Cisco ASR 5000 Series Session Control Manager Administration Guide

Version 12.2

Last Updated February 29, 2012

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-25564-02

Cisco ASR 5000 Series Session Control Manager Administration Guide

CONTENTS

About this Guide

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

This preface includes the following sections:

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.
	Electrostatic Discharge (ESD)	Warns you to take proper grounding precautions before handling ESD sensitive components or devices.

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

Command Syntax Conventions	Description
{ **keyword** or *variable* }	Required keywords and variables are surrounded by braces. They must be entered as part of the command syntax.
[**keyword** or *variable*]	Optional keywords or variables that may or may not be used are surrounded by brackets.

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

Contacting Customer Support

Go to http://www.cisco.com/cisco/web/support/ to submit a service request. A valid Cisco account (username and password) is required to access this site. Please contact your Cisco account representative for additional information.

Chapter 1
Session Control Manager Overview

This chapter contains general overview information about the Session Control Manager (SCM) including:

- Product Description
- Network Deployments and Interfaces
- Features and Functionality - Base Software
- Features and Functionality - Licensed Enhanced Feature Support
- How the SCM Works
- Supported Standards

Product Description

The Session Control Manager (SCM) delivers and controls a robust multimedia environment today, while preparing for the networks of tomorrow. SCM provides an easy on-ramp to deploying Session Initiation Protocol (SIP)-based services and a future-proof migration path to the IP Multimedia Subsystem/Multimedia Domain (IMS/MMD) architectures.

The SCM performs the following functions:

- SIP routing
- Translation and mobility
- Admission control
- Authentication
- Registration
- Emergency Registration
- Packet network access based on pre-established policies and procedures
- Localized policy selection and enforcement
- Multimedia Call Detail Records (CDRs)
- Per-subscriber service facilitation
- SIP Application-level Gateway (ALG)
- Media relay
- Mitigate SIP Denial of Service (DoS)
- Prevent registration hijacking
- Prevent theft of service

The SCM consists of multiple IMS components that can be integrated into a single ASR 5000 platform or distributed as standalone network elements:

- IETF-compliant SIP Proxy/Registrar
- 3GPP/3GPP2-compliant Proxy Call/Session Control Function (P-CSCF)
- 3GPP/3GPP2-compliant Serving Call/Session Control Function (S-CSCF)
    - 3GPP/3GPP2-compliant Interrogating Call/Session Control Function (I-CSCF)
    - 3GPP/3GPP2 Breakout Gateway Control Function (BGCF)
- 3GPP/3GPP2-compliant Emergency Call/Session Control Function (E-CSCF)
- 3GPP/IETF-compliant Access Border Gateway (A-BG)

As standards-based network elements, SCM components can be integrated with each other or with third-party IMS components.

IMS Architecture

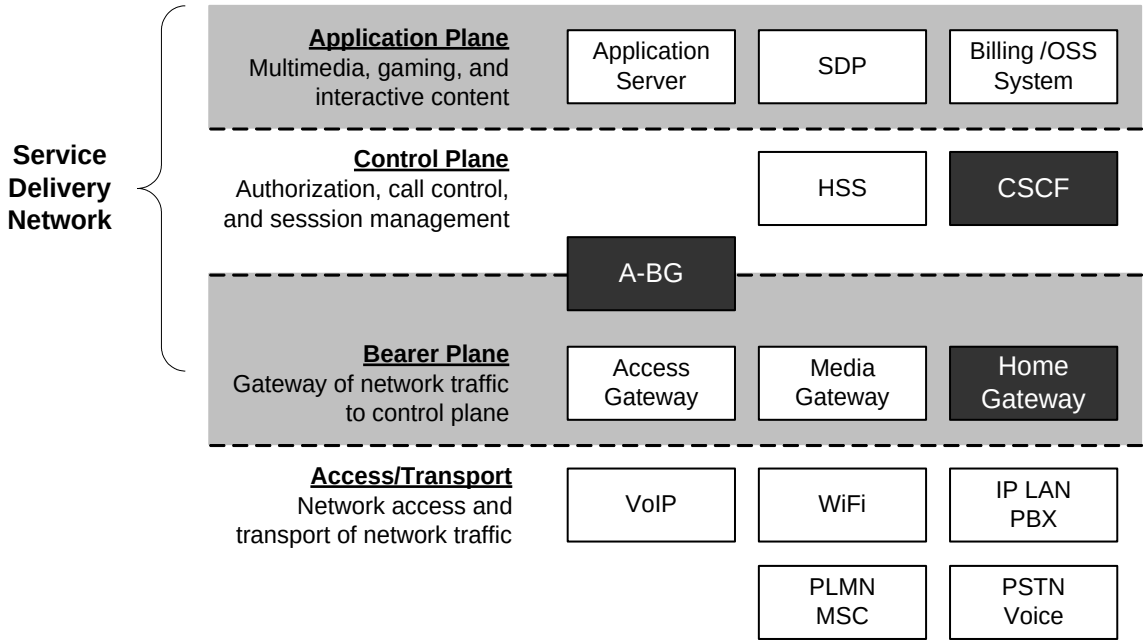
IP Multimedia Subsystem (IMS) specifies a standard architecture for providing combined IP services (voice, data, multimedia) over the existing public switched domain. IMS is an integral part of the 3GPP, 3GPP2, ETSI, and TISPAN network model standards that define circuit switched, packet switched, and IP multimedia domain environments. IMS

also supports multiple access methods such as CDMA2000, DOCSIS, EPS, Ethernet, Fiber, GPRS, WCDMA, WLAN, XDSL, and wireless broadband access.

The call signaling protocol used in IMS is the Session Initiation Protocol (SIP). The primary component in the network for resolving and forwarding SIP messages is the Call/Session Control Function (CSCF). The CSCF provides the control and routing function for all IP sessions accessing the network. CSCFs are located in the control plane or layer of the Service Delivery Network as shown in the figure below.

When the SCM acts as an Access Border Gateway (A-BG), it uses the RFC3261/P-CSCF to provide a SIP/IMS control plane access border, as well as a bearer access border control function. Therefore, the A-BG provides all session border control functions for all SIP UEs attempting to access the mobile network from a network outside of the operator's control and operations.

