# Cisco ASR 5000 Series Packet Data Serving Node Administration Guide

**Version 10.0**

**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-22939-01

Cisco ASR 5000 Series Packet Data Serving Node Administration 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:<br>`Login:` |
| Text represented as **`commands`** | This typeface represents commands that you enter, for example:<br>**`show ip access-list`**<br>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:<br>**`show card`** *`slot_number`*<br>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:<br>Click the **File** menu, then click **New** |

| Command Syntax Conventions | Description |
|---|---|
| { **`keyword`** or *`variable`* } | Required keywords and variables are surrounded by grouped brackets.<br>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).<br>Pipe filters can be used in conjunction with required or optional keywords or variables. For example:<br>**{ nonce \| timestamp }**<br>OR<br>[ **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
# CDMA2000 Wireless Data Services

The ASR 5000 provides wireless carriers with a flexible solution that functions as a Packet Data Support Node (PDSN) in CDMA 2000 wireless data networks.

This overview provides general information about the PDSN including:

- Product Description
- System Components and Capacities
- Features and FunctionalityBase Software
- Features and Functionality - Optional Enhanced Software Features
- CDMA2000 Data Network Deployment Configurations
- Understanding Simple IP and Mobile IP
- Supported Standards

# Product Description

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

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

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

# System Components

This section describes the hardware and software requirements for a PDSN service.

## Licenses

The PDSN is a licensed product. A session use license key must be acquired and installed to use the PDSN service.

The following licenses are available for this product:

- PDSN Software License, 10K Sessions
- PDSN Software License, 1K Sessions

## Hardware Requirements

This section describes the hardware required to enable the PDSN service.

### Platforms

The PDSN service operates on the following platform(s):

- ASR 5000

### ASR 5000 Platform System Hardware Components

The following application and line cards are required to support CDMA2000 wireless data services on the system:

- **System Management Cards (SMCs):** Provides full system control and management of all cards within the ASR 5000 platform. Up to two SMC can be installed; one active, one redundant.
- **Packet Services Cards (PSCs):** Within the ASR 5000 platform, PSCs provide high-speed, multi-threaded PPP processing capabilities to support either PDSN/FA or HA services. Up to 14 PSCs can be installed, allowing for multiple active and/or redundant cards.
- **Switch Processor Input/Outputs (SPIO):** Installed in the upper-rear chassis slots directly behind the SPCs/SMCs, SPIOs provide connectivity for local and remote management, Central Office (CO) alarms. Up to two SPIOs can be installed; one active, one redundant.
- **Ethernet 10/100 and/or Ethernet 1000/Quad Gig-E Line Cards (QGLC):** Installed directly behind PSCs, these cards provide the RP, AAA, PDN, and Pi interfaces to elements in the data network. Up to 26 line cards

should be installed for a fully loaded system with 13 active PSCs, 13 in the upper-rear slots and 13 in the lower-rear slots for redundancy. Redundant PSCs do no not require line cards.

- **Redundancy Crossbar Cards (RCCs):** Installed in the lower-rear chassis slots directly behind the SMCs, RCCs utilize 5 Gbps serial links to ensure connectivity between Ethernet 10/100 or Ethernet 1000 line cards/QGLCs and every PSC in the system for redundancy. Two RCCs can be installed to provide redundancy for all line cards and PSCs.

***Important:*** Additional information pertaining to each of the application and line cards required to support CDMA2000 wireless data services is located in the *Product Overview Guide*.

# Features and Functionality—Base Software

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

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

This section describes following features:

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

## Gx and Gy Support

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

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

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

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

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

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

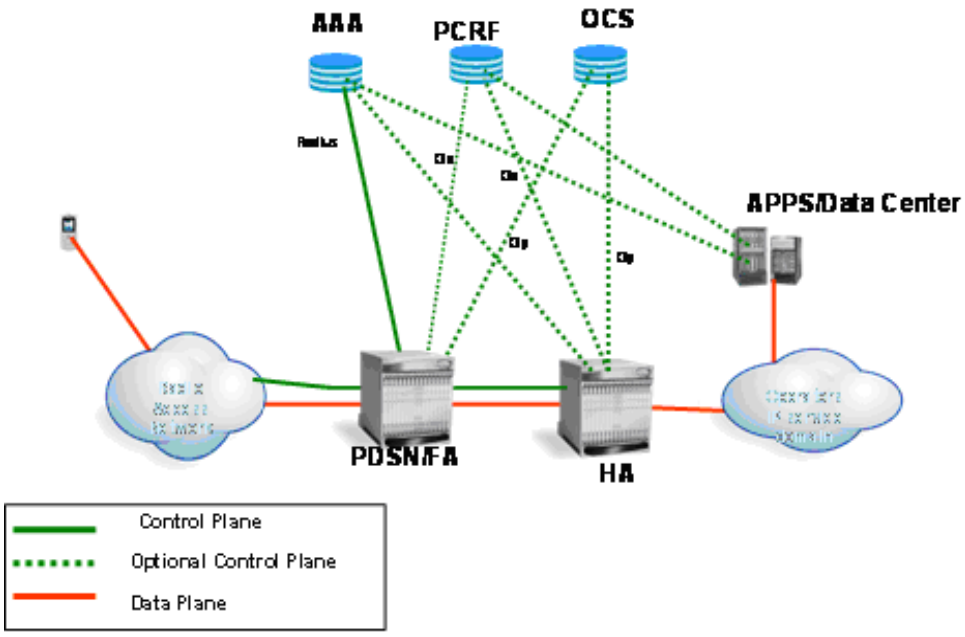
The configuration allows Policy support to be enabled on a per subscriber/APN basis.

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

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

The following figure shows the Gx support for Simple IP.

*Figure 1. Gx for Simple IP*




# RADIUS Support

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

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

## Description

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

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

Among the RADIUS parameters that can be configured are:

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

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

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

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

This feature works in following way:

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

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

***Important:*** For more information on RADIUS AAA configuration, refer AAA Interface Administration and Reference.

# Access Control List Support

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

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

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

There are two primary components of an ACL:

- Rule: A single ACL consists of one or more ACL rules. As discussed earlier, the rule is a filter configured to take a specific action on packets matching specific criteria. Up to 128 rules can be configured per ACL.

  Each rule specifies the action to take when a packet matches the specifies criteria. This section discusses the rule actions and criteria supported by the system.
- Rule Order: A single ACL can consist of multiple rules. Each packet is compared against each of the ACL rules, in the order in which they were entered, until a match is found. Once a match is identified, all subsequent rules are ignored.

***Important:*** For more information on Access Control List configuration, refer IP Access Control List chapter in System Enhanced Feature Configuration Guide.

# IP Policy Forwarding

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

## Description

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

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

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

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

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

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

# AAA Server Groups

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

## Description

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

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

**Important:** For more information on AAA Server Group configuration, refer *AAA Interface Administration and Reference*.

# Overlapping IP Address Pool Support

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

**Important:** For more information on IP pool overlapping configuration, refer VLANs chapter in *System Enhanced Feature Configuration Guide*.

# Routing Protocol Support

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

## Description

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

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

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

  OSPF routing is supported in accordance with the following standards:

  - RFC-1850, OSPF Version 2 Management Information Base, November 1995
  - RFC-2328, OSPF Version 2, April 1998
  - RFC-3101 OSPF-NSSA Option, January 2003

- **Border Gateway Protocol version 4 (BGP-4):** The system supports a subset of BGP (RFC-1771, A Border Gateway Protocol 4 (BGP-4)), suitable for eBGP support of multi-homing typically used to support geographically redundant mobile gateways, is supported.

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

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

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

***Important:*** For more information on IP Routing configuration, refer Routing chapter in *System Enhanced Feature Configuration Guide*.

# Management System Overview

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

## Description

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

These include:

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

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

# Bulk Statistics Support

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

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

## Description

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

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

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

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

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

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

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

# Threshold Crossing Alerts (TCA) Support

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

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

## Description

The following thresholding models are supported by the system:

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

Thresholding reports conditions using one of the following mechanisms:

- **SNMP traps:** SNMP traps have been created that indicate the condition (high threshold crossing and/or clear) of each of the monitored values.

  Generation of specific traps can be enabled or disabled on the chassis. Ensuring that only important faults get displayed. SNMP traps are supported in both Alert and Alarm modes.
- **Logs:** The system provides a facility called threshold for which active and event logs can be generated. As with other system facilities, logs are generated Log messages pertaining to the condition of a monitored value are generated with a severity level of WARNING.

  Logs are supported in both the Alert and the Alarm models.
- **Alarm System:** High threshold alarms generated within the specified polling interval are considered "outstanding" until a the condition no longer exists or a condition clear alarm is generated. "Outstanding" alarms are reported to the system's alarm subsystem and are viewable through the Alarm Management menu in the Web Element Manager.

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

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

# IP Header Compression - Van Jacobson

Implementing IP header compression provides the following benefits:

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

## Description

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

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

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

***Important:*** For more information on IP header compression support, refer IP Header Compression chapter in *System Enhanced Feature Configuration Guide*.

# DSCP Marking

Provides support for more granular configuration of DSCP marking.

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

# Features and Functionality - Optional Enhanced Software Features

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

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

This section describes following features:



## Session Recovery Support

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

### Description

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

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

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

There are two modes for Session Recovery.

- **Task recovery mode:** Wherein one or more session manager failures occur and are recovered without the need to use resources on a standby PSC. In this mode, recovery is performed by using the mirrored "standby-mode" session manager task(s) running on active PACs. The "standby-mode" task is renamed, made active, and is then populated using information from other tasks such as AAA manager.
- **Full recovery mode:** Used when a PSC hardware failure occurs, or when a PSC migration failure happens. In this mode, the standby PSC is made active and the "standby-mode" session manager and AAA manager tasks on the newly activated PSC perform session recovery.

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

***Important:*** For more information on session recovery support, refer Session Recovery chapter in *System Enhanced Feature Configuration Guide*.

# IPv6 Support

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

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

## Description

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

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

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

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

# L2TP LAC Support

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

## Description

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

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

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

***Important:*** For more information on L2TP Access Concentrator support, refer L2TP Access Concentrator chapter in *System Enhanced Feature Configuration Guide*.

# L2TP LNS Support

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

## Description

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

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

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

***Important:*** For more information on L2TP LNS support support, refer L2TP Access Concentrator chapter in *System Enhanced Feature Configuration Guide*.

# Proxy Mobile IP

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

## Description

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

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

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

***Important:*** For more information on Proxy Mobile IP configuration, refer Proxy Mobile IP chapter in System Enhanced Feature Configuration Guide.

# IP Security (IPSec)

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

- RFC 2401, Security Architecture for the Internet Protocol
- RFC 2402, IP Authentication Header (AH)
- RFC 2406, IP Encapsulating Security Payload (ESP)

- RFC 2409, The Internet Key Exchange (IKE)
- RFC-3193, Securing L2TP using IPSEC, November 2001

## Description

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

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

  Once an IPSec tunnel is established between an FA and HA for a particular subscriber, all new Mobile IP sessions using the same FA and HA are passed over the tunnel regardless of whether or not IPSec is supported for the new subscriber sessions. Data for existing Mobile IP sessions is unaffected.
- L2TP: L2TP-encapsulated packets are routed from the system to an LNS/secure gateway over an IPSec tunnel.

***Important:*** For more information on IPSec support, refer IP Security chapter in System Enhanced Feature Configuration Guide.

# Traffic Policing and Rate Limiting

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

## Description

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

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

- **Committed Data Rate (CDR):** The guaranteed rate (in bits per second) at which packets may be transmitted/received for the subscriber during the sampling interval.
- **Peak Data Rate (PDR):** The maximum rate (in bits per second) that packets may be transmitted/received for the subscriber during the sampling interval.

- **Burst-size:** The maximum number of bytes that may be transmitted/received for the subscriber during the sampling interval for both committed (CBS) and peak (PBS) rate conditions. This represents the maximum number of tokens that can be placed in the subscriber's "bucket". Note that the committed burst size (CBS) equals the peak burst size (PBS) for each subscriber.

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

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

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

  - **Drop**: The offending packet is discarded.
  - **Transmit**: The offending packet is passed.
  - **Lower the IP Precedence**: The packet's ToS octet is set to "0", thus downgrading it to Best Effort, prior to passing the packet.
  - **Buffer the Packet**: The packet stored in buffer memory and transmitted to subscriber once traffic flow comes in allowed bandwidth.

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

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

  ***Important:*** For more information on per subscriber traffic policing and shaping, refer Traffic Policing and Shaping chapter in System Enhanced Feature Configuration Guide.

## Intelligent Traffic Control

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

## Description

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

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

***Important:*** For more information on intelligent traffic control support, refer Intelligent Traffic Control chapter in *System Enhanced Feature Configuration Guide*.

# Dynamic RADIUS Extensions (Change of Authorization)

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

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

## Description

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

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

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

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

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

***Important:*** For more information on dynamic RADIUS extensions support, refer CoA, RADIUS, And Session Redirection (Hotlining) chapter in *System Enhanced Feature Configuration Guide*.

# Web Element Management System

Provides a Graphical User Interface (GUI) for performing Fault, Configuration, Accounting, Performance, and Security (FCAPS) management of the ASR 5000.

## Description

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

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

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

# CDMA2000 Data Network Deployment Configurations

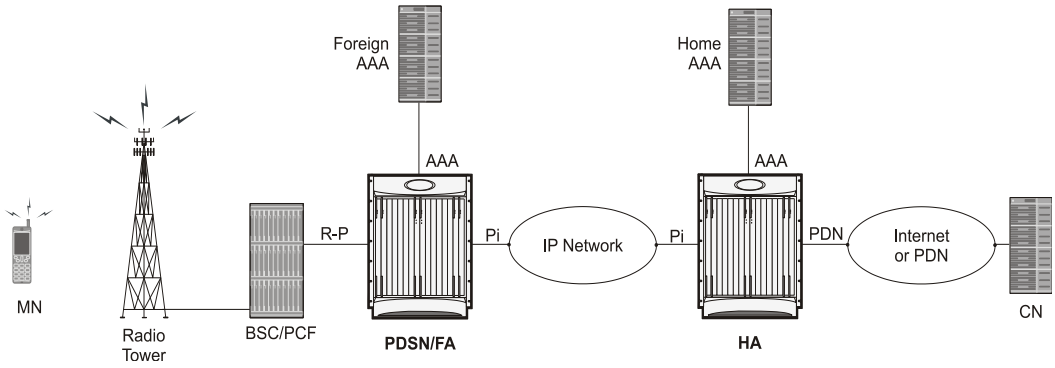
This section provides examples of how the system can be deployed within a wireless carrier's network. As noted previously in this chapter, the system can be deployed in standalone configurations, serving as a Packet Data Serving Node/Foreign Agent (PDSN/FA), a Home Agent (HA), or in a combined PDSN/FA/HA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

## Standalone PDSN/FA and HA Deployments

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

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

***Figure 2. PDSN/FA and HA Network Deployment Configuration Example***




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

### Interface Descriptions

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

### R-P Interface

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

The PCF can be co-located with the Base Station Controller (BSC) as part of the Radio Access Node (RAN). The PDSN/FA is connected to the RAN via Ethernet line cards installed in the rear of the chassis. The system supports either 8-port Fast Ethernet line cards (Ethernet 10/100) or single-port small form-factor pluggable (SFP) optical gigabit Ethernet line cards (Ethernet 1000) or four-port Quad Gig-E line cards (QGLC). These line cards also support outbound IP traffic that carries user data to the HA for Mobile IP services, or to the Internet or Wireless Access Protocol (WAP) gateway for Simple IP services.

### Pi Interfaces

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

### PDN Interfaces

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

### AAA Interfaces

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

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

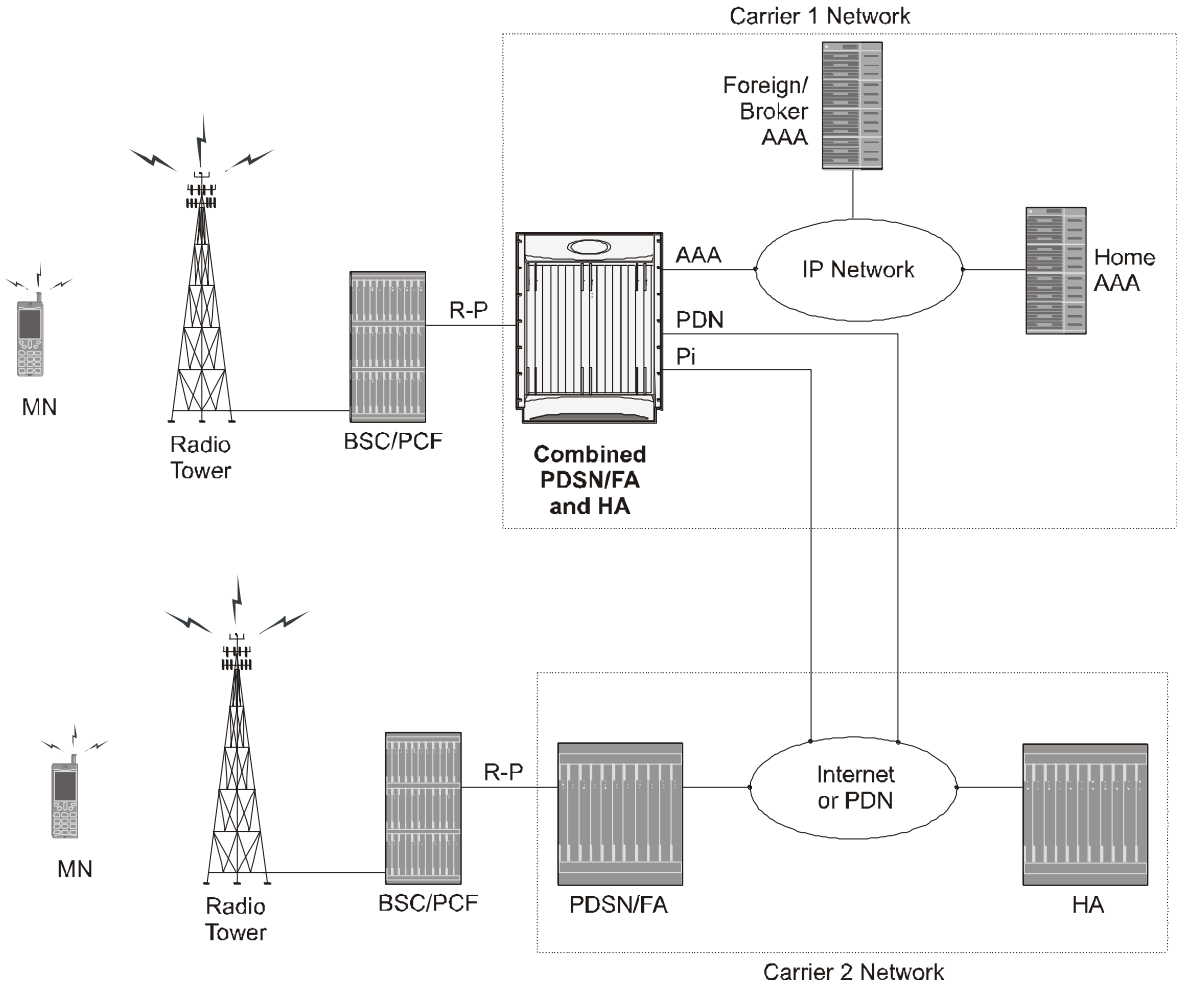
***Important:*** Subscriber AAA interfaces should always be configured using Ethernet line card interfaces for the highest performance. The out-of-band local context should not be used for service subscriber AAA functions.

## Co-Located Deployments

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

The following figure depicts a sample co-located deployment.

