to modify for the L2TP support.

Verifying the APN Configuration

These instructions are used to verify the APN configuration.

Step 1 Verify that your APN configurations were configured properly by entering the following command in Exec Mode in specific context:


```
show apn name apn_name
```

The output is a concise listing of APN parameter settings as configured.


Appendix N

L2TP Network Server

This chapter describes the support for Layer 2 Tunneling Protocol (L2TP) Network Server (LNS) functionality on Cisco® ASR 5x00 chassis and explains how it is configured. The product Administration Guides provide examples and procedures for configuration of basic services on the system. It is recommended that you select the configuration example that best meets your service model, and configure the required elements for that model, as described in the respective product Administration Guide, before using the procedures in this chapter.

 **Important:** The Layer 2 Tunneling Protocol (L2TP) Network Server (LNS) is a licensed Cisco feature. A separate feature license may be required. Contact your Cisco account representative for detailed information on specific licensing requirements. For information on installing and verifying licenses, refer to the *Managing License Keys* section of the *Software Management Operations* chapter in the *System Administration Guide*.

When enabled through the session license and feature use key, LNS functionality is configured as context-level services on the system. LNS services support the termination of L2TP encapsulated tunnels from L2TP Access Concentrators (LACs) in accordance with RFC 2661.

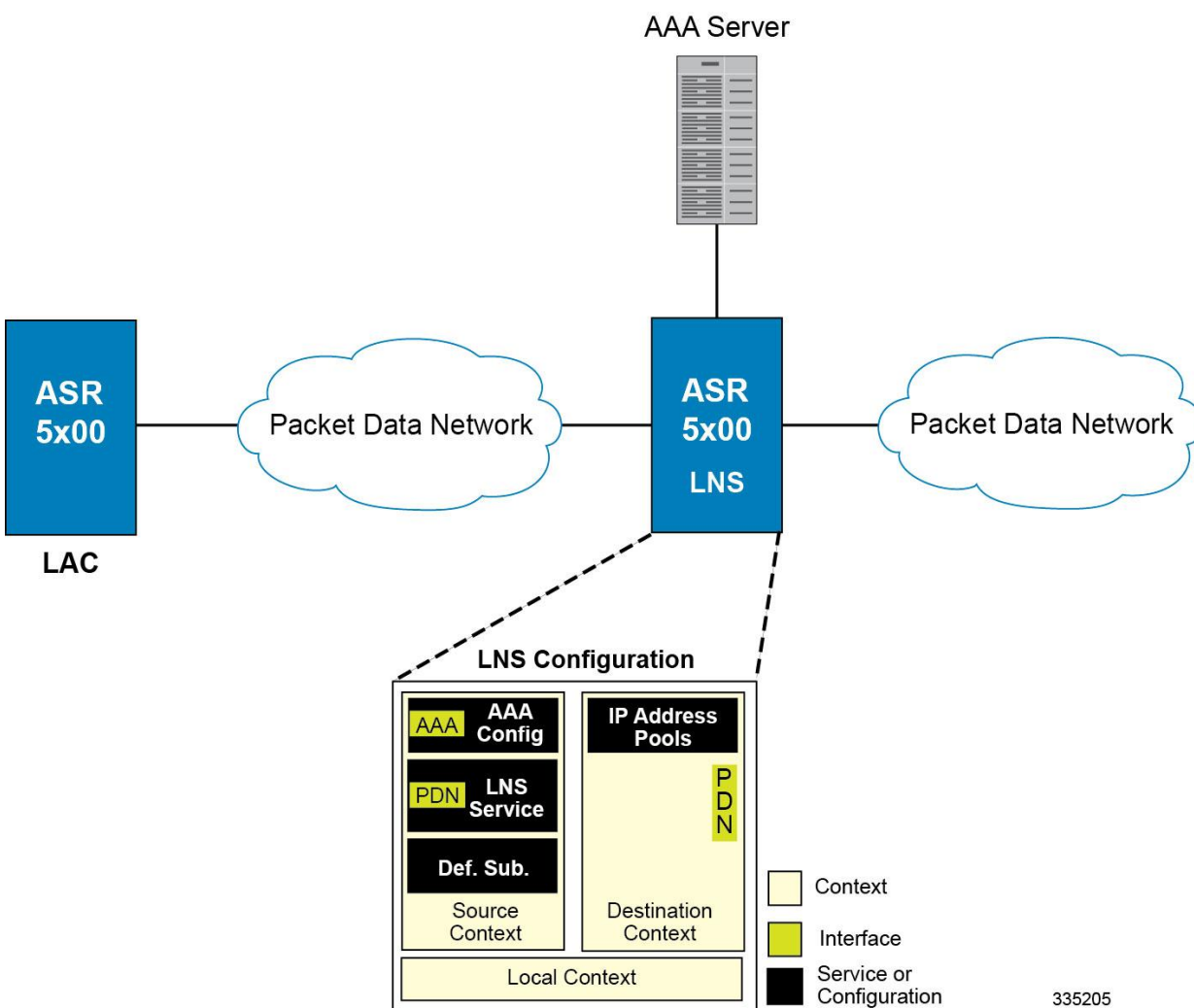
 **Important:** The LNS service uses UDP ports 13660 through 13668 as the source port for receiving packets from the LAC. You can force the LNS to only use the standard L2TP port (UDP Port 1701) with the **single-port-mode** LNS service configuration mode command. Refer to the *Command Line Interface Reference* for more information on this command.

LNS Service Operation

As mentioned previously, LNS functionality on the system is configured via context-level services. LNS services can be configured in the same context as other services supported on the system or in its own context. Each context can support multiple LNS services.

One of the most simple configuration that can be implemented on the system to support Simple IP data applications requires that two contexts (one source and one destination) be configured on the system as shown in the following figure.

Figure 56. LNS Configuration Example



The source context facilitates the LNS service(s) and the PDN and AAA interfaces. The PDN interface is bound to the LNS service and connects L2TP tunnels and sessions from one or more peer LACs. The source context is also be configured to provide AAA functionality for subscriber sessions. The destination context facilitates the packet data network interface(s) and can optionally be configured with pools of IP addresses for assignment to subscriber sessions.

In this configuration, the LNS service in the source context terminates L2TP tunnels from peer LACs and routes the subscriber session data through the destination context to and from a packet data network such as the Internet or a home network.

Information Required

Prior to configuring the system as shown in figure above, a minimum amount of information is required. The following sections describe the information required to configure the source and destination contexts.

Source Context Configuration

The following table lists the information that is required to configure the source context.

Table 49. Required Information for Source Context Configuration

Required Information	Description
Source context name	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the source context will be recognized by the system.
PDN Interface Configuration	
PDN interface name	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the interface will be recognized by the system. Multiple names are needed if multiple interfaces will be configured. These PDN interfaces facilitates the L2TP tunnels/sessions from the LAC and are configured in the source context.
IP address and subnet	These will be assigned to the PDN interface. Multiple addresses and/or subnets are needed if multiple interfaces will be configured.
Physical port number	This specifies the physical port to which the interface will be bound. Ports are identified by the chassis slot number where the line card resides in, followed by the number of the physical connector on the line card. For example, port 17/1 identifies connector number 1 on the card in slot 17. A single physical port can facilitate multiple interfaces.
Physical port description	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the physical port will be recognized by the system. Multiple descriptions are needed if multiple ports will be used. Physical ports are configured within the source context and are used to bind logical PDN interfaces.
Gateway IP address	Used when configuring static routes from the PDN interface(s) to a specific network.
LNS service Configuration	
LNS service name	This is an identification string between 1 and 63 characters (alpha and/or numeric) by which the LNS service will be recognized by the system. Multiple names are needed if multiple LNS services will be used. LNS services are configured in the source context.
Authentication protocols used	Specifies how the system handles authentication: using a protocol (such as CHAP, PAP, or MSCHAP), or not requiring any authentication.
Domain alias for NAI-construction	Specifies a context name for the system to use to provide accounting functionality for a subscriber session. This parameter is needed only if the system is configured to support no authentication.

Required Information	Description
Maximum number of sessions per tunnel	This defines the maximum number of sessions supported by each tunnel facilitated by the LNS service. The number can be configured to any integer value from 1 to 65535. The default is 65535.
Maximum number of tunnels	This defines the maximum number of tunnels supported by the LNS service. The number can be configured to any integer value from 1 to 32000. The default is 32000.
Peer LAC	IP address or network prefix and mask: The IP address of a specific peer LAC for which the LNS service terminates L2TP tunnels. The IP address must be expressed in dotted decimal notation. Multiple peer LACs can be configured. Alternately, to simplify configuration, a group of peer LACs can be specified by entering a network prefix and a mask.
	Secret: The shared secret used by the LNS to authenticate the peer LAC. The secret can be from 1 to 256 alpha and/or numeric characters and is case sensitive.
AAA Interface Configuration	
AAA interface name	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the interface will be recognized by the system. Multiple names are needed if multiple interfaces will be configured. AAA interfaces will be configured in the source context.
IP address and subnet	These will be assigned to the AAA interface. Multiple addresses and/or subnets are needed if multiple interfaces will be configured.
Physical port number	A single physical port can facilitate multiple interfaces.
Physical port description	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the physical port will be recognized by the system. Multiple descriptions are needed if multiple ports will be used. Physical ports are configured within the source context and are used to bind logical AAA interfaces.
Gateway IP address	Used when configuring static routes from the AAA interface(s) to a specific network.
RADIUS Server Configuration	
RADIUS Authentication server	IP Address: Specifies the IP address of the RADIUS authentication server the source context will communicate with to provide subscriber authentication functions. Multiple addresses are needed if multiple RADIUS servers will be configured. RADIUS authentication servers are configured within the source context. Multiple servers can be configured and each assigned a priority.
	Shared Secret: The shared secret is a string between 1 and 15 characters (alpha and/or numeric) that specifies the key that is exchanged between the RADIUS authentication server and the source context. A shared secret is needed for each configured RADIUS server.
	UDP Port Number: Specifies the port used by the source context and the RADIUS authentication server for communications. The UDP port number can be any integer value between 1 and 65535. The default value is 1812.

Required Information	Description
RADIUS Accounting server	IP Address: Specifies the IP address of the RADIUS accounting server that the source context will communicate with to provide subscriber accounting functions. Multiple addresses are needed if multiple RADIUS servers will be configured. RADIUS accounting servers are configured within the source context. Multiple servers can be configured and each assigned a priority.
	Shared Secret: The shared secret is a string between 1 and 15 characters (alpha and/or numeric) that specifies the key that is exchanged between the RADIUS accounting server and the source context. A shared secret is needed for each configured RADIUS server.
	UDP Port Number: Specifies the port used by the source context and the RADIUS Accounting server for communications. The UDP port number can be any integer value between 1 and 65535. The default value is 1813.
RADIUS attribute NAS Identifier	Specifies the name by which the source context will be identified in the Access-Request message(s) it sends to the RADIUS server. The name must be between 1 and 32 alpha and/or numeric characters and is case sensitive.
RADIUS NAS IP address	Specifies the IP address of the source context's AAA interface. A secondary IP address interface can optionally be configured.
Default Subscriber Configuration	
"Default" subscriber's IP context name	Specifies the name of the egress context on the system that facilitates the PDN ports. NOTE: For this configuration, the IP context name should be identical to the name of the destination context.

Destination Context Configuration

The following table lists the information that is required to configure the destination context.

Table 50. Required Information for Destination Context Configuration

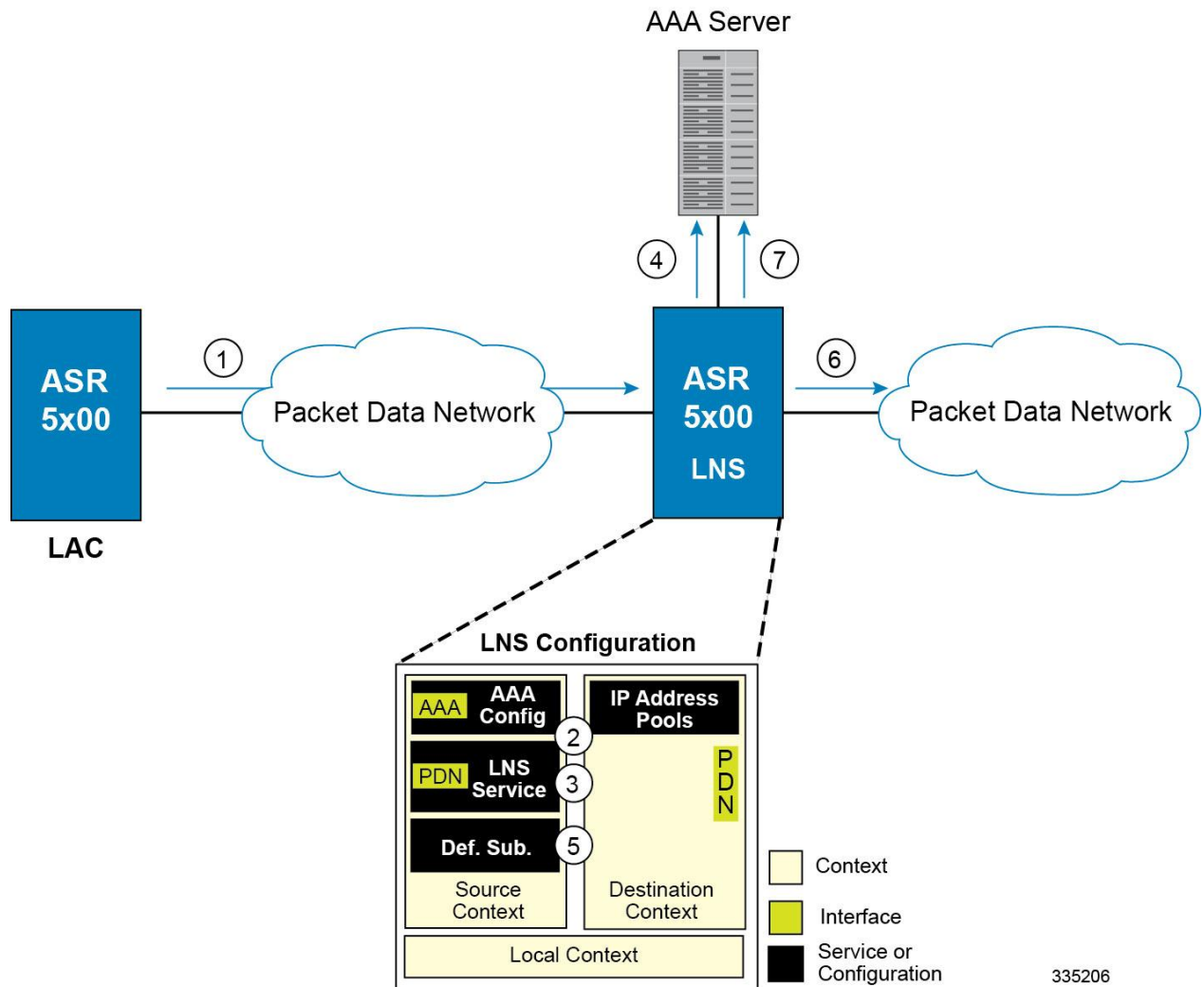
Required Information	Description
Destination context name	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the destination context will be recognized by the system. NOTE: For this configuration, the destination context name should not match the domain name of a specific domain.
PDN Interface Configuration	
PDN interface name	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the interface will be recognized by the system. Multiple names are needed if multiple interfaces will be configured. PDN interfaces are used to connect to a packet network and are configured in the destination context.
IP address and subnet	These will be assigned to the PDN interface. Multiple addresses and/or subnets are needed if multiple interfaces will be configured.

Required Information	Description
Physical port number	A single physical port can facilitate multiple interfaces.
Physical port description(s)	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the physical port will be recognized by the system. Multiple descriptions will be needed if multiple ports will be used. Physical ports are configured within the destination context and are used to bind logical PDN interfaces.
Gateway IP address(es)	Used when configuring static routes from the PDN interface(s) to a specific network.
IP Address Pool Configuration (optional)	
IP address pool name(s)	If IP address pools will be configured in the destination context(s), names or identifiers will be needed for them. The pool name can be between 1 and 31 alpha and/or numeric characters and is case sensitive.
IP pool addresses	An initial address and a subnet, or a starting address and an ending address, are required for each configured pool. The pool will then consist of every possible address within the subnet, or all addresses from the starting address to the ending address. The pool can be configured as public, private, or static.

How This Configuration Works

The following figure and the text that follows describe how this LNS service configuration with a single source and destination context would be used by the system to terminate an L2TP tunnel.

Figure 57. Call Processing Using a Single Source and Destination Context



1. An L2TP tunnel request from a peer LAC is received by the LNS service. The tunnel is to facilitate a subscriber session.
2. The LAC and LNS establish the L2TP tunnel according to the procedures defined in RFC 2661. Once the L2TP tunnel is established, subscriber L2TP sessions can be established.
3. The LNS service determines which context to use in providing AAA functionality for the subscriber session if authentication is enabled for the LNS service. For more information on this process, refer *How the System Selects Contexts in System Administration Guide*.
For this example, the result of this process is that LNS service determined that AAA functionality should be provided by the Source context.
4. The system communicates with the AAA server specified in the Source context's AAA configuration to authenticate the subscriber.
5. Upon successful authentication, the LNS service terminates the subscriber's PPP datagrams from the L2TP session and the system determines which egress context to use for the subscriber session. For more information on egress context selection process, refer *How the System Selects Contexts in System Administration Guide*.


The system determines that the egress context is the destination context based on the configuration of either the Default subscriber's ip-context name or from the SN-VPN-NAME or SN1-VPN-NAME attributes that is configured in the subscriber's RADIUS profile.

6. Data traffic for the subscriber session is routed through the PDN interface in the Destination context.
7. Accounting information for the session is sent to the AAA server over the AAA interface.

Configuring the System to Support LNS Functionality

Many of the procedures required to configure the system to support LNS functionality are provided in the System Administration Guide. The System Administration Guide provides information and procedures for configuring contexts, interfaces and ports, AAA functionality, and IP address pools on the system.