Figure 1. IMS Service Delivery Networks Components

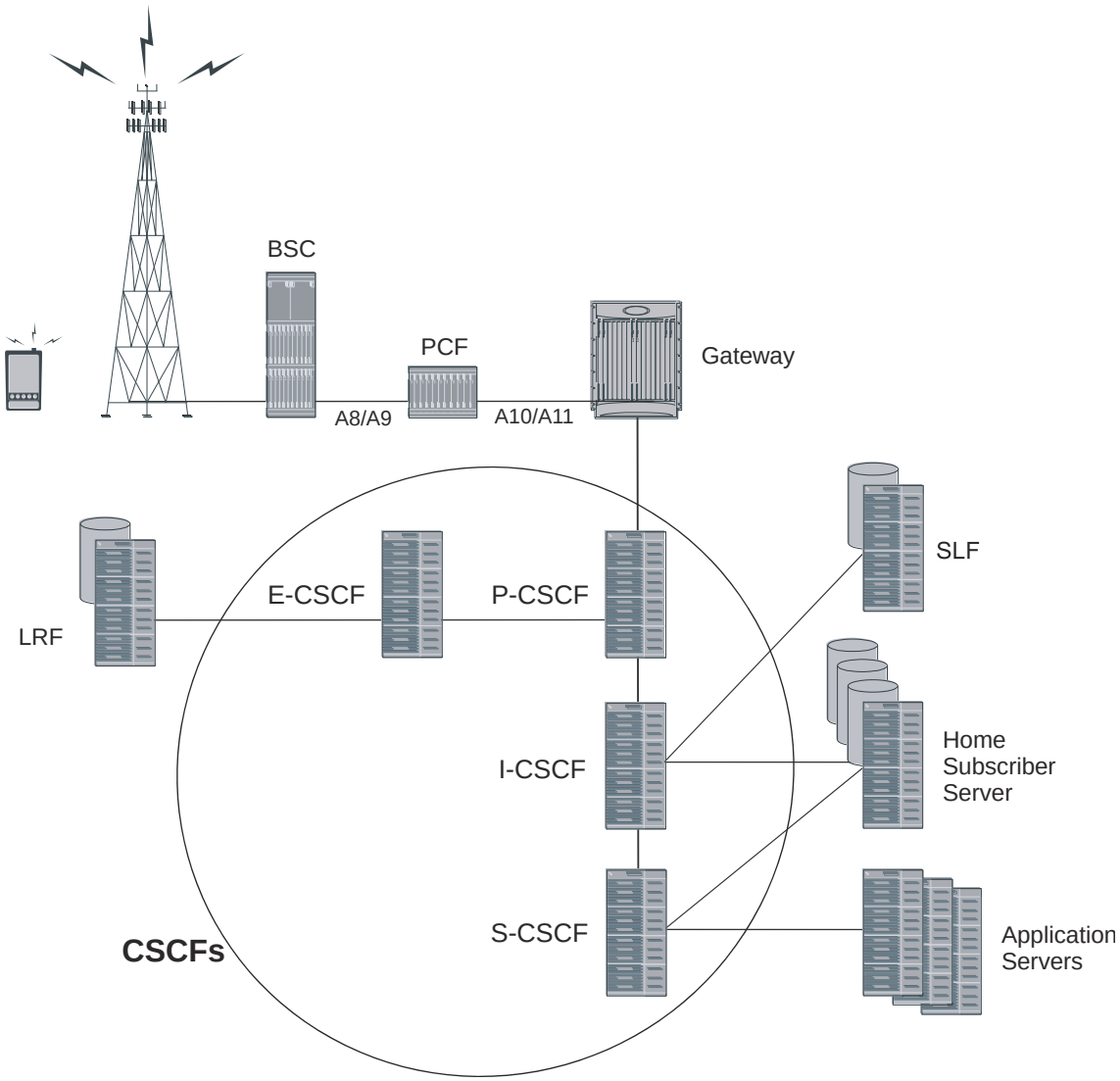


Collectively, CSCFs are responsible for managing an IMS session, including generating Call Detail Records (CDRs). Four functional behaviors are defined for the CSCF:

- Proxy
- Interrogating
- Serving
- Emergency

The following figure shows the general interaction between the CSCF components and the supporting servers.

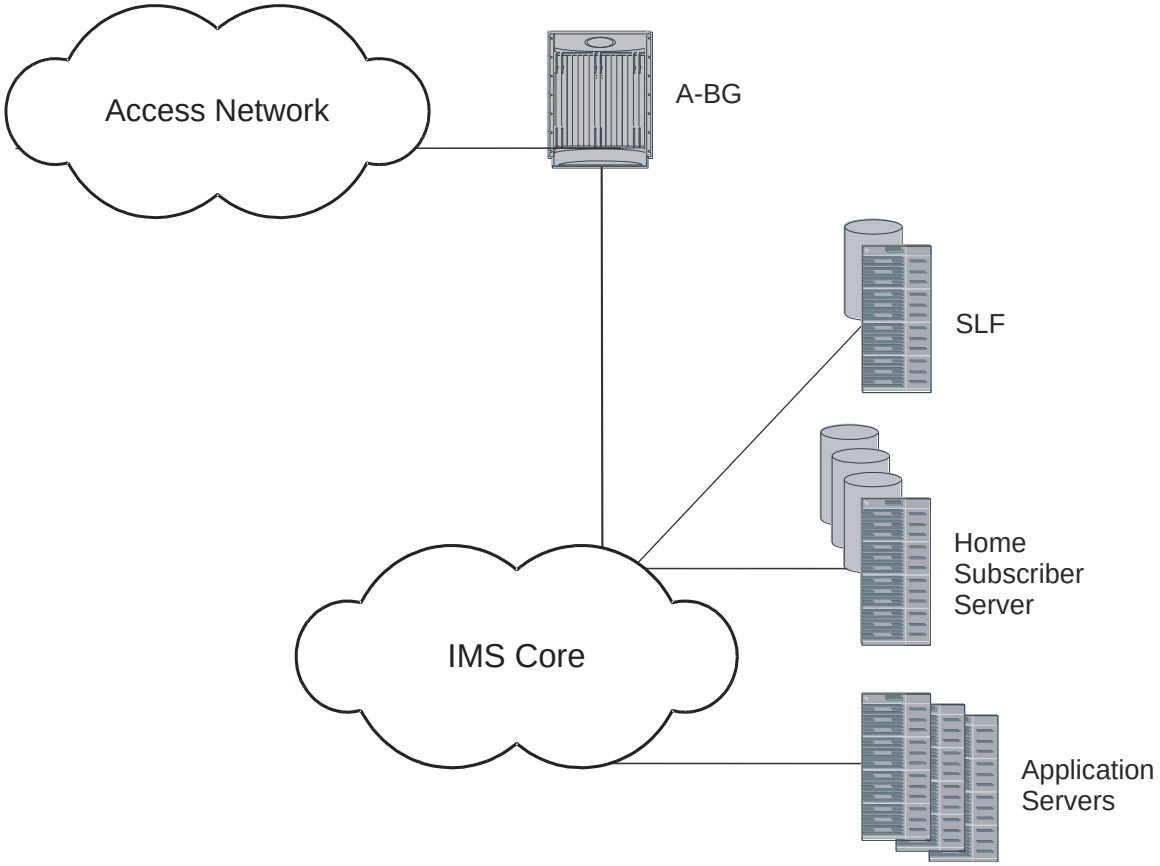
Figure 2. IMS CSCF Components



In addition, the SCM may act as an Access Border Gateway (A-BG).