*Figure 3. Co-located PDSN/FA and HA Configuration Example*




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

# Understanding Simple IP and Mobile IP

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

- Local and public network access
- Private network access

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

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

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

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

## Simple IP

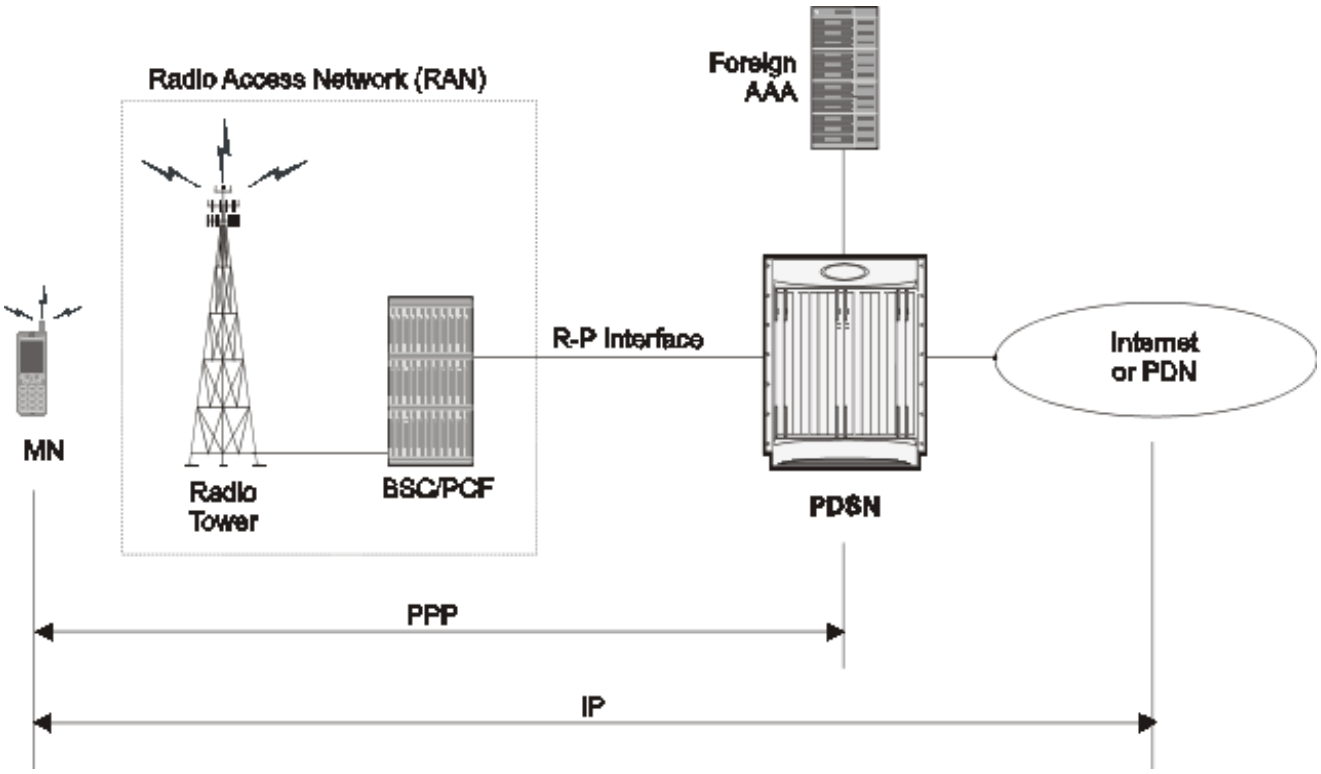
From a packet data perspective, Simple IP is similar to how a dial-up user would connect to the Internet using the Point-to-Point Protocol (PPP) and the Internet Protocol (IP) through an Internet Service Provider (ISP). With Simple IP, the mobile user is assigned a dynamic IP address from a PDSN or AAA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

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

## How Simple IP Works

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

*Figure 4. Simple IP Protocol Usage*




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

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

*Figure 5. Simple IP Call Flow*

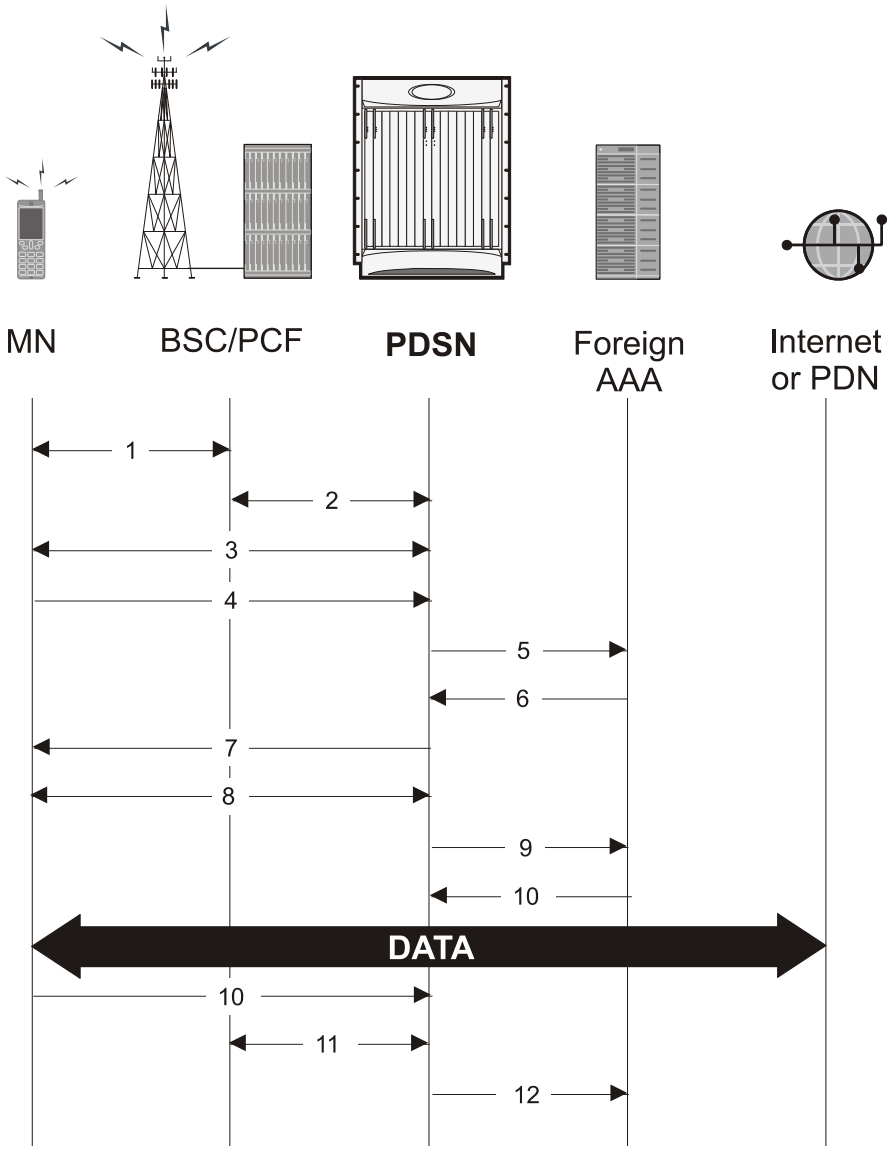



*Table 1. Simple IP Call Flow Description*

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

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

# Mobile IP

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

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

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

## Mobile IP Tunneling Methods

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

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

## IP in IP tunnels

IP in IP tunnels basically encapsulate one IP packet within another using a simple encapsulation technique. To encapsulate an IP datagram using IP in IP encapsulation, an outer IP header is inserted before the datagram's existing IP header. Between them are other headers for the path, such as security headers specific to the tunnel configuration. Each header chains to the next using IP Protocol values. The outer IP header Source and Destination identify the “endpoints” of the tunnel. The inner IP header Source and Destination identify the original sender and recipient of the datagram, while the inner IP header is not changed by the encapsulator, except to decrement the TTL, and remains unchanged during its delivery to the tunnel exit point. No change to IP options in the inner header occurs during delivery of the encapsulated datagram through the tunnel. If needed, other protocol headers such as the IP Authentication header may be inserted between the outer IP header and the inner IP header.

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

## GRE tunnels

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

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

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

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

## Forward Tunneling

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

The following figure shows an example of how forward tunneling is performed.

## Reverse Tunneling

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

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

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

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

Following are some of the advantages of reverse tunneling:

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

## Triangular Routing

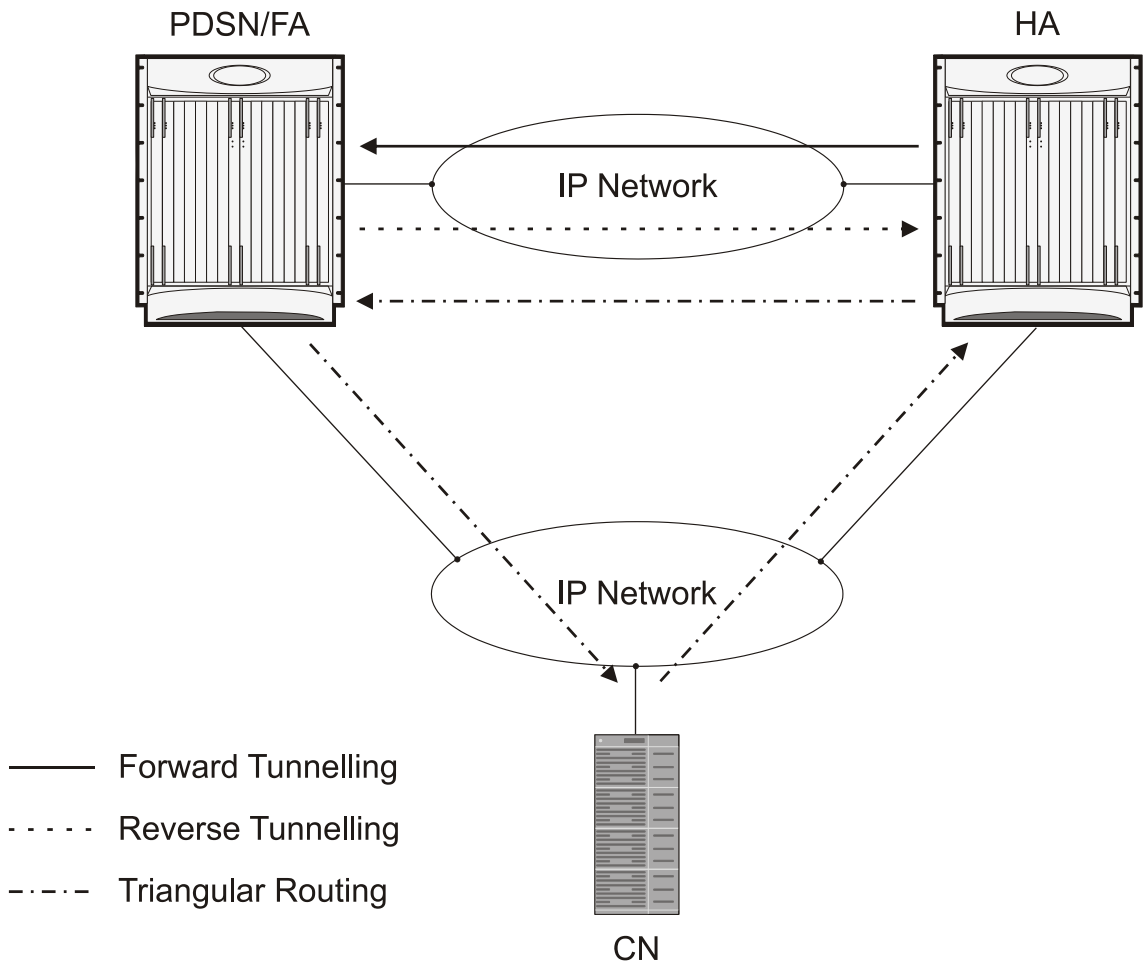
Triangular routing is the path followed by a packet from the MN to the Correspondent Node (CN) via the FA. In this routing scenario, the HA receives all the packets destined to the MN from the CN and redirects them to the MN's care-of-address by forward tunneling. In this case, the MN sends packets to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

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

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

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

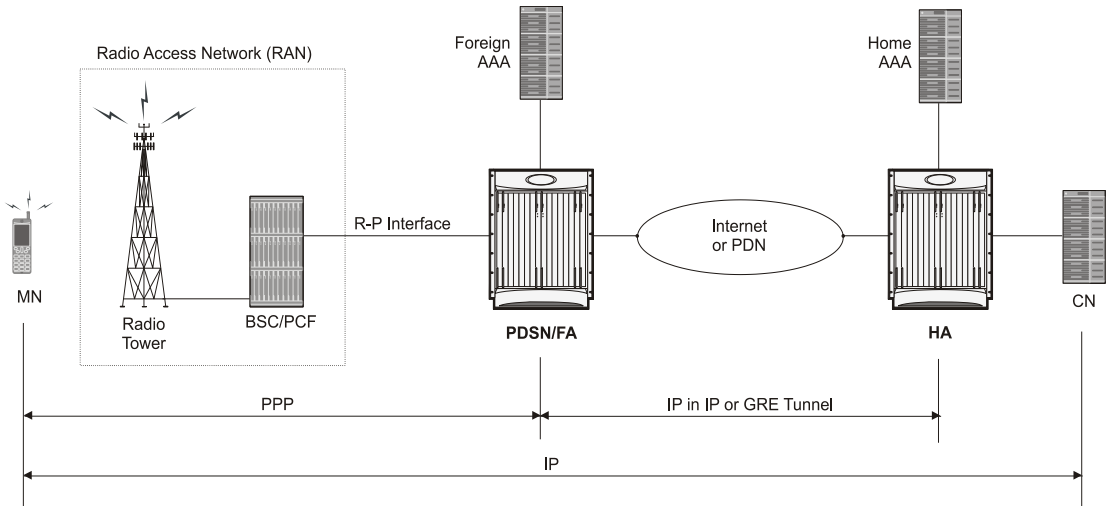
*Figure 6. Mobile IP, FA and HA Tunneling/Transport Methods*




## How Mobile IP Works

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

*Figure 7. Mobile IP Protocol Usage*




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

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

*Figure 8. Mobile IP Call Flow*

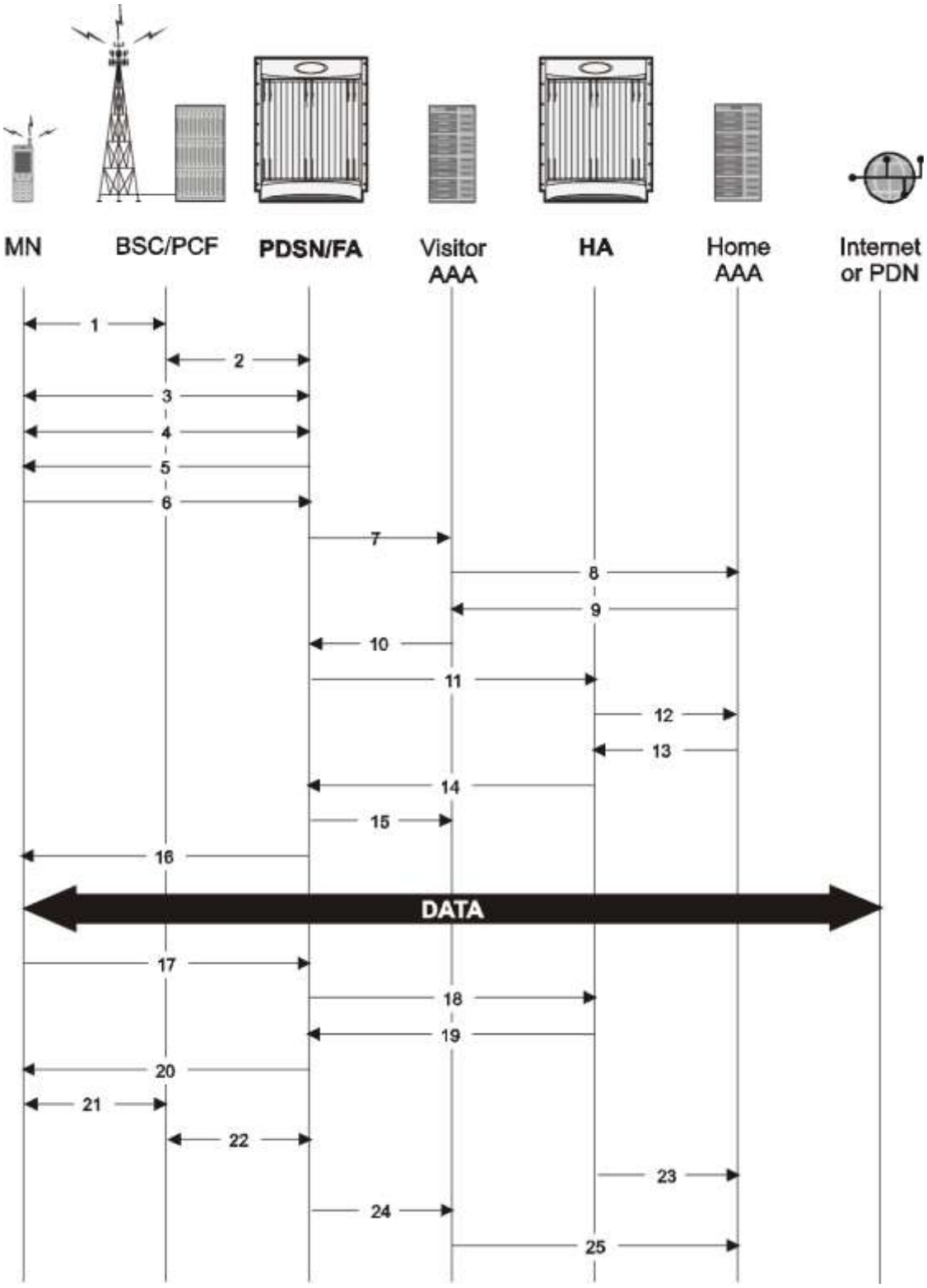


*Table 2. Mobile IP Call Flow Description*

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

# Proxy Mobile IP

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

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

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

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

## How Proxy Mobile IP Works

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

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

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

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

*Figure 9. AAA/PDSN Assigned IP Address Proxy Mobile IP Call Flow*

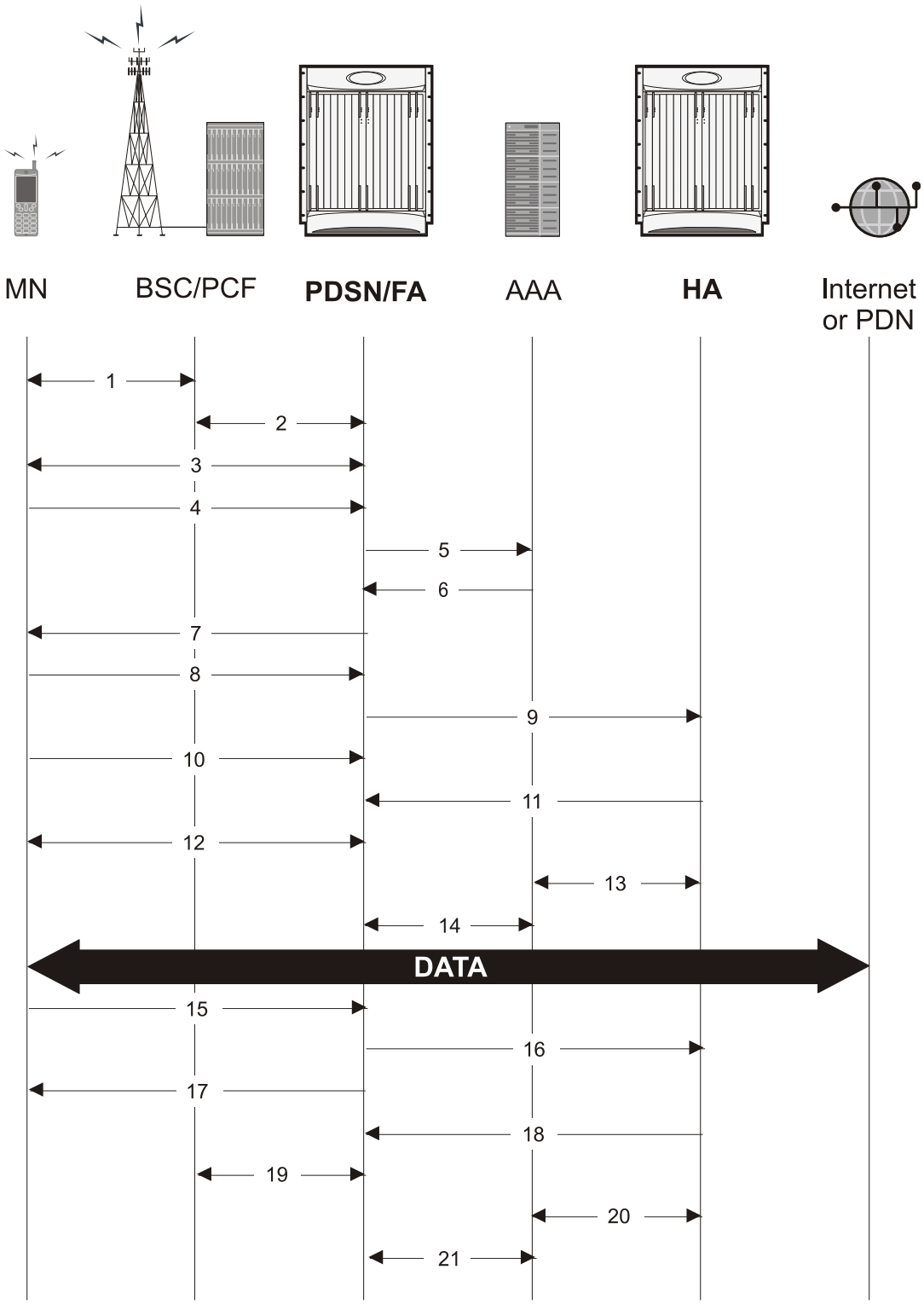


*Table 3. AAA/PDSN Assigned IP Address Proxy Mobile IP Call Flow Description*

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

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

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

*Figure 10. HA Assigned IP Address Proxy Mobile IP Call Flow*

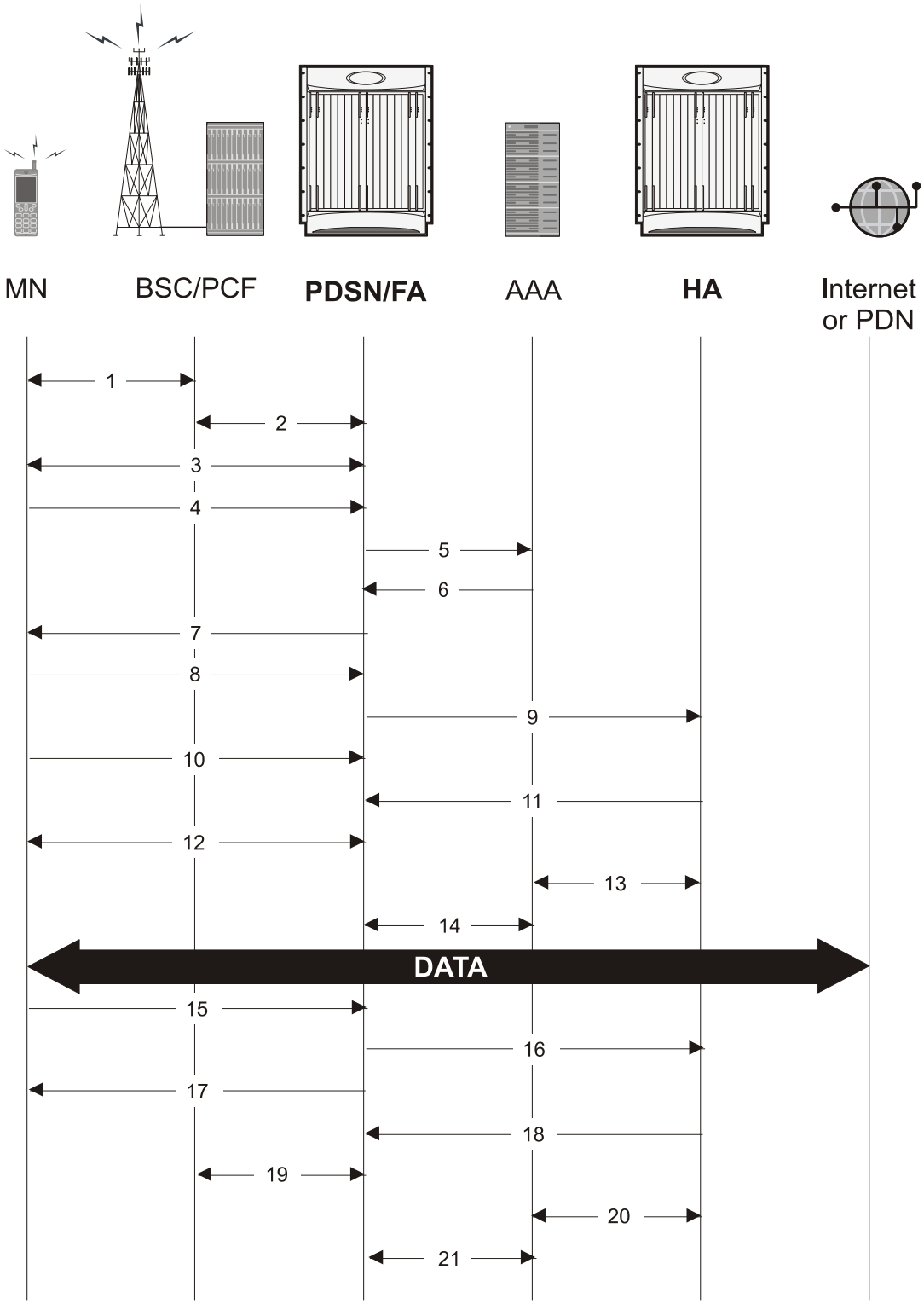


*Table 4. HA Assigned IP Address Proxy Mobile IP Call Flow Description*

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

# Supported Standards

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

## Requests for Comments (RFCs)

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

- RFC-1903, Textual Conventions for Version 2 of the Simple Network Management Protocol (SNMPv2), January 1996
- RFC-1904, Conformance Statements for Version 2 of the Simple Network Management Protocol (SNMPv2), January 1996
- RFC-1905, Protocol Operations for Version 2 of the Simple Network Management Protocol (SNMPv2), January 1996
- RFC-1906, Transport Mappings for Version 2 of the Simple Network Management Protocol (SNMPv2), January 1996
- RFC-1907, Management Information Base for Version 2 of the Simple Network Management Protocol (SNMPv2), January 1996
- RFC-1908, Coexistence between Version 1 and Version 2 of the Internet-standard Network Management Framework, January 1996
- RFC-1918, Address Allocation for Private Internets, February 1996
- RFC-1919, Classical versus Transparent IP Proxies, March 1996
- RFC-1962, The PPP Compression Control Protocol (CCP), June 1996
- RFC-1974, PPP STAC LZS Compression Protocol, August 1996
- RFC-2002, IP Mobility Support, May 1995
- RFC-2003, IP Encapsulation within IP, October 1996
- RFC-2004, Minimal Encapsulation within IP, October 1996
- RFC-2005, Applicability Statement for IP Mobility Support, October 1996
- RFC-2118, Microsoft Point-to-Point Compression (MPPC) Protocol, March 1997
- RFC-2136, Dynamic Updates in the Domain Name System (DNS UPDATE)
- RFC-2211, Specification of the Controlled-Load Network Element Service
- RFC-2246, The Transport Layer Security (TLS) Protocol Version 1.0, January 1999
- RFC-2290, Mobile IPv4 Configuration Option for PPP IPCP, February 1998
- RFC-2328, OSPF Version 2, April 1998
- RFC-2344, Reverse Tunneling for Mobile IP, May 1998
- RFC-2394, IP Payload Compression Using DEFLATE, December 1998
- RFC-2401, Security Architecture for the Internet Protocol, November 1998
- RFC-2402, IP Authentication Header (AH), November 1998
- RFC-2406, IP Encapsulating Security Payload (ESP), November 1998
- RFC-2408, Internet Security Association and Key Management Protocol (ISAKMP), November 1998
- RFC-2409, The Internet Key Exchange (IKE), November 1998
- RFC-2474, Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers, December 1998
- RFC-2475, An Architecture for Differentiated Services, December 1998
- RFC-2484, PPP LCP Internationalization Configuration Option, January 1999
- RFC-2486, The Network Access Identifier (NAI), January 1999
- RFC-2571, An Architecture for Describing SNMP Management Frameworks, April 1999