This section provides information and instructions for configuring LNS services on the system allowing it to communicate with peer LAC nodes.

 **Important:** This section provides the minimum instruction set for configuring an LNS service allowing the system to terminate L2TP tunnels and process data sessions. For more information on commands that configure additional LNS service properties, refer LNS Configuration Mode Commands chapter in *Command Line Interface Reference*.

To configure the system to provide access control list facility to subscribers:

- Step 1** Create the LNS service and bind it to an interface IP address by applying the example configuration in the *Creating and Binding LNS Service* section.
- Step 2** Specify the authentication parameters for LNS service by applying the example configuration in the *Configuring Authentication Parameters for LNS Service* section.
- Step 3** Configure the maximum number of tunnels supported by the LNS service and maximum number of sessions supported per tunnel by applying the example configuration in the *Configuring Tunnel and Session Parameters for LNS Service* section.
- Step 4** Configure peer LACs for the LNS service by applying the example configuration in the *Configuring Tunnel and Session Parameters for LNS Service* section.
- Step 5** *Optional.* Specify the domain alias designated for the context which the LNS service uses for AAA functionality by applying the example configuration in the *Configuring Domain Alias for AAA Subscribers* section.
- Step 6** Verify your LNS service configuration by following the steps in the *Verifying the LNS Service Configuration* section.
- Step 7** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Creating and Binding LNS Service

Use the following example to create the LNS service and bind the IP address to it:

```
configure

context <dest_ctxt_name> -noconfirm

    lns-service <lns_svc_name> -noconfirm

        bind address <ip_address> [ max-subscribers <max_subscriber> ]
```

```
end
```

Notes:

- LNS service has to be configured in destination context.
- Bind address is the interface address that is to serve as an L2TP PDN interface.
- Multiple addresses on the same IP interface can be bound to different LNS services. However, each address can be bound to only one LNS service. In addition, the LNS service can not be bound to the same interface as other services such as a LAC service.

Configuring Authentication Parameters for LNS Service

Use the following example to authentication parameters for LNS service:

```
configure

context <dest_ctxt_name>

    lns-service <lns_svc_name>

        authentication { { [ allow-noauth | chap <pref> | mschap <pref> | | pap <pref> ]
    } | msid-auth }

end
```

Note:

- For more information on authentication procedure and priorities, refer **authentication** command section in LNS Configuration Mode Commands chapter of the *Command Line Interface Reference*.

Configuring Tunnel and Session Parameters for LNS Service

Use the following example to configure the tunnel and session parameters for LNS service:

```
configure

context <dest_ctxt_name>

    lns-service <lns_svc_name>

        max-tunnel <max_tunnels>

        max-session-per-tunnel <max_sessions>

end
```

Note:

- For more information on tunnel and session related parameters, refer LNS Configuration Mode Commands chapter of the *Command Line Interface Reference*.

Configuring Peer LAC servers for LNS Service

Use the following example to configure the peer LAC servers for LNS service:

```
configure

  context <dest_ctxt_name>

    lns-service <lns_svc_name>

      peer-lac { <lac_ip_address> | <ip_address>/<mask> } [ encrypted ] secret
<secret_string> [ description <desc_text> ]

    end
```

Note:

- Multiple LACs can be configured with this command. For more information, refer LNS Configuration Mode Commands chapter of the *Command Line Interface Reference*.

Configuring Domain Alias for AAA Subscribers

Use the following example to create the LNS service and bind the IP address to it:

```
configure

  context <dest_ctxt_name> -noconfirm

    lns-service <lns_svc_name> -noconfirm

      nai-construct domain <domain_alias>

    end
```

Note:

- If this command is enabled, an NAI is constructed for the subscriber in the event that their mobile node does not negotiate CHAP, PAP, or MSCHAP.
- If this option is selected, no further attempts are made to authenticate the user. Instead, the constructed NAI is used for accounting purposes.



Important: This command should only be used if the LNS service is configured to allow “no authentication” using the **authentication allow-noauth** command.

Verifying the LNS Service Configuration

These instructions are used to verify the LNS service configuration.

Step 1 Verify that your LNS service configuration by entering the following command in Exec Mode:

```
show lns-service name service_name
```

The output of this command displays the configuration of the LNS service and should appear similar to that shown below.

```

Service name: testlns

Context:                test

Bind:                   Not Done

Local IP Address:       0.0.0.0

First Retransmission Timeout: 1 (secs)

Max Retransmission Timeout: 8 (secs)

Max Retransmissions:    5

Setup Timeout:          60 (secs)

Max Sessions:           500000      Max
Tunnels:                 32000

Max Sessions Per Tunnel: 65535

Keep-alive Interval:    60          Control Receive Window: 16

Data Sequence Numbers:  Enabled

Tunnel Authentication:  Enabled

Tunnel Switching:      Enabled

Max Tunnel Challenge Length: 16

PPP Authentication:     CHAP 1 PAP 2

Allow Noauthentication: Disabled      MSID
Authentication:         Disabled

No NAI Construct Domain defined

No Default Subscriber defined

IP Src Violation Reneg Limit: 5

IP Src Violation Drop Limit: 10

IP Src Violation Period: 120 (secs)

Service Status:         Not started

Newcall Policy:         None

```

Appendix O

Mobile IP Registration Revocation

This chapter describes Registration Revocation for Mobile-IP and Proxy Mobile-IP and explains how it is configured. The product administration guides provide examples and procedures for configuration of basic services on the system. It is recommended that you select the configuration example that best meets your service model and configure the required elements for that model, as described in this administration guide before using the procedures in this chapter.



Important: This license is enabled by default; however, not all features are supported on all platforms and other licenses may be required for full functionality as described in this chapter.

Overview

Registration Revocation is a general mechanism whereby either the HA or the FA providing Mobile IP functionality to the same mobile node can notify the other mobility agent of the termination of a binding. This functionality provides the following benefits:

- Timely release of Mobile IP resources at the FA and/or HA
- Accurate accounting
- Timely notification to mobile node of change in service

Mobile IP Registration Revocation can be triggered at the FA by any of the following:

- Session terminated with mobile node for whatever reason
- Session renegotiation
- Administrative clearing of calls
- Session Manager software task outage resulting in the loss of FA sessions (sessions that could not be recovered)



Important: Registration Revocation functionality is also supported for Proxy Mobile IP. However, only the HA can initiate the revocation for Proxy-MIP calls.

Mobile IP Registration Revocation can be triggered at the HA by any of the following:

- Administrative clearing of calls
- Inter-Access Gateway handoff. This releases the binding at the previous access gateway/FA
- Session Manager software task outage resulting in the loss of FA sessions (for sessions that could not be recovered)
- Session Idle timer expiry (when configured to send Revocation)
- Any other condition under which a binding is terminated due to local policy (duplicate IMSI detected, duplicate home address requested, etc.)

The FA and the HA negotiate Registration Revocation support when establishing a Mobile IP call. Revocation support is indicated to the Mobile Node (MN) from the FA by setting the 'X' bit in the Agent Advertisement to MN. However the MN is not involved in negotiating the Revocation for a call or in the Revocation process. It only gets notified about it. The X bit in the Agent Advertisements is just a hint to the MN that revocation is supported at the FA but is not a guarantee that it can be negotiated with the HA

At the FA, if revocation is enabled and a FA-HA SPI is configured, the Revocation Support extension is appended to the RRQ received from the MN and protected by the FA-HA Authentication Extension. At the HA, if the RRQ is accepted, and the HA supports revocation, the HA responds with an RRP that includes the Revocation Support extension. Revocation support is considered to be negotiated for a binding when both sides have included a Revocation Support Extension during a successful registration exchange.



Important: The Revocation Support Extension in the RRQ or RRP must be protected by the FA-HA Authentication Extension. Therefore, an FA-HA SPI must be configured at the FA and the HA for this to succeed.

If revocation is enabled at the FA, but an FA-HA SPI is not configured at the FA for a certain HA, then FA does not send Revocation Support Extension for a call to that HA. Therefore, the call may come up without Revocation support negotiated.

If the HA receives an RRQ with Revocation Support Extension, but not protected by FA-HA Auth Extension, it will be rejected with “FA Failed Authentication” error.


If the FA receives a RRP with Revocation Support Extension, but not protected by FA-HA Auth Extension, it will be rejected with “HA Failed Authentication” error.


Also note that Revocation support extension is included in the initial, renewal or handoff RRQ/RRP messages. The Revocation extension is not included in a Deregistration RRQ from the FA and the HA will ignore them in any Deregistration RRQs received.

Configuring Registration Revocation

Support for MIP Registration Revocation requires the following configurations:

- **FA service(s):** Registration Revocation must be enabled and operational parameters optionally configured.
- **HA service(s):** Registration Revocation must be enabled and operational parameters optionally configured.

 **Important:** These instructions assume that the system was previously configured to support subscriber data sessions for a core network service with FA and/or an HA according to the instructions described in the respective product Administration Guide.

 **Important:** Commands used in the configuration samples in this section provide base functionality to the extent that the most common or likely commands and/or keyword options are presented. In many cases, other optional commands and/or keyword options are available. Refer to the *Command Line Interface Reference* for complete information regarding all commands.

Configuring FA Services

Configure FA services to support MIP Registration Revocation by applying the following example configuration:

```
configure

context <context_name>

    fa-service <fa_service_name>

        revocation enable

        revocation max-retransmission <number>

        revocation retransmission-timeout <time>

    end
```

Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Configuring HA Services

Configure HA services to support MIP Registration Revocation by applying the following example configuration:

```
configure

context <context_name>

    ha-service <ha_service_name>
```

```
revocation enable

revocation max-retransmission <number>

revocation retransmission-timeout <time>

end
```

Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Appendix P

Policy Forwarding

This chapter provides information on configuring an enhanced, or extended, service. The product administration guides provide examples and procedures for configuration of basic services on the system. It is recommended that you select the configuration example that best meets your service model and configure the required elements for that model before using the procedures in this chapter.

Sections in this chapter include:

- [Overview](#)
- [IP Pool-based Next Hop Forwarding](#)
- [Subscriber-based Next Hop Forwarding](#)
- [ACL-based Policy Forwarding](#)

Overview

The system can be configured to automatically forward data packets to a predetermined network destination. This can be done in one of three ways:

- IP Pool-based Next Hop Forwarding - Forwards data packets based on the IP pool from which a subscriber obtains an IP address.
- ACL-based Policy Forwarding - Forwards data packets based on policies defined in Access Control Lists (ACLs) and applied to contexts or interfaces.
- Subscriber specific Next Hop Forwarding - Forwards all packets for a specific subscriber.

The simplest way to forward subscriber data is to use IP Pool-based Next Hop Forwarding. An IP pool is configured with the address of a next hop gateway and data packets from all subscribers using the IP pool are forward to that gateway.

Subscriber Next Hop forwarding is also very simple. In the subscriber configuration a nexthop forwarding address is specified and all data packets for that subscriber are forwarded to the specified nexthop destination.

ACL-based Policy Forwarding gives you more control on redirecting data packets. By configuring an Access Control List (ACL) you can forward data packets from a context or an interface by different criteria, such as; source or destination IP address, ICMP type, or TCP/UDP port numbers.

ACLs are applied first. If ACL-based Policy Forwarding and Pool-based Next Hop Forwarding or Subscriber are configured, data packets are first redirected as defined in the ACL, then all remaining data packets are redirected to the next hop gateway defined by the IP pool or subscriber profile.

IP Pool-based Next Hop Forwarding

When an IP pool in a destination context has a Next Hop Forwarding address specified, any subscriber that obtains an IP address from that IP pool has all data coming from the mobile node automatically forwarded to the specified Next Hop Forwarding address.

For more information on creating IP pools, refer to the *System Administration Guide* and for additional information on the `ip pool` command, refer to the *Command Line Interface Reference*.

Configuring IP Pool-based Next Hop Forwarding

Configure Next Hop Forwarding on an existing IP Pool in a destination context by applying the following example configuration:

```
configure

  context <context_name>

    ip pool <pool_name> nexthop-forwarding-address <forwarding_ip_address>

  end
```

Save the configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Subscriber-based Next Hop Forwarding

When a subscriber configuration has a Next Hop Forwarding address specified, any sessions authenticated as that subscriber have all data coming from the mobile node automatically forwarded to the specified Next Hop Forwarding address.

Configuring Subscriber-based Next Hop Forwarding

Configure Next Hop Forwarding for a specific subscriber by applying the following example configuration:

```
configure
  context <context_name>
    subscriber name <subs_name>
      nexthop-forwarding-address <forwarding_ip_address>
    end
```

Save the configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

ACL-based Policy Forwarding

ACL-based Policy Forwarding is a feature in the system that forwards subscriber data based on policies defined in Access Control Lists (ACLs). When ACLs are applied to access groups, priorities are given to the ACLs. The ACL applied with the highest priority is used to define the policy that is used for forwarding the subscriber data.



Important: Refer to *Access Control Lists* for additional information on creating and using ACLs.

Configuring ACL-based Policy Forwarding

Configure ACL-based Policy Forwarding by applying the following example configuration:

```
configure

context <context_name>

    ip access-list <acl_name>

        redirect <interface_name> <next_hop_address> <criteria>

    exit
```

The following example specifies that any IP packet coming from any system on the 192.168.55.0 network that has a destination host address of 192.168.80.1 is to be redirected, or forwarded, through the system interface named *interface2* to the host at 192.168.23.12:

```
redirect interface2 192.168.23.12 ip 192.168.55.0 255.255.255.0 host 192.168.80.1
```

Save the configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Applying the ACL to an IP Access Group

To apply the ACL to the IP access group for the current destination context, go to *Applying the ACL to a Destination Context*.

To apply the ACL to the IP access group for an interface in the current destination context, go to [Applying the ACL to an Interface in a Destination Context](#).

Applying the ACL to a Destination Context

Step 1 At the context configuration mode prompt, enter the following command:

```
ip access-group <acl_name> {in | out} <priority-value>
```

- Step 2** Save the configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Applying the ACL to an Interface in a Destination Context

- Step 1** Set parameters for inbound data by applying the following example configuration:

```
configure
  context <context_name>
    interface <interface_name>
      ip access-group <acl_name> in <priority-value>
    end
```

- Step 2** Set parameters for outbound data by applying the following example configuration:

```
configure
  context <context_name>
    interface <interface_name>
      ip access-group <acl_name> out <priority-value>
    end
```

- Step 3** Save the configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Appendix Q

Pre-paid Billing

This chapter provides information on configuring an enhanced, or extended, service. The product administration guides provides examples and procedures for configuration of basic services on the system. It is recommended that you select the configuration example that best meets your service model and configure the required elements for that model before using the procedures in this chapter.

This chapter includes the following topics:

- [Overview](#)
- [Configuring Standard 3GPP2 Pre-paid Billing](#)
- [Configuring Pre-paid Billing With Custom Behavior](#)
- [3GPP2 Pre-paid Attributes](#)
- [Pre-paid Attributes](#)

Overview

The system supports pre-paid billing for subscriber accounts that use RADIUS Accounting.

The system supports two methods of implementing Pre-paid Billing Support; Standard 3GPP2 Pre-paid Billing and Custom Pre-paid Billing. The 3GPP2 standard is the recommended implementation.

3GPP2 Standard Pre-paid Billing Overview

The prepaid packet data service allows a user to purchase access to the network in advance, based on either volume or duration. When a user connects to a service, the Prepaid Client (PPC) contacts the Prepaid Server (PPS) and verifies that the user has available credits for the service. When a user runs out of credits, service is terminated until the user purchases additional credits.

The Prepaid Data Service implementation is compliant with 3GPP2 IS-835-C. This solution provides a standards based implementation that can effectively interoperate with additional vendors equipment when required. The system primarily uses the PPAC (PrePaid Accounting Capability) and PPAQ (PrePaid Accounting Quota) VSAs to implement PrePaid service. The PPAC VSA is used to determine the capabilities of the PPC. When the PPC sends the PPAC VSA it specifies if it supports duration, volume or both types of PrePaid service. When the PPS sends a PPAC VSA it specifies the type of PrePaid service to use for the particular session. The PPAQ VSA specifies the characteristics of the PrePaid accounting service. This includes quota & threshold values for both duration and volume PrePaid service. Through the use of these VSAs, the PPC and PPS communicate the status of the session and when the user has run out of quota, the service can be terminated.

The PrePaid Client resides on the system and communicates with the PPS through the use of RADIUS messages exchanged with the RADIUS server.

Custom Pre-paid Billing Overview

In the Access-Accept from the RADIUS server the system receives attributes which indicate the number of byte credits available for the subscriber. Byte throughput can be pre-paid for traffic inbound to the system, outbound from the system, or an amount that combines both inbound and outbound traffic. Five attributes are used: one for traffic inbound to the system, one for traffic outbound from the system, one that combines traffic in both directions, one that only indicates that the user should be re-authenticated regardless of the byte counters, and one for the low watermark in percent.

The low watermark value is multiplied by the number of byte credits granted in the Access-Accept to arrive at a threshold. Once the number of byte credits remaining is lower than this number, a new Access-Request is issued. If the Access-Request is issued because the Low Watermark has been reached, then a new Low Watermark is calculated from the number of byte credits granted in the Access-Accept, but only if the number of byte credits granted is a non-zero value. If the Access-Request is issued for any other reason, then the Low Watermark is not re-calculated.

The system re-authorizes an active subscriber that has used up its byte credits by issuing a RADIUS Access-Request to the RADIUS server. A valid Access-Reject or a RADIUS timeout results in immediate disconnect of the subscriber session. An Access-Accept without attributes that authorize more byte credits allows the subscriber session to continue with the remaining credits. An Access-Accept with attributes containing byte credits results in the addition of these byte credits to the subscriber session, and the continuation of the session until the subscriber session byte credits have been reduced to the low watermark received in the access accept. If not received, it defaults to 10%.

The system continues to service the subscriber session while the RADIUS request for re-authorization is in process. If the counter reaches zero before the response the subscriber session is terminated immediately.

