Cisco ASR 5000 Series Enhanced Feature Configuration Guide

Release 8.x and 9.0

Last Updated June 30, 2010

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-22982-01

Cisco ASR 5000 Series Enhanced Feature Configuration Guide

CONTENTS

About this Guide

This document pertains to features and functionality that run on and/or that are related to the Cisco® ASR 5000 Chassis, formerly the Starent Networks ST40.

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.
	Electro-Static Discharge (ESD)	Alerts you to take proper grounding precautions before handling a product.

Typeface Conventions	Description
Text represented as a `screen display`	This typeface represents displays that appear on your terminal screen, for example: `Login:`
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 *`variable`* }	Required keywords and variables are surrounded by grouped brackets. Required keywords and variables are those components that are required to be entered as part of the command syntax.

Command Syntax Conventions	Description
[**keyword** or *variable*]	Optional keywords or variables, or those that a user may or may not choose to use, are surrounded by square brackets.
|	With some commands there may be a group of variables from which the user chooses one. These are called alternative variables and are documented by separating each variable with a vertical bar (also known as a pipe filter). Pipe filters can be used in conjunction with required or optional keywords or variables. For example: **{ nonce | timestamp }** OR [**count** *number_of_packets* | **size** *number_of_bytes*]

Contacting Customer Support

Use the information in this section to contact customer support.

For New Customers: 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 is required to this site. Please contact your local sales or service representative for additional information.

For Existing Customers with support contracts through Starent Networks: Refer to the support area of https://support.starentnetworks.com/ for up-to-date product documentation or to submit a service request. A valid username and password is required to this site. Please contact your local sales or service representative for additional information.

Important: For warranty and repair information, please be sure to include the Return Material Authorization (RMA) tracking number on the outside of the package.

Chapter 1
Command Line Interface Overview

This chapter describes the numerous features in the command line interface (CLI). Included is information about the architecture of the CLI, its command modes and user privileges, how to obtain help within the CLI, and other key items.

The operating system provides the software that controls the overall system logic, control processes, and the CLI. The CLI is a multi-threaded user interface that allows you to manipulate, configure, control, and query the hardware and software components that make up the system and its hosted services. In addition, the CLI can host multiple instances of management and service configuration sessions. This allows multiple users to simultaneously access and manage multiple hosted services.

This section provides the following information about the CLI:

CLI Structure

CLI commands are strings of commands or keywords and user-specified arguments that set or modify specific parameters of the system. Commands are grouped by function and the various command modes with which they are associated.

The structure of the CLI is hierarchical. All users begin at a specific entry point into the system, called the Exec (Execute) Mode, and then navigate through the CLI according to their defined user privileges (access level) by using other command modes.

CLI Command Modes

There are two primary CLI command modes:

- **Exec (Execute) Mode:** The Exec mode is the lowest level in the CLI. The Exec mode is where you execute basic commands such as show, and ping. When you log into the CLI, you are placed in this mode by default.
- **Config (Configuration) Mode:** The Config mode is accessible only by users with administrator and security administrator privileges. If you are an administrative user, in this mode you can add and configure contexts and access the configuration sub-modes to configure protocols, interfaces, ports, services, subscribers, and other service-related items.

As explained above, the entry point into the CLI is called Exec Mode. In the initial CLI login, all users are placed into the default local context, which is the CLI's default management context. From this context, administrative users can access the Config Mode and define multiple service contexts.

Refer to the mode entry-path diagrams at the beginning of each mode chapter in the *Command Line Interface Reference*.

Important: The commands or keywords/variables that are available are dependent on platform type, product version, and installed license(s).

CLI Administrative Users

This section contains information on the administrative user types and privileges supported by the system.

Administrative User Types

There are two types of administrative users supported by the system:

- **Context-level administrative users:** This user type is configured at the context-level and relies on the AAA subsystems for validating usernames and passwords during login. This is true for both administrative user accounts configured locally through a configuration file or on an external RADIUS server. Passwords for these user types are assigned once and are accessible in the configuration file.
- **Local-users:** This user type provides support for ANSI T1.276-2003 password security protection. Local-user account information, such as passwords, password history, and lockout states, is maintained in non-volatile memory on the CompactFlash module and in the Shared Configuration Task (SCT). This information is maintained in a separate file, not in configuration files used by the system. As such, the configured local-user accounts are not visible with the rest of the system configuration.

Local-user and context-level administrative accounts can be used in parallel. However, a mechanism is provided to de-activate context-level administrative user accounts thereby providing access only to local-user accounts.

Authenticating Administrative Users with RADIUS

To authorize users via RADIUS, you must include two RADIUS attributes in the RADIUS Access-Accept message:

- RFC 2865 standard Service-Type
- Starent Vendor-Specific Attribute (VSA) SN1-Admin-Permission.

The default permission is none (0), meaning that service is refused even if properly authenticated via RADIUS.

RADIUS Mapping System

RADIUS server configuration depends on the type of server used and the instructions distributed by the server manufacturer. The following table shows the attribute/value mapping system that is constant, regardless of server manufacturer or model:

Table 1. RADIUS Attribute/Value Mapping System

Attribute	Value

Attribute	Value
Login (Operator)	1
Framed	2
Callback_Login	3
Callback_Framed	4
Outbound	5
Administrative (Administrator)	6
NAS_Prompt	7
Authenticate_Only	8
Callback_NAS_Prompt	9
Call_Check	10
Callback_Administrative	11
Voice	12
Fax	13
Modem_Relay	14
IAPP_Register	15
IAPP_AP_Check	16
Authorize_Only	17
Inspector	19650516
Security_Admin	19660618

RADIUS Privileges

There are four RADIUS privilege roles. The following table shows the relationship between the privilege roles in the CLI configuration and RADIUS Service-Type.

Table 2. CLI Privilege Roles and RADIUS Service Types

CLI Configuration Parameter	RADIUS Service Type	Show Admin Type
administrator	Security_Admin (19660618)	admin
config_administrator	Administrative (6)	cfgadm
operator	NAS_Prompt (7)	oper
inspector	Inspector (19650516)	inspect

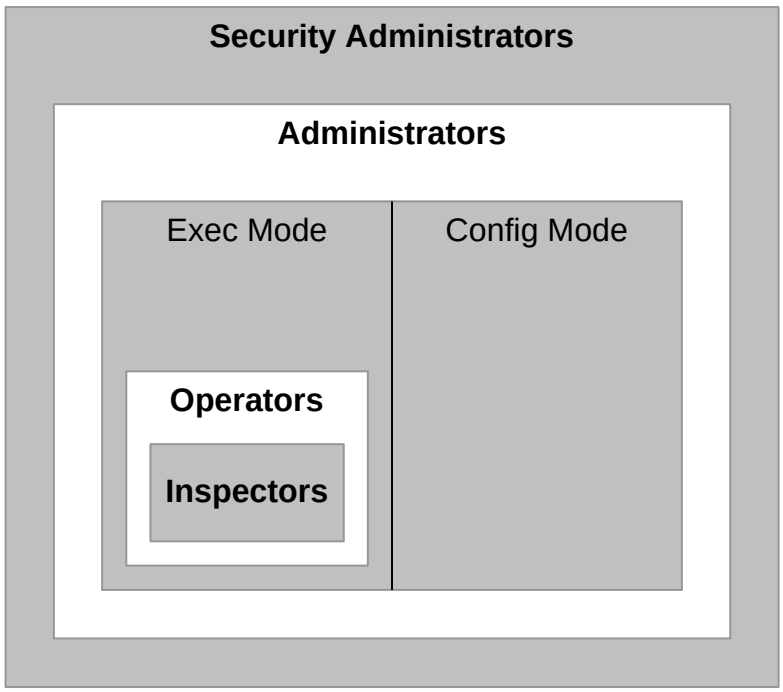
Administrative User Privileges

Regardless of the administrative user type, the system supports four user privilege levels:

- **Inspector**: Inspectors are limited to a small number of read-only Exec Mode commands. The bulk of these are show commands for viewing a variety of statistics and conditions. The Inspector cannot execute show configuration commands and does not have the privilege to enter the Config Mode.
- **Operator**: Operators have read-only privileges to a larger subset of the Exec Mode commands. They can execute all commands that are part of the inspector mode, plus some system monitoring, statistic, and fault management functions. Operators do not have the ability to enter the Config Mode.
- **Administrator**: Administrators have read-write privileges and can execute any command in the CLI except for a few security-related commands that can only be configured by Security Administrators. Administrators can configure or modify system settings and can execute all system commands, including those available to the Operators and Inspectors.
- **Security Administrator**: Security Administrators have read-write privileges and can execute all CLI commands, including those available to Administrators, Operators, and Inspectors.

The following figure represents how user privileges are defined in the CLI configuration modes.

Figure 1. User Privileges



Though the privilege levels are the same regardless of user type, the corresponding user type names differ slightly. The following table displays the privilege level to administrative user type mappings:

Table 3. User Privilege to User Type Mapping

User Type as Defined by T1.276-2003	Local-User Level User	Context-Level User
System Security Administrator	Security Administrator	Administrator
Application Security Administrator	Security Administrator	Administrator
System Administrator	Administrator	Config-Administrator
Application Administrator	Administrator	Config-Administrator
Application User/Operator	Operator	Operator
not applicable	Inspector	Inspector

Configure context-level administrative users in the Context Configuration Mode with the **`administrator`**, **`config-administrator`**, **`operator`**, and **`inspector`** commands.

Configure local-user administrative users at the Global Configuration Mode with the **`local-user username`** command.

You can further refine administrative levels to include access to certain features with the following feature-use administrative user options:

- **Lawful Intercept (LI) Administrative User:** To configure and manage LI-related issues, configure at least one administrative user account with LI functionality privileges.

Important: This privilege is available only for context-level administrative users. In addition, to ensure security in accordance with the standards, LI administrative users must access the system through the Secure Shell Protocol (SSH).

- **Enhanced Charging Service (ECS) Administrative User:** To log in and execute ECS-related commands, configure at least one administrative user account with ECS functionality privileges.

All system users can be configured within any context. However, it is recommended that you configure users in the system's management context called local. Refer to sections later in this chapter for additional information about contexts.

Allowed Commands per User Type

With the exception of security administrators, all other management users are limited to a subset of the entire command list as described in the *Command Line Interface Reference*. This section defines the commands allowed for each management user type. As stated previously, inspectors and operators are limited to only a subset of the Exec Mode commands.

Inspector Mode Commands

In the Exec mode, system inspectors can access the following commands:

- abort
- autoconfirm
- context
- crypto-group
- default terminal
- exit
- help
- logs checkpoint
- monitor subscriber
- no logging active
- no logging trace
- no reveal disabled commands
- no timestamps
- no autoconfirm
- ping
- reveal disabled commands
- show (except show snmp communities and show snmp transports)
- sleep
- start crypto security-association
- terminal length
- terminal width
- timestamps
- traceroute

Operator Mode Commands

In the Exec mode, system operators can access all inspector mode commands plus the following commands:

- aaa test
- alarm cutoff
- bulkstats force
- card
- clear (a subset of all clear command variations)

- debug
- dhcp test
- gtpc test
- gtpp interim
- gtpp test
- gtpu test
- gtpv0 test
- host
- logging active
- logging filter
- logging trace
- newcall
- no card
- no debug
- no newcall policy
- port
- ppp echo-test
- radius interim accounting
- radius test
- rlogin
- show access-group
- show access-list
- show access-flow
- show access statistics
- show configuration
- show snmp transports
- ssh
- telnet
- test alarm

Administrator Mode Commands

Administrators can access all system commands except:

Context Config Mode

- config-administrator
- operator

- inspector
- administrator

Global Config Mode

- snmp community
- snmp user
- local-user
- suspend local-user

Exec Mode

- show snmp communities
- clear (all clear command variations)
- show local-user
- password change local-user

Security Administrator Mode Commands

Security administrators can access all system commands.

CLI Contexts

A context is a group of configuration parameters that apply to the ports, interfaces, and protocols supported by the system. You can configure multiple contexts on the system, each of which resides as a separate, logically independent instance on the same physical device. The CLI can host multiple contexts within a single physical device. This allows wireless service providers to use the same system to support:

- Different levels of service
- Multiple wholesale or enterprise customers or customer groups
- Different classes of customers based on defined Class of Service (CoS) parameters
- IP address pools across multiple contexts, thus saving IP address allocation
- Enhanced security

Each defined context operates independently from any other context(s) in the system. Each context contains its own CLI instance, IP routing tables, access filters, compression methods, and other configured data.

By default, a single system-wide context called local, is used exclusively for the management of the system. Think of the local context as the root directory of the system, since you can define and access all other contexts from this point. You cannot delete the local context. From this location in the CLI, you can:

- Create and configure other service contexts that contain different service configurations
- Configure system-wide services such as CORBA and SNMP management interfaces, physical management ports, system messages, and others

Important: The system requires that you define at least one context in addition to the Local context. This isolates system management functions from application or service functions.

Administrative users add contexts through the Global Configuration Mode. A substantial advantage of configuring numerous service contexts is that it allows operators to broadly distribute different subscribers across the system. This greatly enhances the performance of the system and minimizes the loss of sessions should a failure occur.

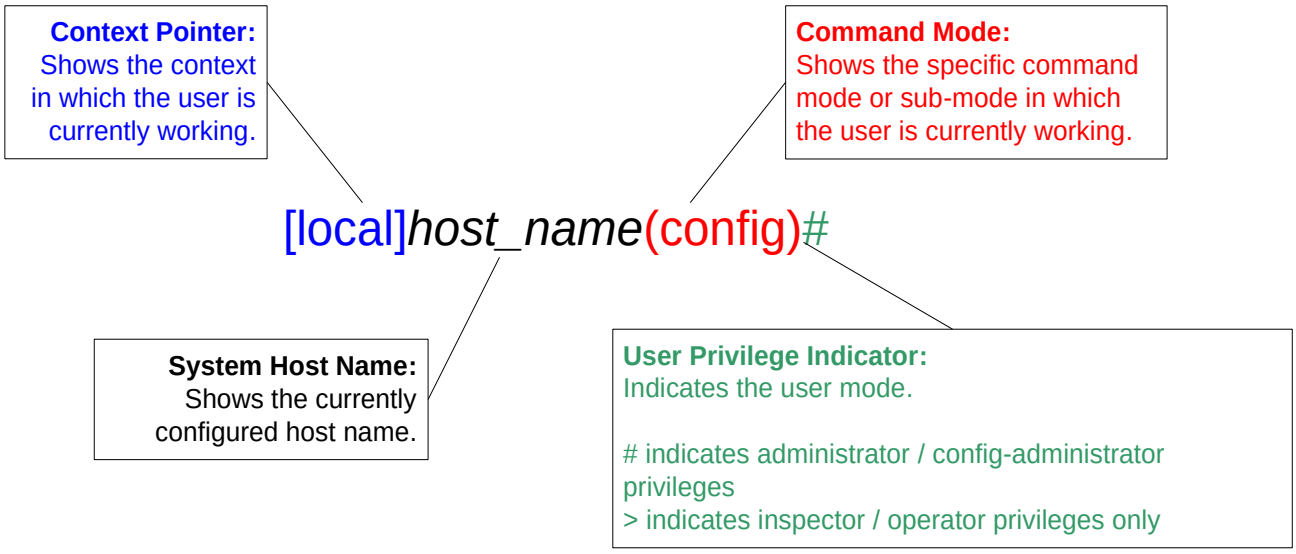
Understanding the CLI Command Prompt

The CLI provides an intuitive command prompt that informs you of:

- Exactly where you are located within the CLI
- The command mode you are using
- Your user privilege level.

The following figure shows the various components of the command prompt.

Figure 2. CLI Command Prompt



CLI Command Syntax

This section describes the components of the CLI command syntax that you should be familiar with prior to using the CLI. These include:

- **Commands**: Specific words that precede, or initiate, a specific function.
- **Keywords**: Specific words that follow a command to more clearly dictate the command's function.
- **Variables**: Alpha, numeric, or alphanumeric values that are user-supplied as part of the command syntax. Sometimes referred to as arguments, these terms further specify the command function.
- **Repetitive keywords (+)**: Specific keyword, that when followed by a plus (+) sign, indicates that more than one of the keywords can be entered within a single command.

Example
In the following example, *slot_number* is the command variable for the **info** keyword:

```
show slot infoslot_number
```

slot_number is a variable representing a particular slot (1 through 48).

Entering and Viewing CLI Commands

This section describes various methods for entering commands into the CLI.

Typing each command keyword, argument, and variable can be time-consuming and increase your chance of making mistakes. The CLI therefore, supports the following features to assist you in entering commands quickly and more accurately. Other features allow you to view the display and review previously entered commands.

Entering Partial CLI Commands

In all of the modes, the CLI recognizes partially-typed commands and keywords, as long as you enter enough characters for the command to be unambiguously recognized by the system. If you do not enter enough characters for the system to recognize a unique command or keyword, it returns a message listing all possible matches for the partial entry.

Example

If you enter the partial command **conf** and press <**Enter**>, you enter the Global Configuration Mode. If you were to enter only **co**, the system would respond with the message:

```
Ambiguous Command
```

CLI Command Auto-completion

Use the command auto-completion feature to automatically complete unique CLI commands. Press the <**Tab**> key after entering enough characters to enable this feature.

Example

[local]*host_name*# sho<Tab>
[local]*host_name*# show

If you do not enter enough characters to allow the CLI to determine the appropriate command to use, the CLI displays all commands that match the characters you entered with auto-completion:

Example

```
[local]host_name# sh<Tab>

show     shutdown

[local]host_name#
```

Enter a question mark (?) after a partial command to display all of the possible matching commands, and their related help text.

Example

```
[local]host_name# sh?

show - Displays information based on a specified argument
```

```
shutdown - Terminates execution of all tasks within the entire chassis

[local]host_name#
```

Using CLI Auto-Pagination

When you enter commands whose expected results exceed the terminal window's vertical display, the auto-pagination function pauses the display each time the terminal window reaches its display limit. Press any key to display the next screen of results.

By default, auto-pagination functionality is disabled. To enable auto-pagination, type the pipe command: **| more**

```
[local]host_name# show configuration | more
```

Important: When auto-pagination is enabled, if a command's output exceeds the terminal window's vertical display parameters, you can exit by entering "**q**". This returns you to the CLI prompt.

Using CLI Autoconfirmation

By default, the system is configured to prompt all administrative users with a confirmation prior to executing certain commands. This functionality serves two purposes:

- Helps ensure that you do not execute an unwanted configuration change.

Example
Saving a configuration:

```
[local]host_name# save configuration

Are you sure ? [Yes | No]:
```

- Indicates potential misspellings of names during configuration. The first time you configure an element name (context, subscribers, services, etc.), the prompt is displayed. The prompt is not displayed for subsequent entries of the name. Therefore, if you see the confirmation prompt after entering the name of a previously configured element, it is likely that you misspelled the name.

Examples
You create context named "newcontext":

```
[local]host_name(config)# context newcontext

Are you sure ? [Yes | No]: yes

[newcontext]host_name(config-ctx)#
```

You revisit the context named “newcontext”:

```
[local]host_name(config)# context newcontext

[newcontext]host_name(config-ctx)#
```

On another occasion, you misspell the context named “newcontext”:

```
[local]host_name(config)# context mewcontext

Are you sure ? [Yes | No]:n

Action aborted

[local]host_name(config)#
```

After aborting the above action, you can again revisit “newcontext”:

```
[local]host_name(config)# context newcontext

[newcontext]host_name(config-ctx)#
```

You can control CLI autoconfirmation at the following levels:

- **Specific administrative user sessions:** To enable or disable autoconfirmation, use the [no] autoconfirm commands while in the Exec mode.
- **All Future Sessions:** To disable or re-enable autoconfirmation for all future sessions, use the [no]autoconfirm commands while in the Global Config mode.
- **For specific commands:** Disable autoconfirmation for various commands that support the -noconfirm keyword, such as the save configuration or card reboot commands.

Regulating the Command Output

For many CLI commands, you can use **| grep** and/or **| more** keywords to regulate or control the command’s output.

Use the | grep keyword to filter through a command’s output for certain expressions or patterns. Only those portions of the output that contain or exclude the pattern are displayed. The | grep has the following syntax:

| grep [-i | -v | --ignore-case | --invert-match] *expression*

Table 4. grep Keywords

Alternative Keyword	Description
-i	Specifies the filtering of the command’s output for a particular expression while ignoring case. Lower case matches the same as upper case.
-v	Specifies the filtering of the command’s output for everything excluding a particular expression.
--ignore-case	The long form of the **-i** option.

Alternative Keyword	Description
--invert-match	The long form of the **-v** option.
expression	Specifies the character pattern to find in the command's output.

Use the **| more** keyword to pause the terminal each time the terminal window reaches its display limit. Press any key to display the next screen. The function of this keyword is identical to the **autoless** command, except that you must manually enter it on a command-by-command basis.

Viewing Command History

To view a history of all commands line by line, simply scroll up or down with the <**up arrow**> and <**down arrow**> cursor keys on the keyboard.

The operating system supports EMACS-style text editing commands. This standard UNIX text editor format allows you to use keyboard-based shortcut keys for maneuvering around the CLI. The following table lists these available shortcut keys.

Table 5. EMACS Shortcut Keystrokes

Shortcut Keys	Description
<Ctrl + p> and <up arrow>	Recalls previous command in the command history
<Ctrl + n> and <down arrow>	Recalls next command in the command history
<Ctrl + f> and <right arrow>	Moves cursor forward by one character in command line
<Ctrl + b> and <left arrow>	Moves cursor backward by one character in command line
<Esc> + <f>	Moves cursor forward by one word in command line
<Esc> + <b>	Moves cursor backward by one word in command line
<Ctrl> + <a>	Moves cursor to the beginning of the command line
<Ctrl> + <e>	Moves cursor to the end of the command line
<Ctrl> + <k>	Deletes the current command line from the insertion point to the end of the line
<Ctrl> + <u>	Deletes the current command line from the insertion point to the beginning of the line
<Ctrl> + <d>	Deletes a single character in the current command line
<Esc> + <d>	Deletes a word in the current command line
<Ctrl> + <c>	Quits editing the current line
<Ctrl> + <l>	Refreshes the display
<Ctrl> + <t>	Transposes (or switches) the two characters surrounding the insertion point

Obtaining CLI Help

The CLI provides context-sensitive help for every command token and keyword available to you. To obtain, use one of these methods:

- **Command Help:** Command help provides assistance for a specific command. Type a question mark (?) at the end of the specific command to accesses help.

Example

```
[local]host_name# test?

test - Performs test on followed mechanism
```

- **Keyword Help:** Keyword help provides assistance in determining the next keyword, argument, or option to use in the command syntax. Enter the command keyword, enter a space, and then type a question mark (?).

Example

```
[local]host_name# test alarm ?

audible - Tests internal audible alarm buzzer on SPC

central-office - Tests specified central office alarm relays on SPIO card

<cr> - newline
```

- **Variable Help:** Variable help provides the correct format, value, or information type for each variable that is part of the command syntax. For commands with variables, enter the command keyword, enter a space, and then type a question mark (?).

Example

```
[local]host_name# show card info ?

<Enter card number as an integer ranging 1 to 48> | - Pipeline <cr> -
Carriage Return or <Enter> key
```

Exiting the CLI and CLI Command Modes

A CLI session is defined as the successful login into the CLI. When you establish a CLI session, you are placed into the system's Exec Mode. Depending upon your user privilege level, you can:

- Use the *local* context to perform system management functions
- Move to an assigned context and work in Exec Mode
- Move to an assigned context as an administrative user and work in Global Configuration Mode or other configuration sub-mode

This section addresses how to properly exit the various modes and the CLI.

Exiting Configuration Sub-modes

To exit a configuration sub-mode and return to the next highest configuration sub-mode or Global Configuration Mode, type the exit command at the system prompt.

Example

```
[context_name]host_name(config-ctx)# exit

[local]host_name(config)#
```

Important: The CLI supports implicit mode-exits when using configuration files. Therefore, configuration files do not have to contain all of the required exit commands for you to leave various sub-config modes.

To exit a sub-mode and return to the Exec Mode, enter the **end** command.

Example

```
[local]host_name(config-ctx)# end

[local]host_name#
```

Exiting Global Configuration Mode

To exit Global Configuration Mode, and return to the Exec Mode prompt, type the **exit** command at the prompt.

Ending a CLI Session

To end a CLI session and exit the CLI, type the **exit** command at the *local* Exec Mode prompt.

Accessing the CLI

Access the CLI through the following methods:

- Local login through a Console port using the RS-232 serial cable supplied with the card
- Remote login using Telnet and Secure Shell (SSH) access to the CLI through any IP interface on the system.

Important: Even though you can access the CLI remotely through any available IP interface, it is recommended that management traffic be isolated from network traffic by using one of the SPIO card management interfaces. You can use remote login methods only after the system has been configured to support the various access methods.

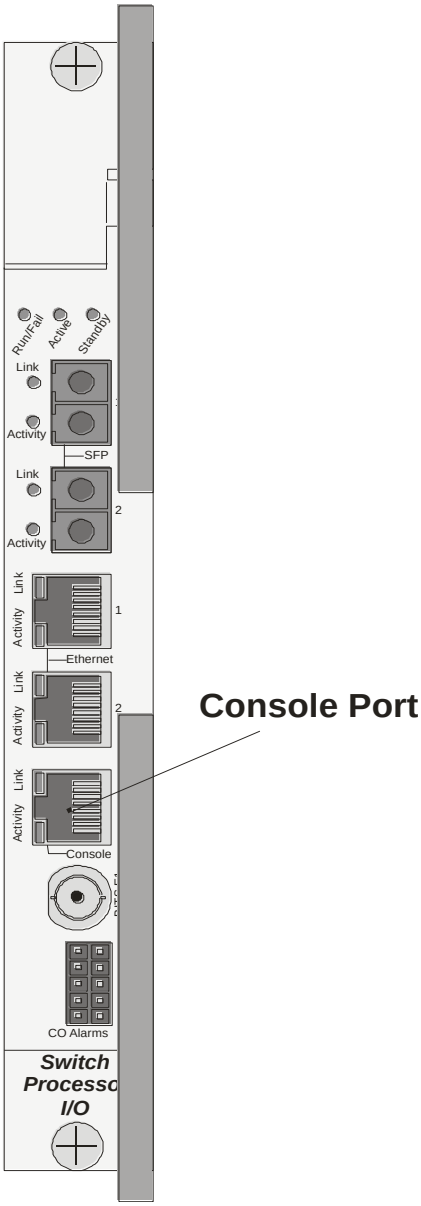
Important: Multiple CLI sessions are supported, but the number of sessions is dependent on the amount of available memory. The Resource Manager reserves enough resources so that as a minimum, seven CLI sessions are assured. One of the CLI sessions is always reserved for use exclusively by a CLI session on an SPIO console interface. Additional CLI sessions beyond the pre-reserved set are permitted if sufficient SPC resources are available. If the Resource Manager is unable to reserve additional resources, you are prompted whether to allow the system to create the new CLI session, even without the reserved resources.

Important: Multiple CLI sessions are supported, but the number of sessions is dependent on the amount of available memory. The Resource Manager reserves enough resources so that as a minimum, 15 CLI sessions are assured. One of the CLI sessions is always reserved for use exclusively by a CLI session on an SPIO console interface. Additional CLI sessions beyond the pre-reserved set are permitted if sufficient SMC resources are available. If the Resource Manager is unable to reserve additional resources, you are prompted whether to allow the system to create the new CLI session, even without the reserved resources.

Accessing the CLI Locally Using the Console Port

This section provides instructions for accessing the CLI locally through the console port.

Figure 3. Console Port



Access the console port with the RJ-45-to-DB-9 serial (EIA-232) cable that is shipped with the Switch Processor Input/Output (SPIO)Connect to a workstation that has a communications application that accesses the workstation's serial port, such as Minicom for Linux or HyperTerminal® for MicroSoft Windows®.

Each of the two SPIO Line Cards installed in the system provides a console port for accessing the CLI. The CLI is only accessible from the SPIO that is active—typically the SPIO installed in chassis slot 24.

For normal operation, the SPC or SMC in chassis slot 8 serves as the active processing card for the system. The SPIO that corresponds to this SPC or SMC is installed in slot 24. For the processing card in chassis slot 9, the corresponding SPIO is installed in slot 25.

Important: In the event of an SPC or SMC switchover, in which processes are switched from the processing card in slot 8 that was previously active to the redundant processing card in slot 9, the SPIO in slot 24 continues to serve as the active SPIO. Therefore, the console port is still accessible through that SPIO.

Follow the instructions below to connect to the console port.

1. Connect the RJ-45 end of the cable to the port labeled *Console*.
2. Connect the DB-9 end of the cable to the serial port on the workstation.
3. Configure the communications application to support the following:

Parameter	Setting
Baud Rate	115,200 bps
Data Bits	8
Parity	None
Stop Bits	1
Flow Control	None

Important: To change the configuration defined in the table above, modify the **`terminal`** command located in the Global configuration mode.

4. At the terminal window, press **Enter**.
5. If no configuration file is present (that is, this is the first time the system is powered), the CLI prompts you as to whether or not you want to use the Quick Setup Wizard. If the system was configured previously, you are prompted to enter a username and password.

Remotely Accessing the CLI

To remotely access the CLI through a defined management interface, you must first configure the remote access method (such as Telnet or SSH).

You can find examples of how to configure this in the *Getting Started* chapter.

Chapter 2
Enhanced Feature Overview

This reference describes the procedures to enable and configure the enhanced services on your system for core network service subscribers in a wireless carrier network. Procedures to configure and administer core network services are described in detail in the respective product administration guide.

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.

Important: All features described in this book are license enabled, unless otherwise shown in the table below. If you have not previously purchased a license bundle(s) or a standalone license for any license-enabled enhanced feature, contact your sales representative for more information.

This chapter provides information on required licenses and network services supported by each feature.

Supported Products and Licensing

The following table provides the list of supported products and license name of enhanced features described in this reference.

Table 6. Supported Products and Licenses

Enhanced Feature Name	Product	Platform	License Name
Always-on	PDSN HA	ST16 ST40	3GPP2 Always-On RP Extensions
BCMCS	PDSN	ST16 ST40	Broadcast & Multicast Services
CoA, RADIUS DM, and Session Redirection (Hotlining)	PDSN GGSN IPSG ASN GW HA PDIF	ST16 ST40	Dynamic Radius extensions (CoA and PoD)
Content Service Steering	PDSN GGSN HA	ST16 ST40	External Service Steering
Content Filtering ICAP Interface Support	GGSN	ST16 ST40	Content Filtering ICAP Interface
Direct Tunnel	SGSN	ST40	SGSN Software License
Dynamic QoS Renegotiation (Traffic Class-based QoS and Network Controlled QoS)	GGSN	ST16 ST40	GGSN Dynamic QoS Renegotiation
GRE Protocol Interface Support	GGSN	ST40	GRE Interface Tunneling
Gx Interface Support	GGSN IPSG HSGW P-GW	ST16 ST40	Dynamic Policy Interface
HA Proxy DNS Intercept	HA	ST16 ST40	HA DNS Intercept Proxy
HA Redundancy for Dynamic Home Agent Assignment	HA	ST16 ST40	--
Intelligent Traffic Control	ASN GW PDSN HA HSGW P-GW S-GW	ST16 ST40	Intelligent Traffic control

Enhanced Feature Name	Product	Platform	License Name
Interchassis Session Recovery	GGSN SCM HA	ST16 ST40	Inter-Chassis Session Recovery
IP Access Control Lists	ASN GW GGSN HA HSGW IPSG MME PDIF PDSN P-GW SCM S-GW SGSN	ST16 ST40	--
IP Header Compression	HSGW PDSN	ST16 ST40	Robust Header Compression
IP Pool Sharing Protocol	PDSN GGSN	ST16 ST40	--
IP Security	ASN GW GGSN HA HSGW IPSG MME PDIF PDSN P-GW SCM S-GW	ST16 ST40	IPSec
Lawful Intercept	PDSN GGSN ASN GW HA MME PDIF LNS SGSN	ST16 ST40	Lawful Intercept/ Enhanced Lawful Intercept
L2TP Access Concentrator	PDSN GGSN IPSG ASN GW	ST16 ST40	L2TP LAC
L2TP Network Server	PDSN/LNS GGSN/LNS	ST16 ST40	L2TP LNS
MIP NAT Traversal	HA	ST16 ST40	MIP NAT Traversal

Enhanced Feature Name	Product	Platform	License Name
Mobile IP Registration Revocation	ASN GW/FA GGSN/FA HA HSGW IPSG PDIF PDSN/FA P-GW SCM S-GW	ST16 ST40	--
Multi Protocol Label Switching (MPLS)	GGSN	ST16 ST40	MPLS
Multimedia Broadcast and Multicast Service	GGSN	ST16 ST40	Multimedia Broadcast & Multicast Service
MSID and PCF Zone Based Call Redirection	HA	ST16 ST40	PDSN RAN Optimization, Bundle 1
Policy-based Management and EV-DO Rev A	PDSN/FA	ST16 ST40	EV-DO Rev A/PDSN
Policy Forwarding	PDSN GGSN ASN GW IPSG SCM PDIF	ST16 ST40	--
Pre-paid Billing	PDSN HA	ST16 ST40	Prepaid Accounting/IS-835C Prepaid Bundle
Proxy-Mobile IP	ASN GW/FA GGSN/FA HSGW IPSG PDIF PDSN/FA P-GW S-GW	ST16 ST40	Proxy MIP
Rejection/Redirection of HA Sessions on Network Failures	PDSN GGSN IPSG ASN GW SCM HA	ST16 ST40	Combo Phone Bundle

Enhanced Feature Name	Product	Platform	License Name
Remote Address-based RADIUS Accounting	PDSN GGSN IPSG ASN GW PDIF HA	ST16 ST40	Destination Based Accounting
Routing	ASN GW GGSN HA HSGW IPSG MME PDSN P-GW SCM S-GW	ST16 ST40	IPv4 Routing Protocols
Session Recovery	ASN GW GGSN HA HSGW IPSG MME PDIF PDSN P-GW SCM SGSN S-GW	ST16 ST40	Session Recovery
Subscriber Overcharging Protection	GGSN SGSN	ST40	UMTS Overcharging Protection
Traffic Policing and Shaping	ASN GW GGSN HSGW HA PDSN P-GW SCM S-GW	ST16 ST40	Per Subscriber Traffic Policing/Shaping
Ty Interface Support	PDSN HA	ST16 ST40	Dynamic Policy Interface

Enhanced Feature Name	Product	Platform	License Name
VLANs	ASN GW GGSN HA HSGW IPSG MME PDIF PDSN P-GW SCM SGSN S-GW	ST16 ST40	--

Chapter 3
Verifying and Saving Your Configuration

This chapter describes how to save the system configuration.

Verifying the Configuration

You can use a number of command to verify the configuration of your feature, service, or system. Many are hierarchical in their implementation and some are specific to portions of or specific lines in the configuration file.

Feature Configuration

In many configurations, specific features are set and need to be verified. Examples include APN and IP address pool configuration. Using these examples, enter the following commands to verify proper feature configuration:

show apn all

The output displays the complete configuration for the APN. In this example, an APN called apn1 is configured.

```
access point name (APN): apn1

authentication context: test

pdp type: ipv4

Selection Mode: subscribed

ip source violation: Checked drop limit: 10

accounting mode: gtpp No early PDUs: Disabled

max-primary-pdp-contexts: 1000000 total-pdp-contexts: 1000000

primary contexts: not available total contexts: not available

local ip: 0.0.0.0

primary dns: 0.0.0.0 secondary dns: 0.0.0.0

ppp keep alive period : 0 ppp mtu : 1500

absolute timeout : 0 idle timeout : 0

long duration timeout: 0 long duration action: Detection

ip header compression: vj

data compression: stac mppc deflate compression mode: normal

min compression size: 128

ip output access-group: ip input access-group:

ppp authentication:

allow noauthentication: Enabled imsi
```

```
 authentication:Disabled
```

Enter the following command to display the IP address pool configuration:

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.

```
context : isp1:

+-----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 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
```

Important: Many features can be configured on the system. There are show commands specifically for these features. Refer to the *Command Line Interface Reference* for more information.

Service Configuration

Verify that your service was created and configured properly by entering the following command:

show *<service_type> <service_name>*

The output is a concise listing of the service parameter settings similar to the sample displayed below. In this example, a P-GW service called pgw is configured.

```
Service name : pgw1

Service-Id : 1
```

```
Context : test1

Status : STARTED

Restart Counter : 8

EGTP Service : egtp1

LMA Service : Not defined

Session-Delete-Delay Timer : Enabled

Session-Delete-Delay timeout : 10000(msecs)

PLMN ID List : MCC: 100, MNC: 99

Newcall Policy : None
```

Context Configuration

Verify that your context was created and configured properly by entering the following command:

show context name *<name>*

The output shows the active context. Its ID is similar to the sample displayed below. In this example, a context named *test1* is configured.

```
Context Name     ContextID     State

------------     ---------     -----

test1            2          Active
```

System Configuration

Verify that your entire configuration file was created and configured properly by entering the following command:

show configuration

This command displays the entire configuration including the context and service configurations defined above.

Finding Configuration Errors

Identify errors in your configuration file by entering the following command:

show configuration errors

This command displays errors it finds within the configuration. For example, if you have created a service named "service1", but entered it as "srv1" in another part of the configuration, the system displays this error.

You must refine this command to specify particular sections of the configuration. Add the **section** keyword and choose a section from the help menu:

show configuration errors section ggsn-service

or

show configuration errors section aaa-config

If the configuration contains no errors, an output similar to the following is displayed:

```
##################################################################################

Displaying Global
AAA-configuration errors
##################################################################################

Total 0 error(s) in this section !
```

Saving the Configuration

Save system configuration information to a file locally or to a remote node on the network. You can use this configuration file on any other systems that require the same configuration.

Files saved locally can be stored in the SPC's/SMC's CompactFlash or on an installed PCMCIA memory card on the SPC/SMC. Files that are saved to a remote network node can be transmitted using either FTP, or TFTP.

Saving the Configuration on the Chassis

These instructions assume that you are at the root prompt for the Exec mode:

```
[local]host_name#
```

To save your current configuration, enter the following command:

```
save configuration url [-redundant] [-noconfirm] [showsecrets] [verbose]
```

Keyword/Variable	Description
url	Specifies the path and name to which the configuration file is to be stored. *url* may refer to a local or a remote file. *url* must be entered using one of the following formats: • **{ /flash | /pcmcia1 | /pcmcia2 }** *[/dir] /file_name* • **file:/{ /flash | /pcmcia1 | /pcmcia2 }** *[/dir] /file_name* • **tftp://**{ *ipaddress*|*host_name*[*:port#*]} [*/directory*] */file_name* • **ftp://**[*username*[*:pwd*] *@*] { *ipaddress*|*host_name*} [*:port#*] [*/directory*] */file_name* • **sftp://**[*username*[:*pwd*] *@*] { *ipaddress*|*host_name*} [*:port#*] [*/directory*] */file_name* **/flash** corresponds to the CompactFlash on the SPC/SMC. **/pcmcia1** corresponds to PCMCIA slot 1. **/pcmcia2** corresponds to PCMCIA slot 2. *ipaddress* is the IP address of the network server. *host_name* is the network server's *hostname*. *port#* is the network server's logical port number. Defaults are: • tftp: 69 - data • ftp: 20 - data, 21 - control • sftp: 115 - data Note: *host_name* can only be used if the **networkconfig** parameter is configured for DHCP and the DHCP server returns a valid nameserv er.dx *username* is the username required to gain access to the server if necessary. *password* is the password for the specified username if required. */directory* specifies the directory where the file is located if one exists. */file_name* specifies the name of the configuration file to be saved. Note: Configuration files should be named with a .cfg extension.
-redundant	Optional: This keyword directs the system to save the CLI configuration file to the local device, defined by the url variable, and then automatically copy that same file to the like device on the Standby SPC/SMC, if available. Note: This keyword will only work for like local devices that are located on both the active and standby SPCs/SMCs. For example, if you save the file to the /pcmcia1 device on the active SPC/SMC, that same type of device (a PC-Card in Slot 1 of the standby SPC/SMC) must be available. Otherwise, a failure message is displayed. Note: If saving the file to an external network (non-local) device, the system disregards this keyword.

Keyword/Variable	Description
-noconfirm	Optional: Indicates that no confirmation is to be given prior to saving the configuration information to the specified filename (if one was specified) or to the currently active configuration file (if none was specified).
showsecrets	Optional: This keyword causes the CLI configuration file to be saved with all passwords in plain text, rather than their default encrypted format.
verbose	Optional: Specifies that every parameter that is being saved to the new configuration file should be displayed.

Important: The **`-redundant`** keyword is only applicable when saving a configuration file to local devices.This command does not synchronize the local file system. If you have added, modified, or deleted other files or directories to or from a local device for the active SPC/SMC, then you must synchronize the local file system on both SPCs/SMCs.

To save a configuration file called system.cfg to a directory that was previously created called cfgfiles on the SPC's/SMC's CompactFlash, enter the following command:

```
save configuration /flash/cfgfiles/system.cfg
```

To save a configuration file called simple_ip.cfg to a directory called host_name_configs using an FTP server with an IP address of 192.168.34.156 on which you have an account with a username of administrator and a password of secure, use the following command:

```
save configuration
ftp://administrator:secure@192.168.34.156/host_name_configs/
simple_ip.cfg
```

To save a configuration file called init_config.cfg to the root directory of a TFTP server with a hostname of config_server, enter the following command:

`save configuration` *`tftp://config_server/init_config.cfg`*

Chapter 4
Access Control Lists

This chapter describes system support for access control lists and explains how they are 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.

Important: You do not require a license to configure ACLs; however, the number of ACLs configured might impact performance significantly.

Important: Not all commands and keywords/variables may be available. This is dependent on the platform type.

This chapter contains the following sections:

Overview

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

Understanding ACLs

This section discusses the two main aspects to ACLs on the system:

- Rule(s)
- Rule Order

Important: Refer to the *ACL Configuration Mode Commands* chapter of the *Command Line Interface Reference* for the full command syntax.

Rule(s)

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.

Important: Configured ACLs consisting of no rules imply a “deny any” rule. The **deny** action and **any** criteria are discussed later in this section. This is the default behavior for an empty 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.

Actions

ACLs specify that one of the following actions can be taken on a packet that matches the specified criteria:

- **Permit**: The packet is accepted and processed.
- **Deny**: The packet is rejected.
- **Redirect**: The packet is forwarded to the specified next-hop address through a specific system interface or to the specified context for processing.

  Important: Redirect rules are ignored for ACLs applied to specific subscribers or all subscribers facilitated by a specific context, or APN for UMTS subscribers.

Criteria

Each ACL consists of one or more rules specifying the criteria that packets will be compared against.

The following criteria are supported:

- **Any**: Filters all packets
- **Host**: Filters packets based on the source host IP address
- **ICMP**: Filters Internet Control Message Protocol (ICMP) packets
- **IP**: Filters Internet Protocol (IP) packets
- **Source IP Address**: Filter packets based on one or more source IP addresses
- **TCP**: Filters Transport Control Protocol (TCP) packets
- **UDP**: Filters User Datagram Protocol (UDP) packets

Each of the above criteria are described in detail in the sections that follow.

Important: The following sections contain basic ACL rule syntax information. Refer to the ACL Configuration Mode Commands chapter of the Command Line Interface Reference for the full command syntax.

- **Any**: The rule applies to all packets.
- **Host**: The rule applies to a specific host as determined by its IP address.
- **ICMP**: The rule applies to specific Internet Control Message Protocol (ICMP) packets, Types, or Codes.

  Important: ICMP type and code definitions can be found at *www.iana.org* as indicated by RFC 3232.

- **IP**: The rule applies to specific Internet Protocol (IP) packets or fragments.
- **IP Packet Size Identification Algorithm**: The rule applies to specific Internet Protocol (IP) packets identification for fragmentation during forwarding.

This configuration is related to the “IP Identification field” assignment algorithm used by the system, when subscriber packets are being encapsulated (such as Mobile IP and other tunneling encapsulation). Within the system, subscriber packet encapsulation is done in a distributed way and a 16 bit IP identification space is divided and distributed to each entity which does the encapsulation, so that unique IP identification value can be assigned for IP headers during encapsulation.

Since this distributed IP Identification space is small, a non-zero unique identification will be assigned only for those packets, which may potentially be fragmented during forwarding (since the IP identification field is only used for reassembly of the fragmented packet). The total size of the IP packet is used to determine the possibility of that packet getting fragmented.

- **Source IP Address**: The rule applies to specific packets originating from a specific source address or a group of source addresses.

- **TCP**: The rule applies to any Transport Control Protocol (TCP) traffic and could be filtered on any combination of source/destination IP addresses, a specific port number, or a group of port numbers.

  Important: TCP port numbers definitions can be found at *www.iana.org*.

- **UDP**: The rule applies to any User Datagram Protocol (UDP) traffic and could be filtered on any combination of source/destination IP addresses, a specific port number, or a group of port numbers.

  Important: UDP port numbers definitions can be found at *www.iana.org*.

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.

Additional rules can be added to an existing ACL and properly ordered using either of the following options:

- Before
- After

Using these placement options requires the specification of an existing rule in the ACL and the configuration of the new rule as demonstrated by the following flow:

[before | after] { *<existing_rule>* }

Configuring ACLs on the System

This section provides information and instructions for configuring ACLs.

Important: This section provides the minimum instruction set for configuring access control list on the system. For more information on commands that configure additional parameters and options, refer *ACL 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 access control list by applying the example configuration in the Creating ACLs section.

Step 2 Specify the rules and criteria for action in ACL list by applying the example configuration in the Configuring Action and Criteria for Subscriber Traffic section.

Step 3 *Optional.* The system provides an "undefined" ACL that acts as a default filter for all packets into the context. The default action is to "permit all". Modify default configuration for "unidentified" ACLs for by applying the example configuration in the Configuring an Undefined ACL section.

Step 4 Verify your ACL configuration by following the steps in the Verifying the ACL Configuration section.

Step 5 Save your configuration as described in the *Verifying and Saving Your Configuration* chapter.

Creating ACLs

To create an ACL, use the following configuration:

```
configure
   context <acl_ctxt_name> [ -noconfirm ]
      ip access-list <acl_list_name>
         end
```

Notes:

- The maximum number of ACLs that can be configured per context is limited by the amount of available memory in the VPN Manager software task. Typically, the maximum is less than 200.

Configuring Action and Criteria for Subscriber Traffic

To create rules to deny/permit the subscriber traffic and apply the rules after or before action, use the following configuration:

```
configure
   context <acl_ctxt_name>  -noconfirm
      ip access-list <acl_list_name>
         deny { <ip_address> | any | host | icmp | ip | log | tcp | udp }
         permit { <ip_address> | any | host | icmp | ip | log | tcp | udp }
         after { deny | permit | readdress | redirect }
         before { deny | permit | readdress | redirect }
         end
```

Notes:

- Use the information provided in the Actions and Criteria sections of this chapter to configure the rules that comprise the ACL. For more information, refer *ACL Configuration Mode Commands* in *Command Line Interface Reference*.
- The maximum number of rules that can be configured per ACL varies depending on how the ACL is to be used. For more information, refer *Engineering Rules* in *System Administration Guide*.

Caution: Unless configured to do otherwise, the system implicitly adds a "deny any" rule to the end of the ACL resulting in the packet being dropped if it does not match any other configured rule. This behavior can be changed by adding a "permit any" rule as the last rule in the ACL.

Configuring an "Undefined" ACL

As discussed previously in this chapter the system uses an "undefined" ACL mechanism for filtering the packet(s) in the event that an ACL that has been applied is not present. This scenario is likely the result of a mis-configuration such as the ACL name being mis-typed during the configuration process.

For these scenarios, the system provides an "undefined" ACL that acts as a default filter for all packets into the context. The default action is to "permit all".

To modify the default behavior for unidentified ACLs, use the following configuration:

```
configure
   context <acl_ctxt_name> -noconfirm
      access-list undefined { deny-all | permit-all }
      end
```

Notes:

- Context name is the name of the context containing the “undefined” ACL to be modified. For more information, refer *Context Configuration Mode Commands* in *Command Line Interface Reference*.

Verifying the ACL Configuration

To verify the ACL configuration:

Step 1 In the Exec Mode, enter the following command:

`show ip access-list`

The following is a sample output of this command. In this example, an ACL named *`acl_1`* was configured.

```
ip access list acl_1
   deny host 1.2.3.4
   deny ip any host 1.2.3.4
   permit any 1.2.4.4
1 ip access-lists are configured.
```

Applying IP ACLs

Once an ACL is configured, it must be applied to take effect.

As discussed earlier, an ACL can be applied to any of the following:

- Applying an ACL to an Individual Interface
- Applying an ACL to All Traffic Within a Context (known as a policy ACL)
- Applying an ACL to an Individual Subscriber
- Applying a Single ACL to Multiple Subscribers
- Applying a Single ACL to Multiple Subscribers via APNs (for 3GPP subscribers only)

Important: ACLs must be configured in the same context in which the subscribers and/or interfaces to which they are to be applied. Similarly, ACLs to be applied to a context must be configured in that context.

If ACLs are applied at multiple levels within a single context (i.e. an ACL is applied to an interface within the context and another ACL is applied to the entire context), they will be processed as shown in the following figure and table.

Figure 4. ACL Processing Order

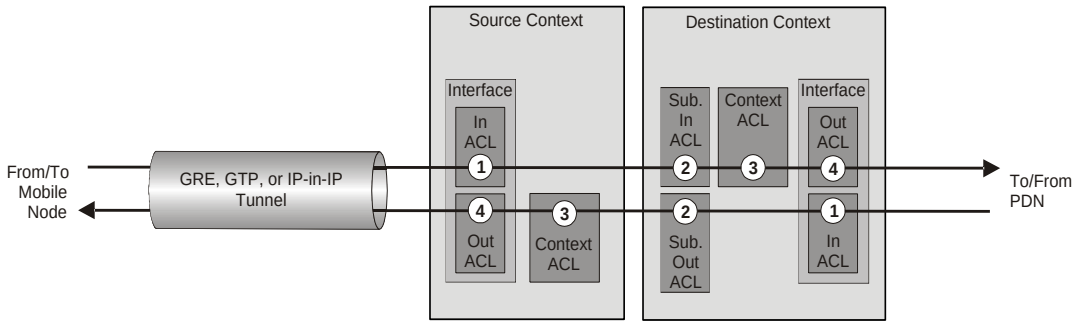


Table 7. ACL Processing Order Descriptions

Packet coming from the mobile node to the packet data network (left to right)	
Order	Description
1	An inbound ACL configured for the receiving interface in the Source Context is applied to the tunneled data (i.e. the outer IP header). The packet is then forwarded to the Destination Context.
2	An inbound ACL configured for the subscriber (either the specific subscriber or for any subscriber facilitated by the context) is applied.
3	A context ACL (policy ACL) configured in the Destination Context is applied prior to forwarding.
4	An outbound ACL configured on the interface in the Destination Context through which the packet is being forwarded is applied.

Packet coming from the packet data network to the mobile node (right to left)	
Order	Description
1	An inbound ACL configured for the receiving interface configured in the Destination Context is applied.
2	An outbound ACL configured for the subscriber (either the specific subscriber or for any subscriber facilitated by the context) is applied. The packet is then forwarded to the Source Context.
3	A context ACL (policy ACL) configured in the Source Context is applied prior to forwarding.
4	An outbound ACL configured on the interface in the Source Context through which the packet is being forwarded is applied to the tunneled data (i.e. the outer IP header).

In the event that an IP ACL is applied that has not been configured (i.e. the name of the applied ACL was configured incorrectly), the system uses an "undefined" ACL mechanism for filtering the packet(s).

This section provides information and instructions for applying ACLs and for configuring an "undefined" ACL.

Applying an ACL to an Individual Interface

This section provides information and instructions for applying one or more ACLs to an individual interface configured on the system.

Important: It is recommended that all ACLs be configured and verified according to the instructions in the Configuring ACLs on the System section of this chapter prior to beginning this procedure.

Important: This section provides the minimum instruction set for applying the ACL list to an interface on the system. For more information on commands that configure additional parameters and options, refer *Ethernet Interface Configuration Mode Commands* chapter in *Command Line Interface Reference*.

To configure the system to provide ACL facility to subscribers:

Step 1 Apply the configured access control list by applying the example configuration in the Applying ACL to Interface section.

Step 2 Verify that ACL is applied properly on interface by following the steps in the Verifying the ACL Configuration on Interface section.

Step 3 Save your configuration as described in the *Verifying and Saving Your Configuration* chapter.

Applying ACL to Interface

To apply the ACL to an interface, use the following configuration:

```
configure
```

```
context <acl_ctxt_name> -noconfirm
   interface <interface_name>
      ip access-group <acl_list_name> { in | out } [ <preference> ]
      end
```

Notes:

- The context name is the name of the ACL context containing the interface to which the ACL is to be applied.
- The ACL to be applied must be configured in the context specified by this command.
- Up to 8 ACLs can be applied to a group provided that the number of rules configured within the ACL(s) does not exceed the 128 rule limit for the interface.

Verifying the ACL Configuration on Interface

This section describes how to verify the ACL configuration.

Step 1 In the Exec Mode, enter the following command:

show configuration context *context_name*

context_name is the name of the context containing the interface to which the ACL(s) was/were applied.

The output of this command displays the configuration of the entire context. Examine the output for the commands pertaining to interface configuration. The commands display the ACL(s) applied using this procedure.

```
configure
   context context_name
      ip access-list acl_name
         deny host ip_address
         deny ip any host ip_address
         exit
      ip access-group access_group_name
      service-redundancy-protocol
         exit
      interface interface_name
         ip address ip_address/mask
         exit
```

```
subscriber default
   exit
aaa group default
   exit
gtpp group default
   end
```

Applying an ACL to All Traffic Within a Context

This section provides information and instructions for applying one or more ACLs to a context configured within a specific context on the system. The applied ACLs, known as policy ACLs, contain rules that apply to all traffic facilitated by the context.

Important: It is recommended that all ACLs be configured and verified according to the instructions in the Configuring ACLs on the System section of this chapter prior to beginning this procedure.

Important: This section provides the minimum instruction set for applying the ACL list to all traffic within a context. For more information on commands that configure additional parameters and options, refer *Context Configuration Mode Commands* chapter in *Command Line Interface Reference*.

To configure the system to provide access control list facility to subscribers:

Step 1 Apply the configured ACL as described in the Applying ACL to Context section.

Step 2 Verify that ACL is applied properly on interface as described in the Verifying the ACL Configuration in a Context section.

Step 3 Save your configuration as described in the *Verifying and Saving Your Configuration* chapter.

Applying ACL to Context

To apply the ACLs to a context, use the following configuration:

```
configure
   context <acl_ctxt_name> [ -noconfirm ]
      ip access-group <acl_list_name> [ in | out ] [ <preference> ]
      end
```

Notes:

- The context name is the name of the ACL context containing the interface to which the ACL is to be applied.
- The context-level ACL are applied only to outgoing packets. The **in** and **out** keywords are deprecated and are only present for backward compatibility.
- The ACL to be applied must be configured in the context specified by this command.
- Up to 8 ACLs can be applied to a group provided that the number of rules configured within the ACL(s) does not exceed the 128 rule limit for the interface.

Verifying the ACL Configuration in a Context

To verify the ACL configuration:

Step 1 Verify that your ACL lists were applied properly by entering the following command in Exec Mode:

show configuration context *context_name*

context_name is the name of the context to which the ACL(s) was/were applied.

The output of this command displays the configuration of the entire context. Examine the output for the commands pertaining to interface configuration. The commands display the ACL(s) applied using this procedure.

```
configure
   context context_name
      ip access-list acl_name
         deny host ip_address
         deny ip any host ip_address
         exit
      ip access-group access_group_name
      service-redundancy-protocol
         exit
      interface interface_name
         ip address ip_address/mask
         exit
      subscriber default
         exit
```

```
aaa group default
   exit
gtpp group default
   end
```

Applying an ACL to an Individual Subscriber

IP ACLs are applied to subscribers via attributes in their profile. The subscriber profile could be configured locally on the system or remotely on a RADIUS server.