The following figure shows the general interaction between the A-BG and the supporting servers.

Figure 3. Access Border Gateway



Proxy-CSCF

The primary point of entry into the IMS network is the Proxy-CSCF (P-CSCF). The P-CSCF is responsible for:

- providing message manipulation to allow for localized services (traffic/weather reports, news, directory services, etc.)
- initiating the breakout of emergency service calls
- Topology Hiding Inter-network Gateway (THIG)
- Quality of Service (QoS) authorization
- number conversions for local dialing plans
- terminating IPSec tunnels
- Transport Layer Security (TLS)
- interworking
- Signaling Compression/Decompression (SIGCOMP)
- charging

The P-CSCF is the handset's first point of entry into the IMS and is also the outbound proxy for SIP. Once the P-CSCF has completed all of the functions for which it is responsible, the call setup is handed off to the Interrogating-CSCF (I-CSCF).

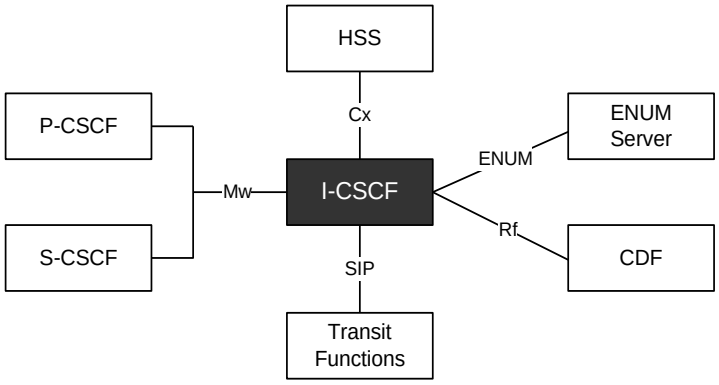
Interrogating-CSCF

The I-CSCF performs mostly as a load distribution device. The I-CSCF queries the Home Subscriber Server (HSS) to identify the appropriate Serving-CSCF (S-CSCF) to which the call is sent. Since the HSS maintains user profile information (much like the Home Location Register (HLR) in the Public Land Mobile Network (PLMN)), the I-CSCF can identify the proper S-CSCF for the call. The I-CSCF may also query a AAA server to determine subscriber profile information using DIAMETER.

Important: The I-CSCF is incorporated into the S-CSCF.

I-CSCF Interfaces

The following diagram shows the interfaces/reference points associated with the I-CSCF:



Serving-CSCF

The Serving-CSCF (S-CSCF) is the access point to services provided to the subscriber. Service examples include session control services, such as call features.

Other services include:

- VPN
- Centralized speed dialing lists
- Charging

The S-CSCF also interacts with the HSS for:

- User authentication
- Subscriber profile download and provisioning filter rules for services
- Network authentication key
- Emergency registration
- Location management
- User data handling

A Breakout Gateway Control Function is integrated into the SCM's S-CSCF to support PSTN calls.

Telephony Application Server (TAS) Basic Supported

The following describe the local basic call features implemented on the S-CSCF:

- Abbreviated Dialing (AD)
- Call Forward Busy Line (CFBL)
- Call Forward No Answer (CFNA)
- Call Forward Not Registered (CFNR)
- Call Forward Unconditional (CFU)
- Call Transfer
- Call Waiting
- Caller ID Display (CID)
- Caller ID Display Blocked (CIDB)
- Feature Code Activation/De-activation
- Follow Me/Find Me
- Locally Allowed Abbreviated Dialing
- Outbound Call Restrictions/Dialing Permissions
- Short Code Dialing

Integrated S/I-CSCF

The following Interrogating-CSCF features are supported for the integrated S/I-CSCF:

- **Assign an S-CSCF to a User Performing SIP Registration** - On a UE registration, the I-CSCF carries out a first step authorization and S-CSCF discovery. For this, the I-CSCF sends a Cx User-Authentication-Request (UAR) to the HSS by transferring the Public and Private User Identities and the visited network identifier (all extracted from the UE REGISTER message). The HSS answers with a Cx User-Authentication-Answer (UAA). The UAA includes the URI of the S-CSCF already allocated to the user. If there is no previously allocated S-CSCF, the HSS returns a set of S-CSCF capabilities that the I-CSCF uses to select the S-CSCF.
- **E.164 Address Translation** - Translates the E.164 address contained in all Request-URIs having the SIP URI with user=phone parameter format into the Tel: URI format before performing the HSS Location Query. In the event the user does not exist, and if configured by operator policy, the I-CSCF may invoke the portion of the transit functionality that translates the E.164 address contained in the Request-URI of the Tel: URI format to a routable SIP URI.
- **Obtain the S-CSCF Address from the HSS** - When the I-CSCF receives a SIP request from another network, it has to route the request to the called party. For this it obtains the S-CSCF address associated with the called party from the HSS by querying with a Cx Location-Information-Request (LIR) message. The Public-Identity AVP in the LIR is the Request-URI of the SIP request. The Location-Information-Answer (LIA) message contains the S-CSCF address in the Server-Name AVP. The request is then routed to the S-CSCF.
- **Route a SIP Request or Forward Response from Another Network** - When the I-CSCF receives a request from another network, it obtains the address of the S-CSCF from the HSS using the procedure detailed above and routes the request to the S-CSCF. Responses are also routed to the S-CSCF.
- **Perform Transit Routing Functions** - The I-CSCF may need to perform transit routing if, based on the HSS query, the destination of the session is not within the IMS. The IMS Transit Functions perform an analysis of the destination address and determine where to route the session. The session may be routed directly to an

MGCF, BGCF, or to another IMS entity in the same network, to another IMS network, or to a CS domain or PSTN.