You can configure Pre-paid Billing support for standard 3GPP2 behavior or custom behavior where you can specify whether or to measure the byte-count on compressed or non-compressed data, set a low-watermark for accounting, and specify a credit renewal interval in the default subscriber configuration for a context or a domain alias.

License Requirements

The Pre-paid Billing is a licensed Cisco feature. A separate feature license may be required. Contact your Cisco account representative for detailed information on specific licensing requirements. For information on installing and verifying licenses, refer to the *Managing License Keys* section of the *Software Management Operations* chapter in the *System Administration Guide*.

Configuring Standard 3GPP2 Pre-paid Billing

This section describes how to enable standard 3GPP2 pre-paid billing support.



Important: Commands used in the configuration samples in this section provide base functionality to the extent that the most common or likely commands and/or keyword options are presented. In many cases, other optional commands and/or keyword options are available. Refer to the *Command Line Interface Reference* for complete information regarding all commands.

Enable pre-paid billing for the default subscriber by applying the following example configuration:

configure

```
context <context_name>

  subscriber default

    prepaid 3gpp2 accounting

  end
```

Enable pre-paid billing for the default subscriber of a domain alias by applying the following example configuration:

configure

```
context <context_name>

  subscriber name <alias_def_sub>

    prepaid 3gpp2 accounting

  end
```

Notes:

- You may add the optional keyword **no-final-access-request** to the **prepaid 3gpp2 accounting** command to stop sending the final online access-request on termination of 3GPP2 prepaid sessions.
- Optional commands: If both duration and volume attributes are received, default preference is given to the duration attribute. To set the preference to the volume attribute, enter the following command:

```
prepaid 3gpp2 preference volume
```

Note that this command alone does not enable pre-paid support. The **prepaid 3gpp2 accounting** command must be executed as shown to enable pre-paid support.

If you are using duration-based quota usage accounting, use the following command to define what behavior specifies the end of the billing duration. The default behavior is the duration quota algorithm set to current-time.


```
prepaid 3gpp2 duration-quota final-duration-algorithm [ current-time | last-airlink-activity-time | last-user-layer3-activity-time ]
```


Note that this command alone does not enable pre-paid support. The **prepaid 3gpp2 accounting** command must be executed as shown to enable pre-paid support.


Save the configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Configuring Pre-paid Billing With Custom Behavior

This section describes how to enable Pre-paid billing support with custom behavior.

 **Important:** If RADIUS attributes are present that conflict with the custom pre-paid settings, the values set by the RADIUS attributes take precedence.

 **Important:** Pre-paid billing support is not available for local subscribers. Even though you can set pre-paid parameters for a local subscriber from the CLI, these settings have no effect on a subscriber session.

 **Important:** Commands used in the configuration samples in this section provide base functionality to the extent that the most common or likely commands and/or keyword options are presented. In many cases, other optional commands and/or keyword options are available. Refer to the *Command Line Interface Reference* for complete information regarding all commands.

Enable custom pre-paid billing for the default subscriber by applying the following example configuration:

configure

```
context <context_name>

    subscriber default

        prepaid custom

    end
```

Enable custom pre-paid billing for the default subscriber of a domain alias by applying the following example configuration:

configure

```
context <context_name>

    subscriber name <alias_def_sub>

        prepaid custom

    end
```

Notes:

- *Optional:* To have custom pre-paid byte credits based on the flow of compressed traffic, use the following command:

```
prepaid custom byte-count compressed
```
- *Optional:* Set the low-watermark for remaining byte credits. This is a percentage of the subscriber session's total credits. When the low-watermark is reached a new RADIUS access-request is sent to the RADIUS server to retrieve more credits. To set the low watermark percentage, enter the following command:

```
prepaid custom low-watermark percent <percentage>
```


- *Optional:* Set the time in seconds to wait before sending a new RADIUS access-request to the RADIUS server to retrieve more credits by entering the following command:

prepaid custom renewal interval <seconds>

- Save the configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

3GPP2 Pre-paid Attributes

Use the attributes listed in the following table to configure a subscriber for 3GPP2 pre-paid billing:

Attribute	Sub-attribute	Description
3GPP2-Pre-Paid-Acct-Capability		This attribute is for setting the prepaid accounting capability.
	Available-In-Client	The optional Available-In-Client Sub-Type, generated by the PrePaid client, indicates the PrePaid Accounting capabilities of the client in the PDSN or HA and shall be bitmap encoded.
	Selected-For-Session	The optional Selected-For-Session Sub-Type, generated by the PrePaid server, indicates the PrePaid Accounting capability to be used for a given session.
3GPP2-Pre-Paid-Accounting-Quota		This attribute specifies the characteristics for PrePaid accounting of the volume and/or duration of a packet data session. It shall be present in all on-line RADIUS Access-Request and on-line RADIUS Access-Accept messages and may be included in other RADIUS Access-Accept messages. Non-used Sub-Types by the PPC and PPS shall be omitted.
	Quota-Identifier	The Quota-Identifier Sub-Type is generated by the PrePaid server at allocation of a Volume and/or Duration Quota. The on-line quota update RADIUS Access-Request message sent from the PPC to the PPS shall include a previously received Quota-Identifier.
	Volume-Quota	The optional Volume-Quota Sub-Type is only present if Volume Based charging is used. In RADIUS Access-Accept message (PPS to PPC direction), it indicates the Volume (in octets) allocated for the session by the PrePaid server. In on-line RADIUS Access-Request message (PPC to PPS direction), it indicates the total used volume (in octets) for both forward and reverse traffic applicable to PrePaid accounting ¹³ . If a Tariff Switch condition was reached during the session, this Sub-Type contains the complete (before and after) volume used, while the Volume-Used-After-Tariff-Switch attribute contains the volume used after the tariff switch condition.
	Volume-Quota-Overflow	The optional Volume-Quota-Overflow Sub-Type is used to indicate how many times the Volume-Quota counter has wrapped around 2^{32} over the course of the service being provided.
	Volume-Threshold	The Volume-Threshold Sub-Type shall always be present if Volume-Quota is present in a RADIUS Access-Accept message (PPS to PPC direction). It is generated by the PrePaid server and indicates the volume (in octets) that shall be used before requesting quota update. This threshold should not be larger than the Volume-Quota.
	Volume-Threshold-Overflow	The optional Volume-Threshold-Overflow Sub-Type is used to indicate how many times the Volume-Threshold counter has wrapped around 2^{32} over the course of the service being provided.
	Duration-Quota	The optional Duration-Quota Sub-Type is only present if Duration Based charging is used. In RADIUS Access-Accept message (PPS to PPC direction), it indicates the Duration (in seconds) allocated for the session by the PrePaid server. In on-line RADIUS Access-Accept message (PPC to PPS direction), it indicates the total Duration (in seconds) since the start of the accounting session related to the Quota-ID.

Attribute	Sub-attribute	Description
	Duration-Threshold	The Duration-Threshold Sub-Type shall always be present if Duration-Quota is present in a RADIUS Access-Accept message (PPS to PPC direction). It represents the duration (in seconds) that shall be used by the session before requesting quota update. This threshold should not be larger than the Duration-Quota and shall always be sent with the Duration-Quota.
	Update-Reason	The Update-Reason Sub-Type shall be present in the on-line RADIUS Access-Request message (PPC to PPS direction). It indicates the reason for initiating the on-line quota update operation. Update reasons 4, 5, 6, 7 and 8 indicate that the associated resources are released at the client side, and therefore the PPS shall not allocate a new quota in the RADIUS Access-Accept message.
	Pre-Paid-Server	The optional, multi-value PrePaid-Server indicates the address of the serving PrePaid System. If present, the Home RADIUS server uses this address to route the message to the serving PrePaid Server. The attribute may be sent by the Home RADIUS server. If present in the incoming RADIUS Access-Accept message, the PDSN shall send this attribute back without modifying it in the subsequent RADIUS Access-Request message, except for the first one. If multiple values are present, the PDSN shall not change the order of the attributes.

These attributes can be found in the following dictionaries:

- 3gpp2
- 3gpp2-835
- starent
- starent-835
- starent-vs1
- starent-vs1-835

For more information, refer to the *AAA and GTP Interface Administration and Reference*.

Pre-paid Attributes

Use the attributes listed in the following table to configure a subscriber for pre-paid billing;

Attribute	Description
SN-Prepaid-Inbound-Octets	If only SN-Prepaid-Inbound-Octets is in the Access-Accept, and the others are not, then the number of outbound credits is infinite.
SN-Prepaid-Outbound-Octets	If only SN-Prepaid-Outbound-Octets is in the Access-Accept, and the others are not, then the number of inbound credits is infinite.
SN-Prepaid-Total-Octets	If only SN-Prepaid-Total-Octets is in the Access-Accept, and the others are not, then pre-paid credits is only enforced on the total byte throughput.
SN-Prepaid-Timeout	SN-Prepaid-Timeout can be used alone or in combination with the other attributes. This integer RADIUS attribute includes a time limit in seconds. Regardless of the values of the Octet counters, the session should send a new authorization request upon timer expiration.
SN-Prepaid-Watermark	SN-Prepaid-Watermark is optional with any of the attributes. If it is not included it defaults to the CLI default subscriber configuration, which defaults to a value of 10%. This watermark applies to any of the pre-paid attributes being enforced.

These attributes can be found in the following dictionaries:

- starent
- starent-vs1
- starent-835
- starent-vs1-835
- custom1 through custom9

Refer to the *AAA and GTP Interface Administration and Reference* for more details.

Appendix R

Proxy-Mobile IP

This chapter describes system support for Proxy Mobile IP and explains how it is configured. The product administration guides provide examples and procedures for configuration of basic services on the system. It is recommended that you select the configuration example that best meets your service model before using the procedures in this chapter.


Proxy Mobile IP provides a mobility solution for subscribers with mobile nodes (MNs) capable of supporting only Simple IP.

This chapter includes the following sections:

- [Overview](#)
- [How Proxy Mobile IP Works in 3GPP2 Network](#)
- [How Proxy Mobile IP Works in 3GPP Network](#)
- [How Proxy Mobile IP Works in WiMAX Network](#)
- [How Proxy Mobile IP Works in a WiFi Network with Multiple Authentication](#)
- [Configuring Proxy Mobile-IP Support](#)

Overview

Proxy Mobile IP provides mobility for subscribers with MNs that do not support the Mobile IP protocol stack.

 **Important:** Proxy Mobile IP is a licensed Cisco feature. A separate feature license may be required. Contact your Cisco account representative for detailed information on specific licensing requirements. For information on installing and verifying licenses, refer to the *Managing License Keys* section of the *Software Management Operations* chapter in the *System Administration Guide*.

The Proxy Mobile IP feature is supported for various products. The following table indicates the products on which the feature is supported and the relevant sections within the chapter that pertain to that product.

Table 51. Applicable Products and Relevant Sections

Applicable Product(s)	Refer to Sections
PDSN	<ul style="list-style-type: none"> • Proxy Mobile IP in 3GPP2 Service • How Proxy Mobile IP Works in 3GPP2 Network • Configuring FA Services • Configuring Proxy MIP HA Failover • Configuring HA Services • Configuring Subscriber Profile RADIUS Attributes • RADIUS Attributes Required for Proxy Mobile IP • Configuring Local Subscriber Profiles for Proxy-MIP on a PDSN • Configuring Default Subscriber Parameters in Home Agent Context
GGSN	<ul style="list-style-type: none"> • Proxy Mobile IP in 3GPP Service • How Proxy Mobile IP Works in 3GPP Network • Configuring FA Services • Configuring Proxy MIP HA Failover • Configuring HA Services • Configuring Subscriber Profile RADIUS Attributes • RADIUS Attributes Required for Proxy Mobile IP • Configuring Default Subscriber Parameters in Home Agent Context • Configuring APN Parameters

Applicable Product(s)	Refer to Sections
ASN GW	<ul style="list-style-type: none"> • Proxy Mobile IP in WiMAX Service • How Proxy Mobile IP Works in WiMAX Network • Configuring FA Services • Configuring Proxy MIP HA Failover • Configuring HA Services • Configuring Subscriber Profile RADIUS Attributes • RADIUS Attributes Required for Proxy Mobile IP • Configuring Default Subscriber Parameters in Home Agent Context
PDIF	<ul style="list-style-type: none"> • How Proxy Mobile IP Works in a WiFi Network with Multiple Authentication • Configuring FA Services • Configuring Proxy MIP HA Failover • Configuring HA Services • Configuring Subscriber Profile RADIUS Attributes • RADIUS Attributes Required for Proxy Mobile IP • Configuring Default Subscriber Parameters in Home Agent Context

Proxy Mobile IP in 3GPP2 Service

For subscriber sessions using Proxy Mobile IP, R-P and PPP sessions get established between the MN and the PDSN as they would for a Simple IP session. However, the PDSN/FA performs Mobile IP operations with an HA (identified by information stored in the subscriber's profile) on behalf of the MN (i.e. the MN is only responsible for maintaining the Simple IP PPP session with PDSN).

The MN is assigned an IP address by either the PDSN/FA or the HA. Regardless of its source, the address is stored in a mobile binding record (MBR) stored on the HA. Therefore, as the MN roams through the service provider's network, each time a hand-off occurs, the MN will continue to use the same IP address stored in the MBR on the HA.

Note that unlike Mobile IP-capable MNs that can perform multiple sessions over a single PPP link, Proxy Mobile IP allows only a single session over the PPP link. In addition, simultaneous Mobile and Simple IP sessions will not be supported for an MN by the FA that is currently facilitating a Proxy Mobile IP session for the MN.

The MN is assigned an IP address by either the HA, a AAA server, or on a static-basis. The address is stored in a mobile binding record (MBR) stored on the HA. Therefore, as the MN roams through the service provider's network, each time a hand-off occurs, the MN will continue to use the same IP address stored in the MBR on the HA.

Proxy Mobile IP in 3GPP Service

For IP PDP contexts using Proxy Mobile IP, the MN establishes a session with the GGSN as it normally would. However, the GGSN/FA performs Mobile IP operations with an HA (identified by information stored in the subscriber's profile) on behalf of the MN (i.e. the MN is only responsible for maintaining the IP PDP context with the GGSN, no Agent Advertisement messages are communicated with the MN).

The MN is assigned an IP address by either the HA, a AAA server, or on a static-basis. The address is stored in a mobile binding record (MBR) stored on the HA. Therefore, as the MN roams through the service provider's network, each time a hand-off occurs, the MN will continue to use the same IP address stored in the MBR on the HA.

Proxy Mobile IP can be performed on a per-subscriber basis based on information contained in their user profile, or for all subscribers facilitated by a specific APN. In the case of non-transparent IP PDP contexts, attributes returned from the subscriber's profile take precedence over the configuration of the APN.

Proxy Mobile IP in WiMAX Service

For subscriber sessions using Proxy Mobile subscriber sessions get established between the MN and the ASN GW as they would for a Simple IP session. However, the ASN GW/FA performs Mobile IP operations with an HA (identified by information stored in the subscriber's profile) on behalf of the MN (i.e. the MN is only responsible for maintaining the Simple IP subscriber session with ASN GW).

The MN is assigned an IP address by either the ASN GW/FA or the HA. Regardless of its source, the address is stored in a mobile binding record (MBR) stored on the HA. Therefore, as the MN roams through the service provider's network, each time a hand-off occurs, the MN will continue to use the same IP address stored in the MBR on the HA.

Note that unlike Mobile IP-capable MNs that can perform multiple sessions over a single session link, Proxy Mobile IP allows only a single session over the session link. In addition, simultaneous Mobile and Simple IP sessions will not be supported for an MN by the FA that is currently facilitating a Proxy Mobile IP session for the MN.

How Proxy Mobile IP Works in 3GPP2 Network

This section contains call flows displaying successful Proxy Mobile IP session setup scenarios. There are multiple scenarios that are dependant on how the MN receives an IP address. The following scenarios are described:

- **Scenario 1:** The AAA server that authenticates the MN at the PDSN allocates an IP address to the MN. Note that the PDSN does not allocate an address from its IP pools.
- **Scenario 2:** The HA assigns an IP address to the MN from one of its locally configured dynamic pools.

Scenario 1: AAA server and PDSN/FA Allocate IP Address

The following figure and table display and describe a call flow in which the MN receives its IP address from the AAA server and PDSN/FA.