To apply an ACL to a RADIUS-based subscriber, use the **`Filter-Id`** attribute. Refer to the *AAA Interface Administration and Reference* for more detail on this attribute.

This section provides information and instructions for applying an ACL to an individual subscriber whose profile is configured locally on the system.

Important: It is recommended that all ACLs be configured and verified according to the instructions in the Configuring ACLs on the System section of this chapter prior to beginning this procedure. Additionally, it is assumed that the subscribers have been previously configured.

Important: This section provides the minimum instruction set for applying the ACL list to all traffic within a context. 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 provide access control list facility to subscribers:

Step 1 Apply the configured access control list by applying the example configuration in the Applying ACL to an Individual Subscriber section.

Step 2 Verify that ACL is applied properly on interface by following the steps in the Verifying the ACL Configuration to an Individual Subscriber section.

Step 3 Save your configuration as described in the *Verifying and Saving Your Configuration* chapter.

Applying ACL to an Individual Subscriber

To apply the ACL to an individual subscriber, use the following configuration:

```
configure
   context <acl_ctxt_name> -noconfirm
      subscriber name <subs_name>
```

```
ip access-group <acl_list_name> [ in | out ]
end
```

Notes:

- The context name is the name of the ACL context containing the interface to which the ACL is to be applied.
- If neither the **in** nor the **out** keyword is specified, the ACL will be applied to all packets in and out.
- The ACL to be applied must be configured in the context specified by this command.
- Up to 8 ACLs can be applied to a group provided that the number of rules configured within the ACL(s) does not exceed the 128 rule limit for the interface.

Verifying the ACL Configuration to an Individual Subscriber

These instructions are used to verify the ACL configuration.

Step 1 Verify that your ACL lists were applied properly by entering the following command in Exec Mode:

show configuration context *context_name*

context_name is the name of the context containing the subscriber *subs1* to which the ACL(s) was/were applied.

The output of this command displays the configuration of the entire context. Examine the output for the commands pertaining to interface configuration. The commands display the ACL(s) applied using this procedure.

```
configure
   context context_name
      ip access-list acl_name
         deny host ip_address
         deny ip any host ip_address
         exit
      ip access-group access_group_name
      service-redundancy-protocol
         exit
      interface interface
         ip address ip_address/mask
         exit
      subscriber default
```

```
   exit
subscriber name subscriber_name
   ip access-group access_group_name in
   ip access-group access_group_name  out
   exit
aaa group default
   exit
gtpp group default
   exit
content-filtering server-group cfsg_name
   response-timeout response_timeout
   connection retry-timeout retry_timeout
   end
```

Applying a Single ACL to Multiple Subscribers

As mentioned in the previous section, IP ACLs are applied to subscribers via attributes in their profile. The subscriber profile could be configured locally on the system or remotely on a RADIUS server.

The system provides for the configuration of subscriber functions that serve as default values when specific attributes are not contained in the individual subscriber's profile. The following table describes these functions.

Table 8. Functions Used to Provide "Default" Subscriber Attributes

Function	Description
Subscriber Named default	Within each context, the system creates a subscriber called default. The profile for the subscriber named default provides a configuration template of attribute values for subscribers authenticated in that context. Any subscriber attributes that are not included in a RADIUS-based subscriber profile is configured according to the values for those attributes as defined for the subscriber named default. **NOTE:** The profile for the subscriber named default is not used to provide missing information for subscribers configured locally.
`default subscriber` Command	This command in the PDSN, FA, and HA service Configuration modes specifies a profile from a subscriber named something other than default to use a configuration template of attribute values for subscribers authenticated in that context. This command allows multiple services to draw "default" subscriber information from multiple profiles.

When configured properly, the functions described in the table above could be used to apply an ACL to:

- All subscribers facilitated within a specific context by applying the ACL to the profile of the subscriber named *default*.
- All subscribers facilitated by specific services by applying the ACL to a subscriber profile and then using the default subscriber command to configure the service to use that subscriber as the "default" profile.

Applying an ACL to the Subscriber Named default

This section provides information and instructions for applying an ACL to the subscriber named *default*.

Important: It is recommended that all ACLs be configured and verified according to the instructions in the Configuring ACLs on the System section of this chapter prior to beginning this procedure.

Important: This section provides the minimum instruction set for applying the ACL list to all traffic within a context. 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 provide access control list facility to subscribers:

Step 1 Apply the configured access control list by applying the example configuration in the Applying an ACL to the Subscriber Named default section.

Step 2 Verify that ACL is applied properly on interface by following the steps in the Verifying the ACL Configuration to the Subscriber Named default section.

Step 3 Save your configuration as described in the *Verifying and Saving Your Configuration* chapter.

Applying an ACL to the Subscriber Named default

To example to apply the ACL to the default subscriber, use the following configuration:

```
configure
   context <acl_ctxt_name>  [ -noconfirm ]
      subscriber name <subs_name>
         ip access-group <acl_list_name> [ in | out ]
         end
```

Notes:

- The context name is the name of the ACL context containing the interface to which the ACL is to be applied.
- If neither the **in** nor the **out** keyword is specified, the ACL will be applied to all packets in and out.
- The ACL to be applied must be configured in the context specified by this command.
- Up to 8 ACLs can be applied to a group provided that the number of rules configured within the ACL(s) does not exceed the 128 rule limit for the interface.

Verifying the ACL Configuration to the Subscriber Named default

These instructions are used to verify the ACL configuration.

Step 1 Verify that your ACL lists were applied properly by entering the following command in Exec Mode:

show configuration context *context_name*

context_name is the name of the context containing the subscriber default to which the ACL(s) was/were applied.

The output of this command displays the configuration of the entire context. Examine the output for the commands pertaining to interface configuration. The commands display the ACL(s) applied using this procedure.

```
configure
   context context_name
      ip access-list acl_name
         deny host ip_address
         deny ip any host ip_address
         exit
      ip access-group access_group_name
      service-redundancy-protocol
         exit
      interface interface
         ip address ip_address/mask
         exit
      subscriber name default
         ip access-group access_group_name in
         ip access-group access_group_name out
         exit
      aaa group default
         exit
```

```
gtpp group default
   exit
content-filtering server-group cfsg_name
   response-timeout response_timeout
   connection retry-timeout retry_timeout
   end
```

Applying an ACL to Service-specified Default Subscribers

This section provides information and instructions for applying an ACL to the subscriber to be used as the "default" profile by various system services.

Important: It is recommended that all ACLs be configured and verified according to the instructions in the Configuring ACLs on the System section of this chapter prior to beginning this procedure. Additionally, it is assumed that the services and subscribers have been previously configured.

Important: This section provides the minimum instruction set for applying the ACL list to all traffic within a context. 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 provide access control list facility to subscribers:

Step 1 Apply the configured access control list by applying the example configuration in the Applying an ACL to Service-specified Default Subscriber section.

Step 2 Verify that ACL is applied properly on interface by following the steps in the Verifying the ACL Configuration to Service-specified Default Subscriber section.

Step 3 Save your configuration as described in the *Verifying and Saving Your Configuration* chapter.

Applying an ACL to Service-specified Default Subscriber

To apply the ACL to a service-specified Default subscriber, use the following configuration:

```
configure
   context <acl_ctxt_name> -noconfirm
      { pdsn-service | fa-service | ha-service } <service_name>
         default subscriber <svc_default_subs_name>
         exit
```

```
subscriber name <svc_default_subs_name>
   ip access-group <acl_list_name> [ in | out ]
   end
```

Notes:

- The context name is the name of the ACL context containing the interface to which the ACL is to be applied.
- If neither the **in** nor the **out** keyword is specified, the ACL will be applied to all packets in and out.
- The ACL to be applied must be configured in the context specified by this command.
- Up to 8 ACLs can be applied to a group provided that the number of rules configured within the ACL(s) does not exceed the 128 rule limit for the interface.

Verifying the ACL Configuration to Service-specified Default Subscriber

To verify the ACL configuration.

Step 1 Verify that your ACL lists were applied properly by entering the following command in Exec Mode:

show configuration context *context_name*

context_name is the name of the context containing the service *pdsn1* having default subscriber to which the ACL(s) was/were applied.

The output of this command displays the configuration of the entire context. Examine the output for the commands pertaining to interface configuration. The commands display the ACL(s) applied using this procedure.

```
configure
   context context_name
      ip access-list acl_name
         deny host ip_address
         deny ip any host ip_address
         exit
      ip access-group access_group_name
      interface interface
         ip address ip_address/mask
         exit
      subscriber default
         exit
```

```
subscriber name subscriber_name
   ip access-group access_group_name in
   ip access-group access_group_name out
   exit
pdsn-service service_name
   default subscriber subscriber_name
   end
```

Applying a Single ACL to Multiple Subscribers via APNs

As mentioned in the previous section, IP ACLs are applied to subscribers via attributes in their profile. The subscriber profile could be configured locally on the system or remotely on a RADIUS server.

To reduce configuration time, ACLs can alternatively be applied to APN templates for GGSN subscriber. When configured, any subscriber packets facilitated by the APN template would then have the associated ACL applied.

This section provides information and instructions for applying an ACL to an APN template.

Important: It is recommended that all ACLs be configured and verified according to the instructions in the Configuring ACLs on the System section of this chapter prior to beginning this procedure.

Important: This section provides the minimum instruction set for applying the ACL list to all traffic within a context. 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 provide access control list facility to subscribers:

Step 1 Apply the configured access control list by applying the example configuration in the Applying an ACL to Multiple Subscriber via APNs section.

Step 2 Verify that ACL is applied properly on interface by following the steps in the Verifying the ACL Configuration to APNs section.

Step 3 Save your configuration as described in the *Verifying and Saving Your Configuration* chapter.

Applying an ACL to Multiple Subscriber via APNs

To apply the ACL to multiple subscribers via APN, use the following configuration:

```
configure
   context <dest_context_name> -noconfirm
      apn <apn_name>
```

```
ip access-group <acl_list_name> [ in | out ]
end
```

Notes:

- The ACL to be applied must be in the destination context of the APN (which can be different from the context where the APN is configured).
- If either the **in** or **out** keyword is not specified, the command is added to the config file twice, once with **in** and once with **out**, and the ACL will be applied to all packets inbound and outbound.
- Up to 8 ACLs can be applied to a group provided that the number of rules configured within the ACL(s) does not exceed the 128 rule limit for the interface.

Verifying the ACL Configuration to APNs

To verify the ACL configuration:

Step 1 Verify that your ACL lists were applied properly by entering the following command in Exec Mode:

show configuration context *context_name*

context_name is the name of the context containing the APN *apn1* having *default* subscriber to which the ACL(s) was/were applied.

The output of this command displays the configuration of the entire context. Examine the output for the commands pertaining to interface configuration. The commands display the ACL(s) applied using this procedure.

```
configure
   context context_name
      ip access-list acl_name
         deny host ip_address
         deny ip any host ip_address
         exit
      ip access-group access_group_name
      interface interface
         ip address ip_adrress/mask
         exit
      subscriber default
         exit
      apn apn_name
```

```
ip access-group access_group_name in
ip access-group access_group_name out
end
```

Chapter 5
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 the respective product Administration Guide, before using the procedures in this chapter.

This chapter contains the following sections:

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, refer to *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 as described in the *Saving Your Configuration* chapter.

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.

Chapter 6
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 the respective product 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

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

To enable BCMCS on the system, you must install the Broadcast & Multicast Services license. For more information on installing licenses, refer to *Software Maintenance Operations* 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 *PDSN 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 as described in the *Verifying and Saving Your Configuration* chapter.

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 Interface Administration and Reference*.

Chapter 7
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 the respective product Administration Guide, before using the procedures in this chapter.

Important: Not all commands and keywords/variables are available or supported. 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.

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.

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 as described in the *Verifying and Saving Your Configuration* chapter.

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 <ip_address>
      end
```

Notes:

- *<context_name>* must be the name of the AAA context where you want to enable CoA and DM. The AAA context must have been configured as described in the *Configuring Context-Level AAA Functionality* section of the *AAA Interface Administration Guide*.
- 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. Use one of the following attributes to identify the system:

- 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 is 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 cause 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
  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 gtpp acct requests              0 Current gtpp acct requests
  0 Total gtpp acct cancelled             0 Total gtpp 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 gtpp charg                      0 Current gtpp charg

357 Total gtpp charg success              0 Total gtpp charg failure

  0 Total gtpp charg cancel               0 Total gtpp 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)

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.

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. 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 r**eaddress 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: 01ca11b1 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: 01ca11b1 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
```

Chapter 8
Congestion Control

This chapter describes the Congestion Control feature.

The following topics are covered in this chapter:

Overview

This section provides an overview of the Congestion Control feature.

Congestion Control monitors the system for conditions that could potentially degrade performance when the system is under heavy load. Typically, these conditions are temporary (for example, high CPU or memory utilization) and are quickly resolved. However, continuous or large numbers of these conditions within a specific time interval may have an impact the system's ability to service subscriber sessions. Congestion control helps identify such conditions and invokes policies for addressing the situation.

Congestion control operation is based on configuring the following:

- **Congestion Condition Thresholds:** Thresholds dictate the conditions for which congestion control is enabled and establishes limits for defining the state of the system (congested or clear). These thresholds function in a way similar to operation thresholds that are configured for the system as described in the *Thresholding Configuration Guide*. The primary difference is that when congestion thresholds are reached, a service congestion policy and an SNMP trap, starCongestion, are generated.

  A threshold tolerance dictates the percentage under the configured threshold that must be reached in order for the condition to be cleared. An SNMP trap, starCongestionClear, is then triggered.

  - **Port Utilization Thresholds**: If you set a port utilization threshold, when the average utilization of all ports in the system reaches the specified threshold, congestion control is enabled.
  - **Port-specific Thresholds**: If you set port-specific thresholds, when any individual port-specific threshold is reached, congestion control is enabled system-wide.
- **Service Congestion Policies:** Congestion policies are configurable for each service. These policies dictate how services respond when the system detects that a congestion condition threshold has been crossed.

Important: This section provides the minimum instruction set for configuring congestion control. Commands that configure additional interface or port properties are provided in the Subscriber Configuration Mode chapters of the Command Line Interface Reference.

Configuring Congestion Control

To configure Congestion Control functionality:

Step 1 Configure Congestion Control Threshold as described in the Configuring the Congestion Control Threshold section.

Step 2 Configure Service Congestion Policies as described in the Configuring Service Congestion Policies section.

Step 3 Enable Congestion Control Redirect Overload Policy as described in the Enabling Congestion Control Redirect Overload Policy section.

Step 4 Configure disconnecting subscribers based on call or inactivity time as described in the Disconnecting Subscribers Based on Call or Inactivity Time section.

Step 5 Save your configuration as described in the *Saving and Verifying Your Configuration* chapter.

Configuring the Congestion Control Threshold

To configure congestion control threshold, apply the following example configuration:

```
configure
   congestion-control threshold max-sessions-per-service-utilization <percent>
   congestion-control threshold tolerance <percent>
   end
```

Notes:

- There are several additional threshold parameters. See the "Global Configuration Mode" chapter of the *Command Line Interface Reference* for more information.
- The tolerance is the percentage under a configured threshold that dictates the point at which the condition is cleared.
- Repeat this configuration as needed for additional thresholds.

Configuring Service Congestion Policies

To create a congestion control policy, apply the following example configuration:

```
configure
   congestion-control policy <service> action { drop | none | redirect | reject
}
```

```
end
```

Notes:

- When the redirect action occurs for PDSN services, the PDSN responds to the PCF with a reply code of 136, "unknown PDSN address" along with the IP address of an alternate PDSN.
- Redirect is not available on the PDIF.
- The default action for the PDIF is "none."
- When the redirect action occurs for HA services, the system responds to the FA with a reply code of 136, "unknown home agent address".
- Redirect can not be used in conjunction with GGSN services.
- Redirect is not available for the LMA service.
- When setting the action to reject, the reply code is 130, "insufficient resources".
- For the GGSN, the reply code is 199, "no resources available".
- For the MME **`redirect`** is not available.

Enabling Congestion Control Redirect Overload Policy

To create a congestion control policy and configure a redirect overload policy for the service, apply the following example configuration:


Important: Redirect is not available on PDIF for this release.

```
configure
   congestion-control
   context <context_name>
      {service_configuration_mode}
         policy overload redirect address
         end
```

Notes:

- *Optional:* If the congestion control policy action was configured to **redirect**, then a redirect overload policy must be configured for the service(s) that are affected.
- There are several service configuration modes that you can configure. See the *Command Line Interface Reference* for a complete list of modes.
- You can set various options for redirection. See the *Command Line Interface Reference* for more information.

- Repeat this configuration example to configure overload policies for additional services configured in the same context.

Verify the Service Overload Policies

To verify that the service overload policies were properly configured, in the Exec Mode, enter the following command:

show *<service_type>* **name** *service_name*

This command lists the entire service configuration. Ensure that the information displayed for the "Overload Policy" is accurate.

Repeat this configuration example to configure additional services in other contexts.

Verify the Congestion Control Configuration

To verify Congestion Control Configuration, in the Exec Mode, enter the following command:

show congestion-control configuration

The following output is a concise listing of all threshold and policy configurations:

```
Congestion-control: enabled
Congestion-control threshold parameters
 system cpu utilization: 80%
 service control cpu utilization: 80%
 system memory utilization: 80%
 message queue utilization: 80%
 message queue wait time: 10 seconds
 port rx utilization: 80%
 port tx utilization: 80%
 license utilization: 100%
 max-session-per-service utilization: 100%
 tolerence limit: 10%
Congestion-control Policy
   pdsn-service: none
   hsgw-service: none
   ha-service: none
```

```
   lma-service: none
   ggsn-service: none

   lns-service: none
   cscf-service: reject
   pdif-service: none

   sgsn-service: none
   mme-service: drop
   asngw-service: none
   asnpc-service: none
   phsgw-service: none
   phspc-service: none
   mipv6ha-service: none
   sgw-service: none
   pgw-service: none
```

Disconnecting Subscribers Based on Call or Inactivity Time

During periods of heavy system load, it may be necessary to disconnect subscribers in order to maintain an acceptable level of system performance. You can establish thresholds to select subscribers to disconnect based on the length of time that a call has been connected or inactive.

To enable overload disconnect for the currently selected subscriber, use the following configuration example:

```
configure
   context <context_name>
      subscriber name <subscriber_name>
         default overload-disconnect threshold inactivity-time <dur_thresh>
         default overload-disconnect threshhold connect-time <dur_thresh>
         end
```

To disable the overload disconnect feature for this subscriber, use the following configuration example:

```
configure

   context <context_name>

      subscriber <subscriber_name>

      no overload-disconnect {[ threshold inactivity-time] | [ threshhold
connect-time]}

      end
```

Notes:

- **overload-disconnect** is not supported for the CSCF service.

Chapter 9
Content Service Steering

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: Internal CSS is a generic feature, if an ECSv2 license is installed on your system, internal CSS can be enabled. A separate license is not required to enable internal CSS.

This chapter contains the following topics:

Overview

Content Service Steering (CSS) directs selective subscriber traffic to In-line services internal to the system based on the content of the data presented by mobile subscribers. CSS is a broad term that includes features such as NAT, HTTP redirection, and DNS redirection.

CSS uses Access Control Lists (ACLs) to redirect selective subscriber traffic flows. ACLs control the flow of packets into and out of the system. ACLs consist of "rules" (ACL rules) or filters that control the action taken on packets matching the filter criteria.

ACLs are configurable on a per-context basis and applies to a subscriber through either a subscriber profile (for PDSN) or an APN profile (for GGSN) in the destination context.

Configuring Internal Content Service Steering

To configure and activate a single CSS service for redirecting all of a subscriber's IP traffic to an internal in-line service:

Step 1 Define an IP ACL as described in the Defining IP Access Lists for Internal CSS section.

Step 2 Optional: Apply an ACL to an individual subscriber as described in the Applying an ACL to an Individual Subscriber (Optional) section.

Step 3 Optional: Apply a single ACL to multiple subscribers as described in the Applying an ACL to Multiple Subscribers (Optional) section.

Step 4 Optional: Apply an ACL to multiple subscribers via APNs as described in the Applying an ACL to Multiple Subscribers via APNs (Optional) section.

Step 5 Save your configuration as described in the *Verifying and Saving Your Configuration* chapter.

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 or keywords/variables may be supported or available. This depends on the platform type and installed license(s).

Defining IP Access Lists for Internal CSS

IP ACLs specify what type of subscriber traffic and which direction (uplink, downlink, or both) traffic is redirected. The IP ACL must be specified in the context in which subscriber authentication is performed.

Caution: To minimize the risk of data loss, do not make configuration changes to ACLs while the system is facilitating subscriber sessions.

Use the following configuration example to define an IP ACL for internal CSS:

```
configure
   context <context_name>
      ip access-list <acl_name>
         redirect css service <service_name> <keywords> <options>
          end
```

Notes:

- *<service_name>* must be an ACS service name.
- For information on the keywords and options available with the **redirect css service** command, see the *ACL Configuration Mode Commands* chapter of the *Command Line Interface Reference*.
- For IPv6 ACLs, the same configurations must be done in the IPv6 ACL Configuration Mode. See the *IPv6 ACL Configuration Mode Commands* chapter of the *Command Line Interface Reference*.

Applying an ACL to an Individual Subscriber (Optional)

For information on how to apply an ACL to an individual subscriber, refer to the *Applying an ACL to an Individual Subscriber* section of the *IP Access Control Lists* chapter.

Applying an ACL to Multiple Subscribers (Optional)

IP ACLs are applied to subscribers via attributes in their profiles. The subscriber profile can be configured locally on the system or remotely on a RADIUS server.

The system provides for the configuration of subscriber functions that serve as default values when specific attributes are not contained in the individual subscriber's profile. When configured properly, the functions can be used to apply an ACL to:

- All subscribers facilitated within a specific context by applying the ACL to the profile of the subscriber named *default*.
- All subscribers facilitated by specific services by applying the ACL to a subscriber profile and then using the default subscriber command to configure the service to use that subscriber as the "default" profile.

Applying an ACL to the Subscriber Named default (Optional)

For information on how to apply an ACL to the *default subscriber*, refer to the *Applying an ACL to the Subscriber Named default* section of the IP *Access Control Lists* chapter.

Applying an ACL to Service-specified Default Subscribers (Optional)

For information on how to apply an ACL to the subscriber to be used as the "default" profile by various system services, refer to the *Applying an ACL to Service-specified Default Subscribers* section of the *IP Access Control Lists* chapter.

Applying an ACL to Multiple Subscribers via APNs (Optional)

This configuration is only applicable to GGSN.

IP ACLs are applied to subscribers via attributes in their profiles. The subscriber profile can be configured locally on the system or remotely on a RADIUS server.

To reduce configuration time, ACLs can alternatively be applied to APN templates. When configured, any subscriber packets facilitated by the APN template would then have the associated ACL applied.

For information on how to apply an ACL to multiple subscribers via APNs, refer to the *Applying a Single ACL to Multiple Subscribers via APNs* section the *IP Access Control Lists* chapter.

Chapter 10
Direct Tunnel

This chapter briefly describes the 3G UMTS **direct tunnel** feature, indicates how it is implemented on various systems (for example, the Serving GPRS Support Node (SGSN), the Gateway GPRS Support Node (GGSN), and the Home NodeB Gateway (HNB-GW), and provides feature configuration procedures. Direct tunnel is an enhanced feature, so some products may require a feature implementation license and all relevant products will require completion of basic service configuration. Refer to your product's administration guide for feature licensing and basic service configuration information.

Currently, the SGSN is the only product that enables configuration of this feature. All other products that support direct tunnel do so by default. It is the SGSN that determines if setup of a direct tunnel is to be allowed or disallowed.

After the feature overview description, this chapter provides configuration procedures for the following:

- Enabling and Disabling GTP-U Direct Tunnels
- Disabling or Enabling DT Access to Specific GGSN(s)
- Disabling or Enabling Direct Tunnels to Specific RNC(s)

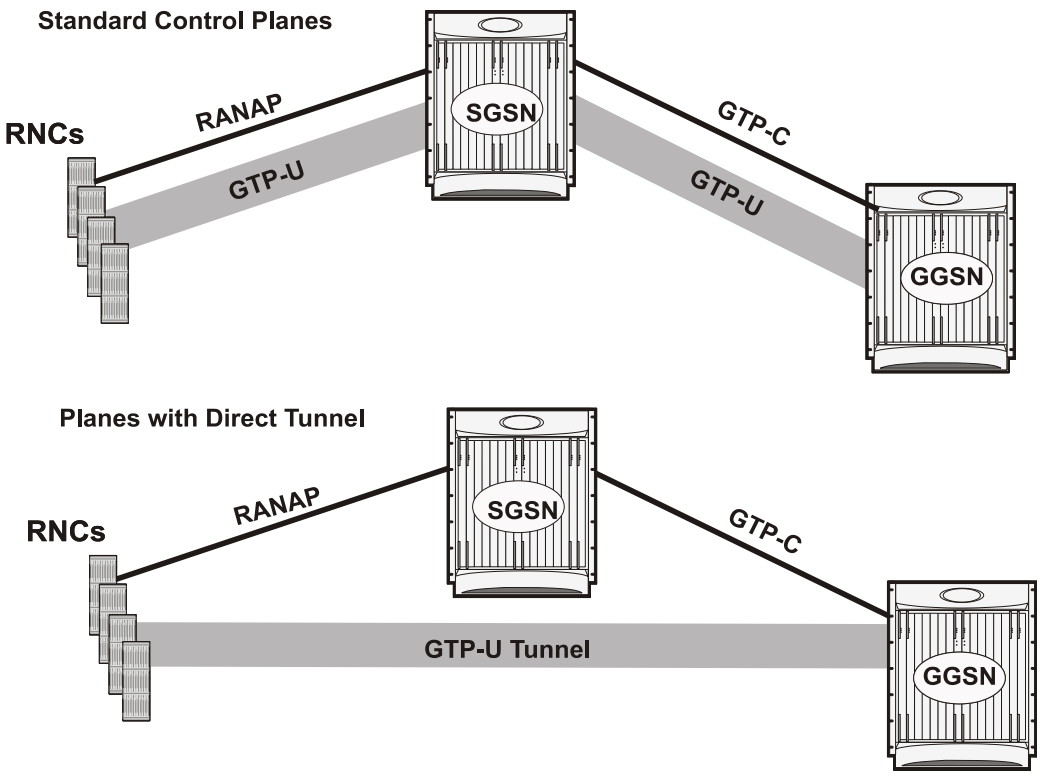
Important: This chapter provides a limited instruction set for configuring direct tunnel. Basic service configuration for the SGSN is provided in the *SGSN Service Configuration Procedures* chapter of the *SGSN Administration Guide*. Command details are provided in the *Command Line Interface Reference*.

Direct Tunnel Feature Overview

The direct tunnel architecture allows the SGSN to establish a direct *user plane* tunnel between the radio access network (RAN) and the GGSN. Once a direct tunnel is established the SGSN continues to handle the *control plane* signaling. This improves the user experience (e.g., expedites web page delivery, reduces round trip delay for conversational services). Additionally, direct tunnel functionality implements the standard *SGSN optimization* to improve the usage of user plane resources (and hardware) by removing the requirement from the SGSN to handle the user plane processing.

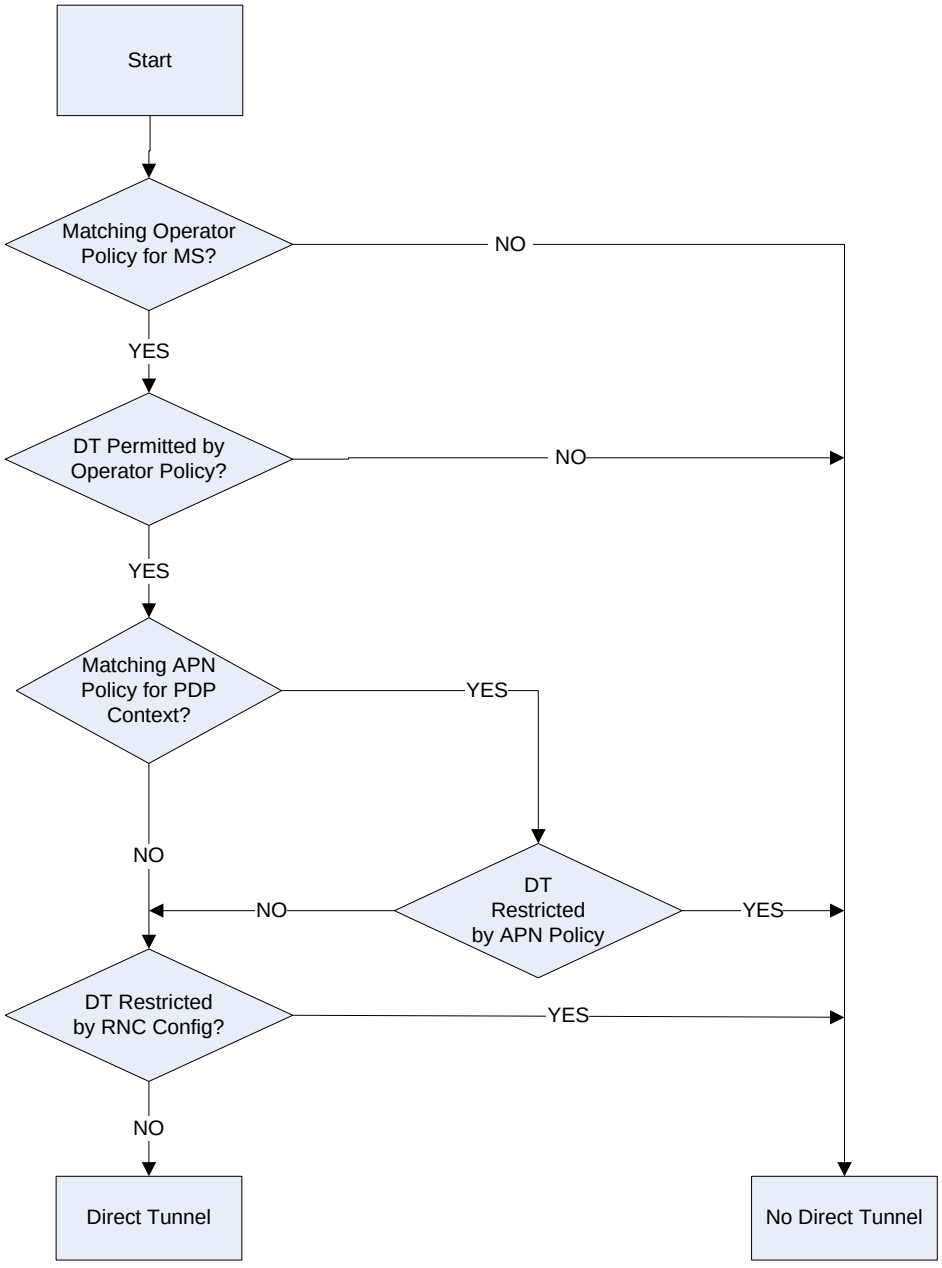
Typically, the SGSN sets up a direct tunnel at PDP context activation. The SGSN uses an Update PDP Context Request towards the GGSN to establish a GTP user plane (GTP-U) tunnel directly between the RNC and the GGSN. This means a significant increase in control plane load on both the SGSN and GGSN components of the packet core. Hence, deployment requires highly scalable GGSNs since the volume and frequency of Update PDP Context messages to the GGSN will increase substantially. The SGSN's platform capabilities ensure control plane capacity will not be a limiting factor with direct tunnel deployment.

Figure 5. GTP-U Direct Tunnel



The following figure shows the logic used within the SGSN to determine if a direct tunnel can be used.

Figure 6. Direct Tunnel Activation Logic



Direct Tunnel Configuration

There are three aspects to direct tunnel configuration on the SGSN:

- enabling or disabling support for direct tunnel functionality on the SGSN
- enabling or disabling direct tunnel (DT) access to specific GGSN(s)
- enabling or disabling direct tunnel (DT) access to specific radio network controller(s) (RNCs)

The procedures in the remaining portion of this chapter cover all three of the aspects noted above.

Enabling and Disabling GTP-U Direct Tunnels

By default, direct tunnel support is *disallowed* on the SGSN and allowed on the GGSN. The SGSN's direct tunnel functionality is enabled within the SGSN operator policy configuration.

SGSN operator policies specify the rules governing the services, facilities, and privileges available to subscribers depending on factors such as roaming agreements between operators, the subscription restrictions for visiting or roaming subscribers, provisioning of defaults, over-riding of standard behavior.

One aspect of a policy is to allow or disallow the setup of direct GTP-U tunnels. If no operator policies are configured, the system looks at the settings in the system operator policy named *default*. If direct tunnel is allowed in the default operator policy, then *any* incoming call that does not have an applicable operator policy configured will have direct tunnel *allowed*.

Enabling a Direct Tunnel

To enable a direct tunnel in an SGSN operator policy, use the following procedure:

Step 1 Enter the global configuration mode.

Step 2 Create a new SGSN operator policy **or** Identify an existing SGSN operator policy.

Step 3 Enable direct tunnel functionality.

Example Configuration

The following is an example of the commands used to enable direct tunneling on the SGSN:

```
config
  sgsn-operator-policy { name <policy_name> | default }
      direct-tunnel attempt-when-permitted
```

Disabling a Direct Tunnel

To disable a direct tunnel in an SGSN operator policy, use the following procedure:

Step 1 Enter the global configuration mode.

Step 2 Identify the existing SGSN operator policy with the DT enabled.

Step 3 Disable the direct tunnel functionality.

Example Configuration

The following is an example of the commands used to disable direct tunneling on the SGSN:

```
config
  sgsn-operator-policy { name <policy_name> | default }
      remove direct-tunnel
```

Disabling or Enabling DT Access to Specific GGSN(s)

In each SGSN operator policy, SGSN APN policies are configured to connect to a GGSN and to control the direct tunnel (DT) access to that GGSN.

Multiple SGSN APN policies can be configured per SGSN operator policy.

By default, DT functionality is *allowed* in SGSN APN policies.

Disabling a Direct Tunnel to a GGSN

To disable a direct tunnel access to a GGSN configured in an SGSN APN policy, use the following procedure:

Step 1 Enter the global configuration mode.

Step 2 Identify the existing SGSN operator policy with the SGSN APN policy.

Step 3 Identify the SGSN APN policy with the GGSN supporting DT.

Step 4 Disable the direct tunnel functionality.

Example Configuration

The following is an example of the commands used to disable direct tunneling with a GGSN:

```
config
  sgsn-operator-policy { name <policy_name> | default }
      apn network-identifier <net_id> | operator-identifier <op_id> }
         direct-tunnel not-permitted-by-ggsn
```

Re-enabling a Direct Tunnel to a GGSN

To re-enable a direct tunnel access to a GGSN configured in an SGSN APN policy, use the following procedure:

- Enter the global configuration mode.
- Identify the existing SGSN operator policy with the SGSN APN policy.
- Identify the SGSN APN policy with the GGSN.
- Enable the direct tunnel functionality.

Example Configuration

The following is an example of the commands used to re-enable direct tunneling with a GGSN:

```
config
  sgsn-operator-policy { name <policy_name> | default }
      apn network-identifier <net_id> | operator-identifier <op_id> }
         default direct-tunnel
```

Disabling or Enabling Direct Tunnels to Specific RNC(s)

SGSN access to radio access controllers (RNCs) is configured in the IuPS service.

Each IuPS service can include multiple RNC configurations that determine communications and features depending on the RNC.

By default, direct tunnel functionality is *enabled* for all RNCs.

Disabling a Direct Tunnel to an RNC

To disable direct tunnel access to an RNC, use the following procedure:

Step 1 Enter the global configuration mode.

Step 2 Identify the context containing the IuPS service configuration for that RNC.

Step 3 Identify the IuPS service configuration.

Step 4 Identify the RNC configuration

Step 5 Disable the direct tunnel access for the RNC.

Example Configuration

The following is an example of the commands used to disable direct tunneling with an RNC:

```
config
  context  <ctx_name>
      iups-service <service_name>
         rnc id <rnc_id>
            direct-tunnel not-permitted-by-rnc
```

Re-enabling a Direct Tunnel to an RNC

To re-enable a direct tunnel access to an RNC configured in an IuPS service configuration, use the following procedure:

Step 1 Enter the global configuration mode.

Step 2 Identify the context containing the IuPS service configuration for that RNC.

Step 3 Identify the IuPS service configuration.

Step 4 Identify the RNC configuration

Step 5 Re-enable the direct tunnel access for the RNC.

Example Configuration

The following is an example of the commands used to re-enable direct tunneling with an RNC:

```
config
```

```
context <ctx_name>
    iups-service <service_name>
       rnc id <rnc_id>
          default direct-tunnel
```

Chapter 11
GRE Protocol Interface

This chapter provides information on Generic Routing Encapsulation protocol interface support in GGSN service node. 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 features described in this chapter is an enhanced feature and need enhanced feature license. This support is only available if you have purchased and installed particular feature support license on your chassis.

This chapter discusses following topics for GRE protocol interface support:

Introduction

GRE protocol functionality adds one additional protocol on ST-series Multimedia Core Platforms (ST40 or higher) to support mobile users to connect to their enterprise networks through Generic Routing Encapsulation (GRE).

GRE tunnels can be used by the enterprise customers of a carrier 1) To transport AAA packets corresponding to an APN over a GRE tunnel to the corporate AAA servers and, 2) To transport the enterprise subscriber packets over the GRE tunnel to the corporation gateway.

The corporate servers may have private IP addresses and hence the addresses belonging to different enterprises may be overlapping. Each enterprise needs to be in a unique virtual routing domain, known as VRF. To differentiate the tunnels between same set of local and remote ends, GRE Key will be used as a differentiator.

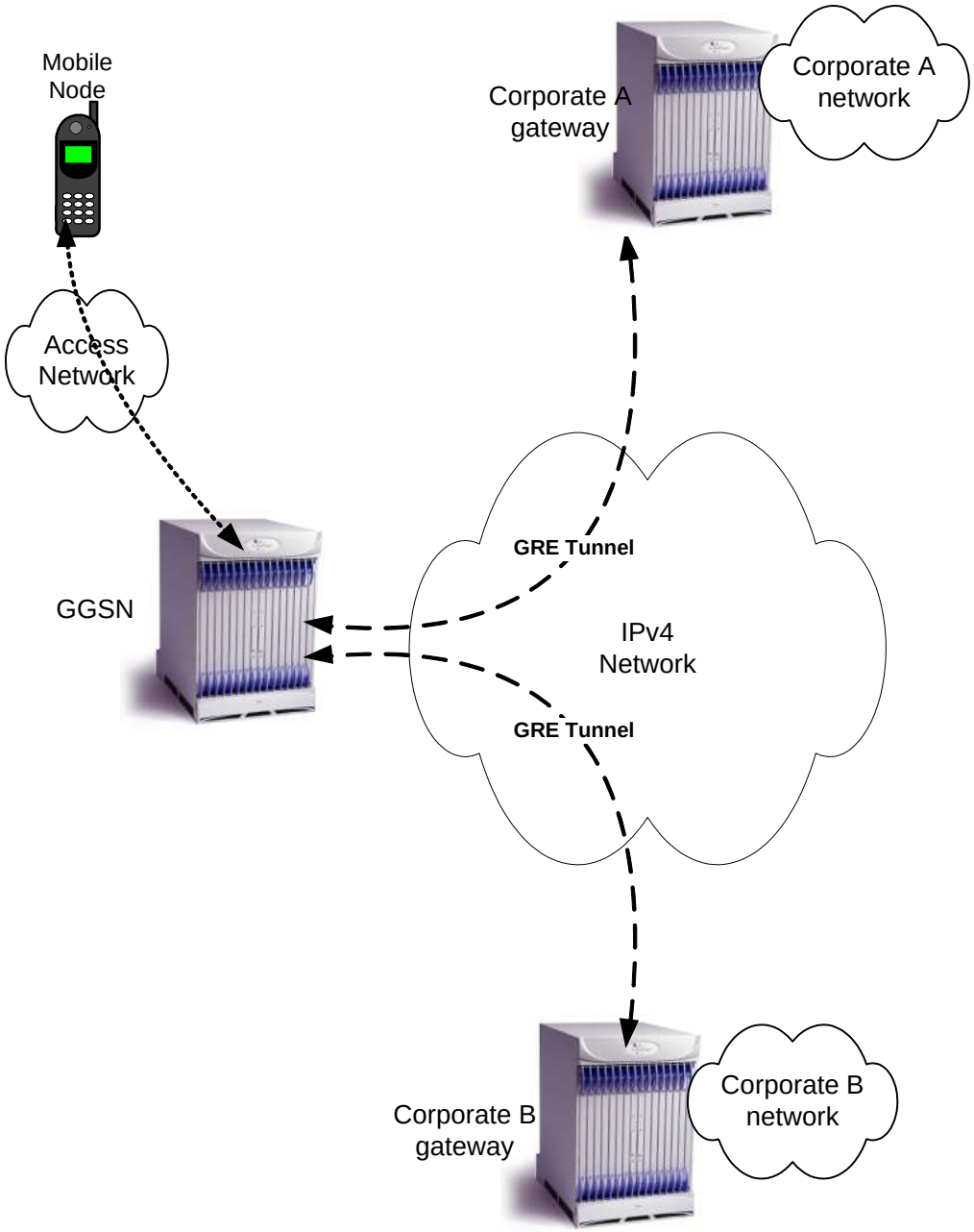
It is a common technique to enable multi-protocol local networks over a single-protocol backbone, to connect non-contiguous networks and allow virtual private networks across WANs. This mechanism encapsulates data packets from one protocol inside a different protocol and transports the data packets unchanged across a foreign network. It is important to note that GRE tunneling does not provide security to the encapsulated protocol, as there is no encryption involved (like IPSEC offers, for example).

GRE Tunneling consists of three main components:

- Passenger protocol-protocol being encapsulated. For example: CLNS, IPv4 and IPv6.
- Carrier protocol-protocol that does the encapsulating. For example: GRE, IP-in-IP, L2TP, MPLS and IPSEC.
- Transport protocol-protocol used to carry the encapsulated protocol. The main transport protocol is IP.

The most simplified form of the deployment scenario is shown in the following figure, in which GGSN has two APNs talking to two corporate networks over GRE tunnels.

Figure 7. GRE Interface Deployment Scenario



Supported Standards

Support for the following standards and requests for comments (RFCs) have been added with this interface support:

- RFC 1701, Generic Routing Encapsulation (GRE)
- RFC 1702, Generic Routing Encapsulation over IPv4 networks
- RFC 2784, Generic Routing Encapsulation (GRE)
- RFC 2890, Key and Sequence Number Extensions to GRE

Supported Networks and Platforms

This feature supports all ASR5000 platforms with StarOS Release 9.0 or later running GGSN and/or SGSN service for the core network services.

Licenses

This feature support requires following feature license/s installed on the system:

- 600-00-7861
- 600-00-7862

Services and Application on GRE Interface

GRE interface implementation provides following functionality with GRE protocol support.

How GRE Interface Support Works

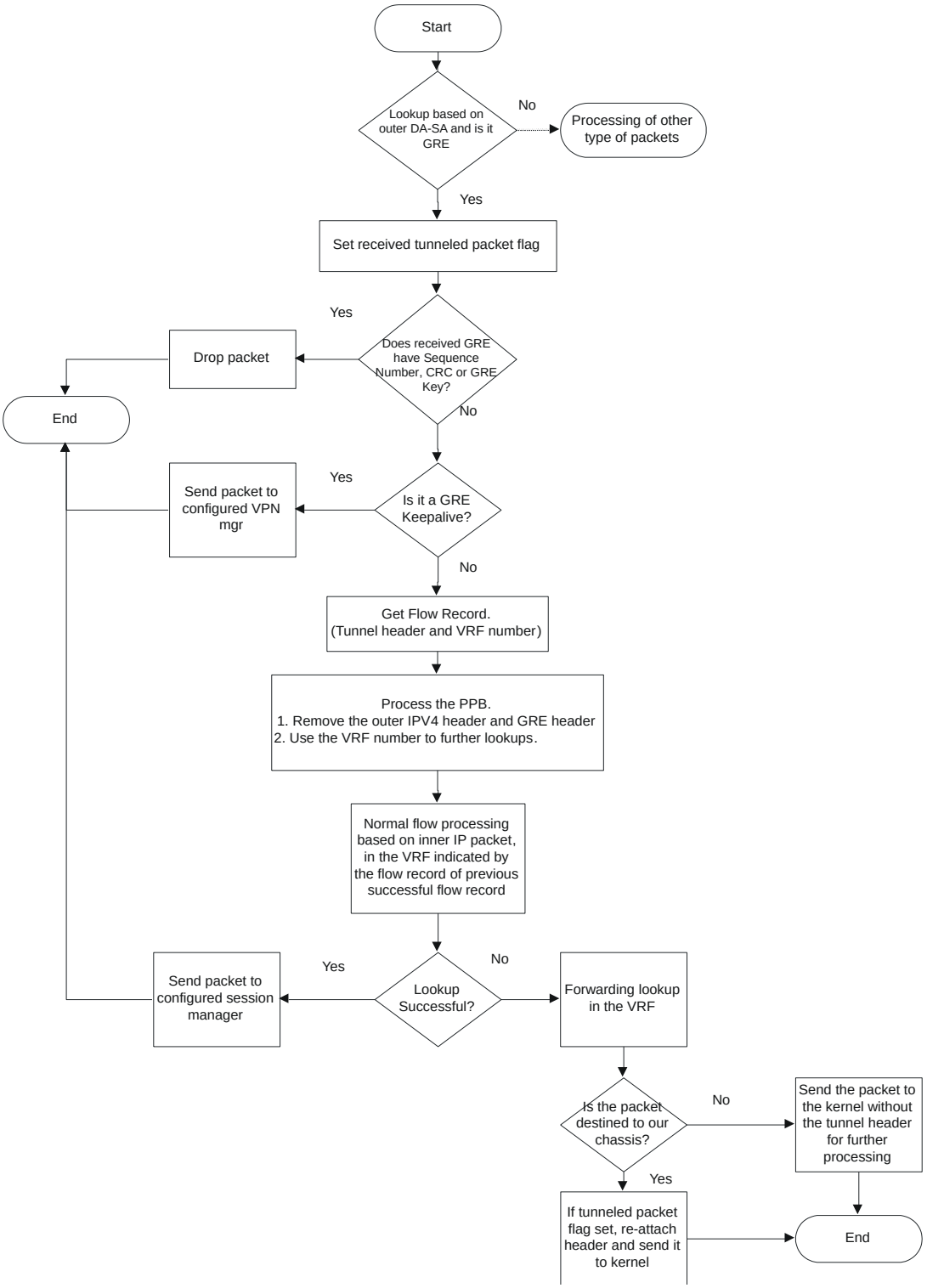
The GRE interface provides two types of data processing; one for ingress packets and another for egress packets.

Ingress Packet Processing on GRE Interface

Figure given below provides a flow of process for incoming packets on GRE interface.

Note that in case the received packet is a GRE keep-alive or a ping packet then the outer IPV4 and GRE header are not stripped off (or get reattached), but instead the packet is forwarded as is to the VPN manager or kernel respectively. In case of all other GRE tunneled packets the IPV4 and GRE header are stripped off before sending the packet for a new flow lookup.

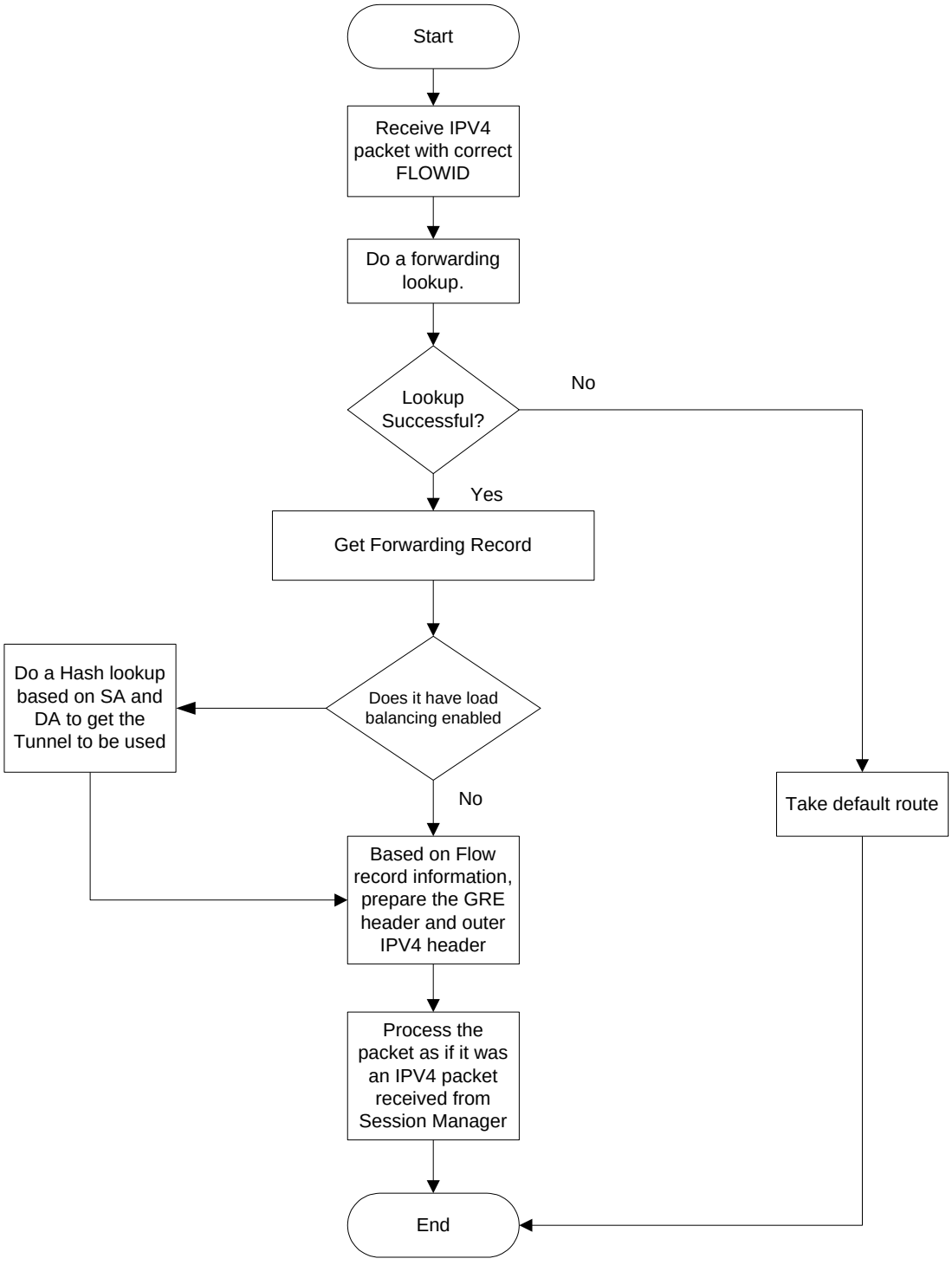
Figure 8. Ingress Packet Processing on GRE Interface



Egress Packet Processing on GRE Interface

Figure given below provides a flow of process for outgoing packets on GRE interface:

Figure 9. Egress Packet Processing on GRE Interface



GRE Interface Configuration

This section provides a high-level series of steps and the associated configuration examples for configuring the system with GRE interface in GGSN services.

Important: This section provides the minimum instruction set to enable the GRE Protocol Interface support functionality on GGSN in UMTS network. Commands that configure additional function for this feature are provided in the Command Line Interface Reference.

These instructions assume that you have already configured the system level configuration as described in System Administration Guide and specific product Administration Guide.

To configure the system to support GRE tunnel interface:

Step 1 Configure the virtual routing and forwarding (VRF) in a context by applying the example configurations presented in the Virtual Routing And Forwarding (VRF) Configuration section.

Step 2 Configure the GRE tunnel interface in a context by applying the example configurations presented in the GRE Tunnel Interface Configuration section.

Step 3 Enable OSPF for the VRF and for the given network by applying the example configurations presented in the Enabling OSPF for VRF section.

Step 4 Associate IP pool and AAA server group with VRF by applying the example configurations presented in the Associating IP Pool and AAA Group with VRF section.

Step 5 Associate APN with VRF through AAA server group and IP pool by applying the example configurations presented in the Associating APN with VRF section.

Step 6 Optional. If the route to the server is not learnt from the corporate over OSPFv2, static route can be configured by applying the example configurations presented in the Static Route Configuration section.

Step 7 Save the changes to system configuration by applying the example configuration given in *Verifying and Saving Your Configuration* chapter.

Step 8 Verify configuration of GRE and VRF related parameters by applying the commands provided in the *Verifying Your Configuration* section of this chapter.

Virtual Routing And Forwarding (VRF) Configuration

This section provides the configuration example to configure the VRF in a context:

```
configure

  context <vpn_context_name> -noconfirm ]

     ip vrf <vrf_name>
```

```
ip maximum-routes <max_routes>
end
```

Notes:

- <*vpn_context_name*> is the name of the system context you want to use for VRF. For more information, refer System Administration Guide.
- A maximum of 100 VRFs in one context and up to 1024 VRFs on one chassis can be configured on system.
- <*vrf_name*> is name of the VRF which is to be associated with various interfaces.
- A maximum of 10000 routes can be configured through ip maximum-routes <*max_routes*> command.

GRE Tunnel Interface Configuration

This section provides the configuration example to configure the GRE tunnel interface and associate a VRF with GRE interface:

```
configure
  context <vpn_context_name>
    ip interface <intfc_name> tunnel
      ip vrf forwarding <vrf_name>
        ip address <internal_ip_address/mask>
        tunnel-mode gre
        source interface <non_tunn_intfc_to_corp>
        destination address <global_ip_address>
        keepalive interval <value> num-retry <retry>
        end
```

Notes:

- <*vpn_context_name*> is the name of the system context you want to use for GRE interface configuration. For more information, refer Command Line Interface Reference.
- A maximum of 511 GRE tunnels + 1 non-tunnel interface can be configured in one context. System needs at least 1 non-tunnel interface as a default.
- <*intfc_name*> is name of the IP interface which is defined as a tunnel type interface and to be used for GRE tunnel interface.
- <*vrf_name*> is the name of the VRF which is preconfigured in context configuration mode.
- <*internal_ip_address*/mask> is the network IP address with sub-net mask to be used for VRF forwarding.

- *<non_tunn_intfc_to_corp>* is the name a non-tunnel interface which is required by system as source interface and preconfigured. For more information on interface configuration refer System Administration Guide.
- *<global_ip_address>* is a globally reachable IP address to be used as a destination address.

Enabling OSPF for VRF

This section provides the configuration example to enable the OSPF for VRF to support GRE tunnel interface:

```
configure
  context <vpn_context_name>
    router ospf
      ip vrf <vrf_name>
      network <internal_ip_address/mask>
      end
```

Notes:

- *<vpn_context_name>* is the name of the system context you want to use for OSPF routing. For more information, refer *Routing* chapter in *System Enhanced Feature Configuration Guide*.
- *<vrf_name>* is the name of the VRF which is preconfigured in context configuration mode.
- *<internal_ip_address/mask>* is the network IP address with sub-net mask to be used for OSPF routing.

Associating IP Pool and AAA Group with VRF

This section provides the configuration example for associating IP pool and AAA groups with VRF:

```
configure
  context <vpn_context_name>
    ip pool <ip_pool_name> <internal_ip_address/mask> vrf <vrf_name>
      exit
    aaa group <aaa_server_group>
      ip vrf <vrf_name>
      end
```

Notes:

- <*vpn_context_name*> is the name of the system context you want to use for IP pool and AAA server group.
- <*ip_pool_name*> is name of a preconfigured IP pool. For more information refer System Administration Guide.
- <*aaa_server_group*> is name of a preconfigured AAA server group. For more information refer AAA Interface Administrtion and Reference.
- <*vrf_name*> is the name of the VRF which is preconfigured in context configuration mode.
- <*internal_ip_address/mask*> is the network IP address with sub-net mask to be used for IP pool.