- RFC-2572, Message Processing and Dispatching for the Simple Network Management Protocol (SNMP), April 1999
- RFC-2573, SNMP Applications, April 1999
- RFC-2574, User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3), April 1999
- RFC-2597, Assured Forwarding PHB Group, June 1999
- RFC2598 - Expedited Forwarding PHB, June 1999
- RFC-2618, RADIUS Authentication Client MIB, June 1999
- RFC-2620, RADIUS Accounting Client MIB, June 1999
- RFC-2661, Layer Two Tunneling Protocol “L2TP”, August 1999
- RFC-2697, A Single Rate Three Color Marker, September 1999
- RFC-2698, A Two Rate Three Color Marker, September 1999
- RFC-2784, Generic Routing Encapsulation (GRE) - March 2000, IETF
- RFC-2794, Mobile IP Network Access Identifier Extension for IPv4, March 2000
- RFC-2809, Implementation of L2TP Compulsory Tunneling via RADIUS, April 2000
- RFC-2845, Secret Key Transaction Authentication for DNS (TSIG), May 2000
- RFC-2865, Remote Authentication Dial In User Service (RADIUS), June 2000
- RFC-2866, RADIUS Accounting, June 2000
- RFC-2867, RADIUS Accounting Modifications for Tunnel Protocol Support, June 2000
- RFC-2868, RADIUS Attributes for Tunnel Protocol Support, June 2000
- RFC-2869, RADIUS Extensions, June 2000
- RFC-3007, Secure Domain Name System (DNS) Dynamic Update, November 2000
- RFC-3012, Mobile IPv4 Challenge/Response Extensions, November 2000
- RFC-3095, Robust Header Compression (ROHC): Framework and four profiles: RTP, UDP, ESP and uncompressed, July 2001
- RFC-3101, OSPF NSSA Option, January 2003.
- RFC-3141, CDMA2000 Wireless Data Requirements for AAA, June 2001
- RFC-3143, Known HTTP Proxy/Caching Problems, June 2001
- RFC-3193, Securing L2TP using IPSEC, November 2001
- RFC-3241 Robust Header Compression (ROHC) over PPP, April 2002
- RFC-3409, Lower Layer Guidelines for Robust (RTP/UDP/IP) Header Compression, December 2002
- RFC-3519, NAT Traversal for Mobile IP, April 2003
- RFC-3543, Registration Revocation in Mobile IPv4, August 2003
- RFC 3576 - Dynamic Authorization Extensions to Remote Authentication Dial In User Service (RADIUS), July 2003
- RFC-3706, A Traffic-Based Method of Detecting Dead Internet Key Exchange (IKE) Peers, February 2004
- RFC-3759, Robust Header Compression (ROHC): Terminology and Channel Mapping Examples, April 2004
- RFC-3588, Diameter Based Protocol, September 2003

- RFC-4005, Diameter Network Access Server Application, August 2005
- RFC-4006, Diameter Credit-Control Application, August 2005
- Draft, Generalized Key Distribution Extensions for Mobile IP
- Draft, AAA Keys for Mobile IP

# TIA and Other Standards

## Telecommunications Industry Association (TIA) Standards

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

## Object Management Group (OMG) Standards

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

## 3GPP2 Standards

- 3GPP2 A.S0001-A v2: 3GPP2 Access Network Interfaces Interoperability Specification (also known as 3G-IOS v4.1.1)
- 3GPP2 P.S0001-A-3: Wireless IP Network Standard
- 3GPP2 P.S0001-B: Wireless IP Network Standard

- 3GPP2 S.R0068: Link Layer Assisted Robust Header Compression
- [9] 3GPP2 C.S0047-0: Link Layer Assisted Service Options for Voice-over-IP: Header Removal (SO60) and Robust Header Compression (SO61)
- 3GPP2 A.S0008 v3.0 Interoperability Specification (IOS) for High Rate Packet Data (HRPD) Access Network Interfaces
- 3GPP2 A.S0015-0 v2: Interoperability Specification (IOS) for CDMA2000 11 Access Network Interfaces — Part 5 (A3 and A7 12 Interfaces) (Partial Support) (also know as 3G-IOSv4.2)
- 3GPP2 P.S0001-B V1.0.0 Wireless IP Network Standard October 25, 2002 (relating to MIP interactions with IPSEC)
- 3GPP2 P.S0001 (TIA/EIA/IS-835-1) Version 1.0, Wireless IP Network Standard - December 10, 1999
- 3GPP2 P.R0001 (TSB115) Version 1.0.0, Wireless IP: Architecture Based on IETF Protocols - July 14, 2000
- 3GPP2 3GPP2 X.S0011-005-C Version: 1.0.0, CDMA2000 Wireless IP Network Standard: Accounting Services and 3GPP2 RADIUS VSAs - August 2003
- 3GPP2 X.S0011-006-C Version: 1.0.0, CDMA2000 Wireless IP Network Standard: PrePaid Packet Data Service - Date: August 2003
- 3GPP2 TSGA A.S0013-c v0.4 Interoperability Specification (IOS) for CDMA2000 June 2004
- 3GPP2 TSG-A A.S.0017-C baseline Interoperability Specification (IOS) for CDMA2000 Access Network Interfaces - Part 7(A10 and A11 Interfaces) (IOS v5.0 baseline) June 2004
- 3GPP2 A.S0012-D Segmentation for GRE January, 2005
- Inter-operability Specification (IOS) for CDMA2000 Access Network Interfaces
- 3GPP2 X.S0011-005-D Accounting Services and 3GPP2 RADIUS VSAs, February 2006
- 3GPP2 TSG-X (PSN) X.P0013-014-0, Service Based Bearer Control – Ty Interface Stage-3

## IEEE Standards

- 802.1Q VLAN Standard

# Chapter 2
# Understanding the Service Operation and Configuration

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

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

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

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

# Terminology

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

## Contexts

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

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

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

Contexts on the system can be categorized as follows:

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

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

## AAA Realms

A AAA realm is the location within the AAA context where subscriber-specific templates can be defined that are applied to subscribers who match that realm. A AAA realm is considered part of the AAA context; and the AAA

context itself is also considered to be a realm. There may be many different AAA realms defined within a single AAA context.

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

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

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

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

# Ports

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

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

# Logical Interfaces

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

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

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

- Management interface: This interface provides the system's point of attachment to the management network. The interface supports remote access to the system CLI, Common Object Request Broker Architecture

(CORBA)-based management via the Web Element Manager application, and event notification via the Simple Network Management Protocol (SNMP).

The system defaults to a Local context which should not be deleted. Management interfaces are defined in the Local management context and should only be bound to the ports on the Switch Processor Input/Output (SPIO) cards .

- R-P interface: Also referred to as the A10/A11 interface, this interface is the communications path between the Radio Node (also referred to as a PCF) and the PDSN.

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

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

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

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

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

  PDN interfaces are bound to Ethernet ports.

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

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

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

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

# Bindings

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

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

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

bound to a logical interface will cause the logical interface to take on the characteristics of an R-P interface within a 3G CDMA2000 network.

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

# Services

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

- PDSN services: Required for both Simple IP and Mobile IP applications, PDSN services define PDSN functionality for the system. The PDSN service must be bound to a logical interface within the same context. Once bound, the interface takes on the characteristics of an R-P interface. Multiple services can be bound to the same logical interface. Therefore, a single physical port can facilitate multiple R-P interfaces.

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

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

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

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

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

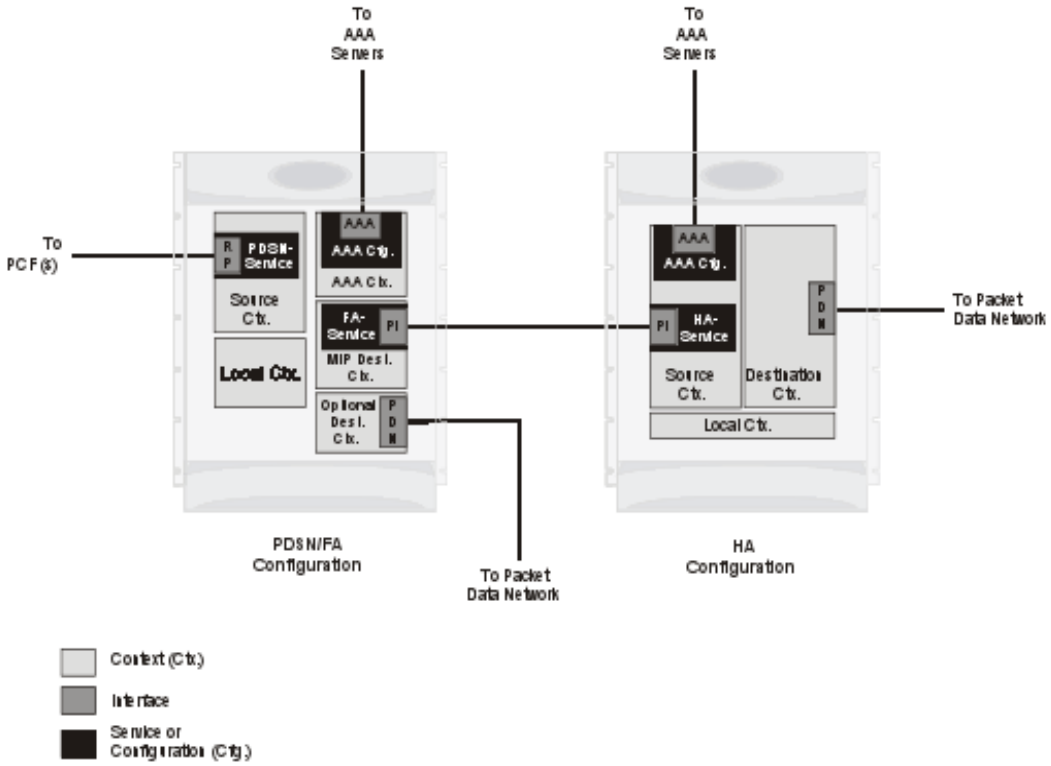
  If the HA service is configured within the same system as the PDSN/FA, then it should be configured within the same context as the FA service. This context, then, would also facilitate the PDSN interfaces to the data network.

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

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

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

*Figure 11. Service, Interface, and Context Relationship Within the System for CDMA 2000 Networks*




# AAA Servers

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

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

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

# Subscribers

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

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

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

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

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

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

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

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

## Default Subscribers and Realm-based Subscriber Templates

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

## Default Subscriber

When each context is created, the system automatically creates a subscriber named default. There is only one default subscriber per context. The profile for the subscriber named default provides a configuration template of attribute values

for subscribers who are remotely authenticated in that context. Any subscriber information that is not included in a RADIUS-based subscriber's user profile is configured according to the defaults defined for the default subscriber.

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

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

## Realm-based Subscriber Templates

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

Since each realm is used for a specific group of subscribers (e.g. corporate subscribers who may only have access to a specific corporate network that is protected by a virtual private network), each realm must have the ability to define what AAA attributes should be applied to these different subscriber groups. This is achieved through the use of realm-based subscriber templates.

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

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

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

As depicted above, a context named "ingress" contains:

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

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

If a subscriber enters the system with a domain name that matches the context name "ingress" (example: user1@ingress), then the [ingress] context would be used for authentication. If the RADIUS server authenticates the subscriber and returns no value for the idle-timeout attribute, then this subscriber would be assigned the value contained

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

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

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

# How the System Selects Contexts

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

## Context Selection for Context-level Administrative User Sessions

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

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

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

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

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

***Table 5. Context-level Administrative User AAA Context Selection***

| Item | Description |
|---|---|
| 1 | During authentication, the system determines if local authentication is enabled in the local management context.<br>If it is, the system attempts to authenticate the administrative user in the localout-of-band management context. If it is not, proceed to item 2 in this table.<br>If the administrative user's username is configured, authentication is performed using the AAA configuration within the local management context. If not, proceed to item 2 in this table. |

| Item | Description |
|---|---|
| 2 | If local authentication is disabled on the system or if the administrative user's username is not configured in the local management context, then the system determines if a domain was received as part of the username.<br>If there is a domain and it matches the name of a configured context or domain, then the AAA configuration within that context is used.<br>If there is a domain and it does not match the name of a configured context or domain, go to item 4 in this table.<br>If there is no domain as part of the username, go to item 3 in this table. |
| 3 | If there was no domain specified in the username or the domain is not recognized, the system determines if an AAA Administrator Default Domain is configured.<br>If the default domain is configured and it matches a configured context, then the AAA configuration within the AAA Administrator Default Domain context is used.<br>If the default domain is not configured or does not match a configured context or domain, go to item 4 item this table |
| 4 | If a domain was specified as part of the username but it did not match a configured context, or if a domain was not specified as part of the username, the system determines if the AAA Administrator Last Resort context parameter is configured.<br>If a last resort context is configured and it matches a configured context, then the AAA configuration within that context is used.<br>If a last resort context is not configured or does not match a configured context or domain, then the AAA configuration within the local management context is used. |

# Context Selection for Subscriber Sessions

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

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

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

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

## AAA Context Selection for Subscriber Sessions

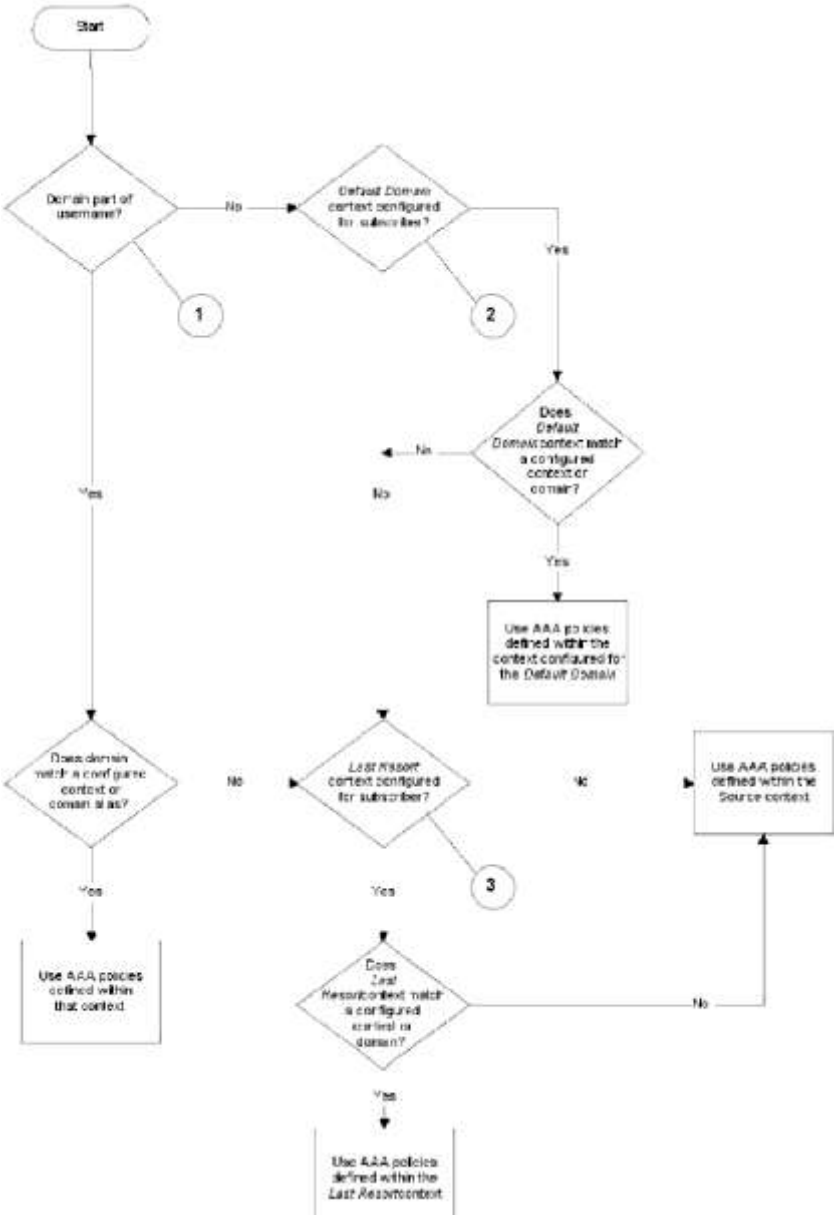
The following table and figure describe the process that the system uses to select an AAA context for a subscriber.