- **Generate CDRs** - The I-CSCF generates CDRs for its interactions. Upon completing a Cx query, the I-CSCF sends an Accounting Request with the Accounting-Record-Type set to EVENT. The CDF acknowledges the data received and creates an I-CSCF CDR.

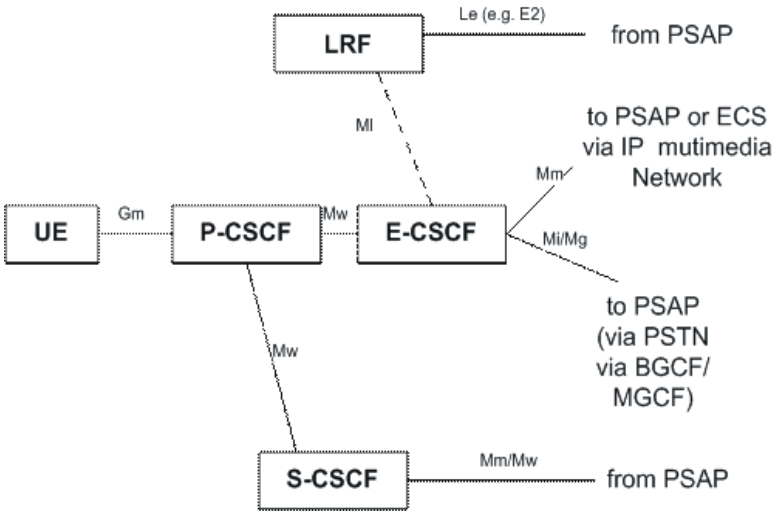
Emergency-CSCF

The Emergency-CSCF (E-CSCF) is a network element in IMS which is responsible for routing an emergency call to a Public Safety Answering Point (PSAP).

To identify the next hop PSAP, E-CSCF interacts with the Location Retrieval Function (LRF). LRF provides the necessary routing information so that E-CSCF can route the request to the appropriate PSAP.

E-CSCF Interfaces

The following diagram shows the interfaces/reference points associated with the E-CSCF:



A-BG

The A-BG is responsible for:

- Border Control for both Signaling and Bearer
- CALEA Support
    - SIP and media taps
- Call Admission and Access Control
    - Access Control based on IP, URL, SIP Identity, and Session Limits
- Intelligent Routing
    - Least Cost, Congestion Based, Call Type, Domain Based
    - As a SIP ALG, supports signaling and media routing with overlapping address ranges
- SIP Application-level Gateway (SIP-ALG)
    - SIP NAT Traversal

- SIP NAT (IPv4 <–> IPv6 translation)
- Media Relay (Header Manipulation): RTP, MSRP

- SIP Security
  - Prevent Theft of Service
    - Prevent CSCF bypass
    - Robust authentication procedures
    - SIP message checking
  - Prevent Registration Hijacking
    - Authenticate Re-Register (S-CSCF)
    - Early IMS Security: DoS attack prevention, impersonating a server
    - UA authentication (prevent server impersonation)
    - AKA authentication mechanism (further protection)
  - Prevent Message Tampering (IPSec)
  - Prevent Early Session Tear Down
    - Early IMS Security prevents a different user releasing existing session
  - Mitigate SIP Denial of Service (DoS)
    - P-CSCF DoS Attack Prevention
    - Blocking of user/IP address
      - after repeated authentication and bad request failure in Register/INVITE
    - Dropping of Register
      - containing Contact header pointing to CSCF service ip:port
    - Limited number of contacts on which Forking is allowed
    - Dropping of Requests
      - coming from source address other than the Register request's source address
- Topology Hiding Inter-network Gateway (THIG)
- Transport Layer Security (TLS)

Technical Specifications

The following table provides product specifications for the SCM.

Table 1. Session Control Manager Technical Specifications

	Description
Service Instances	Dual-mode proxy: simultaneously supports IETF & 3GPP/3GPP2 Proxies

	Description
SIP	• IETF SIP Proxy/Registrar • 3GPP/3GPP2 Proxy Call Session Control Function (P-CSCF) • Stateful session and subscriber aware control • Signaling Compression/Decompression (SIGCOMP) • Auto discovery, subscriber privacy, network security, call fraud prevention, thwarting network overload conditions
SIP Message Handling	Forking, error handling and discard, header stripping and insertion, Multiple public user identities
Logical Interfaces	• IETF: SIP Proxy/Registrar • 3GPP: Mw, Gm, Rx, Rf, Cx, Sh, Dx, MI • 3GPP2: Mw, Gm, Tx, Rf, Cx, Sh, Dx, MI

Platform Requirements

The SCM service runs on a Cisco® ASR 5x00 chassis running StarOS. The chassis can be configured with a variety of components to meet specific network deployment requirements. For additional information, refer to the Installation Guide for the chassis and/or contact your Cisco account representative.