Associating APN with VRF

This section provides the configuration example for associating an APN with VRF through AAA group and IP pool:

```
configure
  context <vpn_context_name>
    apn <apn_name>
      aaa group <aaa_server_group>
      ip address pool name <ip_pool_name>
      end
```

Notes:

- <*vpn_context_name*> is the name of the system context you want to use for APN configuration.
- <*ip_pool_name*> is name of a preconfigured IP pool. For more information refer System Administration Guide.
- <*aaa_server_group*> is name of a preconfigured AAA server group. For more information refer AAA Interface Administrtion and Reference.
- <*vrf_name*> is the name of the VRF which is preconfigured in context configuration mode.

Static Route Configuration

This section provides the optional configuration example for configuring static routes when the route to the server is not learnt from the corporate over OSPFv2:

```
configure
  context <vpn_context_name>
    ip route <internal_ip_address/mask> tunnel <tunnel_intfc_name> vrf
<vrf_name>
```

```
end
```

Notes:

- <*vpn_context_name*> is the name of the system context you want to use for static route configuration.
- <*internal_ip_address/mask*> is the network IP address with sub-net mask to be used as static route.
- <*tunnel_intfc_name*> is name of a predefined tunnel type IP interface which is to be used for GRE tunnel interface.
- <*vrf_name*> is the name of the VRF which is preconfigured in context configuration mode.

Verifying Your Configuration

This section explains how to display and review the configurations after saving them in a .cfg file as described in Saving Your Configuration chapter of this guide and also to retrieve errors and warnings within an active configuration for a service.

Important: All commands listed here are under Exec mode. Not all commands are available on all platforms.

These instructions are used to verify the GRE interface configuration.

Step 1 Verify that your interfaces are configured properly by entering the following command in Exec Mode:

`show ip interface`

The output of this command displays the configuration of the all interfaces configured in a context.

```
Intf Name:       foo1
Intf Type:       Broadcast
Description:
IP State:        UP (Bound to 17/2 untagged, ifIndex 285343745)
IP Address:      1.1.1.1              Subnet Mask:     255.255.255.0
Bcast Address:   1.1.1.255            MTU:             1500
Resoln Type:     ARP                  ARP timeout:     60 secs
L3 monitor LC-port switchover: Disabled
Number of Secondary Addresses: 0
Intf Name:       foo2
Intf Type:       Tunnel (GRE)
Description:
VRF:             vrf-tun
IP State:        UP (Bound to local address 1.1.1.1 (foo1), remote
address 5.5.5.5)
IP Address:      10.1.1.1             Subnet Mask:     255.255.255.0
Intf Name:       foo3
Intf Type:       Tunnel (GRE)
Description:
```

```
        IP State:        DOWN (<state explaining the reason of being down>)
        IP Address:      20.20.20.1             Subnet Mask:     255.255.255.0
```

Step 2 Verify that GRE keep alive is configured properly by entering the following command in Exec Mode:

`show ip interface gre-keepalive`

The output of this command displays the configuration of the keepalive for GRE interface configured in a context.

Chapter 12
Gx Interface Support

This chapter provides information on configuring Gx interface to support policy and charging control for subscribers in GPRS/UMTS networks.

The IMS service provides application support for transport of voice, video, and data independent of access support. Roaming IMS subscribers in GPRS/UMTS 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 chapter you select the configuration example that best meets your service model, and configure the required elements for that model as described in the *Gateway GPRS Support Node Administration Guide* or the *IP Services Gateway Administration Guide*.

Important: The IMS Authorization Service feature described in this chapter is only available if you have purchased and installed a Dynamic Policy Interface feature license on the chassis. If you have not previously purchased this enhanced feature, contact your sales representative for more information.

The following topics are covered in this chapter:

Rel. 6 Gx Interface

Important: The Rel 6. Gx functionality is supported on StarOS 8.0 and later.

This section describes the following topics:

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 local technical support 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).

Licensing

This feature requires the following license to be installed on the chassis:

[600-00-7585] *Dynamic Policy Interface* — license for IMS Authorization Service feature

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
- 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.

Supported Networks and Platforms

This feature is supported on all ASR 5000 Series Platforms with StarOS Release 8.0 or later running GGSN service for the core network services.

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 10. Rel. 6 Gx IMS Authorization Call Flow

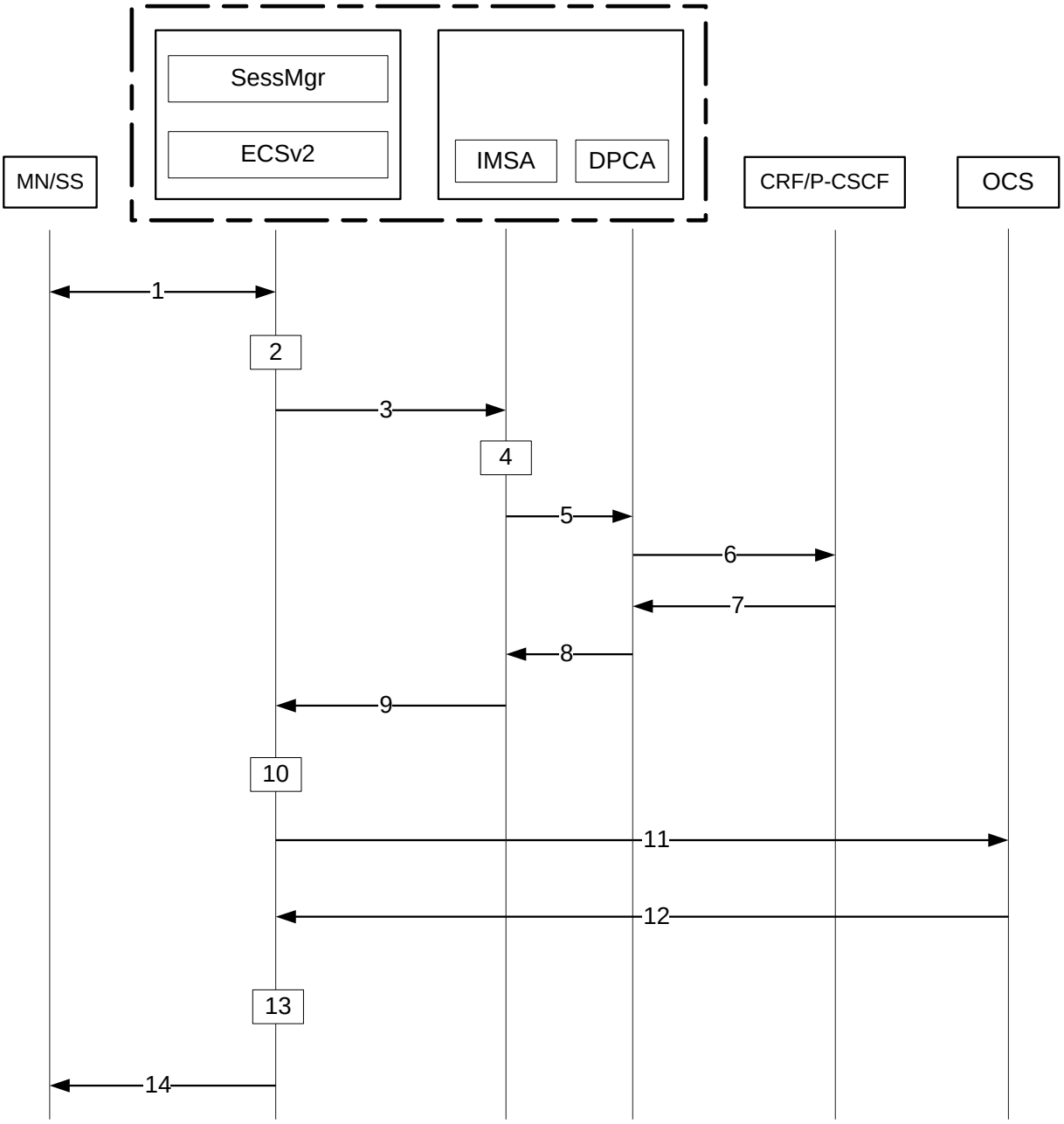


Table 9. Rel. 6 Gx IMS Authorization Call flow Description

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 (e.g. APN, QoS etc.).
12	OCS returns a CCA-Initial message to activate the statically configured rulebase and includes pre-emptive 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 as described in the *Verifying and Saving Your Configuration* chapter.

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 <ims_auth_service>

         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 |
msisdn-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
| msisdn-modulus | round-robin }

            end
```

Notes:

- *<context_name>* must be the name of the context where you want to enable IMS Authorization Service.
- *<ims_auth_service>* 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, please contact your local service representative.

- Option: 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>*

- Option: 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>* **]**

- Option: 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 }

- Option: 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 | msisdn-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 *<ims_auth_service>*

Applying IMS Authorization Service to an APN

After configuring IMS Authorization service at the context-level, an APN within the same context 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 section.

```
configure
   context <context_name>
      apn <apn_name>
         ims-auth-service <ims_auth_service>
```

```
end
```

Notes:

- *<context_name>* must be the name of the context in which the IMS Authorization service was configured.
- *<ims_auth_service>* 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 *<ims_auth_service>*

<ims_auth_service> must be the name of the IMS Authorization Service configured for IMS authentication.

Saving the Configuration

To save changes made to the system configuration, refer to the *Verifying and Saving Your Configuration* chapter.

Rel. 7 Gx Interface

Important: The Rel 7. Gx functionality is supported only on StarOS 8.1 and StarOS 9.0 and later.

This section describes the following topics:

Introduction

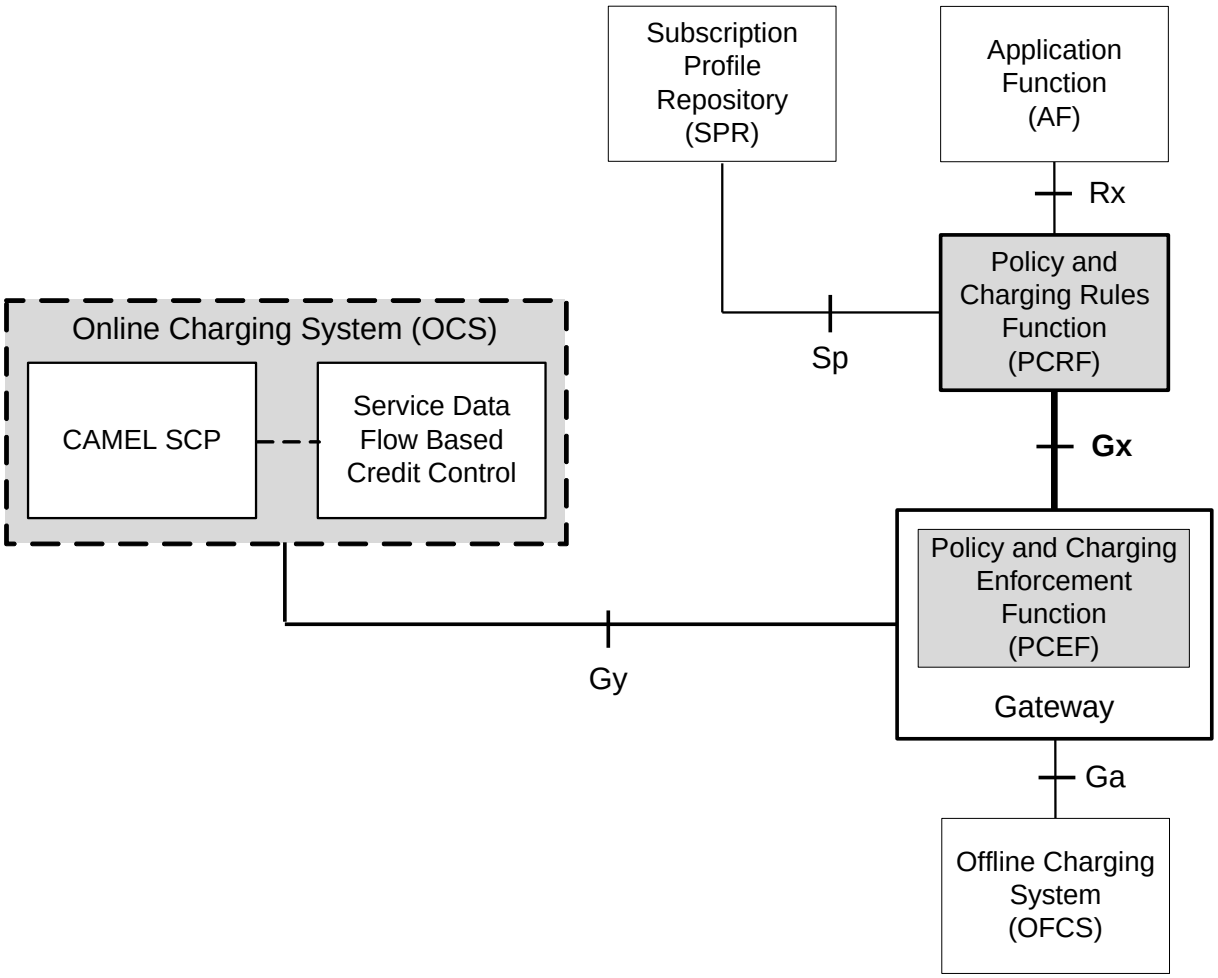
For IMS deployment in GPRS/UMTS networks the system uses Rel. 7 Gx interface for policy-based admission control support and flow-based charging. The Rel. 7 Gx interface supports enforcing policy control features like gating, bandwidth limiting, etc., and also supports flow-based charging. This is accomplished 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.

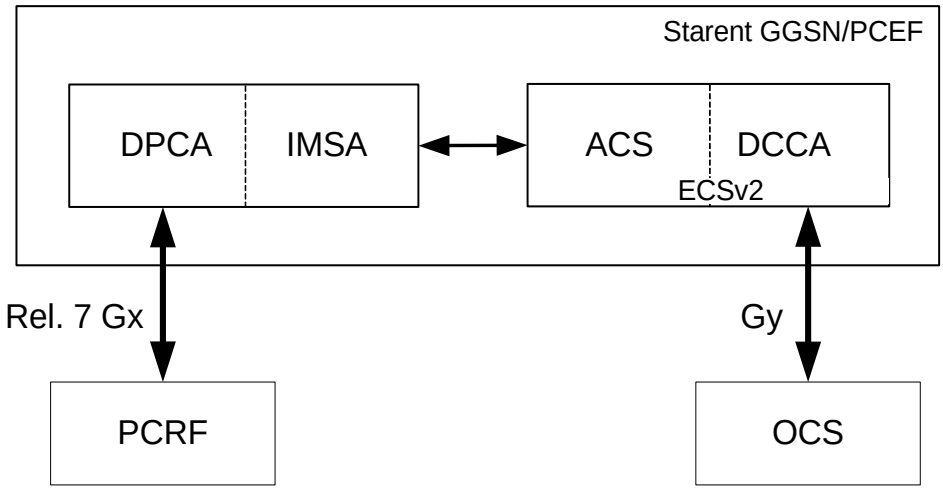
The Rel. 7 Gx reference point is located between the Gateway and the PCRF. This reference point is used for provisioning and removal of PCC rules from the PCRF to the Gateway, and the transmission of traffic plane events from the Gateway to the PCRF. The Gx reference point can be used for charging control, 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 11. 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 12. PCC Architecture within Cisco PCEF



Supported Networks and Platforms

This feature is supported on all ASR 5000 Series Platforms with StarOS Release 8.1 and later running GGSN service for the core network services.

Licensing

This feature requires the following licenses to be installed on the chassis:

- [600-00-7585] *Dynamic Policy Interface* — for IMS Authorization Service feature
- [600-00-7574] *Enhanced Charging Bundle 2 1k Sessions* — for ECS functionality

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

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.
  - 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.

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, i.e. 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, i.e. 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.
  - 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 (i.e., rules which 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, shall cause the PCEF to trigger a PCRF interaction to request PCC rules from the PCRF for an established IP CAN session. The PCEF shall stop the timer once the PCEF triggers a REVALIDATION_TIMEOUT event.

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 (e.g. 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, i.e. 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)

  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

Important: In this release, PCEF does not send the default online and offline charging method in the Credit Control Request - Initial (CCR(I)).

Charging Correlation

For the purpose of charging correlation between SDF level and application level (e.g. 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 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), the Allocation and Retention Priority (ARP), and authorized bitrates for uplink and downlink.
- Charging Key (i.e. 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, etc.

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.

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.

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. 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 (e.g. 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.

Volume Reporting Over Gx

This section describes the Volume Reporting over Gx feature.

Supported Standards

The Volume Reporting over Gx feature is based on the following standard:

3GPP TS 29.212 V9.1.0 (2009-12): 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: Only total data usage reporting is supported. Uplink/Downlink level reporting is not supported in this release.

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: Usage reporting on Bearer termination is not 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

Important: In this release, the Usage-Monitoring-Informtion AVP is expected to be present without a new threshold to disable the usage monitoring after usage reporting when the threshold is breached. If no Usage-Monitoring-Informtion AVP is present in the CCA-U, the monitoring will continue with the old threshold value and the reporting happens when the threshold is breached again.

- 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.

- 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 both predefined rules and dynamic rule definitions.

  - 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.

  If the non-standard volume reporting flag is set, the 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.
- 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.
- PCRF Requested Usage Report


Important: PCRF Requested Usage Reporting is not supported in this release.

- Revalidation Timeout: 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).

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.

How R7 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.

Figure 13. Rel. 7 Gx IMS Authorization Call Flow

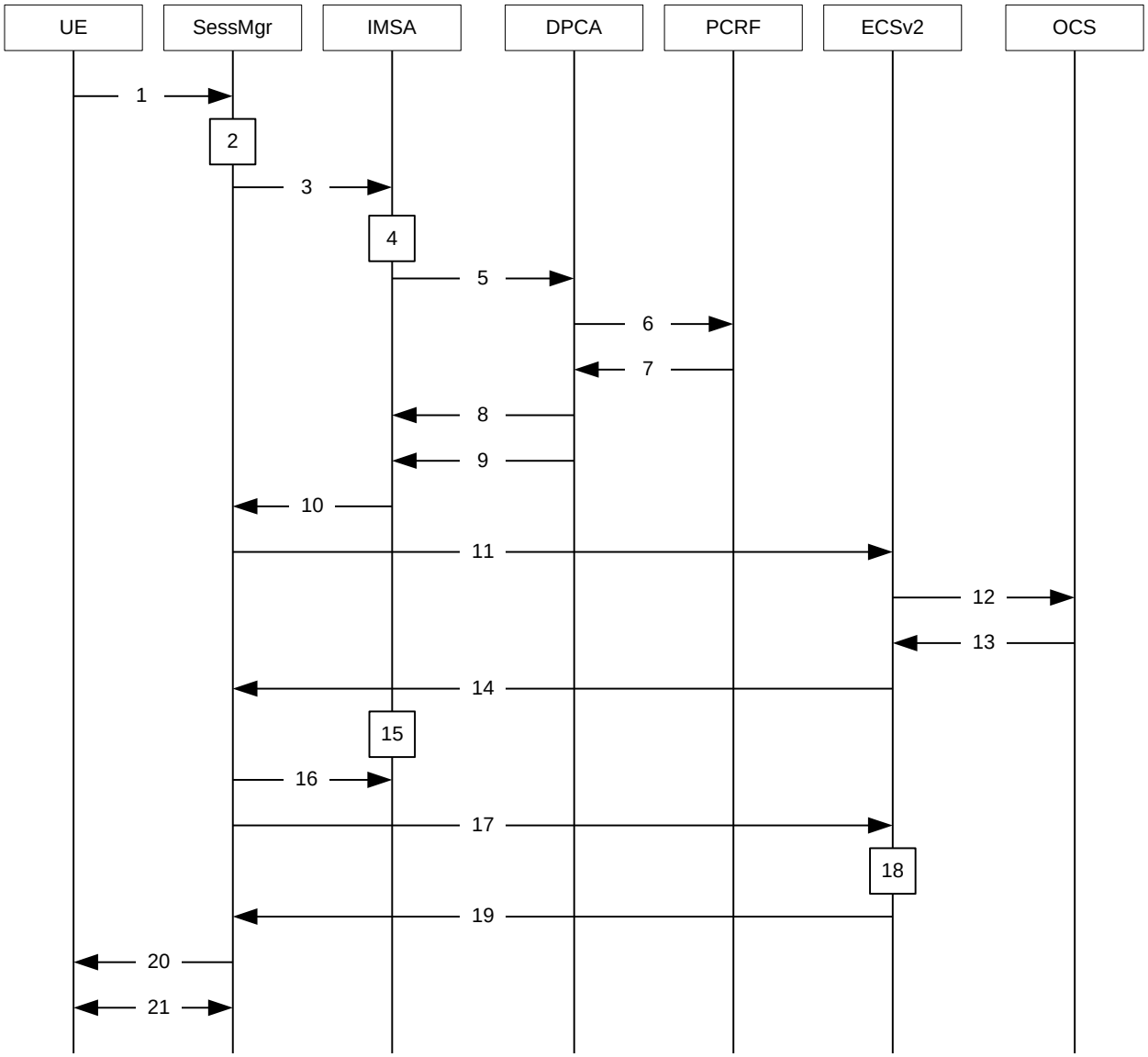


Table 10. Rel. 7 Gx IMS Authorization Call flow Description

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 (ex msisdn).
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. 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, etc. 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, 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 APN name, 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 APN/AAA) and GPRS specific attributes (e.g. APN, UMTS QoS, etc.).
13	OCS returns a CCA initial message that may activate a statically configured Rulebase and may include pre-emptive 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 until the primary PDP context is established, all RAR messages from the PCRF are rejected.
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-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 Save your configuration as described in the *Verifying and Saving Your Configuration* chapter.

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 <ims_auth_service>

         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>
```

```
            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.
- *<ims_auth_service>* 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, please contact your local service 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.
- Option: 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>*
- *Option*: 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>* **]**

- *Option*: 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.

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 *<ims_auth_service>*

Applying IMS Authorization Service to an APN

After configuring IMS Authorization service at the context-level, an APN within the same context 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 <ims_auth_service>
         active-charging rulebase <rulebase_name>
         end
```

Notes:

- *<context_name>* must be the name of the context in which the IMS Authorization service was configured.

- *<ims_auth_service>* 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.
- Provided interpretation of the Gx rulebase is chosen to be ECS group-of-ruledefs, configure the following command available in the Active Charging Service Configuration Mode:

  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 *<ims_auth_service>*

<ims_auth_service> must be the name of the IMS Authorization service configured for IMS authentication.

Saving the Configuration

To save changes to the configuration, refer to the *Verifying and Saving Your Configuration* chapter.

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 11. Gathering Rel. 7 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**
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**

Statistics/Information	Action to perform
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 }

Chapter 13
HA Proxy DNS Intercept

This chapter describes the system's support for the HA Proxy DNS Intercept feature 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.

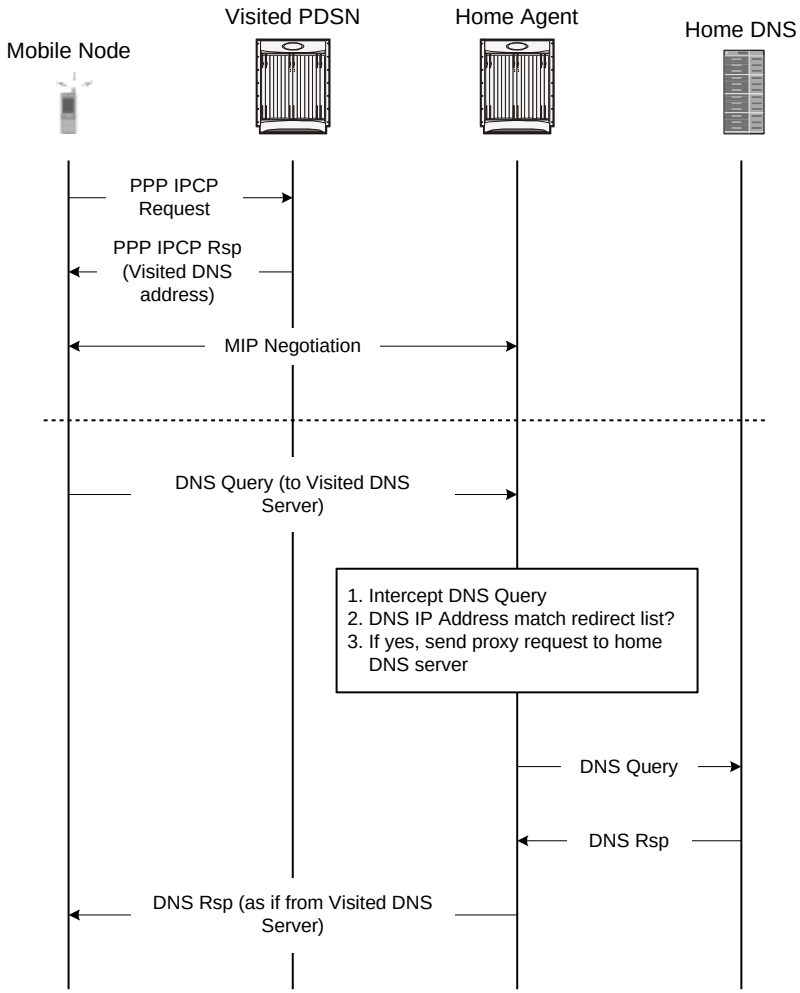

Important: HA Proxy DNS Intercept is a license-enabled feature.

Overview

An inherent problem in many mobile IP scenarios is the placement of the foreign network's Domain Name Server (DNS) behind a firewall. When a mobile user roams into a foreign network and the DNS address is returned to the home network, the home network does not have access to the foreign network's DNS. A common solution is to implement IS-835D, but the majority of legacy mobile handsets and most current handsets do not support this standard.

To address this, a proxy DNS intercept feature is available for the Home Agent (HA). This feature, when configured, looks for DNS packets and compares the DNS IP address in the destination address field to a configured rules list. If the destination address matches an address on a "pass through" rules list, the packets are allowed to continue without modification. If the destination address is on a "redirect" rules list, the packets are intercepted and the visited network's DNS IP address is replaced with the home network's DNS IP address while the call is accessing the home network. When the DNS response is returned to the mobile node, the HA removes the home network's DNS address and returns the original visited network's address so the mobile node is not aware that a modification has occurred. The flow in the following figure provides an example of what happens when a visited networks DNS address is intercepted by the HA.

Figure 14. HA Proxy DNS Intercept Flow



Configuring Proxy DNS Intercept

To configure the Proxy DNS Intercept feature:

Step 1 Enable the Proxy DNS Intercept feature in the subscriber destination context as described in the Enabling Proxy DNS Intercept in the Destination Context section.

Step 2 Create the list of rules used to specify how an intercepted DNS packet is to be processed as described in the Creating the Proxy DNS Intercept Rules List section.

Step 3 Associate a system-configured subscriber with a configured Proxy DNS rules list as described in the Associating a Proxy DNS Intercept Rules List With a Subscriber section.

Step 4 Save your configuration as described in the *Saving Your Configuration* chapter.

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.

Enabling Proxy DNS Intercept in the Destination Context

Use the following example to enable the Proxy DNS Intercept feature in the subscriber destination context:

```
configure
   context <context_name>
      ip dns-proxy source-address <ip_address>
      end
```

Notes:

- The **ip dns-proxy source-address** *<ip_address>* command must be entered in the destination context for the subscriber. If there are multiple destination contexts for different subscribers, the command must be entered in each context. This feature uses UDP port 53.
- *<ip_address>* must be the interface in the current context where all redirected DNS requests will be sent.

Creating the Proxy DNS Intercept Rules List

Use the following example to create the list of rules in the AAA context, which is used to specify how an intercepted DNS packet is to be processed:

```
configure
   context <context_name>
      proxy-dns intercept-list <name>
         pass-thru <ip_address> [ /<ip_mask> ]
         redirect <ip_address> [ /<ip_mask> ]
         end
```

Notes:

- *<name>* must be the name of the rules list for later association with a subscriber.
- Up to 64 separate rules lists can be configured in a single AAA context.
- Use the **pass-thru** command to set the DNS IP addresses that should be allowed through the intercept feature.
- Use the **redirect** command to set the DNS IP addresses that should be redirected by the intercept feature to the home DNS. Use the optional **primary-dns** and **secondary-dns** keywords to specify the IP addresses of primary and secondary home DNS servers. Refer to the *Command Line Interface Reference* for more information regarding these optional keywords.

  Important: If a packet does not match the pass-thru or redirect rule, the packet is dropped. If the optional keywords **primary-dns** or **secondary-dns** are not configured, DNS messages are redirected to the primary-dns-server (or the secondary-dns-server) configured for the subscriber OR inside the context.

- Up to 128 rules of any type can be configured per rules list.

Associating a Proxy DNS Intercept Rules List With a Subscriber

Use the following example to associate a system-configured subscriber with a configured Proxy DNS rules list.

```
configure
   context <context_name>
      subscriber name <user_name>
         proxy-dns intercept list-name <name>
         end
```

Chapter 14
HA Redundancy for Dynamic Home Agent Assignment

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.

This chapter includes the following topics:

Feature Description

This feature provides a mechanism for a system functioning as a Home Agent (HA) to communicate status information to a properly configured RADIUS server. The status information is used by the RADIUS server to determine the availability and readiness of the HA to accept Mobile IP (MIP) subscriber sessions. The RADIUS server's awareness of the HA status allows it to dynamically assign immediately available HAs to subscriber sessions.

When a RADIUS server assigns an HA to a Mobile Node (MN), it is very important that only active, or accessible, HAs are selected for the assignment. Therefore, it is necessary for the RADIUS server to detect the availability of each HA before assigning it to an MN. This feature allows the RADIUS server to gather and maintain a list of available HAs through a detection mechanism that provides frequent updates.

With this feature, bogus authentication messages, called probe authentication messages, are exchanged between the RADIUS server and the HA. The HA periodically sends Access-Request messages to the RADIUS server. The RADIUS server distinguishes the probe authentication request from other regular subscriber authentication messages, validates them, and sends proper response.

The probe Access-Request contains the following attributes and expects an Access-Accept from the RADIUS server.

```
User-Name = Probe-User

User-Password = 18 7F 88 02 82 1D B6 F6 70 48 B9 A1 4C 92 C3 3E

NAS-IP-Address = 182.168.65.2

Service-Type = Authenticate_Only

Event-Timestamp = 1255598429
```

User-Name and User-Password are configurable in the system.

If an Access-Accept message is sent in response to the probe authentication request, the RADIUS server updates the status of the HA as active. If an Access-Reject message is sent, the RADIUS server updates only the statistics without any further action. If the RADIUS server misses receiving a configured number of probe authentication requests, the HA, and all of its associated IP addresses, is marked as down, or inaccessible. When an HA is marked as down, a backup HA and its associated IP addresses are made active and used for assignment in the place of the inaccessible HA.

Supported Implementations

This feature is supported on system installations that are configured as Home Agents and are configured to communicate with a AAA Service Controller that supports the configuration of Active and Backup HAs. For more information on a compatible AAA Service Controller, contact your designated customer support engineer.

Configuring HA Redundancy for Dynamic Home Agent Assignment

Important: The commands or keywords/variables that are available are dependent on platform type, product version, and installed license(s).

Step 1 Configure the AAA Service Controller as described in the AAA Service Controller documentation.

Step 2 Configure RADIUS support on the HA as described in the Configuring RADIUS Support on the HA section.

Step 3 Save the configuration as described in *Verifying and Saving Your Configuration.*

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 the AAA Service Controller

The AAA Service Controller should be configured with the following parameters. For configuration information refer to the AAA Service Controller documentation.

- Authentication-Probe User profile:
  - Probe Username
  - Probe Password
- HA Client information:
  - HA Client IPv4 address (NAS-IP-Address attribute)
  - HA client secret (authenticator)
  - Whether the HA client is a Primary or Backup HA client
- One or more HA Service addresses for each HA client address.
- The number of missed probe authentication requests before the HA Client is marked as down.
- The number of seconds to wait for a probe authentication request from the HA client (timeout period).
- The number of seconds to wait for a backup HA server to be in the active state after a reboot, known as backup-hold-timeout.

Configuring RADIUS Support on the HA

Use the following example to configure RADIUS support on the HA:

```
configure
   context <context_name>
      radius server <ip_address> [ encrypted ] key <value>
      radius probe-interval <seconds>
      radius probe-max-retries <retries>
      radius probe-timeout <idle_seconds>
      end
```

Notes:

- <context_name> must be the name of the AAA context that the HA service uses for authentication.
- A number of optional keywords and variables are available for the **radius server** command. Refer to the *Command Line Interface Reference* for more information regarding this command.
- Option: To configure HA redundancy with AAA server group, in the Context Configuration Mode, use the following command:

  ```
  aaa group <group_name>
  ```

  <group_name> must be the name of the AAA group designated for AAA functionality within the context. A total of 400 server groups can be configured system-wide including the default server-group unless **aaa large-configuration** is enabled. For information on configuring context-level AAA functionality, refer to the AAA Interface Administration and Reference.

Verifying RADIUS Server Configurations

This section provides information to verify connectivity to the RADIUS server, and information to view counters and statistics that can be useful in troubleshooting issues.

Step 1 Verify connectivity to the RADIUS server by sending a test probe message to the RADIUS server by entering the following command:

```
radius test probe authentication server <ip_address> port <port_number> [
username <username> password <password> ]
```

Important: Any response, including **Access-Reject** and **Access-Accept** from the AAA server, is considered to mean that the AAA server is alive.

The following is a sample of the output of a successful probe authentication test.

```
[local]host_name# radius test probe authentication server 192.168.20.1
port 1812

Authentication from authentication server 192.168.20.1, port 1812

Authentication Success: Access-Accept received

Round-trip time for response was 714.2 ms
```

Step 2 View the RADIUS counters by entering the following command:

```
show radius counters { all | server <ip_address> [ port <port_number> ] }
[ | { grep <grep_options> | more } ]
```

The following is a sample output of the command displaying RADIUS Probe counters.

```
Server-specific Probing Counters

 ------------------------------

 State: Down

 Number of transactions issued:3

 Number of successful transactions:2

 Number of failed transactions:1

 Last successful transaction time: Thu Aug 26 17:40:32 2004
```

```
Last failed transaction time:Thu Aug 26 17:40:39 2004

Last roundtrip time:3.2 ms
```

Step 3 View AAA Manager statistics by entering the following command:

```
show session subsystem [ full | facility aaamgr [ all | instance <id> ] ]
[ verbose ] [ | { grep <grep_options> | more }]
```

The following is a sample output of the command displaying authentication probe statistics in the output.

```
AAAMgr: Instance 261

 4 Total aaa requests 0 Current aaa requests

 3 Total aaa auth requests 0 Current aaa auth requests

 0 Total aaa auth probes 0 Current aaa auth probes

 1 Total aaa acct requests 0 Current aaa acct requests
```

Chapter 15
ICAP Interface Support

This chapter provides information on configuring the external Active Content Filtering servers for a core network service subscriber. This chapter also describes the configuration and commands that are used to implement this feature.

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 respective product Administration Guide, before using the procedures in this chapter.

Supported Networks and Platforms

This feature supports ST16 and Cisco® ASR 5000 Chassis for the core network services configured on the system.

Licensing

External Content Filtering Server support through Internet Content Adaptation Protocol (ICAP) interface is a licensed feature. To enable this feature on your chassis you must install the following license, along with other required core network and in-line service licenses:

- 600-00-7578
- 600-00-8530

For information on additional license requirements for this feature, please contact your local sales representative.

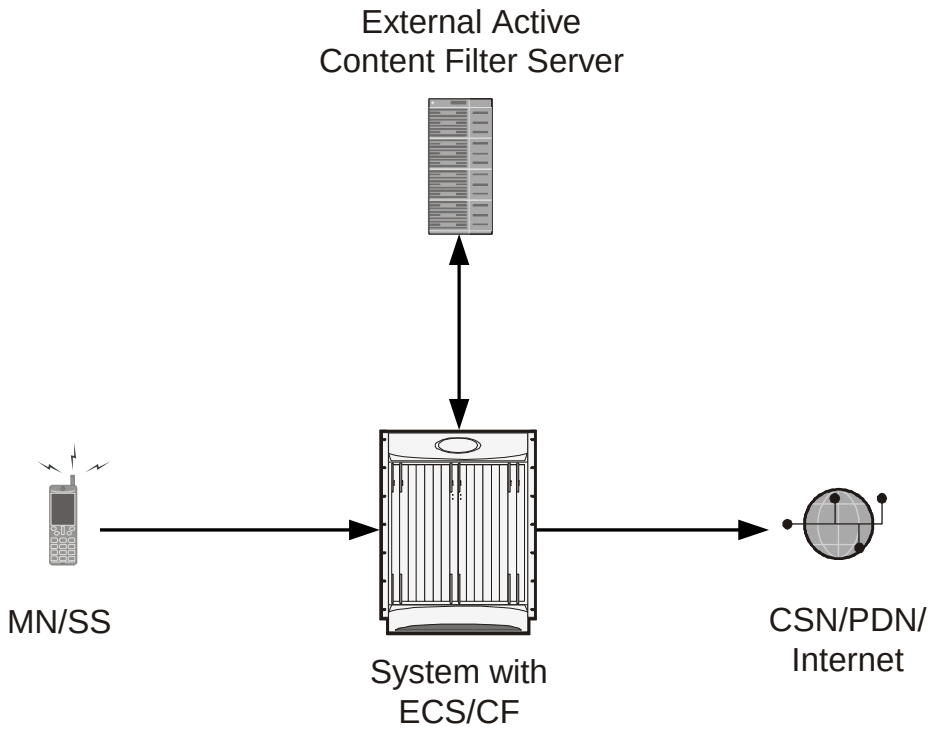
Important: For information on obtaining and installing licenses, refer to *Managing License Keys* in the *System Administration Guide*.

ICAP Interface Support Overview

This feature supports streamlined ICAP interface to leverage Deep Packet Inspection (DPI) to enable external application servers to provide their services without performing DPI, and without being inserted in the data flow. For example with an external Active Content Filtering (ACF) Platform.

A high-level view of the streamlined ICAP interface support for external ACF is shown in the following figure:

Figure 15. High-Level View of Streamlined ICAP Interface with external ACF



The system with ECS is configured to support DPI and the system uses this capability for content charging as well.

If a subscriber initiates a WAP (WAP1.x or WAP2.0) or Web session, the subsequent GET/POST request is detected by the DPI function. The URL of the GET/POST request is extracted and passed, along with subscriber identification information and the subscriber request, in an ICAP message to the application server. The application server 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 to the GET/POST with:

- A 200 OK message if the request is accepted
- A 302 Redirect message in case of redirection. This redirect message includes the URL to which the subscriber should be redirected
- A 403 Denied message is the request should be blocked

Depending on the response received, the system with ECS will either pass the request unmodified, or discard the message and respond to the subscriber with the appropriate redirection or block message.

Content charging is performed by the Active Charging Service (ACS) only after the request has been controlled by the application server. This guarantees the appropriate interworking between the external application and content-based billing. In particular, this guarantees that charging will be applied to the appropriate request in case of redirection, and that potential charging-based redirections (i.e. Advice of Charge, Top Up page, etc.) will not interfere with the decisions taken by the application server.

Functions of the ACF include:

- Retrieval of subscriber policies based on the subscriber identity passed in the ICAP message
- Determining the appropriate action (permit, deny, redirect) to take for the type of content based on subscriber profile
- Communication of the action (permit, deny, or redirect) decision for the URL back to the ACS module

Configuring ICAP Interface Support

This section describes how to configure the Content Filtering Server Group (CFSG) through Internet Content Adaptation Protocol (ICAP) interface between ICAP client and ACF server (ICAP server).

Important: This section provides the minimum instruction set for configuring external content filtering servers on ICAP interface on the system. For more information on commands that configure additional parameters and options, refer CFSG Configuration Mode Commands chapter in Command Line Interface Reference.

To configure the system to provide ICAP interface support for external content filtering servers:

Step 1 Create the Content Filtering Server Group and create ICAP interface with origin (local) IP address of chassis by applying the example configuration in the Creating ICAP Server Group and Address Binding section.

Step 2 Specify the active content filtering server (ICAP sever) IP addresses and configure other parameters for ICAP server group by applying the example configuration in the Configuring ICAP Server and Other Parameters section.

Step 3 Configure the content filtering mode to external content filtering server group mode in ECS rule base by applying the example configuration in the Configuring ECS Rulebase for ICAP Server Group section.

Step 4 *Optional*. Configure the charging action to forward HTTP/WAP GET request to external content filtering servers on ICAP interface in Active Charging Configuration mode by applying the example configuration in the Configuring Charging Action for ICAP Server Group section.

Step 5 Verify your ICAP interface and external content filtering server group configuration by following the steps in the Verifying the ICAP Server Group Configuration section.

Step 6 Save your configuration as described in the *Verifying and Saving Your Configuration* chapter.

Creating ICAP Server Group and Address Binding

Use the following example to create the ICAP server group and bind the IP addresses:

```
configure
   context <icap_ctxt_name> [ -noconfirm ]
      content-filtering server-group <icap_svr_grp_name> [ -noconfirm ]
         origin address <ip_address>
         end
```

Notes:

- <*ip_address*> is local IP address of the CFSG endpoint.

Configuring ICAP Server and Other Parameters

Use the following example to configure the active content filtering (ICAP server) and other related parameters:

```
configure

   context <icap_context_name>

      content-filtering server-group <icap_server_grp_name>

         icap server <ip_address> [port <port_number>][max <max_msgs>][priority
<priority>]

         deny-message <msg_string>

         response-timeout <timeout>

         connection retry-timeout <retry_timeout>

         failure-action {allow | content-insertion <content_string> | discard |
redirect-url <url> | terminate-flow}

         dictionary {custom1 | custom2 | standard}

         end
```

Notes:

- In StarOS 8.1 and later, a maximum of five ICAP servers can be configured per Content Filtering Server Group. In StarOS 8.0, only one ICAP Server can be configured per Content Filtering Server Group.
- The maximum outstanding request per ICAP connection configured using the optional **max** *<max_msgs>* keyword is limited to one. Therefore, any other value configured using the **max** keyword will be ignored.
- *Optional*. To configure the ICAP URL extraction behavior, in the Content Filtering Server Group configuration mode, enter the following command:

  url-extraction { after-parsing | raw }

  By default, percent-encoded hex characters in URLs sent from the ACF client to the ICAP server will be converted to corresponding ASCII characters and sent.

Configuring ECS Rulebase for ICAP Server Group

Use the following example to configure the content filtering mode to ICAP server mode in the ECS rulebase for content filtering:

```
configure

   require active-charging [optimized-mode]
```

```
active-charging service <acs_svc_name> [-noconfirm]
   rulebase <rulebase_name> [-noconfirm]
      content-filtering mode server-group <cf_server_group>
      end
```

Notes:

- In StarOS 8.1, the **optimized-mode** keyword enables ACS in the Optimized mode, wherein ACS functionality is managed by SessMgrs. In StarOS 8.1, ACS must be enabled in the Optimized mode.
- In StarOS 8.3, the **optimized-mode** keyword is obsolete. With or without this keyword ACS is always enabled in Optimized mode.
- In StarOS 8.0 and StarOS 9.0 and later, the **optimized-mode** keyword is not available.

Configuring Charging Action for ICAP Server Group

Use the following example to configure the charging action to forward HTTP/WAP GET request to ICAP server for content processing:

```
configure
   active-charging service <acs_svc_name>
      charging-action <charging_action_name> [ -noconfirm ]
         content-filtering processing server-group
         end
```

Verifying the ICAP Server Group Configuration

This section explains how to display and review the configurations after saving them in a .cfg file as described in *Verifying and Saving Your Configuration* chapter of this guide and also to retrieve errors and warnings within an active configuration for a service.

Important: All commands listed here are under Exec mode. Not all commands are available on all platforms.

These instructions are used to verify the configuration for this feature.

Step 1 Verify your ICAP Content Filtering Server Group configuration by entering the following command in Exec Mode:

show content-filtering server-group

The following is a sample output. In this example, an ICAP Content Filtering server group named *icap_cfsg1* was configured.

```
Content Filtering Group:     icap_cfsg1

  Context:                     icap1

  Origin Address:              1.2.3.4

  ICAP Address(Port):          1.2.3.4(1344)

  Max Outstanding:             256

  Priority:                    1

  Response Timeout:      30(secs)      Connection Retry
Timeout:      30(secs)

  Dictionary:                   standard

  Timeout Action:              terminate-flow

  Deny Message:                "Service Not Subscribed"

  URL-extraction:               after-parsing

  Content Filtering Group Connections: NONE

Total content filtering groups mathing specified criteria:  1
```

Step 2 Verify any configuration error in your configuraiont by entering the following command in Exec Mode:

show configuration errors

Chapter 16
Intelligent Traffic Control

The product administration guides provide examples and procedures to configure 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.

Important: Intelligent Traffic Control is only available if you have purchased and installed a feature license Intelligent Traffic Controlon your system. If you have not previously purchased this enhanced feature, contact your sales representative for more information.

This chapter covers the following topics:

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 A features are only available if you have purchased and installed the session use EV-DO Rev A PDSN License on your system. If you have not previously purchased this enhanced feature, contact your sales representative for more information.

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.

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 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:

Step 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.

Step 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.

Step 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.

Step 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.

Step 5 Verify your flow-based traffic policing configuration by following the steps in the Verifying Flow-based Traffic Policing Configuration section.

Step 6 Save your configuration as described in the *Verifying and Saving Your Configuration* chapter.

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.

Chapter 17
Interchassis Session Recovery

This chapter provides information on configuring interchassis session recovery (ICSR). 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.interchassis session recovery.

This chapter discusses the following:

Overview

The interchassis session recovery feature provides the highest possible availability for continuous call processing without interrupting subscriber services. This is accomplished through the use of redundant chassis. The chassis are configured as primary and backup with one being active and one standby. Both chassis are connected to the same AAA server. A checkpoint duration timer is used to control when subscriber data is sent from the active chassis to the standby chassis. If the active chassis handling the call traffic goes out of service, the standby chassis transitions to the active state and continues processing the call traffic without interrupting the subscriber session. The chassis determines which is active through a propriety TCP-based connection called a redundancy link. This link is used to exchange Hello messages between the primary and backup chassis and must be maintained for proper system operation.


Important: ICSR is supported on chassis configured for GGSN or HA services only.

Interchassis Communication

Chassis configured to support interchassis session recovery communicate using periodic Hello messages. These messages are sent by each chassis to notify the peer of its current state. The Hello message contains information about the chassis such as its configuration and priority. A dead interval is used to set a time limit for a Hello message to be received from the chassis' peer. If the standby chassis does not receive an Hello message from the active chassis within the dead interval, the standby chassis transitions to the active state. In situations where the redundancy link goes out of service, a priority scheme is used to determine which chassis processes the session. The following priority scheme is used:

- router identifier
- chassis priority
- SPIO MAC address

Checkpoint Messages

Checkpoint messages are sent from the active chassis to the standby chassis. Checkpoint messages are sent at specific intervals and contain all the information needed to recreate the sessions on the standby chassis, if that chassis were to become active. Once a session exceeds the checkpoint duration, checkpoint data is collected on the session. The checkpoint parameter determines the amount of time a session must be active before it is included in the checkpoint message.

AAA Monitor

AAA servers are monitored using the authentication probe mechanism. AAA servers are considered up if the authentication-probe receives a valid response. AAA servers are considered down when the max-retries count specified in the configuration of the AAA server has been reached. The service-redundancy protocol will initiate a switchover when none of the configured AAA servers responds to an authentication probe. AAA probing is only be performed on the active chassis.

Important: A switchover event caused by a AAA monitoring failure is non-revertible. If the newly active chassis fails to monitor the configured AAA servers it remains as the active chassis until either a manual switchover, or another non-AAA failure event causes the system to switchover.

BGP Interaction

The service-redundancy protocol implements non-revertible switchover behavior by using a mechanism to adjust the route modifier value for the advertised loopback/IP Pool routes. The initial value of the route modifier value is determined by the chassis configured role and is initialized to a value that is higher than a normal operational value. This ensures that in the event of an SRP link failure and a SRP task failure that the correct chassis is still preferred in the routing domain. The Active and Standby chassis share the route modifier values they are currently using. When BGP advertises the loopback and ip pool routes, it converts the route modifier into an autonomous systems (AS) path prepend count. The Active chassis always has a lower route modifier, and thus prepends less to the AS-path attribute. This causes the route to be preferred in the routing domain. In the event that communication on the redundancy link is lost, and both chassis in the redundant pair are claiming to be Active. The previously Active chassis is still preferred since it is advertising a smaller AS-path into the BGP routing domain. The route modifier is incremented as switchover events occur. A threshold will be implemented to determine when the route modifier should be reset to its initial value to avoid rollover.

Requirements

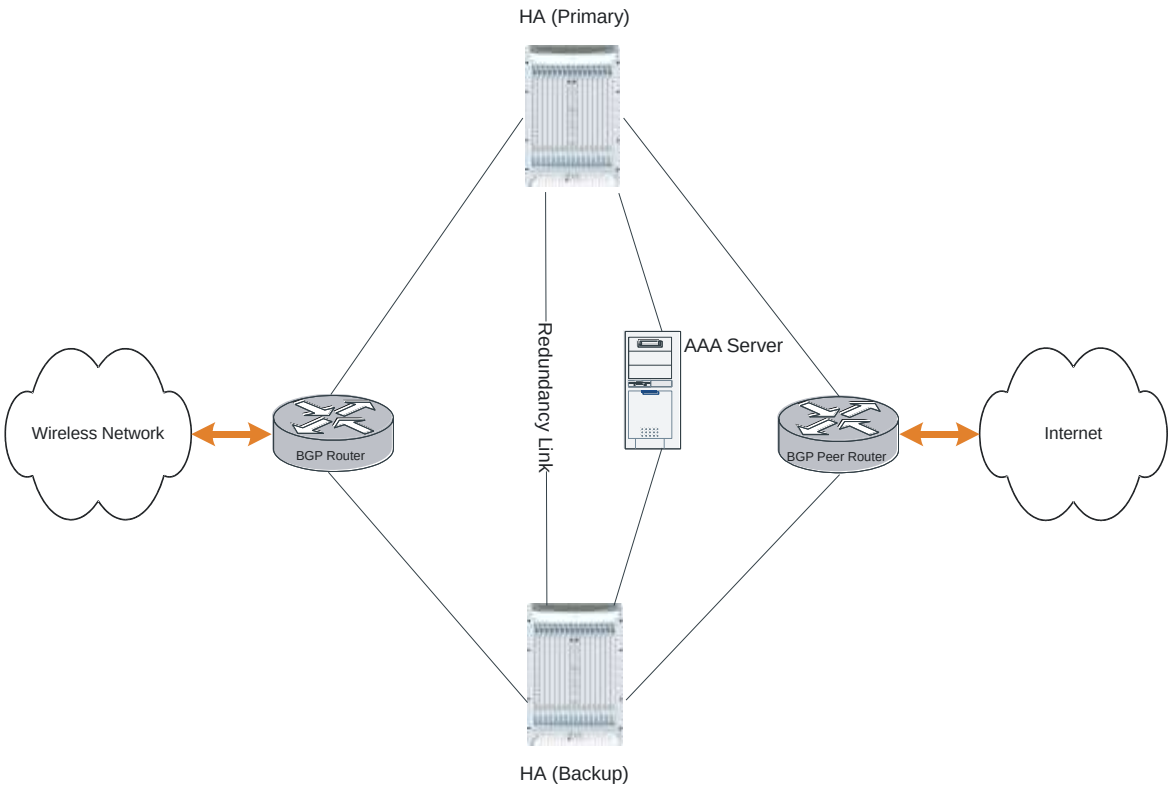
ICSR configurations require the following:

- Two chassis configured for the same service types- The services must be bound on an srp activated loopback interface.
- Three contexts:
  - Redundancy - to configure the primary and backup chassis redundancy.
  - Source - AAA configuration the specified nas-ip-address must be the IP address of an interface bound to an HA or any core network service configured within the same context.
  - Destination - to configure monitoring and routing to the PDN.
- AAA RADIUS server
- Border Gateway Protocol (BGP) - ICSR uses the route modifier to determine the chassis priority.

Important: ICSR is a licensed feature. Be sure that each chassis has the appropriate license before using the procedures in this chapter. To do this, log in to both chassis and execute a **`show license information`** command. Interchassis Session Recovery feature is listed as **`Inter-Chassis Session Recovery`**. If the chassis is not licensed, please contact your local sales representative.

The following figure shows an ICSR network.

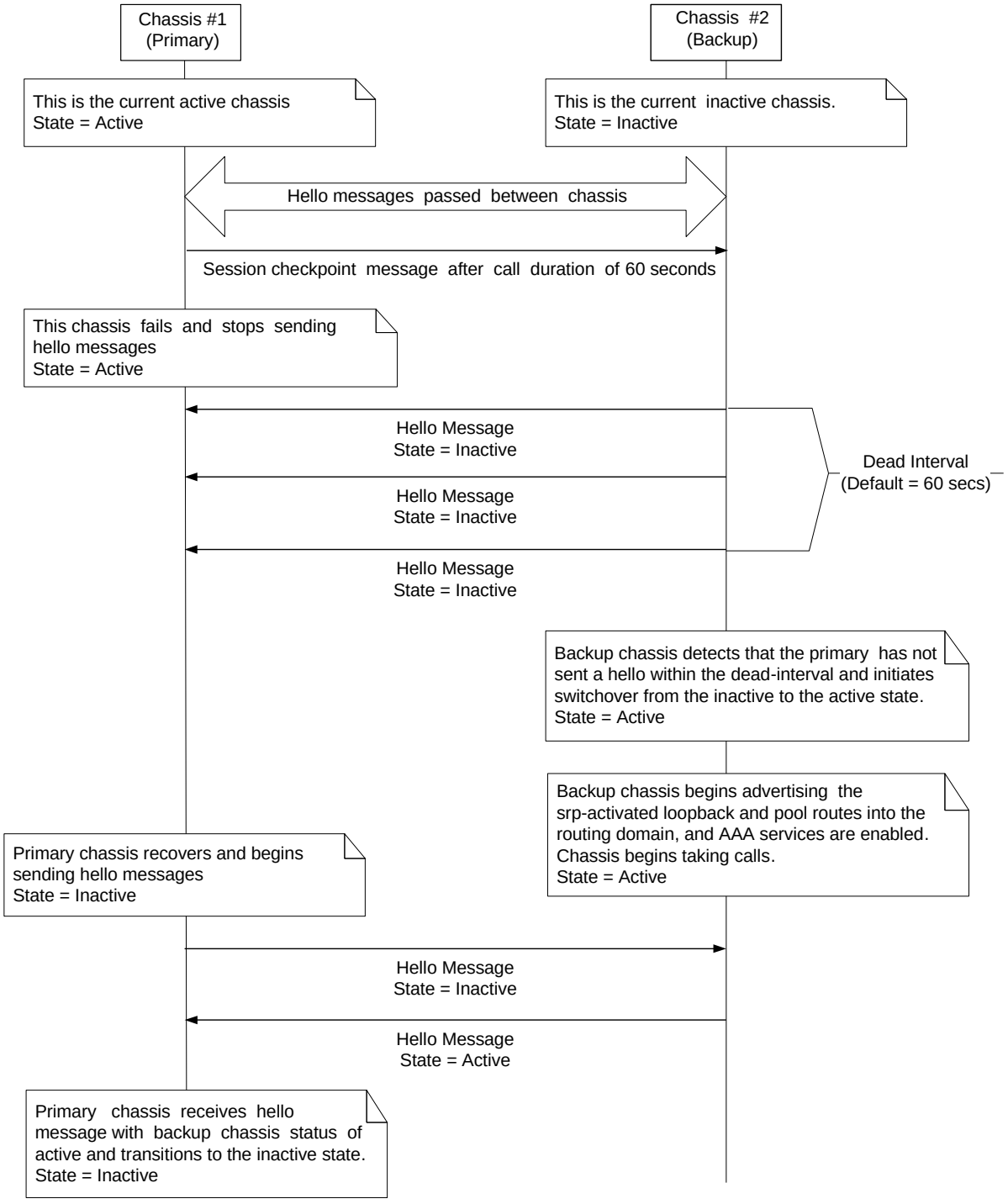
Figure 16. Interchassis Session Recovery Network Diagram



ICSR Operation

This section provides an operational flow for ICSR. The following figure shows an ICSR process flow.