***Table 6. Subscriber AAA Context Selection***

| Item | Description |
|---|---|

| Item | Description |
|---|---|
| 1 | During authentication, the system determines if a domain was received as part of the username.<br>If there is a domain and it matches the name of a configured context or domain alias, then the AAA configuration within that context is used. |
| 2 | If there was no domain specified in the username, the system determines if an AAA Subscriber Default Domain was configured. The AAA Subscriber Default Domain parameter is a system-wide AAA parameter that provides the system with the name of a context or domain that can provide AAA functions.<br>If the AAA Subscriber Default Domain is configured and it matches a configured context or domain, then the AAA configuration within the AAA Subscriber Default Domain context is used.<br>If the AAA Subscriber Default Domain is not configured or does not match a configured context or domain, then the system determines if an AAA Subscriber Last Resort is configured. |

*Figure 12. Subscriber AAA Context Selection*




## Destination Context Selection For Subscriber Sessions

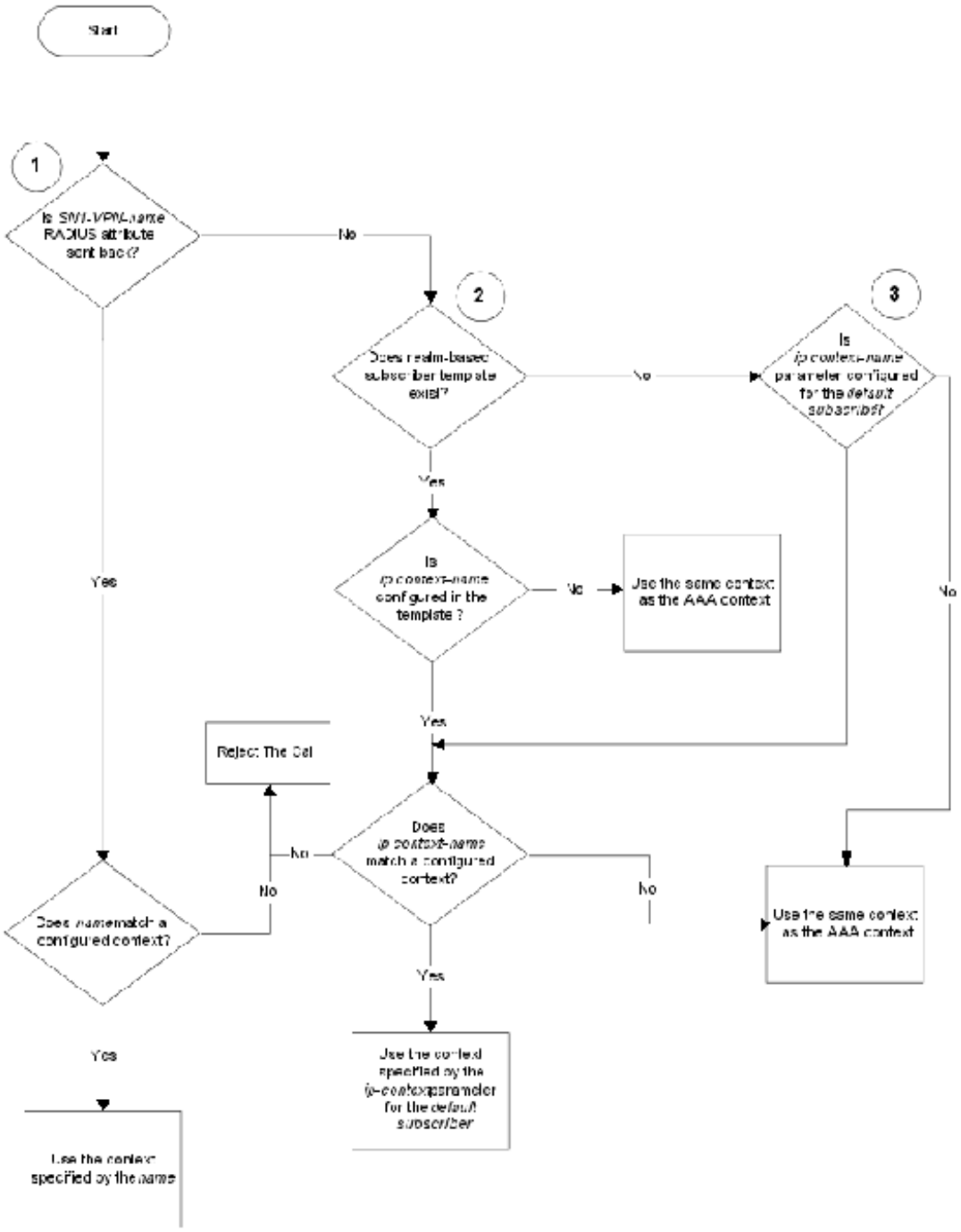
This section provides information on how a destination context is selected for subscribers whose profiles are configured on a RADIUS AAA server and for those whose profiles are locally configured. Note that the destination context for context-level administrative users is always the local management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

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

***Table 7. Subscriber Destination Context Selection***

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

*Figure 13. Subscriber Destination Context Selection*



# Chapter 3
# Simple IP Configuration Examples

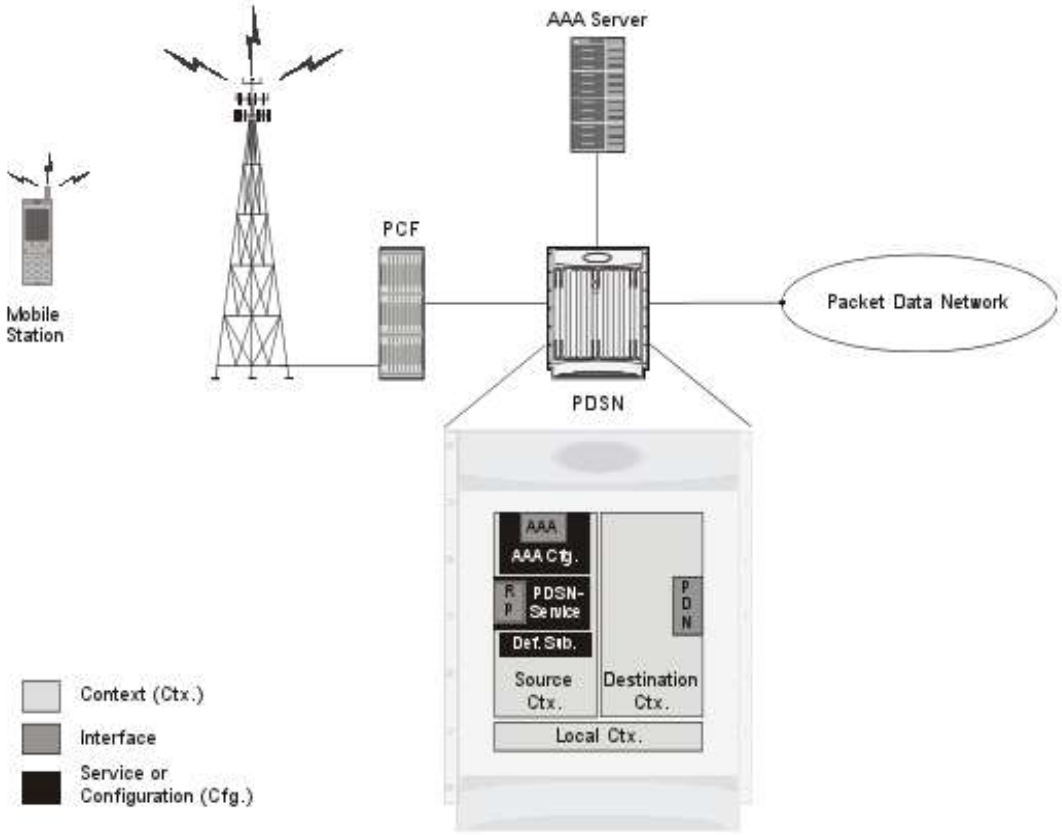
This chapter provides information for several configuration examples that can be implemented on the system to support Simple IP data services.

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

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

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

*Figure 14. Simple IP Support Using a Single Source and Destination Context*




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

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

# Information Required

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

## Source Context Configuration

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

***Table 8. Required Information for Source Context Configuration***

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

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

| Required Information | Description |
|---|---|
| RADIUS Authentication server | IP Address:<br>Specifies the IP address of the RADIUS authentication server the source context will communicate with to provide subscriber authentication functions.<br>Multiple addresses are needed if multiple RADIUS servers will be configured.<br>RADIUS authentication servers are configured within the source context. Multiple servers can be configured and each assigned a priority. |
| | Shared Secret:<br>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.<br>A shared secret is needed for each configured RADIUS server. |
| | UDP Port Number:<br>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:<br>Specifies the IP address of the RADIUS accounting server that the source context will communicate with to provide subscriber accounting functions.<br>Multiple addresses are needed if multiple RADIUS servers will be configured.<br>RADIUS accounting servers are configured within the source context. Multiple servers can be configured and each assigned a priority. |
| | Shared Secret:<br>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.<br>A shared secret is needed for each configured RADIUS server. |
| | UDP Port Number:<br>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.<br>**NOTE:** For this configuration, the IP context name should be identical to the name of the destination context. |

## Destination Context Configuration

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

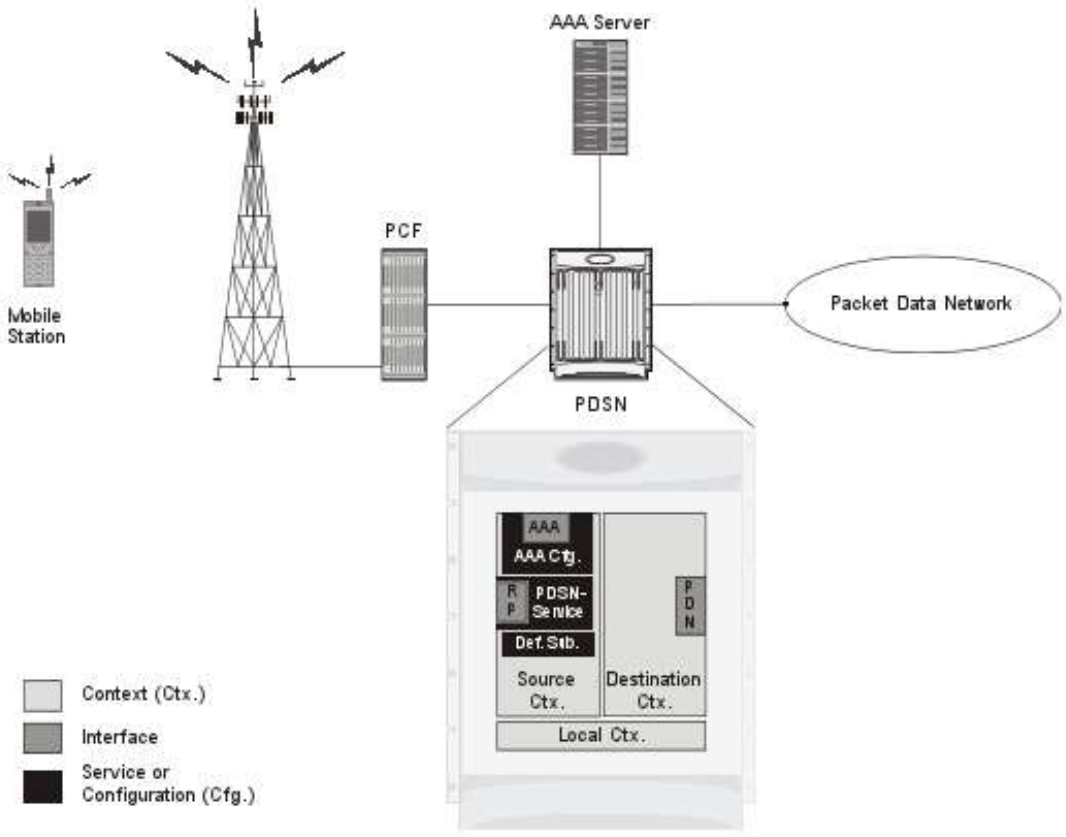
*Table 9. Required Information for Destination Context Configuration*

| Required Information | Description |
|---|---|
| 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.<br>**NOTE:** For this configuration, the destination context name should **not** match the domain name of a specific domain. |
| PDN Interface Configuration | |
| PDN interface name | This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the interface will be recognized by the system.<br>Multiple names are needed if multiple interfaces will be configured.<br>PDN interfaces are configured in the destination context. |
| IP address and subnet | These will be assigned to the PDN interface.<br>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.<br>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.<br>Multiple descriptions will be needed if multiple ports will be used.<br>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.<br>The pool can be configured as public, private, or static. |

# How This Configuration Works

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

*Figure 15. Call Processing Using a Single Source and Destination Context*




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

   For this example, the result of this process is that PDSN service determined that AAA functionality should be provided by the *Source* context.
3. The system communicates with the AAA server specified in the *Source* context's AAA configuration to authenticate the subscriber.
4. Upon successful authentication, the system determines which egress context to use for the subscriber session. This process is described in the How the System Selects Contexts section located in the Understanding the System Operation and Configuration chapter of the *System Administration Guide*.

   The system determines that the egress context is the destination context based on the configuration of either the *Default* subscriber's *ip-context* name or from the *SN-VPN-NAME* or *SN1-VPN-NAME* attributes that is configured in the subscriber's RADIUS profile.
5. Data traffic for the subscriber session is routed through the PDN interface in the *Destination* context.

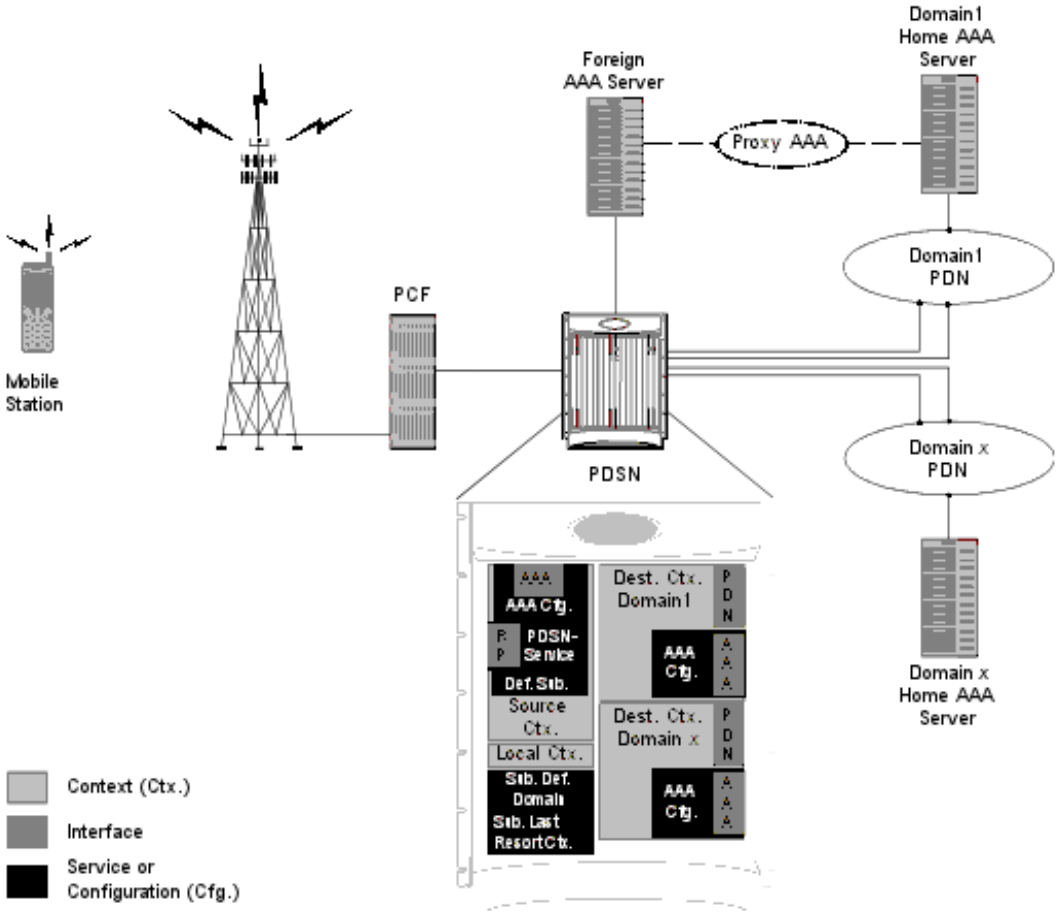
6. Accounting information for the session is sent to the AAA server over the AAA interface.

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

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

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

*Figure 16. Simple IP Support Using a Single Source Context and Multiple Outsourced Destination Contexts*



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

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

# Information Required

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

## Source Context Configuration

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

***Table 10. Required Information for Source Context Configuration***

| Required Information | Description |
|---|---|
| Source context name | This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the source context will be recognized by the system. |
| R-P Interface Configuration | |
| R-P interface name | This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the interface will be recognized by the system.<br>Multiple names are needed if multiple interfaces will be configured.<br>R-P interfaces are configured in the source context. |
| IP address and subnet | These will be assigned to the R-P interface.<br>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.<br>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.<br>Multiple descriptions are needed if multiple ports will be used.<br>Physical ports are configured within the source context and are used to bind logical R-P interfaces. |
| Gateway IP address | Used when configuring static routes from the R-P interface(s) to a specific network. |
| PDSN service Configuration | |

| Required Information | Description |
|---|---|
| PDSN service name | This is an identification string between 1 and 63 characters (alpha and/or numeric) by which the PDSN service will be recognized by the system.<br>Multiple names are needed if multiple PDSN services will be used.<br>PDSN services are configured in the source context. |
| UDP port number for R-P traffic | Specifies the port used by the PDSN service and the PCF for communications. The UDP port number and can be any integer value between 1 and 65535. The default value is 699. |
| Authentication protocols used | Specifies how the system handles authentication: using a protocol (such as CHAP, PAP, or MSCHAP), or not requiring any authentication. |
| Domain alias for NAI-construction | Specifies a context name for the system to use to provide accounting functionality for a subscriber session. This parameter is needed only if the system is configured to support no authentication. |
| Security Parameter Index Information | **PCF IP address:**<br>Specifies the IP address of the PCF that the PDSN service will be communicating with. The PDSN service allows the creation of a security profile that can be associated with a particular PCF.<br>Multiple IP addresses are needed if the PDSN service will be communicating with multiple PCFs. |
| | **Index:**<br>Specifies the shared SPI between the PDSN service and a particular PCF. The SPI can be configured to any integer value between 256 and 4294967295.<br>Multiple SPIs can be configured if the PDSN service is to communicate with multiple PCFs. |
| | **Secret:**<br>Specifies the shared SPI secret between the PDSN service and the PCF. The secret can be between 1 and 127 characters (alpha and/or numeric).<br>An SPI secret is required for each SPI configured. |
| | **Hash-algorithm:**<br>Specifies the algorithm used to hash the SPI and SPI secret. The possible algorithms that can be configured are MD5 per RFC 1321 and keyed-MD5 per RFC 2002. The default is MD5.<br>A hash-algorithm is required for each SPI configured. |
| | **Replay-protection process:**<br>Specifies how protection against replay-attacks is implemented. The possible processes are nonce and timestamp. The default is timestamp with a tolerance of 60 seconds.<br>A replay-protection process is required for each SPI configured. |
| Subscriber session lifetime | Specifies the time in seconds that an A10 connection can exist before its registration is considered expired.<br>The time is expressed in seconds and can be configured to any integer value between 1 and 65534, or the timer can be disabled to set an infinite lifetime. The default value is 1800 seconds. |
| AAA Interface Configuration | |
| AAA interface name | This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the interface will be recognized by the system.<br>Multiple names are needed if multiple interfaces will be configured.<br>AAA interfaces will be configured in the source context. |
| IP address and subnet | These will be assigned to the AAA interface.<br>Multiple addresses and/or subnets are needed if multiple interfaces will be configured. |

| Required Information | Description |
|---|---|
| Physical port number | This specifies the physical port to which the interface will be bound. Ports are identified by the chassis slot number where the line card resides in, followed by the number of the physical connector on the line card. For example, port 17/1 identifies connector number 1 on the card in slot 17.<br>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.<br>Multiple descriptions are needed if multiple ports will be used.<br>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:<br>Specifies the IP address of the RADIUS authentication server the source context will communicate with to provide subscriber authentication functions.<br>Multiple addresses are needed if multiple RADIUS servers will be configured.<br>RADIUS authentication servers are configured within the source context. Multiple servers can be configured and each assigned a priority. |
| | Shared Secret:<br>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.<br>A shared secret is needed for each configured RADIUS server. |
| | UDP Port Number:<br>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:<br>Specifies the IP address of the RADIUS accounting server that the source context will communicate with to provide subscriber accounting functions.<br>Multiple addresses are needed if multiple RADIUS servers will be configured.<br>RADIUS accounting servers are configured within the source context. Multiple servers can be configured and each assigned a priority. |
| | Shared Secret:<br>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.<br>A shared secret is needed for each configured RADIUS server. |
| | UDP Port Number:<br>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 | |

| Required Information | Description |
|---|---|
| "Default" subscriber's IP context name | Specifies the name of the egress context on the system that facilitates the PDN ports.<br>**NOTE:** For this configuration, the IP context name should be identical to the name of the destination context. |

## Destination Context Configuration

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

***Table 11. Required Information for Destination Context Configuration***

| Required Information | Description |
|---|---|
| Destination context name | This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the destination context will be recognized by the system.<br>**NOTE:** For this configuration, the destination context name should **not** match the domain name of a specific domain. |
| PDN Interface Configuration | |
| PDN interface name | This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the interface will be recognized by the system.<br>Multiple names are needed if multiple interfaces will be configured.<br>PDN interfaces are configured in the destination context. |
| IP address and subnet | These will be assigned to the PDN interface.<br>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.<br>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.<br>Multiple descriptions will be needed if multiple ports will be used.<br>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.<br>The pool can be configured as public, private, or static. |
| AAA Interface Configuration | |

| Required Information | Description |
|---|---|
| AAA interface name | This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the interface will be recognized by the system.<br>Multiple names are needed if multiple interfaces will be configured.<br>AAA interfaces will be configured in the source context. |
| IP address and subnet | These will be assigned to the AAA interface.<br>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.<br>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.<br>Multiple descriptions are needed if multiple ports will be used.<br>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:<br>Specifies the IP address of the RADIUS authentication server the source context will communicate with to provide subscriber authentication functions.<br>Multiple addresses are needed if multiple RADIUS servers will be configured.<br>RADIUS authentication servers are configured within the source context. Multiple servers can be configured and each assigned a priority. |
| | Shared Secret:<br>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.<br>A shared secret is needed for each configured RADIUS server. |
| | UDP Port Number:<br>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:<br>Specifies the IP address of the RADIUS accounting server that the source context will communicate with to provide subscriber accounting functions.<br>Multiple addresses are needed if multiple RADIUS servers will be configured.<br>RADIUS accounting servers are configured within the source context. Multiple servers can be configured and each assigned a priority. |
| | Shared Secret:<br>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.<br>A shared secret is needed for each configured RADIUS server. |
| | UDP Port Number:<br>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. |

| Required Information | Description |
|---|---|
| RADIUS NAS IP address | Specifies the IP address of the source context's AAA interface. A secondary IP address interface can optionally be configured. |

## System-Level AAA Configuration

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

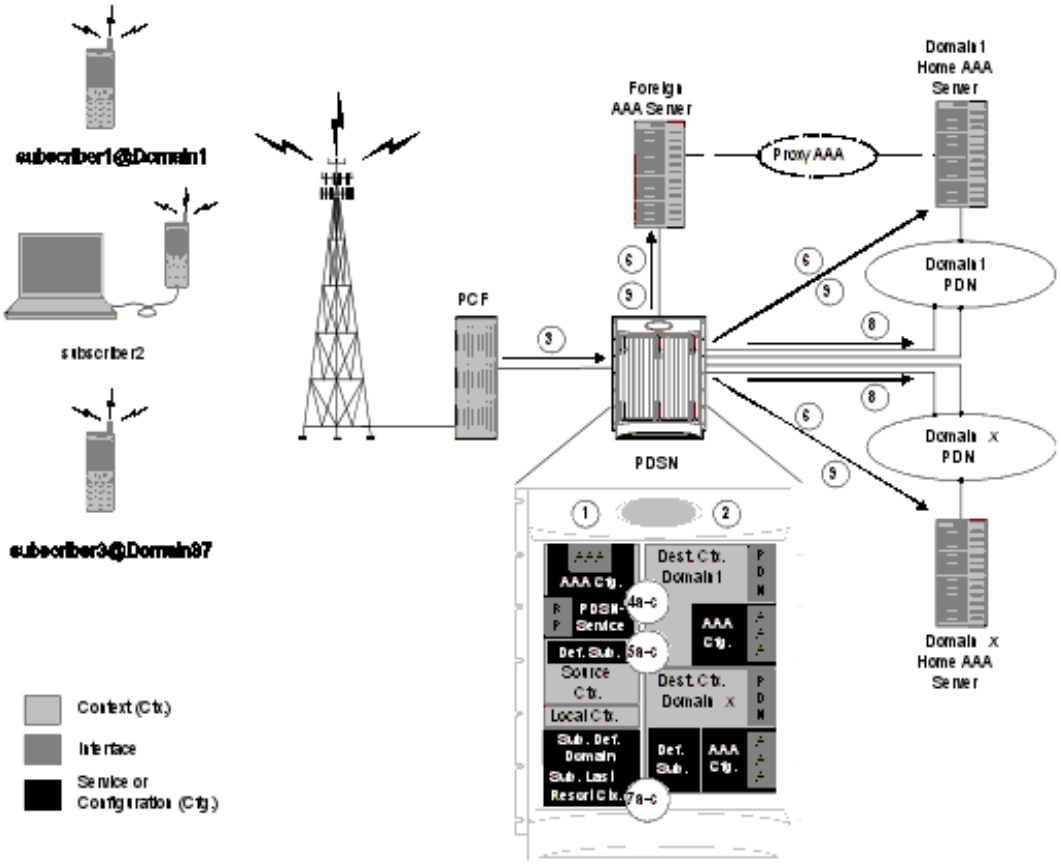
*Table 12. Required Information for System-Level AAA Configuration*

| Required Information | Description |
|---|---|
| Subscriber default domain name | Specifies the name of a context that can provide AAA functions in the event that the domain-part of the username is missing or poorly formed.<br>This parameter will be applied to all subscribers if their domain can not be determined from their username regardless of what domain they are trying to access.<br>**NOTE:** The default domain name can be the same as the source context. |
| Subscriber Last-resort context | Specifies the name of a context that can provide AAA functions in the event that the domain-part of the username was present but does not match the name of a configured destination context.<br>This parameter will be applied to all subscribers if their specified domain does not match a configured destination context regardless of what domain they are trying to access.<br>**NOTE:** The last-resort context name can be the same as the source context. |
| Subscriber username format | Specifies the format of subscriber usernames as to whether or not the username or domain is specified first and the character that separates them. The possible separator characters are:<br>• @<br>• %<br>• -<br>• \<br>• #<br>• /<br>Up to six username formats can be specified. The default is *username @.*<br>**NOTE:** The username string is searched from right to left for the separator character. Therefore, if there is one or more separator characters in the string, only the first one that is recognized is considered the actual separator. For example, if the default username format was used, then for the username string *user1@enterprise@isp1*, the system resolves to the username *user1@enterprise* with domain *isp1*. |

# How This Configuration Works

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

*Figure 17. Call Processing Using a Single Source and Destination Context*




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

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

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

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

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

3. The system communicates with the AAA server specified in the *Source* context's AAA configuration to authenticate the subscriber.