Figure 58. AAA/PDSN Assigned IP Address Proxy Mobile IP Call Flow

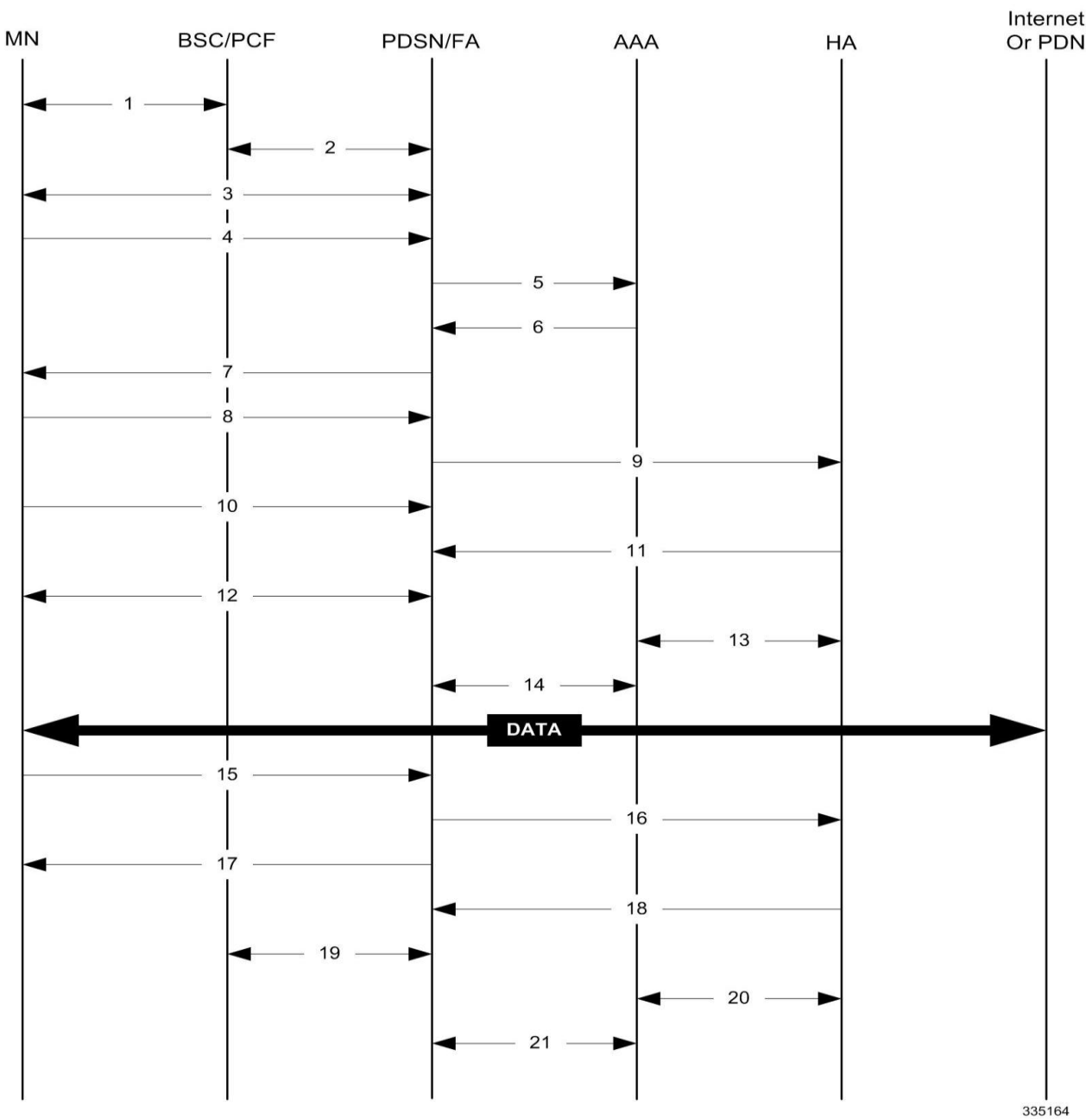


Table 52. AAA/PDSN Assigned IP Address Proxy Mobile IP Call Flow Description

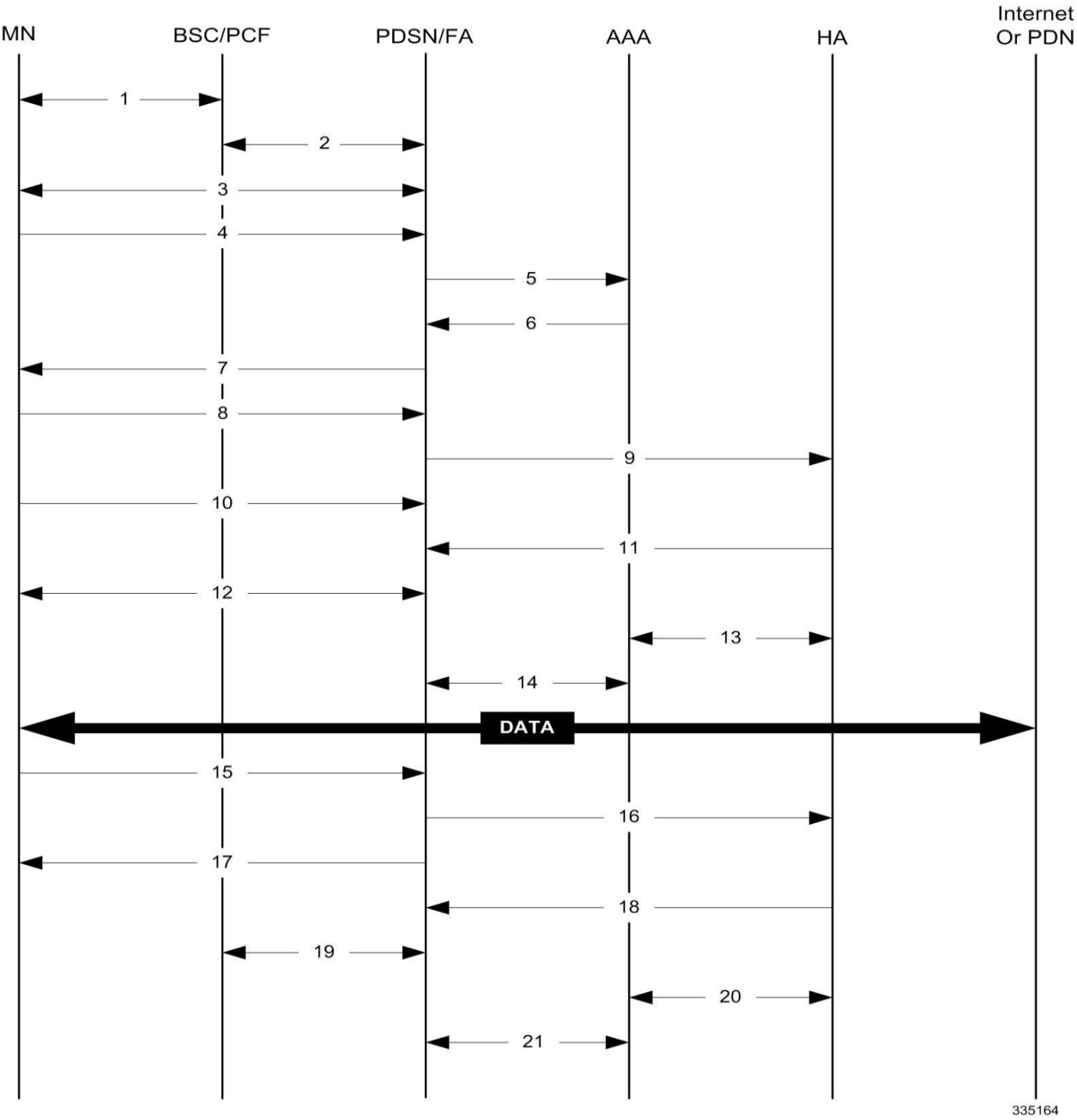
Step	Description
1	Mobile Node (MN) secures a traffic channel over the airlink with the RAN through the BSC/PCF.
2	The PCF and PDSN/FA establish the R-P interface for the session.
3	The PDSN/FA and MN negotiate Link Control Protocol (LCP).

Step	Description
4	Upon successful LCP negotiation, the MN sends a PPP Authentication Request message to the PDSN/FA.
5	The PDSN/FA sends an Access Request message to the RADIUS AAA server.
6	The RADIUS AAA server successfully authenticates the subscriber and returns an Access Accept message to the PDSN/FA. The Accept message may contain various attributes to be assigned to the MN including the MN's Home Address (IP address) and the IP address of the HA to use.
7	The PDSN/FA sends a PPP Authentication Response message to the MN.
8	The MN sends an Internet Protocol Control Protocol (IPCP) Configuration Request message to the PDSN/FA with an MN address of 0.0.0.0.
9	The PDSN/FA forwards a Proxy Mobile IP Registration Request message to the HA. The message includes fields such as the MN's home address, the IP address of the FA (the care-of-address), and the FA-HA extension (security parameter index (SPI)).
10	While the FA is communicating with the HA, the MN may send additional IPCP Configuration Request messages.
11	The HA responds with a Proxy Mobile IP Registration Response after validating the home address against its pool. The HA also creates a mobile binding record (MBR) for the subscriber session.
12	The MN and the PDSN/FA negotiate IPCP. The result is that the MN is assigned the home address originally specified by the AAA server.
13	While the MN and PDSN/FA are negotiating IPCP, the HA and AAA server initiate accounting.
14	Upon completion of the IPCP negotiation, the PDSN/FA and AAA server initiate accounting fully establishing the session allowing the MN to send/receive data to/from the PDN.
15	Upon completion of the session, the MN sends an LCP Terminate Request message to the PDSN to end the PPP session.
16	The PDSN/FA sends a Proxy Mobile IP De-registration Request message to the HA.
17	The PDSN/FA send an LCP Terminate Acknowledge message to the MN ending the PPP session.
18	The HA sends a Proxy Mobile IP De-Registration Response message to the FA terminating the Pi interface
19	The PDSN/FA and the PCF terminate the R-P session.
20	The HA and the AAA server stop accounting for the session.
21	The PDSN and the AAA server stop accounting for the session.

Scenario 2: HA Allocates IP Address

The following figure and table display and describe a call flow in which the MN receives its IP address from the HA.

Figure 59. HA Assigned IP Address Proxy Mobile IP Call Flow



335164

Table 53. HA Assigned IP Address Proxy Mobile IP Call Flow Description

Step	Description
1	Mobile Node (MN) secures a traffic channel over the airlink with the RAN through the BSC/PCF.
2	The PCF and PDSN/FA establish the R-P interface for the session.

Step	Description
3	The PDSN/FA and MN negotiate Link Control Protocol (LCP).
4	Upon successful LCP negotiation, the MN sends a PPP Authentication Request message to the PDSN/FA.
5	The PDSN/FA sends an Access Request message to the RADIUS AAA server.
6	The RADIUS AAA server successfully authenticates the subscriber and returns an Access Accept message to the PDSN/FA. The Accept message may contain various attributes to be assigned to the MN including the IP address of the HA to use.
7	The PDSN/FA sends a PPP Authentication Response message to the MN.
8	The MN sends an Internet Protocol Control Protocol (IPCP) Configuration Request message to the PDSN/FA with an MN address of 0.0.0.0.
9	The PDSN/FA forwards a Proxy Mobile IP Registration Request message to the HA. The message includes fields such as a Home Address indicator of 0.0.0.0, the IP address of the FA (the care-of-address), the IP address of the FA (the care-of-address), and the FA-HA extension (security parameter index (SPI)).
10	While the FA is communicating with the HA, the MN may send additional IPCP Configuration Request messages.
11	The HA responds with a Proxy Mobile IP Registration Response. The response includes an IP address from one of its locally configured pools to assign to the MN (its Home Address). The HA also creates a mobile binding record (MBR) for the subscriber session.
12	The MN and the PDSN/FA negotiate IPCP. The result is that the MN is assigned the home address originally specified by the AAA server.
13	While the MN and PDSN/FA are negotiating IPCP, the HA and AAA server initiate accounting.
14	Upon completion of the IPCP negotiation, the PDSN/FA and AAA server initiate accounting fully establishing the session allowing the MN to send/receive data to/from the PDN.
15	Upon completion of the session, the MN sends an LCP Terminate Request message to the PDSN to end the PPP session.
16	The PDSN/FA sends a Proxy Mobile IP De-registration Request message to the HA.
17	The PDSN/FA send an LCP Terminate Acknowledge message to the MN ending the PPP session.
18	The HA sends a Proxy Mobile IP De-Registration Response message to the FA terminating the Pi interface
19	The PDSN/FA and the PCF terminate the R-P session.
20	The HA and the AAA server stop accounting for the session.
21	The PDSN and the AAA server stop accounting for the session.

How Proxy Mobile IP Works in 3GPP Network

This section contains call flows displaying successful Proxy Mobile IP session setup scenarios in 3GPP network.

The following figure and the text that follows describe a sample successful Proxy Mobile IP session setup call flow in 3GPP service.

Figure 60. Proxy Mobile IP Call Flow in 3GPP

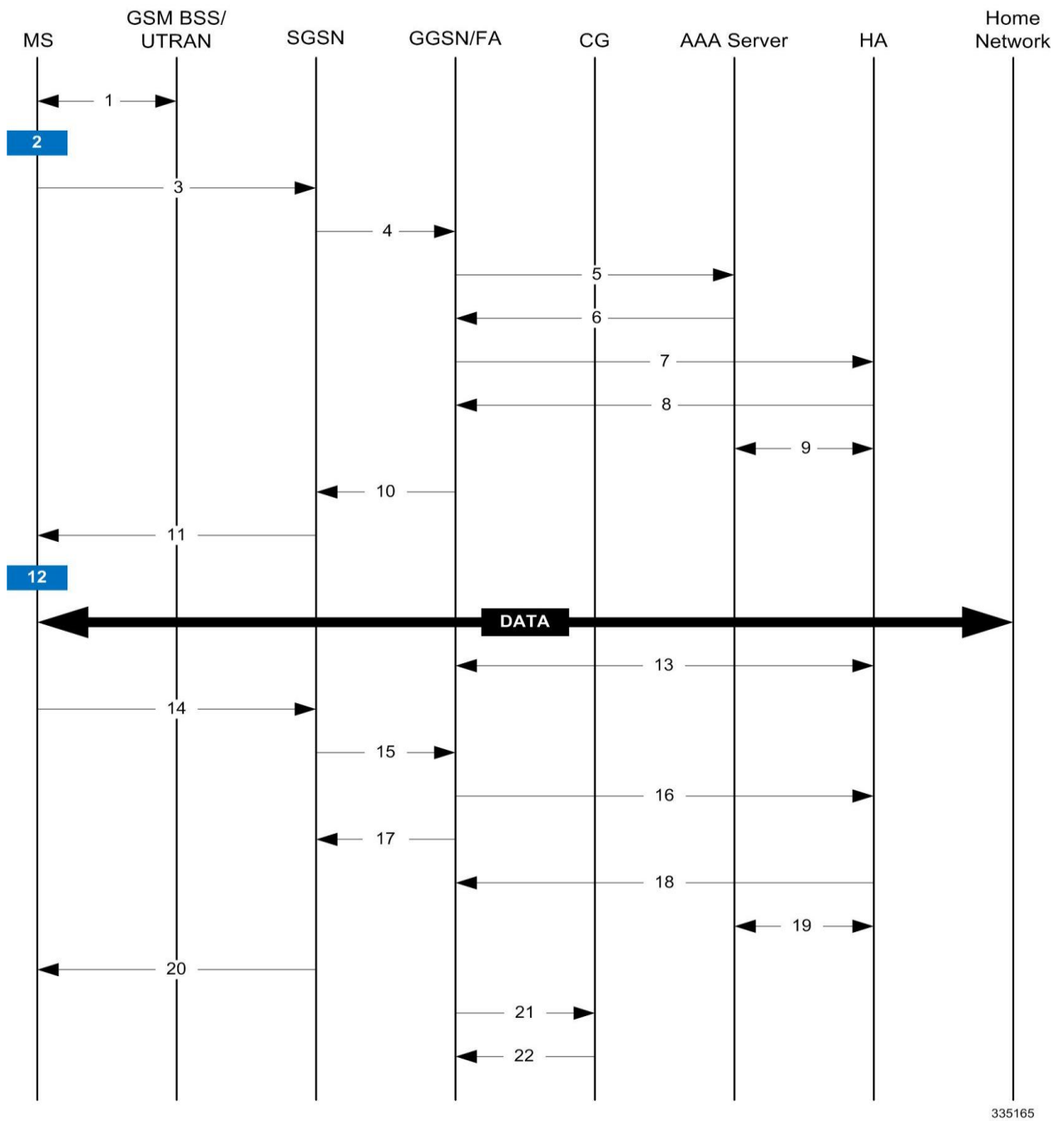


Table 54. Proxy Mobile IP Call Flow in 3GPP Description

Step	Description
1	The mobile station (MS) goes through the process of attaching itself to the GPRS/UMTS network.

Step	Description
2	<p>The terminal equipment (TE) aspect of the MS sends AT commands to the mobile terminal (MT) aspect of the MS to place it into PPP mode.</p> <p>The Link Control Protocol (LCP) is then used to configure the Maximum-Receive Unit size and the authentication protocol (Challenge-Handshake Authentication Protocol (CHAP), Password Authentication Protocol (PAP), or none). If CHAP or PAP is used, the TE will authenticate itself to the MT, which, in turn, stores the authentication information.</p> <p>Upon successful authentication, the TE sends an Internet Protocol Control Protocol (IPCP) Configure-Request message to the MT. The message will either contain a static IP address to use or request that one be dynamically assigned.</p>
3	<p>The MS sends an Activate PDP Context Request message that is received by an SGSN. The message contains information about the subscriber such as the Network layer Service Access Point Identifier (NSAPI), PDP Type, PDP Address, Access Point Name (APN), quality of service (QoS) requested, and PDP configuration options.</p>
4	<p>The SGSN authenticates the request message and sends a Create PDP Context Request message to a GGSN using the GPRS Tunneling Protocol (GTPC, “C” indicates the control signalling aspect of the protocol). The recipient GGSN is selected based on either the request of the MS or is automatically selected by the SGSN. The message consists of various information elements including: PDP Type, PDP Address, APN, charging characteristics, and tunnel endpoint identifier (TEID, if the PDP Address was static).</p>
5	<p>The GGSN determines if it can facilitate the session (in terms of memory or CPU resources, configuration, etc.) and creates a new entry in its PDP context list and provides a Charging ID for the session.</p> <p>From the APN specified in the message, the GGSN determines whether or not the subscriber is to be authenticated, if Proxy Mobile IP is to be supported for the subscriber, and if so, the IP address of the HA to contact.</p> <p>Note that Proxy Mobile IP support can also be determined by attributes in the user’s profile. Attributes in the user’s profile supersede APN settings.</p> <p>If authentication is required, the GGSN attempts to authenticate the subscriber locally against profiles stored in memory or send a RADIUS Access-Request message to a AAA server.</p>
6	<p>If the GGSN authenticated the subscriber to a AAA server, the AAA server responds with a RADIUS Access-Accept message indicating successful authentication and any attributes for handling the subscriber PDP context.</p>
7	<p>If Proxy Mobile IP support was either enabled in the APN or in the subscriber’s profile, the GGSN/FA forwards a Proxy Mobile IP Registration Request message to the specified HA. The message includes such things as the MS’s home address, the IP address of the FA (the care-of-address), and the FA-HA extension (security parameter index (SPI)).</p>
8	<p>The HA responds with a Proxy Mobile IP Registration Response. The response includes an IP address from one of its locally configured pools to assign to the MS (its Home Address). The HA also creates a mobile binding record (MBR) for the subscriber session.</p>
9	<p>The HA sends an RADIUS Accounting Start request to the AAA server which the AAA server responds to.</p>
10	<p>The GGSN replies with an affirmative Create PDP Context Response using GTPC. The response will contain information elements such as the PDP Address representing either the static address requested by the MS or the address assigned by the GGSN, the TEID used to reference PDP Address, and PDP configuration options specified by the GGSN.</p>
11	<p>The SGSN returns an Activate PDP Context Accept message to the MS. The message includes response to the configuration parameters sent in the initial request.</p>
12	<p>The MT, will respond to the TE’s IPCP Config-request with an IPCP Config-Ack message.</p> <p>The MS can now send and receive data to or from the PDN until the session is closed or times out. Note that for Mobile IP, only one PDP context is supported for the MS.</p>
13	<p>The FA periodically sends Proxy Mobile IP Registration Request Renewal messages to the HA. The HA sends responses for each request.</p>
14	<p>The MS can terminate the data session at any time. To terminate the session, the MS sends a Deactivate PDP Context Request message that is received by the SGSN.</p>

Step	Description
15	The SGSN sends a Delete PDP Context Request message to the GGSN facilitating the data session. The message includes the information elements necessary to identify the PDP context (i.e., TEID, and NSAPI).
16	The GGSN removes the PDP context from memory and the FA sends a Proxy Mobile IP Deregistration Request message to the HA.
17	The GGSN returns a Delete PDP Context Response message to the SGSN.
18	The HA replies to the FA with a Proxy Mobile IP Deregistration Request Response.
19	The HA sends an RADIUS Accounting Stop request to the AAA server which the AAA server responds to.
20	The SGSN returns a Deactivate PDP Context Accept message to the MS.
21	The GGSN delivers the GGSN Charging Detail Records (G-CDRs) to a charging gateway (CG) using GTP Prime (GTPP). Note that, though not shown in this example, the GGSN could optionally be configured to send partial CDRs while the PDP context is active.
22	For each accounting message received from the GGSN, the CG responds with an acknowledgement.