Licenses

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

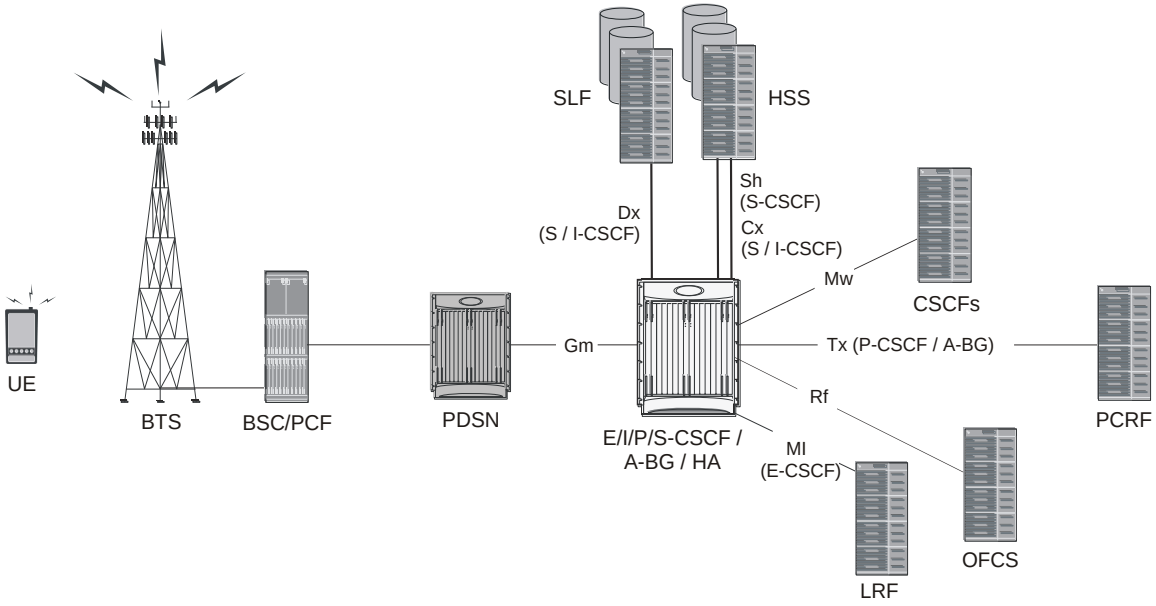
Network Deployments and Interfaces

SCM in a CDMA2000 Data Network Deployment

Integrated CSCF / A-BG / HA

The SCM is designed to function within a CDMA2000 PDSN network. By combining the SCM with a carrier-class Home Agent, a number of advantages emerge such as increased performance, distributed architecture, and high availability. As shown in the figure below, the SCM supports a number of interfaces used to communicate with other components in an IMS environment and supports the interface used to bridge the CDMA network.

Figure 4. CDMA2000 CSCF/A-BG/HA SCM Deployment Example



Logical Network Interfaces (Reference Points)

Interfaces, used to support IMS in a CDMA network, can be defined within two categories: SIP and DIAMETER. The SCM incorporates standards-based interfaces for both SIP and DIAMETER network architectures.

SIP Interfaces

The following table provides descriptions of SIP interfaces supported by the SCM in a CDMA2000 network deployment.

Table 2. SIP Interfaces in a CDMA Network

Interface	Description
Gm	The reference point between the P-CSCF and the User Equipment (UE).The Gm interface provides SIP signaling between the PDSN and the P-CSCF.
MI	The reference point between the E-CSCF and Location Retrieval Function (LRF). The MI interface is used for routing an emergency call to a Public Safety Answering Point (PSAP). The E-CSCF interacts with the Location Retrieval Function (LRF) to identify the next hop PSAP.
Mw	The reference point between the P/S-CSCF and other CSCFs.The Mw interface provides SIP signaling between two CSCFs.

DIAMETER Interfaces

The following table provides descriptions of DIAMETER interfaces supported by the SCM in a CDMA2000 network deployment.

Table 3. DIAMETER Interfaces in a CDMA Network

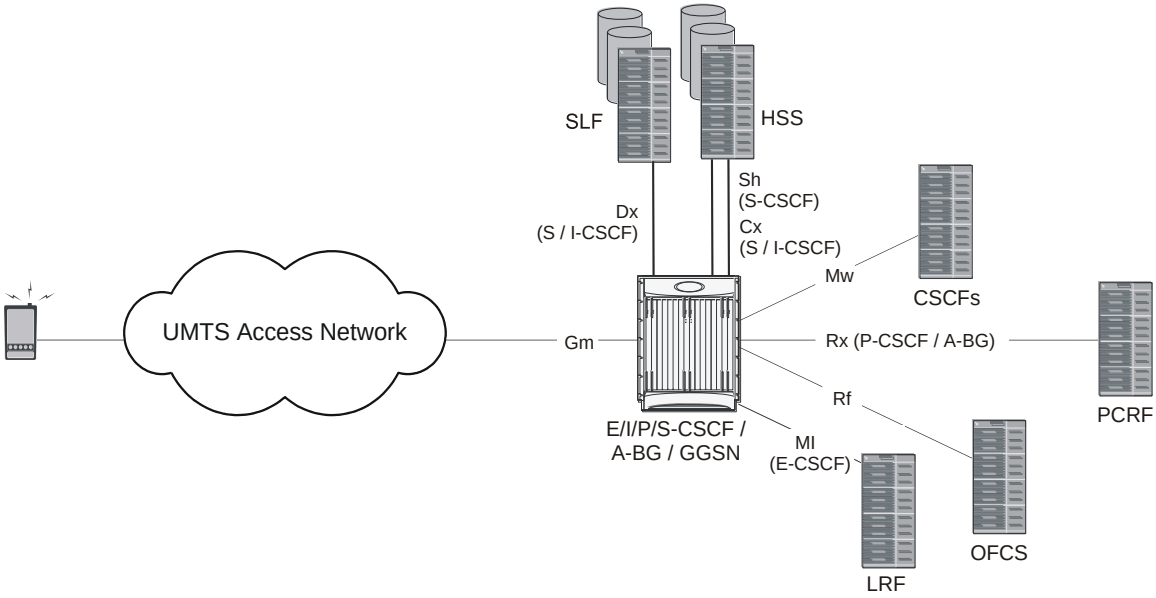
Interface	Description
Cx	The reference point between the S/I-CSCF and the Home Subscriber Server (HSS). The Cx interface is used to authenticate subscribers, provides server assignments, push user profile information from the HSS to the S-CSCF, and, when necessary, transmit a network initiated de-registration.
Dx	The reference point between the S/I-CSCF and Subscriber Location Function (SLF). The Dx interface is used to proxy queries to a subscriber data server (such as an HSS) in which subscription data for a user can be found. The SLF receives a query for the subscriber data server, looks up the address of appropriate subscriber data server, and proxies the query to the appropriate subscriber data server.
Rf	The reference point between the P-CSCF and the Offline Charging System (OFCS). The Rf interface is used to transfer charging information that will not affect, in real-time, the service being rendered. For more information, refer to the 3GPP2 specification X.S0013-007-A v1.0.
Sh	The reference point between the S-CSCF and Home Subscriber Server (HSS). The Sh interface is used for retrieval and update of call feature data parameters.
Tx	The reference point between the P-CSCF/A-BG and the Charging Rule Function (CRF)/Policy Decision Point (PDP) (PCRF) used for Service Based Bearer Control (SBBC). It identifies any P-CSCF/A-BG restrictions to be applied to the identified packet flows.

SCM in a GSM/UMTS Data Network Deployment

CSCF / A-BG / GGSN Deployment

The SCM is designed to function within a UMTS GGSN network. As shown in following figure, the SCM supports a number of interfaces used to communicate with other components in an IMS environment and supports the interface used to bridge the GGSN network.

Figure 5. GSM/UMTS CSCF/A-BG/GGSN SCM Deployment Example



Logical Network Interfaces (Reference Points)

Interfaces, used to support IMS in a UMTS network, can be defined within two categories: SIP and DIAMETER. The SCM incorporates standards-based interfaces for both SIP and DIAMETER network architectures.

SIP Interfaces

The following table provides descriptions of SIP interfaces supported by the SCM in a GSM/UMTS network deployment.