Figure 17. ICSR Flow Diagram



Chassis Initialization

When the chassis are simultaneously initialized, they send Hello messages to their configured peer. The peer sends a response, establishes communication between the chassis, and messages are sent that contain configuration information.

Important: If the chassis are GGSNs, the messages include the APN tables.

During initialization, if both chassis are misconfigured in the same mode - both active (primary) or both standby (backup), then the chassis with the highest priority (highest number set with SRP **`priority`**) becomes active and the other chassis becomes the standby.

If the chassis priorities are the same, the system compares the two MAC addresses and the chassis with the higher SPIO MAC address becomes active. For example, if the chassis have MAC addresses of *`00-02-43-03-1C-2B`* and *`00-02-43-03-01-3B`*, the last 3 sets of octets (the first 3 sets are the vendor code) are compared. In this example, the *`03-1C-2B`* and *`03-01-3B`* are compared from left to right. The first pair of octets in both MAC addresses are the same, so the next pairs are compared. Since the *`01`* is lower than the *`1C`*, the chassis with the SPIO MAC address of *`00-02-43-03-1C-2B`* becomes active and the other chassis the standby.

Chassis Operation

This section describes how the chassis communicate, maintain subscriber sessions, and perform chassis switchover.

Chassis Communication

There is one chassis in the active state and one in the standby state. They both send Hello messages at each hello interval. Subscriber sessions that exceed the checkpoint session duration are included in checkpoint messages that are sent to the standby chassis. The checkpoint message contains subscriber session information so if the active chassis goes out of service, the backup chassis becomes active and is able to continue processing the subscriber sessions. Additional checkpoint messages occur at various intervals where subscriber session information is updated on the standby chassis.

Chassis Switchover

If the active chassis goes out of service the standby chassis continues to send Hello messages. If the standby chassis does not receive a response to the Hello messages within the dead interval, the standby chassis initiates a switchover. During the switch over, the standby chassis begins advertising the srp-activated loopback and pool routes into the routing domain. Once the chassis becomes active, it continues to process existing AAA services, subscriber sessions that had checkpoint information, and is able to establish new subscriber sessions as well.

When the primary chassis is back in service it sends Hello messages to the configured peer. The peer sends a response, establishes communication between the chassis, and Hello messages are sent that contain configuration information. The primary chassis receives an Hello message that shows the backup chassis state as active and the primary chassis becomes standby. The Hello message now continue to be sent to each peer and checkpoint information is now sent from the active chassis to the standby chassis at regular intervals.

When chassis switchover occurs, the session timers are recovered. The MIP HA session recovery is recreated with the full lifetime to avoid potential loss of the session and the possibility that a renewal update was lost in the transient checkpoint update process.

Configuring Interchassis Session Recovery (ICSR)

Important: The ICSR configuration must be the same on the primary and backup chassis. If each chassis has a different srp configuration, the session recovery feature does not function and sessions cannot be recovered when the active chassis goes out of service.

Important: This section provides the minimum instruction set for configuring ICSR on the system. For more information on commands that configure additional parameters and options, refer to the *Command Line Interface Reference*.

Procedures described here assume the following:

- The chassis have been installed and configured with core network services.

  For more configuration information and instructions on configuring services, refer to the respective product Administration Guide.
- In addition, the IP address pools must be **`srp activated`**.
- AAA server is installed and configured.

  For more configuration information and instructions on configuring the AAA server, refer to the *AAA Interface Administration and Reference*.
- BGP router installed and configured. See *Routing* for more information on configuring BGP services.

To configure the Interchassis Session Recovery on a primary and/or backup chassis:

Step 1 Configure the service redundancy protocol context by applying the example configuration in the Configuring the Service Redundancy Protocol (SRP) Context section.

Step 2 Modify the source context of core network service for ICSR by applying the example configuration in the Modifying the Source Context for ICSR section.

Step 3 Modify the destination context of core network service for ICSR by applying the example configuration in the Modifying the Destination Context for ICSR section.

Step 4 Optional. Disable the bulk statistics collection on standby system by applying the example configuration in the Disabling Bulk Statistics Collection on a Standby System section.

Step 5 Verify your primary and backup chassis configuration by following the steps in the Verifying the Primary and Backup Chassis Configuration section.

Step 6 Save your configuration as described in the *Verifying and Saving Your Configuration* chapter.

Configuring the Service Redundancy Protocol (SRP) Context

To configure the system to work for interchassis session recovery:

Step 1 Create the chassis redundancy context and bind it to primary chassis IP address by applying the example configuration in the Creating and Binding the SRP Context section.

Step 2 Configure the chassis redundancy context with priority, chassis mode, heart-beat interval, dead-interval and peer IP address by applying the example configuration in the Configuring the SRP Context Parameters section.

Step 3 Configure the SRP context with interface parameters, like interface name, IP address and port number to communicate with other chassis by applying the example configuration in the Configuring the SRP Context Interface Parameters section.

Step 4 Verify your SRP context configuration by following the steps in the Verifying SRP Configuration section.

Step 5 Save your configuration as described in the *Verifying and Saving Your Configuration* chapter.

Creating and Binding the SRP Context

Use the following example to create the SRP context bind it to primary chassis IP address:

Important: ICSR is configured using two systems. Be sure to create the redundancy context on both systems. CLI commands must be executed on both systems. Always make configuration changes on the primary system first. It would be a good idea to log on both chassis before continuing. Before starting this configuration, determine which system to configure as the primary and use that login session.

```
configure
   context <srp_ctxt_name> [ -noconfirm ]
      service-redundancy-protocol
         bind address <ip_address>
         end
```

Notes:

- ICSR should be configured and maintained in a separate context.
- Be sure to bind the local IP address to the primary chassis. When configuring the backup chassis, be sure to bind the local IP address to the backup chassis.

Configuring the SRP Context Parameters

This configuration assign a chassis mode, priority, and configure the redundancy link between the primary and backup systems:

Important: CLI commands must be executed on both systems. Always make configuration changes on the primary system first. It would be a good idea to log on both chassis before continuing.

```
configure
   context <srp_ctxt_name>
      service-redundancy-protocol
         chassis-mode { primary | backup }
         priority <priority>
         peer-ip-address <ip_address>
         hello-interval <dur_sec>
         dead-interval <dead_dur_sec>
         end
```

Notes:

- ICSR should be configured and maintained in a separate context.
- When assigning the chassis mode on the backup chassis be sure to enter backup.
- The priority is used to determine which chassis becomes active when the redundancy link goes out of service. The higher priority chassis has the lower number. Be sure to assign different priorities to each chassis.
- Be sure to use the backup chassis IP address as the peer to the primary chassis. Use the primary chassis IP address as the peer to the backup chassis.
- The dead interval must be a higher value than the hello interval. The dead interval should be at least three times greater than the hello interval. For example, if the hello interval is 10, the dead interval should be at least 30. System performance is severely impacted if the hello interval and dead interval are not set properly.

Configuring the SRP Context Interface Parameters

This procedure configures communication interface with IP address and port number for the SRP context to communicate with chassis:

Important: CLI commands must be executed on both systems. Always make configuration changes on the primary system first. It would be a good idea to log on both chassis before continuing.

```
configure
   context <vpn_ctxt_name> [ -noconfirm ]
      interface <srp_if_name>
         ip-address { <ip_address> | <ip_address>/<mask> }
```

```
      exit
   exit
port ethernet <slot_num>/<port_num>
   description <des_string>
   medium { auto | speed { 10 | 100 | 1000 } duplex { full | half } }
   no shutdown
   bind interface <srp_if_name> <srp_ctxt_name>
   end
```

Notes:

Verifying SRP Configuration

Step 1 Verify that your SRP contexts were created and configured properly by entering the following command in Exec Mode:

```
show srp info
```

The output of this command given below is the sample output. In this example, a SRP context called *srp1* was configured and you can observe some parameters configured as default.

```
Service Redundancy Protocol:
------------------------------------------------------------------------
Context: srp1
Local Address: 0.0.0.0
Chassis State: Init
Chassis Mode: Backup
Chassis Priority: 125
Local Tiebreaker: 00-00-00-00-00-00
Route-Modifier: 34
Peer Remote Address: 0.0.0.0
Peer State: Init
Peer Mode: Init
```

```
Peer Priority: 0

Peer Tiebreaker: 00-00-00-00-00-00

Peer Route-Modifier: 0

Last Hello Message received: -

Peer Configuration Validation: Initial

Last Peer Configuration Error: None

Last Peer Configuration Event: -

Connection State: None
```

Modifying the Source Context for ICSR

To modify the source context of core service:

Step 1 Add the Border Gateway Protocol (BGP) router AS-path and configure the HA IP address, neighbor IP address, remote IP address in source context, where the core network service is configured, by applying the example configuration in the Configuring BGP Router and HA Address section.

Step 2 Configure the service redundancy context with BGP neighbor context and IP address to monitor the BGP link activity by applying the example configuration in the Configuring SRP Context for BGP section.

Step 3 Verify your BGP context configuration by following the steps in the Verifying BGP Configuration section.

Step 4 Save your configuration as described in the *Verifying and Saving Your Configuration* chapter.

Configuring BGP Router and HA Address

Use the following example to create the BGP context and network addresses.

```
configure
   context <source_ctxt_name>
      router bgp <AS_num>
         network <ha_ip_address>
         neighbor <neighbor_ip_address> remote-as <AS_num>
         end
```

Notes:

- Source context is the context where core network service is configured.

Configuring SRP Context for BGP

Use the following example to configure the BGP context and IP addresses in SRP context.

```
configure
   context <srp_ctxt_name>
      service-redundancy-protocol
         monitor bgp context <source_ctxt_name> <neighbor_ip_address>
         end
```

Notes:

Verifying BGP Configuration

Step 1 Verify your BGP configuration by entering the following command in Exec Mode:

```
show srp monitor bgp
```

Modifying the Destination Context for ICSR

To modify the destination context of core service:

Step 1 Add the Border Gateway Protocol (BGP) router and configure the HA IP address, neighbor IP address, remote IP address in destination context, where the core network service is configured, by applying the example configuration in the Configuring BGP Router and HA Address in Destination Context section.

Step 2 Configure the service redundancy context with BGP neighbor context and IP address to monitor the BGP link activity by applying the example configuration in the Configuring SRP Context for BGP for Destination Context section.

Step 3 Set the subscriber mode to default by following the steps in the Setting Subscriber to Default Mode section.

Step 4 Verify your BGP context configuration by following the steps in the Verifying BGP Configuration in Destination Context section.

Step 5 Save your configuration as described in the *Verifying and Saving Your Configuration* chapter.

Configuring BGP Router and HA Address in Destination Context

Use the following example to create the BGP context and network addresses.

```
configure
   context <dest_ctxt_name>
      router bgp <AS_num>
         network <ha_ip_address>
         neighbor <neighbor_ip_address> remote-as <AS_num>
         end
```

Notes:

- AS-path number is the autonomous systems path number for this BGP router.

Configuring SRP Context for BGP for Destination Context

Use the following example to configure the BGP context and IP addresses in SRP context.

```
configure
   context <srp_ctxt_name>
      service-redundancy-protocol
         monitor bgp context <dest_ctxt_name> <neighbor_ip_address>
         end
```

Notes:

Setting Subscriber to Default Mode

Use the following example to set the subscriber mode to Default.

```
configure
   context <dest_ctxt_name>
      subscriber default
      end
```

Notes:

Verifying BGP Configuration in Destination Context

Step 1 Verify your BGP configuration by entering the following command in Exec Mode:

```
show srp monitor bgp
```

Disabling Bulk Statistics Collection on a Standby System

You can optionally configure bulk statistics not to be collected from a system when it is in the standby mode of operation.

Important: When this feature is enabled and a system transitions to standby state any pending accumulated statistics data is transferred at the first opportunity. After that no additional statistics gathering takes place until the system comes out of standby state.

Use the following example to disable the bulk statistics collection on a standby system.

```
configure
   bulkstat mode
      no gather-on-standby
      end
```

Notes:

- Repeat this procedure for both systems.

Verifying the Primary and Backup Chassis Configuration

These instructions are used to compare the ICSR configuration on both chassis.

Step 1 Enter the following command on both chassis at the Exec mode:

Verify that both chassis have the same srp configuration information. The output looks similar to following:

```
config
 context source
```

```
 interface haservice loopback

 ip address 172.17.1.1 255.255.255.255 srp-activate

 #exit

 radius attribute nas-ip-address address 172.17.1.1

 radius server 192.168.83.2 encrypted key 01abd002c82b4a2c port 1812

 radius accounting server 192.168.83.2 encrypted key 01abd002c82b4a2c
port 1813

 ha-service ha-pdsn

 mn-ha-spi spi-number 256 encrypted secret
6c93f7960b726b6f6c93f7960b726b6f hash-algorithm md5

 fa-ha-spi remote-address 192.168.82.0/24 spi-number 256 encrypted secret
1088bdd6817f64df

bind address 172.17.1.1

 #exit

 #exit

 context destination

 ip pool dynamic 172.18.0.0 255.255.0.0 public 0 srp-activate

 ip pool static 172.19.0.0 255.255.240.0 static srp-activate

 #exit

 context srp

 service-redundancy-protocol

 #exit

 #exit

end
```

Chapter 18
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 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: Use of RoHC requires that a valid RoHC license key be installed. Contact your local Sales or Support representative for information on how to obtain a license.

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 as described in the *Saving Your Configuration* chapter.

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: Use of RoHC requires that a valid license key be installed. Contact your local Sales or Support representative for information on how to obtain a license.

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:

Step 1 Enable RoHC header compression by applying the example configuration in the Enabling RoHC Header Compression for PPP section.

Step 2 Verify your RoHC header compression configuration by following the steps in the Verifying the Header Compression Configuration section.

Step 3 Save your configuration as described in the *Saving Your Configuration* chapter.

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: Use of RoHC requires that a valid license key be installed. Contact your local Sales or Support representative for information on how to obtain a license.

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 th *Command Line Interface Reference*.

To configure the system to enable both RoHC and VJ header compression to IP headers:

Step 1 Enable the RoHC and VJ header compression by applying the example configuration in the Enabling RoHC and VJ Header Compression for PPP section.

Step 2 Verify your RoHC and VJ header compression configuration by following the steps in the Verifying the Header Compression Configuration section.

Step 3 Save your configuration as described in the *Saving Your Configuration* chapter.

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.

- 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 Service

This section explains how to set RoHC settings in the PDSN 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 chapter in Command Line Interface Reference.

To configure the system to disable the RoHC header compression feature at the PDSN Service over SO67:

Step 1 Disable header compression by applying the example configuration in the Enabling ROHC Header Compression with PDSN section.

Step 2 Verify your RoHC configuration by following the steps in the Verifying the Header Compression Configuration section.

Step 3 Save your configuration as described in the *Saving Your Configuration* chapter.

Enabling RoHC Header Compression with PDSN

Use the following example to disable 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.

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: Use of RoHC requires that a valid license key be installed. Contact your local Sales or Support representative for information on how to obtain a license.

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 as described in the *Saving Your Configuration* chapter.

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 appling 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 as described in the *Saving Your Configuration* chapter.

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 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 chapter in Command Line Interface Reference.

To configure the system to disable the IP header compression feature at the PDSN 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 as described in the *Saving Your Configuration* chapter.

Disabling RoHC Header Compression

Use the following example to disable the header compression over PPP:

```
configure
   context <ctxt_name>
      pdsn-service <svc_name>
         no ip header-compression RoHC
         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 disable RoHC over SO67.

Verifying the Header Compression Configuration

These instructions are used to verify the header compression configuration.

- 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 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.

Chapter 19
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

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 18. IPSP Operation for New Sessions

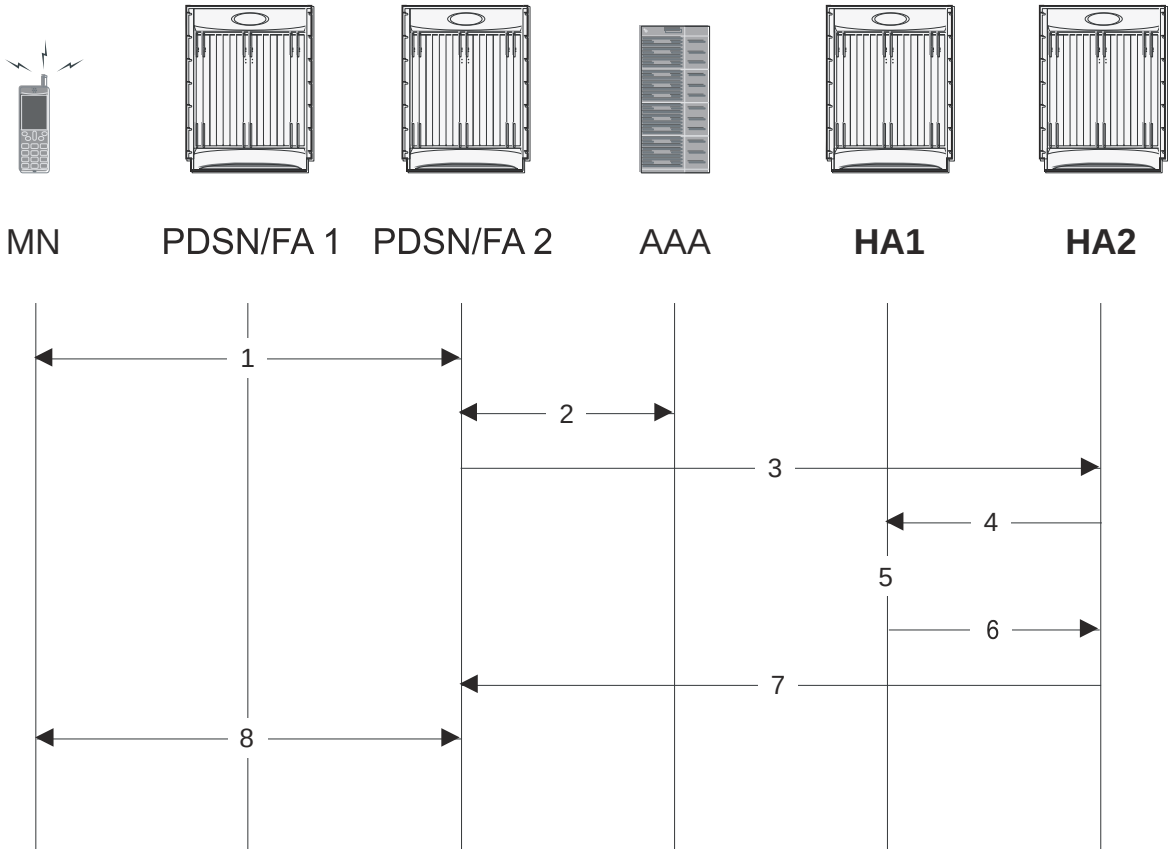


Table 12. IPSP Operation for New Sessions Description

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 19. IPSP Operation for Session Handoffs

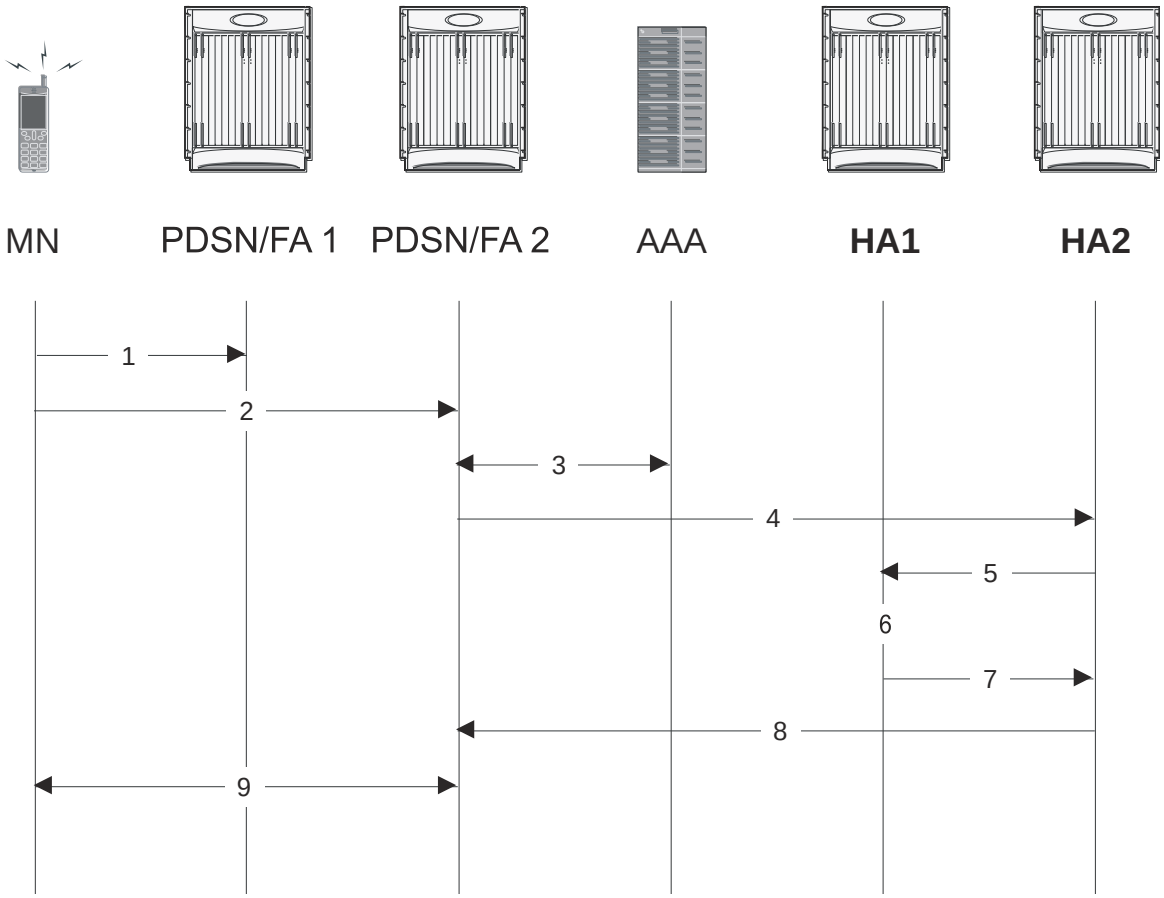


Table 13. IPSP Operation for Session Handoffs Description

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.

Step	Description
7	HA1 allocates the original IP address to HA2 for the session.
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 as described in the *Saving Your Configuration* chapter.

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 *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.

Step 1 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 *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 as described in the *Saving Your Configuration* chapter.

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.

Chapter 20
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.

IP Security is a license enabled feature. You must purchase and install a license key before you can use this feature.

Caution: IPSec parameter configurations saved using this release may not function properly with older software releases.

This chapter contains the following sections:

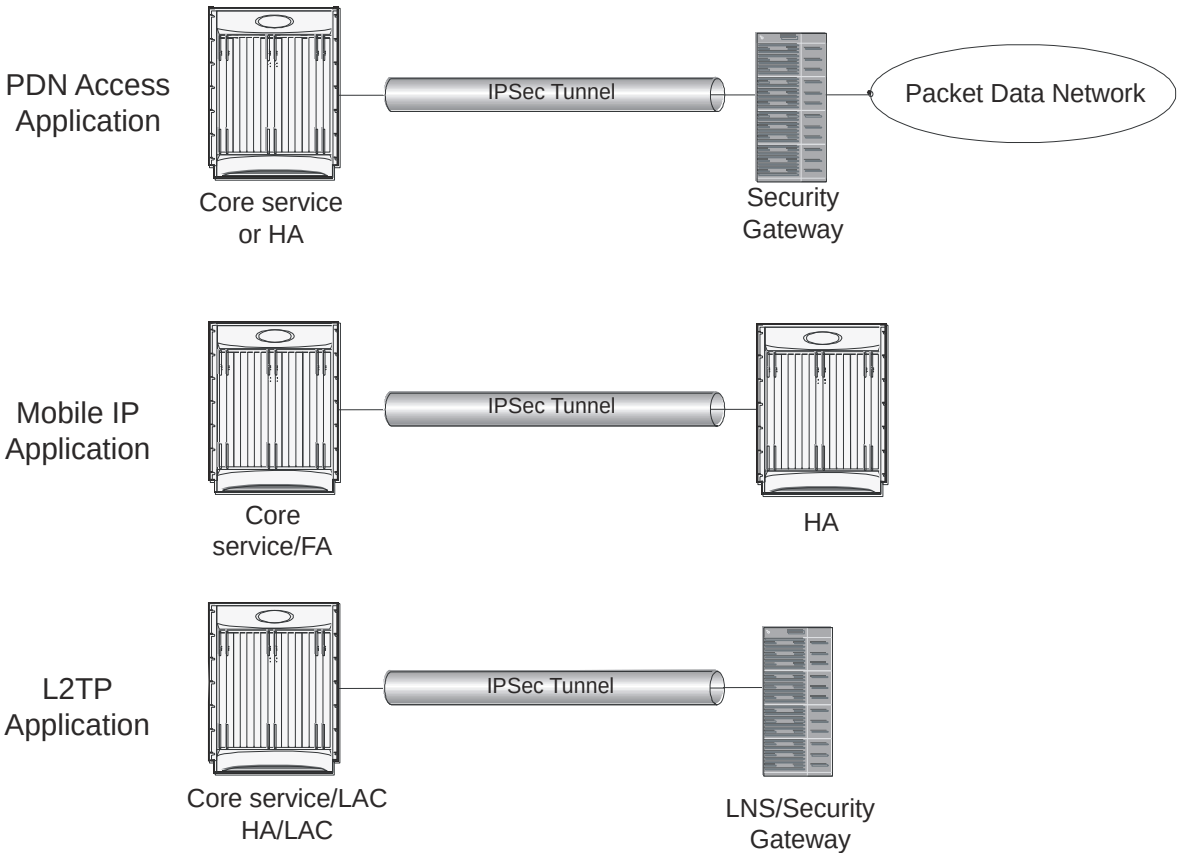
- APN Template Configuration to Support L2TP

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 20. 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.

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	• 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	• 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	• 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 21. IPSec PDN Access Processing

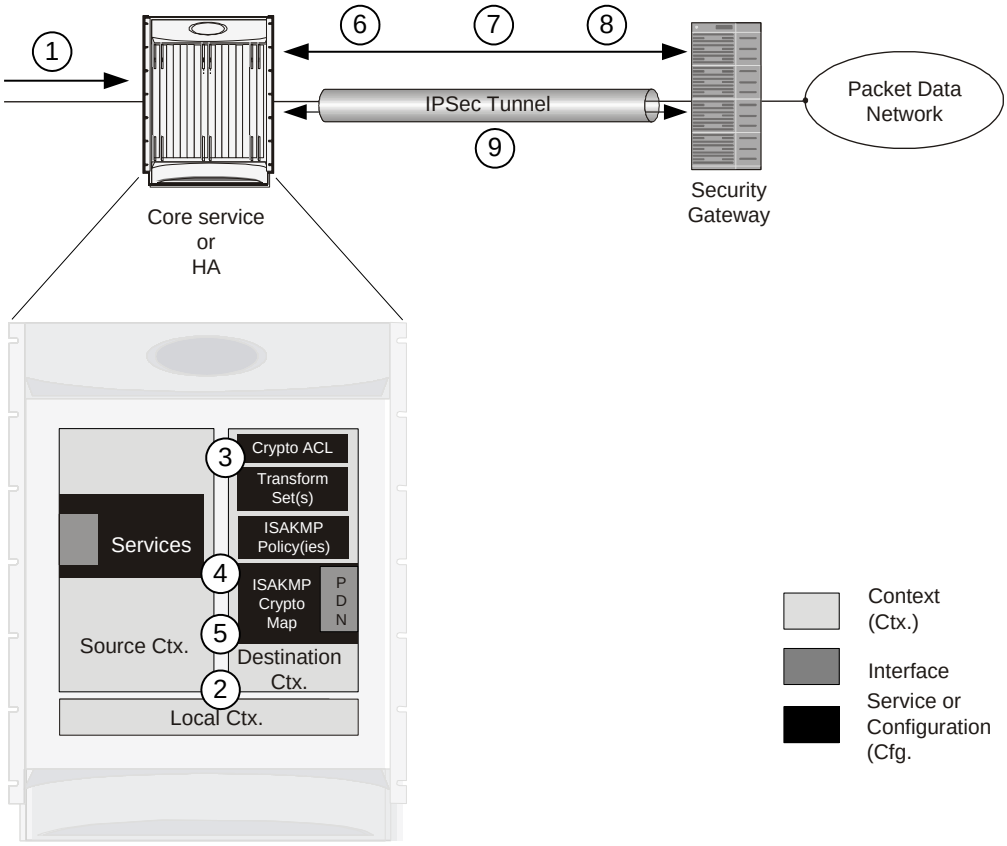


Table 14. IPSec PDN Access Processing

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).
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: • 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 as described in *Verifying and Saving Your Configuration.*

Implementing IPSec for Mobile IP Applications

This section provides information on the following topics:

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 22. IPSec-based Mobile IP Session Processing

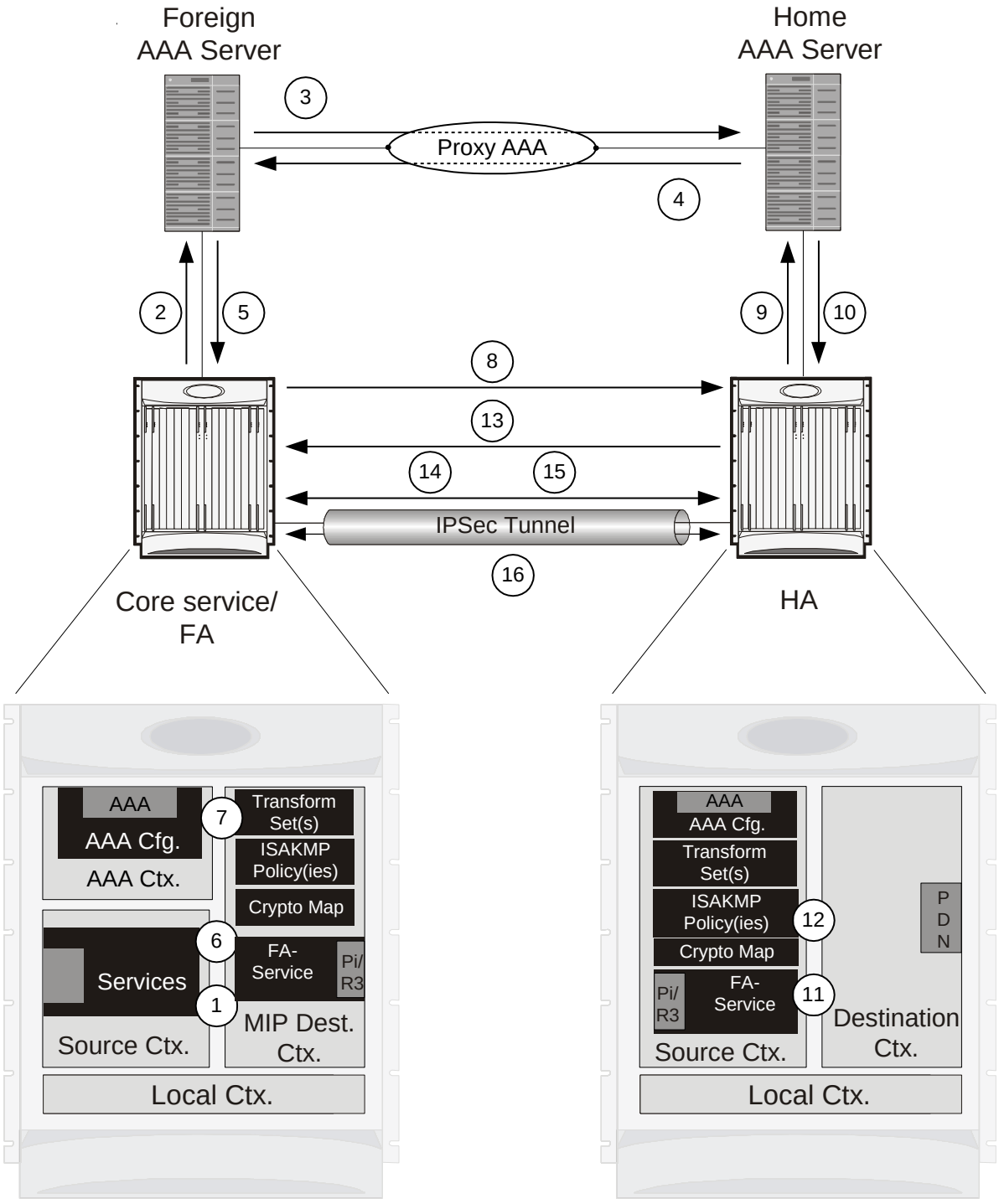


Table 15. IPSec-based Mobile IP Session Processing

Step	Description

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: • 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: • 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: • 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: • 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.	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: • 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 or 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 or 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 as described in *Verifying and Saving Your Configuration.*

Implementing IPSec for L2TP Applications

This section provides information on the following topics:

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 23. Attribute-based L2TP, IPSec-Encrypted Session Processing

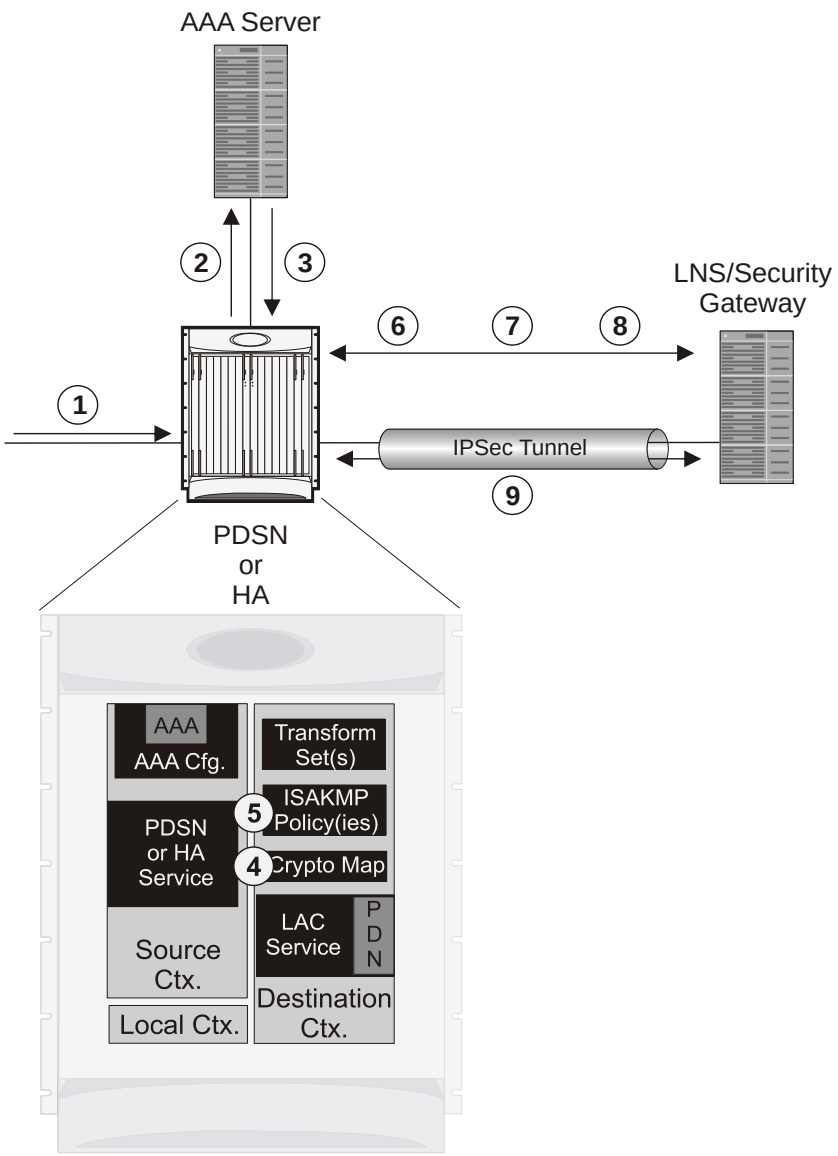


Table 16. Attribute-based L2TP, IPSec-Encrypted Session Processing

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: • 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 as described in *Verifying and Saving Your Configuration.*

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 24. PDSN Compulsory L2TP, IPSec-Encrypted Session Processing

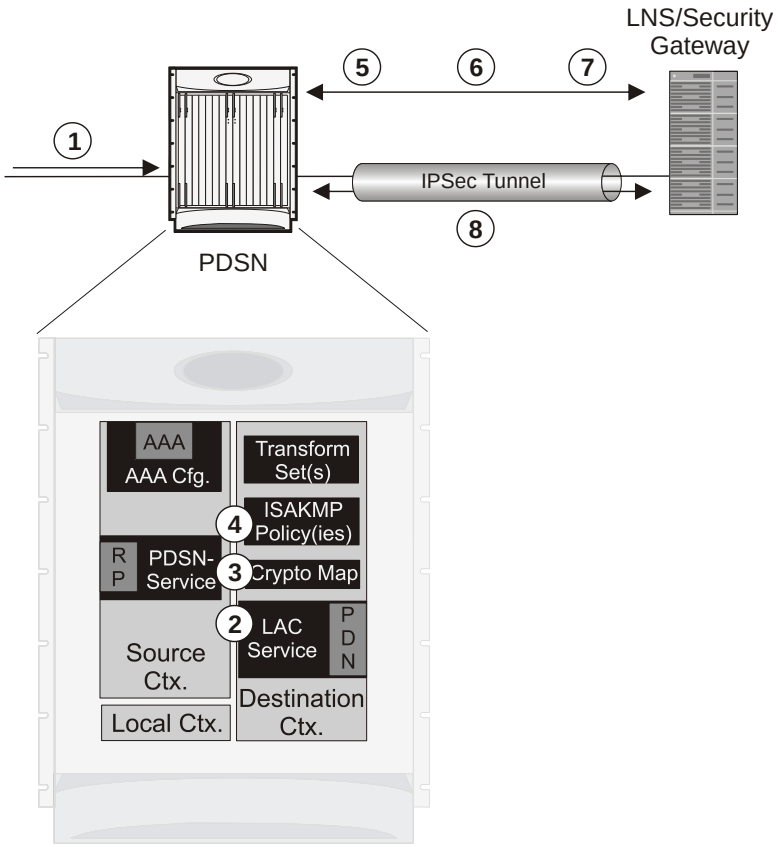


Table 17. PDSN Compulsory L2TP, IPSec-Encrypted Session Processing

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: • 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: • 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 as described in *Verifying and Saving Your Configuration.*

How IPSec is Used for L2TP Configurations on the GGSN

and the text that follows describe how IPSec-encrypted attribute-based L2TP sessions are processed by the system.

Figure 25. GGSN PDP Context Processing with IPSec-Encrypted L2TP

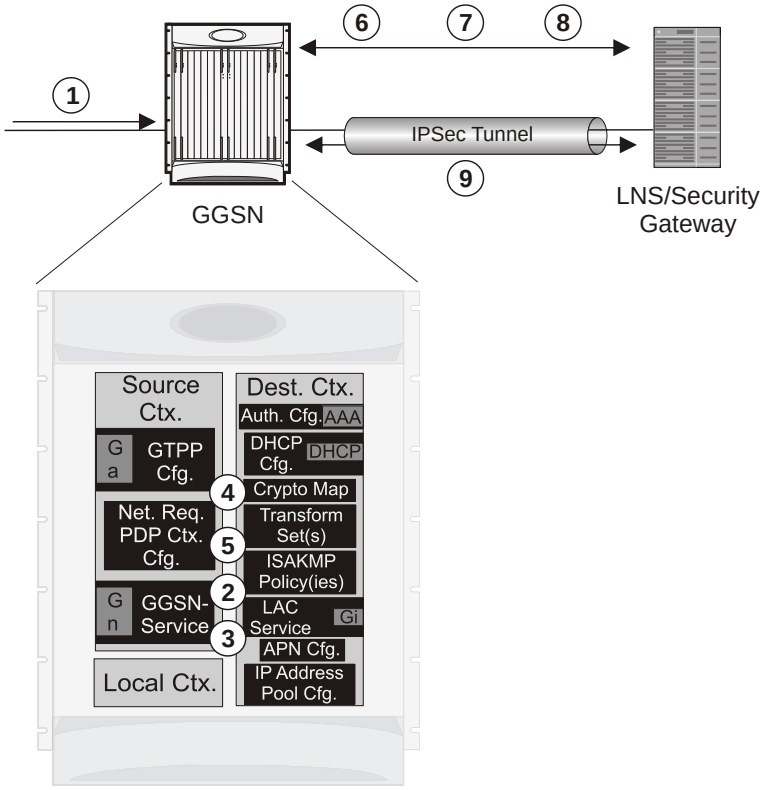


Table 18. GGSN PDP Context Processing with IPSec-Encrypted L2TP

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.
4.	From the crypto map, the system determines the following: • 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

Step	Description
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 as described in *Verifying and Saving Your Configuration.*

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 as described in the *Verifying and Saving Your Configuration* chapter.

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 as described in the *Verifying and Saving Your Configuration* chapter.

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 I*SAKMP 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 as described in the *Verifying and Saving Your Configuration* chapter.

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 as described in the *Verifying and Saving Your Configuration* chapter.

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 as described in the *Verifying and Saving Your Configuration* chapter.

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.
- <*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: sdfsdfasdf342d32

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 as described in the *Verifying and Saving Your Configuration* chapter.

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 crypot 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 as described in the *Verifying and Saving Your Configuration* chapter.

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
<preshared_secret> ]
         isakmp default crypto-map <map_name> [ secret <preshared_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 crypot 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 as described in the *Verifying and Saving Your Configuration* chapter.

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
<preshared_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 crypot 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 19. 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 as described in the *Verifying and Saving Your Configuration* chapter.

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 crypot 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 name service_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 20. 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 as described in the *Verifying and Saving Your Configuration* chapter.

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 as described in the *Verifying and Saving Your Configuration* chapter.

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:

```
sshow 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 as described in the *Verifying and Saving Your Configuration* chapter.

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:

```
sshow 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 as described in the *Verifying and Saving Your Configuration* chapter.

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.

Step 1 Verify that your APN is configured properly with L2TP by entering the following command in Exec Mode in specific context:

```
show apn { all | name apn_name }
```

The output of this command is a concise listing of FA service parameter settings configured on the system.

Chapter 21
L2TP Access Concentrator

This chapter describes the Layer 2 Tunneling Protocol (L2TP) Access Concentrator (LAC) functionality support on ST16 and Cisco® ASR 5000 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: This product requires the purchase of a separate session licence and feature key in order to function as described.

When enabled though 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.

L2TP Session and Tunnel Capacities

The system is capable of supporting L2TP tunnels for all subscriber sessions or on a session-by-session basis.

Each L2TP tunnel can facilitate one or more subscriber sessions. The number of supported L2TP sessions and tunnels corresponds to the number of active Packet Accelerator Cards (PACs) or Packet Services Cards (PSCs)

The following tables lists PDSN Simple IP tunnel and session capacities for the ST16 and ASR 5000 respectively.

Important: The capacities stated in this section are the maximum supported by the components. Actual capacities are dependent upon hardware and software configurations.

Table 21. L2TP LAC Tunnel and Session Capacity per ST16 for PDSN Simple IP

	Capacity/PAC	Capacity/Full System1	Minimum # of LAC services Required2
Tunnels	32,000	32,000	1
Sessions	40,000	500,000	1
Transactions/Second	4003	40004	N/A
1. A full system consists of 13 active PACs.			
2. Multiple LAC services can be configured to reduce the amount of traffic over a single interface.			
3. Transactions include PPP and LAC			
4. Transactions include PPP and LAC session setup assuming that a tunnel already exists.			

Table 22. L2TP LAC Tunnel and Session Capacity per ASR 5000 for PDSN Simple IP

	Capacity/PSC	Capacity/Full System1	Minimum # of LAC services Required2
Tunnels	32,000	32,000	1
Sessions	120,000	1,500,000	1
Transactions/Second	8003	10,0004	N/A
1. A full system consists of 13 active PSCs.			
2. Multiple LAC services can be configured to reduce the amount of traffic over a single interface.			
3. Transactions include PPP and LAC			
4. Transactions include PPP and LAC session setup assuming that a tunnel already exists.			

The following table lists HA tunnel and session capacities for the ST16 and ASR 5000 respectively.

Table 23. L2TP LAC Tunnel and Session Capacity per ST16 for HA

	Capacity/PAC	Capacity/Full System1	Minimum # of LAC services Required2
Tunnels	32,000	32,000	1
Sessions	25,000	300,000	1
Transactions/Second	2503	30004	N/A
1. A full system consists of 13 active PACs.			
2. Multiple LAC services can be configured to reduce the amount of traffic over a single interface.			
3. Transactions include PPP and LAC			
4. Transactions include PPP and LAC session setup assuming that a tunnel already exists.			

Table 24. L2TP LAC Tunnel and Session Capacity per ASR 5000 for HA

	Capacity/PSC	Capacity/Full System1	Minimum # of LAC services Required2
Tunnels	32,000	32,000	1
Sessions	83,000	1,000,000	1
Transactions/Second	5003	6,0004	N/A
1. A full system consists of 13 active PSCs.			
2. Multiple LAC services can be configured to reduce the amount of traffic over a single interface.			
3. Transactions include PPP and LAC			
4. Transactions include PPP and LAC session setup assuming that a tunnel already exists.			

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	• *Supported LAC Service Configurations for PDSN Simple IP*
	• *Supported LAC Service Configuration for Mobile IP*
	• *Configuring Subscriber Profiles for L2TP Support*
	  • *RADIUS and Subscriber Profile Attributes Used*
	  • *Configuring Local Subscriber Profiles for L2TP Support*
	  • *Tunneling All Subscribers in a Specific Context Without Using RADIUS Attributes*
	• *Configuring LAC Services*
	• *Modifying PDSN Services for L2TP Support*

GGSN/SGSN/FA	• *Supported LAC Service Configurations for the GGSN*
	• *Supported LAC Service Configuration for Mobile IP*
	• *Configuring Subscriber Profiles for L2TP Support*
	  • *RADIUS and Subscriber Profile Attributes Used*
	  • *Configuring Local Subscriber Profiles for L2TP Support*
	• *Configuring LAC Services*
	• *Modifying APN Templates to Support L2TP*

ASN GW	• *Supported LAC Service Configuration for Mobile IP*
	• *Configuring Subscriber Profiles for L2TP Support*
	  • *RADIUS and Subscriber Profile Attributes Used*
	  • *Configuring Local Subscriber Profiles for L2TP Support*
	  • *Tunneling All Subscribers in a Specific Context Without Using RADIUS Attributes*
	• *Configuring LAC Services*

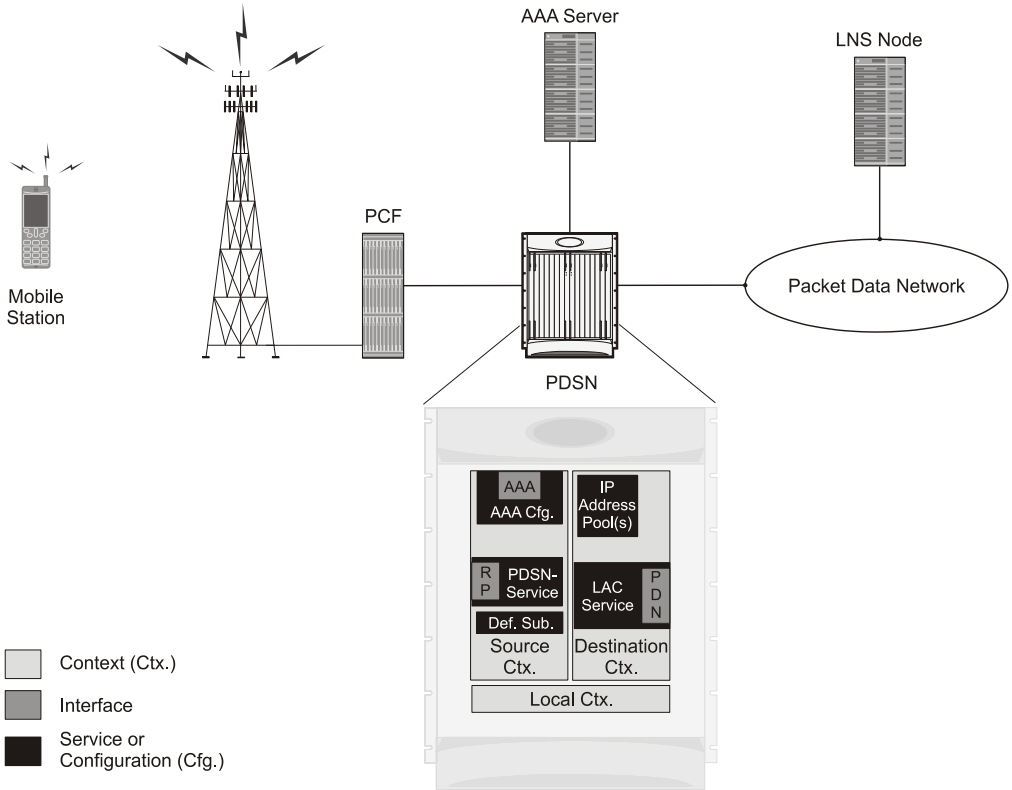
Supported LAC Service Configurations for PDSN Simple IP

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 26. LAC Service Configuration for SIP



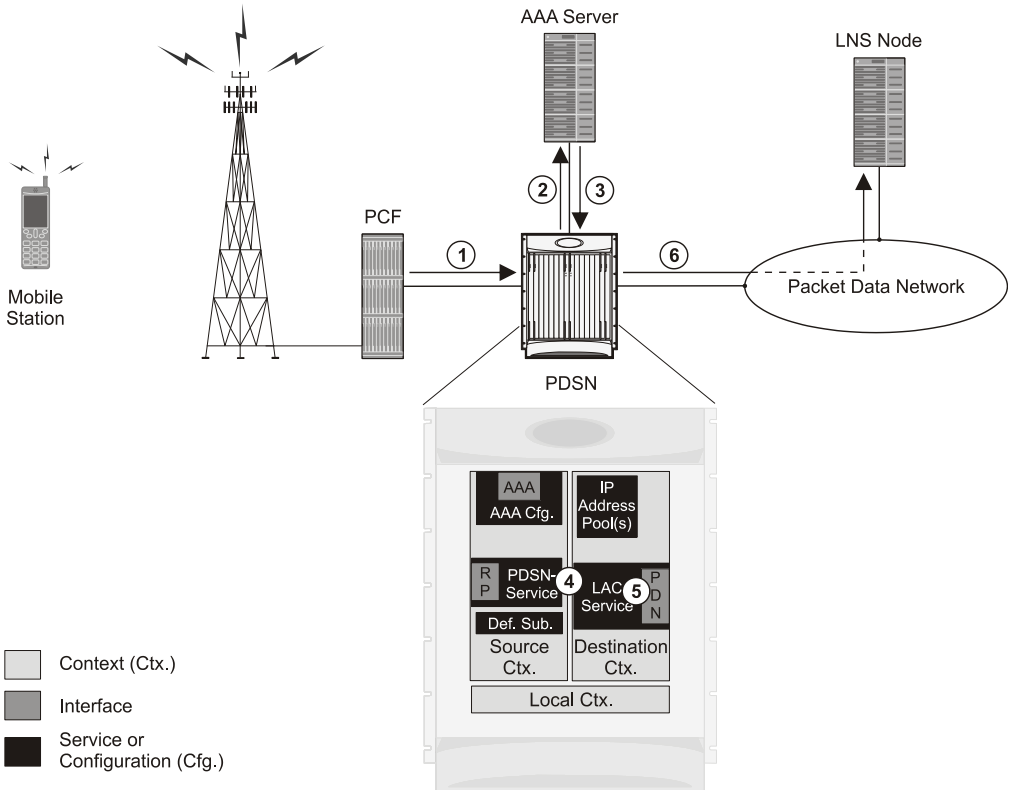
Attribute-based Tunneling

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 27. 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 as described in *Verifying and Saving Your Configuration*.

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

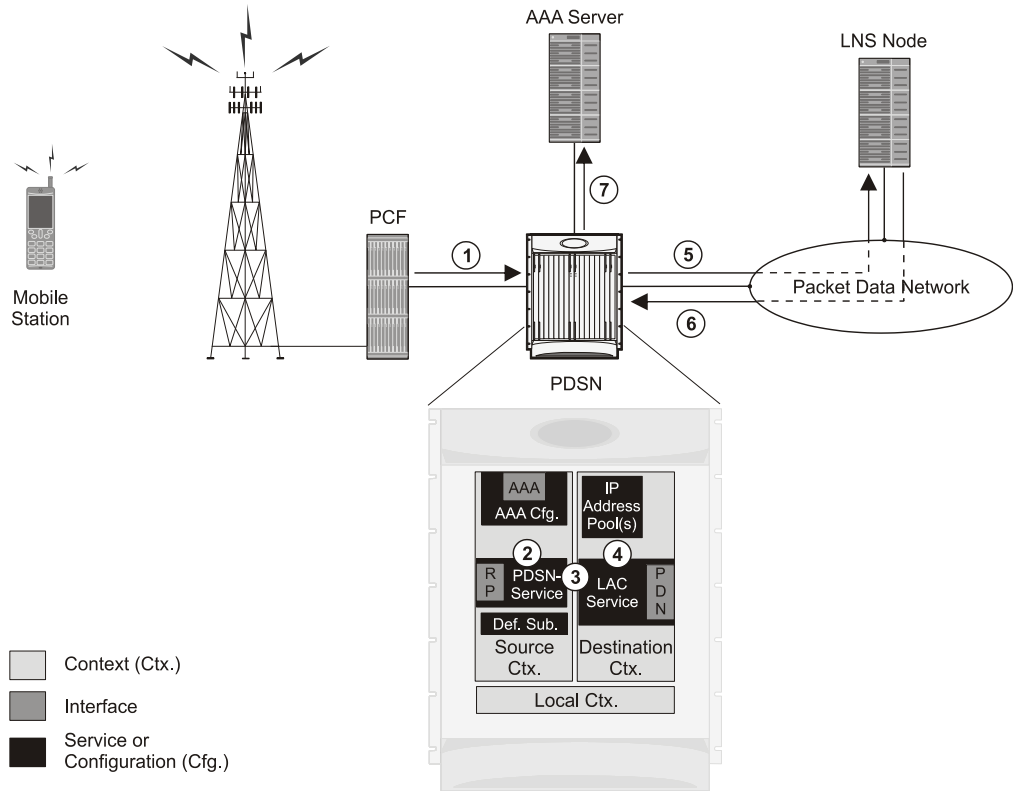
PDSN Service-based compulsory tunneling enables wireless operators to send all PPP traffic to remote LNS peers over an L2TP tunnel for authentication. This means that no PPP authentication is performed by the system.

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 28. 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.

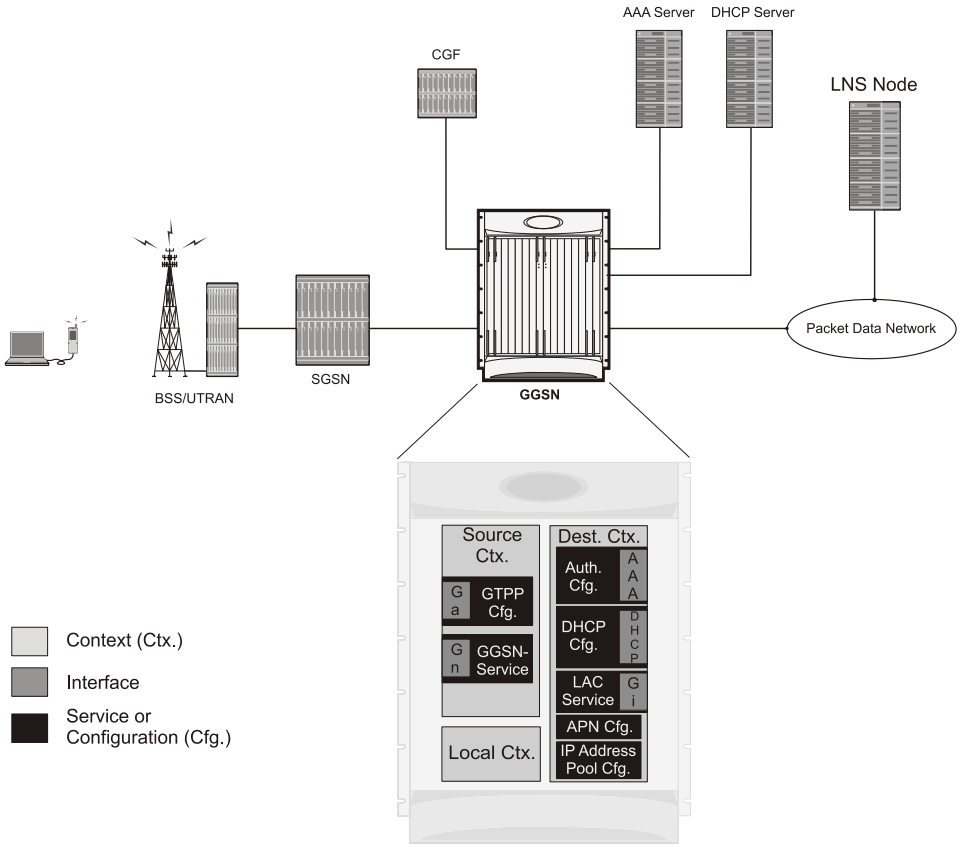
Step 2 Configure the PDSN service(s) according to the instructions located in the *Modifying PDSN Services for L2TP Support* section of this chapter.

Step 3 Save your configuration as described in *Verifying and Saving Your Configuration*.

Supported LAC Service Configurations for the GGSN

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 29. 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-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-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:

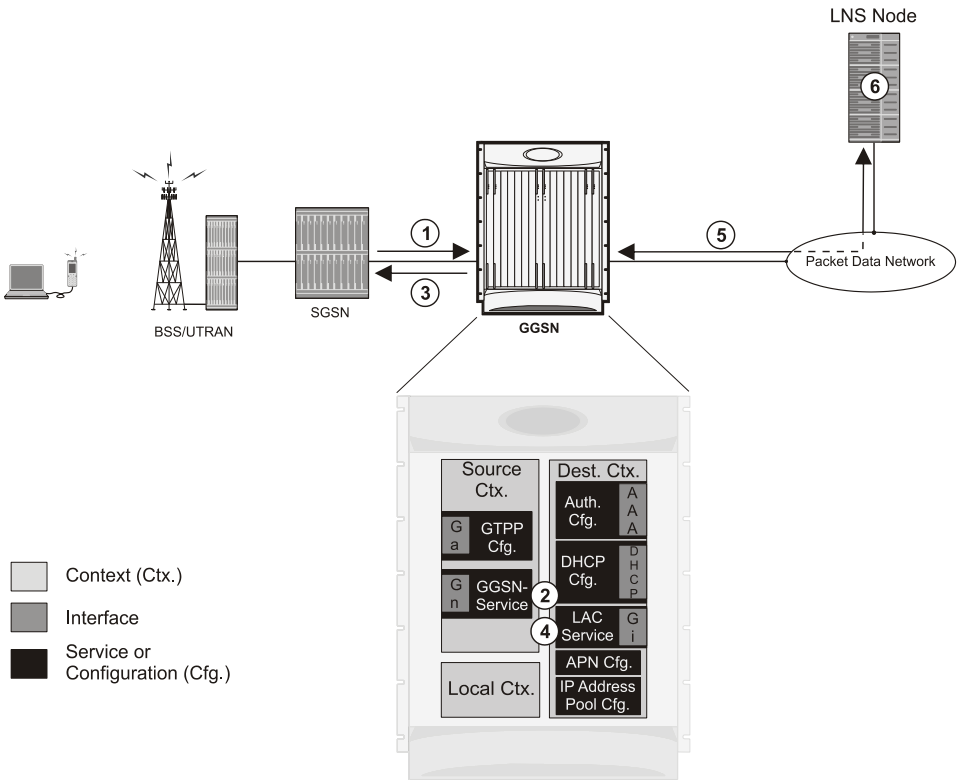
- **Transparent IP:** The APN template's L2TP parameter settings will be applied to the session.
- **Non-transparent IP:** Since authentication is required, L2TP parameter attributes in the subscriber profile (if configured) will take precedence over the settings in the APN template.
- **PPP:** The APN 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 30. 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.

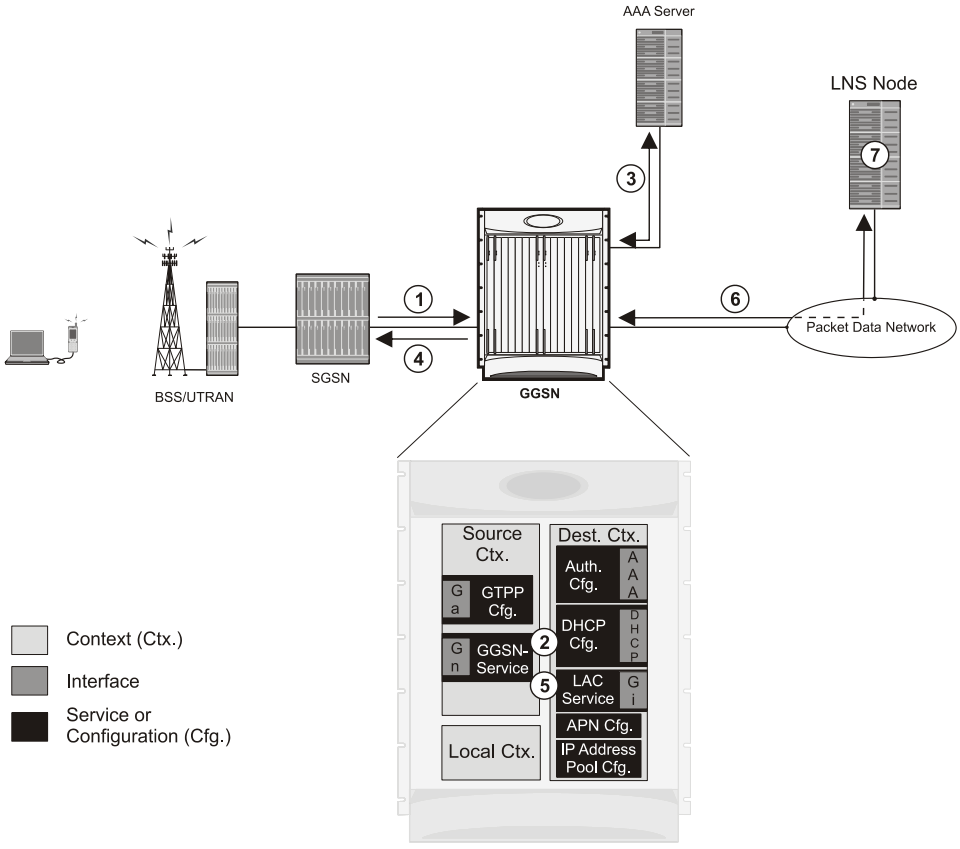
The APN configuration indicates such things as the IP address of the LNS, the system destination context in which a LAC service is configured, and the outbound username and password that will be used by the LNS to authenticate incoming sessions. If no outbound information is configured, the subscriber's 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 31. 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.

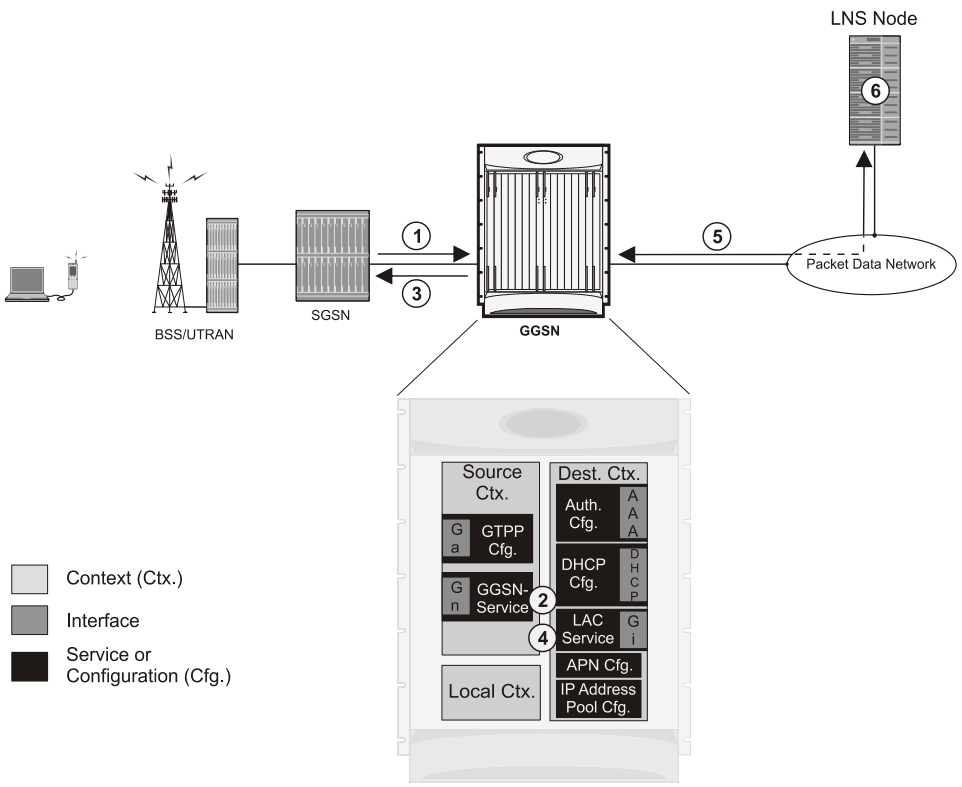
   The message may include attributes indicating that session data is to be tunneled using L2TP, and the name and location of the LAC service to use. An attribute could also be provided indicating the LNS peer to connect to.

   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 32. 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 to Support L2TP

This section provides a list of the steps required to configure the GGSN 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.

Step 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.

Step 2 Configure one or more LAC services according to the information and instructions located in the *Configuring LAC Services* section of this chapter.

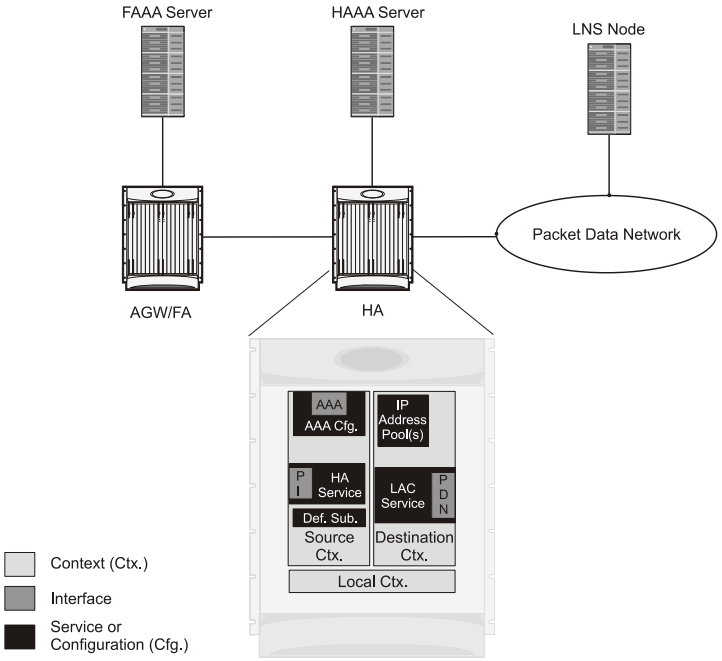
Step 3 Save your configuration as described in *Verifying and Saving Your Configuration* chapter.

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.

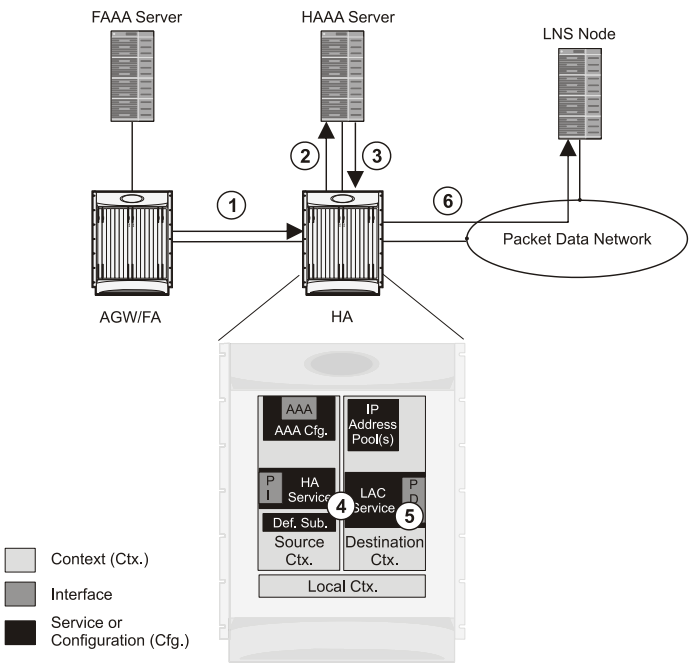
Figure 33. LAC Service Configuration for MIP



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 34. 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 as described in *Verifying and Saving Your Configuration* chapter.

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 25. 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

RADIUS Attribute	Local Subscriber Attribute	Description	Variable
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
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.	• **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 LAC Service Configuration Mode Commands chapter in 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 as described in *Verifying and Saving Your Configuration* chapter.

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 LAC Service Configuration Mode Commands chapter in 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 as described in the *Verifying and Saving Your Configuration* chapter.

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
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 LAC Service Configuration Mode Commands chapter in 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 as described in the *Verifying and Saving Your Configuration* chapter.

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 LAC Service Configuration Mode Commands chapter in 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 as described in the *Verifying and Saving Your Configuration* chapter.

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 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.

Chapter 22
L2TP Network Server

This chapter describes the support for Layer 2 Tunneling Protocol (L2TP) Network Server (LNS) functionality on ST16 and Cisco® ASR 5000 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: This product requires that you buy a license and feature use key. Not all features and functions may be functioning on all platforms.

When enabled though 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.

L2TP LNS Session and Tunnel Capacities

The system is capable of supporting L2TP tunnels for all subscriber sessions or on a session-by-session basis.

Each L2TP tunnel can facilitate one or more subscriber sessions. The number of supported L2TP sessions and tunnels corresponds to the number of active Packet Accelerator Cards (PACs) or Packet Services Cards (PSCs) available to the system.

The following table lists tunnel and session capacities for the ST16,

Table 26. L2TP LNS Tunnel and Session Capacity per ST16

	Capacity/PAC	Capacity/Full System1	Minimum # of LNS services Required2
Tunnels	32,000	32,000	1
Sessions	40,000	500,000	1
Transactions/Second	5003	6,0004	N/A
1. A full system consists of 13 active PACs.			
2. Multiple LAC services can be configured to reduce the amount of traffic over a single interface.			
3. Transactions include PPP and LNS.			
4. Transactions include PPP and LNS session setup assuming that a tunnel already exists.			

Table 27. L2TP LNS Tunnel and Session Capacity per ASR 5000

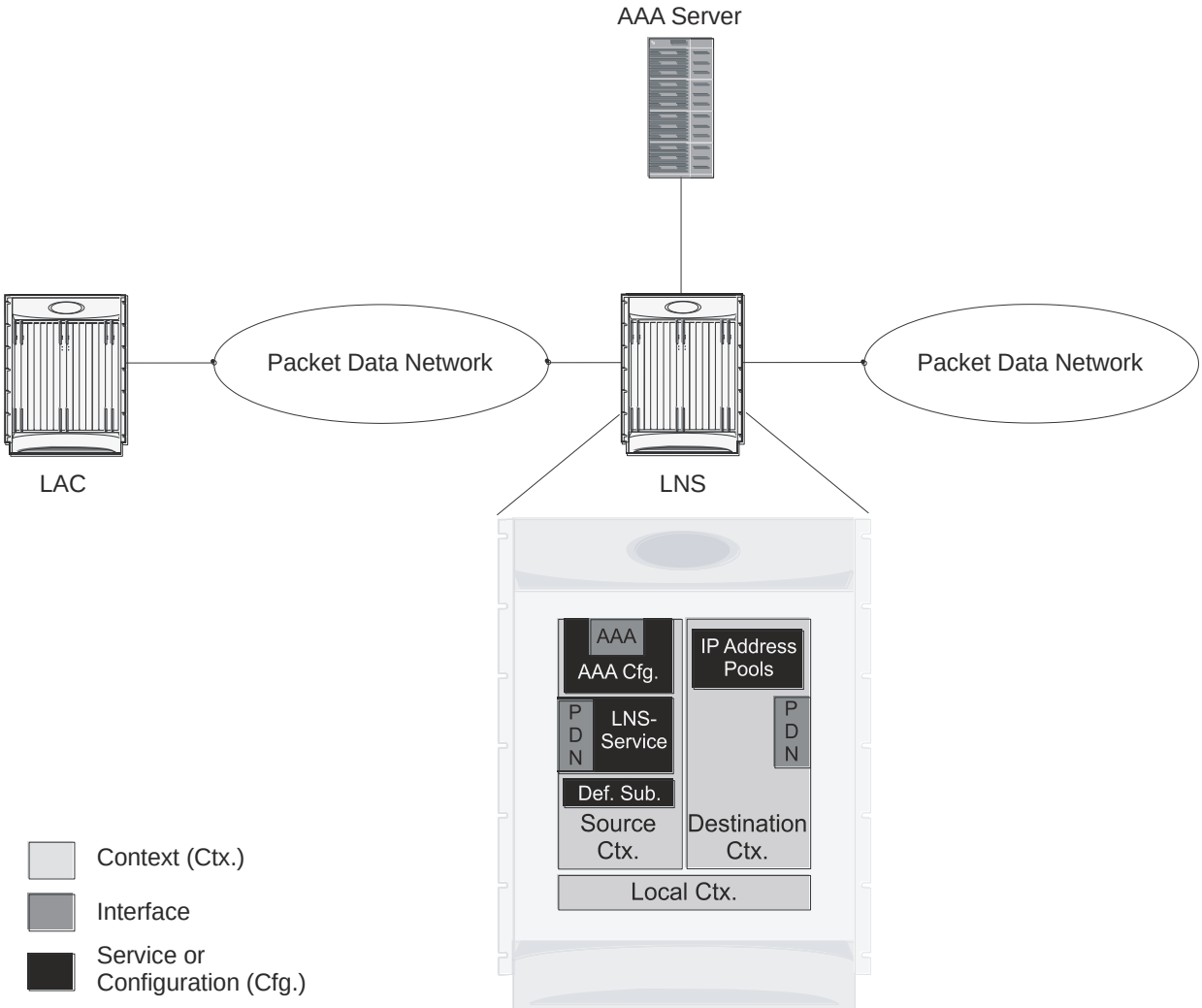
	Capacity/PSC	Capacity/Full System1	Minimum # of LNS services Required2
Tunnels	32,000	32,000	1
Sessions	125,000	1,500,000	1
Transactions/Second	8003	10,0004	N/A
1. A full system consists of 13 active PSCs.			
2. Multiple LAC services can be configured to reduce the amount of traffic over a single interface.			
3. Transactions include PPP and LNS.			
4. Transactions include PPP and LNS session setup assuming that a tunnel already exists.			

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 35. 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 28. 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.

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.
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.

Required Information	Description
	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 29. Required Information for Destination Context Configuration

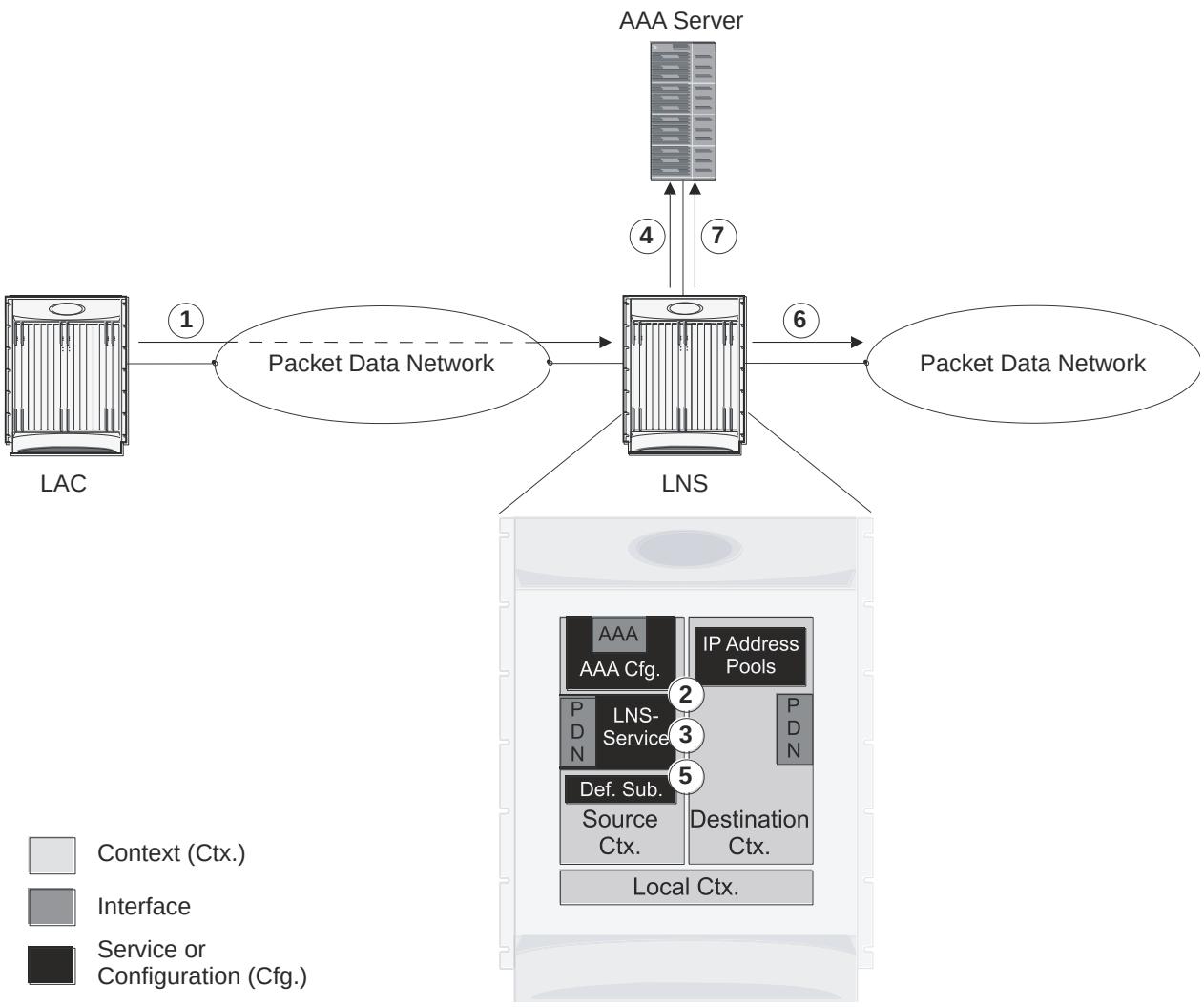
Required Information	Description
Destination context name	This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the destination context will be recognized by the system. **NOTE:** For this configuration, the destination context name should **not** match the domain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

Required Information	Description
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.
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 36. 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 as described in the *Verifying and Saving Your Configuration* chapter.

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 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 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 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
```

Chapter 23
MIP NAT Traversal

This chapter describes support for MIP NAT traversal and how to enable it on the system. 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: MIP NAT traversal functionality is enabled through a feature license key.

Overview

If a Mobile Node (MN) supports Mobile IP Network Address Translation (MIP NAT) traversal, it can indicate to the Home Agent (HA) that it is able to use MIP UDP tunneling when the HA sees that the Registration Request (RRQ) has traversed a NAT device.

The HA determines that the RRQ has passed through a NAT device by comparing the care-of-address in the RRQ with the source IP address of the RRQ. If they are different, and the D bit is set in the RRQ, then it indicates that the RRQ has passed through a NAT device.

If NAT is not detected but the Force (F) bit is set in the RRQ along with a UDP Tunnel Request, the HA rejects the call with the code 129 in the Registration Response (RRP). You can configure a parameter to force the HA to accept these types of requests for UDP tunneling in the absence of NAT.

When the D bit is not set and a mismatch occurs between the source address and the care-of-address, this could be a case when a mobile is registering through an FA using different addresses for signaling and data traffic. This registration behavior is allowed by the HA service.

The MN and HA negotiate UDP tunneling support during Mobile IP call setup. The MN includes a UDP Tunnel Request Extension in the RRQ sent to the HA. This extension optionally specifies the encapsulation type to be used as well (IP, GRE, or Minimal IP). The system only supports IP encapsulation at this time. Note also that the D bit must be set when UDP Tunneling is requested.

If the HA supports the requested form of tunneling, and the registration is successful, it responds with a UDP Tunnel Reply Extension in the RRP and specifies the keepalive interval the MN should use.

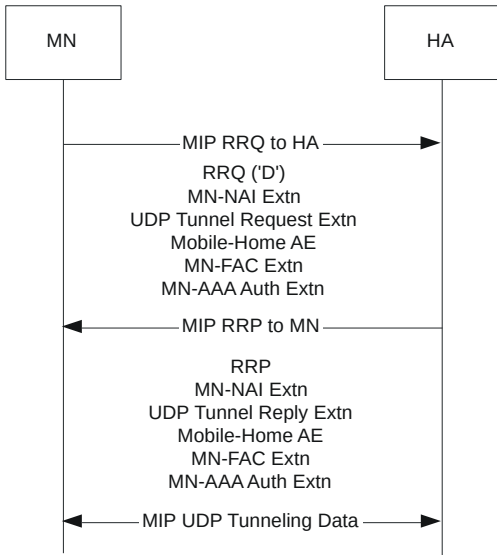
If HA does not accept the requested type of UDP tunneling, it ignores the UDP Tunnel Request extension and does not include the UDP Tunnel Reply extension in the Registration Reply. Error code 142 is used in the RRP to indicate to the MN that the requested UDP tunnel encapsulation is unavailable.

The UDP Tunnel Request extension is included in all initial, renewal, and handoff RRQ and RRP messages. The UDP Tunnel Request extension is not included in a Deregistration RRQ from the MN and the HA ignores them if they are included in Dereg RRQs received.

When MIP NAT Traversal is used, normally reverse tunneling is also used. However, this is not required by the HA.

An example of successful MIP UDP Tunneling negotiation is shown below.

Figure 37. MIP UDP Tunneling negotiation between MN and HA



The following table lists the various cases possible in UDP Tunneling negotiation during Mobile IP call establishment.

Table 30. MIP UDP Tunneling Negotiation Cases

Case	RRQ received at the HA	Action at HA
1	NAT detected, UDP Tunnel Request sent, NAT Traversal enabled	Accept call with IP-UDP tunneling, UDP Tunnel Reply included in RRP
2	NAT detected, UDP Tunnel Request sent, NAT Traversal disabled at the HA	Reject with code 129
3	NAT not detected, UDP Tunnel Request sent, F bit not set	Accept call with IP-IP tunneling, UDP Tunnel Reply not included
4	NAT not detected, UDP Tunnel Request sent, F bit set, forced UDP tunnel NOT allowed	Reject with code 129
5	NAT not detected, UDP Tunnel Request sent, F bit set, forced UDP tunnel allowed	Accept call with IP-UDP tunneling, UDP Tunnel Reply included in RRP
6	UDP Tunnel Request sent, D bit not set	Reject with code 134
7	NAT detected, UDP Tunnel Request not sent	Reject with code 129

Enabling MIP NAT Traversal

MIP NAT traversal must be enabled for the desired HA service on the system.

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.

To enable MIP NAT traversal, set parameters by applying the following example configuration:

```
configure
   context <context_name>
      ha-service <name>
         nat-traversal
         end
```

Notes:

- Optionally, you can configure the HA to accept requests when NAT is not detected but the Force (F) bit is set in the RRQ with the UDP Tunnel Request by entering the following command: **nat-traversal force-accept**

Save your configuration as described in the Saving Your Configuration chapter.

Viewing MIP NAT Traversal Statistics

Use the following commands in **exec mode** to list statistics that include information about MIP NAT Traversal:

- **monitor {protocol | subscriber}**- Use the MIP Tunnel option to trace IP-UDP tunneled datagrams.
- **show ha-service** *service_name* - Shows the MIP NAT Traversal configuration for the specified HA service.
- **show mipha statistics** - Lists IP-UDP tunnel statistics for Home Agent calls specified.
- **show mipha full** - Displays NAT, UPD, and encapsulation information for Home Agent calls specified.
- **show subscribers full** - Displays NAT, UPD, and encapsulation information for the subscribers specified.
- **{show | clear} subscribers ccoa-only** - Show or clear sessions for subscribers that registered a MIP colocated COA directly with the HA.
- **{show | clear} subscribers mip-udp-tunnel-only** - Show or clear sessions for subscribers that negotiated MIP UDP tunneling with the HA.

Refer to *Exec Mode Commands* chapter in *Command Line Interface Reference* for details on using these commands.

Chapter 24
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 the respective product 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 as described in *Saving Your Configuration*.

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 as described in *Saving Your Configuration*.

Chapter 25
MSID and PCF Zone Based Call Redirection

This chapter describes MSID and PCF zone based call redirection. 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 features described in this chapter are only available if you have purchased and installed a feature license for PDSN RAN Optimization, Bundle 1.

Overview

MSID and PCF zone based call redirection allows calls from a specific MSID or a specific PCF zone to be redirected to an alternate PDSN. These features are only applicable to new calls; handoffs are accepted by the PDSN in all cases. If the PDSN is in the process of starting up, the overload policy is applied before the zone/IMSI based call redirection. Once the PDSN is ready to accept new calls, the zone/IMSI based call redirection policy is applied before the overload policy. Upon receiving an RRQ from a PCF, the PDSN sends an RRP with the code 136 - Unknown PDSN Address.

MSID Based Call Redirection

The PDSN contains a table of MSIDs and the corresponding set of PDSNs to which the call should be redirected. It allows the configuration of up to 16 wildcard MSIDs per PDSN service. The wildcard must be a single-digit match represented by the "$" character. For example, the MSID 847$$$$$$$$12$$ would match all MSIDs starting with 847 followed by any eight digits, followed by 12 and any two additional digits.

When a new call arrives, the PDSN attempts to match the MSID with the configured list of wildcard MSIDs. If a match is found, the call is redirected to one of the PDSNs by IP address using a weighted round-robin algorithm. If more than one match is found, the algorithm selects the match with the longest matching prefix.

PCF Zone Based Call Redirection

Groups of PDSNs maybe configured with a specific numbered zone. When a new call arrives, the PDSN checks the PCF for a specified zone number. If the PCF matches the specified zone, the call is redirected to a PDSN within the zone using a weighted round-robin algorithm. If the PCF from which the call arrived does not belong to a zone, or if no PDSNs are configured for the specified zone, the call is not redirected. Similarly, if a zone is configured for a PCF address and the current PDSN-service address is a member of that zone, the call is not redirected.

Caution: These two features introduce additional lookups in the call setup path and could impact the call setup rate.

Important: If both MSID and PCF zone based call redirection are configured, MSID based call redirection will have a higher precedence.

Configuring MSID and PCF Zone Based Call Redirection

This section describes the process of setting up MSID and PCF zone based call redirection from the command line interface.

Caution: Incorrect configuration of the MSID and PCF Zone based Call Redirection features could result in sessions failing to be established. For example, if PDSN1 is configured to redirect sessions to PDSN2 while PDSN2 is configured to redirect sessions to PDSN1, a loop is created in which all sessions would fail to be connected. In addition, sessions will not be established if the PDSN to which the sessions are being redirected is unavailable.

Configuring MSID Based Call Redirection

To configure MSID based call redirection, you must create a new policy that defines a wildcard match list, a list of PDSNs to redirect to, and their respective weights.

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.

Configure MSID based call redirection by applying the following example configuration:

```
configure
   context <context_name>
      pdsn-service <pdsn_service_name>
         policy msid-match <msid_with_wildcards> redirect <address>
weight <weight_num> <address2> weight <weight_num>...<address16> weight
<weight_num>
         end
```

Notes:

- You may repeat the **policy msid-match** command as needed, to a maximum of 16 wildcard MSIDs per PDSN service.

Save your configuration as described in the Saving Your Configuration chapter.

Configuring PCF Zone Based Call Redirection

To configure PCF zone based call redirection, you must create a new policy that defines a zone match list, a list of PDSNs to redirect to, and their respective weights.

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.

To configure PCF zone based call redirection:

Step 1 Set parameters by applying the following example configuration:

```
configure

   context <context_name>

      pdsn-service <pdsn_service_name>

         spi remote-address <pcf_ip_address> spi-number <number> secret
<secret> zone <zone_id> zone <zone_id2>...zone <zone_id32>

         policy pcf-zone-match <zone_number> redirect <address> weight
<weight_num> <address2> weight <weight_num>...<address16> weight
<weight_num>

         end
```

Notes:

- You may repeat the **spi remote-address** <pcf_ip_address> **spi-number** *<number>* **secret** *<secret>* [**zone** *<zone_id>*] [**zone** *<zone_id2>...* **zone** *<zone_id32>*] command as necessary. You can configure up to 32 PCF zones per PDSN service.
- You may repeat the **policy pcf-zone-match** *<zone_number>* { **redirect** *<address>* } [**weight** *<weight_num>*] [*<address2>* [weight *<weight_num>*]...*<address16>* [**weight** *<weight_num>*]] command as necessary, up to a maximum of 32 defined PCF zones and 16 defined PDSNs per PDSN service.

Step 2 Save your configuration as described in the *Saving Your Configuration* chapter.

Chapter 26
Multimedia Broadcast and Multicast Service

This chapter provides information on Multimedia Broadcast and Multicast Service (MBMS) functionality on GGSN. 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 features described in this chapter are only available if you have purchased and installed MBMS feature support license on your chassis.

This chapter discusses following topics for MBMS support:

Introduction

MBMS is an IP datacast type of service in GSM and UMTS cellular network. It eliminates unnecessary replication of data on UMTS wireless networks by transmitting a single stream of data to multiple users. By delivering a single, unidirectional data stream to many subscribers, MBMS makes more efficient use of wireless network resources than traditional point to point connections.

MBMS is a solution for transferring light video and audio clips with a suitable method for mass communications.

MBMS functionality on the system is provided by an existing GGSN service and is enabled by a valid services license.

The main features supported by the Multimedia Broadcast & Multicast Services are:

- Individual user network control functions and provide forward MBMS user data to SGSN

> ***Important:*** The ASR 5000 platform supports 225 downlink SGSNs per MBMS Bearer Service through NPU assisted data flow processing. NPU assisted data processing is available on ASR 5000 platforms with release 8.1 or later only.

- Support for intra-GGSN and inter-GGSN mobility procedures
- Generate charging data per multicast service for each user for both prepaid and post paid subscribers.
- Multicast proxy-host functionality
- Support for MBMS-specific Gmb messages
- Authentication of MBMS flow-ids using a MBMS controller
- Establishment and teardown of MBMS bearer paths using the multicast framework
- Support for framing HDLC-like and segment based framing
- Accounting for the MBMS flows to charge the originator of the content

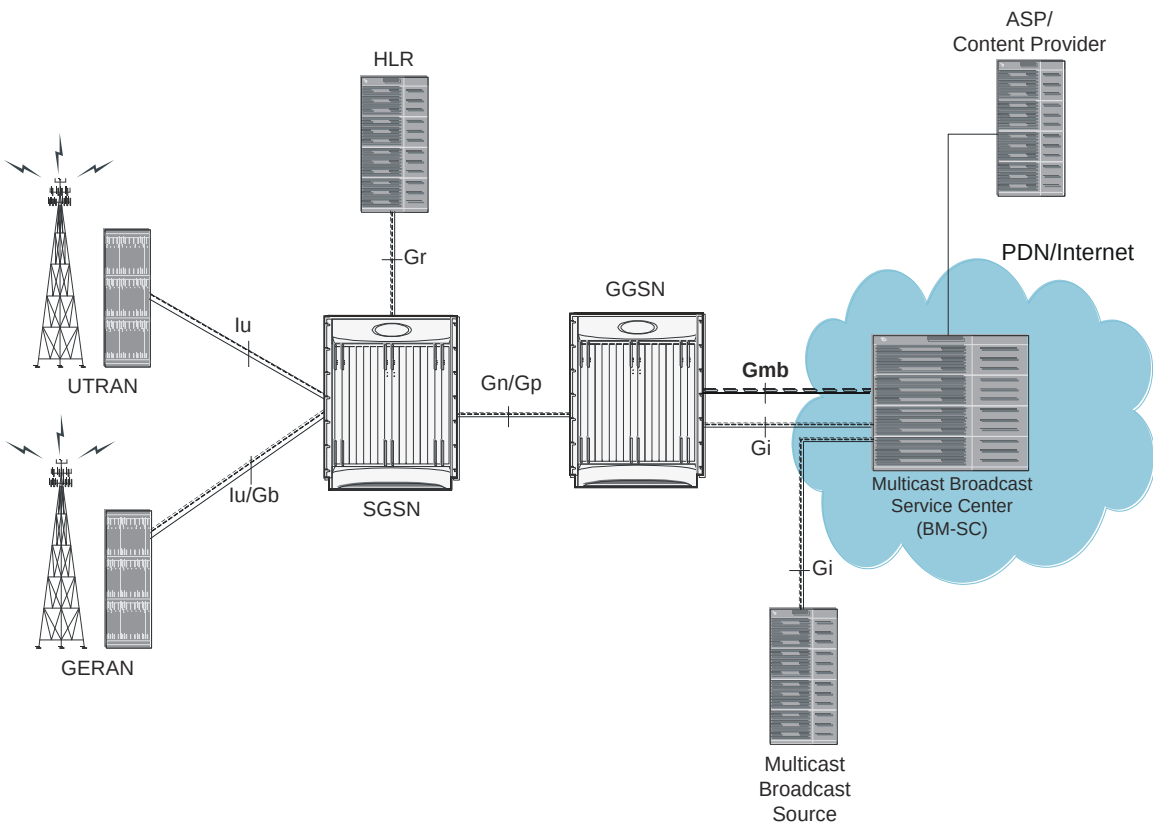
This service provides two mode of operations:

- MBMS Broadcast Mode
- MBMS Multicast Mode

A broadcast mode is a unidirectional point-to-multipoint service in which data is transmitted from a single source to multiple terminals (UE/MS) in the associated broadcast service area/cell area. The transmitted data can be text to light multimedia services (Audio, Video etc). On the other hand multicast mode is a unidirectional point-to-multipoint service in which data is transmitted from a single source to a pre-defined multicast group of users that are subscribed to the specific multicast service and have joined the multicast group in the associated multicast service area.

The following figure shows the reference architecture of MBMS service in UMTS network.

Figure 38. MBMS Reference Architecture in UMTS network



The GGSN provides the following functionality to perform MBMS services:

- serves as an entry point for IP multicast traffic as MBMS data. It provides establishment of bearer plan and teardown of the established bearer plan upon notification from the BM-SC.
- provides functionality to receive MBMS specific IP multicast traffic and to route this data to the proper GTP tunnels set-up as part of the MBMS bearer service.
- provides features, that are not exclusive to MBMS, for the MBMS bearer service, like charging data collection, flow-based charging, optional message screening etc.

MBMS is able to use NPU assisted MBMS data flow processing on ASR 5000 platforms so that system can relieve the Session Manager to provide better performance and processing. Currently with NPU assisted data processing, ASR 5000 can support 225 SGSNs per MBMS Bearer Service for downlink of MBMS data.

Supported Standards

Support for the following standards and requests for comments (Rafts) have been added with the MBMS functionality:

- 3GPP TS 22.146: Multimedia Broadcast/Multicast Service; Stage 1 (Release 6)
- 3GPP TS 22.246: MBMS user services; Stage 1 (Release 6)
- 3GPP TS 23.246: MBMS; Architecture and functional description (Release 6)
- 3GPP TS 26.346: MBMS; Protocols and codecs (Release 6)
- 3GPP TS 33.246: Security of Multimedia Broadcast/Multicast Service
- 3GPP TS 32.251: Telecommunication management; Charging management; Packet Switched (PS) domain charging
- 3GPP TS 32.273: Telecommunication management; Charging management; Multimedia Broadcast and Multicast Service (MBMS) charging
- RFC 3588, Diameter Base Protocol

Supported Networks and Platforms

This feature supports all ASR 5000 platforms running StarOS Release 8.0 or later with GGSN service for the core network services.

Licenses

This feature support requires any one of the following feature licenses installed on the system:

- 600-00-7581
- 600-00-7621

Services and Application in MBMS

MBMS service can be used as an enabler for various data streaming services. Compared to traditional broadcast services like cell broadcast, MBMS provides multimedia capabilities with relatively high data rates and considerably greater multimedia capabilities.

Some of the applications for MBMS are:

- News clips
- Audio streams
- Combined audio and picture/video clips
- Video distribution services, either via streaming, carousel, or download methods
- Localized services like tourist information, weather alerts etc.
- Content distribution
- Game delivery

The charging of the MBMS bearer service can be done based on events, content, or flows.

MBMS provides the authentication, key distribution, and data protection for the multicast service users.

MBMS References and Entities

Following are the major components and entities required for MBMS service.

Gmb Reference

The Gmb reference point handles the broadcast multicast service center (BM-SC) related signaling, which includes the user specific and bearer service messages.

MBMS bearer service specific Gmb signaling includes:

- MBMS bearer context establishment by GGSN and registering of GGSN at BM-SC.
- Release of MBMS bearer context at GGSN and de-registration of GGSN from the BM-SC.
- Session start/stop indication from BM-SC to GGSN including session attributes like QoS or MBMS service area.

User specific Gmb signaling includes:

- BM-SC authorization of user specific MBMS multicast service activation at the GGSN.
- Reporting of successful user specific MBMS multicast service activation by GGSN to BM-SC to synchronize the BM-SC UE MBMS context and charging with the MBMS UE contexts in GGSN.
- Reporting of release or deactivation of user specific MBMS multicast service activation by GGSN to BM-SC to synchronize the BM-SC UE MBMS context and charging with the MBMS UE contexts in GGSN.
- BM-SC initiated deactivation of user specific MBMS bearer service when the MBMS user service is terminated.

MBMS UE Context

A MBMS UE context is defined per UE. Session Manager assign a separate context structure for a MBMS UE Context.

Session Manager maintains the following information as part of MBMS UE Context:

- IP multicast address: IP multicast address identifying an MBMS bearer that the UE has joined.
- APN: Access Point Name on which this IP multicast address is defined.
- SGSN address: The IP address of SGSN
- IMSI: IMSI identifying the user.
- TEID for Control Plane: The Tunnel Endpoint Identifier for the control plane between SGSN and GGSN.
- MBMS NSAPI: Network layer Service Access Point Identifier which identifies an MBMS UE Context.

Important: For capacity and resource purpose one MBMS UE context is equal to one PDP context.

MBMS Bearer Context

The MBMS bearer context is created in the SGSN and GGSN for each provisioned MBMS service. This is created when the first MS requests for this service or when a downstream node requests it. Once created, an MBMS context can be in two states:

- Active - is the state in which network resources are required for the transfer of MBMS data.
- Standby - is the state in which no network resources are required.

The MBMS Bearer Context contains all information describing a particular MBMS bearer service and is created in each node involved in the delivery of the MBMS data.

Broadcast Multicast Service Center (BM-SC)

The BM-SC includes functions for MBMS user service provisioning and delivery. It serves as an entry point for content provider MBMS transmissions, used to authorize and initiate MBMS Bearer Services within the PLMN. It can also be used to schedule and deliver MBMS transmissions.

The BM-SC consists of five sub-functions:

- Membership function
- Session and Transmission function
- Proxy and Transport function
- Service Announcement function
- Security function.

BM-SC is a functional entity and must exist for each MBMS User Service.

How MBMS Works

The Multimedia Broadcast Multicast System provides two types of service provisioning; broadcast and multicast modes. This section describes the procedure of these modes.

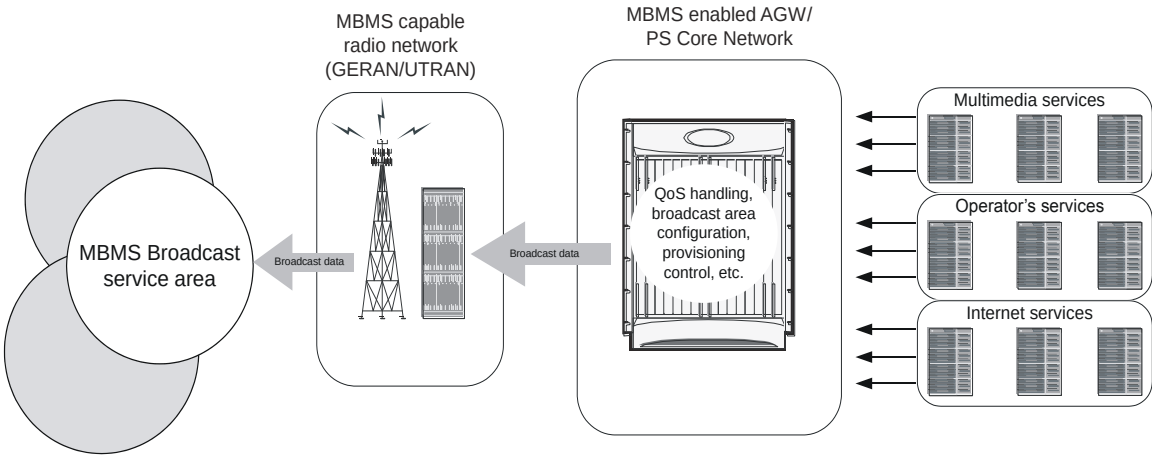
MBMS Broadcast Mode

The broadcast mode provides unidirectional point-to-multipoint type transmission of multimedia data from a single source to all users that found in a defined broadcast service area. This mode uses radio resources efficiently, since the data is transmitted over a common channel.

MBMS data transmission adapts to the suitable RAN capabilities, depending on the availability of radio resources too. If needed, the bit rate of MBMS data may be varied in order to optimized radio resources.

The following figure shows the basic outline of broadcast mode procedure of an MBMS service in order to broadcast MBMS data within the defined broadcast service area via a packet switched core network.

Figure 39. Basic Procedure of MBMS Broadcast Mode



The broadcast service may include one or more successive broadcast sessions. The user can control the enabling or disabling of the MBMS broadcast mode service.

MBMS Broadcast Mode Procedure

The MBMS performs following steps for broadcast mode user service:

Step 1 Service Announcement: Through the service announcement mechanisms, like SMS, WAP, users informed about the available MBMS services.

Step 2 Session Start: This is the phase where BM-SC has data to send and this triggers establishment of network resources for data transfer irrespective of whether a given user has activated the service or not.

Step 3 MBMS Notification: Notifies the MS of a impending MBMS data transfer.

Step 4 Data Transfer: It is the phase when MBMS data are transferred to the UEs.

Step 5 Session Stop: In this phase, the BM-SC determines that it has no more data to send for a time period and so the network resources can be released.

MBMS Multicast Mode

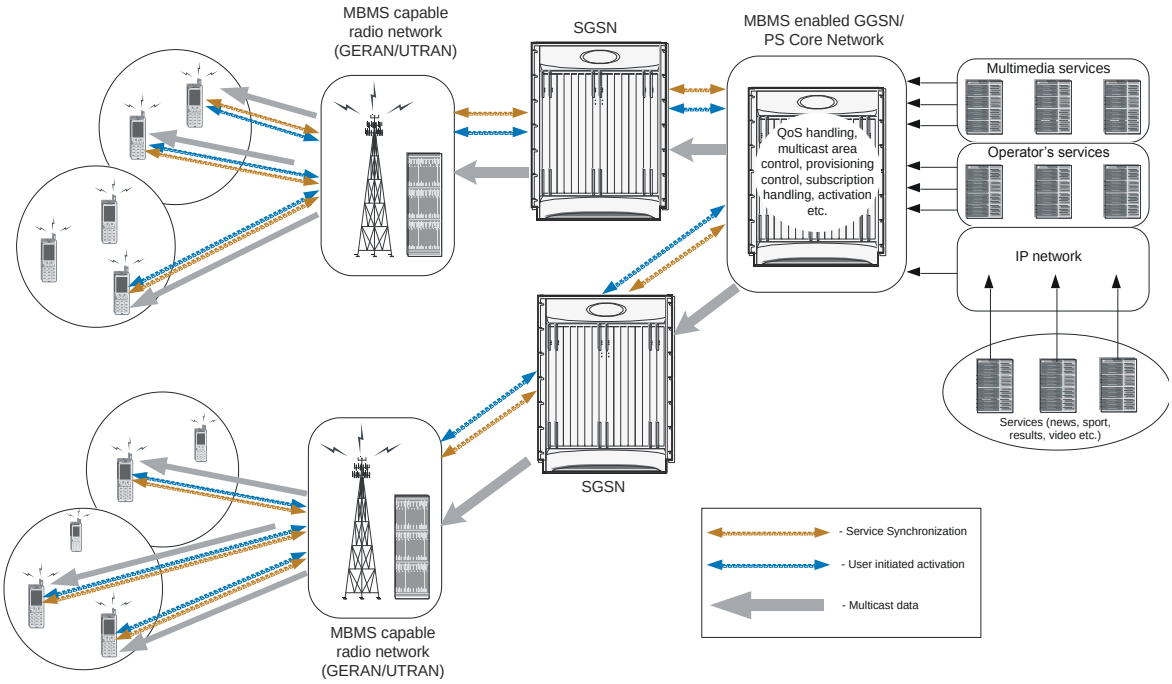
The multicast mode provides unidirectional point-to-multipoint type transmission of multimedia data from a single content source to a group of subscribers that subscribed to specific multicast service separately. The basic difference between broadcast and multicast modes is that the user does not need to subscribe in each broadcast service separately, whereas in multicast mode the services cab be order separately. The subscription and group joining for the multicast mode service can be done by the operator, user, or a separate service provider.

Like broadcast mode the multicast mode allows the unidirectional point-to-multipoint transmission of multimedia data within the multicast service area. The multicast mode uses radio resources in efficient way by using common radio channel as in broadcast mode. Data is transmitted over the multicast service area as defined by the network operator.

The multicast mode provides the flexibility for the network to selectively transmit to those cells within the multicast service area that contains members of a multicast group.

The following figure shows the basic outline of multicast mode procedure of an MBMS service in order to multicast MBMS data within the defined multicast service area via a packet switched core network.

Figure 40. Basic Procedure of MBMS Multicast Mode



A multicast service might consist of a single on-going session or may include several simultaneous multicast sessions over and extended period of time.

Some example of multicast mode service are:

- transmission of sports video clips to subscribers on charging basis
- transmission of news, movie, song, and audio clips to subscribed users on charging basis

MBMS Multicast Mode Procedure

The MBMS performs following steps for multicast mode user service:

Step 1 Subscription: Establishes the relationship between the user and the service provider, which allows the user to receive the related MBMS multicast service.

Step 2 Service Announcement: Through the service announcement mechanisms like, SMS, WAP, users shall be informed about the available MBMS services.

Step 3 Joining: This is the process by which a subscriber joins a multicast group, i.e. the user indicates to the network that he/she wants to receive Multicast mode data of a specific MBMS bearer service.

Step 4 Session Start: This is the phase where BM-SC is ready to send data and this triggers establishment of network resources for data transfer irrespective of whether a given user has activated the service or not.

Step 5 MBMS Notification: Notifies the MS of a impending MBMS data transfer.

Step 6 Data Transfer: It is the phase when MBMS data are transferred to the UEs.

Step 7 Session Stop: In this phase, the BM-SC determines that it has no more data to send for a time period and so the network resources can be released.

Step 8 Leaving: In this phase, the user leaves a MBMS group through an Internet Group Management Protocol (IGMP) Leave message.

MBMS Configuration

This section provides a high-level series of steps and the associated configuration examples for configuring the system with MBMS user service in GGSN services.

To configure the system to perform Multimedia Broadcast and Multicast service:

Step 1 Configure the BM-SC profile in a context by applying the example configurations presented in the *BMSC Profile Configuration* section.

Step 2 Configure the MBMS charging parameters in GTPP Server Group Configuration mode by applying the example configurations presented in the *MBMS GTPP Configuration* section.

Step 3 Configure the MBMS accounting, supported contexts, timeout parameters, and BMSC profile association with APN in APN configuration mode by applying the example configurations presented in the *MBMS APN Configuration* section.

Step 4 Enable the MBMS user service provisioning mode in GGSN and configure the number of MBMS UE and MBMS bearer context in GGSN configuration mode by applying the example configurations presented in the *MBMS Provisioning* section.

Step 5 Save the changes to system configuration by applying the example configuration found in *Verifying and Saving Your Configuration* chapter.

Step 6 Verify configuration of MBMS service related parameters by applying the commands provided in the *Managing Your Configuration* section of this chapter.

BMSC Profile Configuration

This section provides the configuration example to configure the BM-SC profile in a context:

```
configure

   context <vpn_context_name> [ -noconfirm ]

      bmsc-profile name <profile_name> [ -noconfirm ]

         default gmb diameter dictionary

         gmb diameter endpoint <endpoint_name>

         gmb diameter peer-select peer < peer_name> [ realm <realm_name> ] [
secondary-peer <sec_peer_name> [ realm <sec_realm_name> ]]

         default gmb user-data mode-preference

         end
```

MBMS GTPP Configuration

This section provides the configuration example to configure the GTPP server parameters in GTPP group configuration mode for MBMS charging:

```
configure
   context <vpn_context_name> [ -noconfirm ]
      gtpp group default
         gtpp mbms buckets <cc_bucket>
         gtpp mbms interval <duration_sec>
         gtpp mbms tariff time1 <mins> <hours> [ time2 <mins> <hours> ]
         gtpp mbms volume <download_bytes>
         end
```

MBMS APN Configuration

This section provides the configuration example to enable the BM-SC profile for an APN and to configure the MBMS accounting, supported contexts, and timeout parameters in APN configuration mode:

```
configure
   context <vpn_context_name>
      apn <apn_name> [ -noconfirm ]
         mbms bmsc-profile name <profile_name>
         default max-contexts
         accounting mode gtpp
         default mbms bearer timeout { absolute | idle }
         default mbms ue timeout absolute
         end
```

MBMS Provisioning

This section provides the configuration example for provisioning of MBMS service mode for a GGSN service and associating the MBMS policy for multcast broadcast within the GGSN service in GGSN service configuration mode:

```
configure
   context <vpn_context_name>
      ggsn-service <ggsn_service_name>
         mbms policy multicast broadcast
         end
```

Save the Configuration

To save changes made to the system configuration for this service, refer *Verifying and Saving Your Configuration* chapter.

Managing Your Configuration

This section explains how to display and review the configurations after saving them in a .cfg file as described in *Saving Your Configuration* chapter of this guide and also to retrieve errors and warnings within an active configuration for a service.

Important: All commands listed here are under Exec mode. Not all commands are available on all platforms.

Output descriptions for most of the commands are located in *Command Line Interface Reference*.