How Proxy Mobile IP Works in WiMAX Network

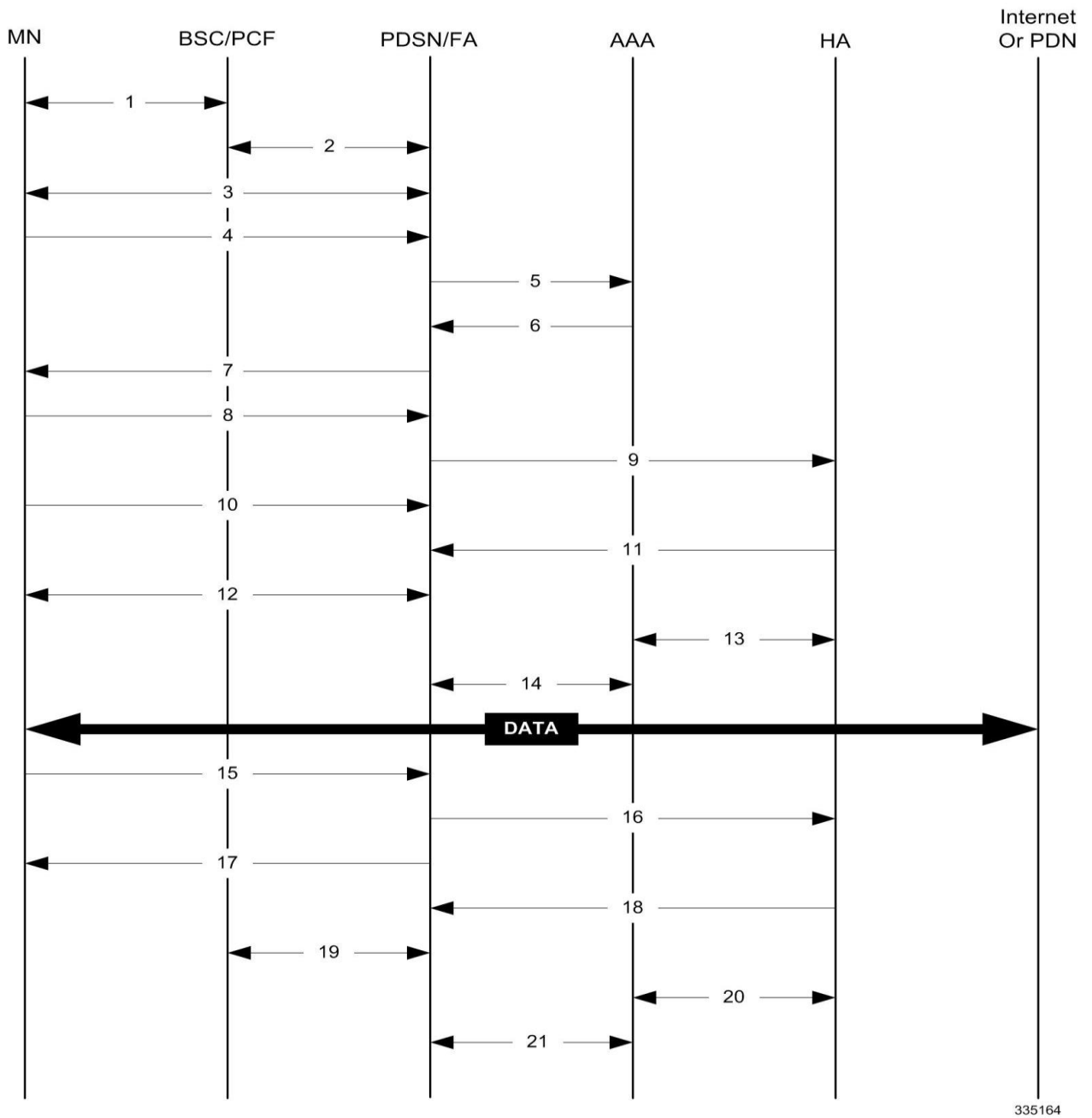
This section contains call flows displaying successful Proxy Mobile IP session setup scenarios. There are multiple scenarios that are dependant on how the MN receives an IP address. The following scenarios are described:

- **Scenario 1:** The AAA server that authenticates the MN at the ASN GW allocates an IP address to the MN. Note that the ASN GW does not allocate an address from its IP pools.
- **Scenario 2:** The HA assigns an IP address to the MN from one of its locally configured dynamic pools.

Scenario 1: AAA server and ASN GW/FA Allocate IP Address

The following figure and table display and describe a call flow in which the MN receives its IP address from the AAA server and ASN GW/FA.

Figure 61. AAA/ASN GW Assigned IP Address Proxy Mobile IP Call Flow



335164

Table 55. AAA/ASN GW Assigned IP Address Proxy Mobile IP Call Flow Description

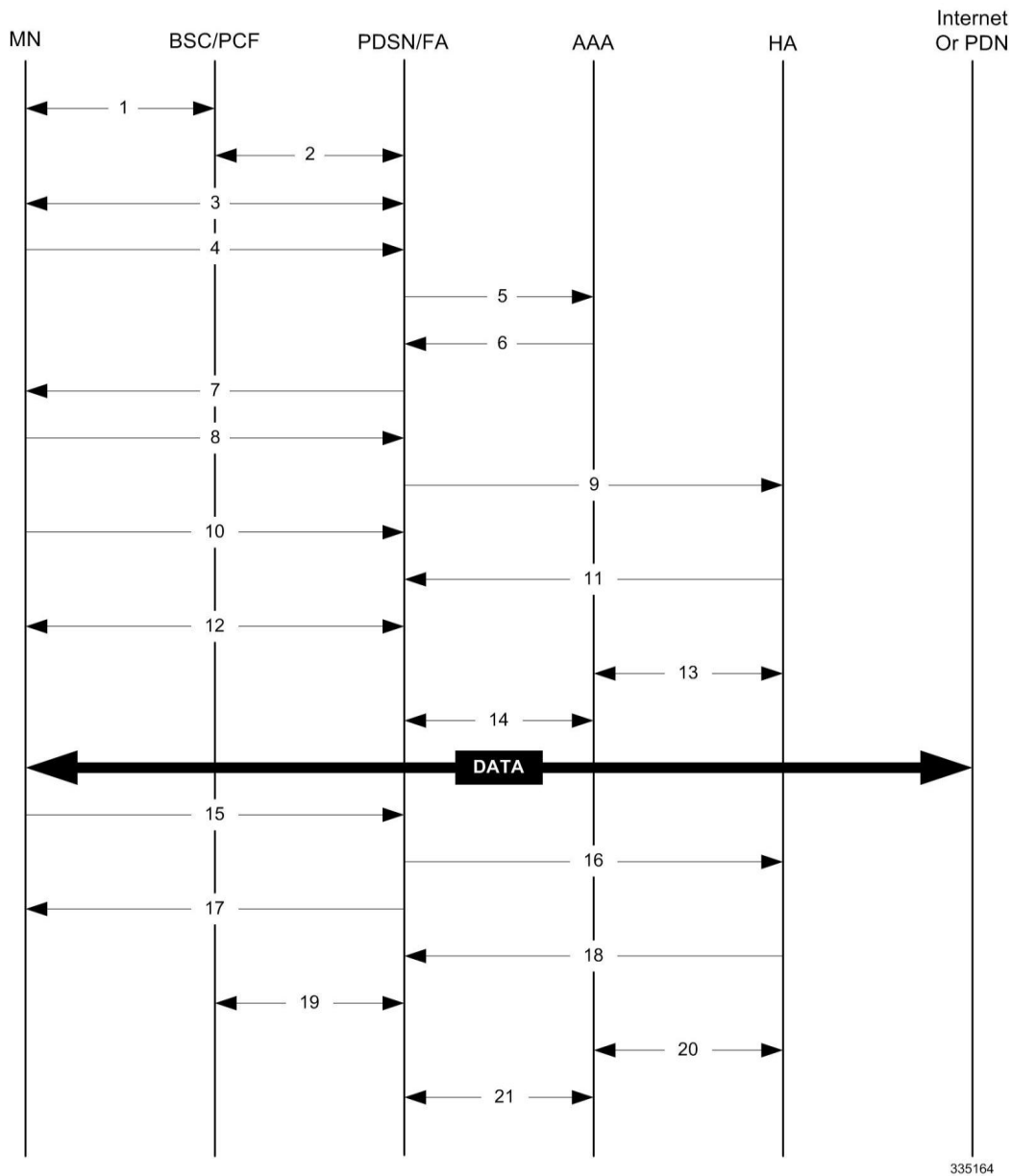
Step	Description
1	Mobile Node (MN) secures a traffic channel over the airlink with the BS.
2	The BS and ASN GW/FA establish the R6 interface for the session.

Step	Description
3	The ASN GW/FA and MN negotiate Link Control Protocol (LCP).
4	Upon successful LCP negotiation, the MN sends a PPP Authentication Request message to the ASN GW/FA.
5	The ASN GW/FA sends an Access Request message to the RADIUS AAA server.
6	The RADIUS AAA server successfully authenticates the subscriber and returns an Access Accept message to the ASN GW/FA. The Accept message may contain various attributes to be assigned to the MN including the MN's Home Address (IP address) and the IP address of the HA to use.
7	The ASN GW/FA sends a EAP Authentication Response message to the MN.
8	The MN sends an Internet Protocol Control Protocol (IPCP) Configuration Request message to the ASN GW/FA with an MN address of 0.0.0.0.
9	The ASN GW/FA forwards a Proxy Mobile IP Registration Request message to the HA. The message includes fields such as the MN's home address, the IP address of the FA (the care-of-address), and the FA-HA extension (security parameter index (SPI)).
10	While the FA is communicating with the HA, the MN may send additional IPCP Configuration Request messages.
11	The HA responds with a Proxy Mobile IP Registration Response after validating the home address against its pool. The HA also creates a mobile binding record (MBR) for the subscriber session.
12	The MN and the ASN GW/FA negotiate IPCP. The result is that the MN is assigned the home address originally specified by the AAA server.
13	While the MN and ASN GW/FA are negotiating IPCP, the HA and AAA server initiate accounting.
14	Upon completion of the IPCP negotiation, the ASN GW/FA and AAA server initiate accounting fully establishing the session allowing the MN to send/receive data to/from the PDN.
15	Upon completion of the session, the MN sends an LCP Terminate Request message to the ASN GW to end the subscriber session.
16	The PDSN/FA sends a Proxy Mobile IP De-registration Request message to the HA.
17	The ASN GW/FA send an LCP Terminate Acknowledge message to the MN ending the subscriber session.
18	The HA sends a Proxy Mobile IP De-Registration Response message to the FA terminating the R3 interface
19	The ASN GW/FA and the BS terminate the R6 session.
20	The HA and the AAA server stop accounting for the session.
21	The ASN GW and the AAA server stop accounting for the session.

Scenario 2: HA Allocates IP Address

The following figure and table display and describe a call flow in which the MN receives its IP address from the HA.

Figure 62. HA Assigned IP Address Proxy Mobile IP Call Flow



335164

Table 56. HA Assigned IP Address Proxy Mobile IP Call Flow Description

Step	Description
1	Mobile Node (MN) secures a traffic channel over the airlink with the BS.
2	The BS and ASN GW/FA establish the R6 interface for the session.
3	The ASN GW/FA and MN negotiate Link Control Protocol (LCP).
4	Upon successful LCP negotiation, the MN sends an EAP Authentication Request message to the ASN GW/FA.

Step	Description
5	The ASN GW/FA sends an Access Request message to the RADIUS AAA server.
6	The RADIUS AAA server successfully authenticates the subscriber and returns an Access Accept message to the ASN GW/FA. The Accept message may contain various attributes to be assigned to the MN including the IP address of the HA to use.
7	The ASN GW/FA sends an EAP Authentication Response message to the MN.
8	The MN sends an Internet Protocol Control Protocol (IPCP) Configuration Request message to the ASN GW/FA with an MN address of 0.0.0.0.
9	The ASN GW/FA forwards a Proxy Mobile IP Registration Request message to the HA. The message includes fields such as a Home Address indicator of 0.0.0.0, the IP address of the FA (the care-of-address), the IP address of the FA (the care-of-address), and the FA-HA extension (security parameter index (SPI)).
10	While the FA is communicating with the HA, the MN may send additional IPCP Configuration Request messages.
11	The HA responds with a Proxy Mobile IP Registration Response. The response includes an IP address from one of its locally configured pools to assign to the MN (its Home Address). The HA also creates a mobile binding record (MBR) for the subscriber session.
12	The MN and the ASN GW/FA negotiate IPCP. The result is that the MN is assigned the home address originally specified by the AAA server.
13	While the MN and ASN GW/FA are negotiating IPCP, the HA and AAA server initiate accounting.
14	Upon completion of the IPCP negotiation, the ASN GW/FA and AAA server initiate accounting fully establishing the session allowing the MN to send/receive data to/from the PDN.
15	Upon completion of the session, the MN sends an LCP Terminate Request message to the ASN GW to end the subscriber session.
16	The ASN GW/FA sends a Proxy Mobile IP De-registration Request message to the HA.
17	The ASN GW/FA send an LCP Terminate Acknowledge message to the MN ending the PPP session.
18	The HA sends a Proxy Mobile IP De-Registration Response message to the FA terminating the R3 interface
19	The ASN GW/FA and the BS terminate the R6 session.
20	The HA and the AAA server stop accounting for the session.
21	The ASN GW and the AAA server stop accounting for the session.

How Proxy Mobile IP Works in a WiFi Network with Multiple Authentication

Proxy-Mobile IP was developed as a result of networks of Mobile Subscribers (MS) that are not capable of Mobile IP operation. In this scenario a PDIF acts a mobile IP client and thus implements Proxy-MIP support.

Although not required or necessary in a Proxy-MIP network, this implementation uses a technique called Multiple Authentication. In Multi-Auth arrangements, the device is authenticated first using HSS servers. Once the device is authenticated, then the subscriber is authenticated over a RADIUS interface to AAA servers. This supports existing EV-DO servers in the network.

The MS first tries to establish an IKEv2 session with the PDIF. The MS uses the EAP-AKA authentication method for the initial device authentication using Diameter over SCTP over IPv6 to communicate with HSS servers. After the initial Diameter EAP authentication, the MS continues with EAP MD5/GTC authentication.

After successful device authentication, PDIF then uses RADIUS to communicate with AAA servers for the subscriber authentication. It is assumed that RADIUS AAA servers do not use EAP methods and hence RADIUS messages do not contain any EAP attributes.