Table 4. SIP Interfaces in a GSM/UMTS Network

Interface	Description
Gm	The reference point between the P-CSCF and the User Equipment (UE). The Gm interface provides SIP signaling between the GGSN and the P-CSCF.
MI	The reference point between the E-CSCF and Location Retrieval Function (LRF). The MI interface is used for routing an emergency call to a Public Safety Answering Point (PSAP). The E-CSCF interacts with the Location Retrieval Function (LRF) to identify the next hop PSAP.

Interface	Description
Mw	The reference point between the P/S-CSCF and other CSCFs. The Mw interface provides SIP signaling between two CSCFs.

DIAMETER Interfaces

The following table provides descriptions of DIAMETER interfaces supported by the SCM in a GSM/UMTS network deployment.

Table 5. DIAMETER Interfaces in a GSM/UMTS Network

Interface	Description
Cx	The reference point between the S/I-CSCF and the Home Subscriber Server (HSS). The Cx interface is used to authenticate subscribers, provides server assignments, push user profile information from the HSS to the S-CSCF, and, when necessary, transmit a network initiated de-registration.
Dx	The reference point between the S/I-CSCF and Subscriber Location Function (SLF). The Dx interface is used to proxy queries to a subscriber data server (such as an HSS) in which subscription data for a user can be found. The SLF receives a query for the subscriber data server, looks up the address of appropriate subscriber data server, and proxies the query to the appropriate subscriber data server.
Rf	The reference point between the P-CSCF and the Offline Charging System (OFCS). The Rf interface is used to transfer charging information that will not affect, in real-time, the service being rendered. For more information, refer to the 3GPP2 specification X.S0013-007-A v1.0.
Rx	The reference point between the P-CSCF/A-BG and the Charging Rule Function (CRF)/Policy Decision Point (PDP) (PCRF). The Rx interface (3GPP 29.211) is used to exchange Flow Based Charging (FBC) control information between the PCRF and the P-CSCF/A-BG. The CRF uses the information to make FBC decisions that are then exchanged with the Traffic Plane Function (TPF). This interface is used in a 3GPP2 Release 7 implementation.
Sh	The reference point between the S-CSCF and Home Subscriber Server (HSS). The Sh interface is used for retrieval and update of call feature data parameters.

Voice over LTE (VoLTE)

CSCF Core / EPC Core Deployment

Mobile operators are migrating to the next generation 4G architecture based on Long Term Evolution (LTE) and the Evolved Packet Core (EPC). LTE/EPC supports only IP-based services, and it does not provide a method for legacy CS voice transport. The migration from circuit-based voice to packet voice and multimedia services is a key consideration in the successful deployment of an LTE/EPC solution. Operators must consider how to migrate and deploy an infrastructure that enables the introduction of a full suite of SIP-based services that provide subscribers with their existing voice and SMS services plus sets the framework for additional services, including video, Push to Talk over Cellular (PoC), IPTV, presence, and instant messaging.

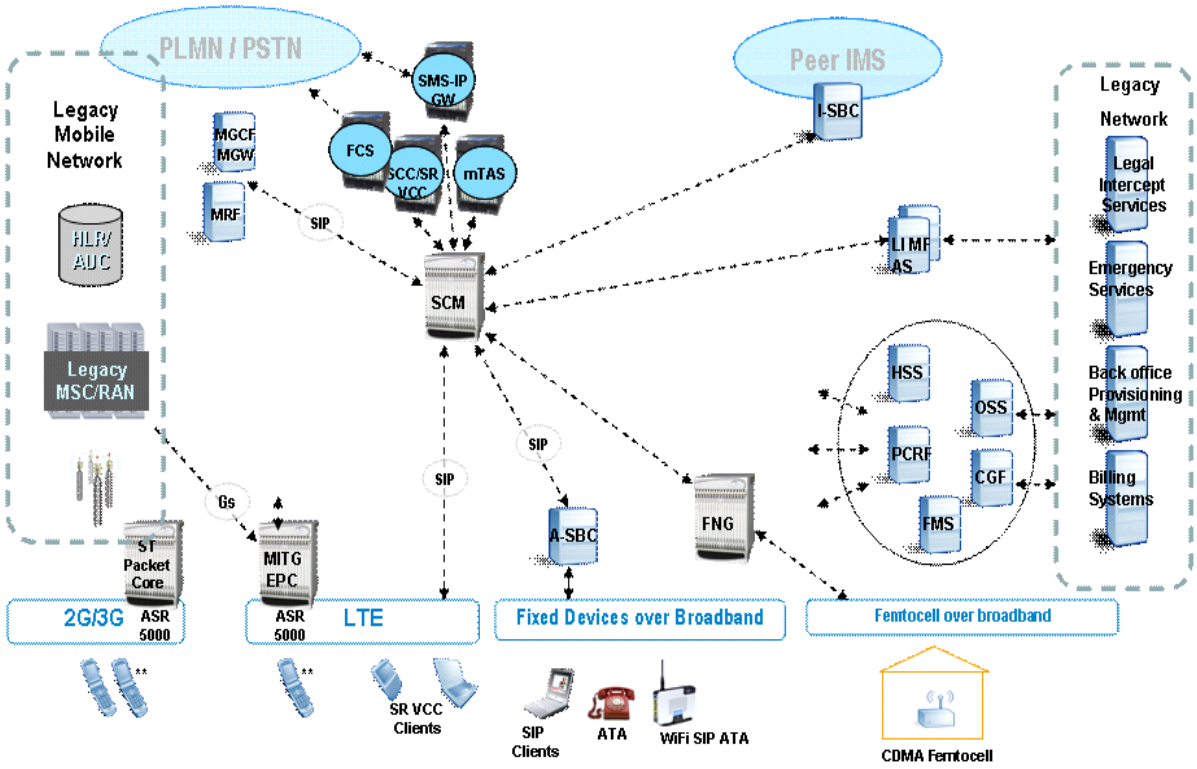
IMS has been chosen as the standard for providing circuit-based services over the all-IP LTE infrastructure. The long-term strategy based on IMS has been under standardization in 3GPP using MMTel TAS in conjunction with SCC server (TS 23.237) and the standard IMS core. In addition, the One Voice Initiative, a group of operators and carriers, has defined the preferred way to ensure the smooth introduction and delivery of voice and SMS services on LTE networks worldwide. One Voice aims to ensure compatibility between networks and devices by creating a common profile, which defines an optimal set of existing 3GPP functionalities for use by vendors and operators. The One Voice initiative has accelerated the move to an IMS solution for LTE networks.