4. Upon successful authentication, the system determines which egress context to use for the subscriber session. This process is described in the How the System Selects Contexts section located in the Understanding the System Operation and Configuration chapter of the *System Administration Guide*.

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

5. Data traffic for the subscriber session is routed through the PDN interface in the *Destination* context.

6. Accounting information for the session is sent to the AAA server over the AAA interface.

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

- For subscriber1, the PDSN service determines that a domain name is present and is Domain1.
- For subscriber2, the PDSN service determines that no domain name is present.
- For subscriber3, the PDSN service determines that a domain name is present and is Domain37.

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

# Chapter 4
# Mobile IP Configuration Examples

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

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

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

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

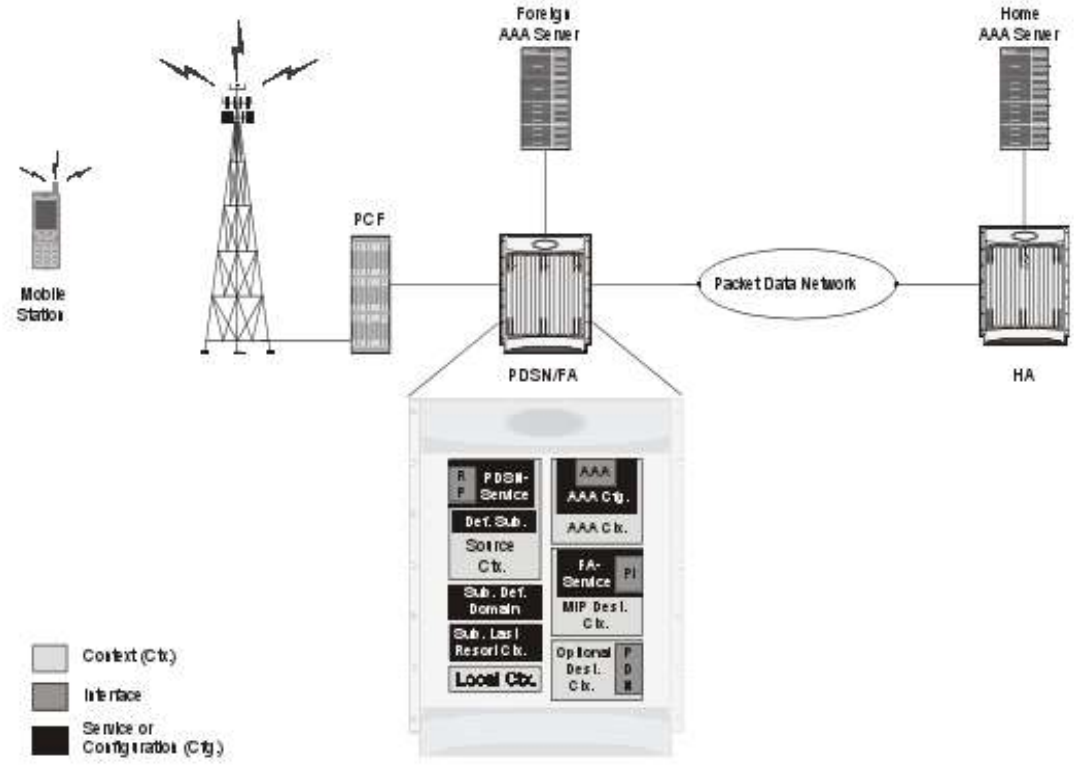
The system's PDSN/FA configuration for Mobile IP applications is best addressed with three contexts (one source, one AAA, and one Mobile IP destination) configured as shown in the figure below.

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

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

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

*Figure 18. Mobile IP Support using the system as a PDSN/FA*




# Information Required

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

## Source Context Configuration

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

*Table 13. Required Information for Source Context Configuration*

| Required Information | Description |
|---|---|
| Source context name | This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the source context will be recognized by the system. |

| Required Information | Description |
|---|---|
| R-P Interface Configuration | |
| R-P interface name | This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the interface will be recognized by the system.<br>Multiple names are needed if multiple interfaces will be configured.<br>R-P interfaces are configured in the source context. |
| IP address and subnet | These will be assigned to the R-P interface.<br>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.<br>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.<br>Multiple descriptions are needed if using multiple ports.<br>Physical ports are configured within the source context and are used to bind logical R-P interfaces. |
| Gateway IP address | Used when configuring static routes from the R-P interface(s) to a specific network. |
| PDSN service Configuration | |
| PDSN service name | This is an identification string between 1 and 63 characters (alpha and/or numeric) by which the PDSN service will be recognized by the system.<br>Multiple names are needed if using multiple PDSN services.<br>PDSN services are configured in the source context. |
| UDP port number for R-P traffic | Specifies the port used by the PDSN service and the PCF for communications. The UDP port number and can be any integer value between 1 and 65535. The default value is 699. |
| Authentication protocols used | Specifies how the system handles authentication: using a protocol (such as CHAP, PAP, or MSCHAP), or not requiring any authentication. |
| Domain alias for NAI-construction | Specifies a context name for the system to use to provide accounting functionality for a subscriber session. This parameter is needed only if the system is configured to support no authentication. |
| Security Parameter Index Information | **PCF IP address:**<br>Specifies the IP address of the PCF that the PDSN service will be communicating with. The PDSN service allows the creation of a security profile that can be associated with a particular PCF.<br>Multiple IP addresses are needed if the PDSN service is to communicate with multiple PCFs. |
| | **Index:**<br>Specifies the shared SPI between the PDSN service and a particular PCF. The SPI can be configured to any integer value between 256 and 4294967295.<br>Configure multiple SPIs if the PDSN service is to communicate with multiple PCFs. |
| | **Secret:**<br>Specifies the shared SPI secret between the PDSN service and the PCF. The secret can be between 1 and 127 characters (alpha and/or numeric).<br>An SPI secret is required for each SPI configured. |
| | **Hash-algorithm**:<br>Specifies the algorithm used to hash the SPI and SPI secret. The possible algorithms that can be configured are MD5 per RFC 1321 and keyed-MD5 per RFC 2002. The default is MD5.<br>A hash-algorithm is required for each SPI configured. |

| Required Information | Description |
|---|---|
| | **Replay-protection process:**<br>Specifies how protection against replay-attacks is implemented. The possible processes are nonce and timestamp. The default is timestamp with a tolerance of 60 seconds.<br>A replay-protection process is required for each SPI configured. |
| Subscriber session lifetime | Specifies the time in seconds that an A10 connection can exist before its registration is considered expired.<br>The time is expressed in seconds and can be configured to any integer value between 1 and 65534, or the timer can be disabled to set an infinite lifetime. The default value is 1800 seconds. |
| Mobile IP FA context name | Specifies the name of the context in which the FA service is configured. |

## AAA Context Configuration

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

***Table 14. Required Information for AAA Context Configuration***

| Required Information | Description |
|---|---|
| AAA context name | This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the AAA context will be recognized by the system.<br>**NOTE:** If a separate system is used to provide HA functionality, the AAA context name should match the name of the context in which the AAA functionality is configured on the HA machine. |
| 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.<br>Multiple names are needed if multiple interfaces will be configured.<br>AAA interfaces will be configured in the source context. |
| IP address and subnet | These will be assigned to the AAA interface.<br>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.<br>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.<br>Multiple descriptions are needed if multiple ports will be used.<br>Physical ports are configured within the destination context and are used to bind logical AAA interfaces. |
| Gateway IP address(es) | Used when configuring static routes from the AAA interface(s) to a specific network. |
| Foreign RADIUS Server Configuration | |

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

## Mobile IP Destination Context Configuration

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

*Table 15. Required Information for Destination Context Configuration*

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

| Required Information | Description |
|---|---|
| Mobile IP destination context name | This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the destination context will be recognized by the system.<br>**NOTE:** For this configuration, the destination context name should **not** match the domain name of a specific domain. It should, however, match the name of the context in which the HA service is configured if a separate system is used to provide HA functionality. |
| Pi Interface Configuration | |
| Pi 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.<br>Multiple names are needed if multiple interfaces will be configured.<br>Pi interfaces are configured in the destination context. |
| IP address and subnet | These will be assigned to the Pi interface.<br>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.<br>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.<br>Multiple descriptions are needed if multiple ports will be used.<br>Physical ports are configured within the destination context and are used to bind logical Pi interfaces. |
| Gateway IP address(es) | Used when configuring static routes from the Pi interface(s) to a specific network. |
| FA Service Configuration | |
| FA service name | This is an identification string between 1 and 63 characters (alpha and/or numeric) by which the FA service will be recognized by the system.<br>Multiple names are needed if multiple FA services will be used.<br>FA services are configured in the destination context. |
| UDP port number for Mobile IP traffic | Specifies the port used by the FA service and the HA for communications. The UDP port number can be any integer value between 1 and 65535. The default value is 434. |
| Security Parameter Index (indices) Information | HA IP address:<br>Specifies the IP address of the HAs with which the FA service communicates. The FA service allows the creation of a security profile that can be associated with a particular HA. |
| | Index:<br>Specifies the shared SPI between the FA service and a particular HA. The SPI can be configured to any integer value between 256 and 4294967295. Multiple SPIs can be configured if the FA service is to communicate with multiple HAs. |
| | Secrets:<br>Specifies the shared SPI secret between the FA service and the HA. The secret can be between 1 and 127 characters (alpha and/or numeric).An SPI secret is required for each SPI configured. |
| | Hash-algorithm:<br>Specifies the algorithm used to hash the SPI and SPI secret. The possible algorithms that can be configured are MD5 per RFC 1321 and keyed-MD5 per RFC 2002. The default is hmac-md5.<br>A hash-algorithm is required for each SPI configured. |

| Required Information | Description |
|---|---|
| FA agent advertisement lifetime | Specifies the time (in seconds) that an FA agent advertisement remains valid in the absence of further advertisements.<br>The time can be configured to any integer value between 1 and 65535. The default is 9000. |
| Number of allowable unanswered FA advertisements | Specifies the number of unanswered agent advertisements that the FA service will allow during call setup before it will reject the session.<br>The number can be any integer value between 1 and 65535. The default is 5. |
| Maximum mobile-requested registration lifetime allowed | Specifies the longest registration lifetime that the FA service will allow in any Registration Request message from the mobile node.<br>The lifetime is expressed in seconds and can be configured between 1 and 65534. An infinite registration lifetime can be configured by disabling the timer. The default is 600 seconds. |
| Registration reply timeout | Specifies the amount of time that the FA service will wait for a Registration Reply from an HA.<br>The time is measured in seconds and can be configured to any integer value between 1 and 65535. The default is 7. |
| Number of simultaneous registrations | Specifies the number of simultaneous Mobile IP sessions that will be supported for a single subscriber.<br>The maximum number of sessions is 3. The default is 1.<br>NOTE: The system will only support multiple Mobile IP sessions per subscriber if the subscriber's mobile node has a static IP address. |
| Mobile node re-registration requirements | Specifies how the system should handle authentication for mobile node re-registrations.<br>The FA service can be configured to always require authentication or not. If not, the initial registration and de-registration will still be handled normally. |

# System-Level AAA Configuration

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

*Table 16. Required Information for System-Level AAA Configuration*

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

| Required Information | Description |
|---|---|
| Subscriber username format | Specifies the format of subscriber usernames as to whether or not the username or domain is specified first and the character that separates them. The possible separator characters are:<br>• @<br>• %<br>• -<br>• \<br>• #<br>• /<br>Up to six username formats can be specified. The default is *username @*.<br>**NOTE:** The username string is searched from right to left for the separator character. Therefore, if there is one or more separator characters in the string , only the first one that is recognized is considered the actual separator. For example, if the default username format was used, then for the username string *user1@enterprise@isp1*, the system resolves to the username *user1@enterprise* with domain *isp1*. |

## Optional Destination Context

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

***Table 17. Required Information for Destination Context Configuration***

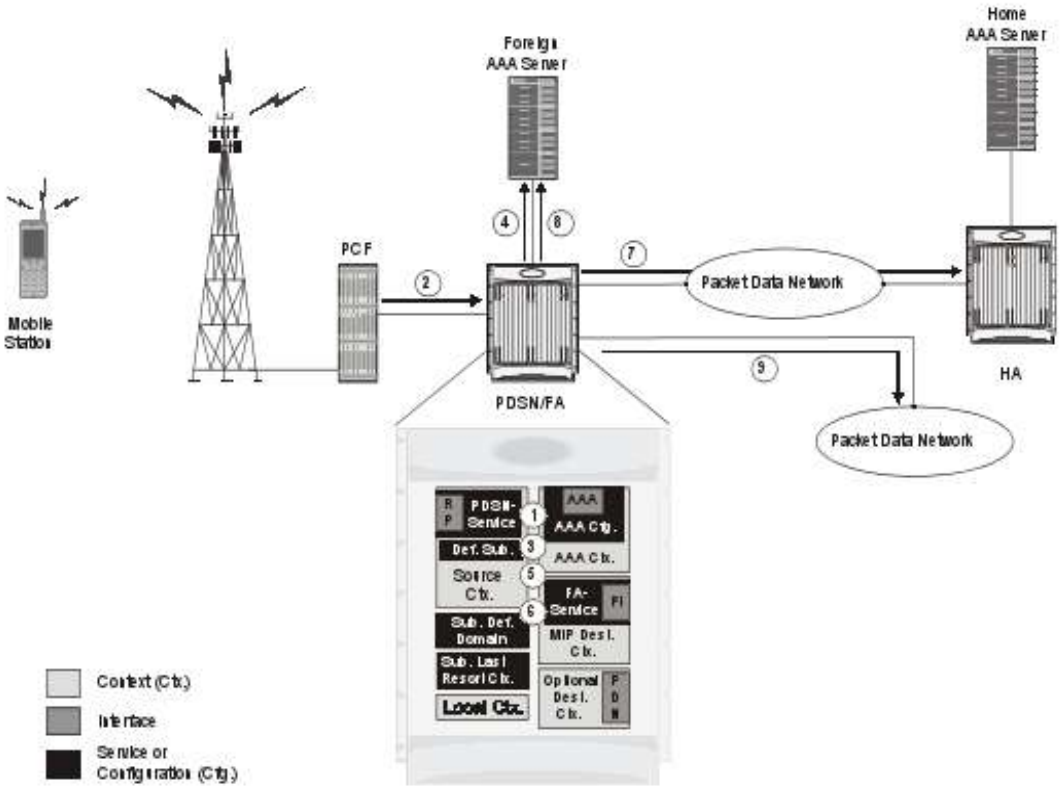
| Required Information | Description |
|---|---|
| Destination context name | This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the destination context will be recognized by the system.<br>**NOTE**: For this configuration, the destination context name should not match the domain name of a specific domain. |
| PDN Interface Configuration | |
| PDN interface name | This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the interface will be recognized by the system.<br>Multiple names are needed if multiple interfaces will be configured.<br>PDN interfaces are configured in the destination context. |
| IP address and subnet | These will be assigned to the PDN interface.<br>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.<br>A single physical port can facilitate multiple interfaces. |

| Required Information | Description |
|---|---|
| Physical port description | This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the physical port will be recognized by the system.<br>Multiple descriptions are needed if multiple ports will be used.<br>Physical ports are configured within the destination context and are used to bind logical PDN interfaces. |
| Gateway IP address(es) | Used when configuring static routes from the PDN interface(s) to a specific network. |
| IP Address Pool Configuration | |
| IP address pool name | Each IP address pool is identified by a name. The pool name can be between 1 and 31 alpha and/or numeric characters and is case sensitive.<br>IP address pools are configured in the destination context(s). Multiple address pools can be configured within a single context. |
| IP pool addresses | An initial address and a subnet, or a starting address and an ending address, are required for each configured pool. The pool will then consist of every possible address within the subnet , or all addresses from the starting address to the ending address.<br>The pool can be configured as public, private, or static.<br>If this IP pool is being used for Interchassis Session Recovery, it must be a static and srp-activated. |

# How This Configuration Works

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

*Figure 19. Call Processing When Using the system as a PDSN/FA*




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

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

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

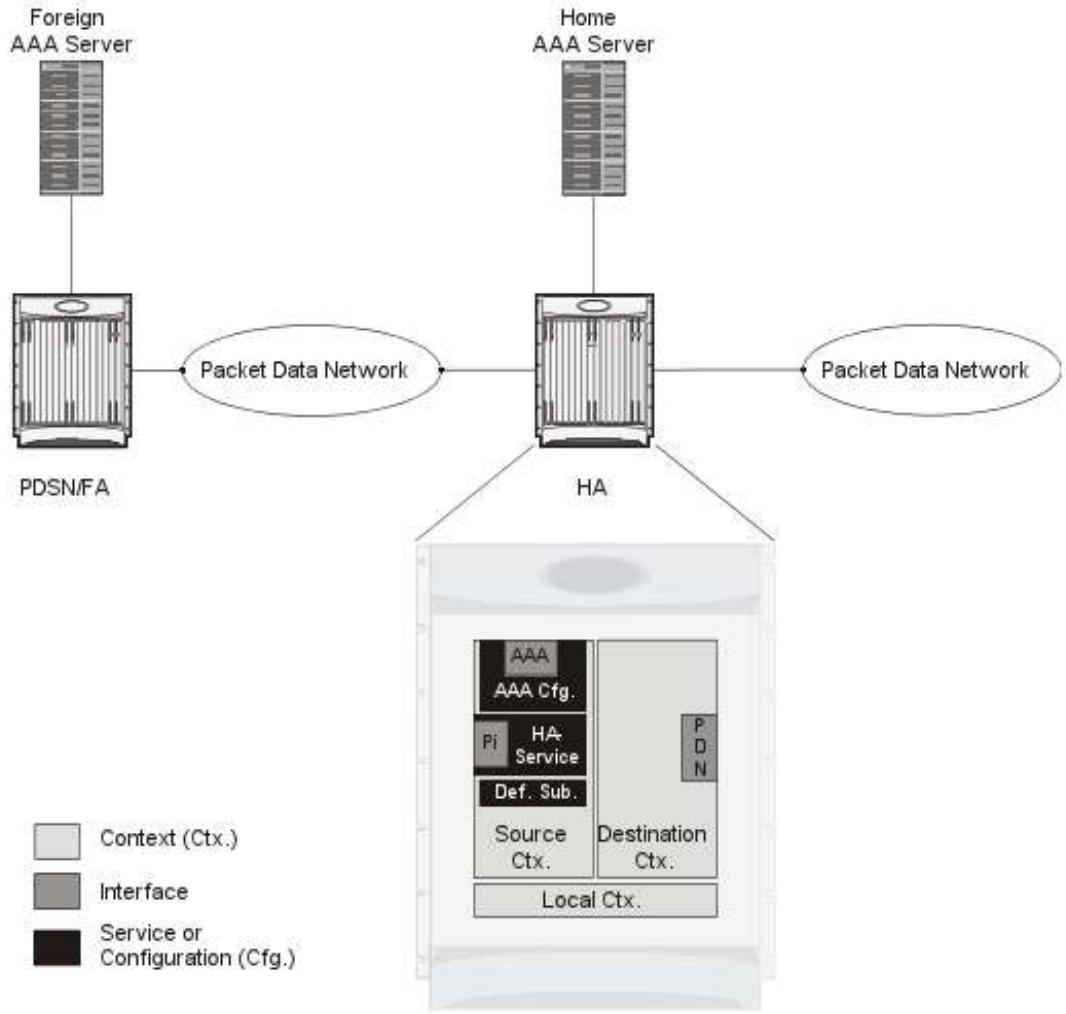
7. The FA service then establishes a connection to the specified HA over the Pi interface.
8. Accounting messages for the session are sent to the Foreign AAA server over the AAA interface.
9. If reverse tunneling is disabled, then subscriber data traffic would have been routed over the PDN interface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