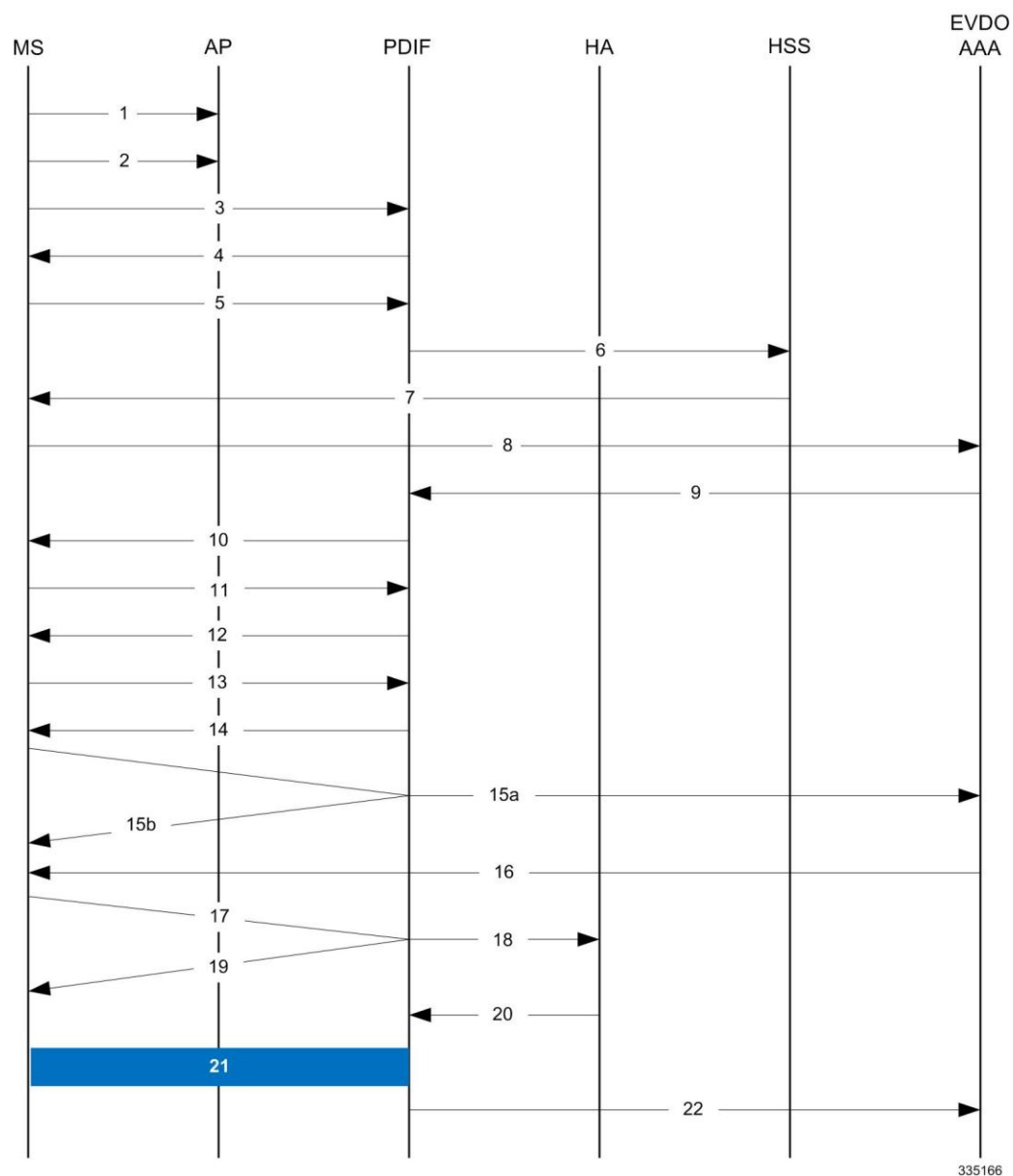
Assuming a successful RADIUS authentication, PDIF then sets up the IPSec Child SA tunnel using a Tunnel Inner Address (TIA) for passing control traffic only. PDIF receives the MS address from the Home Agent, and passes it on to the MS through the final AUTH response in the IKEv2 exchange.

When IPSec negotiation finishes, the PDIF assigns a home address to the MS and establishes a CHILD SA to pass data. The initial TIA tunnel is torn down and the IP address returned to the address pool. The PDIF then generates a RADIUS accounting START message.

When the session is disconnected, the PDIF generates a RADIUS accounting STOP message.

The following figures describe a Proxy-MIP session setup using CHAP authentication (EAP-MD5), but also addresses a PAP authentication setup using EAP-GTC when EAP-MD5 is not supported by either PDIF or MS.

Figure 63. Proxy-MIP Call Setup using CHAP Authentication



335166

Table 57. Proxy-MIP Call Setup using CHAP Authentication

Step	Description
1	On connecting to WiFi network, MS first send DNS query to get PDIF IP address
2	MS receives PDIF address from DNS
3	MS sets up IKEv2/IPSec tunnel by sending IKE_SA_INIT Request to PDIF. MS includes SA, KE, Ni, NAT-DETECTION Notify payloads in the IKEv2 exchange.

Step	Description
4	PDIF processes the IKE_SA_INIT Request for the appropriate PDIF service (bound by the destination IP address in the IKEv2 INIT request). PDIF responds with IKE_SA_INIT Response with SA, KE, Nr payloads and NAT-Detection Notify payloads. If multiple-authentication support is configured to be enabled in the PDIF service, PDIF will include MULTIPLE_AUTH_SUPPORTED Notify payload in the IKE_SA_INIT Response. PDIF will start the IKEv2 setup timer after sending the IKE_SA_INIT Response.
5	On receiving successful IKE_SA_INIT Response from PDIF, MS sends IKE_AUTH Request for the first EAP-AKA authentication. If the MS is capable of doing multiple-authentication, it will include MULTI_AUTH_SUPPORTED Notify payload in the IKE_AUTH Request. MS also includes IDi payload which contains the NAI, SA, TSr, CP (requesting IP address and DNS address) payloads. MS will not include AUTH payload to indicate that it will use EAP methods.
6	On receiving IKE_AUTH Request from MS, PDIF sends DER message to Diameter AAA server. AAA servers are selected based on domain profile, default subscriber template or default domain configurations. PDIF includes Multiple-Auth-Support AVP, EAP-Payload AVP with EAP-Response/Identity in the DER. Exact details are explained in the Diameter message sections. PDIF starts the session setup timer on receiving IKE_AUTH Request from MS.
7	PDIF receives DEA with Result-Code AVP specifying to continue EAP authentication. PDIF takes EAP-Payload AVP contents and sends IKE_AUTH Response back to MS in the EAP payload. PDIF allows IDr and CERT configurations in the PDIF service and optionally includes IDr and CERT payloads (depending upon the configuration). PDIF optionally includes AUTH payload in IKE_AUTH Response if PDIF service is configured to do so.
8	MS receives the IKE_AUTH Response from PDIF. MS processes the exchange and sends a new IKE_AUTH Request with EAP payload. PDIF receives the new IKE_AUTH Request from MS and sends DER to AAA server. This DER message contains the EAP-Payload AVP with EAP-AKA challenge response and challenge received from MS.
9	The AAA server sends the DEA back to the PDIF with Result-Code AVP as “success.” The EAP-Payload AVP message also contains the EAP result code with “success.” The DEA also contains the IMSI for the user, which is included in the Callback-Id AVP. PDIF uses this IMSI for all subsequent session management functions such as duplicate session detection etc. PDIF also receives the MSK from AAA, which is used for further key computation.
10	PDIF sends the IKE_AUTH Response back to MS with the EAP payload.
11	MS sends the final IKE_AUTH Request for the first authentication with the AUTH payload computed from the keys. If the MS plans to do the second authentication, it will include ANOTHER_AUTH_FOLLOWS Notify payload also.
12	PDIF processes the AUTH request and responds with the IKE_AUTH Response with the AUTH payload computed from the MSK. PDIF does not assign any IP address for the MS pending second authentication. Nor will the PDIF include any configuration payloads. a. If PDIF service does not support Multiple-Authentication and ANOTHER_AUTH_FOLLOWS Notify payload is received, then PDIF sends IKE_AUTH Response with appropriate error and terminate the IKEv2 session by sending INFORMATIONAL (Delete) Request. b. If ANOTHER_AUTH_FOLLOWS Notify payload is not present in the IKE_AUTH Request, PDIF allocates the IP address from the locally configured pools. However, if proxy-mip-required is enabled, then PDIF initiates Proxy-MIP setup to HA by sending P-MIP RRQ. When PDIF receives the Proxy-MIP RRP, it takes the Home Address (and DNS addresses if any) and sends the IKE_AUTH Response back to MS by including CP payload with Home Address and DNS addresses. In either case, IKEv2 setup will finish at this stage and IPSec tunnel gets established with a Tunnel Inner Address (TIA).
13	MS does the second authentication by sending the IKE_AUTH Request with IDi payload to include the NAI. This NAI may be completely different from the NAI used in the first authentication.

Step	Description
14	<p>On receiving the second authentication IKE_AUTH Request, PDIF checks the configured second authentication methods. The second authentication may be either EAP-MD5 (default) or EAP-GTC. The EAP methods may be either EAP-Passthru or EAP-Terminated.</p> <p>a. If the configured method is EAP-MD5, PDIF sends the IKE_AUTH Response with EAP payload including challenge.</p> <p>b. If the configured method is EAP-GTC, PDIF sends the IKE_AUTH Response with EAP-GTC.</p> <p>c. MS processes the IKE_AUTH Response:</p> <ul style="list-style-type: none"> • If the MS supports EAP-MD5, and the received method is EAP-MD5, then the MS will take the challenge, compute the response and send IKE_AUTH Request with EAP payload including Challenge and Response. • If the MS does not support EAP-MD5, but EAP-GTC, and the received method is EAP-MD5, the MS sends legacy-Nak with EAP-GTC.
15(a)	<p>PDIF receives the new IKE_AUTH Request from MS.</p> <p>If the original method was EAP-MD5 and MD5 challenge and response is received, PDIF sends RADIUS Access Request with corresponding attributes (Challenge, Challenge Response, NAI, IMSI etc.).</p>
15(b)	If the original method was EAP-MD5 and legacy-Nak was received with GTC, the PDIF sends IKE_AUTH Response with EAP-GTC.
16	PDIF receives Access Accept from RADIUS and sends IKE_AUTH Response with EAP success.
17	PDIF receives the final IKE_AUTH Request with AUTH payload.
18	PDIF checks the validity of the AUTH payload and initiates Proxy-MIP setup request to the Home Agent if proxy-mip-required is enabled. The HA address may be received from the RADIUS server in the Access Accept (Step 16) or may be locally configured. PDIF may also remember the HA address from the first authentication received in the final DEA message.
19	If proxy-mip-required is disabled, PDIF assigns the IP address from the local pool.
20	PDIF received proxy-MIP RRP and gets the IP address and DNS addresses.
21	PDIF sets up the IPSec tunnel with the home address. On receiving the IKE_AUTH Response MS also sets up the IPSec tunnel using the received IP address. PDIF sends the IKE_AUTH Response back to MS by including the CP payload with the IP address and optionally the DNS addresses. This completes the setup.
22	PDIF sends a RADIUS Accounting start message.



Important: For Proxy-MIP call setup using PAP, the first 14 steps are the same as for CHAP authentication. However, here they deviate because the MS does not support EAP-MD5 authentication, but EAP-GTC. In response to the EAP-MD5 challenge, the MS instead responds with legacy-Nak with EAP-GTC. The diagram below picks up at this point.

Figure 64. Proxy-MIP Call Setup using PAP Authentication

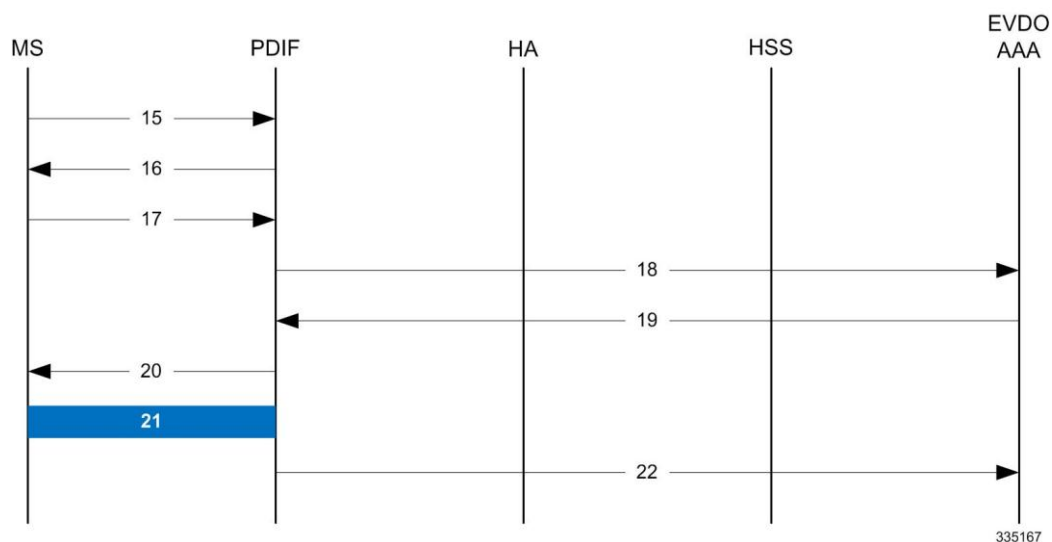


Table 58. Proxy-MIP Call Setup using PAP Authentication

Step	Description
15	MS is not capable of CHAP authentication but PAP authentication, and the MS returns the EAP payload to indicate that it needs EAP-GTC authentication.
16	PDIF then initiates EAP-GTC procedure, and requests a password from MS.
17	MS includes an authentication password in the EAP payload to PDIF.
18	Upon receipt of the password, PDIF sends a RADIUS Access Request which includes NAI in the User-Name attribute and PAP-password.
19	Upon successful authentication, the AAA server returns a RADIUS Access Accept message, which may include Framed-IP-Address attribute.
20	The attribute content in the Access Accept message is encoded as EAP payload with EAP success when PDIF sends the IKE_AUTH Response to the MS.
21	The MS and PDIF now have a secure IPSec tunnel for communication.
22	Pdif sends an Accounting START message.

Configuring Proxy Mobile-IP Support

Support for Proxy Mobile-IP requires that the following configurations be made:



Important: Not all commands and keywords/variables may be supported. This depends on the platform type and the installed license(s).

- **FA service(s):** Proxy Mobile IP must be enabled, operation parameters must be configured, and FA-HA security associations must be specified.
- **HA service(s):** FA-HA security associations must be specified.
- **Subscriber profile(s):** Attributes must be configured to allow the subscriber(s) to use Proxy Mobile IP. These attributes can be configured in subscriber profiles stored locally on the system or remotely on a RADIUS AAA server.
- **APN template(s):** Proxy Mobile IP can be supported for every subscriber IP PDP context facilitated by a specific APN template based on the configuration of the APN.



Important: These instructions assume that the system was previously configured to support subscriber data sessions as a core network service and/or an HA according to the instructions described in the respective product administration guide.

Configuring FA Services

Use this example to configure an FA service to support Proxy Mobile IP:

configure

```

context <context_name>

    fa-service <fa_service_name>

        proxy-mip allow

            proxy-mip max-retransmissions <integer>

            proxy-mip retransmission-timeout <seconds>

            proxy-mip renew-percent-time percentage

            fa-ha-spi remote-address { ha_ip_address | ip_addr_mask_combo } spi-number
            number { encrypted secret enc_secret | secret secret } [ description string ] [ hash-
            algorithm { hmac-md5 | md5 | rfc2002-md5 } | replay-protection { timestamp | nonce } |
            timestamp-tolerance tolerance ]

        authentication mn-ha allow-noauth

    end

```

Notes:

- The **proxy-mip max-retransmissions** command configures the maximum number re-try attempts that the FA service is allowed to make when sending Proxy Mobile IP Registration Requests to the HA.
- **proxy-mip retransmission-timeout** configures the maximum amount of time allowed by the FA for a response from the HA before re-sending a Proxy Mobile IP Registration Request message.
- **proxy-mip renew-percent-time** configures the amount of time that must pass prior to the FA sending a Proxy Mobile IP Registration Renewal Request.

Example

If the advertisement registration lifetime configured for the FA service is 900 seconds and the renew-time is configured to 50%, then the FA requests a lifetime of 900 seconds in the Proxy MIP registration request. If the HA grants a lifetime of **600** seconds, then the FA sends the Proxy Mobile IP Registration Renewal Request message after **300** seconds have passed.

- Use the **fa-ha-spi remote-address** command to modify configured FA-HA SPIs to support Proxy Mobile IP. Refer to the *Command Line Interface Reference* for the full command syntax.



Important: Note that FA-HA SPIs **must** be configured for the Proxy-MIP feature to work, while it is optional for regular MIP.

- Use the **authentication mn-ha allow-noauth** command to configure the FA service to allow communications from the HA without authenticating the HA.

Verify the FA Service Configuration

Use the following command to verify the configuration of the FA service:

```
show fa-service name <fa_service_name>
```

Notes:

- Repeat this example as needed to configure additional FA services to support Proxy-MIP.
- Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Proceed to the optional [Configuring Proxy MIP HA Failover](#) section to configure Proxy MIP HA Failover support or skip to the [Configuring HA Services](#) section to configure HA service support for Proxy Mobile IP.

Configuring Proxy MIP HA Failover

Use this example to configure Proxy Mobile IP HA Failover:



Important: This configuration in this section is optional.

When configured, Proxy MIP HA Failover provides a mechanism to use a specified alternate Home Agent for the subscriber session when the primary HA is not available. Use the following configuration example to configure the Proxy MIP HA Failover:

```
configure

context <context_name>

    fa-service <fa_service_name>

        proxy-mip ha-failover [ max-attempts <max_attempts> | num-attempts-
before-switching <num_attempts> | timeout <seconds> ]
```

Notes:

- Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.


Configuring HA Services

Use the following configuration example to configure HA services to support Proxy Mobile IP.

```
configure

context <context_name>

    ha-service <ha_service_name>
```

 **Important:** Note that FA-HA SPIs must be configured for the Proxy MIP feature to work while it is optional for regular MIP. Also note that the above syntax assumes that FA-HA SPIs were previously configured as part of the HA service as described in respective product Administration Guide. The **replay-protection** and **timestamp-tolerance** keywords should only be configured when supporting Proxy Mobile IP.

```
    fa-ha-spi remote-address <fa_ip_address> spi-number <number> { encrypted secret
<enc_secret> | secret <secret> } [ description <string> ] [ hash-algorithm { hmac-md5 |
md5 | rfc2002-md5 } ] replay-protection { timestamp | nonce } | timestamp-tolerance
<tolerance> ]

    authentication mn-ha allow-noauth

    authentication mn-aaa allow-noauth

end
```

Notes:

- Repeat this example as needed to configure additional HA services to support Proxy-MIP.
- Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

To verify the configuration of the HA service:


```
context <context_name>

    show ha-service name <ha_service_name>
```

Configuring Subscriber Profile RADIUS Attributes

In order for subscribers to use Proxy Mobile IP, attributes must be configured in their user profile or in an APN for 3GPP service. As mentioned previously, the subscriber profiles can be located either locally on the system or remotely on a RADIUS AAA server.