To do this:	Enter this command:
View Administrative Information	
Display Current Administrative User Access	
View a list of all administrative users currently logged on to the system	**show administrators**
View the context in which the administrative user is working, the IP address from which the administrative user is accessing the CLI, and a system generated ID number	**show administrators session id**
View information pertaining to local-user administrative accounts configured for the system	**show local-user verbose**
View statistics for local-user administrative accounts	**show local-user statistics verbose**
View information pertaining to your CLI session	**show cli**
Determining the System's Uptime	
View the system's uptime (time since last reboot)	**show system uptime**
View the Status of Configured NTP Servers	
View the status of the configured NTP servers	**show ntp status**
View the Statistics of Broadcast Multicast service	
View the full information of all broadcast-multicast service session	**show multicast-sessions full all**
View the status of all broadcast multicast-service session	**show session in-progress**
View all session for broadcast-multicast service	**show multicast-sessions all**
View Subscribers Currently Accessing the System	
View a listing of subscribers currently accessing the system	**show subscribers all**
View information for a specific subscriber	show subscribers full username **<***user_name***>**
View the MBMS Related Information	
Display Configured MBMS service	
View the configuration of a context	**show configuration context <***vpn_ctxt_name***>**

To do this:	Enter this command:
View configuration errors for GGSN service	**show configuration errors section ggsn-service [verbose] [| {grep <*grep_options*> | more }]**
Display BM-SC server Information	**show bmsc servers**

Gathering MBMS Statistics

The following table lists the commands that can be used to gather the statistics for MBMS.

Important: All commands listed here are under Exec mode. For more information on these commands, refer *Executive Mode Commands* chapter in *Command Line Interface Reference*.

Table 31. Gathering Statistics

Statistics Wanted	Action to Perform	Information to Look For
Gmb interface statistics for APN and BM-SC profile	At the Exec Mode prompt, enter the following command: **show gmb statistics** [**apn** <*apn_name*> | **bmsc-profile** <*bmsc_profile_name*>] [**verbose**]	The output of this command displays the statistics about the Gmb interface session for MBMS on an APN.
Detailed MBMS bearer service statistics	At the Exec Mode prompt, enter the following command: **show mbms bearer-service** [**mcast-address** <*mcast_address*>] [**apn** <*apn_name*>] [**bmsc-profile** <*bmsc_profile_name*>] **[service-type { multicast | broadcast }] [summary | full] [all]**	The output of this command displays the MBMS bearer service statistics.
Detailed statistics of MBMS multicast sessions	At the Exec Mode prompt, enter the following command: **show multicast-sessions**	The output of this command displays the detailed statistics of MBMS multicast session running on system.

Chapter 27
MultiProtocol Label Switching (MPLS) Support

This chapter describes the system's support for Multi Protocol Label Switching (MPLS) and explains how it is configured. The product administration guides provide examples and procedures for configuration of basic services on specific systems. 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.

When enabled through a feature license key, the system supports MPLS to provide a VPN from the GGSN/PDSN to the corporate's network. In this VPN, an APN that terminates on one GGSN/PDSN can change and terminate on another.

MPLS is used to negotiate the routes and segregate the traffic. The GGSN/PDSN learns the default route from the connected Provider Edge (PE), while the PE populates its routing table with the routes provided by the GGSN/PDSN.

This chapter includes following sections:

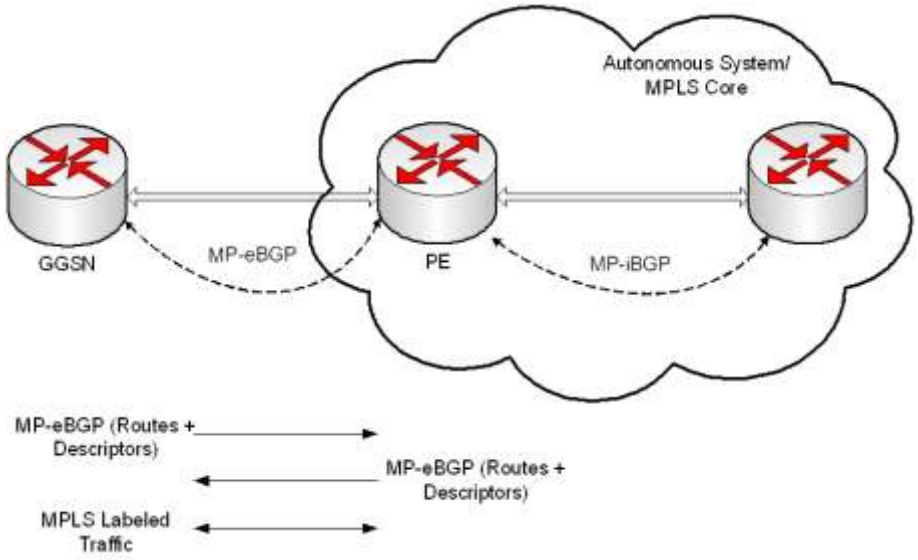
Overview

As seen in the following scenario, the chassis can be deployed as a router while supporting MPLS in a network.

- Chassis as MPLS-Customer Edge (MPLS-CE) with Provider Edge (PE)

Chassis as MPLS-CE with PE

Figure 41. Chassis as MPLS-CE with PE



The system in this scenario uses static MPLS labels for ingress and egress traffic. For configuration information on static label, refer to the Configuring MPLS over BGP with Static Labels section.

The system is in a separate autonomous system (AS) from the Provider Edge (PE). It communicates with the PE and routes for all VPNs supported on the PE that are exchanged with the system. Routes belonging to different VPNs are logically separated, using separate virtual route forwarding tables (VRFs).

Routes for each VPN are advertised as VPN-IPv4 routes, where route distinguishers are prepended to regular IPv4 routes to allow them to be unique within the routing table. Route targets added to the BGP extended community attributes identify different VPN address spaces. The particular upstream BGP peer routing domain (VPN), from which a route is to be imported by the downstream peer into an appropriate VRF, is identified with an extended community in the advertised NLRI, which is in fact an MPLS label.

The Customer Edge (CE) also advertises routes to the PE using NLRIs that include route distinguishers to differentiate VPNs, and extended community MPLS labels to identify VRFs.

There is a single MPLS-capable link between the CE and the PE. MP-BGP communicates across this link as a TCP session over IP. Data packets are sent bidirectionally as MPLS encapsulated packets.

This solution does not use any MPLS routing protocols. The MPLS label corresponding to the immediate upstream neighbor is statically configured on the downstream router, and similarly in the reverse direction.

When forwarding subscriber packets in the upstream direction to the PE, the CE encapsulates packets with MPLS headers that identify the upstream VRF (the label sent as an extended community with the NLRI) and the immediate next hop. When the PE receives a packet it switches the label to an MPLS-labeled packet.

The CE does not run any MPLS routing protocol (LDP or RSVP-TE).

When receiving data packets in the downstream direction from the PE, the label is checked to identify the destination VRF. Then the packet is de-encapsulated into an IP packet and sent to the session subsystem for processing.

Important: MPLS ping/trace route debugging facilities are not supported.

Engineering Rules

- Up to 100 virtual routing tables per context. This allows up to 100 BGP-VPNs per context.
- Up to 5k “host routes” spread across multiple VRFs per BGP process. Limited to 6000 pool routes per chassis.
- Up to 1024 VRFs per chassis.

Benefits

MPLS provides networks with a more efficient way to manage applications and move information between locations. MPLS prioritizes network traffic, so administrators can specify which applications should move across the network ahead of others.

Supported RFCs

The following RFCs related to MPLS are partially supported:

- IETF RFC 4364 BGP/MPLS IP VPNs
- IETF RFC 3032 MPLS Label Stack Encoding

Configuring MPLS over BGP with Static Labels

This section describes the procedures required to configure the system as an MPLS-CE to interact with a PE with static MPLS label support.

The base configuration, as described in the *Routing* chapter of the *System Enhanced Feature Configuration Guide*, must be completed prior to attempt the configuration procedure described below.

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.

To configure the system for MPLS:

Step 1 Create a VRF on the router and assign a VRF-ID by applying the example configuration in the *Create VRF with Route-distinguisher and Route-target* section.

Step 2 Set the neighbors and address family to exchange routing information with a peer router by applying the example configuration in the Set Neighbors and Address Family section.

Step 3 Redistribute connected routes between routing domains by applying the example configuration in the Redistribute Connected Routes section.

Step 4 Establish BGP peering with peer router by applying the example configuration in the Establish BGP Peering with Peer Router section.

Step 5 Configure the address family and redistribute the connected routes into BGP by applying the example configuration in the Configure Address Family section. This takes any routes from another protocol and redistributes them to BGP neighbors using the BGP protocol.

Step 6 Configure IP Pools with MPLS labels for input and output by applying the example configuration in the Configure IP Pools with MPLS Labels section.

Step 7 *Optional*. Bind DHCP service to work with MPLS labels for input and output by applying the example configuration in the Bind DHCP Service with MPLS Labels *Bind DHCP Service with MPLS Labels* section.

Step 8 Save your configuration as described in the *Verifying and Saving Your Configuration* chapter in this guide.

Create VRF with Route-distinguisher and Route-target

Use this example to first create a VRF on the router and assign a VRF-ID. The second **`ip vrf`** command creates the route-distinguisher and route-target.

```
configure
   context <context_name> -noconfirm
```

```
ip vrf <vrf_name>
router bgp <as_number>
   ip vrf <vrf_name>
      route-distinguisher <as_value>
      route-target export <as_value>
      end
```

Set Neighbors and Address Family

Use this example to set the neighbors and address family to exchange routing information with a peer router.

```
configure
   context <context_name>
      ip vrf <vrf_name>
         router bgp <as_number>
         ip vrf <vrf_name>
            neighbor <ip_address> remote-as <AS_num>
            address-family <type>
            neighbor <ip_address> activate
            end
```

Redistribute Connected Routes

Use this example to redistribute connected routes between routing domains.

```
configure
   context <context_name>
      ip vrf <vrf_name>
         router bgp <as_number>
         ip vrf <vrf_name>
            address-family <type> vrf <vrf_name>
```

```
redistribute connected
exit
redistribute connected
end
```

Establish BGP Peering with Peer Router

Use this example to establish BGP peering with peer router.

```
configure
   context <context_name>
      router bgp <as_number>
         neighbor <ip_address> remote-as <AS_num>
         address-family <type>
         neighbor <ip_address> activate
         end
```

Configure Address Family

Use this example to configure the **address-family** and to **redistribute** the connected routes into BGP. This takes any routes from another protocol and redistributes them to BGP neighbors using the BGP protocol.

```
configure
   context <context_name>
      router bgp <as_number>
         address-family <type> vrf <vrf_name>
            redistribute connected
            exit
         redistribute connected
         end
```

Configure IP Pools with MPLS Labels

Use this example to configure IP Pools with MPLS labels for input and output.

```
configure
   context <context_name> -noconfirm
      ip pool <name> <ip_addr_mask_combo> private vrf <vrf_name> mpls-
label input <in_label_value> output <out_label_value1>
         interface <name> loopback
            ip address <ip_addr_mask_combo> srp-activate
            exit
         interface <name>
            ip address <ip_address subnet_mask>
            exit
         interface <name>
            ip address <ip_address subnet_mask>
            exit
         interface <name>
            ip address <ip_address subnet_mask>
            exit
         interface <name>
            ipv6 address <ip_addr_mask_combo>
            end
```

Bind DHCP Service with MPLS Labels

Use this example to bind DHCP service with MPLS labels for input and output.

```
configure
   context <dest_ctxt_name>
      dhcp-service <dhcp_svc_name>
```

```
bind address <ip_address> [ nexthop-forwarding-address
<nexthop_ip_address> [ mpls-label input <in_mpls_label_value> output
<out_mpls_label_value1> [ <out_mpls_label_value2> ]]]

end
```

Notes:

- To ensure proper operation, DHCP functionality should be configured within a destination context.
- Optional keyword **nexthop-forwarding-address <*nexthop_ip_address*> [mpls-label input <*in_mpls_label_value*> output <*out_mpls_label_value1*> [*out_mpls_label_value2*]]** applies DHCP over MPLS traffic.

Chapter 28
PDIF Session Recovery

In the telecommunications industry, over 90 percent of all equipment failures are software-related. With robust hardware failover and redundancy protection, any card-level hardware failures on the system can quickly be corrected. However, software failures can occur for numerous reasons, many times without prior indication. For this reason, we have introduced a new solution to recover subscriber sessions in the event of failure.

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.

The PDIF supports a session recovery procedure wherein the system can be configured to store redundant call state information and recover this information after certain types of hardware and software failures.

Important: Session Recovery can only be enabled through a feature use license key. If you have not previously purchased this enhanced feature, contact your sales representative for more information.

The session recovery feature, even when the feature use key is present, is disabled by default on the system.

This chapter provides information on the following topics:

Session Recovery

This section examines how the PDIF recovers from two types of failure: migration failures and task failures.

How Session Recovery Works in PDIF

PDIF call traffic is encrypted by the IPSec protocol. When the encrypted packet arrives on the ASR 5000, it first needs to be decrypted in the data path. The Daughter Card has an IPSec SA to do the decryption. Consequently, it is very important to have the Daughter Card recover before any software recovery in order to continue call traffic processing.

IPSec Controller does not send an IPSec Manager death notification to any subsystem. This allows both the Session Manager and Daughter Card to continue carrying subscriber traffic using NPU flows and IPSec SAs to transmit the data.

It also allows the Daughter Card to continue to receive and decrypt IPSec tunnel data.

When IPSec Manager recovers, IPSec Controller sends the configured PDIF crypto template to the IPSec Manager. Upon receiving the template, IPSec Manager initializes the template policy and creates the service entry for this template policy. It also initializes an IKEv2 stack instance. IPSec Manager uses the AAA API to recover calls in bulk. On receiving a response from the AAA, IPSec Manager repopulates each subscriber policy data structure. It also takes care of information that needs to be generated dynamically, (reprogramming the Datapath, creating skip lists, etc.)

To provide faster Datapath recovery without affecting call traffic, Daughter Card Manager is used to repopulate the Daughter Cards with the SAs faster and more efficiently; meanwhile IPSec Manager and Session Manager can recover in the background at relatively slower pace without the call data being interrupted.

On restart, IPSec Manager allocates the Datapath instance used to update the Daughter Card with the IPSec SAs and to create an NPU flow mapped to the Daughter Card IPSec SA entry.

When an IPSec tunnel is established, two IPSec SAs are created: one for receiving encrypted data and another for encrypting and transmitting encrypted data. In the interests of speed, a standby Daughter Card Manager on a standby PSC stores IPSec SAs for the whole system.

Once the PSC becomes active, Daughter Card Manager quickly programs the Daughter Card with available IPSec SAs. The most active calls are given priority in order to prevent traffic interruption.

Migration vs. Task Failure

Planned migration requires that all the task data be transferred over the new CPU. Task migration time (hence total outage time) depends upon task size. In general, a longer outage can be anticipated since the whole task has to be successfully migrated before it can be executed. Task failure, on the other hand, creates a new task that is operational right away. Thus, it can recover its own session while it services other sessions as they are recovered. So, the perceived outage is shorter in the case of failure recovery than for a migration.

Planned PSC Migration

This is the case when a system administrator decides to migrate an ASR 5000 application card (PSC). When the migration command is issued, all the tasks on affected card are notified to get ready for migration. After task migration is complete, all the tasks are notified and any post-migration adjustments will be done. At the end of the migration process, each IPSec Manager is returned to its previous normal operational state before it started migration.

Unplanned PSC Migration

This is the case when there is a system fault and a card has failed, or when an individual task fails. When tasks are restarted as a result of either card failure or task failure, it is recovered with saved information and put back into its previous state of operation. In either planned or unplanned recovery, only tunnels that are in an already established state can be recovered. Any non-established tunnels may be lost.

Hardware Requirements and Configuration

Because session recovery requires numerous hardware resources, such as memory, control processors, NPU processing capacity, etc., some additional hardware may be required to ensure that enough resources are available to fully support this feature.

Important: PDIF is designed to run on an ASR 5000 chassis using only SMC controller cards and PSC application cards.

Important: A minimum of four PSCs (three active and one standby) per individual chassis is required to use this feature.

To allow for complete session recovery in the event of a hardware failure during a PSC migration, a minimum of three active and two standby PSCs should be deployed.

To assist you in your network design and capacity planning, the following list provides information that should be considered.

- Subscriber capacity is decreased depending on the hardware configuration.A fully configured chassis (12 active PSCs and 2 standby PSCs) would experience a smaller decrease in subscriber capacity versus a minimally configured chassis (3 active PSCs and 2 standby PSCs).
- The amount by which control transaction processing capacity is reduced.
- The reduction in subscriber data throughput.
- The recovery time for a failed software task (e.g. session manager).
- The recovery time for a failed PSC (hardware failure).

If a PSC migration is being performed, this may temporarily impact the ability to perform session recovery as hardware resources (e.g. memory, processors, etc.) that may be needed are not available during this operation.To avoid this condition, a minimum of two standby PSCs should be configured.

Enabling or Disabling Session Recovery from the CLI

Enabling or disabling session recovery is done on a chassis-wide basis.

Enabling Session Recovery on an Out-of-Service System

The following procedure is for a system that does not have any contexts configured.

Step 1 At the Exec mode prompt, verify that the session recovery feature is enabled by the session and feature use license on the system by entering the following command:

```
show license info
```

The output of this command appears similar to the example shown below. Session Recovery is shown in bold. If there were no current license, another would have to be applied before Session Recovery could be enabled.

Enabled Features:

```
# Feature Applicable Part Numbers
# ---------------------------------------
# HA: [ 600-00-7502 / 600-00-7505
# 600-00-7592 / 600-00-7593 ]
# + RADIUS AAA Server Groups [ none ]
# IPv4 Routing Protocols [ none ]
# Proxy MIP: [ 600-00-7512 / 600-00-7549
# 600-00-8538 ]
# + FA [ none ]
# IPv6 [ 600-00-7521 / 600-00-7576 ]
# Lawful Intercept [ 600-00-7522 ]
# 600-00-7643 / 600-00-7663 ]
# Enhanced Lawful Intercept [ 600-00-7567 / 600-00-8534 ]
# PDIF: [ none ]
# + FA [ none ]
# + Session Recovery [ 600-00-7513 / 600-00-7546
```

```
# 600-00-7552 / 600-00-7554
# 600-00-7566 / 600-00-7594
# 600-00-9100 / 600-00-9101
# 600-00-7638 / 600-00-7640
# 600-00-7634 / 600-00-7595 ]
# + RADIUS AAA Server Groups [ none ]
# PDIF: [ 600-00-8539 ]
# + FA [ none ]
# + IPSec [ 600-00-7507 / 600-00-8537 ]
# + RADIUS AAA Server Groups [ none ]
# Session Limits:
# Sessions Session Type
# -------- -----------------------
# 20000 HA
```

Step 2 From the Global Config Mode, enter the following:

```
configure
   require session recovery
   end
```

Step 3 Save your configuration as described in *Verifying and Saving Your Configuration.*

Enabling Session Recovery on an In-Service System

When enabling session recovery on a system that already has a saved configuration, the session recovery commands are automatically placed before any service configuration commands in the configuration file.

Step 1 Check that the license supports Session Recovery as shown above. If not, replace the license before proceeding.

Step 2 Use the following configuration example to enable Session Recovery.

```
configure
   require session recovery
```

```
end
```

Important: This feature does not take effect until after the system has been restarted.

Step 3 Save your configuration as described in *Verifying and Saving Your Configuration.*

Step 4 Perform a system restart by entering the following command:

```
reload
```

The following prompt appears:

```
Are you sure? [Yes|No]:
```

Answer:

```
Yes
```

More advanced users may opt to simply insert the require session recovery command syntax into an existing configuration file using a text editor or other means, and then applying the configuration file manually. Caution should be taken when doing this to ensure that this command is placed among the first few lines of any existing configuration file to ensure that it appears before the creation of any non-local context.

Disabling the Session Recovery Feature

To disable the session recovery feature on a system, enter the following command from the Global Configuration mode prompt:

```
configure
   no require session recovery
   end
```

Important: If this command is issued on an in-service system, then the system must be restarted by issuing the **reload** command.

Preserved Session States

Session state is periodically stored (checkpointed) by using redundant processes to carry the existing call state information.

Preserved session states includes the following:

- Per-session idle timer (± one checkpoint interval)
- Per-session DPD timer (± one checkpoint interval)
- Per-session rekeying timer (± one checkpoint interval)
- Accounting packet transmission
- STR transmission

Scope of Data Recovery

In addition to maintaining call state information, information is kept in order to:

- Recover IPSec Manager policies, all template maps, and all subscriber maps.
- Use the policies (including templates) to recover pertinent CHILD SAs, flowIDs and statistics.
- Recover or reconfigure NPU flowids and data path handles.
- Recover and restore IKEv2 stack state for all tunnels.
- Supply the IKEv2 stack with needed data statistics to determine rekey and DPD states.
- Recover Diameter session information

Possible Recovery Failures

There are some situations wherein session recovery may not operate properly. These include:

- Additional software or hardware failures during the session recovery operation. An example of this would be if the Daughter Card Manager used to repopulate the Daughter Cards with the preserved SAs failed.
- A lack of hardware resources (i.e., PSC memory and control processors) to support session recovery.
- Some statistics, such as those collected and maintained on a per manager basis (AAA manager, session manager, etc.) are not recovered after the failure.

Show Session Recovery Status Command

Information about the current state of the session recovery subsystem is available from an Exec Mode command.

show session recovery status verbose

The output (below) shows the current state of the session recovery system. It displays all the tasks involved with the recovery process. It can also tell when was the last checkpoint (save) was done and how many sessions are to be recovered on each manager:

```
show session recovery status verbose

Session Recovery Status:

 Overall Status : Ready For Recovery

 Last Status Update : 10 seconds ago

 Overall Status Update : 3 seconds ago

 ----sessmgr--- ----aaamgr---- demux

 cpu state active standby active standby active status

---- ------- ------ ------- ------ ------- ------ -------------

 2/0 Standby 0 10 0 10 0 Good

 4/0 Active 0 0 0 0 4 Good (Demux)

 6/0 Active 10 1 10 1 0 Good

11/0 Active 10 1 10 1 0 Good

13/0 Standby 0 10 0 10 0 Good
```

Chapter 29
Policy-Based Management and EV-DO Rev A

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.

Important: The EV-DO Rev A features described in this chapter are only available if you have purchased and installed a session use license *EV-DO Rev A PDSN License* on your system. If you have not previously purchased this enhanced feature, contact your sales representative for more information.

Important: If the *EV-DO Rev A PDSN License* is not available or if the number of Rev A sessions exceeds the license limits, the Rev A calls are downgraded to Rev 0 calls.

EV-DO Rev A is an enhanced version of CDMA2000 EV-DO that provides the foundation for next-generation, IP-based, broadband wireless voice and data networks. EV-DO Rev A is backward compatible and interoperable with EV-DO Rev 0 networks and devices, with continued support for both Simple IP and Mobile IP sessions.

EV-DO Rev A includes higher data rates (3.1 Mbps on the forward link and 1.8Mbps on the reverse link) and improved Quality of Service (QoS) support.

This chapter describes the following:

Policy-based Management Overview

Policy-based management provides a way to allocate network resources; primarily network bandwidth, Quality of Service (QoS), security, and accounting according to defined business policies. These policies can be either static or dynamic. In a static policy the traffic flow is predefined by exact rules. For dynamic policies, the flows are created dynamically by external signaling. The dynamic policies validate the flows and apply the necessary policy actions required for each flow.

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 access control decisions, provide QoS enforcement and accounting decisions, etc.

A policy is defined as

policy: condition = > action

- **condition**: specifies the flow-parameters such as source-address, destination-address, source-port, destination-port, protocol, etc. for ingress and/or egress packets.
- **action**: specifies a set of treatments for a flow/packet when condition matches. These actions are typically based on flow classification and QoS processing for individual flows and DSCP marking.

Supported Standards

The following standards were referenced for the system's EV-DO Rev A implementation:

- CDMA2000 Wireless IP Network Standard: Introduction, X.P0011-001-D v 0.5
- CDMA2000 Wireless IP Network Standards: Quality of Service and Header Reduction, X.P0011-004-D v 0.5
- CDMA2000 Wireless IP Network Standards: Accounting Services and 3GPP2 RADIUS VSAs, X.P0011-005-D v 0.5
- Interoperability Specification (IOS) for CDMA2000 Access Network Interfaces - Part 7 (A10 and A11 Interfaces), A.S0017-C v0.4, IOS v5.0, June 2004
- Interoperability Specification (IOS) for High Rate Packet Data (HRPD) Radio Access Network Interfaces with Session Control in the Packet Control Function, V&V Version, A.S0009 v0.3, May 2005
- Perkins, IPv4 Mobility, RFC 3344, August 2002
- Braden, R.,et al, Resource ReSerVation Protocol (RSVP) -- Version 1, RFC 2205, September 1997, Proposed Standard.

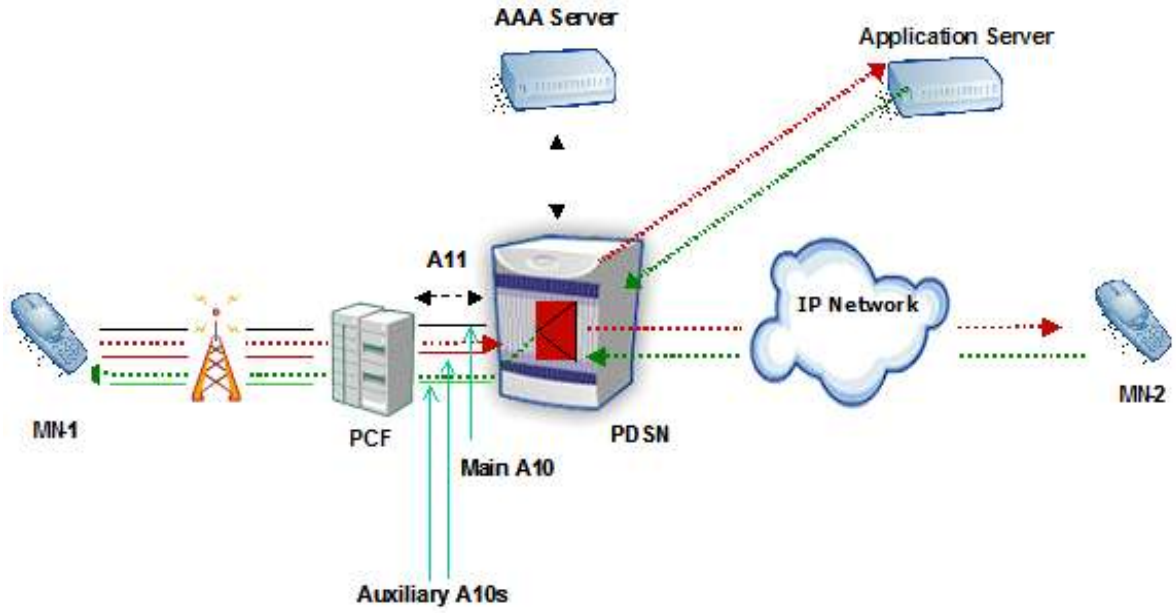
Quality of Service in EV-DO Rev A

The QoS feature in Rev A enables a Mobile Node (MN) to negotiate and setup QoS for the air interface with the RAN while using the PDSN for QoS authorization and classifying user traffic flows. Network entities including the PDSN, PCF, AAA server, RAN and the mobile are all involved in setting up the desired QoS parameters for the MN. QoS may be applied for both Simple IP and Mobile IP calls.

The QoS is created in a subscriber profile on the AAA server. When the subscriber is authenticated, the AAA server passes the subscriber's QoS on to the PDSN, which then passes the information on to the PCF. The PCF determines whether or not to grant the subscriber the QoS.

The following graphic shows an EV-DO Rev A network.

Figure 42. EV-DO Rev A Network



The interface between the MN and the RAN can support multiple simultaneous connections of the same or different packet data service options. These connections are referred to as packet data flows or link flows.

The three main entities involved in the QoS procedures for Rev A are the MN, RAN and the PDSN. The PDSN authenticates the MN during call setup and obtains the QoS profile for the subscriber. The PDSN conveys the subscriber's QoS profile to the PCF using A11 messaging. The PDSN and RAN store the QoS profile attributes.

Important: The RAN can map multiple IP flows to a single A10 connection.

When the MN requests QoS parameter setup with the RAN, the RAN verifies the request using a stored QoS profile. If the subscriber is authorized for the requested QoS parameters, the RAN provides the MN with a granted QoS profile and also sends the granted QoS profile to the PDSN via A11 messaging. If QoS was granted, the PCF sets up new auxiliary A10 connections for carrying traffic corresponding to the granted QoS and maps the requested flows over the

A10 connections. The PCF uses A11 messaging to signal new A10 connection creation and to convey the requested and granted QoS parameters and the mapping of flows to A10 connections.

Important: If the subscriber is not authorized, the RAN denies the requested QoS profile to the MN and downgrades the flow to best effort traffic.

The MN sends Traffic Flow Templates (TFTs) that identify different IP flows in both the forward and reverse directions to the PDSN. The PDSN stores the TFTs. For traffic towards the mobile network, the PDSN uses the TFTs to identify the traffic flows and to map the flows to different A10 connections and for flow-based accounting. For traffic from the mobile network, the PDSN looks up the flow for per-flow based accounting. The PDSN also tracks accounting per A10 connection in addition to a per-flow basis.

Flow Mapping

Flow mapping includes these basic areas:

- Basic TFT processing
- Forward Traffic Processing

Basic TFT processing

The MN may be assigned one IP address. This IP address is not associated with any particular A10 connection. The MN may send Traffic Flow Templates (TFTs) for flow mapping to the PDSN carried over RSVP reservation messages. The TFTs are used to map forward traffic to the main or auxiliary A10 connections and to indicate if a specific flow treatment (such as header compression) should be applied for the matching packets. The TFTs are also used by PDSN to associate packets from forward and reverse traffic to flows for flow-based accounting and QoS enforcement. Since the MN may be assigned multiple IP addresses, each TFT corresponds to a specific IP address.

The TFTs can contain:

- Source and destination ports
- Source and destination addresses
- Protocol
- Type of service (TOS in IPv4)

After the main and auxiliary A10 connections are created, the MN sends TFTs to the PDSN via a reservation message (RSVP).

Forward Traffic Processing

When a packet arrives at the PDSN from the external IP network, the packet is matched against the set of packet filters stored at the PDSN. The packet is sent down the A10 connection that matches the flow id in the packet filter.

If flow treatment is specified for that matching packet filter, such as ROHC compression, the specified treatment is applied to the packets. Otherwise, the default treatment is applied. If no packet filters match the incoming packet, the PDSN forwards the packet over the main A10 connection.

EV-DO Rev A Call Setup

The following figure describes the setup procedures for Rev A calls. Note that Rev A supports both Simple IP and Mobile IP calls.

Figure 43. EV-DO Rev A Call Setup

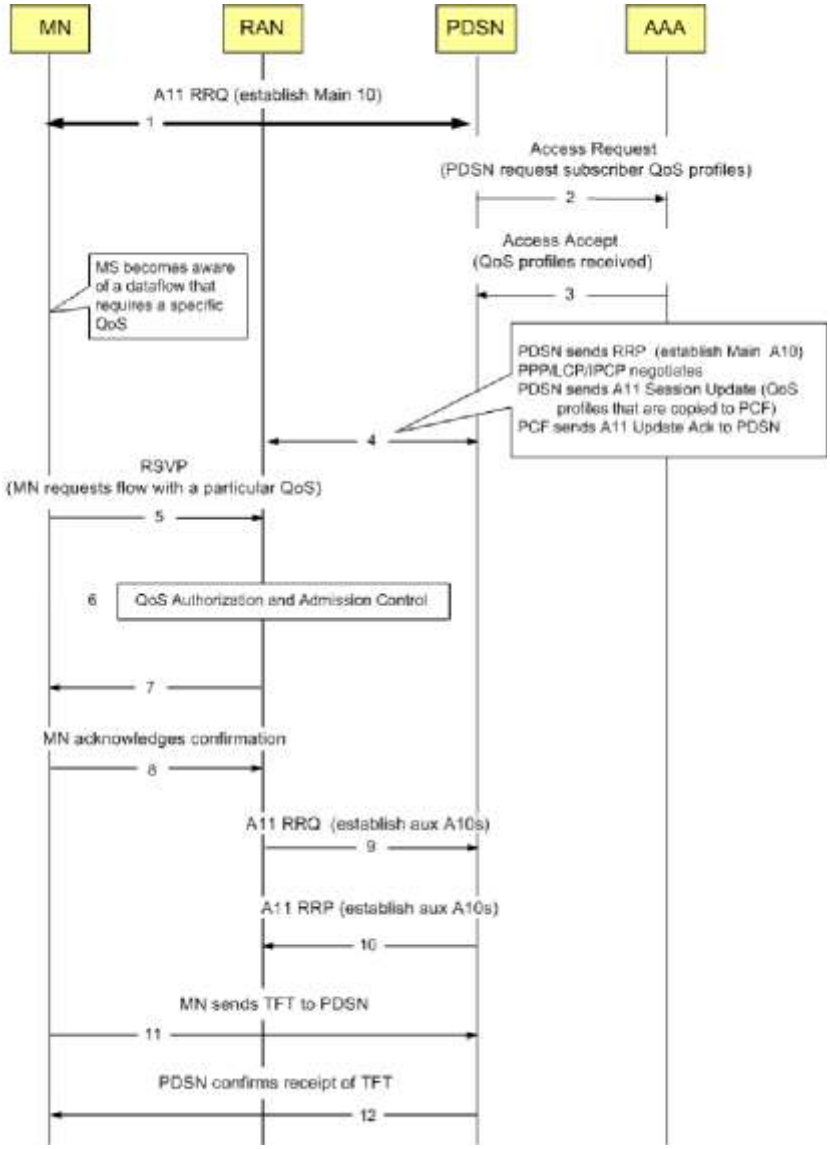


Table 32. EV-DO Rev A Call Setup Description

Step	Description
1	Main service connection is setup. Here, the PDSN receives an A11 RRQ indicating that this is a request to setup the main service connection for a Rev A call. For main A10s, the A11 RRQ always includes an SR_ID of 1, a forward traffic flow ID of 255, and a reverse traffic flow ID of 255.
2	The PDSN sends an Access Request message to the AAA to authenticate the subscriber and to retrieve the subscriber's QoS profile. (If no authentication is required, the PDSN uses the QoS profile from the default subscriber template.)
3	The AAA sends the subscriber's profile that includes the QoS profile to the PDSN.
4	The PDSN stores the QoS attributes from the subscribers profile and sends a copy to the RAN via A11 Session Update message The RAN also stores the QoS attributes.
5	The AT sends an IP flow reservation message to the RAN that includes the requested levels of QoS for each data flow.
6	The RAN determines if the subscriber is authorized using the QoS attributes it stored from step 4. If the subscriber is authorized, the RAN sends a QoS Granted message to the AT. Otherwise, the RAN denies the QoS requested by the RAN.
7	The AT sends the RAN an Acknowledgment.
8	The RAN sends an A11 RRQ to the PDSN that includes SR_IDs for the auxiliary A10 connections and the QoS granted the MS.
9	The PDSN verifies the QoS parameters granted to the MS using the attributes it stores in step 4 and returns an A11 RRP.
10	The AT sends the TFTs—which identify the different flows in both the forward and reverse directions—to the PDSN via modified RSVP Reservation message. (RSVP formats the TFTs for the PDSN.)
11	The PDSN sends Reservation Confirm message to the AT and uses the packet filters within the TFTs to map flows to the auxiliary A10 connections. The PDSN also stores the TFTs for flow look up when mapping flows in the reverse direction and for flow-based accounting. During intra-PDSN handoff, PDSN will send the subscriber QoS Profile that it received in the Access-Accept during authentication to the new PCF. This includes the list of authorized profile IDs in addition to other QoS attributes, but this does not contain the requested/granted profile IDs. The PDSN may send the profile in the A11 RRP or generates an RP Session Update message and sends it to the PCF. The system assumes that the AT uses the same NAI, even if it initiates multiple simultaneous sessions.

Call Flow for Updating QoS for Dynamic Flows

The PDSN and the PCF can perform QoS update procedures to modify the QoS that is granted for an IP flow. The QoS update procedure is applicable to dynamic flows only.

You can configure the system to ensure that the granted QoS is downgraded to best effort (i.e. class x to best effort but not class x to class x-1). The reservation for the downgraded flow is removed at the RNC and PCF and the user detail record for the flow is closed at the PDSN.

In a downgrade scenario, the PDSN downgrades the profile ID for a flow from the Profile ID that was originally granted by the RAN. The profile ID is downgraded to a lower precedence profile ID or to best effort.

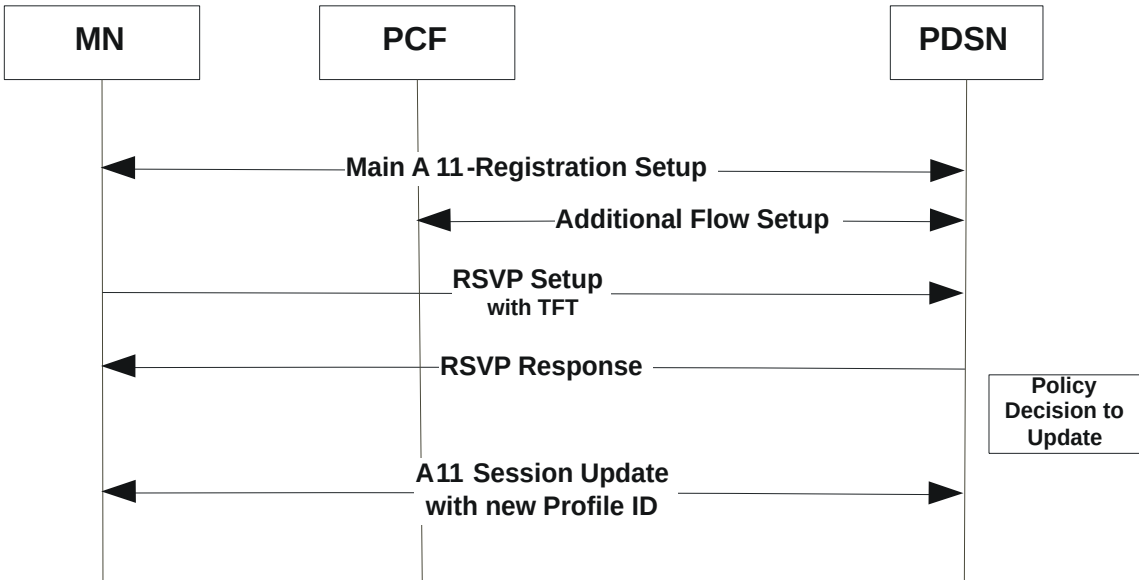
Important: If the *EV-DO Rev A PDSN License* is not available or if the number of Rev A sessions exceeds the license limits, the Rev A calls are downgraded to Rev 0 calls. If the profile ID does not match, the flow will be downgraded to best effort.

In an update scenario, the updates may be triggered by one of the following:

- TFT validation failure—The MN sent a TFT for a flow/profile id that violates the configured policy for the flow/profile-id.
- Traffic Violation: The MN is using the wrong link flow to send reverse data.
- Operator triggered: The Operator triggered an update through the CLI.
- Traffic Policing: Traffic policing triggered an update.

The following call flow describes a successful flow setup with QoS update from PDSN.

Figure 44. Successful Flow Setup with QoS Update from PDSN



1. The Main A11 registration setup includes the following actions:
   - The High Rate Packet Data (HRPD) connection is established over the radio link between the MN and the PCF.
   - The PCF recognizes that no A10 connection associated with the MN is available and selects a PDSN. The PCF sends an A11-Registration Request message to the PDSN. The PCF also includes the Main Service Option for HRPD (SO59) in the message to indicate to the PDSN that this is the main service connection. The PCF includes the Connection Setup Airlink record in this message. If the main connection is active, and the PCF has received enough information about the flow, the PCF also includes the Active-Start airlink record for the main A10 connection.
   - The A11-Registration Request message is validated and the PDSN accepts the connection by returning an A11-Registration Reply message and acceptance. The A10 connection binding information at the PDSN is updated to point to the PCF.
   - The PDN sends a list of authorized profile ids specified in the Subscriber profile in A11-Session-Update message to the PCF. Then the PCF sends Session Update Ack back to the PDSN.

2. The PCF sends an A11 Registration Request message indicating any auxiliary A10 (if it is required), the complete flow mapping information, requested profile ids from the mobile and the granted profile id for each flow.
    - The PDSN sends an A11 Registration Reply message including all the auxiliary A10s it has for the session.
    - The MN sends the TFT for the flow to be setup with the packet filters that identify the flow.
    - The PDSN sends a confirmation back to the MN to accept the valid TFT.
3. The PDSN updates the granted QoS for the flow because of a policy update trigger (QoS policy trigger, Operator trigger etc).
    - The PCF sends a Session Update acknowledge message indicating that it accepted the profile id change.
    - The PCF sends a new Registration Request message, with the new granted profile id and the flow mappings.
4. The A11 Session updates with the new profile IDs is complete.
   The PDSN accepts the Registration Request message, updates the flow mappings, and responds with an RRP.

EV-DO Rev A Important Commands

Important: If you are using EV-DO Rev A without Intelligent Traffic Control (ITC), you must define the type of traffic policing as **dynamic**. If you are using EV-DO Rev A with ITC, you can use commands to set flow-based policies using class-map, policy-map, and policy groups. Refer to EV-DO Rev A with ITC Support for an overview of Rev A and ITC. Refer to the *Intelligent Traffic Control* chapter or additional information on configuring flow-based traffic policies.

The commands listed in the following table are used to configure and monitor EV-DO Rev A commands. Detailed information, including description, syntax, and usage for each of these commands can be found in the *Command Line Interface Reference*.

Table 33. EV-DO Rev A Commands

Command	Mode	Description
Configuration Commands		
data-over-signaling	PDSN Service Configuration	Enables or disables the data-over signaling feature for A10 packets.
ip allowed-dscp	Subscriber Configuration	Sets the QoS Differentiated Services (DiffServ) marking that a subscriber session is allowed.
ip qos-dscp	Subscriber Configuration	Configures quality of service options for the current subscriber using the differentiated services code point method.
qos traffic-police	Subscriber Configuration	Sets the quality of service setting to the system default.
Monitor Commands		
logging filter active facility sessmgr level debug	Exec	Can be used to verify policy is applied.
monitor protocol A10, A11, User L3	Exec	Can be used to verify signal and data between the PDSN and PCF, and between the PDSN and mobile.
show rp full	Exec	Can be used to verify Session Update Send Reason.
show rp full all	Exec	Can be used to verify GRE Key associated with A10 instance.
show rsvp	Exec	Can be used to verify RSVP related statistics/counters.
show subscribers	Exec	Can be used to verify summary rev-A flow and filter information.
show subscribers all	Exec	Can be used to verify access technology.
show subscribers full	Exec	Can be used to verify subscriber policy.

RADIUS Accounting for EV-DO Rev A

RADIUS accounting for EV-DO Rev A traffic can either be session-based or flow-based. By default the system is set to session-based accounting. In session-based accounting, only a single accounting message is generated for the subscriber session, not separate accounting messages for individual A10 connections or flows. When the system is set for flow-based accounting, separate messages are sent for each flow and A10 sessions. Flow-based accounting includes the following options:

- **all-flows:** Generates separate RADIUS accounting messages per access flow. Separate accounting messages are not generated for data path connections. (For example, separate messages are not sent for the main A10 or auxiliary connections.)
- **auxiliary-flows:** Generates RADIUS accounting records for the main data path connection and for access-flows for all auxiliary data connections. (For example, separate RADIUS accounting messages are generated for the main A10 session and for access-flows within auxiliary A10 connections. The main A10 session accounting does not include octets or other accounting information from the auxiliary flows.)
- **main-a10-only:** Configures access-flow-based single accounting messages (for example only single start/interim/stop) are generated for the main A-10 flows only.
- **none:** Separate RADIUS accounting messages are generated for all data path connections (for example, PDSN main or auxiliary A10 connections) but not for individual access-flows. This is essentially A10 connection-based accounting.

RADIUS Attributes

The RADIUS attributes listed in the following table are used to configure EV-DO Rev A accounting for subscribers configured on remote RADIUS servers. More information on these attributes can be found in the *AAA Attribute Reference*.

Attribute	Description
3GPP2-Flow-Id	Displays the 3GPP2-Flow-Id.
3GPP2-Flow-Status	Displays the 3GPP2-Flow-Status.
3GPP2-Release-Indicator	Displays the 3GPP2-Release-Indicator.
3GPP2-Rev-Pdch-Rc	Displays the 3GPP2-Rev-Pdch-Rc.
3GPP2-Subnet	Displays the 3GPP2-Subnet.

The RADIUS attributes listed in the following table are used to configure RSVP Signaling for subscribers configured on remote RADIUS servers. More information on these attributes can be found in the *AAA Attribute Reference*.

Attribute	Description
ThreeGPP2-Rsvp-Sig-Inbound-count	Specifies the RSVP signaling octets sent by the Mobile Node.

Attribute	Description
ThreeGPP2-Rsvp-Sig-In-Pkts	Specifies the Number of RSVP signaling packets sent by the Mobile Node.
ThreeGPP2-Rsvp-Sig-Outbound-count	Specifies the RSVP signaling octets sent to the Mobile Node.
ThreeGPP2-Rsvp-Sig-Out-Pkts	Specifies the Number of RSVP signaling packets sent to the Mobile Node.

These attributes can be found in the StarentVSA and the StarentVSA1 dictionaries.

EV-DO Rev A with ITC Support

Important: The EV-DO Rev A features described in this section are only available if you have purchased and installed session use licenses *EV-DO Rev A PDSN License* and *Intelligent Traffic Control License* on your system. The Intelligent Traffic Control license enables policy related commands which allow DSCP marking, traffic policing, DOS Marking, etc. If you have not previously purchased these enhanced features, contact your sales representative for more information.

You can configure your system to support both EV-DO Rev A and Intelligent Traffic Controls (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 Intelligent Traffic Controls (ITC) and configuring flow-based traffic policing on your system, refer to the *Flow-based Traffic Policing Configuration* section in this manual.

Flow-based Traffic Policy

Flow-based traffic policies and traffic policing are 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 with the following building blocks:

- **Class Maps**: The basic building block of a flow-based traffic policing. Class maps are used to control over the packet classification.
- **Policy Maps**: A more advanced building block for flow-based traffic policing. Policy maps manage admission control based on the Class Maps and the corresponding flow treatment based on QoS traffic policing or QoS DSCP marking.
- **Policy Group**: This is a set of one or more Policy Maps applied to a subscriber. Policy groups also resolve conflict if a flow matches multiple policies.

Important: For more information on Intelligent Traffic Controls (ITC) and configuring flow-based traffic policies on your system, to the *Flow-based Traffic Policing Configuration* section in this manual.

ITC Important Commands

Important: If you are using EV-DO Rev A with ITC, you can use commands to set flow-based policies using class-map, policy-map and policy groups. Refer to EV-DO Rev A with ITC Support for an overview of Rev A and ITC. Refer to the *Intelligent Traffic Control* chapter for additional information on configuring flow-based traffic policies.

The commands listed in the following table are used to configure and monitor ITC. Detailed information, including description, syntax, and usage for each of these commands can be found in the *Command Line Interface Reference*.

Table 34. ITC Commands

Command	Mode	Description
Configuration Commands		
match dst-ip-address	Class-Map Configuration Mode	Specifies a traffic classification rule based on the destination IP address of packets.
match dst-ip-address	Class-Map Configuration Mode	Specifies a traffic classification rule based on the destination IP address of packets.
match dst-port-range	Class-Map Configuration Mode	Specifies a traffic classification rule based on the range of destination ports of L4 packets.
match ip-tos	Class-Map Configuration Mode	Specifies a traffic classification rule based on the IP Type of Service value in ToS field of packet.
match packet-size	Class-Map Configuration Mode	Specifies a traffic classification rule based on the size of packet. ***Important:*** This is only applicable for static policies.
match src-port-range	Class-Map Configuration Mode	Specifies a traffic classification rule based on the range of source ports of L4 packets.
match srp-ip-address	Class-Map Configuration Mode	Specifies a traffic classification rule based on the source IP address of packets.
class-map	Context Configuration Mode	Deletes/creates and enters the Class-Map configuration mode within the current destination context to configure the match rules for packet classification to flow-based traffic policing for a subscriber session flow.
policy-map	Context Configuration Mode	Deletes/creates and enters the Policy-Map configuration mode within the current destination context to configure the flow-based traffic policing for a subscriber session flow.
policy	Traffic Policy Group Configuration	Assigns the traffic policies, pre-configured in Policy-Map configuration mode, to a Policy Group for flow-based traffic policing to a subscriber session flow.
qos encaps-header dscp-marking	Traffic Policy-Map Configuration	Specifies the DSCP code value marked in IP header of packet/flow to determine the QoS for traffic policing.
qos traffic-police	Traffic Policy-Map Configuration	Enables and configures QoS policy for the flow-based traffic policing to subscriber session flow on per-flow basis.

Command	Mode	Description
3gpp2 data-over-signaling marking	Traffic Policy-Map Configuration	Indicates 3GPP2 related traffic flow for data over signaling channel.
Monitor Commands		
logging filter active facility sessmgr level debug	Exec	Can be used to verify policy is applied, A10 is added, flow is added, tft is validated, etc.
monitor protocol A10, A11, User L3	Exec	Can be used to verify signal and data between the PDSN and PCF, and between the PDSN and mobile.
show rp full	Exec	Can be used to verify Session Update Send Reason.
show rp full all	Exec	Can be used to verify GRE Key associated with A10 instance.
show rsvp	Exec	Can be used to verify RSVP related statistics/counters.
show subscribers	Exec	Can be used to verify summary rev-A flow and filter information.
show subscribers access-flows	Exec	Can be used to verify summary of flow-mapping, QoS profile, and flow-based policy information.
show subscribers access-flows full	Exec	Can be used to verify details of flow-mapping.
show subscribers all	Exec	Can be used to verify access technology.
show subscribers full	Exec	Can be used to verify subscriber policy.
show subscribers tft	Exec	Can be used to verify details of installed traffic flow template.

DSCP Marking Commands

Policy CLI commands are used to configure the policy for the Differentiated Services Code Point (DSCP) marking. DSCP marking includes configurations to set the DSCP for the inner or outer packet or to copy the inner marking to the outer packet.

DSCP markings are applied to signaling and bearer packets destined for the Radio Network Controller (RNC) (in the forward direction) and to the Application Server (AS) in the reverse direction.

The *Command Line Interface Reference* includes several chapters with more specific information about commands needed to configure DSCP. Refer to the following chapters:

- Class-map Configuration Mode Commands
- Traffic Policy-map Configuration Mode Commands
- Traffic Policy-group Configuration Mode Commands
- PDSN Service Configuration Mode Commands

Chapter 30
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

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 as described in the *Saving Your Configuration* chapter.

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 as described in the *Saving Your Configuration* chapter.

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 as described in the *Saving Your Configuration* chapter.

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 as described in the *Saving Your Configuration* chapter.

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 as described in the *Saving Your Configuration* chapter.

Chapter 31
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

The system supports pre-paid billing for subscriber accounts that use RADIUS Accounting. 3GPP2 pre-paid billing support is disabled by default and you must obtain and install a license key to enable it.

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 bye-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.

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 as described in the *Verifying and Saving Your Configuration* chapter.

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 as described in the *Verifying and Saving Your Configuration* chapter.

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 accounting13. 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-vsa1
- starent-vsa1-835

For more information, refer to the *AAA 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-VSA1
- Starent-835
- Starent-VSA1-835
- custom1 through custom9

Refer to the *AAA Interface Administration and Reference* for more details.

Chapter 32
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

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

Important: This feature is enabled as part of a license bundle or with the purchase of a standalone Proxy-MIP license. Other licenses might be required to enable all the features described in this chapter. If you do not have the appropriate license(s), please contact your sales advisor.

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 35. Applicable Products and Relevant Sections

Applicable Product(s)	Refer to Sections
PDSN	• 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	• 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	• 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	• 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 45. AAA/PDSN Assigned IP Address Proxy Mobile IP Call Flow

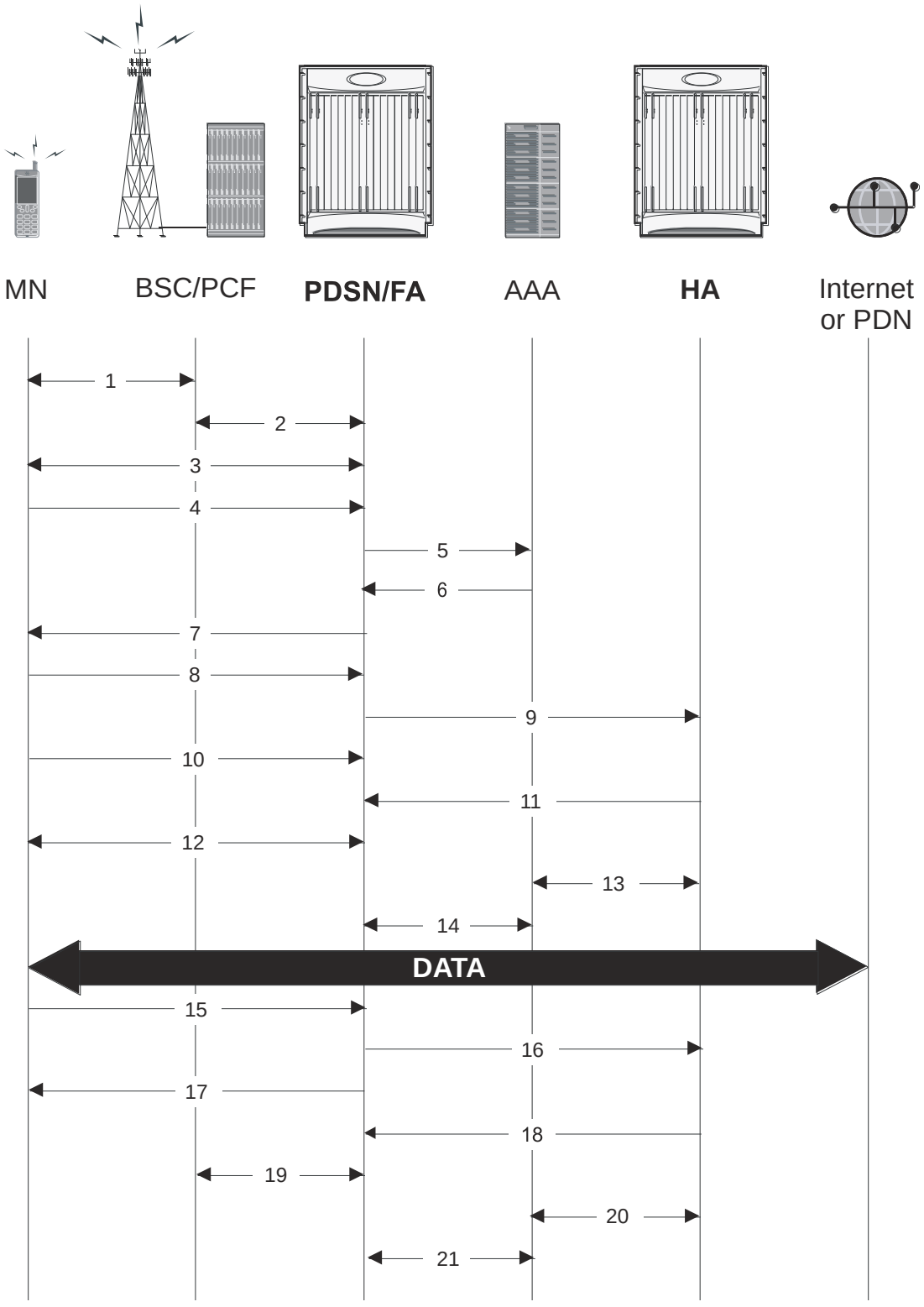


Table 36. AAA/PDSN Assigned IP Address Proxy Mobile IP Call Flow Description

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 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 it's 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 46. HA Assigned IP Address Proxy Mobile IP Call Flow

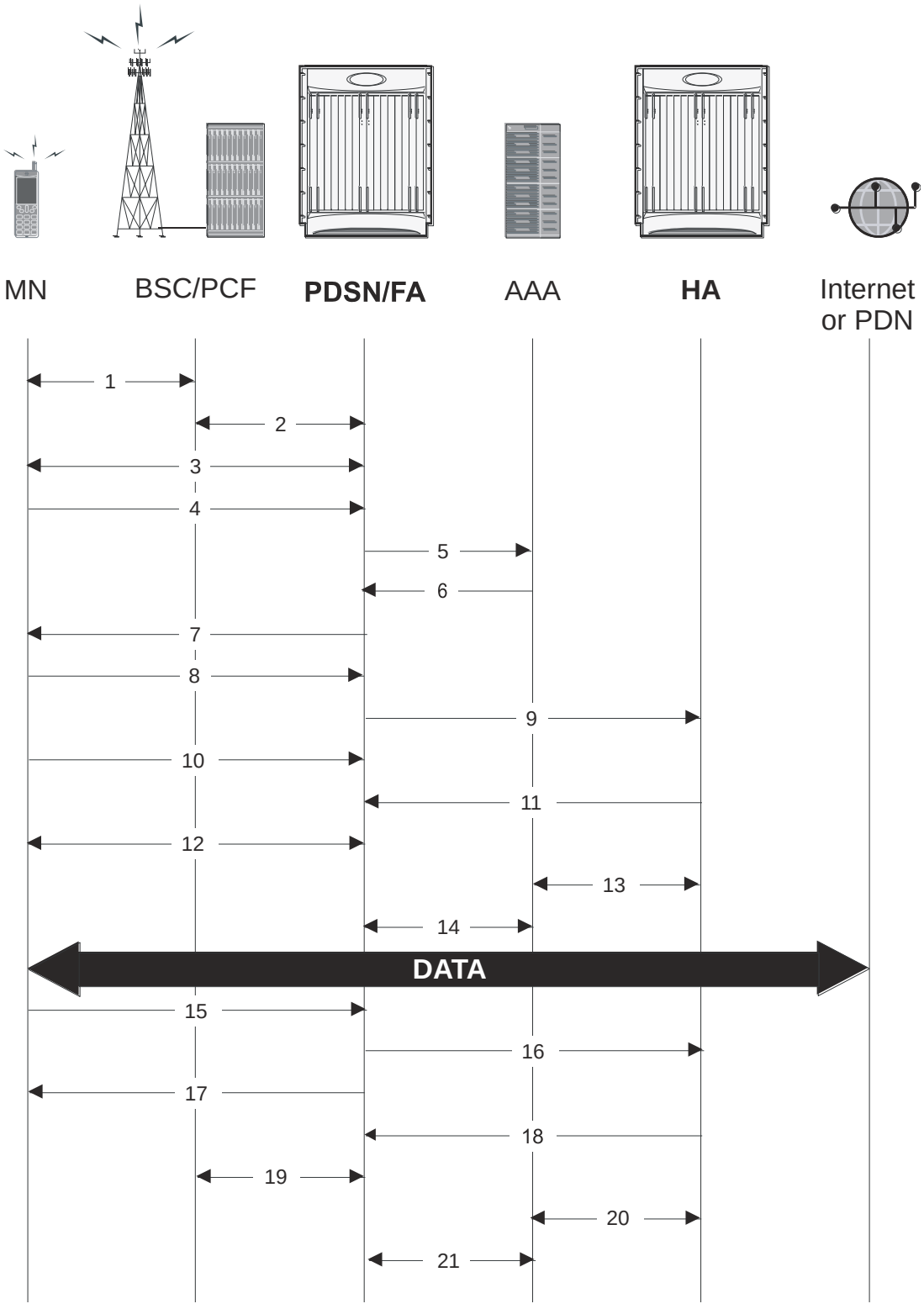


Table 37. HA Assigned IP Address Proxy Mobile IP Call Flow Description

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 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 a sample successful Proxy Mobile IP session setup call flow in 3GGP service.