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

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

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

***Figure 20. Mobile IP Support Using the system as an HA***



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

# Information Required

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

## Source Context Configuration

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

***Table 18. Required Information for Source Context Configuration***

| Required Information | Description |
|---|---|
| Source context name | 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.<br>NOTE: The name of the source context should be the same as the name of the context in which the FA-context is configured if a separate system is being used to provide PDSN/FA functionality. |
| Pi Interface Configuration | |
| Pi 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.<br>Multiple names are needed if multiple interfaces will be configured.<br>Pi interfaces are configured in the source context.<br>If this interface is being used for Interchassis Session Recovery, you must specify a loopback interface type after the interface_name. |
| IP address and subnet | These will be assigned to the Pi interface.<br>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.<br>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.<br>Multiple descriptions are needed if using multiple ports.<br>Physical ports are configured within the source context and are used to bind logical Pi interfaces. |
| Gateway IP address | Used when configuring static routes from the R-P interface(s) to a specific network. |
| HA service Configuration | |

| Required Information | Description |
|---|---|
| HA service name | This is an identification string between 1 and 63 characters (alpha and/or numeric) by which the HA service will be recognized by the system.<br>Multiple names are needed if multiple HA services will be used.<br>HA services are configured in the destination context. |
| UDP port number for Mobile IP traffic | Specifies the port used by the HA service and the FA for communications. The UDP port number can be any integer value between 1 and 65535. The default value is 434. |
| Mobile node re-registration requirements | Specifies how the system should handle authentication for mobile node re-registrations.<br>The HA service can be configured as follows:Always require authentication<br>Never require authentication (NOTE: the initial registration and de-registration will still be handled normally)<br>Never look for mn-aaa extension<br>Not require authentication but will authenticate if mn-aaa extension present |
| FA-to-HA Security Parameter Index Information | FA IP address:<br>The HA service allows the creation of a security profile that can be associated with a particular FA. This specifies the IP address of the FA that the HA service will be communicating with.<br>Multiple FA addresses are needed if the HA will be communicating with multiple FAs. |
| | Index:<br>Specifies the shared SPI between the HA service and a particular FA. The SPI can be configured to any integer value between 256 and 4294967295.<br>Multiple SPIs can be configured if the HA service is to communicate with multiple FAs. |
| | Secret:<br>Specifies the shared SPI secret between the HA service and the FA. The secret can be between 1 and 127 characters (alpha and/or numeric).<br>An SPI secret is required for each SPI configured. |
| | Hash-algorithm:<br>Specifies the algorithm used to hash the SPI and SPI secret. The possible algorithms that can be configured are MD5 per RFC 1321 and keyed-MD5 per RFC 2002. The default algorithm is hmac-md5.<br>A hash-algorithm is required for each SPI configured. |
| Mobile Node Security Parameter Index Information | Index:<br>Specifies the shared SPI between the HA service and the mobile node(s). The SPI can be configured to any integer value between 256 and 4294967295.<br>Multiple SPIs can be configured if the HA service is to communicate with multiple mobile nodes. |
| | Secret(s):<br>Specifies the shared SPI secret between the HA service and the mobile node. The secret can be between 1 and 127 characters (alpha and/or numeric).<br>An SPI secret is required for each SPI configured. |
| | Hash-algorithm:<br>Specifies the algorithm used to hash the SPI and SPI secret. The possible algorithms that can be configured are MD5 per RFC 1321 and keyed-MD5 per RFC 2002. The default algorithm is hmac-md5.<br>A hash-algorithm is required for each SPI configured. |
| | Replay-protection process:<br>Specifies how protection against replay-attacks is implemented. The possible processes are nonce and timestamp. The default is timestamp with a tolerance of 60 seconds.<br>A replay-protection process is required for each mobile node-to-HA SPI configured. |

| Required Information | Description |
|---|---|
| Maximum registration lifetime | Specifies the longest registration lifetime that the HA service will allow in any Registration Request message from the mobile node.<br>The time is measured in seconds and can be configured to any integer value between 1 and 65534. An infinite registration lifetime can also be configured by disabling the timer. The default is 600. |
| Maximum number of simultaneous bindings | Specifies the maximum number of "care-of" addresses that can simultaneously be bound for the same user as identified by NAI and Home address.<br>The number can be configured to any integer value between 1 and 5. The default is 3. |
| AAA Interface Configuration | |
| AAA interface name | This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the interface will be recognized by the system.<br>Multiple names are needed if multiple interfaces will be configured.<br>AAA interfaces will be configured in the source context. |
| IP address and subnet | These will be assigned to the AAA interface.<br>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.<br>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.<br>Multiple descriptions are needed if multiple ports will be used.<br>Physical ports are configured within the source context and are used to bind logical AAA interfaces. |
| Gateway IP address | Used when configuring static routes from the AAA interface(s) to a specific network. |
| Home RADIUS Server Configuration | |
| Home RADIUS Authentication server | IP Address:Specifies the IP address of the home RADIUS authentication server the source context will communicate with to provide subscriber authentication functions.<br>Multiple addresses are needed if multiple RADIUS servers will be configured.<br>Home RADIUS authentication servers are configured within the source context. Multiple servers can be configured and each assigned a priority. |
| | Shared Secret:<br>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.<br>A shared secret is needed for each configured RADIUS server. |
| | UDP Port Number:<br>Specifies the port used by the source context and the home RADIUS authentication server for communications. The UDP port number can be any integer value between 1 and 65535. The default value is 1812. |
| Home RADIUS Accounting server | IP Address:<br>Specifies the IP address of the home RADIUS accounting server that the source context will communicate with to provide subscriber accounting functions.<br>Multiple addresses are needed if multiple RADIUS servers will be configured.Home RADIUS accounting servers are configured within the source context. Multiple servers can be configured and each assigned a priority. |

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

*Table 19. Required Information for Destination Context Configuration*

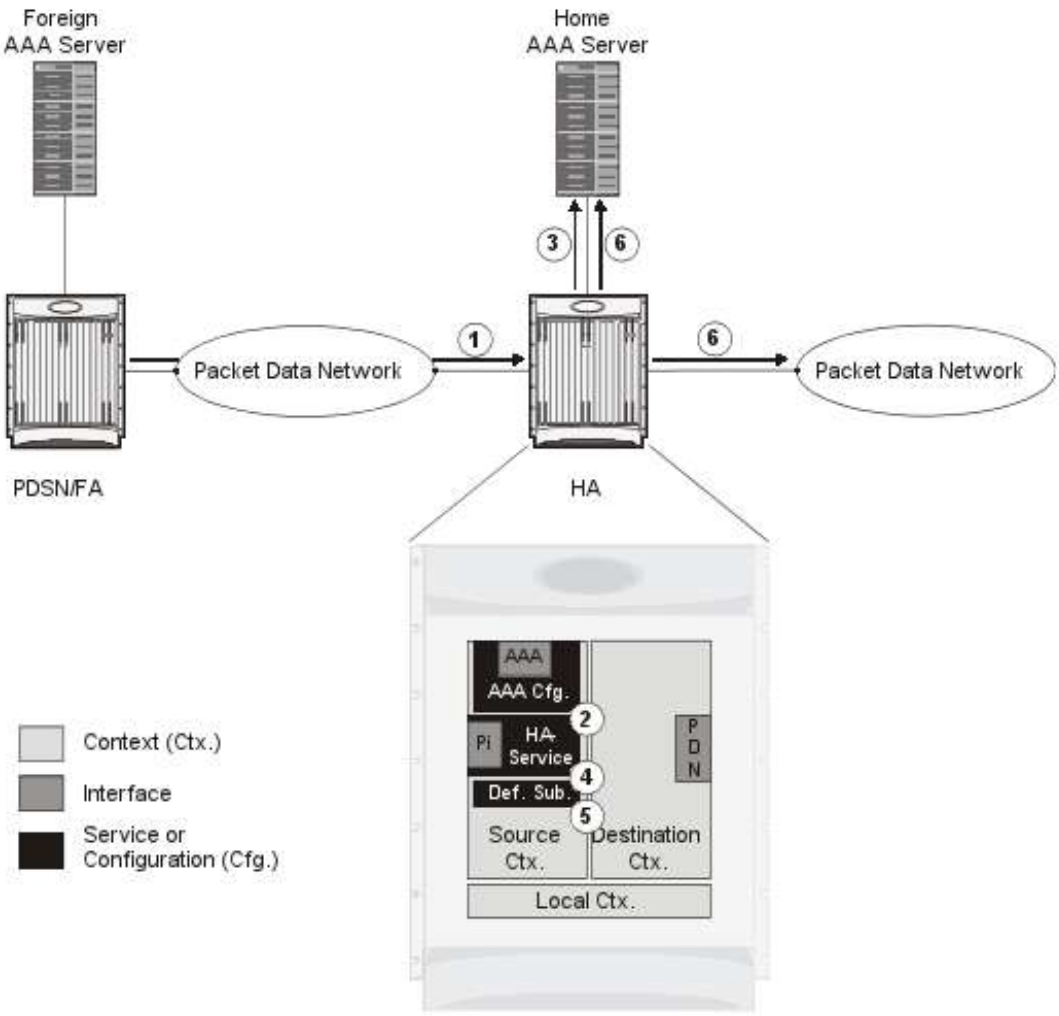
| Required Information | Description |
|---|---|
| Destination context name | This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the destination context will be recognized by the system.<br>**NOTE:** For this configuration, the destination context name should **not** match the domain name of a specific domain. |
| PDN Interface Configuration | |
| PDN interface name | This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the interface will be recognized by the system.<br>Multiple names are needed if multiple interfaces will be configured.<br>PDN interfaces are configured in the destination context. |
| IP address and subnet | These will be assigned to the PDN interface.<br>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.<br>A single physical port can facilitate multiple interfaces. |

| Required Information | Description |
|---|---|
| Physical port description | This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the physical port will be recognized by the system.<br>Multiple descriptions are needed if multiple ports will be used.<br>Physical ports are configured within the destination context and are used to bind logical PDN interfaces. |
| Gateway IP address(es) | Used when configuring static routes from the PDN interface(s) to a specific network. |
| IP Address Pool Configuration | |
| IP address pool name | Each IP address pool is identified by a name. The pool name can be between 1 and 31 alpha and/or numeric characters and is case sensitive.<br>IP address pools are configured in the destination context(s). Multiple address pools can be configured within a single context. |
| IP pool addresses | An initial address and a subnet, or a starting address and an ending address, are required for each configured pool. The pool will then consist of every possible address within the subnet , or all addresses from the starting address to the ending address.<br>The pool can be configured as public, private, or static.<br>If this IP pool is being used for Interchassis Session Recovery, it must be a static and srp-activated. |

# How This Configuration Works

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

*Figure 21. Call Processing When Using the system as a PDSN/FA*




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

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

4. The system then communicates with the foreign AAA server specified in the AAA context's AAA configuration to authenticate the subscriber.
5. Upon successful authentication, the PDSN service determines the IP address of the subscriber's HA using either an attribute returned in the Access Accept message, or the address specified by the mobile.

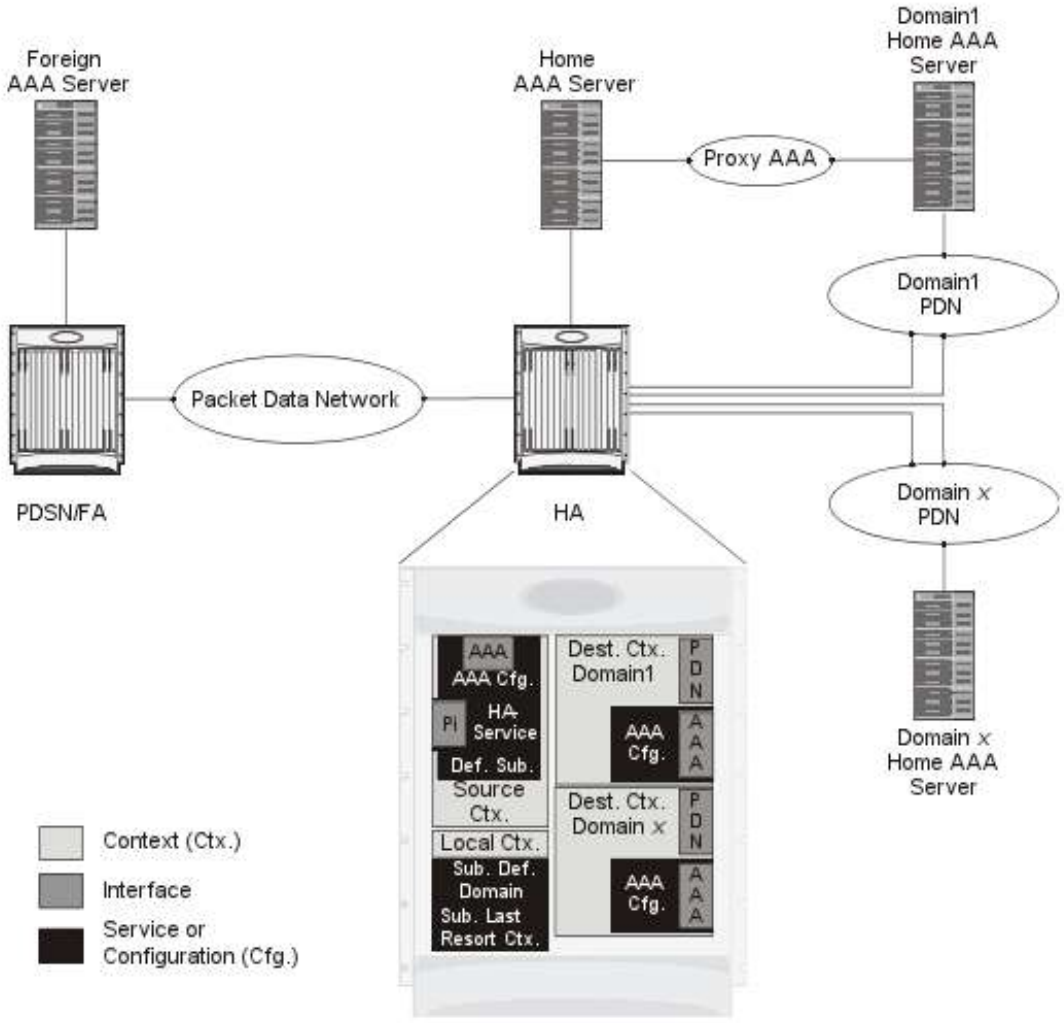
   The PDSN service uses the Mobile IP FA context name to determine what destination context is facilitating the FA service. In this example, it determines that it must use the *MIP Destination* context.
6. The PDSN service passes the HA IP address to the FA service.
7. The FA service then establishes a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

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

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

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

***Figure 22. The system as an HA Using a Single Source Context and Multiple Outsourced Destination Contexts***



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

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

# Information Required

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

## Source Context Configuration

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

***Table 20. Required Information for Source Context Configuration***

| Required Information | Description |
|---|---|
| Source context name | This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the source context will be recognized by the system. |
| Pi Interface Configuration | |
| Piinterface name | This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the interface will be recognized by the system.<br>Multiple names are needed if multiple interfaces will be configured.<br>Pi interfaces are configured in the source context. |
| IP address and subnet | These will be assigned to the Pi interface.<br>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.<br>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.<br>Multiple descriptions are needed if using multiple ports.<br>Physical ports are configured within the source context and are used to bind logical Pi interfaces. |
| Gateway IP address | Used when configuring static routes from the Pi interface(s) to a specific network. |
| HA service Configuration | |

| Required Information | Description |
|---|---|
| HA service name | This is an identification string between 1 and 63 characters (alpha and/or numeric) by which the PDSN service will be recognized by the system.<br>Multiple names are needed if using multiple HA services.<br>HA services are configured in the source context. |
| UDP port number for Mobile IP traffic | Specifies the port used by the HA service and the FA for communications. The UDP port number can be any integer value between 1 and 65535. The default value is 434. |
| Mobile node re-registration requirements | Specifies how the system should handle authentication for mobile node re-registrations.<br>The HA service can be configured as follows:<br>Always require authentication<br>Never require authentication (NOTE: the initial registration and de-registration will still be handled normally)<br>Never look for mn-aaa extension<br>Not require authentication but will authenticate if mn-aaa extension present |
| FA-to-HA Security Parameter Index Information | **FA IP address:**<br>The HA service allows the creation of a security profile that can be associated with a particular FA. This specifies the IP address of the FA that the HA service will be communicating with.<br>Multiple FA addresses are needed if the HA will be communicating with multiple FAs. |
| | **Index**:<br>Specifies the shared SPI between the HA service and a particular FA. The SPI can be configured to any integer value between 256 and 4294967295.<br>Multiple SPIs can be configured if the HA service is to communicate with multiple FAs. |
| | **Secret**:<br>Specifies the shared SPI secret between the HA service and the FA. The secret can be between 1 and 127 characters (alpha and/or numeric).<br>An SPI secret is required for each SPI configured. |
| | **Hash-algorithm**:<br>Specifies the algorithm used to hash the SPI and SPI secret. The possible algorithms that can be configured are MD5 per RFC 1321 and keyed-MD5 per RFC 2002. The default algorithm is hmac-md5.<br>A hash-algorithm is required for each SPI configured. |
| Mobile Node Security Parameter Index Information | **Index**:<br>Specifies the shared SPI between the HA service and the mobile node(s). The SPI can be configured to any integer value between 256 and 4294967295.<br>Multiple SPIs can be configured if the HA service is to communicate with multiple mobile nodes. |
| | **Secret(s):**<br>Specifies the shared SPI secret between the HA service and the mobile node. The secret can be between 1 and 127 characters (alpha and/or numeric).<br>An SPI secret is required for each SPI configured. |
| | **Hash-algorithm**:<br>Specifies the algorithm used to hash the SPI and SPI secret. The possible algorithms that can be configured are MD5 per RFC 1321 and keyed-MD5 per RFC 2002. The default algorithm is hmac-md5.<br>A hash-algorithm is required for each SPI configured. |

| Required Information | Description |
|---|---|
| | **Replay-protection process**:<br>Specifies how protection against replay-attacks is implemented. The possible processes are nonce and timestamp. The default is timestamp with a tolerance of 60 seconds.<br>A replay-protection process is required for each mobile node-to-HA SPI configured. |
| Maximum registration lifetime | Specifies the longest registration lifetime that the HA service will allow in any Registration Request message from the mobile node.<br>The time is measured in seconds and can be configured to any integer value between 1 and 65535. An infinite registration lifetime can also be configured by disabling the timer. The default is 600. |
| Maximum number of simultaneous bindings | Specifies the maximum number of "care-of" addresses that can simultaneously be bound for the same user as identified by NAI and Home address.<br>The number can be configured to any integer value between 1 and 5. The default is 3. |
| AAA Interface Configuration | |
| AAA interface name | This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the interface will be recognized by the system.<br>Multiple names are needed if multiple interfaces will be configured.<br>AAA interfaces will be configured in the source context. |
| IP address and subnet | These will be assigned to the AAA interface.<br>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.<br>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.<br>Multiple descriptions are needed if multiple ports will be used.<br>Physical ports are configured within the source context and are used to bind logical AAA interfaces. |
| Gateway IP address | Used when configuring static routes from the AAA interface(s) to a specific network. |
| Home RADIUS Server Configuration | |
| Home RADIUS Authentication server | IP Address:<br>Specifies the IP address of the home RADIUS authentication server the source context will communicate with to provide subscriber authentication functions.<br>Multiple addresses are needed if multiple RADIUS servers will be configured.<br>Home RADIUS authentication servers are configured within the source context. Multiple servers can be configured and each assigned a priority. |
| | Shared Secret:<br>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.<br>A shared secret is needed for each configured RADIUS server. |
| | UDP Port Number:<br>Specifies the port used by the source context and the home RADIUS authentication server for communications. The UDP port number can be any integer value between 1 and 65535. The default value is 1812. |

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

***Table 21. 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.<br>**NOTE:** For this configuration, the destination context name should **not** match the domain name of a specific domain. |
| PDN Interface Configuration | |
| PDN interface name | This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the interface will be recognized by the system.<br>Multiple names are needed if multiple interfaces will be configured.<br>PDN interfaces are configured in the destination context. |

| Required Information | Description |
|---|---|
| IP address and subnet | These will be assigned to the PDN interface.<br>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.<br>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.<br>Multiple descriptions are needed if multiple ports will be used.<br>Physical ports are configured within the destination context and are used to bind logical PDN interfaces. |
| Gateway IP address(es) | Used when configuring static routes from the PDN interface(s) to a specific network. |
| IP Address Pool Configuration | |
| IP address pool name | Each IP address pool is identified by a name. The pool name can be between 1 and 31 alpha and/or numeric characters and is case sensitive.<br>IP address pools are configured in the destination context(s). Multiple address pools can be configured within a single context. |
| IP pool addresses | An initial address and a subnet, or a starting address and an ending address, are required for each configured pool. The pool will then consist of every possible address within the subnet , or all addresses from the starting address to the ending address.<br>The pool can be configured as public, private, or static.<br>If this IP pool is being used for Interchassis Session Recovery, it must be a static and srp-activated. |
| AAA Interface Configuration | |
| AAA interface name | This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the interface will be recognized by the system.<br>Multiple names are needed if multiple interfaces will be configured.<br>AAA interfaces will be configured in the source context. |
| IP address and subnet | These will be assigned to the AAA interface.<br>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.<br>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.<br>Multiple descriptions are needed if multiple ports will be used.<br>Physical ports are configured within the source context and are used to bind logical AAA interfaces. |
| Gateway IP address | Used when configuring static routes from the AAA interface(s) to a specific network. |
| Home RADIUS Server Configuration | |

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

## System-Level AAA Configuration

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

*Table 22. Required Information for System-Level AAA Configuration*

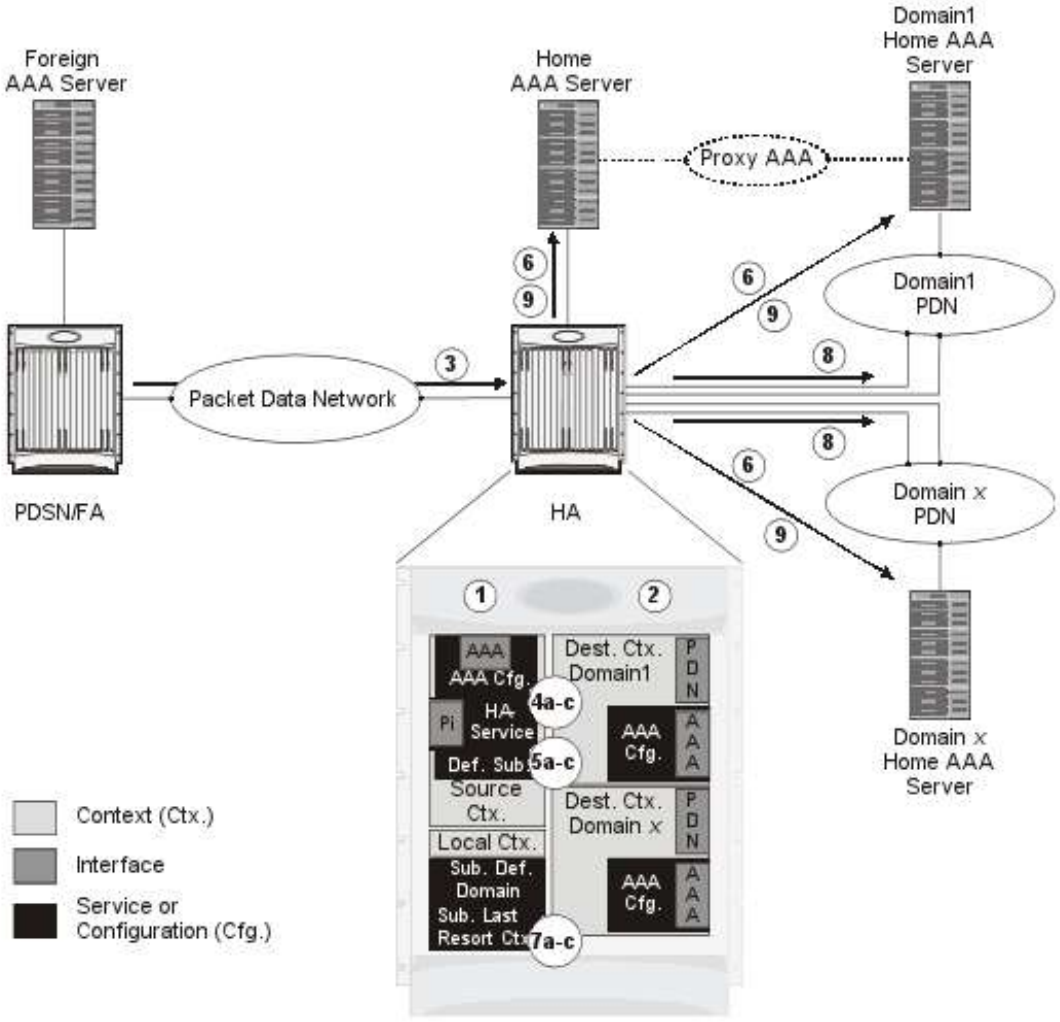
| Required Information | Description |
|---|---|

| Required Information | Description |
|---|---|
| Subscriber default domain name | Specifies the name of a context that can provide AAA functions in the event that the domain-part of the username is missing or poorly formed.<br>This parameter will be applied to all subscribers if their domain can not be determined from their username regardless of what domain they are trying to access.<br>**NOTE:** The default domain name can be the same as the source context. |
| Subscriber Last-resort context | Specifies the name of a context that can provide AAA functions in the event that the domain-part of the username was present but does not match the name of a configured destination context.<br>This parameter will be applied to all subscribers if their specified domain does not match a configured destination context regardless of what domain they are trying to access.<br>**NOTE:** The last-resort context name can be the same as the source context. |
| Subscriber username format | Specifies the format of subscriber usernames as to whether or not the username or domain is specified first and the character that separates them. The possible separator characters are:<br>• @<br>• %<br>• -<br>• \<br>• #<br>• /<br>Up to six username formats can be specified. The default is *username @*.<br>**NOTE:** The username string is searched from right to left for the separator character. Therefore, if there is one or more separator characters in the string , only the first one that is recognized is considered the actual separator. For example, if the default username format was used, then for the username string *user1@enterprise@isp1*, the system resolves to the username *user1@enterprise* with domain *isp1*. |

# How This Configuration Works

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

*Figure 23. Call Processing When Using the system as a PDSN/FA*




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

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

4. The system then communicates with the foreign AAA server specified in the AAA context's AAA configuration to authenticate the subscriber.
5. Upon successful authentication, the PDSN service determines the IP address of the subscriber's HA using either an attribute returned in the Access Accept message, or the address specified by the mobile.