This section provides information on the RADIUS attributes that must be used and instructions for configuring locally stored profiles/APNs in support of Proxy Mobile IP.

 **Important:** Instructions for configuring RADIUS-based subscriber profiles are not provided in this document. Please refer to the documentation supplied with your server for further information.

RADIUS Attributes Required for Proxy Mobile IP

The following table describes the attributes that must be configured in profiles stored on RADIUS AAA servers in order for the subscriber to use Proxy Mobile IP.

Table 59. Required RADIUS Attributes for Proxy Mobile IP

Attribute	Description	Values
SN-Subscriber-Permission OR SN1-Subscriber-Permission	Indicates the services allowed to be delivered to the subscriber. For Proxy Mobile IP, this attribute must be set to Simple IP.	<ul style="list-style-type: none"> None (0) Simple IP (0x01) Mobile IP (0x02) Home Agent Terminated Mobile IP (0x04)
SN-Proxy-MIP OR SN1-Proxy-MIP	Specifies if the configured service will perform compulsory Proxy-MIP tunneling for a Simple-IP subscriber. This attribute must be enabled to support Proxy Mobile IP.	<ul style="list-style-type: none"> Disabled - do not perform compulsory Proxy-MIP (0) Enabled - perform compulsory Proxy-MIP (1)
SN-Simultaneous-SIP-MIP OR SN1-Simultaneous-SIP-MIP	Indicates whether or not a subscriber can simultaneously access both Simple IP and Mobile IP services. <div>  Important: Regardless of the configuration of this attribute, the FA facilitating the Proxy Mobile IP session will not allow simultaneous Simple IP and Mobile IP sessions for the MN. </div>	<ul style="list-style-type: none"> Disabled (0) Enabled (1)

Attribute	Description	Values
SN-PDSN-Handoff-Req-IP-Addr OR SN1-PDSN-Handoff-Req-IP-Addr	Specifies whether or not the system should reject and terminate the subscriber session when the proposed address in IPCP by the mobile does not match the existing address that was granted by the chassis during an Inter-chassis handoff. This can be used to disable the acceptance of 0.0.0.0 as the IP address proposed by the MN during the IPCP negotiation that occurs during an Inter-chassis handoff. This attribute is disabled (do not reject) by default.	<ul style="list-style-type: none"> Disabled - do not reject (0) Enabled - reject (1)
3GPP2-MIP-HA-Address	This attribute sent in an Access-Accept message specifies the IP Address of the HA. Multiple attributes can be sent in Access Accept. However, only the first two are considered for processing. The first one is the primary HA and the second one is the secondary (alternate) HA used for HA Failover.	IPv4 Address

Configuring Local Subscriber Profiles for Proxy-MIP on a PDSN

This section provides information and instructions for configuring local subscriber profiles on the system to support Proxy Mobile IP on a PDSN.

configure

```

context <context_name>

    subscriber name <subscriber_name>

    permission pdsn-simple-ip

    proxy-mip allow

    inter-pdsn-handoff require ip-address

    mobile-ip home-agent <ha_address>

    <optional> mobile-ip home-agent <ha_address> alternate

    ip context-name <context_name>

end

```

Verify that your settings for the subscriber(s) just configured are correct.

```
show subscribers configuration username <subscriber_name>
```

Notes:

- Configure the system to enforce the MN's use of its assigned IP address during IPCP negotiations resulting from inter-PDSN handoffs. Sessions re-negotiating IPCP will be rejected if they contain an address other than that which was granted by the PDSN (i.e. 0.0.0.0). This rule can be enabled by entering the **inter-pdsn-handoff require ip-address** command.
- Optional: If you have enabled the Proxy-MIP HA Failover feature, use the **mobile-ip home-agent ha_address alternate** command to specify the secondary, or alternate HA.

- Repeat this example as needed to configure additional FA services to support Proxy-MIP.
- Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Configuring Local Subscriber Profiles for Proxy-MIP on a PDIF

This section provides instructions for configuring local subscriber profiles on the system to support Proxy Mobile IP on a PDIF.

configure

```
context <context-name>

    subscriber name <subscriber_name>

    proxy-mip require
```

Note

subscriber_name is the name of the subscriber and can be from 1 to 127 alpha and/or numeric characters and is case sensitive.

Configuring Default Subscriber Parameters in Home Agent Context

It is very important that the subscriber default, configured in the same context as the HA service, has the name of the destination context configured. Use the configuration example below:

configure

```
context <context_name>

    ip context-name <context_name>

end
```

Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Configuring APN Parameters

This section provides instructions for configuring the APN templates to support Proxy Mobile IP for all IP PDP contexts they facilitate.



Important: This is an optional configuration. In addition, attributes returned from the subscriber's profile for non-transparent IP PDP contexts take precedence over the configuration of the APN.

These instructions assume that you are at the root prompt for the Exec mode:

```
[local]host_name#
```

Step 1 Enter the configuration mode by entering the following command:

```
configure
```

The following prompt appears:

```
[local]host_name(config)#
```

Step 2 Enter context configuration mode by entering the following command:

```
context <context_name>
```

context_name is the name of the system destination context designated for APN configuration. The name must be from 1 to 79 alpha and/or numeric characters and is case sensitive. The following prompt appears:

```
[<context_name>]host_name(config-ctx)#
```

Step 3 Enter the configuration mode for the desired APN by entering the following command:

```
apn <apn_name>
```

apn_name is the name of the APN that is being configured. The name must be from 1 to 62 alpha and/or numeric characters and is not case sensitive. It may also contain dots (.) and/or dashes (-). The following prompt appears:

```
[<context_name>]host_name(config-apn)#
```

Step 4 Enable proxy Mobile IP for the APN by entering the following command:

```
proxy-mip required
```

This command causes proxy Mobile IP to be supported for all IP PDP contexts facilitated by the APN.

Step 5 *Optional.* GGSN/FA MN-NAI extension can be skipped in MIP Registration Request by entering following command:

```
proxy-mip null-username static-homeaddr
```

This command will enable the accepting of MIP Registration Request without NAI extensions in this APN.

Step 6 Return to the root prompt by entering the following command:

```
end
```

The following prompt appears:

```
[local]host_name#
```

Step 7 Repeat *step 1* through *step 6* as needed to configure additional APNs.

Step 8 Verify that your APNs were configured properly by entering the following command:

```
show apn { all | name <apn_name> }
```

Keyword	Description
all	Displays configuration information for all configured APN.

Keyword	Description
name	Displays configuration information for the APN with the specified name. apn_name is the name of the APN.

The output is a detailed listing of configured APN parameter settings.

- Step 9** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Appendix S

Rejection/Redirection of HA Sessions on Network Failures

This chapter provides information on configuring an enhanced, or extended, service. The product Administration Guides provide examples and procedures for configuration of basic services on the system. It is recommended that you select the configuration example that best meets your service model, and configure the required elements for that model, as described in the respective product Administration Guide, before using the procedures in this chapter.

The following sections are included in this chapter:

- [Overview](#)
- [Configuring HA Session Redirection](#)
- [RADIUS Attributes](#)

Overview

This feature enables the HA service to either reject new calls or redirect them to another HA when a destination network connection failure is detected. When network connectivity is re-established, the HA service begins to accept calls again in the normal manner.

The way this is implemented in the system is as follows:

- A policy is configured in the HA service that tells the service what action to take when network connectivity is lost. New calls are either directed to one of up to 16 different IP addresses or all new calls are rejected until network connectivity is restored.
- In the destination context, a network reachability server is configured. This is a device on the destination network to which ping packets are periodically sent to determine if the network is reachable. As soon as a network reachability server is configured, pinging of the server commences whether or not the server name is bound to a subscriber or an IP pool.
- The name of the network reachability server configured in the destination context is bound to either a local subscriber profile or an IP pool. If the subscriber is authenticated by an AAA server, RADIUS attributes may specify the network reachability server for the subscriber. (If an IP pool has a network reachability server name bound to it, that takes precedence over both the RADIUS attributes and the local subscriber configuration.)

Configuring HA Session Redirection

This section provides instructions for configuring rejection or redirection of HA sessions on the event of a network failure. These instructions assume that there is a destination context, and HA service, an IP pool, and a subscriber already configured and that you are at the root prompt for the Exec mode:

```
[local]host_name#
```

Step 1 Enter the global configuration mode by entering the following command:

```
configure
```

The following prompt appears:

```
[local]host_name(config)#
```

Step 2 Enter context configuration mode by entering the following command:

```
context <context_name>
```

context_name is the name of the destination context where the HA service is configured. The name must be from 1 to 63 alpha and/or numeric characters and is case sensitive. The following prompt appears:

```
[<context_name>]host_name(config-ctx)#
```

Step 3 Enter the HA service configuration mode by entering the following command:

```
ha-service <ha_service_name>
```

ha_service_name is the name of the HA service. The name must be from 1 to 63 alpha and/or numeric characters and is case sensitive. The following prompt appears:

```
[<context_name>]host_name(config-ha-service)#
```

Step 4 Configure the action for the HA service to take when network connectivity is lost by entering the following command:

```
policy nw-reachability-fail { reject [ use-reject-code { admin-prohibited |  
insufficient-resources } ] | redirect <ip_addr1> [ weight <value> ] [ <ip_addr2>  
[ weight <value> ] ] ... [ <ip_addr16> [ weight <value> ] ] }
```

Keyword/Variable	Description
reject	Upon network reachability failure reject all new calls for this context.
use-reject-code { admin-prohibited insufficient-resources }	When rejecting calls send the specified reject code. If this keyword is not specified the admin-prohibited reject code is sent by default.

Keyword/Variable	Description
<pre> redirect <ip_addr1> [weight <value>] [<ip_addr2> [weight <value>]] ... [<ip_addr16> [weight <value>]] </pre>	<p>Upon network reachability failure redirect all calls to the specified IP address.</p> <p><ip_addr>: This must be an IPv4 address. Up to 16 IP addresses and optional weight values can be entered on one command line.</p> <p>weight <value>: When multiple addresses are specified, they are selected in a weighted round-robin scheme. If a weight is not specified, the entry is automatically assigned a weight of 1.</p> <p><value> must be an integer from 1 through 10.</p>

Step 5 Enter the following command to return to the context configuration mode:

```
exit
```

The following prompt appears:

```
[<context_name>]host_name(config-ctx)#
```

Step 6 Specify the network device on the destination network to which ping packets should be sent to test for network reachability, by entering the following command:

```

nw-reachability server <server_name> [ interval <seconds> ] [ local-addr
<ip_addr> ] [ num-retry <num> ] [ remote-addr <ip_addr> ] [ timeout < seconds> ]

```

Keyword/Variable	Description
<i>server_name</i>	A name for the network device that is sent ping packets to test for network reachability.
interval <seconds>	Default: 60 seconds Specifies the frequency in seconds for sending ping requests. <seconds> must be an integer from 1 through 3600.
local-addr <ip_addr>	Specifies the IP address to be used as the source address of the ping packets; If this is unspecified, an arbitrary IP address that is configured in the context is used. <ip_addr> must be an IP v4 address.
num-retry <num>	Default: 5 Specifies the number of retries before deciding that there is a network-failure. <num> must be an integer from 0 through 100.
remote-addr <ip_addr>	Specifies the IP address of a network element to use as the destination to send the ping packets for detecting network failure or reachability. <ip_addr> must be an IPv4 address.
timeout < seconds>	Default: 3 seconds Specifies how long to wait, in seconds, before retransmitting a ping request to the remote address. <seconds> must be an integer from 1 through 10.

- Step 7** Repeat *step 6* to configure additional network reachability servers.
- Step 8** To bind a network reachability server to an IP pool, continue with *step 9*. To bind a network reachability server to a local subscriber profile, skip to *step 11*.
- Step 9** To bind a network reachability server name to an IP pool, enter the following command:

```
ip pool <pool_name> nw-reachability server <server_name>
```

<code><pool_name></code>	The name of an existing IP pool in the current context.
<code>nw-reachability server <server_name></code>	Bind the name of a configured network reachability server to the IP pool and enable network reachability detection for the IP pool. This takes precedence over any network reachability server settings in a subscriber configuration or RADIUS attribute. <code><server_name></code> : The name of a network reachability server that has been defined in the current context. This is a string of from 1 through 16 characters.

- Step 10** Repeat *step 9* for additional IP pools in the current context then skip to *step 13*.

- Step 11** Enter the subscriber configuration mode by entering the following command:

```
subscriber { default | name <subs_name> }
```

Where **default** is the default subscriber for the current context and *subs_name* is the name of the subscriber profile that you want to configure for network reachability. The following prompt appears:

```
[<context_name>]host_name(config-subscriber)#
```

- Step 12** To bind a network reachability server name to the current subscriber in the current context, enter the following command:

```
nw-reachability server <server_name>
```

Where *server_name* is the name of a network reachability server that has been defined in the current context.

- Step 13** Return to the executive mode by entering the following command:

```
end
```

The following prompt appears:

```
[local]host_name#
```

- Step 14** Enter the executive mode for the destination context for which you configured network reachability by entering the following command:

```
context <context_name>
```

Where *context_name* is the name of the destination context for which you configured network reachability. The following prompt appears:

```
[context_name]host_name#
```

Step 15 Check the network reachability server configuration by entering the following command

```
show nw-reachability server all
```

The output of this command appears similar to the following:

```
Server remote-addr local-addr state
-----
nw-server1 192.168.100.20 192.168.1.10 Down

Total Network Reachability Servers: 1 Up: 0
```

Ensure that the remote and local addresses are correct. The state column indicates whether or not the server is reachable (Up) or unreachable (Down).

Step 16 Check the HA service policy by entering the following command:

```
show ha-service name <ha_service_name>
```

Where *<ha_service_name>* is the name of the HA service in the current context for which you configured a network reachability policy. The output of this command includes information about the network reachability policy that looks similar to the following:

```
NW-Reachability Policy: Reject (Reject code: Admin Prohibited)
```

Step 17 Check the network reachability server name bound to an IP pool by entering the following command:

```
show ip pool pool-name <pool_name>
```

Where *<pool_name>* is the name of the IP pool to which you bound a network reachability server name. The output of this command includes information about the network reachability server name that looks similar to the following:

```
Network Reachability Detection Server: nw-server1
```

Step 18 Check the network reachability server name bound to a local subscriber profile by entering the following command:

```
show subscribers configuration username <subscriber_name>
```

Where *<subscriber_name>* is the name of the local subscriber to which you bound a network reachability server name. The output of this command includes information about the network reachability server name that looks similar to the following:

```
network reachability detection server name: nw-server1
```

Step 19 Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

RADIUS Attributes

Attributes defined in a subscriber profile stored remotely on a RADIUS server can be used to bind the network reachability server to a subscriber session. Use the following attributes to bind a network reachability server to a subscriber session;

- **SN-Nw-Reachability-Server-Name**
- **SN1-Nw-Reachability-Server-Name**

The attributes have one possible value, which is a variable that is a string of from 1 to 15 characters in length. This should be the name of the configured network reachability server.

The **SN-Nw-Reachability-Server-Name** attribute is contained in the following dictionaries:

- starent
- starent-835

The **SN1-Nw-Reachability-Server-Name** attribute is contained in the following dictionaries:

- starent-vsai
- starent-vsai-835

Refer to the *AAA Interface Administration and Reference* for more details.

Appendix T

Remote Address-based RADIUS Accounting

This chapter provides information on configuring an enhanced, or extended, service. The product administration guides provide examples and procedures for the configuration of basic services on the system. It is recommended that you select the configuration example that best meets your service model, and configure the required elements for that model before using the procedures in this chapter.

This chapter includes the following sections:

- [Overview](#)
- [Configuring Remote Address-based Accounting](#)
- [Subscriber Attribute Configuration](#)

Overview

Remote address-based RADIUS accounting counts the number of octets exchanged between individual subscribers and specific remote IP addresses, or networks, during a packet data session. Data from the subscriber to the remote addresses, and data from the remote addresses to the subscriber are accounted for separately.

The remote addresses for which to collect RADIUS accounting data are configured in lists on a per-context basis. Individual subscribers are associated with particular address lists through the configuration or specification of an attribute in their locally configured or RADIUS server-based profiles. Once the lists and subscriber profiles are configured, accounting data collection can be enabled on the system.

Remote address-based RADIUS accounting is implemented in the system according to the specifications described in TIA/EIA/IS-835-B, CDMA2000 Wireless IP Network Standard, October 2002 and 3GPP2 X.S0011-005-D.

License Requirements

The Remote address-based RADIUS Accounting is a licensed Cisco feature. A separate feature license may be required. Contact your Cisco account representative for detailed information on specific licensing requirements. For information on installing and verifying licenses, refer to the Managing License Keys section of the Software Management Operations chapter in the *System Administration Guide*.

Configuring Remote Address-based Accounting

To configure this functionality, a list of up to ten remote addresses or networks is configured in the authentication context, the list is assigned to a subscriber, and remote address collection is enabled.

Use the following configuration example to configure remote address-based accounting:

```
configure

context <context_name>

    radius group <group_name>

    radius accounting ip remote-address list <list_id>

    address <ipv4_address/ipv6_address> netmask <netmask>

end
```

Verifying the Remote Address Lists

Use the following command to verify the remote address lists:

```
show configuration context <context_name>
```

Output similar to the following is displayed.

```
[local] host_name # show configuration context <context_name>
```

```
configure

context <context_name>

    subscriber default

    exit

radius accounting ip remote-address list 1

    address <ipv4_address/ipv6_address> netmask <netmask>

    address <ipv4_address/ipv6_address> netmask <netmask>

    address <ipv4_address/ipv6_address> netmask <netmask>

end
```

Notes:

- Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Subscriber Attribute Configuration

Subscriber attributes are configured as part of their profile. Subscriber profiles can be configured either remotely on a RADIUS server or locally on the system.

This section provides information and procedures on the attributes used to support this functionality.



Important: Since the instructions for configuring subscribers differ between RADIUS server applications, this section only provides the individual attributes that can be added to the subscriber profile. Please refer to the documentation that shipped with your RADIUS server for instructions on configuring subscribers.