Cisco's ASR 5000 chassis supports two major elements for the evolution of voice and SMS from the circuit network to the target network IMS. The ASR 5000 provides an LTE/EPC solution with high performance and integrated intelligence. The Cisco MME, as part of the ASR 5000, supports Circuit Switch Fallback as a baseline capability. In addition, the same ASR 5000 supports the full high performance IMS CSCF core (P/I/S/E-CSCF and BGCF) functionality. This functionality can be provided as a standalone function or integrated into the EPC functions to provide lower Total Cost of Ownership for the solution. For example, the P-GW and SCM can be integrated into a single multimedia core platform. This reduces the cost of entry and the transition to VoLTE, thus lowering the OPEX, plus reduces the number of network elements, network interfaces, and call set up latency.

Other features include:

- Easy on-ramp, with interworking of RFC3261 SIP and IMS SIP
- High availability, with intra/inter-chassis session recovery
- Intelligent integration
- IP mobility, with access-independent platform (mobile, WiFi, WiMAX, etc.)
- Performance and scalability
- Regulatory service support
    - Support for local number portability
    - Support for emergency call
    - Support for Lawful Intercept
- SIP routing engine
    - Secure and controlled deployment
    - SIP routing, translation, and monitoring
    - Support for route failover and back up route selection

Figure 6. VoLTE Deployment Example



Features and Functionality - Base Software

The following is a list containing a variety of features found in the SCM and the benefits they provide.

This section describes the following features:

- AS Selection
- Bulk Statistics Support
- Call Abort Handling
- Call Forking
- Call Types Supported
- Congestion Control
- DSCP Marking
- Early IMS Security
- Emergency Call Support
- Error Handling
- Future-proof Solution
- HSS Selection
- Intelligent Integration
- Interworking Function
- IPv6 Support
- Management System Overview
- MGCF Selection
- MSRP Support
- NPDB Support
- Presence Enabled
- Redirection
- Redundancy and Session Recovery
- Registration Event Package
- Signaling Compression (SigComp)
- SIP Denial of Service (DoS) Attack Prevention
- SIP Intelligence at the Core
- SIP Large Message Support
- SIP Routing Engine
- Shared Initial Filter Criteria (SiFC)
- Telephony Application Server (TAS) Basic Supported
- Threshold Crossing Alerts (TCA) Support
- TPS (Transaction per Second) Based Overload Control Towards AS

- Trust Domain

AS Selection

The S-CSCF may select the Application Server (AS) peer server group based on subscriber prefix, ip-type, or capability. The selected AS group should have an active AS list, standby AS list, and default AS list.

In addition, the S-CSCF is able to skip third party registration to the AS by a configured time after initial registration. After skipping the configured number of times, the third party register should be sent again to AS to reduce overload on AS.

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.

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. Following is a list of supported schemas for SCM:

- **Card**: Provides card-level statistics
- **Context**: Provides context-level statistics
- **CSCF**: Provides CSCF service statistics
- **CSCFINTF**: Provides CSCF interface statistics
- **Diameter-acct**: Provides Diameter Accounting statistics
- **Diameter-auth**: Provides Diameter Authentication statistics
- **Map**: Provides Map service statistics
- **Nat-realm**: Provides NAT realm statistics
- **Port**: Provides port-level statistics
- **System**: Provides system-level 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 system 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, system host name, system 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.

Important: For more information on bulk statistic configuration, refer to the *Configuring and Maintaining Bulk Statistics* chapter in the *System Administration Guide*.

Call Abort Handling

Call abort handling provides resource cleanup in error scenarios and makes sure resources that are not being used can be used for new calls. This feature is managed gracefully for a P-CSCF failure and CLI-initiated subscriber and session clean up.

Call Forking

Call forking allows subscribers to receive calls wherever they are by enabling multi-location UE registration.

Call Types Supported

In the IMS architecture, telephony features are normally provided by an external application server. Providing these features with the S-CSCF:

- Reduces the need for an additional SIP AS
- Simplifies the network architecture
- Improves latency for call setup and feature invocation

The following call types are supported:

- **Directory service, toll-free, long distance, international, and operator-assisted calls** - are supported through translation lists.
- **Emergency calls -** are managed through the addition of an Emergency Call/Session Control Function (E-CSCF) that routes emergency calls to a Public Safety Answering Point (PSAP).
- **Mobile-to-Mobile SIP calls** - supports SIP-based VoIP calls between mobile data users.
- **Public Switched Telephone Network (PSTN) calls** - can be routed through a 3GPP/2 compliant BGCF located in the S-CSCF.

Congestion Control

The congestion control feature allows you to set policies and thresholds and specify how the system reacts when faced with a heavy load condition.

Congestion control monitors the system for conditions that could potentially degrade performance when the system is under heavy load. Typically, these conditions are temporary (for example, high CPU or memory utilization) and are quickly resolved. However, continuous or large numbers of these conditions within a specific time interval may have an

impact the system's ability to service subscriber sessions. Congestion control helps identify such conditions and invokes policies for addressing the situation.

Congestion control operation is based on configuring the following:

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

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

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

CSCF performs congestion control based on the memory usage inside every sessmgr at two levels.

Level 1: For every new call/event received, the system checks if sessmgr memory-usage is above a threshold value (such as 95 percent). If it is, memory-congestion is triggered and new call messages are rejected with 500 SIP response. Memory congestion is disabled when memory usage drops by a tolerance value (default is 10 percent).

Level 2: If the sessmgr usage reaches 100 percent, all newly received SIP messages are dropped at the socket layer in that sessmgr except for the BYE message. The new SIP messages are not processed until the memory reaches the threshold value (95 percent).

A trap is also generated whenever sessmgr is in congestion state

Important: For more information on congestion control, refer to the *Congestion Control* chapter in the *Cisco ASR 5000 System Administration Guide*.

DSCP Marking

Provides support for more granular configuration of DSCP marking.

For Interactive Traffic class, the P-CSCF/A-BG supports per-service configurable DSCP marking for Uplink and Downlink direction based on Allocation/Retention Priority in addition to the current priorities.

The following matrix may be used to determine the Diffserv markings used based on the configured traffic class and Allocation/Retention Priority:

Table 6. Default DSCP Value Matrix

Allocation Priority	1	2	3
Traffic Handling Priority			
1	ef	ef	ef

Allocation Priority	1	2	3
2	af21	af21	af21
3	af21	af21	af21

Early IMS Security

Early IMS security allows authenticating the UE without IMS protocols and clients. Based on the 3GPP TR 33.978 specification, the SCM supports security inter-operation with 2G and non-IPSec user devices.