Figure 47. Proxy Mobile IP Call Flow in 3GPP

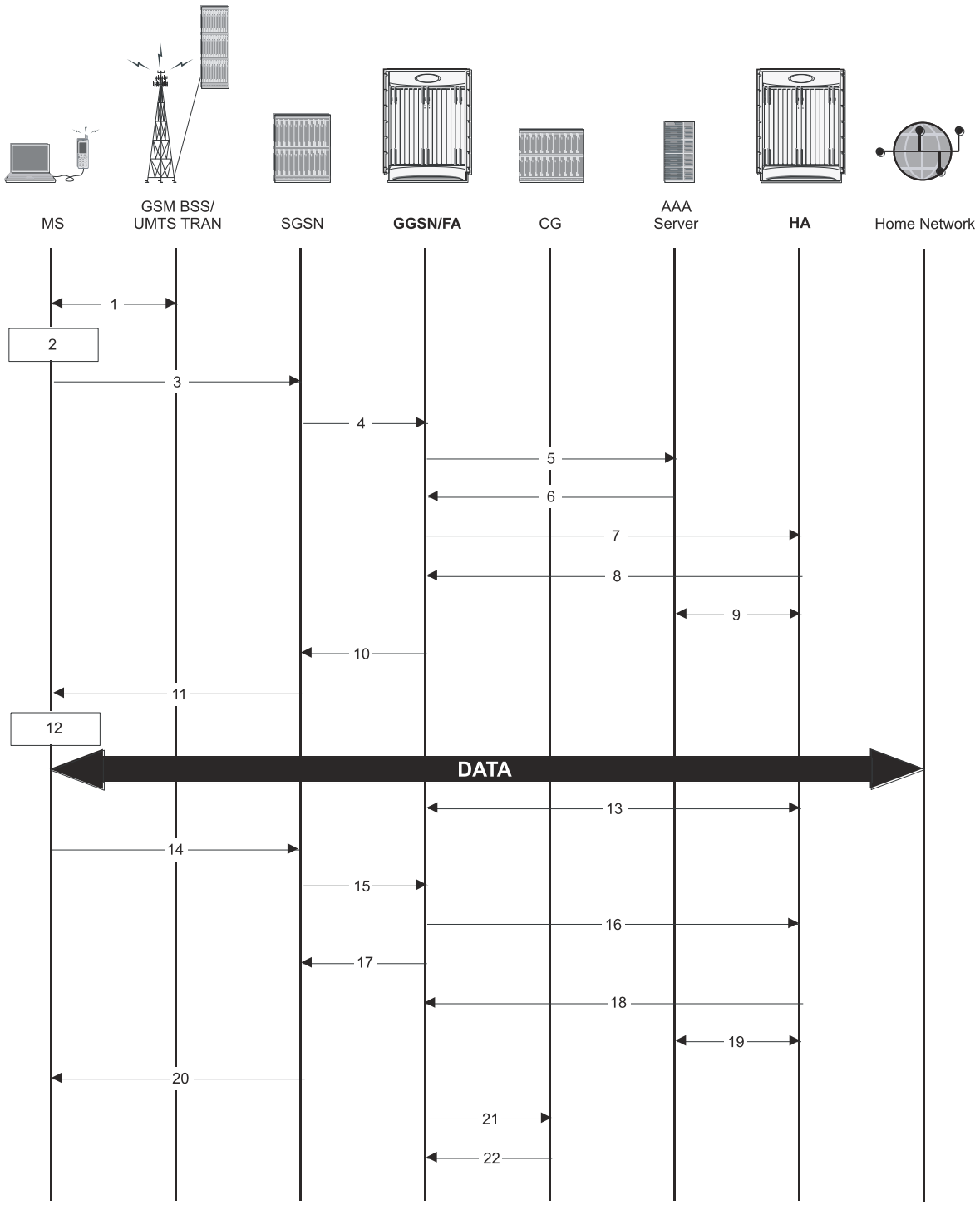


Table 38. Proxy Mobile IP Call Flow in 3GPP Description

Step	Description

Step	Description
1	The mobile station (MS) goes through the process of attaching itself to the GPRS/UMTS network.
2	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. 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. 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.
3	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.
4	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).
5	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. 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. 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. 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.
6	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.
7	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)).
8	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.
9	The HA sends an RADIUS Accounting Start request to the AAA server which the AAA server responds to.
10	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.
11	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.
12	The MT, will respond to the TE's IPCP Config-request with an IPCP Config-Ack message. 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.
13	The FA periodically sends Proxy Mobile IP Registration Request Renewal messages to the HA. The HA sends responses for each request.
14	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.

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 48. AAA/ASN GW Assigned IP Address Proxy Mobile IP Call Flow

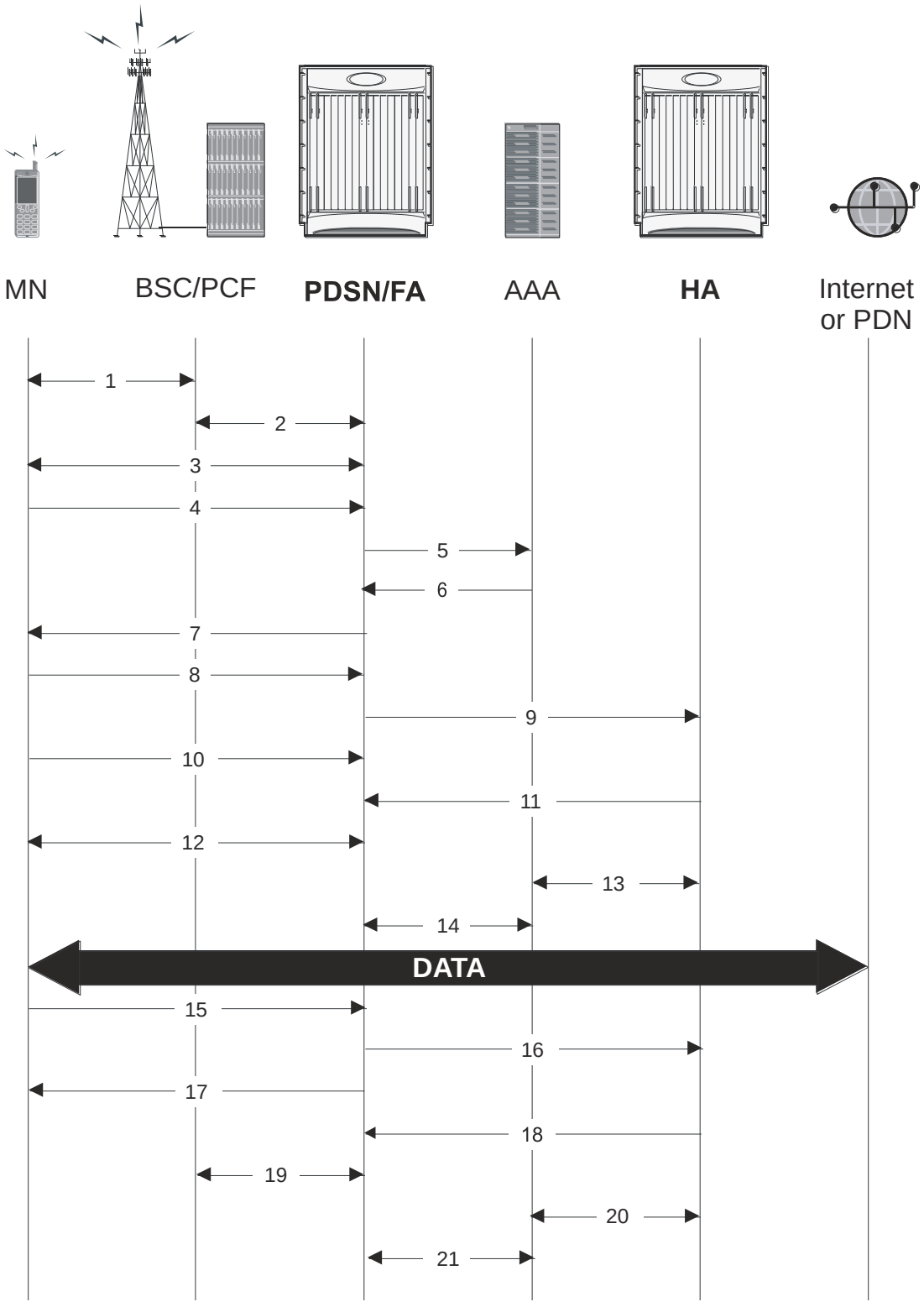


Table 39. AAA/ASN GW Assigned IP Address Proxy Mobile IP Call Flow Description

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.
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 it's 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 49. HA Assigned IP Address Proxy Mobile IP Call Flow

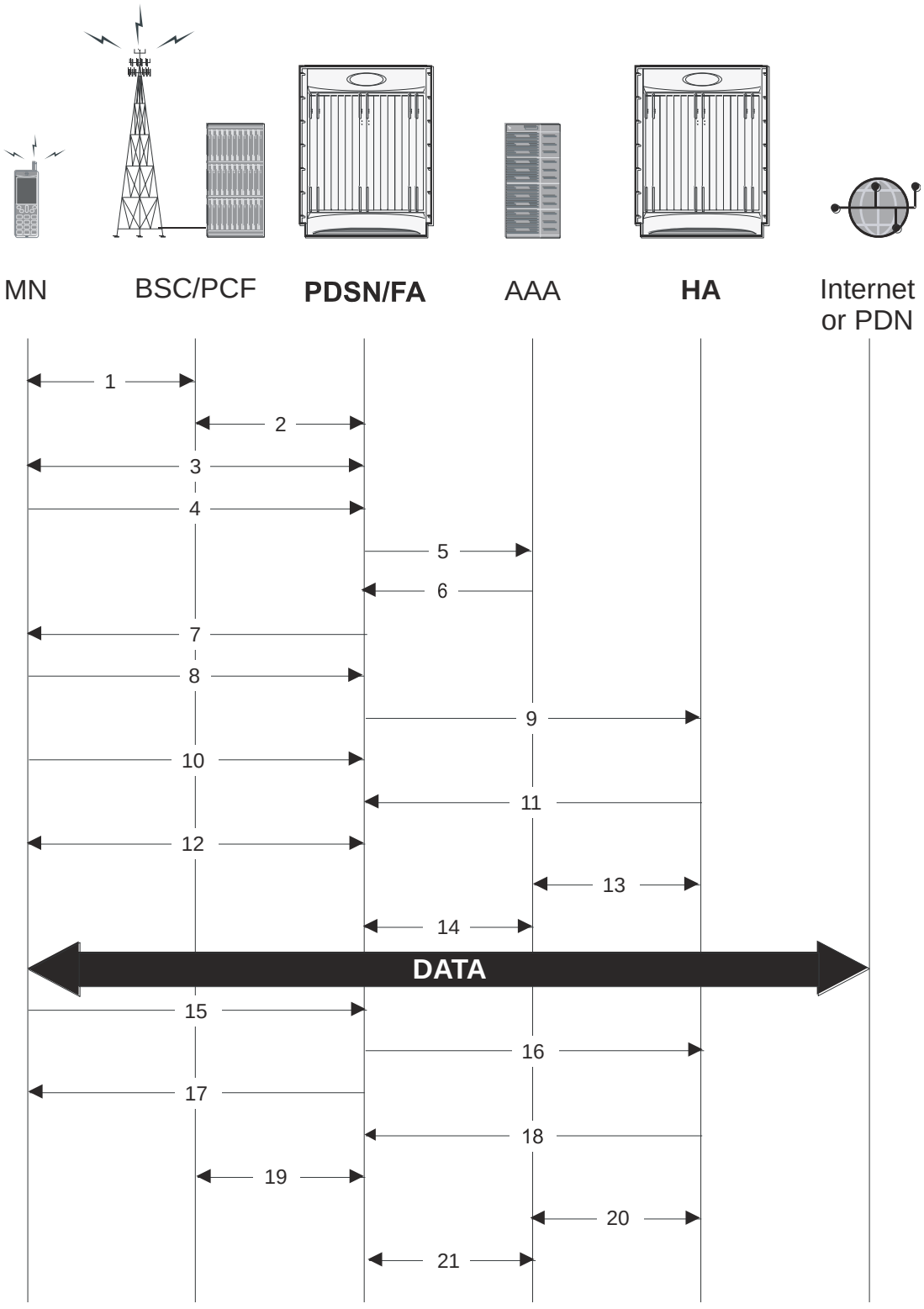


Table 40. HA Assigned IP Address Proxy Mobile IP Call Flow Description

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.
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 50. Proxy-MIP Call Setup using CHAP Authentication

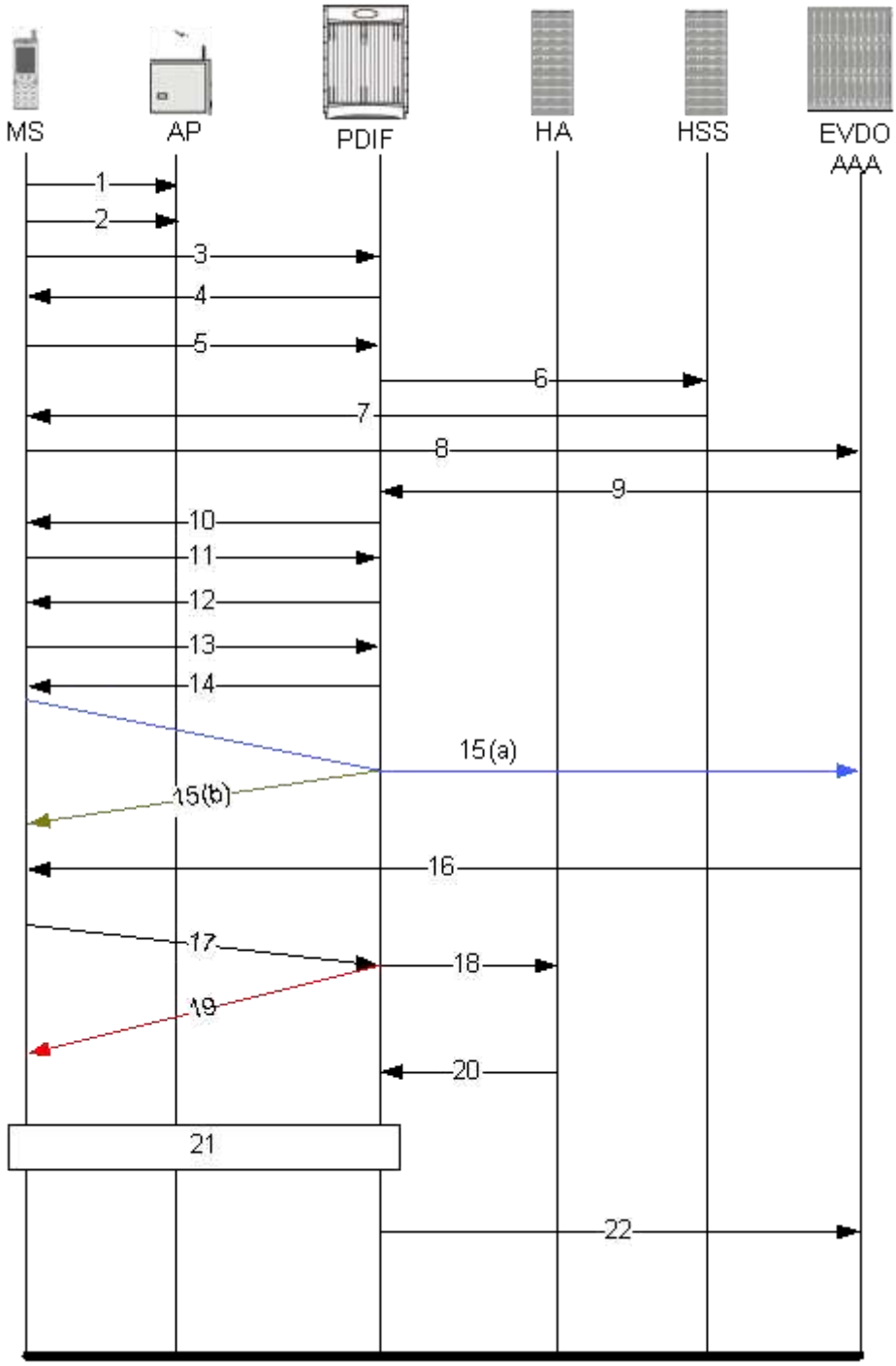


Table 41. Proxy-MIP Call Setup using CHAP Authentication

Step	Description

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.
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, TSi, 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	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. a. If the configured method is EAP-MD5, PDIF sends the IKE_AUTH Response with EAP payload including challenge.b. If the configured method is EAP-GTC, PDIF sends the IKE_AUTH Response with EAP-GTC.c. MS processes the IKE_AUTH Response: • 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)	PDIF receives the new IKE_AUTH Request from MS. 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.).
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 51. Proxy-MIP Call Setup using PAP Authentication

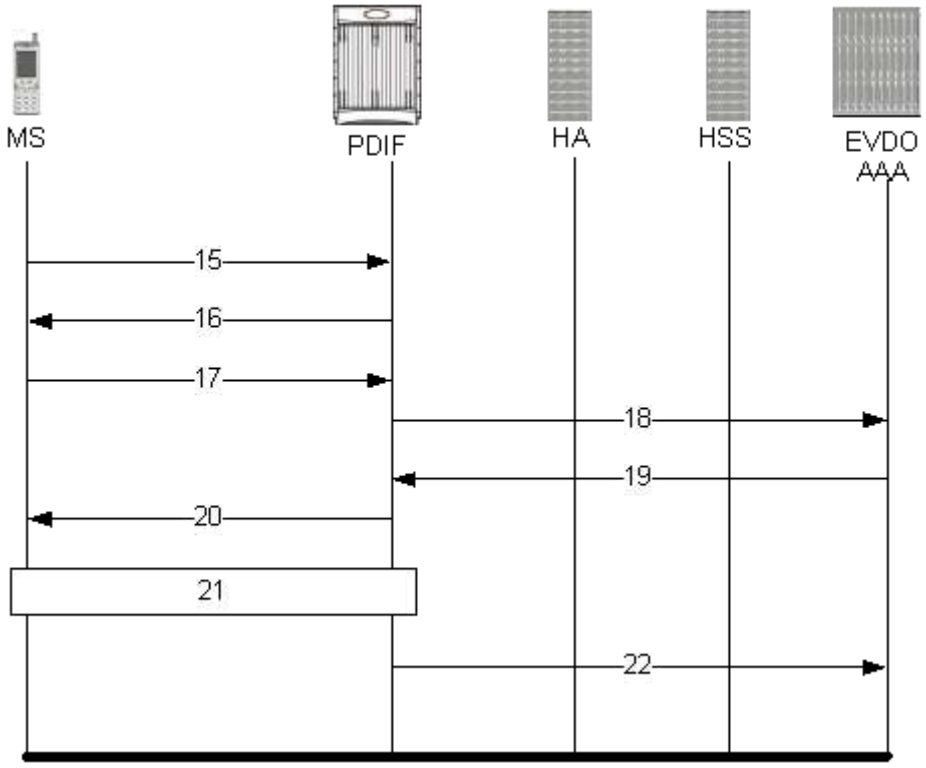


Table 42. Proxy-MIP Call Setup using PAP Authentication

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 as described in *Verifying and Saving Your Configuration*.

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 as described in *Verifying and Saving Your Configuration*.

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 as described in *Verifying and Saving Your Configuration*.

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 43. 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.	• 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.	• Disabled - do not perform compulsory Proxy-MIP (0) • Enabled - perform compulsory Proxy-MIP (1)

Attribute	Description	Values
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. ***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.	• Disabled (0) • Enabled (1)
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.	• 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 as described in *Verifying and Saving Your Configuration*.

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 as described in *Verifying and Saving Your Configuration*.

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 enables 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
	Displays configuration information for all configured APN.
	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 as described in *Verifying and Saving Your Configuration*.

Chapter 33
QoS Management

This chapter describes the Quality of Service (QoS) management on ST16 and Cisco® ASR 5000 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.

This chapter describes the following:

Introduction

The QoS Traffic Policing functionality supported by the GGSN implements QoS for subscribers based on the configuration of the APN template used as described in *Traffic Policing and Shaping* in this guide. As a result, all subscriber PDP contexts using the APN receive the same QoS level. This could lead to unused or under-utilized bandwidth by some subscribers and thus reducing the amount of resources available to others.

Dynamic QoS Renegotiation

Dynamic QoS Renegotiation minimizes the risk of bandwidth mis-appropriation. This feature allows the GGSN to analyze application traffic, and trigger QoS renegotiation with the SGSN to optimize service performance.

In Dynamic QoS Renegotiation, the GGSN performs packet inspection of application traffic to detect the type of service being utilized and automatically renegotiates the QoS to the appropriate level with a maximum QoS level corresponding to the level granted by the HLR. QoS renegotiation is performed by sending an update PDP context request to the SGSN. This solution is optimal since the appropriate QoS level is always granted to the subscriber without any requirement on the handset or on the core network. The only prerequisite is QoS renegotiation support on the SGSN. In this model, over reservation of radio resources is avoided, while maintaining the appropriate bandwidth for subscribers with real requirements.

The ST16 and ASR 5000 supports L7 stateful analysis and QoS Renegotiation. Combining these functionalities results in Dynamic QoS Renegotiation. The system also generates CDRs (or real time charging information) that includes the current QoS information and the service accessed. This enables intelligent application-based charging of services, taking into account the granted QoS. It also enables rebates when it was not possible to provide the QoS level required by an application.

Important: For L7 traffic analysis ECSv2 license is required.

How Dynamic QoS Renegotiation Works

Implementation of Dynamic QoS Renegotiation involves the following:

- Initial QoS
- Service Detection
- Classification of Application Traffic
- Quality of Service Renegotiation

Initial QoS

When the session is established, an initial level of QoS must be assigned to the subscriber. The GGSN may either grant the requested QoS, or grant a lower QoS level (minimum or intermediate level). The initial QoS remains in effect until the SGSN or GGSN requests a change. When Dynamic QoS Renegotiation is enabled, there are several conditions when the system would request a QoS change.

- Services detected that do not need high QoS: After a configurable time period of a subscriber having terminated services that require high QoS, the system could lower the QoS to a value more appropriate to the services actually being used.

- Services detected that require higher QoS: As soon as a subscriber begins using a service that needs a high QoS, the system immediately attempts to raise the QoS through its service detection capability.

Service Detection

The Application analysis approach to service detection uses application level (L7) information. In the ST16 and ASR 5000, application analysis is stateful—keeping track of the application state.


Important: For L7 traffic analysis ECSv2 license is required.

Classification of Application Traffic

Application traffic can be classified into the following: Conversational, Streaming, Interactive 1, Interactive 2, Interactive 3, or Background. For more information refer to the Traffic Policing and Shaping chapter. Traffic class can be configured in the charging-action, but it does not take direction as a parameter. However, a rule matching only uplink or only downlink packets associated that with the charging-action can be configured.

For QoS renegotiation a way is needed to find out what kind of data packets are flowing through for a particular user to associate a given traffic class with the user's current usage pattern. It can be done through packet inspection as for a subscriber profile, Access Control List (ACL) does the inspection. Limits for each traffic class can be configured in the APN. The same infrastructure is reused to perform Dynamic QoS Renegotiation.

After classification of traffic, if required by subscriber profile, Dynamic QoS Renegotiation takes place.

L4 Packet Inspection

The advantages of L4 packet analysis is no or low impact on the system performance, and enables QoS adaptation with very limited impact on the system capacity. L4 packet inspection is fully supported by the system.

L7 Packet Inspection

The advantages of L7 packet analysis is higher impact on the system performance, and QoS adaptation with very limited impact on the system capacity. L7 packet inspection involves complete application layer analysis and copes with customized applications.

Dynamic QoS Renegotiation

Dynamic QoS Renegotiation minimizes the risk of bandwidth mis-appropriation. This feature allows the GGSN to analyze application traffic, and trigger QoS renegotiation with the SGSN to optimize service performance.

In Dynamic QoS Renegotiation, the GGSN performs packet inspection of application traffic to detect the type of service being utilized and automatically renegotiates the QoS to the appropriate level with a maximum QoS level corresponding

to the level granted by the HLR. QoS renegotiation is performed by sending an update PDP context request to the SGSN. This solution is optimal since the appropriate QoS level is always granted to the subscriber without any requirement on the handset or on the core network. The only prerequisite is QoS renegotiation support on the SGSN. In this model, over reservation of radio resources is avoided, while maintaining the appropriate bandwidth for subscribers with real requirements.

The ST16 and ASR 5000 supports L7 stateful analysis and QoS Renegotiation. Combining these functionalities results in Dynamic QoS Renegotiation. The system also generates CDRs (or real time charging information) that includes the current QoS information and the service accessed. This enables intelligent application-based charging of services, taking into account the granted QoS. It also enables rebates when it was not possible to provide the QoS level required by an application.

Important: For L7 traffic analysis ECSv2 license is required.

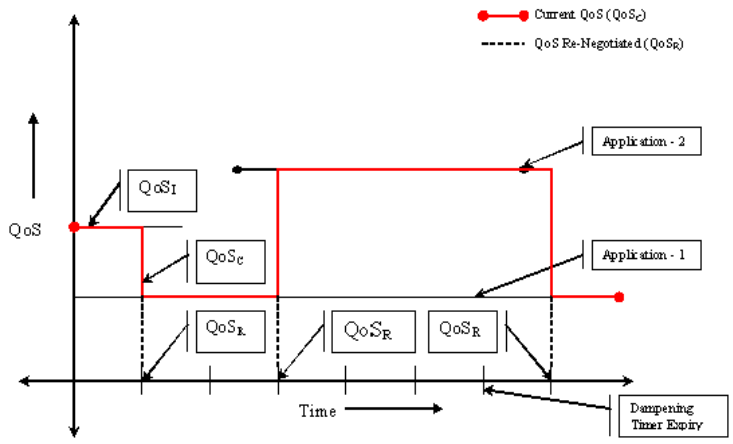
QoS Renegotiation for a Subscriber QoS Profile

The following is the overall Dynamic QoS Renegotiation process.

1. When the subscriber attaches to the network, following things happen:
   - Dampening timer is started for the subscriber.
   - QoSI is assigned to subscriber. This becomes the QoSC till a re-negotiation occurs, as shown in the following figure.
   - The traffic class bit-field is cleared.
2. As the subscriber starts using some applications, the traffic gets classified on the basis of type of data packets or traffic as mentioned in section *Classification of Application Traffic* and accordingly the corresponding bit in Traffic-class-bitfield get set.
3. Following is the mechanics of QoS renegotiation:
   - Examine traffic-class-bitfield to find out the highest bit that is set. This gives the desired QoS Traffic Class (QoSD). The associated uplink/downlink peak-data-rate and guaranteed-data-rate values will be taken from the configured parameters for this traffic class in the subscriber APN. For more information refer to the Traffic Policing and Shaping chapter.
   - If QoSC matches QoSD, no QoS renegotiation is required. Otherwise, it sends an Update PDP Context Request to the SGSN with the QoSD values and QoS renegotiation starts.
   - Reset the dampening timer.
   - The traffic-class-bitfield is cleared.
4. Following are the conditions under which QoS renegotiation happens:
   - When a higher priority traffic is detected, QoS is renegotiated immediately, without waiting the dampening times to expire. Thus for example, if the current traffic has a QoS of interactive class and it detects streaming traffic, it will upgrade the QoS at once to Streaming.
   - When system detects lower priority traffic, it waits for the expiry of the dampening timer before lowering the QoS.
   - During “silence” or no-traffic, QoS renegotiation requests will not be initiated.

   As seen in the following figure, the QoS profile for the subscriber goes through three renegotiations to match the QoS profile of the (highest priority) application currently being used.

5. *Dynamic QoS Renegotiation graph*



When there is no traffic, traffic class drops to “Background”, and the corresponding QoS profile will negotiated as described above.

Network Controlled QoS (NCQoS)

Network-controlled QoS is the method by which the QoS for a PDP context (primary or secondary) is updated on the request of the GGSN through Network Requested Update PDP Context (NRUPC) message. It can also activate a new secondary PDP context on Network Requested Secondary PDP Context Activation (NRSPCA) message from the GGSN.

How Network Controlled QoS (NCQoS) Works

The GGSN activates or modifies a bearer in case of a service flow matching a statically provisioned Policy and Charging Control (PCC) rules. The network, based on QoS requirements of the application/service determines what bearers are needed and either modifies an existing bearer or activates a new one.

Statically provisioned PCC rules, called Network Requested Operation (NRO) rules, are configured as charging rules in Active Charging Service (ACS). As a part of charging action for such rules, QoS-needed and corresponding Traffic Flow Template (TFT) packet filter is configured. QoS-needed mainly consists of QoS Class Identifier (QCI) and data rates. Whereas, TFT mainly consists of uplink and downlink packet filter information.

WARNING: This feature does not work in conjunction with IMS-Authorization service.

When a packet arrives, Active Charging Service (ACS) analyzes it and proceeds for rule matching based on the priority in the rulebase. If an NRO rule bound to the context on which the packet arrived matches, ACS applies the bandwidth limit and gating. If an NRO rule bound to some other context matches, ACS discards the packet.

If an unbound NRO rule matches, ACS finds a context with the same QCI as the NRO rule, where context's Maximum Bit Rate (MBR) and matched rule's MBR (context's MBR + matched rule's MBR) is less than the MBR for that QCI in the APN. If such a context is found, NRUPC for that context is triggered. If the request succeeds, the rule will be bound to that context.

Important: The packet that triggers the NRUPC request is discarded.

If no context satisfying the MBR limit is found, or if there is no context with the same QCI as the NRO rule, the system triggers NRSPCA. If the request succeeds, the rule will be bound to that context.

Important: The packet that triggers the NRSPCA request is discarded.

TFTs from the charging-action associated with the NRO rule are also sent as part of the NRUPC/NRSPCA request, and sent back as part of Create PDP Context response.

Finally, if a non-NRO rule matches, ACS proceeds with the normal processing of that packet. Non-NRO charging-actions can still do "flow action" or ITC (limit-for-flow-type and limit-for-bandwidth).

ACS also takes care of following:

- Before ACS makes an NRUPC/NRSPCA request, it checks if there is any outstanding request for the same QCI for the same subscriber. If there is any, it will not make the new request, and it discards the packet.

- After a context is terminated, ACS unbinds all the rules bound to that context. Such a rule can later be bound to some other context when a packet matches that rule.

Important: The packet that triggers the NRUPC/NRSPCA request is discarded.

Configuring Dynamic QoS Renegotiation

This section describes how to configure per-APN based Dynamic QoS Renegotiation.

Caution: For Dynamic QoS Renegotiation, two RADIUS attributes are required for remote subscriber configuration. For a particular subscriber, these attributes can be overridden without considering the timeout for Dynamic QoS Renegotiation and whether Dynamic QoS Renegotiation is enabled or not.

To configure Dynamic QoS Renegotiation:

Step 1 Configure an Access Control List (ACL), as described in the Configuring ACL for Dynamic QoS Renegotiation section.

Step 2 Configure an APN for Dynamic QoS Renegotiation as described in the Configuring APNs for Dynamic QoS Renegotiation section.

Step 3 Save your configuration as described in the *Verifying and Saving Your Configuration* chapter.

Step 4 Monitor the operations as described in the Monitoring Dynamic QoS Renegotiation Operation 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.

Configuring ACL for Dynamic QoS Renegotiation

Configuring an ACL and applying it to an APN template are required to specify permission and treatment levels for Dynamic QoS Renegotiation.

Use the following example to configure an ACL for Dynamic QoS Renegotiation:

```
configure

   context <context_name>

      ip access-list <acl_name>

         permit { tcp | udp } ........ treatment { background |
conversational | interactive-1 | interactive-2 | interactive-3 |
streaming }

         end
```

Notes:

- *<context_name>* must be the name of the destination context in which you want to configure the ACL. The same context must be used for APN configuration.

- For information on configuring the rules that comprise the ACL, in the IP Access Control Lists chapter, see the Configuring ACLs on the System section.

Configuring Charging Action for Dynamic QoS Renegotiation

Use the following example to configure charging action parameters for Dynamic QoS Renegotiation support:

```
configure

   active-charging service <service_name>

      charging-action <charging_action_name> [ -noconfirm ]

         qos-renegotiate traffic-class { background | conversational |
interactive <priority> | streaming }

         flow action discard [ downlink | uplink ]

            flow limit-for-bandwidth direction { downlink | uplink }
peak-data-rate <bps> peak-burst-size <bytes> violate-action { discard |
lower-ip-precedence } [ committed-data-rate <bps> committed-burst-size
<bytes> [ exceed-action { discard | lower-ip-precedence } ] ]

            end
```

Notes:

- A maximum of eight packet filters can be configured per charging action.

Configuring Rulebase for Dynamic QoS Renegotiation

Use the following example to configure rulebase parameters for Dynamic QoS Renegotiation support:

```
configure

   active-charging service <service_name>

      rulebase <rulebase_name> [ -noconfirm ]

         qos-renegotiate timeout <timeout>

         end
```

Configuring APNs for Dynamic QoS Renegotiation

Use the following example to configure an APN template's QoS profile in support of Dynamic QoS Renegotiation:

```
configure
   context <context_name>
      apn <apn_name>
         ip access-group <acl_name> [ in | out ]
         end
```

Notes:

- *<context_name>* must be the name of the destination context in which you have already configured the ACL, and want to configure the APN template.
- *<acl_name>* must be the name of the ACL that you have already configured in the context.
- If in the **ip access-group** command of the APN Configuration Mode, the optional **in** or **out** keywords are not specified, the ACL will be applied to all packets, in and out.

Configuring Network Controlled QoS (NCQoS)

To configure NCQoS:

Step 1 Configure packet filter parameters as described in the Configuring Packet Filter for NCQoS section.

Step 2 Configure charging rules and actions as described in the Configuring Charging Action for NCQoS section.

Step 3 Configure APN template and enable bearer control mode for NCQoS as described in the Configuring APN for NCQoS section.

Step 4 Save your configuration as described in the *Verifying and Saving Your Configuration* chapter.

Step 5 Monitor the operations as described in the Monitoring Dynamic QoS Renegotiation Operation 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.

Configuring Packet Filter for NCQoS

Use the following example to configure packet filter parameters for NCQoS support:

```
configure

   active-charging service <service_name>

      packet-filter <filter_name> [ -noconfirm ]

         ip local-port { = <port_num> | range <start_port_num> to <end_port_num>
}

         ip protocol { = <proto_num> | range <start_proto_num> to
<end_proto_num> }

         ip remote-address { = { <ip_address> | <ip_address/mask> } | { range {
<ip_address> | <ip_address/mask> } to { <ip_address> | <ip_address/mask> } }

         ip remote-port { = <port_num> | range <start_port_num> to
<end_port_num> }

         direction { bi-directional | download | upload }

         priority <priority>

         end
```

Configuring Charging Action for NCQoS

Use the following example to configure charging action parameters for NCQoS support:

```
configure
   active-charging service <service_name>
      charging-action <charging_action_name> [ -noconfirm ]
         qos-class-identifier <identifier>
         flow action discard [ downlink | uplink ]
         tft packet-filter <filter_name>
         flow limit-for-bandwidth direction { downlink | uplink } peak-data-rate
<bps> peak-burst-size <bytes> violate-action { discard | lower-ip-precedence }
         end
```

Notes:

- A number of optional keywords and variable are available for the **`flow limit-for-bandwidth direction`** command. Refer to the *Command Line Interface Reference* for more information regarding this command.

Configuring APN for NCQoS

Use the following example to enable Bearer Control Mode (BCM) for NCQoS support:

```
configure
   context <context_name>
      apn <apn_name>
         bearer-control-mode [ mixed | ms-only | none ]
         end
```

Notes:

- To enable NCQoS, bearer-control-mode in the APN Configuration Mode must be configured with **`mixed`** mode.

Monitoring Dynamic QoS Renegotiation Operation

Use the following steps to verify/monitor Dynamic QoS Renegotiation operations:

Step 1 Verify that your APNs were configured properly by entering the following command:

```
show apn { all | name <apn_name> }
```

The output is a listing of APN parameter settings.

Step 2 Verify that the ACLs have been properly applied by entering the following command:

```
show apn name <apn_name>
```

<apn_name> must be the name of the APN configured in the *Configuring APNs for Dynamic QoS Renegotiation* section.

The output of this command displays the APN's configuration. Examine the output for the ip output access-group and ip input access-group fields. For more details refer to the *Applying a Single ACL to Multiple Subscribers* section in this guide.

Step 3 Verify that your ACL was configured properly by entering the following command:

```
show ip access-list <acl_name>
```

The output is a concise listing of IP Access Control List parameter settings.

Step 4 Monitor your QoS renegotiation status for a subscriber by entering the following command:

```
show subscriber ggsn-only full
```

The output is a concise listing of subscribers' settings.

Step 5 For L7 based QoS Renegotiation, view how many time QoS renegotiations have happened for that session by entering the following command:

```
show active-charging sessions full all
```

Step 6 View the statistics of APN related to QoS renegotiation parameters by entering the following command:

```
show apn statistics [ all | name <apn_name> ]
```

The output is a listing of APN statistics related to QoS Renegotiation.

Event IDs Pertaining to Dynamic QoS Renegotiation

The Session Manager facility provides several events that can be useful for diagnosing errors that could occur with implementation of Dynamic QoS Renegotiation feature.

The following table displays information pertaining to these events.

Table 44. Event IDs in Session Manager Pertaining to Dynamic QoS Renegotiation

Event	Event ID	Type	Additional Information
QoS Renegotiation timer started for subscriber	10917	Info	"Indicates that the Dynamic QoS Renegotiation timer was started for the subscriber"
QoS Renegotiation timer stopped for subscriber	10918	Info	"Indicates that the Dynamic QoS Renegotiation timer was stopped for the subscriber"
QoS Renegotiation timer expired for subscriber	10919	Info	"Indicates that the Dynamic QoS Renegotiation timer was expired for the subscriber"
QoS Renegotiation message sent for subscriber	10920	Info	"Indicates that the Dynamic QoS Renegotiation message was sent for the subscriber"
L4 classification done for subscriber traffic	10921	Info	"Indicates the kind of L4 classification that was done for the subscriber traffic."

RADIUS Attributes

The RADIUS attributes listed in the following table are used to configure Dynamic QoS Renegotiation for subscribers configured on remote RADIUS servers. More information on these attributes can be found in the *AAA Interface Administration and Reference*.

Table 45. RADIUS Attributes Required for Dynamic QoS Renegotiation Support

Attribute	Description
SN-Enable-QoS-Renegotiation (or SN1-Enable-QoS-Renegotiation)	Enables the Dynamic QoS Renegotiation for specific profile application. This attribute displays "enable qos renegotiation".
SN-QoS-Renegotiation-Timeout (or SN1-QoS-Renegotiation-Timeout)	Timeout duration for dampening time for QoS renegotiation to specific profile application. This attribute displays "qos renegotiation timeout".

Chapter 34
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 sctions are included in this chapter:

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.
redirect *<ip_addr1>* [**weight** *<value>*] [*<ip_addr2>* [**weight** *<value>*]] ... [*<ip_addr16>* [**weight** *<value>*]]	Upon network reachability failure redirect all calls to the specified IP address. *<ip_addr>*: This must be an IPv4 address. Up to 16 IP addresses and optional weight values can be entered on one command line. **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. *<value>* must be an integer from 1 through 10.

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
server_name	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.

Keyword/Variable	Description
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>*

<pool_name>	The name of an existing IP pool in the current context.
nw-reachability server *<server_name>*	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. *<server_name>*: 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 as described in *Verifying and Saving Your Configuration*.

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-vsa1
- starent-vsa1-835

Refer to the *AAA Interface Administration and Reference* for more details.

Chapter 35
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

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.

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 <ip_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 <ip_address> netmask <netmask>
      address <ip_address> netmask <netmask>
      address <ip_address> netmask <netmask>
      end
```

Notes:

- Save your configuration as described in the *Verifying and Saving Your Configuration* chapter.

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, see 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
configure
   context <context_name>
```

```
radius accounting ip remote-address collection
end
```

Notes:

- Save your configuration as described in *Verifying and Saving Your Configuration* chapter.

Chapter 36
Routing

This chapter provides information on configuring an enhanced, or extended, service. The product administration guides provide examples and procedures for configuration 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.

Routing Policies

This section describes how to configure the elements you need to specify routing policies. Routing policies modify and redirect routes to and from the system to satisfy specific routing needs.

Use the following building blocks to configure routing policies:

- **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 provide detailed control over routes during route selection or route advertisement by a routing protocol, and in route redistribution between routing protocols. For this level of control you use IP Prefix Lists, Route Access Lists and AS Path Access Lists to specify IP addresses, address ranges, and Autonomous System Paths.

Creating IP Prefix Lists

Use the following configuration example to create IP Prefix Lists:

```
config
  context <context_name>
    ip prefix-list name <list_name> { deny | permit }
<network_address/net_mask>
```

Notes:

- Set the IP prefix list to deny, permit or match any prefix.
- IPv4 and IPv6 addresses are supported.
- Save your configuration as described in Saving your Configuration.

Creating Route Access Lists

Use the following procedure to create a Route Access List:

```
config
   context <context_name>
      route-access-list { extended identifier } { deny | permit } [ip address
] <ip_address>
```

```
route-access-list named <list_name> { deny | permit } {
<ip_address/mask> | any } [ exact-match ]

route-access-list standard identifier { permit | deny } {<ip_address>
<wildcard_mask> | any | host <network_address> }
```

Notes:

- A maximum of 64 access lists are supported per context.
- Save your configuration as described in Verifying and Saving Your Configuration.

Creating AS Path Access Lists

Use the following procedure to create an AS Path Access List:

```
config

   context     <context_name>

ip as-path access-list <list_name> [ { deny | permit } <reg_expr> ]
```

Notes:

- Save your configuration as described in Verifying and Saving Your Configuration.

Creating Route Maps

Use the following configuration example to create a Route Map:

```
  config

     context <context_name>

        route-map< map_name > { deny | permit } < seq_number >
```

Notes:

- Use the match and set commands in Route Map Configuration mode to configure the route map. Refer to the ASR 5000 Series Command Line Interface Reference for more information on these commands.
- Save your configuration as described in Verifying and Saving Your Configuration.

Sample Configuration

The example below shows a configuration that creates two route access lists, applies them to a route map, and uses that route map for a BGP router neighbor.

```
config
   context isp1
      route-access-list named RACLin1a permit 88.151.1.0/30
      route-access-list named RACLin1a permit 88.151.1.4/30
      route-access-list named RACLany permit any
      route-map RMnet1 deny 100
         match ip address route-access-list RACLin 1 a
         #exit
         route-map RMnet1 deny 200
         match ip address route-access-list RACLin 1 b
         #exit
      route-map RMnet1 permit 1000
         match ip address route-access-list RACLany
         #exit
      router bgp 1
         neighbor 152.20.1.99 as-path 101
         neighbor 152.20.1.99 route-map RMnet1
```

Static Routing

The system supports static network route configuration on a per context basis. Define network routes by specifying an IP address and mask for the route, the name of the interface in the currant context that the route must use, and a next hop IP address.

Adding Static Routes to a Context

To add static routes to a context configuration, you must know the names of the interfaces that are configured in the current context. Use the following command to list the interfaces in the current context:

```
show ip interface
```

Information for all interfaces configured in the current context is displayed as shown in the following example.

```
[ local ]< host_name > #show ip interface

Intf Name: Egress 1

Description:

IP State: Up (Bound to 24/1 untagged ifIndex 402718721)

IP Address: 192.168.231.5

Subnet Mask: 255.255.255.0

Bcast Address: 192.168.231.255

MTU: 1500

Resoln Type: ARP ARP timeout: 3600 secs

L3 monitor LC-port switchover: Disabled

Number of Secondary Addresses: 0

Total interface count: 1
```

The first line of information for each interface lists the interface name for the current context as shown in the example output. In this case, there is one interface with the name Egress 1.

config

context *<context_name>*

ip route { < *ip_address* | *ip_mask* > | < *ip_addr_mask_combo* > } { **next-hop** } < *next_hop_address* > | < *egress_name* > [**precedence**] < *precedence* > [**cost**] < *cost* >

Notes:

You can configure a maximum of 1200 static routes per context. Save your configuration as described in Verifying and Saving Your Configuration.

Deleting Static Routes From a Context

Use the following configuration example to remove static routes from a contexts configuration:

config

context *context_name*

no ip route { < *ip_address* > < *ip_mask* > | < *ip_addr_mask_combo* > } < *next_hop_address* > < *egress_name* > [**precedence** < *precedence* >] [**cost** < *cost* >]

Notes:

- Save your configuration as described in Verifying and Saving Your Configuration.

OSPF Routing

This section gives an overview of OSPF (Open Shortest Path First) routing and its implementation in the system. It also provides the procedure for enabling the base OSPF functionality, and lists the commands that are available for more complex configuration.

OSPF routing is included with the IPV4 Routing Protocols feature. You must purchase and install a license key before you can use this feature.

Important: During system task recovery, it is possible for a dynamically-learned forwarding entry to incorrectly remain in the system forwarding table if that forwarding entry has been removed from the dynamic routing protocol during the recovery.

OSPF Version 2 Overview

OSPF is a link-state routing protocol, an interior gateway protocol (IGP) that routes IP packets using the shortest path first based solely on the destination IP address in the IP packet header. IP packets are routed are not encapsulated in any further protocol headers as they transit the network. An Autonomous System (AS), or Domain, is defined as a group of networks within a common routing infrastructure.

OSPF is a dynamic routing protocol that quickly detects topological changes in the AS (such as router interface failures) and calculates new loop-free routes after a period of convergence. This period of convergence is short and involves a minimum of routing traffic.

In a link-state routing protocol, each router maintains a database, referred to as the link-state database, that describes the Autonomous System's topology.Each participating router has an identical database. Each individual piece of this database is a particular router's local state (for example, the router's usable interfaces and reachable neighbors). The router distributes its local state throughout the Autonomous System by flooding.

All routers run the same algorithm in parallel. From the link-state database, each router constructs a tree of shortest paths with itself as root to each destination in the Autonomous System. Externally derived routing information appears on the tree as leaves. The cost of a route is described by a single dimensionless metric.

OSPF allows sets of networks to be grouped together. Such a grouping is called an area. The topology of this area is hidden from the rest of the AS, which enables a significant reduction in routing traffic. Also, routing within the area is determined only by the area's own topology, lending the area protection from bad routing data. An area is a generalization of an IP subnetted network.

OSPF enables the flexible configuration of IP subnets so that each route distributed by OSPF has a destination and mask. Two different subnets of the same IP network number may have different sizes (that is, different masks). This is commonly referred to as variable-length subnetting. A packet is routed to the best (longest or most specific) match. Host routes are considered to be subnets whose masks are "all ones" (0xffffffff).

OSPF traffic can be authenticated or non-authenticated, or can use no authentication, simple/clear text passwords, or MD5-based passwords. This means that only trusted routers can participate in the Autonomous System's routing. You can specify a variety of authentication schemes and, in fact, you can configure separate authentication schemes for each IP subnet.

Externally derived routing data (for example, routes learned from an exterior protocol such as BGP) is advertised throughout the AS. This externally derived data is kept separate from the OSPF protocol'

s link state data. Each external route can also be tagged by the advertising router, enabling the passing of additional information between routers on the boundary of the AS.

Link-State Algorithm

OSPF uses a link-state algorithm in order to build and calculate the shortest path to all known destinations. The algorithm by itself is quite complicated. The following is a very high level, simplified way of looking at the various steps of the algorithm:

1. Upon initialization or update in routing information, an OSPF-enabled router generates a link-state advertisement (LSA). This LSA represents the collection of all link-states on that router.
2. All routers exchange link-states by means of flooding. Each router that receives a link-state update stores a copy in its link-state database and then propagates the update to other routers.
3. After the database of each router is completed, the OSPF-enabled router calculates a Shortest Path Tree to all destinations. The router uses the Dijkstra algorithm to calculate the shortest path tree. The algorithm places each router at the root of a tree and calculates the shortest path to each destination based on the cumulative cost required to reach that destination. Each router has its own view of the topology even though all OSPF-enabled routers build a shortest path tree using the same link-state database. The destinations, associated cost, and the next hop to reach those destinations form the IP routing table.
4. If no changes in the OSPF network occur, such as link cost or an added or deleted network, OSPF is quiet. Any changes that occur are communicated via link-state update packets, and the Dijkstra algorithm is recalculated to again find the shortest path.

Basic OSPFv2 Configuration

This section describes how to implement basic OSPF routing functionality.

Enabling OSPF Routing For a Specific Context

Use the following configuration example to enable OSPF Routing for a specific context:

```
config
   context <context_name>
      router ospf
      end
```

Notes:

- Save your configuration as described in Verifying and Saving Your Configuration.

Enabling OSPF Over a Specific Interface

After you enable OSPF, specify the networks on which it will run. Use the following command to enable OSPF:

network *< network_ip_address >* */* *< network_mask >* **area** {*< area_id >* | *< area_ip_address >* }

Important: The default cost for OSPF on the system is 10. To change the cost, refer to the **ip ospf cost** command in the Ethernet Interface Configuration mode. For detailed information on this command refer to the Cisco ASR 5000 Series Command Line Interface Reference.

Notes:

- Save your configuration as described in Verifying and Saving Your Configuration.

Redistributing Routes Into OSPF (Optional)

Redistributing routes into OSPF means any routes from another protocol that meet specified a specified criterion, such as route type, metric, or rule within a route-map, are redistributed using the OSPFv2 protocol to all OSPF areas. This is an optional configuration.

```
config
   context < context_name >
      router ospf
         redistribute { connected | rip | static }
         end
```

Notes:

- Save your configuration as described in Verifying and Saving Your Configuration.

Confirming OSPF Configuration Parameters

To confirm the OSPF router configuration, use the following command and look for the section labeled router ospf in the screen output:

show config context *< ctxt_name >* [**verbose**]

Viewing Routing Information

To view routing information for the current context, at the Executive mode level, use one of the following commands;

- show ip route: Display information for all types of routes in the current contexts routing table.
- show ip static-route: Display information only for static routes in the current contexts routing table.
- show ip ospf: Display OSPF process summary information in the current context.

This example shows sample output of the command, **show ip route**.

```
[local]host_name# show ip route

"*" indicates the Best or Used route. Destination Nexthop Protocol Prec
Cost Interface

*44.44.44.0/24 208.230.231.50 static 1 0 local1

*192.168.82.0/24 0.0.0.0 connected 0 0

*192.168.83.0/24 0.0.0.0 connected 0 0

 208.230.231.0/24 0.0.0.0 ospf 110 10 local1

*208.230.231.0/24 0.0.0.0 connected 0 0 local1

Total route count: 5
```

Equal Cost Multiple Path (ECMP)

The system supports ECMP for routing protocols. ECMP distributes traffic across multiple routes that have the same cost to lessen the burden on any one route.

```
config
   context < context_name >
      ip routing maximum-paths [ max_no ]
```

Notes:

- Save your configuration as described in Verifying and Saving Your Configuration.

BGP-4 Routing

The Border Gateway Protocol 4 (BGP-4) routing protocol is supported through a BGP router process that is implemented at the context level.

The Border Gateway Protocol (BGP) is an inter-Autonomous System routing protocol. An Autonomous System (AS) is a set of routers under a single technical administration that use an interior gateway protocol and common metrics to route packets within the AS. The set of routers uses an exterior gateway protocol to route packets to other ASs.

BGP runs over TCP. This eliminates the need for the BGP protocol to implement explicit update fragmentation, retransmission, acknowledgement, and sequencing information. Any authentication scheme used by TCP may be used in addition to BGP's own authentication mechanisms.

BGP routers exchange network reachability information with other BGP routers. This information builds a picture of AS connectivity from which routes are filtered and AS level policy decisions are enforced.

BGP-4 provides classless inter-domain routing. This includes support for advertising an IP prefix and eliminates the concept of network class within BGP. BGP-4 also allows the aggregation of routes, including the aggregation of AS paths.

Overview of BGP Support

When using Mobile IP, mobile devices communicate to the Internet through Home Agents (HAs). HAs assign IP addresses to the mobile node from a configured pool of addresses. These addresses are also advertised to Internet routers through an IP routing protocol to ensure dynamic routing. The BGP-4 protocol is used as a monitoring mechanism between an HA and Internet router with routing to support Interchassis Session Recovery. (Refer to the Interchassis Session Recovery chapter in this manual for more information.)