   The PDSN service uses the Mobile IP FA context name to determine what destination context is facilitating the FA service. In this example, it determines that it must use the *MIP Destination* context.
6. The PDSN service passes the HA IP address to the FA service.
7. The FA service then establishes a connection to the specified HA over the Pi interface.
8. Accounting messages for the session are sent to the Foreign AAA server over the AAA interface.

# Chapter 5
# Simple IP and Mobile IP in a Single System Configuration Example

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

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

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

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

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

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

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

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

*Figure 24. sipmip_cfg_ex_1a.wmf*

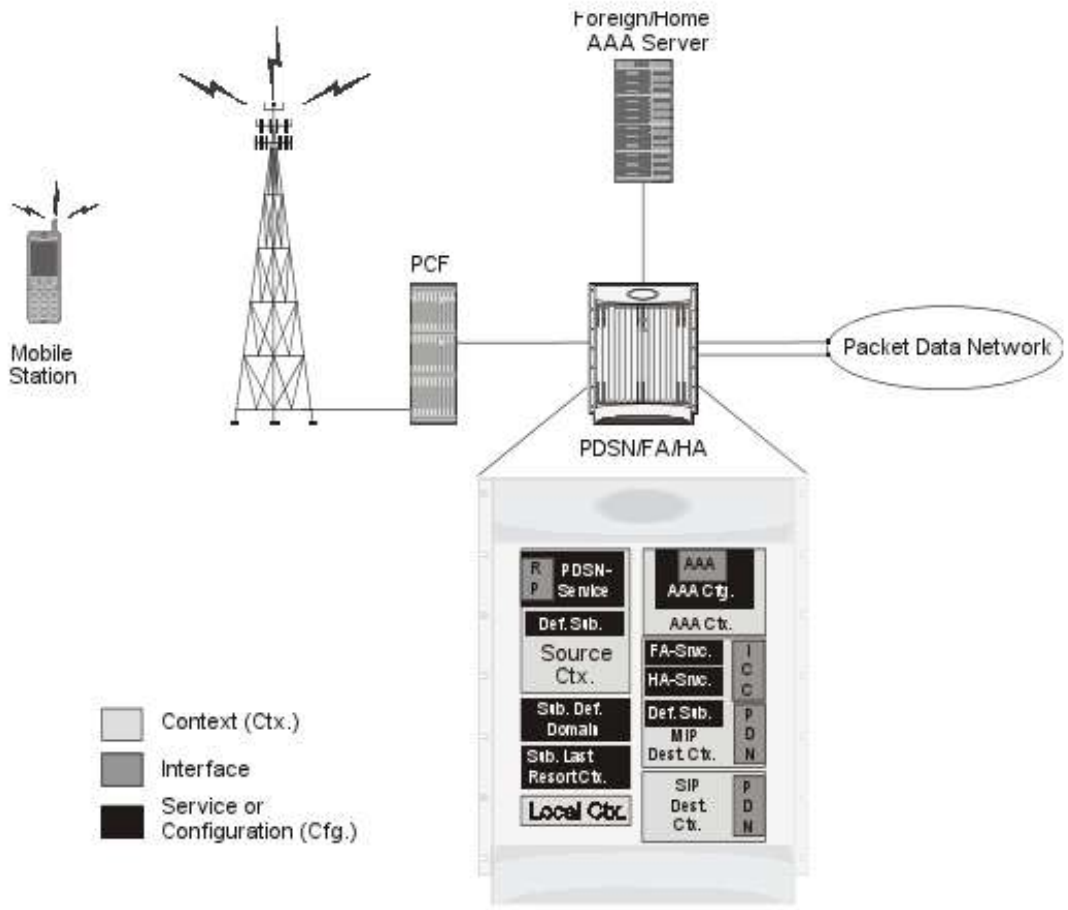



*Figure 25. Simple and Mobile IP Support Within a Single System*

# Information Required

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

## Source Context Configuration

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

*Table 23. Required Information for Source Context Configuration*

| Required Information | Description |
|---|---|
| Source context name | This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the source context will be recognized by the system. |
| R-P Interface Configuration | |
| R-P interface name | This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the interface will be recognized by the system.<br>Multiple names are needed if multiple interfaces will be configured.<br>R-P interfaces are configured in the source context. |
| IP address and subnet | These will be assigned to the R-P interface.<br>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.<br>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.<br>Multiple descriptions are needed if multiple ports will be used.<br>Physical ports are configured within the source context and are used to bind logical R-P interfaces. |
| Gateway IP address | Used when configuring static routes from the R-P interface(s) to a specific network. |
| PDSN service Configuration | |
| PDSN service name | This is an identification string between 1 and 63 characters (alpha and/or numeric) by which the PDSN service will be recognized by the system.<br>Multiple names are needed if multiple PDSN services will be used.<br>PDSN services are configured in the source context. |
| UDP port number for R-P traffic | Specifies the port used by the PDSN service and the PCF for communications. The UDP port number and can be any integer value between 1 and 65535. The default value is 699. |
| Authentication protocols used | Specifies how the system handles authentication: using a protocol (such as CHAP, PAP, or MSCHAP), or not requiring any authentication. |
| Domain alias for NAI-construction | Specifies a context name for the system to use to provide accounting functionality for a subscriber session. This parameter is needed only if the system is configured to support no authentication. |
| Security Parameter Index Information | **PCF IP address:**<br>Specifies the IP address of the PCF that the PDSN service will be communicating with. The PDSN service allows the creation of a security profile that can be associated with a particular PCF.<br>Multiple IP addresses are needed if the PDSN service will be communicating with multiple PCFs. |
| | **Index:**<br>Specifies the shared SPI between the PDSN service and a particular PCF. The SPI can be configured to any integer value between 256 and 4294967295.<br>Multiple SPIs can be configured if the PDSN service is to communicate with multiple PCFs. |
| | **Secret:**<br>Specifies the shared SPI secret between the PDSN service and the PCF. The secret can be between 1 and 127 characters (alpha and/or numeric).<br>An SPI secret is required for each SPI configured. |

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

## AAA Context Configuration

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

*Table 24. Required Information for AAA Context Configuration*

| Required Information | Description |
|---|---|
| AAA context name | This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the AAA context will be recognized by the system. |
| AAA Interface Configuration | |
| AAA interface name | This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the interface will be recognized by the system.<br>Multiple names are needed if multiple interfaces will be configured.<br>AAA interfaces will be configured in the source context. |
| IP address and subnet | These will be assigned to the AAA interface.<br>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.<br>A single physical port can facilitate multiple interfaces. |

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

## Mobile IP Destination Context Configuration

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

*Table 25. Required Information for Destination Context Configuration*

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

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

| Required Information | Description |
|---|---|
| Number of simultaneous registrations | Specifies the number of simultaneous Mobile IP sessions that will be supported for a single subscriber.<br>The maximum number of sessions is 3. The default is 1.<br>**NOTE:** The system will only support multiple Mobile IP sessions per subscriber if the subscriber's mobile node has a static IP address. |
| Mobile node re-registration requirements | Specifies how the system should handle authentication for mobile node re-registrations.<br>The FA service can be configured to always require authentication or not. If not, the initial registration and de-registration will still be handled normally. |
| HA service Configuration | |
| HA service name | This is an identification string between 1 and 63 characters (alpha and/or numeric) by which the HA service will be recognized by the system.<br>Multiple names are needed if multiple HA services will be used.<br>HA services are configured in the destination context. |
| UDP port number for Mobile IP traffic | Specifies the port used by the HA service and the FA for communications. The UDP port number can be any integer value between 1 and 65535. The default value is 434. |
| Mobile node re-registration requirements | Specifies how the system should handle authentication for mobile node re-registrations.<br>The HA service can be configured as follows:<br>• Always require authentication<br>• Never require authentication (**NOTE:** the initial registration and de-registration will still be handled normally)<br>• Never look for mn-aaa extension<br>• Not require authentication but will authenticate if mn-aaa extension present |
| FA-to-HA Security Parameter Index Information | **FA IP address:**<br>The HA service allows the creation of a security profile that can be associated with a particular FA. This specifies the IP address of the FA that the HA service will be communicating with.<br>Multiple FA addresses are needed if the HA will be communicating with multiple FAs. |
| | **Index:**<br>Specifies the shared SPI between the HA service and a particular FA. The SPI can be configured to any integer value between 256 and 4294967295.<br>Multiple SPIs can be configured if the HA service is to communicate with multiple FAs. |
| | **Secret:**<br>Specifies the shared SPI secret between the HA service and the FA. The secret can be between 1 and 127 characters (alpha and/or numeric).<br>An SPI secret is required for each SPI configured. |
| | **Hash-algorithm:**<br>Specifies the algorithm used to hash the SPI and SPI secret. The possible algorithms that can be configured are MD5 per RFC 1321 and keyed-MD5 per RFC 2002. The default algorithm is hmac-md5.<br>A hash-algorithm is required for each SPI configured. |
| Mobile Node Security Parameter Index Information | **Index:**<br>Specifies the shared SPI between the HA service and the mobile node(s). The SPI can be configured to any integer value between 256 and 4294967295.<br>Multiple SPIs can be configured if the HA service is to communicate with multiple mobile nodes. |

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

## Simple IP Destination Context

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

***Table 26. Required Information for Destination Context Configuration***

| Required Information | Description |
|---|---|
| Destination context name | This is an identification string between 1 and 79 characters (alpha and/or numeric) by which the destination context will be recognized by the system.<br>**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.<br>Multiple names are needed if multiple interfaces will be configured.<br>PDN interfaces are configured in the destination context. |
| IP address and subnet | These will be assigned to the PDN interface.<br>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.<br>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.<br>Multiple descriptions are needed if multiple ports will be used.<br>Physical ports are configured within the destination context and are used to bind logical PDN interfaces. |
| Gateway IP address(es) | Used when configuring static routes from the PDN interface(s) to a specific network. |
| IP Address Pool Configuration (optional) | |
| IP address pool name | Each IP address pool is identified by a name. The pool name can be between 1 and 31 alpha and/or numeric characters and is case sensitive.<br>IP address pools are configured in the destination context(s). Multiple address pools can be configured within a single context. |
| IP pool addresses | An initial address and a subnet, or a starting address and an ending address, are required for each configured pool. The pool will then consist of every possible address within the subnet , or all addresses from the starting address to the ending address.<br>The pool can be configured as public, private, or static. |

## System-Level AAA Parameter Configuration

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

*Table 27. Required Information for System-Level AAA Configuration*

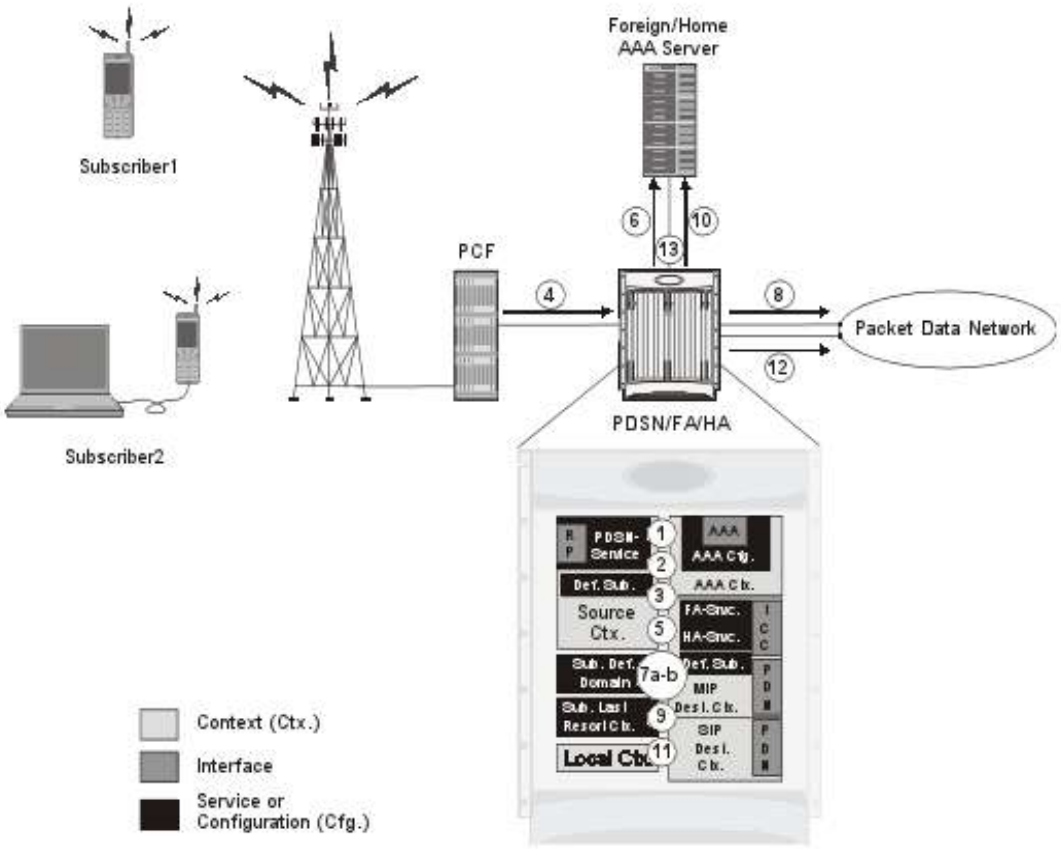
| Required Information | Description |
|---|---|
| Subscriber default domain name | Specifies the name of a context that can provide AAA functions in the event that the domain-part of the username is missing or poorly formed.<br>This parameter will be applied to all subscribers if their domain can not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

| Required Information | Description |
|---|---|
| Subscriber Last-resort context | Specifies the name of a context that can provide AAA functions in the event that the domain-part of the username was present but does not match the name of a configured destination context.<br>This parameter will be applied to all subscribers if their specified domain does not match a configured destination context regardless of what domain they are trying to access.<br>**NOTE:** The last-resort context name can be the same as the source context. |
| Subscriber username format | Specifies the format of subscriber usernames as to whether or not the username or domain is specified first and the character that separates them. The possible separator characters are:<br>• @<br>• %<br>• -<br>• \<br>• #<br>• /<br>Up to six username formats can be specified. The default is *username @*.<br>**NOTE:** The username string is searched from right to left for the separator character. Therefore, if there is one or more separator characters in the string , only the first one that is recognized is considered the actual separator. For example, if the default username format was used, then for the username string *user1@enterprise@isp1*, the system resolves to the username *user1@enterprise* with domain *isp1*. |

# How This Configuration Works

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

*Figure 26. Call Processing When Using the System as a PDSN, FA, and HA*




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

1. The system-level AAA settings were configured as follows:
   - Default domain name = *AAA*
   - Subscriber username format = *username @*
   - Last-resort context name = *AAA*
2. The Default Subscriber was configured with an IP context name of *SIP Destination.*
3. The Mobile IP FA context name parameter within the PDSN service was configured to the *MIP Destination* context.
4. Sessions for *Subscriber1* and *Subscriber2* are received by the PDSN service over the R-P interface from the PCF.
5. The PDSN service determines which context to use to provide foreign AAA functionality for each session. This process is described in the How the System Selects Contexts section located in the Understanding the System Operation and Configuration chapter of the *System Administration Guide*.

For this configuration, the result of this process for both *Subscriber1* and *Subscriber2* would be that the system determines that AAA functionality should be provided by the *AAA* context.

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

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

# Chapter 6
# Service Configuration Procedures

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

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

This section includes the following topics:



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

***Important:*** For hardware supporting them, at least one Packet Accelerator Card (PAC) or Packet Services Card (PSC) must be made active prior to service configuration. Information and instructions for configuring PACs/PSCs to be active can be found in the Configuring System Settings chapter of the System Administration Guide.

# Creating and Configuring PDSN Services

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

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

Use this example to configure PDSN services:

```
configure
   context <name>
      pdsn-service <name>
         ip local-port <port#>
         authentication allow-noauth
         authentication chap 1 mschap 2 pap 3 allow-noauth
         nai-construct domain <alias>
         spi remote-address <pcf_ip_address> spi-number <number> { secret
<secret> }
         lifetime <time>
         gre protocol-type { any | byte-stream | ppp }
         bind address address
         exit
      ppp lcp-start-delay <seconds>
      no ppp renegotiation retain-ip-address
      end
```

Notes:

- Optional: If you are implementing Mobile IP data services, configure the name of the context in which the FA service is configured by entering the **`mobile-ip foreign-agent context fa_context_name [fa-service <name>]`** command.
- Optionally configure the PDSN service to monitor all PCFs that it is associated with, enter the **`pcf-monitor`** command.

- Optionally configure the PDSN behavior for A11 RRQ related parameters. **airlink bad-sequence-number deny** can be used to deny A11 RRQ messages that have an unsupported Vendor Id or invalid Airlink Sequence number (less than or equal to a previously received sequence number). Keywords and options that configure additional PDSN service behavior for A11 RRQs with this command are provided in the Command Line Interface Reference.
- Optionally use the **no dormant-transition initial-session-setup** command to configure the PDSN behavior to terminate A10 session, when the PDSN receives the A11-RRQ (Type 4) before the session for the original MN is established completely.
- Optionally use the **no pcf-session-id-change restart-ppp** command to configure the PDSN behavior to disable the ppp renegotiation, when the PDSN receives the A11 RRQ (Type 4) with a change in GRE key or PCF session Id, from current PCF and no change in PCF/PANID/CANID.
- Optionally use the **setup-timeout***<seconds>* command to change the maximum amount of time, in seconds, allowed to set up a session. The default setting is 60 seconds.
- Optionally configure a delay before starting LCP to avoid the first LCP Configuration Request being lost because the RP link may not be ready even if it has indicated it is active. Losing an LCP Config Request increases the total session setup time.
- Optional: You can configure the system whether to retain the currently allocated IP address for the session or to release the current IP address, and a new IP address is to allocate after PPP renegotiation.
- To retain the allocated IP during PPP renegotiation use the **[default] ppp renegotiation retain-ip-address** command

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

- Optionally configure the MSID length to reject the A11-RRQs with illegal IMSI value by entering the **[ default ] msid length [ min** *min_length* **] max** *max_length* command:

  By default it will use the default MSID length as per standard. Detailed informations are provided in Command Line Interface Reference.
- The nai-construct domain command should only be used if the PDSN service is configured to allow no authentication using the authentication allow-noauth command.
- Multiple SPIs can be configured within the PDSN service in order to accommodate a single PDSN interface communicating with multiple PCFs.
- An infinite lifetime can be configured using the no lifetime command.
- Multiple addresses on the same IP interface can be bound to different PDSN services. However, each address can be bound to only one PDSN service.
- The hardware configuration and features installed can affect the maximum subscriber sessions that can be supported.
- Repeat this configuration as needed to create and bind additional PDSN services to any other interfaces.
- Save your configuration as described in Saving Your Configuration.