Supported RADIUS Attributes

The following RADIUS attributes are used to configure remote address-based RADIUS accounting for a subscriber session. For specific information on each attribute, if you are using StarOS 12.3 or an earlier release, see the *AAA and GTPP Interface Administration and Reference*. If you are using StarOS 14.0 or a later release, refer to the *AAA Interface Administration and Reference*.

- 3GPP2-Remote-Addr-Table-Index
- 3GPP2-Remote-IPv4-Address
- 3GPP2-Remote-IPv4-Addr-Octets

Configuring Local Subscriber Profiles

Use the following example to configure local subscriber profiles to support the Remote Address-based RADIUS Accounting feature:

```
configure
  context <context_name>
    subscriber name <name>
      radius accounting ip remote-address list-id <list_id>
    end
  end
configure
  context <context_name>
    radius accounting ip remote-address collection
  end
```

Notes:

- Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Appendix U

Traffic Policing and Shaping

This chapter describes the support of per subscriber Traffic Policing and Shaping feature on Cisco's Chassis and explains the commands and RADIUS attributes that are used to implement this feature. The product Administration Guides provide examples and procedures for configuration of basic services on the system. It is recommended that you select the configuration example that best meets your service model, and configure the required elements for that model, as described in the respective product Administration Guide, before using the procedures in this chapter.



Important: Traffic Policing and Shaping is a licensed Cisco feature. A separate feature license may be required. Contact your Cisco account representative for detailed information on specific licensing requirements. For information on installing and verifying licenses, refer to the *Managing License Keys* section of the *Software Management Operations* chapter in the *System Administration Guide*.

This chapter included following procedures:

- [Overview](#)
- [Traffic Policing Configuration](#)
- [Traffic Shaping Configuration](#)
- [RADIUS Attributes](#)

Overview

This section describes the traffic policing and shaping feature for individual subscriber. This feature is comprised of two functions:

- Traffic Policing
- Traffic Shaping

Traffic Policing

Traffic policing enables the configuring and enforcing of bandwidth limitations on individual subscribers and/or APN of a particular traffic class in 3GPP/3GPP2 service.

Bandwidth enforcement is configured and enforced independently on the downlink and the uplink directions.

A Token Bucket Algorithm (a modified trTCM) [RFC2698] is used to implement the Traffic-Policing feature. The algorithm used measures the following criteria when determining how to mark a packet:

- **Committed Data Rate (CDR):** The guaranteed rate (in bits per second) at which packets can be transmitted/received for the subscriber during the sampling interval.
- **Peak Data Rate (PDR):** The maximum rate (in bits per second) that subscriber packets can be transmitted/received for the subscriber during the sampling interval.
- **Burst-size:** The maximum number of bytes that can be transmitted/received for the subscriber during the sampling interval for both committed (CBS) and peak (PBS) rate conditions. This represents the maximum number of tokens that can be placed in the subscriber's "bucket". Note that the committed burst size (CBS) equals the peak burst size (PBS) for each subscriber.

The system can be configured to take any of the following actions on packets that are determined to be in excess or in violation:

- **Drop:** The offending packet is discarded.
- **Transmit:** The offending packet is passed.
- **Lower the IP Precedence:** The packet's ToS bit is set to "0", thus downgrading it to Best Effort, prior to passing the packet. Note that if the packet's ToS bit was already set to "0", this action is equivalent to "Transmit".

Traffic Shaping

Traffic Shaping is a rate limiting method similar to the Traffic Policing, but provides a buffer facility for packets exceeded the configured limit. Once the packet exceeds the data-rate, the packet queued inside the buffer to be delivered at a later time.

The bandwidth enforcement can be done in the downlink and the uplink direction independently. If there is no more buffer space available for subscriber data system can be configured to either drop the packets or kept for the next scheduled traffic session.





Important: Traffic Shaping is not supported on the GGSN, P-GW, or SAEGW.

Traffic Policing Configuration


Traffic Policing is configured on a per-subscriber basis. The subscribers can either be locally configured subscribers on the system or subscriber profiles configured on a remote RADIUS server.

In 3GPP service Traffic policing can be configured for subscribers through APN configuration as well.

 **Important:** In 3GPP service attributes received from the RADIUS server supersede the settings in the APN.

 **Important:** Commands used in the configuration samples in this section provide base functionality to the extent that the most common or likely commands and/or keyword options are presented. In many cases, other optional commands and/or keyword options are available. Refer to the *Command Line Interface Reference* for complete information regarding all commands.

Configuring Subscribers for Traffic Policing

 **Important:** Instructions for configuring RADIUS-based subscriber profiles are not provided in this document. Please refer to the documentation supplied with your server for further information.

Step 1 Configure local subscriber profiles on the system to support Traffic Policing by applying the following example configurations:

Step a To apply the specified limits and actions to the downlink (data to the subscriber):

```
configure
    context <context_name>
        subscriber name <user_name>
            qos traffic-police direction downlink
        end
    end
```

Step b To apply the specified limits and actions to the uplink (data from the subscriber):

```
configure
    context <context_name>
        subscriber name <user_name>
            qos traffic-police direction uplink
        end
    end
```

Notes:

- There are numerous keyword options associated with the **qos traffic-police direction { downlink | uplink }** command.
- Repeat for each additional subscriber to be configured.



Important: If the exceed/violate action is set to “lower-ip-precedence”, the TOS value for the outer packet becomes “best effort” for packets that exceed/violate the traffic limits regardless of what the **ip user-datagram-tos-copy** command in the Subscriber Configuration mode is configured to. In addition, the “lower-ip-precedence” option may also override the configuration of the **ip qos-dscp** command (also in the Subscriber Configuration mode). Therefore, it is recommended that command not be used when specifying this option.

Step 2 Verify the subscriber profile configuration by applying the following example configuration:

```
context <context_name>

show subscriber configuration username <user_name>
```

Step 3 Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Configuring APN for Traffic Policing in 3GPP Networks

This section provides information and instructions for configuring APN template’s QoS profile in support of Traffic Policing.

The profile information is sent to the SGSN(s) in response to GTP Create/Update PDP Context Request messages. If the QoS profile requested by the SGSN is lower than the configured QoS profile configured, the profile requested by the SGSN is used. If the QoS profile requested by the SGSN is higher, the configured rates are used.

Note that values for the committed-data-rate and peak-data-rate parameters are exchanged in the GTP messages between the GGSN and the SGSN. Therefore, the values used may be lower than the configured values. When negotiating the rate with the SGSN(s), the system convert this to a value that is permitted by GTP as shown in the table below.

Table 60. Permitted Values for Committed and Peak Data Rates in GTP Messages

Value (bps)	Increment Granularity (bps)
From 1000 to 63,000	1,000 (e.g 1000, 2000, 3000, ... 63000)
From 64,000 to 568,000	8,000 (e.g. 64000, 72000, 80000, ... 568000)
From 576,000 to 8,640,000	64,000 (e.g. 576000, 640000, 704000, ... 86400000)
From 8,700,000 to 16,000,000	100,000 bps (e.g. 8700000, 8800000, 8900000, ... 16000000)

Step 1 Set parameters by applying the following example configurations:

Step a To apply the specified limits and actions to the downlink (the Gn direction):

```
configure
```

```

context <context_name>

  apn <apn_name>

    qos rate-limit downlink

  end

```

Step b To apply the specified limits and actions to the uplink (the Gi direction):

configure

```

context <context_name>

  apn <apn_name>

    qos rate-limit uplink

  end

```

Notes:

- There are numerous keyword options associated with **qos rate-limit { downlink | uplink }** command.
- *Optionally*, configure the maximum number of PDP contexts that can be facilitated by the APN to limit the APN's bandwidth consumption by entering the following command in the configuration:

```
max-contents primary <number> total <total_number>
```

- Repeat as needed to configure additional Qos Traffic Policing profiles.



Important: If a “subscribed” traffic class is received, the system changes the class to background and sets the following: The uplink and downlink guaranteed data rates are set to 0. If the received uplink or downlink data rates are 0 and traffic policing is disabled, the default of 64 kbps is used. When enabled, the APN configured values are used. If the configured value for downlink max data rate is larger than can fit in an R4 QoS profile, the default of 64 kbps is used. If either the received uplink or downlink max data rates is non-zero, traffic policing is employed if enabled for the background class. The received values are used for responses when traffic policing is disabled.

Step 2 Verify that your APNs were configured properly by entering the following command:

```
show apn { all | name <apn_name> }
```


The output is a concise listing of configured APN parameter settings.


Step 3 Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.


Traffic Shaping Configuration

Traffic Shaping is configured on a per-subscriber basis. The subscribers can either be locally configured subscribers on the system or subscriber profiles configured on a remote RADIUS server.

In 3GPP service Traffic policing can be configured for subscribers through APN configuration as well.


 **Important:** In 3GPP, service attributes received from the RADIUS server supersede the settings in the APN.

 **Important:** Commands used in the configuration samples in this section provide base functionality to the extent that the most common or likely commands and/or keyword options are presented. In many cases, other optional commands and/or keyword options are available. Refer to the *Command Line Interface Reference* for complete information regarding all commands.

 **Important:** Traffic Shaping is not supported on the GGSN, P-GW, or SAEGW.

Configuring Subscribers for Traffic Shaping

This section provides information and instructions for configuring local subscriber profiles on the system to support Traffic Shaping.

 **Important:** Instructions for configuring RADIUS-based subscriber profiles are not provided in this document. Please refer to the documentation supplied with your server for further information.

Step 1 Set parameters by applying the following example configurations:

Step a To apply the specified limits and actions to the downlink (data to the subscriber):

```
configure
    context <context_name>
        subscriber name <user_name>
            qos traffic-shape direction downlink
        end
```

Step b To apply the specified limits and actions to the uplink (data to the subscriber):

```
configure
    context <context_name>
        subscriber name <user_name>
            qos traffic-shape direction uplink
```



```
end
```

Notes:

- There are numerous keyword options associated with **qos traffic-shape direction { downlink | uplink }** command.
- Repeat for each additional subscriber to be configured.



Important: If the exceed/violate action is set to “lower-ip-precedence”, the TOS value for the outer packet becomes “best effort” for packets that exceed/violate the traffic limits regardless of what the **ip user-datagram-tos-copy** command in the Subscriber Configuration mode is configured to. In addition, the “lower-ip-precedence” option may also override the configuration of the **ip qos-dscp** command (also in the Subscriber Configuration mode). Therefore, it is recommended that command not be used when specifying this option.

Step 2 Verify the subscriber profile configuration by applying the following example configuration:

```
context <context_name>

show subscriber configuration username <user_name>
```

Step 3 Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Configuring APN for Traffic Shaping in 3GPP Networks

This section provides information and instructions for configuring APN template’s QoS profile in support of Traffic Shaping.

The profile information is sent to the SGSN(s) in response to GTP Create/Update PDP Context Request messages. If the QoS profile requested by the SGSN is lower than the configured QoS profile configured, the profile requested by the SGSN is used. If the QoS profile requested by the SGSN is higher, the configured rates are used.

Note that values for the committed-data-rate and peak-data-rate parameters are exchanged in the GTP messages between the GGSN and the SGSN. Therefore, the values used may be lower than the configured values. When negotiating the rate with the SGSN(s), the system convert this to a value that is permitted by GTP as shown in the following table.

Table 61. Permitted Values for Committed and Peak Data Rates in GTP Messages

Value (bps)	Increment Granularity (bps)
From 1000 to 63,000	1,000 (e.g 1000, 2000, 3000, ... 63000)
From 64,000 to 568,000	8,000 (e.g. 64000, 72000, 80000, ... 568000)
From 576,000 to 8,640,000	64,000 (e.g. 576000, 640000, 704000, ... 86400000)
From 8,700,000 to 16,000,000	100,000 bps (e.g. 8700000, 8800000, 8900000, ... 16000000)

Step 1 Set parameters by applying the following example configurations.

Step a To apply the specified limits and actions to the downlink (data to the subscriber):

```
configure
    context <context_name>
        subscriber name <user_name>
            qos rate-limit downlink
        end
    end
```

Step b To apply the specified limits and actions to the uplink (data to the subscriber):

```
configure
    context <context_name>
        apn <apn_name>
            qos rate-limit uplink
        end
    end
```

Step 2 *Optional.* Configure the maximum number of PDP contexts that can be facilitated by the APN to limit the APN's bandwidth consumption by entering the following command in the configuration:

```
configure
    context <context_name>
        apn <apn_name>
            max-contexts primary <number> total <total_number>
        end
    end
```

Notes:

- There are numerous keyword options associated with **qos rate-limit direction { downlink | uplink }** command.

For more information on commands, refer *Command Line Interface Reference*

- If the exceed/violate action is set to **lower-ip-precedence**, this command may override the configuration of the **ip qos-dscp** command in the GGSN service configuration mode for packets from the GGSN to the SGSN. In addition, the GGSN service **ip qos-dscp** command configuration can override the APN setting for packets from the GGSN to the Internet. Therefore, it is recommended that command not be used in conjunction with this action.
- Repeat as needed to configure additional Qos Traffic Policing profiles.
- Note that, if a “subscribed” traffic class is received, the system changes the class to background and sets the following:
 - The uplink and downlink guaranteed data rates are set to 0.
 - If the received uplink or downlink data rates are 0 and traffic policing is disabled, the default of 64 kbps is used. When enabled, the APN configured values are used.

- If the configured value for downlink max data rate is larger than can fit in an R4 QoS profile, the default of 64 kbps is used.
- If either the received uplink or downlink max data rates is non-zero, traffic policing is employed if enabled for the background class. The received values are used for responses when traffic policing is disabled.

Step 3 Verify that your APNs were configured properly by entering the following command:

```
show apn { all | name <apn_name> }
```

The output is a concise listing of configured APN parameter settings.

Step 4 Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

RADIUS Attributes

Traffic Policing for CDMA Subscribers

The RADIUS attributes listed in the following table are used to configure Traffic Policing for CDMA subscribers (PDSN, HA) configured on remote RADIUS servers. More information on these attributes can be found in the *AAA Interface Administration and Reference*.

Table 62. RADIUS Attributes Required for Traffic Policing Support for CDMA Subscribers

Attribute	Description
SN-QoS-Tp-Dnlk (or SN1-QoS-Tp-Dnlk)	Enable/disable traffic policing in the downlink direction.
SN-Tp-Dnlk-Committed-Data-Rate (or SN1-Tp-Dnlk-Committed-Data-Rate)	Specifies the downlink committed-data-rate in bps.
SN-Tp-Dnlk-Peak-Data-Rate (or SN1-Tp-Dnlk-Committed-Data-Rate)	Specifies the downlink peak-data-rate in bps.
SN-Tp-Dnlk-Burst-Size (or SN1-Tp-Dnlk-Burst-Size)	Specifies the downlink-burst-size in bytes. NOTE: It is recommended that this parameter be configured to at least the greater of the following two values: 1) 3 times greater than packet MTU for the subscriber connection, OR 2) 3 seconds worth of token accumulation within the “bucket” for the configured peak-data-rate.
SN-Tp-Dnlk-Exceed-Action (or SN1-Tp-Dnlk-Exceed-Action)	Specifies the downlink exceed action to perform.
SN-Tp-Dnlk-Violate-Action (or SN1-Tp-Dnlk-Violate-Action)	Specifies the downlink violate action to perform.
SN-QoS-Tp-Upk (or SN1-QoS-Tp-Upk)	Enable/disable traffic policing in the downlink direction.

Attribute	Description
SN-Tp-Uplk-Committed-Data-Rate (or SN1-Tp-Uplk-Committed-Data-Rate)	Specifies the uplink committed-data-rate in bps.
SN-Tp-Uplk-Peak-Data-Rate (or SN1-Tp-Uplk-Committed-Data-Rate)	Specifies the uplink peak-data-rate in bps.
SN-Tp-Uplk-Burst-Size (or SN1-Tp-Uplk-Burst-Size)	Specifies the uplink-burst-size in bytes. NOTE: It is recommended that this parameter be configured to at least the greater of the following two values: 1) 3 times greater than packet MTU for the subscriber connection, OR 2) 3 seconds worth of token accumulation within the “bucket” for the configured peak-data-rate.
SN-Tp-Uplk-Exceed-Action (or SN1-Tp-Uplk-Exceed-Action)	Specifies the uplink exceed action to perform.
SN-Tp-Uplk-Violate-Action (or SN1-Tp-Uplk-Violate-Action)	Specifies the uplink violate action to perform.

Traffic Policing for UMTS Subscribers

The RADIUS attributes listed in the following table are used to configure Traffic Policing for UMTS subscribers configured on remote RADIUS servers. More information on these attributes can be found in the *AAA Interface Administration and Reference*.

Table 63. RADIUS Attributes Required for Traffic Policing Support for UMTS Subscribers

Attribute	Description
SN-QoS-Conversation-Class (or SN1-QoS-Conversation-Class)	Specifies the QOS Conversation Traffic Class.
SN-QoS-Streaming-Class (or SN1-QoS-Streaming-Class)	Specifies the QOS Streaming Traffic Class.

■ RADIUS Attributes

Attribute	Description
SN-QoS-Interactive1-Class (or SN1-QoS-Interactive1-Class)	Specifies the QOS Interactive Traffic Class.
SN-QoS-Interactive2-Class (or SN1-QoS-Interactive2-Class)	Specifies the QOS Interactive2 Traffic Class.
SN-QoS-Interactive3-Class (or SN1-QoS-Interactive3-Class)	Specifies the QOS Interactive3 Traffic Class.
SN-QoS-Background-Class (or SN1-QoS-Background-Class)	Specifies the QOS Background Traffic Class.
SN-QoS-Traffic-Policy (or SN1-QoS-Traffic-Policy)	<p>This compound attribute simplifies sending QoS values for Traffic Class (the above attributes), Direction, Burst-Size, Committed-Data-Rate, Peak-Data-Rate, Exceed-Action, and Violate-Action from the RADIUS server.</p> <p>This attribute can be sent multiple times for different traffic classes. If Class is set to 0, it applies across all traffic classes.</p>