The objective of BGP-4 protocol support is to satisfy routing requirements and to monitor communications with Internet routers. BGP-4 may trigger an active to standby switchover to keep subscriber services from being interrupted.

The following BGP-4 features are supported:

- Exterior Border Gateway Protocol (EBGP) multi-hop
- Route Filtering for inbound and outbound routes
- Route redistribution and route-maps

IP pool routes and loopback routes are advertised in the BGP domain in the following ways:

- Through BGP configuration mode redistribution commands, all or some of the connected routes are redistributed into the BGP domain. (IP pool and loopback routes are present in the IP routing table as connected routes.) The routemap command provides the flexibility to change many BGP attributes.
- Through the BGP configuration mode network commands, connected routes are explicitly configured for advertisement into the BGP domain. The network routemap command provides the flexibility to change many BGP attributes. Refer to the Cisco Systems ASR 5000 Command Line Interface Reference for details on the BGP configuration mode commands.

If a BGP task restarts because of a processing card failure, a migration, a crash, or the removal of a processing card, all peering session and route information is lost.

Configuring BGP

This section describes how to configure and enable basic BGP routing support in the system.

```
config
   context <context_name>
      router { ospf | bgp < as_number >
         neighbor < IP_address > { remote-as < AS_num > }
```

Notes:

- A maximum of 64 BGP peers are supported per context.
- Save your configuration as described in Verifying and Saving Your Configuration.

Redistributing Routes Into BGP (Optional)

Redistributing routes into BGP simply means that any routes from another protocol that meet a specified criterion, such as a route type, or a rule within a route-map, are redistributed through the BGP protocol to all BGP areas. This is an optional configuration.

```
config
   context <context_name>
      router{ ospf | bgp < as_number > }
         redistribute{bgp | connected | static } [ metric ] <
metric_value > ] [ metric-type ] {1 | 2 } ] [ route-map ] <
route_map_name ]
```

Notes:

- The redistribution options are connected, ospf, rip, or static.
- A maximum of 64 route-maps are supported per context.
- Save your configuration as described in Verifying and Saving Your Configuration.

Chapter 37
Session Recovery

In the telecommunications industry, over 90 percent of all equipment failures are software-related. With robust hardware failover and redundancy protection, any card-level hardware failures on the system can quickly be corrected. However, software failures can occur for numerous reasons, many times without prior indication. For this reason, we have introduced a new solution to recover subscriber sessions in the event of failure.

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.

This feature is available for the following functions:

- PDSN services supporting simple IP, Mobile IP, and Proxy Mobile IP
- GGSN services for IPv4 and PPP PDP contexts
- SGSN services (3G and 2.5G services) for IPv4 and PPP PDP contexts
- HA services supporting Mobile IP and/or Proxy Mobile IP session types with or without per-user Layer 3 tunnels
- LNS session types
- IPSG-only systems
- Any session needing L2TP LAC support (excluding regenerated PPP on top of an HA/GGSN session)

Session recovery is **not supported** for the following functions:

- Mobile IP sessions with L2TP
- Multiple MIP sessions
- Destination-based accounting recovery
- Any session using IPv6 (PDSN/GGSN/SGSN/LNS)
- Any session needing L2TP LAC support (including regenerated PPP on top of an HA/GGSN session)
- GGSN network initiated connections
- MIP session with multiple concurrent bindings
- MIP/L2TP with IPSEC integration
- GGSN session using more than 1 service instance

Important: Session Recovery can only be enabled through a feature use license key. If you have not previously purchased this enhanced feature, contact your sales representative for more information.

When session recovery occurs, the system reconstructs the following subscriber information:

- Data and control state information required to maintain correct call behavior.
- A minimal set of subscriber data statistics; required to ensure that accounting information is maintained.
- A best-effort attempt to recover various timer values such as call duration, absolute time, and others.

- The idle time timer is reset to zero and the re-registration timer is reset to its maximum value for HA sessions to provide a more conservative approach to session recovery.

Session Recovery is also useful for Software Patch Upgrade activities. If session recovery feature is enabled during the software patch upgrading, it helps to permit preservation of existing sessions on the active PAC/PSC/PSC2 during the upgrade process. For more details refer to *Software Patch Upgrade* in the *System Administration Guide*.

Important: Any partially connected calls (e.g., a session where HA authentication was pending but has not yet been acknowledged by the AAA server) are not recovered when a failure occurs.

How Session Recovery Works

This section provides an overview of how this feature is implemented and the recovery process.

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. Naturally, these mirrored processes require both memory and processing resources, which means that additional hardware may be required to enable this feature (see the *Additional Hardware Requirements* section).

Additionally, other key system-level software tasks, such as VPN manager, are performed on a physically separate Packet Accelerator Card (PAC)/Packet Services Card (PSC/PSC2) to ensure that a double software fault (e.g., session manager and VPN manager fails at same time on same card) cannot occur. The PAC/PSC/PSC2 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.

There are two modes of 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 PAC/PSC/PSC2. In this mode, recovery is performed by using the mirrored "standby-mode" session manager task(s) running on active PACs/PSCs/PSC2s. The "standby-mode" task is renamed, made active, and is then populated using information from other tasks such as AAA manager. In case of Task failure, limited subscribers will be affected and will suffer outage only until the task starts back up.
- **Full PAC/PSC/PSC2 recovery mode**: Used when a PAC/PSC/PSC2 hardware failure occurs, or when a planned PAC/PSC/PSC2 migration fails. In this mode, the standby PAC/PSC/PSC2 is made active and the "standby-mode" session manager and AAA manager tasks on the newly activated PAC/PSC/PSC2 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 application cards to ensure task recovery.

There are some situations wherein session recovery may not operate properly. These include:

- Additional software or hardware failures during the session recovery operation. An example of this would be if an AAA manager were to fail while the state information it contained was being used to populate the newly activated session manager task.
- A lack of hardware resources (i.e., PAC/PSC/PSC2 memory and control processors) to support session recovery.

Important: After a session recovery operation, some statistics, such as those collected and maintained on a per manager basis (AAA Manager, Session Manager, etc.) are in general not recovered, only accounting/billing related information is checkpointed/recovered.

Additional Hardware Requirements

Because session recovery requires numerous hardware resources, such as memory, control processors, NPU processing capacity, etc., some additional hardware may be required to ensure that enough resources are available to fully support this feature.

Important: A minimum of four PACs/PSCs/PSC2s (three active and one standby) per individual chassis is required to use this feature.

To allow for complete session recovery in the event of a hardware failure during a PAC/PSC migration, a minimum of three active PACs/PSCs/PSC2s and two standby PACs/PSCs/PSC2s should be deployed.

To assist you in your network design and capacity planning, the following list provides information that should be considered.

- Subscriber capacity is decreased depending on the hardware configuration. A fully configured chassis (12 active PACs/PSCs/PSC2s and 2 standby PACs/PSCs/PSC2s) would experience a smaller decrease in subscriber capacity versus a minimally configured chassis (3 active PACs/PSCs/PSC2s and 2 standby PAC/PSCs/PSC2s).
- The amount by which control transaction processing capacity is reduced.
- The reduction in subscriber data throughput.
- The recovery time for a failed software task (e.g., session manager).
- The recovery time for a failed PAC/PSC/PSC2 (hardware failure).

If a PAC/PSC/PSC2 migration is being performed, this may temporarily impact the ability to perform session recovery as hardware resources (e.g., memory, processors, etc.) that may be needed are not available during this operation. To avoid this condition, a minimum of two standby PACs/PSCs/PSC2s should be configured.

Configuring the System to Support Session Recovery

The following configuration procedures allow you to configure the session recovery feature for either an operational system that is currently in-service (able to accept incoming calls) or a system that is out-of-service (not part of your production network and therefore not processing any live subscriber/customer data).

Important: Session recovery can only be enabled through a feature use license key. If you have not previously purchased this enhanced feature, contact your sales representative for more information.

The session recovery feature, even when the feature use key is present, is disabled by default on the system.

Enabling Session Recovery

As noted earlier, session recovery can be enabled on a system that is out-of-service (OoS) and does not yet have any contexts configured, or on an in-service system that is currently capable of processing calls. However, if the system is in-service, it must be restarted before the session recovery feature takes effect. Each procedure is shown below.

Enabling Session Recovery on an Out-of-Service System

The following procedure is for a system that does not have any contexts configured.

To enable the session recovery feature on an out-of-service (OoS) system, follow the procedure below. This procedure assumes that you begin at the Exec mode prompt.

Step 1 At the Exec mode prompt, verify that the session recovery feature is enabled by the session and feature use license on the system by entering the following command:

```
show license info
```

The output of this command appears similar to the example shown below. Note that the session recovery feature is bold-faced in this example.

```
Key Information (installed key):
   Comment                 <Host Name>
   CF Device 1             Model: "SanDiskSDCFB-512"
                           Serial Number: "115212D1904T0314"
   CF Device 2             Model: "SanDiskSDCFB-512"
                           Serial Number: "115206D1904S5951"
```

```
   Date of Issue          Thursday May 12 14:35:50 EDT 2005
   Issued By              <Vendor Name>
   Key Number             17120
Enabled Features:
   Part Number  Quantity  Feature
   -----------  --------  ----------------------
   xxx-xx-xxxx        15  PDSN/GGSN/SGSN (10K)
        [none]         -  FA
        [none]         -  IPv4 Routing Protocols
   xxx-xx-xxxx         -  IPSec
   xxx-xx-xxxx         -  2TP LAC (PDSN/GGSN/SGSN)
   xxx-xx-xxxx         1  L2TP LNS (10K)
   xxx-xx-xxxx         6  L2TP LNS (1K)
   xxx-xx-xxxx         -  Session Recovery (PDIF/PDSN/GGSN/SGSN)
        [none]          - Session Recovery (HA)
   xxx-xx-xxxx         -  PCF Monitoring
   xxx-xx-xxxx         -  Layer 2 Traffic Management
 Session Limits:
               Sessions  Session Type
               --------  ----------------------
                 150000  PDSN/GGSN/SGSN
 Status:
                  16000  L2TP LNS
   CF Device 1           Does not match either SPC
   CF Device 2           Does not match either SPC
   License Status        Good (Not Redundant)
```

Important: If the Session Recovery feature appears as Disabled, then you cannot enable this feature until a new license key is installed in the system.

Step 2 Use the following configuration example to enable session recovery.

```
configure
   require session recovery
   end
```

Step 3 Save your configuration as described in the *Saving Your Configuration* section in the *System Administration Guide*.

The system, when started, enables session recovery, creates all mirrored "standby-mode" tasks, and performs PAC/PSC/PSC2 reservations and other operations automatically.

Step 4 After the system has been configured and placed in-service, you should verify the preparedness of the system to support this feature as described in *Viewing Session Recovery Status.*

Enabling Session Recovery on an In-Service System

When enabling session recovery on a system that already has a saved configuration, the session recovery commands are automatically placed before any service configuration commands in the configuration file.

To enable the session recovery feature on an in-service system, follow the procedure below. This procedure assumes that you begin at the Exec mode prompt.

Step 1 At the Exec mode prompt, verify that the session recovery feature is enabled by the session and feature use license on the system by entering the following command:

```
show license info
```

The output of this command appears similar to the example shown below. Note that the session recovery feature is bold-faced in this example.

```
Key Information (installed key):
   Comment                <Host Name>
   CF Device 1            Model: "SanDiskSDCFB-512"
                          Serial Number: "115212D1904T0314"
   CF Device 2            Model: "SanDiskSDCFB-512"
                          Serial Number: "115206D1904S5951"
   Date of Issue          Thursday May 12 14:35:50 EDT 2005
   Issued By              <Vendor Name>
   Key Number             17120
Enabled Features:
```

```
  Part Number  Quantity  Feature
  -----------  --------  ------------------------
  xxx-xx-xxxx        15  PDSN/GGSN/SGSN (10K)
       [none]        -   FA
       [none]        -   IPv4 Routing Protocols
  xxx-xx-xxxx        -   IPSec
  xxx-xx-xxxx        -   2TP LAC (PDSN/GGSN/SGSN)
  xxx-xx-xxxx        1   L2TP LNS (10K)
  xxx-xx-xxxx        6   L2TP LNS (1K)
  xxx-xx-xxxx        -   Session Recovery (PDIF/PDSN/GGSN/SGSN)
       [none]         - Session Recovery (HA)
  xxx-xx-xxxx        -   PCF Monitoring
  xxx-xx-xxxx        -   Layer 2 Traffic Management
Session Limits:
               Sessions  Session Type
               --------  ----------------------
                 150000  PDSN/GGSN/SGSN
Status:
                  16000  L2TP LNS
  CF Device 1            Does not match either SPC
  CF Device 2            Does not match either SPC
  License Status         Good (Not Redundant)
```

Important: If the Session Recovery feature for HA appears as Disabled, then you cannot enable this feature until a new license key is installed in the system.

Step 2 Use the following configuration example to enable session recovery.

```
configure
   require session recovery
   end
```

Important: This feature does not take effect until after the system has been restarted.

Step 3 Save your configuration as described in *Saving Your Configuration.*

Step 4 Perform a system restart by entering the following command:

```
reload
```

The following prompt appears:

```
Are you sure? [Yes|No]:
```

Confirm your desire to perform a system restart by entering the following:

```
yes
```

The system, when restarted, enables session recovery and creates all mirrored "standby-mode" tasks, performs PAC/PSC/PSC2 reservations, and other operations automatically.

Step 5 After the system has been restarted, you should verify the preparedness of the system to support this feature as described in the *Viewing Session Recovery Status* section.

Important: More advanced users may opt to simply insert the **`require session recovery`** command syntax into an existing configuration file using a text editor or other means, and then applying the configuration file manually. Caution should be taken when doing this to ensure that this command is placed among the first few lines of any existing configuration file to ensure that it appears before the creation of any non-local context.

Disabling the Session Recovery Feature

To disable the session recovery feature on a system, enter the following command from the Global Configuration mode prompt:

```
no require session recovery
```

Important: If this command is issued on an in-service system, then the system must be restarted by issuing the **`reload`** command.

Viewing Session Recovery Status

To determine if the system is capable of performing session recovery, when enabled, enter the following command from the Exec mode prompt.

show session recovery status [**verbose**]

The output of this command should be similar to the examples shown below.

```
[local]host_name# show session recovery status
Session Recovery Status:
   Overall Status         : SESSMGR Not Ready For Recovery
   Last Status Update     : 1 second ago

[local]host_name# show session recovery status
Session Recovery Status:
  Overall Status         : Ready For Recovery
  Last Status Update     : 8 seconds ago

[local]host_name# show session recovery status verbose
Session Recovery Status:
  Overall Status         : Ready For Recovery
  Last Status Update     : 2 seconds ago

              ----sessmgr---     ----aaamgr----     demux
 cpu state    active  standby    active  standby    active  status
---- -------  ------  -------    ------  -------    ------  ------------
 1/1 Active   2       1          1       1          0       Good
 1/2 Active   1       1          0       0          0       Good
 1/3 Active   1       1          3       1          0       Good
 2/1 Active   1       1          1       1          0       Good
 2/2 Active   1       1          0       0          0       Good
 2/3 Active   2       1          3       1          0       Good
 3/0 Active   0       0          0       0          1       Good (Demux)
 3/2 Active   0       0          0       0          1       Good (Demux)
```

```
4/1 Standby  0       2          0       1          0       Good
4/2 Standby  0       1          0       0          0       Good
4/3 Standby  0       2          0       3          0       Good


[local]host_name#
```

Viewing Recovered Session Information

Per subscriber session information is available to show any changes in session recovery status. A new field has been added to the show subscriber debug-info command that is named “Redundancy Status”. This field shows whether or not the session has been recovered or is the original information. There are two valid outputs for this field:

- **Original** - indicating that this is the original session information, containing all event states and time information.
- **Recreated Session** - indicating that this session was reconstructed during a session recovery operation.

This command can be executed before or after a session recovery operation has been performed, and would show information relative to the specific session.

To view session state information and any session recovery status, enter the following command:

show subscriber debug-info {**callid** | **msid** | **username**}

Keyword/Variable	Description
callid *id*	Displays subscriber information for the call specified by *id*. The call ID must be specified as an 8-byte hexadecimal number.
msid *id*	Displays information for the mobile user identified by *id*. *id* must be from 7 to 16 digits specified as an IMSI, MIN, or RMI. Wildcard characters $ and * are allowed. The * wildcard matches multiple characters and the $ wildcard matches a single character. If you do not want the wildcard characters interpreted as a wildcard enclose them in single quotes (‘). For example; ‘$’.
username *name*	Displays information for connections for the subscriber identified by *name*. The user must have ben previously configured. *name* must be a sequence of characters and/or wildcard characters ('$' and '*') from 1 to 127 characters in length. The * wildcard matches multiple characters and the $ wildcard matches a single character. If you do not want the wildcard characters interpreted as wildcard enclose them in single quotes (‘). For example; ‘$’.

The following example shows the output of this command both before and after a session recovery operation has been performed. The “Redundancy Status” fields in this example have been bold-faced for clarity.

```
username: user1       callid: 01ca11b1         msid: 0000100003
  Card/Cpu: 4/2
  Sessmgr Instance: 7
  Primary callline:
  Redundancy Status: Original Session
  Checkpoints    Attempts    Success    Last-Attempt    Last-Success
     Full:             69         68         29800ms         29800ms
     Micro:           206        206         20100ms         20100ms
   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_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
       SMGR_STATE_CONNECTED           SMGR_EVT_AUTH_REQ
       SMGR_STATE_CONNECTED           SMGR_EVT_AUTH_SUCCESS
       SMGR_STATE_CONNECTED           SMGR_EVT_AUTH_REQ
       SMGR_STATE_CONNECTED           SMGR_EVT_AUTH_SUCCESS
       SMGR_STATE_CONNECTED           SMGR_EVT_AUTH_REQ
       SMGR_STATE_CONNECTED           SMGR_EVT_AUTH_SUCCESS
       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
```

Notice that is the example above, where the session has been recovered/recreated, that state events (FSM Event State field) no longer exist. This field is re-populated as new state changes occur.

Chapter 38
Subscriber Overcharging Protection

This chapter provides information on overcharging protection support for subscribers in UMTS network while loss of radio coverage occurred. 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 features described in this chapter is an enhanced feature and need enhanced feature license. This support is only available if you have purchased and installed particular feature support license on your chassis.

This chapter discusses following topics for feature support of Subscriber Overcharging Protection on loss of radio coverage for UE in UMTS networks:

- Introduction
- Supported Standards
- Supported Networks and Platforms
- Licenses
- Overcharging Protection Configuration
- Save the Configuration
- Verifying Your Configuration

Introduction

Mobile carriers typically are always looking for the solution for maximizing their network resource and, at the same time, looking for the method to effectively use of their network resource. Some mobile carriers charge their subscribers based on counts in x-CDR.

Consider scenario where a mobile is using streaming services or downloading very large files from external sources and it goes out of coverage. If this download is happening on Background/Interactive traffic class then the GGSN is unaware of such loss of connectivity as SGSN do not the perform Update PDP Context procedure to set QoS to 0kbps (this is done when traffic class is either Streaming or Conversational only). The GGSN still keeps on sending the downlink packets to SGSN, the SGSN does paging and finds out the mobile is not responding. SGSN will then drop the packets. In such cases, the G-CDR will have increased counts but S-CDR will not have them.

Some operators charge the subscribers based on G-CDR and overcharging can happen in the scenario given above. This feature is provided to avoid the overcharging is such cases.

This feature uses private extension for Loss Of Radio Coverage (LORC) sent by SGSN in Update PDP Context message to set QoS to 0 kbps, on detecting loss of radio coverage of UE. GGSN becomes aware of the LORC status by recognizing the message (from SGSN) and discards the downlink packets if LORC status indicates loss of radio coverage and also stops discarding downlink packets, if LORC status indicates gain of radio coverage of UE.

Supported Standards

This is a proprietary feature and implemented on the basis of internal standards.

Supported Networks and Platforms

This feature supports Cisco® ASR 5000 Chassis with StarOS Release 9.0 or later running GGSN and/or SGSN service for the core network services.

Licenses

This feature support requires following feature license installed on the system:

- 600-00-7820

Overcharging Protection Configuration

This section provides a high-level series of steps and the associated configuration examples for configuring the system with overcharging protection to subscriber on loss of radio coverage for UE in GGSN services.

Important: This section provides the minimum instruction set to configure the system to avoid the overcharging due to loss of radio coverage at the GGSN in UMTS network. Commands that configure additional function for this feature are provided in the Command Line Interface Reference.

These instructions assume that you have already configured the system-level configuration as described in *System Administration Guide* and specific product Administration Guide.

To configure the system to support overcharging protection on LORC in GGSN service:

Step 1 Configure the GTPC private extension in a GGSN service by applying the example configurations presented in the *GTP-C Private Extension Configuration* section below.

Step 2 Save the changes to system configuration by applying the example configuration found in Save the Configuration section of this chapter.

Step 3 Verify configuration of overcharging protection on LORC related parameters by applying the commands provided in the Verifying Your Configuration section of this chapter.

GTP-C Private Extension Configuration

This section provides the configuration example to configure the GTP-C private extensions for GGSN service:

```
configure
   context <vpn_context_name>
      ggsn-service <ggsn_svc_name>
         gtpc private-extension loss-of-radio-coverage
         end
```

Notes:

- *<vpn_context_name>* is the name of the system context where specific GGSN service is configured. For more information, refer *GGSN Service Administration Guide*.
- *<ggsn_svc_name>* is name of the GGSN service where you want to enable the overcharging protection for subscribers due to LORC.

Save the Configuration

To save changes made to the system configuration for this service, refer *Verifying and Saving Your Configuration* chapter.

Verifying Your Configuration

This section explains how to display and review the configurations after saving them in a .cfg file as described in *Verifying and Saving Your Configuration* chapter of this guide and also to retrieve errors and warnings within an active configuration for a service.

Important: All commands listed here are under Exec mode. Not all commands are available on all platforms.

These instructions are used to verify the overcharging protection support configuration.

Step 1 Verify that your overcharging support is configured properly by entering the following command in Exec Mode:

show ggsn-service name *ggsn_svc_name*

The output of this command displays the configuration for overcharging protection configured in the GGSN service *ggsn_svc_name*.

```
Service name:           ggsn_svc_name
  Context:                service
  Accounting Context Name:service
  Bind:                   Done
  Local IP Address:       192.169.1.1       Local IP Port:       2123
.
.
  GTP Private Extensions:
       Preservation Mode
       LORC State
```

Step 2 Verify that GTP-C private extension is configured properly for GGSN subscribers by entering the following command in Exec Mode:

```
show subscribers ggsn-only full
```

The output of this command displays the LORC state information and number of out packets dropped due to LORC.

Chapter 39
Traffic Policing and Shaping

This chapter describes the support of per subscriber Traffic Policing and Shaping feature on ST16 and Cisco® ASR 5000 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.

This chapter included following procedures:

Overview

This section describes the traffic policing and shaping feature for individual subscriber. This feature is comprises 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.

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
```

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
```

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 the configuration as described in the *Verifying and Saving Your Configuration* chapter.

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 46. 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 the configuration as described in the *Verifying and Saving Your Configuration* chapter.

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.

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 the configuration as described in the *Verifying and Saving Your Configuration* chapter.

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 47. 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)

Value (bps)	Increment Granularity (bps)
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
```

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
```

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
```

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 the configuration as described in the *Verifying and Saving Your Configuration* chapter.

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 48. 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-Uplk (or SN1-QoS-Tp-Uplk)	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 49. 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.

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)	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. This attribute can be sent multiple times for different traffic classes. If Class is set to 0, it applies across all traffic classes.

Chapter 40
Ty Interface Support

This chapter provides information on configuring the IP Multimedia Subsystem (IMS) Authorization Service support with Ty interface support for IMS subscriber in cdma2000 networks. This chapter also describes the configuration and commands with Diameter attributes that are used to implement this feature.

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 *Packet Data Serving Node Administration Guide* for PDSN/FA service and *Home Agent Administration Guide* for HA service, before using the procedures in this chapter.

Important: The IMS Authorization Service feature described in this chapter is only available if you have purchased and installed a Dynamic Policy Interface feature license on chassis. If you have not previously purchased this enhanced feature, contact your sales representative for more information.

This chapter includes the following:

Overview

Important: The system supports IMS authorization for IMS subscribers with Ty interface in cdma2000 networks. IMS authorization support is a license enabled feature and you must obtain and install a license key to enable it.

IMS service provides wide application support for transport of voice, video, and data independent of the access support. The roaming IMS subscriber in cdma2000 networks requires specific interface support for service access in this network.

Apart from other functionality sufficient, uninterrupted, consistent, and seamless user experience is expected by a subscriber during an application session. It is also important that the subscriber gets charged only for the amount of resources consumed by the particular IMS application used.

In view of required flow bandwidth and QoS, the ASR 5000 Series Platforms provide 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.

The system provides Ty interface to implement IMS authorization in cdma2000 networks for Access Gateways (PDSN and HA). 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 50. Applicable Products and Relevant Sections

Applicable Product(s)	Refer to Sections

PDSN	• Access Gateway Functionality for IMS Authorization
	• How it Works
	  • Ty Interface Support with AGW
	• Configuring IMS Authorization Service
	• Enabling IMS Authorization and QoS Profile
	  • Configuring IMS Authorization in PDSN Service
	  • Configuring Policy Map and DSCP Marking for PDSNHA Service
	  • Applying IMS Authorization to a Subscriber

HA	• Access Gateway Functionality for IMS Authorization
	• How it Works
	  • Ty Interface Support with AGW
	• Configuring IMS Authorization Service
	• Enabling IMS Authorization and QoS Profile
	  • Configuring Policy Map and DSCP Marking for PDSNHA Service
	  • Applying IMS Authorization to a Subscriber

Supported Standards

Support for the following standards and requests for comments (RFCs) have been added with the IMS authorization service functionality:

- 3GPP2 TSG-X (PSN) X.P0013-014-0.8, Service Based Bearer Control – Ty Interface Stage-3
- RFC 3588, Diameter Base Protocol
- RFC 4006, Diameter Credit-Control Application

Supported Networks and Platforms

This feature supports all ASR 5000 Series Platforms for the core network services configured on the system.

Interfaces for IMS Authorization

For IMS deployment in cdma2000 networks the system uses policy-based admission control support and/or flow based charging. The interfaces supported to perform SBLP and FBC in IMS authorization service described in the following sections.

Ty Interface for PDSN/FA/HA

The Ty reference is an interface between an Access Gateway (AGW) and a Policy and Charging Rules Function (PCRF) in support of Service Based Bearer Control.

The system, acting as an AGW, performs the Traffic Plane Function (TPF) and Policy and Charging Enforcement Function (PCEF) functionality. PCRF may be co-located with the P-CSCF, but they are different entities.

The Ty interface performs interactions for the purpose of local authorization of bearer level QoS resources on the resources negotiated at the application layer and/or based on local policy. In addition it supports the passing of information that may be used to establish flow based charging policy. The Ty interface also provides the following features:

- Co-ordination between the application and the bearer level for per-flow accounting
- Notification of loss of bearer from the bearer control entity to the application
- Application control of the flow of packets on the bearer including the restriction of flow end-points (flow filtering)
- General policy considerations for the allocation of resources on the bearer in the local domain.

Terminology and Definition

This section provides the descriptions of the terms used in this chapter for IMS authorization service support.

- **Binding Information:** The binding information associates a bearer to the IP flows of a session. The binding information is based on IMS session information for IMS sessions and sent by UE. The system receives the binding information from the UE during bearer activation or modification. The binding information consists of flow identifiers, QoS information, and flow description, if any for the IMS session.
- **Flow Identifier:** In an IMS session an IP flow is indicated uniquely by means of a flow identifier. The flow identifier is created based on the ordinal number of the flow stream and of the IP flow in the session where the IP flows are arranged based on the ports used.
- **Binding Mechanism:** This mechanism is used to associate a bearer with the IP flow(s) of an IMS session in the PDF.
- **Service-based Local Policy (SBLP):** This term refers to the instantiation of a policy for use of bearer resources in the access network based on Authorization by a service. In the context of Ty interface this is the combined QoS given to a set of IP flows for an IMS session.
- **Service Based Authorization:** This term refers to the authorization for use of bearer resources in the access network based on a determination by the application, possibly due to negotiation involving the user. In general, bearer resources are authorized if the resources requested at the bearer do not exceed the resources negotiated or requested at the service level.
- **Charging Rule:** This is a set of information that contains the key to identify the packet flow(s) and defines how the flow(s) is to be charged. The Traffic Plane Function (TPF)/ PCEF installs the charging rules either statically or dynamically.
- **Dynamic Charging Rule:** The charging rules that depend upon real time analysis of the IP flow data being negotiated at the time of the session setup.
- **Main A10**: This is the primary bearer used for all the signaling. The traffic TFTs can be associated with both main A10 and auxiliary A10.
- **Auxiliary A10**: This is the secondary bearer used for data traffic only.

Access Gateway Functionality for IMS Authorization

This section describes the functionality of AGW service for IMS authorization to an IMS subscriber session in its networks.

The AGW contains and supports the Policy Enforcement Point (PEP)/PCEF function in Service Based Local Policy architecture and the Traffic Plane Function (TPF)/PCEF in the Flow Based charging (FBC) architecture. The individual requirements for each of these models are described in the following sections.

Policy Enforcement Point in SBLP

The Policy Enforcement Point in SBLP controls the quality of service that is provided to a combined set of IP flows. The basic PEP functionality for SBLP support is identified as:

- Support for “Gate” Functionality
- Support for Bearer Authorization
- Charging Correlation

Support for 'gate' Functionality

The PEP implements “gate” functionality which controls the flow of packets in the user plane based on the status of the gate. The gate operations are controlled on the basis of policy. It controls the allow and discard of IP packets in “open” or “closed” state of the gate. A gate applies to unidirectional flow(s) and consists of packet classifier and the associated gate status. The gate status is updated dynamically (to open/to close) based on policy decisions from PDF. When there are changes to a packet flow associated with an application (e.g. a mobile user puts a session on hold, or introduces a new media flow or terminates another), these changes are communicated to PEPs.

The gating of packet flows allows the operator the ability to control the use of the relevant IP resources, depending on the application being offered. The operator uses this capability to control the destination and source of the bearer. This gating function helps to prevent leakage of service and denial of service attacks in the network.

Support for Bearer Authorization

The system authorizes the use of bearer resources for the IMS media flow transport purpose. To achieve this, the system performs query and/or get unsolicited policy decision messages from the PDF. The decision messages indicate the IP flow(s) and the max value authorized for the resource which in this context is the QoS bandwidth used by the bearer.

The system also enforces this limit on the usage of the bearer for the transport of the IMS application media.

Charging Correlation

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 AGW provides the co-relation identifiers associated with the bearer along with its address to the PDF. The PDF in turn sends the IMS Charging Identifier (ICID), which is provided by the P-CSCF, to the AGW. The system generates the charging records including the co-related identifier as well as the ICID if received from PDF, so that the correlation of charging data can be done with the billing system.

The policy authorization transactions are carried over the Ty interface, which is based on the Diameter protocol.

Flow Based Charging with TPF/PCEF

Flow based charging handles differentiated charging of the bearer usage based on real time analysis of the service data flows. Generally charging for the usage of bearer is based on subscriber at the bearer level. Flow based charging provides a more granular charging mechanism where the charging is a dynamic function of the IP flows used by the media for an application.

The Charging Rule Function (CRF) or PCRF is capable of provisioning charging rules to the system where the charging rules are statically or dynamically defined at CRF/PCRF. A charging rule has a name, rating group, service identifier, service data flow filter and other charging related parameters associated with it.

The service data flow filters identify a flow and are unidirectional. These filters are associated with an IP 5 tuple (protocol, local address, local port, remote address, remote port). In case of static or predefined filters, the IP flow will be inspected by attributes beyond the 5 tuple of port, address and protocol. It is the function of Traffic Plane Function (TPF)/ Policy and Charging Enforcement Function (PCEF) to analyze the data packet based on the installed flow filters. If a packet does not match with the filters for the bearer, it is dropped by the TPF/PCEF. TPF/PCEF counts packets based on individual flow for charging purpose.

The charging rules also include the type of charging, (online or offline), the primary and secondary addresses for the offline, and online charging systems as applicable for the credit control of the session.

CRF/PCRF is also capable of generating unsolicited messages for provisioning charging rules for a bearer. The TPF/PCEF is required to enforce the charging rules from any decision received from CRF/PCRF.

The system supports the following charging rule operations over the Ty interface:

- Installation, modification, and removal of dynamic charging rule using Ty interface.
- Provide high priority to the dynamic charging rules over the static charging rules configured with the system.

Important: The system does not need any static configuration knowledge of traffic categories (Rating-Groups) dynamically received by CRF/PCRF and packets dropped due to the closure of "gate" are not counted against the quota for the matching category.

A dynamic charging rule has the following attributes:

- a charging rule name
- service data flow filter(s)
- precedence
- "gate" status
- QoS parameters

Policy Mapping between PCRF and PCEF

This section describes the process of policy mapping between PCRF/P-CSCF and PCEF/AGW.

The Operator can associate each subscriber with one static policy-group and one dynamic policy-group. The static policy-group parameters for the subscriber are configured by the operator, and the dynamic policy-group is automatically created from the Charging-Rule-Definition AVP returned by the PCRF for the subscriber. The contents of the dynamic policy-group can be updated on request from the PCRF.

The system applies the following rules to decide the order in which charging rules are applied for a subscriber:

- If the AGW service instance is NOT configured for SBBC service any static policy configured for the subscriber will be applied.
- If the AGW service instance is configured for SBBC and PCRF disables the charging any static policy configured for the subscriber will be applied.
- If the AGW service instance is configured for SBBC and PCRF enables the charging:
  - specific charging rules installed by the PCRF for the subscriber will be applied.
  - default charging rules installed by the PCRF for the subscriber will be applied.
  - any static policy configured for the subscriber will be applied.

The following rules decide the order of evaluation of Charging-Rules associated with the Policy-Group:

- The Charging -Rule-Definitions with the highest precedence will be evaluated first.
- If there are multiple Charging -Rule-Definitions with the same precedence the newer rule will be evaluated first.
- If there are multiple Charging -Rule-Definitions in the same RAR, CCA messages with the same precedence, the charging rules will be evaluated in the order of the occurrence in packet.

The charging rule definitions returned by the PCRF are translated in the policy infrastructure for the AGW. The section *Maintaining the Dynamic Policy-Group* defines the mapping of different entities in the policy infrastructure for the dynamic policy-group.

Maintaining the Dynamic Policy-Group

This section defines the mapping of the **Charging-Rule-Install** AVP to the dynamic policy-group and related policy infrastructure. The rules for creating a dynamic policy group are as follows:

- Create a dynamic policy group, if one does not exist, for each Charging-Rule-Install AVP in the packet:
  - If a charging rule with the same name exists, then replace all the policy-maps and class-maps with new information and update any precedence change in the policy-group.
  - If charging-rule does not exists; then create class-map, policy-map, and policy-group entries.

Important: For more information on Class-Map, Policy-Map, and Policy-Group, refer to the *Intelligent Traffic Control* chapter.

How it Works

This section describes the IMS authorization and dynamic policy support in a cdma2000 networks.

The DPCA is the Ty interface to the Policy Control and Charging Rule Function (PCRF). The PCRF contains both PDF (Policy Decision Function) and CRF (Charging Rule Function). The PCRF may reside in Proxy-Call Session Control Function (P-CSCF) or on stand-alone system.

The interface between IMSA with PCRF is the Ty interface and between Session Manager and Online Charging Service (OCS) is the Tx interface.

Important: Online charging is not supported in this release.

Ty Interface Support with AGW

IMS authorization support performs in following way to provide the IMS authorization service to a subscriber in cdma2000 network.

When an AGW receives a packet in downlink or uplink direction, it matches the packets against all the downlink/uplink PCC rules that configured by the PCRF for the subscriber. On the basis of matching rules it selects one PCC rule and applies the ‘gating policy’ as available in selected PCCC rule to the flow. It also applies the traffic policing to the flow on the basis of downlink/uplink bandwidth limit provided in authorized QoS profile available in the PCC rule. To provide the bandwidth limit the AGW uses 3 color 2 bucket in the color blind mode to enforce the bandwidth.

If flow is in uplink direction AGW applies the routing function on the packet or extract the flow-identifier for the downlink flow with the use of Flow-Identifier AVP in PCC rule.

Based on the flow -id the AGW selects the A10 to transport the packet and adds the flow identifier attributes to the GRE header and forwards the packet on the selected A10.

The following figure provides a high-level view of the IMS authorization process between a system and IMS components that is initiated by the MN. The table explains each step in detail.

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 52. IMS Authorization Call Flow

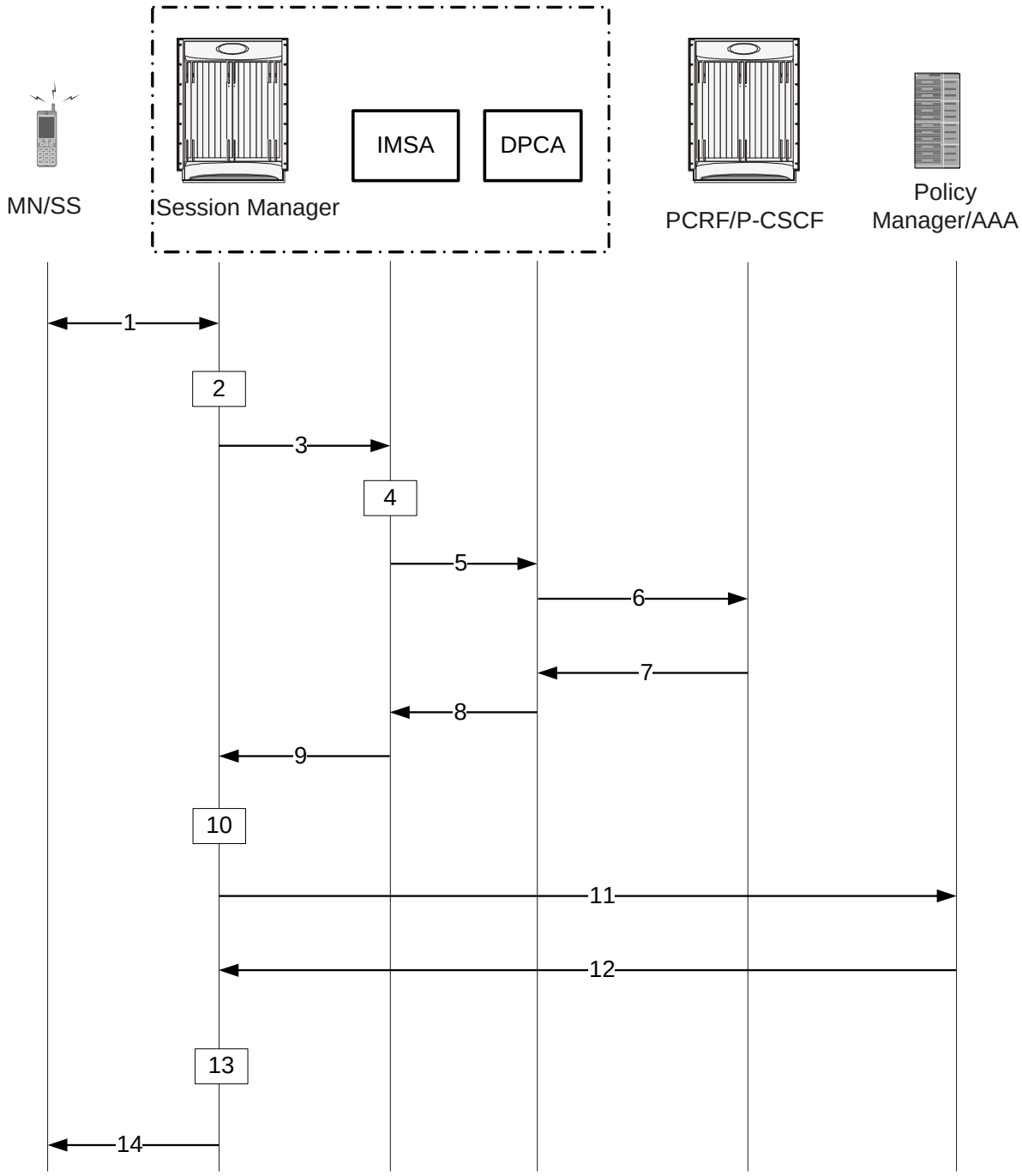


Table 51. IMS Authorization Call flow Description

Step	Description
1	IMS subscriber (MN) sends request for primary bearer(A10) activation/creation.

Step	Description
2	Session manager allocates IP address to MN.
3	Session manager sends IMS authorization request to IMS authorization service (IMSA).
4	IMSA creates a session with the PCRF on the basis of PCRF configuration.
5	IMSA sends request to DPCA module to issue the authorization request to selected PCRF.
6	DPCA sends a CCR-initial message to the selected PCRF. This message includes the IP address allocated to MN.
7	CCA message sent to DPCA. If a preconfigured rule set for the bearer is provided in PCRF, 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 the ACS manager.
11	ACS Manager 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 3GPP2 specific attributes (e.g. Subscriber, QoS etc.).
12	OCS returns a CCA-Initial message to activate the statically configured rulebase and includes pre-emptive credit quotas.
13	ACS manager responds to session manager with the response message for dynamic rule configuration.
14	On the basis of response for the bearer authorization, session manager sends the response to the MN and activates/rejects the call.
Note: Step 10 through 13 are not relevant to this release.	

Ty Interface Support with HA

IMS authorization support uses Mobile IP to manage session with HA. The Ty interface between HA and PCRF provides following functions:

- Local policy based authorization and bearer-level QoS negotiation
- Accounting co-ordination on a per flow basis, between bearer-level and application-level
- Flow control and filtering on the basis of subscriber profile
- Bearer resource allocation to the subscriber

In HA service deployment, the HA matches the packets received against all the downlink/uplink PCC rules that configured by the PCRF for the subscriber, and applies the ‘gating policy’ and traffic policing to the flow on the basis of applicable PCC rule. It also applies the Authorized-QoS profile available in the PCC rule. After applying the gating policy and QoS profile, the HA tunnels the packet to the AGW for downlink traffic or applies routing function on the packet for uplink traffic flow.

The following figure provides a high-level view of the IMS authorization process between a system and IMS components that is initiated by the MN. The following table explains each step in detail.

Figure 53. IMS Authorization HA Call Flow

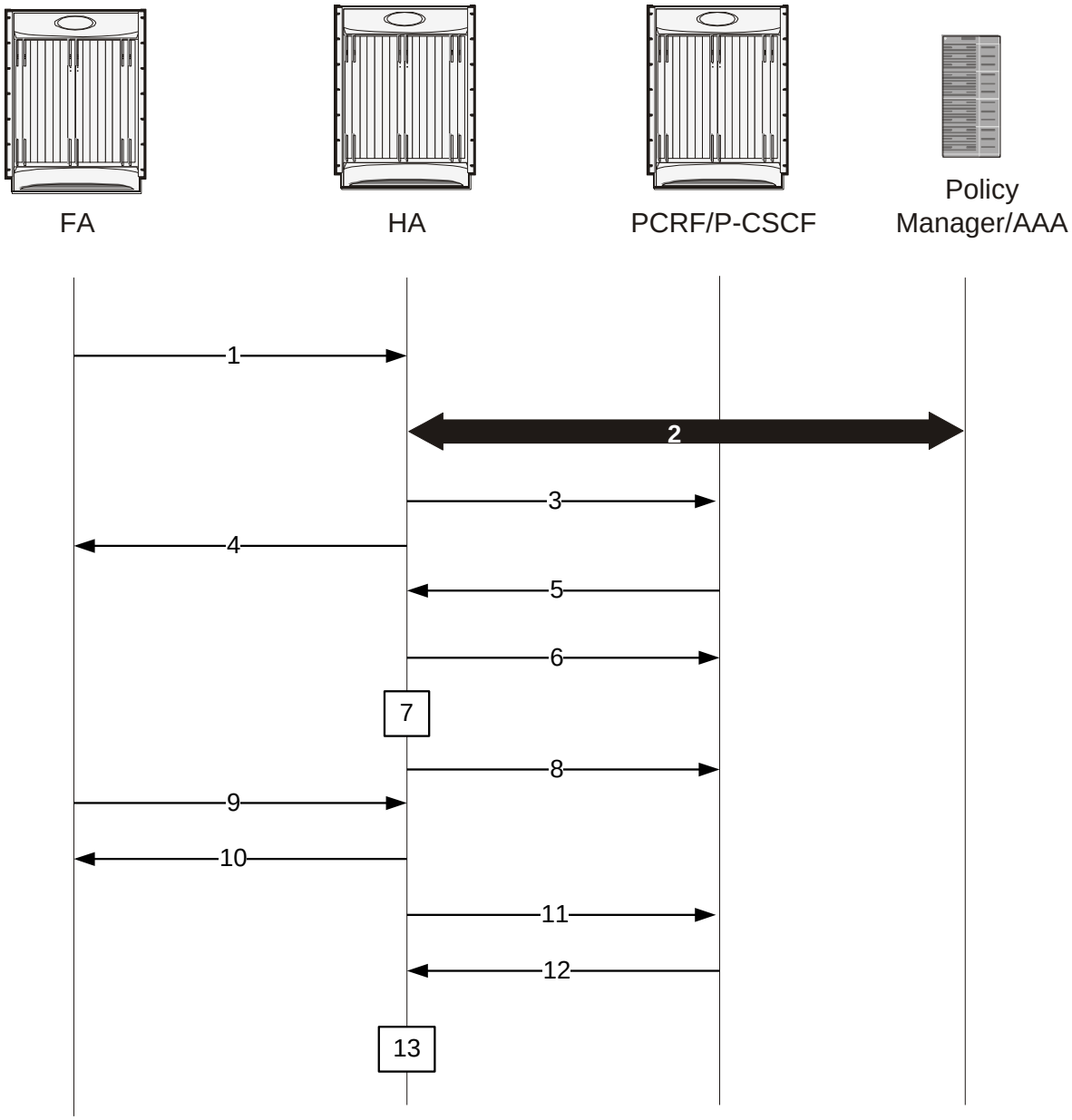


Table 52. IMS Authorization HA Call flow Description

Step	Description
1	FA forwards the Registration Request MIP_RRQ to the HA.
2	HA performs AAA authentication for subscriber, if required.
3	After authentication CCR-Session-Up is initiated by HA with PCRF.
4	HA sends MIP-RRP message to FA.

Step	Description
5	PCRF with CCA-Session-Up to HA. • If the response is failure the call is rejected. • If the response is succeeded the charging rule, if available for this response, is installed and session established.
6	After a request for modification of an existing media flow is requested, the PCRF initiates an RAR request with a modified Authorized QoS for the specific media flow.
7	HA updates the existing flows with new QoS, bandwidth, flow description, and gating parameters.
8	HA responds to the PCRF with a RAA message.
9	After the session MN wants to terminate the session and initiates the termination of session procedure. The FA sends an RRQ to HA initiating the termination of the Mobile IP tunnel.
10	HA responds with an RRP message to the FA indicating successful completion of the termination procedure.
11	HA generates a CC-Request-Session-Terminate message and sends to the PCRF.
12	PCRF clears all the flows corresponding to the session and responds with a CC-Answer-Session-Terminate message to HA.
13	HA clears all the flow information for the subscriber session.

Configuring IMS Authorization Service

Use the following example to enable IMS authorization service for IMS subscriber in cdma2000 networks:

```
configure
   context <name>
      ims-auth-service name <name>
      p-cscf discovery <option>
      end
```

Notes:

- 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.
- Define the P-CSCF discovery algorithm for either discovery or table options.
- Optional. Specify the trigger events to initiate re-authorization for a subscriber in IMS authorization service using the **reauth-trigger** command.


Important: The trigger events specified are applicable on the events generated by the PCRF only.

Configuring the Policy Control Settings

1. Enter the Policy Control Configuration mode for authorization and policy control parameter in IMS authorization service by entering the following command:

```
configure
   context <name>
      ims-auth-service name <name>
         policy-control
            diameter origin endpoint <endpoint_name>
            diameter dictionary <dictionary_type>
            failure-handling <handling_option>
            diameter host-select row-precedence <precedence_value>
```

```
            diameter host-select reselect subscriber-limit <subs_limit> time-
interval <duration>

            end
```

Notes:

- **failure-handling** specifies the handling of the failure of Enhanced-Policy Decision Function (E-PDF).
- **diameter host-select** adds or appends rows with primary and/or secondary host name to a Diameter host table with precedence.
- **reselect subscriber-limit** defines the pacing of the reselection or switching of the Enhanced-Policy Decision Function (E-PDF) after a change occurs in table configuration for an IMS Authorization Service.
- Optional. Define the method or algorithm to select Diameter host table by entering **diameter host-select table { 1 | 2 } [algorithm { ip-address-modulus | msisdn-modulus | round-robin }]**
- Repeat as needed to configure more IMS authorization services as required within the specified context.

Important: A maximum of 16 authorization services can be configured globally in the system. There is also a system limit for maximum number of total configured services.

Verifying your configuration

Step 1 Verify your configured parameters for IMS authorization service by entering the following command:

```
show ims-authorization service name <name>
```

Important: For more information on keywords/options of the **show ims-authorization** command, refer to the Executive Mode Commands chapter in *Command Line Interface Reference*.

Step 2 Save your configuration as described in *Saving Your Configuration* chapter.

Step 3 Proceed to Configuring IMS Authorization in cdma2000 Networks section to enable the IMS authorization in a core service and associate the configured IMS Authorization service to a subscriber within that context.

Enabling IMS Authorization and QoS Profile

This section describes the procedure to configure the QoS profile identifier for QoS mapping and associates an IMS authorization through Ty interface at the service level.

This section also provides the configuration to associate an IMS authorization service functionality to a previously configured subscriber within the same context in cdma2000 networks.

Configuring IMS Authorization in PDSN Service

This section describes how to associate an IMS authorization service at the service level for all subscriber in that service and to create the QoS profile id for QoS mapping in cdma2000 networks.

This procedure assumes that you are at the Context Configuration mode level in the same context as specified in Configuring IMS Authorization Service section and following prompt is appearing:

```
config
   context <name>
      pdsn-service <pdsn_svc_name>
         qos-profile-id-mapping profile-id <id_num> <options>
         end
```

Notes:

- You can configure qos-profile parameters to define description, downlink/uplink bandwidth, drop-rate, latency, and the maximum drop percentage.
- Repeat this configuration to add additional subscribers for IMS Authorization service.

Verifying your configuration for the PDSN Service

Step 1 Verify the IMS authorization configuration for the PDSN service(s) by entering the following command:

```
show dynamic-policy statistics pdsn-service <pdsn_svc_name>
```

Step 2 Save your configuration as described in *Saving Your Configuration* chapter.

Configuring Policy Map and DSCP Marking for PDSN/HA Service

This section describes how to configure the DSCP marking and policy map for subscribers configured in a destination context for IMS authorization.

```
configure
   context <name>
      class-map name <class_name>
         match any
         exit
      policy-map name <policy_name>
         type template
         qos traffic-police committed <bps> peak <bps> burst-size <bytes>
exceed-action <option>
         qos user-datagram dscp-marking <dscp_code>
         qos encaps-header dscp-marking
         class-map name <class_name>
         exit
      policy-group name <policy_group>
         end
```

Notes:

- Save your configuration as described in *Saving Your Configuration* chapter.

Applying IMS Authorization to a Subscriber

This section describes how to associate an IMS authorization service with individual subscriber(s) in cdma2000 networks.

This configuration example assumes that you are at the context configuration mode level of the same context as specified in the Configuring IMS Authorization Service section

```
config
   context <name>
      subscriber name <subs_name>
```

```
ims-auth-service <ims_auth_name>
end
```

Notes:

- Repeat this configuration example to configure additional subscribers for IMS Authorization service.

Verifying the IMS Authorization configuration

Step 1 Verify the IMS authorization configuration for the subscriber(s)

```
context <context_name>
   show subscribers ims-auth-service
```

Step 2 Save your configuration as described in *Saving Your Configuration* chapter.

Chapter 41
VLANs

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 before using the procedures in this chapter.

Sections in this chapter include:

Overview

Virtual LANs (VLANs) provide greater flexibility in the configuration and use of contexts and services.

They are configured as “tags” on a per-port basis and allow more complex configurations to be implemented. The VLAN tag allows a single physical port to be bound to multiple logical interfaces that can be configured in different contexts. Therefore, each Ethernet port can be viewed as containing many logical ports when VLAN tags are employed.

Important: VLANs are supported in conjunction with ports on the Ethernet 10/100 and 1000 line cards. (VLAN tagging is not supported for SPIO ports.) The system supports the configuration of VLANs as follows:

- Ethernet 1000 Line Card: 1024 VLANs per port.
- Ethernet 10/100 Line Card: Maximum of 256 VLANs per port and a maximum of 1017 tagged VLANs and 8 untagged VLANs per Line Card. (VLANs including tagged and/or untagged across all the ports of a single 10/100 Line Card can not be more than 1025.)

This chapter includes the following procedures:

- Creating VLAN Tags
- Configuring Subscriber VLAN Associations

Creating VLAN Tags

Use the following example to create VLANs on a port and bind them to pre-existing interfaces. For information on creating interfaces, refer to the System Administration Guide.

```
config
   port ethernet <slot/port>
      no shutdown
      vlan <vlan_tag_ID>
      no shutdown
      bind interface <interface_name> <context_name>
      end
```

Notes:

- A maximum of 1024 VLANs can be configured per port.
- Configure a subscriber-vlan to associate a VLAN with specific subscribers. Refer to the *Configuring Subscriber VLAN Associations* section of this chapter for more information.
- Repeat this procedure as needed to configure additional VLANs for the port.

Verify the port configuration

Use the following command to verify the port configuration:

```
show port info <slot/port>
```

An example of this command's output is shown below:

```
Port: 17/1
 Port Type : 10/100 Ethernet
 Description : (None Set)
 Controlled By Card : 1 (Packet Accelerator Card)
 Redundancy Mode : Card Mode
 Redundant With : 33/1
 Physical ifIndex : 285278208
 Administrative State : Enabled
```

```
Configured Duplex : Auto
Configured Speed : Auto
MAC Address : 00-05-47-01-11-00
Link State : Up
Link Duplex : Unknown
Link Speed : Unknown
Untagged:
Logical ifIndex : 285278209
Operational State : Down, Active
Tagged VLAN: VID 10
Logical ifIndex : 285278210
VLAN Type : Subscriber
Administrative State : Enabled
Operational State : Up, Active
Number of VLANs : 1
```

Notes:

- *Optional.* Repeat this configuration as needed to configure additional ports.
- *Optional.* Configure VLAN-subscriber associations if needed.
- Save your configuration as described in *Saving Your Configuration*.

Configuring Subscriber VLAN Associations

Subscriber traffic can be routed to specific VLANs based on the configuration of their user profile. Using this functionality provides a mechanism for routing all traffic from a subscriber over the specified VLAN. All packets destined for the subscriber must also be sent using only IP addresses valid on the VLAN or they will be dropped.

RADIUS Attributes Used

The following RADIUS attributes can be configured within subscriber profiles on the RADIUS to allow the association of a specific VLAN to the subscriber:

- **SN-Assigned-VLAN-ID :** In the Starent VSA dictionary
- **SN1-Assigned-VLAN-ID :** In the Starent VSA1 dictionary

Important: Since the instructions for configuring subscriber profiles 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.

Configuring Local Subscriber Profiles

Use the configuration example below to configure VLAN associations within local subscriber profiles on the system.

Important: These instructions assume that you have already configured subscriber-type VLAN tags according to the instructions provided in the *Creating VLAN Tags* section of this chapter.

```
config
   context <context_name>
      subscriber name <user_name>
         ip vlan <vlan_id>
         end
```

Verify the subscriber profile configuration

Use the following command to view the configuration for a subscriber profile:

show subscriber configuration username <*user_name*>

Notes:

- Repeat this command for each additional subscriber.
- Save your configuration as described in *Saving Your Configuration*.