# Verifying the PDSN Services

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

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

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

```
Service name: pdsn1
 Context: test1
 Bind: Not Done
 Local IP Address: 0.0.0.0 Local IP Port: 699
 Lifetime: 00h30m00s Retransmission Timeout: 3 (secs)
 Max Retransmissions: 5 Setup Timeout : 60 (secs)
 No MIP FA Context defined
 No NAI construct domain defined
 GRE Sequence Numbers: Enabled GRE Protocol Type: Any
 GRE Reorder Timeout: 100 msec GRE Sequence Mode: None
 GRE Checksum: Disabled GRE Checksum Verification: Disabled
 Enable Data Available Indicator: Yes Inter-PDSN handoffs have MEI: No
 Reg discard on bad extension: No Reg discard on GRE key change: No
 Reg ack deny terminates session: No Reg update wait timeout: No
 Deny newcall if no rev. tunnel: No
 Terminate session on R-P errors: No Max retried replies on reg deny: 3
 Deny using zero GRE key: No Deny if session already closed: No
 Deny if session already dormant: No Deny if session already active: No
 Deny if CoA & src addr mismatch: No
 Deny newcall if no conn setup: No (Deny code: Reason Unspecified)
 RRQ with bad airlink seq num: Accept(Deny code: Poorly Formed Request)
 Deny if CRP to RP H/O in progress:No
 Handoff with no conn setup: Accept
 Accept H/O if sess being disc: No
```

```
PPP Authentication: CHAP 1 PAP 2
Allow Noauthentication: Disabled MSID Authentication: Disabled
Fragment PPP Data: Enabled
GRE Flow Control: Disabled
GRE Flow Control Timeout: 10000 msec
GRE Flow Control Timeout Action: disconnect-session
Max sessions: 500000
Alt-PPP: Disabled
PPP Tunnel Type: None No PPP Tunnel Context defined
No Default Subscriber defined
IP SRC-Violation Reneg Limit: 5 IP SRC-Violation Drop Limit: 10
IP SRC-Violation Clear-on-ValidPDU: No IP SRC-Violation Period: 120 secs
Always-On-Indication: Disabled SDB Indication for Echo Req: Disabled
SPI(s):
Service Status: Not started
Overload Policy: Reject (Reject code: Admin Prohibited)
Newcall Policy: None
Service Option Policy: Enforce
Service Options: 7,15,22,23,24,25,33,59
PCF Monitor Config: Disabled
```

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

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

# Creating and Configuring FA Services

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

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

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

```
configure
   context <name>
      fa-service <name>
         ip local-port <port#>
         fa-ha-spi remote-address <ha_ip_address> spi-number number
{ encrypted secret <secret> | secret <secret> }
         advertise adv-lifetime <time>
         advertise num-adv-sent <number>
         advertise reg-lifetime <reg_time>
         multiple-reg <number>
         authentication mn-aaa { always | ignore-after-handoff | init-reg |
init-reg-except-handoff | renew-and-dereg-noauth | renew-reg-noauth }
         reg-timeout <time>
         bind address <address> max-subscribers <max#>
         end
```

Following are a few things to be aware of:

- The **ip local-port** command configures the User Datagram Protocol (UDP) port for the Pi interfaces' IP socket.
- A maximum of 2048 FA-HA SPIs can be configured for a single FA service.
- The agent advertisement lifetime is the amount of time that an FA agent advertisement remains valid in the absence of further advertisements.
- An infinite registration lifetime can be configured using the **no advertise reg-lifetime** command.

- The system only supports multiple Mobile IP sessions per subscriber if the subscriber's mobile node has a static IP address. The system only allows a single Mobile IP session for mobile nodes that receive a dynamically assigned home IP address.The hardware configuration and features installed can affect the maximum subscriber sessions that can be supported.
- Optionally configure the FA service for controlling the negotiation and sending of the I-bit in revocation messages by adding the **`revocation negotiate-i-bit`** comand. By default, it will not send I-bit in revocation message.
- Repeat the configuration as needed to create and bind additional FA services to any other interfaces.

# Verifying the FA Service

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

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

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

```
Service name: fa1
 Context: xxx
  Bind: Done Max Subscribers: 500000
 Local IP Address: 195.20.20.3 Local IP Port: 434
 Lifetime: 00h10m00s Registration Timeout: 45 (secs)
 Advt Lifetime: 02h30m00s Advt Interval: 5000 (msecs)
 Num Advt: 5
 Advt Prefix Length Extn: NO
 Reverse Tunnel: Enabled GRE Encapsulation: Enabled
 Optimize Tunnel Reassembly: Disabled Allow Priv Addr w/o Rev Tunnel:
Disabled
 Dynamic MIP Key Update: Enabled Ignore Dynamic MIP Key: Disabled
 Remove MN-AAA/MN-FAC extns: Disabled
 Proxy MIP: Enabled Proxy MIP Max Retransmissions: 5
 Proxy MIP Retrans Timeout: 3 (secs) Proxy MIP Renew Percent Time: 75%
```

```
 SPI(s):

  FAHA: Remote Addr: 195.30.30.3/32

 Hash Algorithm: HMAC_MD5 SPI Num: 1000 Replay Protection: Timestamp
Timestamp Tolerance: 60 FAHA: Remote Addr: 195.30.30.2/32  Hash
Algorithm: HMAC_MD5 SPI Num: 1000 Replay Protection: Timestamp Timestamp
Tolerance: 60 FAHA: Remote Addr: 195.30.30.1/32  Hash Algorithm: HMAC_MD5
SPI Num: 1000 Replay Protection: Timestamp Timestamp Tolerance: 60

 FAHA: Remote Addr: 195.20.20.4/32

 Hash Algorithm: HMAC_MD5 SPI Num: 1000

 Replay Protection: Timestamp Timestamp Tolerance: 60

 IPSEC Crypto Map(s):

 Peer HA Addr: 195.30.30.2

 Crypto Map: test

 GRE Sequence Numbers: Disabled GRE Sequence Mode: None

 GRE Reorder Timeout: 100 msec

 GRE Checksum: Disabled GRE Checksum Verification: Disabled

  Registration Revocation: Enabled Reg-Revocation I bit: Enabled

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

 Reg-Rev on InternalFailure: Enabled

 Default Subscriber: None

 Max sessions: 500000

 Max challenge len: 16

 Challenge Window: 2

 Service Status: Started

 MN-AAA Auth Policy: Always

 MN-HA Auth Policy: Always

 Newcall Policy: None

 Idle Timeout Mode: Normal

 Ignore Stale Challenge: Disabled
```

**Step 2** Save your configuration as described in the Saving Your Configuration chapter.

# Creating and Configuring HA Services

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

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

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

```
configure
   context <name>
      ha-service <name>
         ip local-port <port#>
         authentication mn-aaa { allow-noauth | always | noauth |
renew-reg-noauth }
         fa-ha-spi remote-address fa_ip_address spi-number <number> { encrypted
secret <secret> | secret <secret> }
         mn-ha-spi spi-number <number> { encrypted secret <secret> | secret
<secret> }         reg-lifetime <time>         simultaneous-bindings
<number>         bind address <address> max-subscribers <max#>         end
```

Following are a few things to be aware of:

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

# Verifying the HA Service

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

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

The output is a concise listing of HA service parameter settings. In this example, a HA service called ha1 was configured.

```
Service name: ha1
 Context: ha
 Bind: Done Max Subscribers: 500000
 Local IP Address: 192.168.4.10 Local IP Port: 434
 Lifetime: 00h10m00s Simul Bindings: 3
 Reverse Tunnel: Enabled GRE Encapsulation: Enabled
 Optimize Tunnel Reassembly: Enabled Setup Timeout: 60 sec
SPI(s):
 MNHA: Remote Addr: 0.0.0.0
 Hash Algorithm: MD5 SPI Num: 1000
 Replay Protection: Timestamp Timestamp Tolerance: 60
 Permit Any Hash Algorithm: Disabled
 FAHA: Remote Addr: 195.20.20.6/32
 Hash Algorithm: HMAC_MD5 SPI Num: 1000
 Replay Protection: Timestamp Timestamp Tolerance: 60
 FAHA: Remote Addr: 195.20.20.5/32
 Hash Algorithm: HMAC_MD5 SPI Num: 1000
 Replay Protection: Timestamp Timestamp Tolerance: 60
 FAHA: Remote Addr: 195.20.20.3/32
 Hash Algorithm: HMAC_MD5 SPI Num: 1000
 Replay Protection: Timestamp Timestamp Tolerance: 60
 FAHA: Remote Addr: 195.20.20.2/32
```

```
 Hash Algorithm: HMAC_MD5 SPI Num: 1000
 Replay Protection: Timestamp Timestamp Tolerance: 60
IPSEC Crypto Map(s):
 Peer FA Addr: 192.168.4.1
 Crypto Map: test
 'S' Key expires at: No Valid S-Key
'S' Lifetime Skew: 00h00m10s
 IPSEC AAA Context: xxx
GRE Sequence Numbers: Disabled GRE Sequence Mode: None
 GRE Reorder Timeout: 100 msec
 GRE Checksum: Disabled GRE Checksum Verification: Disabled
 Registration Revocation: Enabled Reg-Revocation I bit: Enabled
 Reg-Revocation Max Retries: 3 Reg-Revocation Timeout: 3 (secs)
 Reg-Rev Handoff old-FA: Enabled Reg-Rev Idle-Timeout: Enabled
Default Subscriber: None
 Max Sessions: 500000
 Service Status: Started
 MN-AAA Auth Policy: Always
 MN-HA Auth Policy: Always
 IMSI Auth: Disabled
 AAA accounting: Enabled
 Idle Timeout Mode: Aggressive
 Newcall Policy: None
 Overload Policy: Reject (Reject code: Admin Prohibited)
 NW-Reachability Policy: Reject (Reject code: Admin Prohibited)
```

**Step 2** Save your configuration as described in the Saving Your Configuration chapter.

# Configuring IP Address Pools on the System

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

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

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

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

To configure the IP pool:

- Create the IP pool for IPv4 addresses in system context by applying the example configuration.
- Optional. Configure the IP pool for IPv6 addresses in system context by applying the example.
- Optional. Configure the overlap-pool addresses to routing by applying the example configuration.
- Verify your IP pool configuration.
- Save your configuration as described in the Saving Your Configuration chapter.

## Creating IPv4 Pool

Use the following example to create the IPv4 address pool:

```
configure
   context <dest_ctxt_name>
      ip pool <pool_name> <ip_address/mask>
      end
```

Following are a few things to be aware of:

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

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

# Creating IPv6 Pool

Use the following example to create the IPv6 address pool:

```
configure
   context <dest_ctxt_name>
      ipv6 pool <pool_name> 6to4 local-endpoint <ip_address>
      end
```

Following are a few things to be aware of:

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

# Adding Overlap-Pool Addresses to Routing

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

```
configure
   context <context_name>
      [ no | default ] ip routing overlap-pool
```

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

# Verifying IP Pool Configuration

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

**show ip pool**

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

```
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
RG00 pool3       30.30.0.0 255.255.0.0 0 65534
SG00 pool2       20.20.0.0 255.255.0.0 10 65524
PG00 pool1       10.10.0.0 255.255.0.0 0 65534
SG00 vpnpool      192.168.1.250 192.168.1.254 0 5
Total Pool Count: 5
```

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

**show ipv6 pools**

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

# Chapter 7
# 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:<br>• **{ /flash \| /pcmcia1 \| /pcmcia2 }** *[ /dir ] /file_name*<br>• **file:/{ /flash \| /pcmcia1 \| /pcmcia2 }** *[ /dir ] /file_name*<br>• **tftp://**{ *ipaddress*\|*host_name*[ *:port#* ]} [ */directory*] */file_name*<br>• **ftp://**[ *username*[ *:pwd*] *@* ] { *ipaddress*\|*host_name*} [ *:port#* ] [ */directory*] */file_name*<br>• **sftp://**[ *username*[ :*pwd*] *@* ] { *ipaddress*\|*host_name*} [ *:port#* ] [ */directory*] */file_name*<br>**/flash** corresponds to the CompactFlash on the SPC/SMC.<br>**/pcmcia1** corresponds to PCMCIA slot 1.<br>**/pcmcia2** corresponds to PCMCIA slot 2.<br>*ipaddress* is the IP address of the network server.<br>*host_name* is the network server's *hostname*.<br>*port#* is the network server's logical port number. Defaults are:<br>• tftp: 69 - data<br>• ftp: 20 - data, 21 - control<br>• sftp: 115 - data<br>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<br>*username* is the username required to gain access to the server if necessary.<br>*password* is the password for the specified username if required.<br>*/directory* specifies the directory where the file is located if one exists.<br>*/file_name* specifies the name of the configuration file to be saved.<br>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.<br>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.<br>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 8
# Monitoring the Service

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

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

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

# Monitoring System Status and Performance

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

***Table 28. System Status and Performance Monitoring Commands***

| To do this: | Enter this command: |
|---|---|
| **View Congestion-Control Statistics** | |
| View Congestion-Control Statistics | |
| View Congestion-Control Statistics | **show congestion-control statistics { a11mgr \| gtpcmgr \| hamgr \| l2tpmgr }** |
| View Subscriber Information | |
| View session resource status | **show resources session** |
| Display Subscriber Configuration Information | |
| View locally configured subscriber profile settings (must be in context where subscriber resides) | **show subscribers configuration username** *subscriber_name* |
| View remotely configured subscriber profile settings | **show subscribers aaa-configuration username** *subscriber_name* |
| View Subscribers Currently Accessing the System | |
| View a listing of subscribers currently accessing the system | **show subscribers all** |
| Display PCF-Summary Session Counters | |
| View PCF-summary session counters | **show session counters pcf-summary** |
| Display Session State Statistics | |
| View session state statistics | **show session progress** |
| Display Session State PCF Statistics | |
| View session state PCF statistics | **show session progress pcf all** |

| To do this: | Enter this command: |
|---|---|
| Display Session Subsystem and Task Statistics<br><br>**Important:** Refer to the System Software Task and Subsystem Descriptions appendix in the System Administration Guide for additional information on the Session subsystem and its various manager tasks. | |
| View A11 Manager statistics | **show session subsystem facility a11mgr all** |
| View AAA Manager statistics | **show session subsystem facility aaamgr all** |
| View FA Manager statistics | **show session subsystem facility famgr all** |
| View L2TP demux manager statistics | **show session subsystem facility l2tpdemux all** |
| View L2TP Manager statistics | **show session subsystem facility l2tpmgr all** |
| View Session Manager statistics | **show session subsystem facility sessmgr all** |
| View Session Recovery Status | |
| View session recovery status | **show session recovery status [verbose]** |
| Display Session Disconnect Reasons | |
| View session disconnect reasons with verbose output | **show session disconnect-reasons** |
| View Point-to-Point Protocol Statistics | |
| Display a Summary of PPP Counter Status | |
| View cumulative subscriber session PPP counters | **show ppp** |
| Display PPP Counters for a Specific Subscriber | |
| View individual subscriber session PPP counters | **show ppp username** *subscriber_name* |
| View individual subscriber session PPP error and data counters | **show ppp counters username** *subscriber_name* |
| View individual subscriber session detailed PPP counters | **show ppp full username** *subscriber_name* |
| Display PPP Statistics for PDSN Services | |
| Views PPP statistics for a all PDSN services | **show ppp statistics pdsn-service** |
| Views PPP statistics for a specific PDSN service | **show ppp statistics pdsn-service** *service_name* |

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

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

# Clearing Statistics and Counters

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

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

# Chapter 9
# Troubleshooting the System

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

# Test Commands

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

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

## Using the PPP Echo-Test Command

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

The command has the following syntax:

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

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

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

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

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

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

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

# Appendix A
# Engineering Rules

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

# Interface and Port Rules

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

## R-P Interface Rules

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

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

## Pi Interface Rules

### FA to HA Rules

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

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

- A Pi interface is created once the IP address of a logical interface is bound to an FA service.
- The logical interface(s) that will be used to facilitate the Pi interface(s) must be configured within the egress context.
- FA services must be configured within the egress context.

- If the system is configured as a FA is communicating with a system configured as an HA, then it is recommended that the name of the context in which the FA service is configured is identical to the name of the context that the HA service is configured in on the other system.
- Each FA service may be configured with the Security Parameter Index (SPI) of the HA that it will be communicating with over the Pi interface.
- Multiple SPIs can be configured within the FA service to allow communications with multiple HAs over the Pi interface. It is best to define SPIs using a netmask to specify a range of addresses rather than entering separate SPIs. This assumes that the network is physically designed to allow this communication.
- Depending on the services offered to the subscriber, the number of sessions facilitated by the Pi interface can be limited.

## HA to FA

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

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

# Subscriber Rules

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

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

# Service Rules

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

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

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

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

# Appendix B
# Supported Registration Reply Codes

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

# PDSN Service Reply Codes

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

***Table 29. Supported PDSN Service Registration Reply Codes***

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

# FA Service Reply Codes

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

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

***Table 30. Supported PDSN Service Registration Reply Codes***

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

| Reply Code(Hex / Base 10) | Description | Notes |
|---|---|---|
| 4EH / 78 | Registration Denied - registration timeout | Sent when FA reg-timeout is exceeded. |
| 4FH / 79 | Registration Denied - reverse tunneling delivery style unavailable | Sent if the Encapsulating Delivery Style Extension sent by the mobile is not supported by the FA service. |
| 50H / 80 | Registration Denied - home network unreachable (ICMP error received) | Sent when the FA service can not contact the home network due to an Internet Control Message Protocol (ICMP) error. |
| 51H / 81 | Registration Denied - home agent host unreachable (ICMP error received) | Sent when the FA service can not contact the HA host due to an Internet Control Message Protocol (ICMP) error. |
| 52H / 82 | Registration Denied - home agent port unreachable (ICMP error received) | Sent when the FA service can not contact the HA port due to an Internet Control Message Protocol (ICMP) error. |
| 58H / 88 | Registration Denied - home agent unreachable (other ICMP error received) | Sent when the FA service can not contact the HA due to an Internet Control Message Protocol (ICMP) error. |
| 60H / 96 | Registration Denied - missing home address | Sent when the FA service could not determine the IP address of the mobile node. |
| 61H / 97 | Registration Denied - missing NAI | Sent when the FA service could not determine the subscriber's network access identifier. |
| 62H / 98 | Registration Denied - missing home agent | Sent when the FA service could not determine the IP address of the mobile node's home agent. |
| 68H / 104 | Registration Denied - unknown challenge | Sent if the FA cannot validate the Mobile IP mobile-to-foreign agent advertisement challenge extension provided in the Registration Request. |
| 69H / 105 | Registration Denied - missing challenge | Sent if the mobile node's Registration Request does not include a mobile-to-foreign agent advertisement challenge extension. |
| 6AH / 106 | Registration Denied - stale challenge | Sent when the mobile node has sent a Registration Request with a challenge value that was already used before. |

# Appendix C
# Mobile-IP and Proxy-MIP Timer Considerations

This appendix is intended to provide a brief explanation of the considerations for lifetime, idle, and absolute timer settings that must be understood when setting up a system in a Mobile-IP or Proxy-MIP environment. The focus of the document is to understand the call flow and understand the timer values that must be applied to make the system function in the most efficient manner.

# Call Flow Summary

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

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

1. The call arrives at the system and A11 (or L2TP, for Closed RP interfaces) (or L2TP, for Closed RP interfaces) is processed successfully. The call arrives at the system and R6 isThe call arrives at the system and R6 is processed successfully.The GGSN receives a Create PDP Context Request Message
2. PPP negotiation is started. At this point,PPP negotiation is started. At this point, since authentication is not performed the system does not have a username or password. So during the PPP phase, the system selects the default subscriber in the source context for a subscriber template (DNS, and timer settings can be configured in the default subscriber template). Once PPP is successfully established the system understands that the call is a Mobile IP call. since authentication is not performed the system does not have a username or password. So during the PPP phase, the system selects the default subscriber in the source context for a subscriber template (DNS, and timer settings can be configured in the default subscriber template). Once PPP is successfully established the system understands that the call is a Mobile IP call.The system determines the propertiesThe system determines the properties
3. The new RRQ is accepted by the FA and sent to the HA. The HA authenticates the user and compares the requested lifetime to the configured MIP lifetime in the HA-service and the subscriber idle and absolute timeouts. If the MIP lifetime is lower it is be sent back to the mobile; if the MIP lifetime is higher the system sends back an RRQ accept with the lifetime set to 5 seconds less than the lower of the idle or absolute timeout for the user.

   The following CLI command sequence is used to configure the Mobile IP reg-lifetime in the HA service:

```
configure
   context <host_name>
      ha-service <ha_service_name>
         reg-lifetime <value>
         end
```

# Timer Values and Recommendations

The following table shows values that would be populated under a number of different configured scenarios.

***Table 31. Sample Call Flow Timer Scenarios***

| Scenario | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| Mobile Sub. MIP Lifetime | 600 | 600 | 600 | 600 | 600 | 600 | 600 |
| Source Context Default Sub-Source Context Default Sub-Absolute | 300 | 300 | 300 | 300 | 300 | 300 | 300 |
| Source Context Default Sub-Source Context Default Sub-Idle | 300 | 300 | 300 | 300 | 300 | 300 | 300 |
| FA-Service Advertise Reg-Lifetime | 400 | 400 | 400 | 400 | 400 | 400 | 400 |
| Mobile Sub. Profile AAA Context Timeout idle | 500 | 500 | 500 | 500 | 500 | 500 | 500 |
| HA-Service MIP Lifetime | 400 | 400 | 400 | 400 | 400 | 400 | 400 |
| Agent Advertisement Reg-Lifetime | 295 | 295 | 295 | 295 | 295 | 295 | 295 |
| Mobile Sub. MIP RRQ requested lifetime | 295 | 295 | 295 | 295 | 295 | 295 | 295 |
| FA MIP RRP Lifetime | 295 | 295 | 295 | 295 | 295 | 295 | 295 |
| FA MIP RRP | success | success | success | success | success | success | Lifetime too long |

Based on the table above, the recommended guidelines are as follows:

- If you are going to use timeout idle settings for subscribers, it is recommended that you configure the timeout idle parameter in the source context default subscriber to a value that is less than or equal to the lowest timeout idle for any subscriber.
- If you are going to use timeout absolute settings for subscribers, it is recommended that you configure the timeout absolute in the source context default subscriber to a value that is less than or equal to the lowest timeout idle for any subscriber.

Failure to follow these recommendations could result in lifetime too long failures when the FA processes the subscriber profileAPN template and finds an idle timeout that is less than the proposed MIP lifetime in the mobile RRQ.

# Controlling the Mobile IP Lifetime on a Per-Domain Basis

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

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

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

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

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

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

```
configure
   context <aaa_context_name>
      subscriber default
         ip context-name <abc>
         exit
      subscriber name <ptt.bigco.com>
         timeout idle <3605>
         ip context-name <abc>
         exit
      subscriber name <bigco.com>
         timeout idle <7205>
         ip context-name <abc>
         exit
```

```
      domain <ptt.bigco.com> default subscriber <ptt.bigco.com>
      domain <bigco.com> default subscriber <bigco.com>
         end
configure
   context <ha_context_name>
      subscriber default
      exit   ha-service <ha>
      idle-timeout-mode normal      reg-lifetime <7200>
      end
configure
   context <fa_context_name>
      fa-service <fa>
         advertise reg-lifetime <7200>
         end
```

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

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

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

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

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

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

The following optional modifier is also supported:

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