Emergency Call Support

P-CSCF gives priority to emergency calls, especially in a congested network. In addition, P-CSCF rejects new calls to any user who is in an emergency call.

Error Handling

The SCM supports consistent management of errors in a framework that considers existing and future standards and specifications.

Future-proof Solution

The SCM eliminates the capital and operational barriers associated with deploying traditional, server-based SIP proxies that lack carrier-class characteristics, occupy valuable rack space, and require numerous network interfaces, while also introducing additional control hops in the network that add call setup latency.

When operators deploy IMS/MMD, profitability will improve because a seamless on-ramp will be provided by simultaneously supporting 3GPP/3GPP2-based standards, P-CSCF functionality, and IETF SIP standards.

HSS Selection

This feature allows selection of multiple HSS within the same domain for different subscribers; this allows load distribution among multiple HSS. To select different HSS for different subscribers of the same domain, the S-CSCF allows configuration of matching criteria for selecting an AAA group name per subscriber.

When a subscriber registers, the selection criteria are compared and the AAA group name from the matching entry will be picked up. The selected AAA group will be used for all HSS interactions for that subscriber.

A maximum of three criteria can be configured per entry. A maximum of 1024 such entries can be configured.

HSS selection need not be done for Re-Register.

Intelligent Integration

For deployed platforms, no new hardware is necessary to install or manage. Functionality is enabled with a simple software download.

Intelligent integration lowers operational expenditure and reduces the number of network elements, network interfaces, and call setup latency.

Interworking Function

The SCM allows non-IMS UEs (pre IMS or RFC3261-compliant UEs) to work with the IMS core. When UEs are not IMS compliant, having this protocol interworking function at the edge allows the IMS core to be IMS compliant. After the interworking function inserts all necessary IMS headers toward the IMs core, the call appears to the IMS core network elements as if it is coming from an IMS-compliant UE.

The feature allows simultaneous support of IETF SIP and 3GPP/3GPP2 IMS/MMD clients.

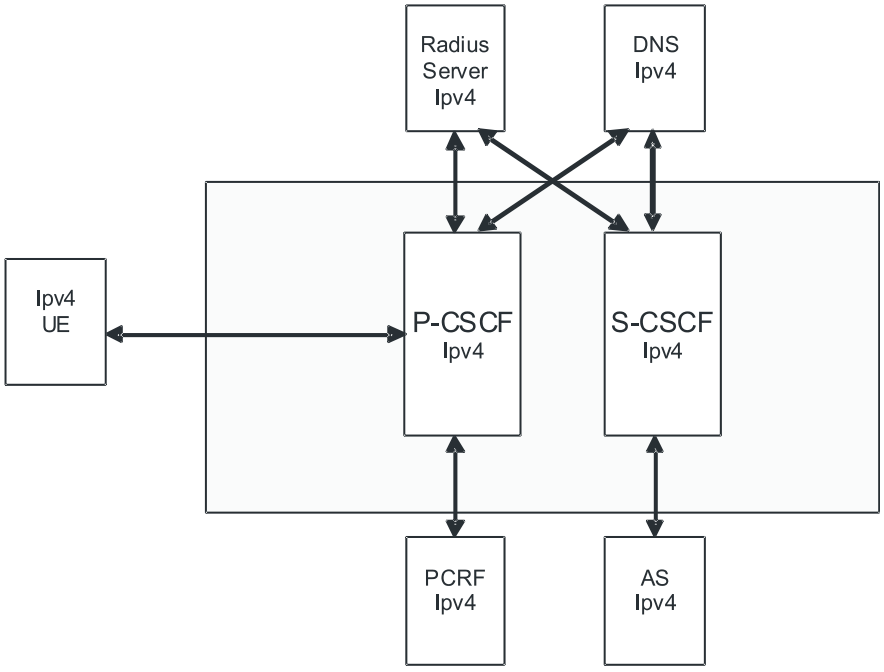
IPv6 Support

In addition to supporting IPv4, the SCM supports IPv6 addressing. A CSCF service can be configured with v6 addresses to support an all v6 network.

Important: For this feature, you may bind a CSCF service to either an IPv4 address or to an IPv6 address, but not both simultaneously.

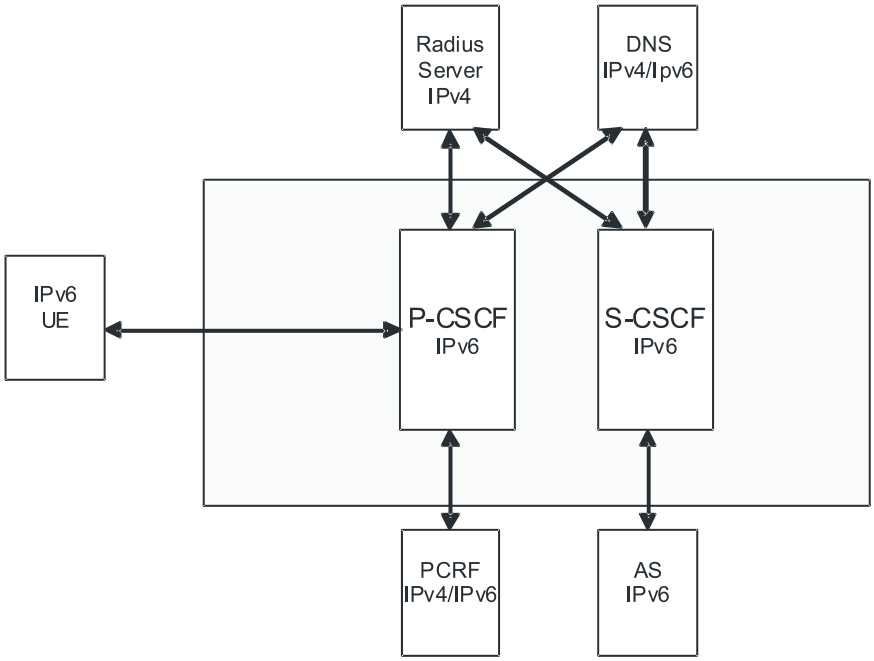
The following diagram shows the implementation where CSCF supports only IPv4.

Figure 7. IPv4 Configuration



With IPv6 support, the configuration supported would look like the following diagram. The DNS server could be either IPv4 or IPv6.

Figure 8. IPv6 Configuration



Important: The policy interface to PCRF will be IPv6 based when DIAMETER supports IPv6.

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 (Cisco 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.

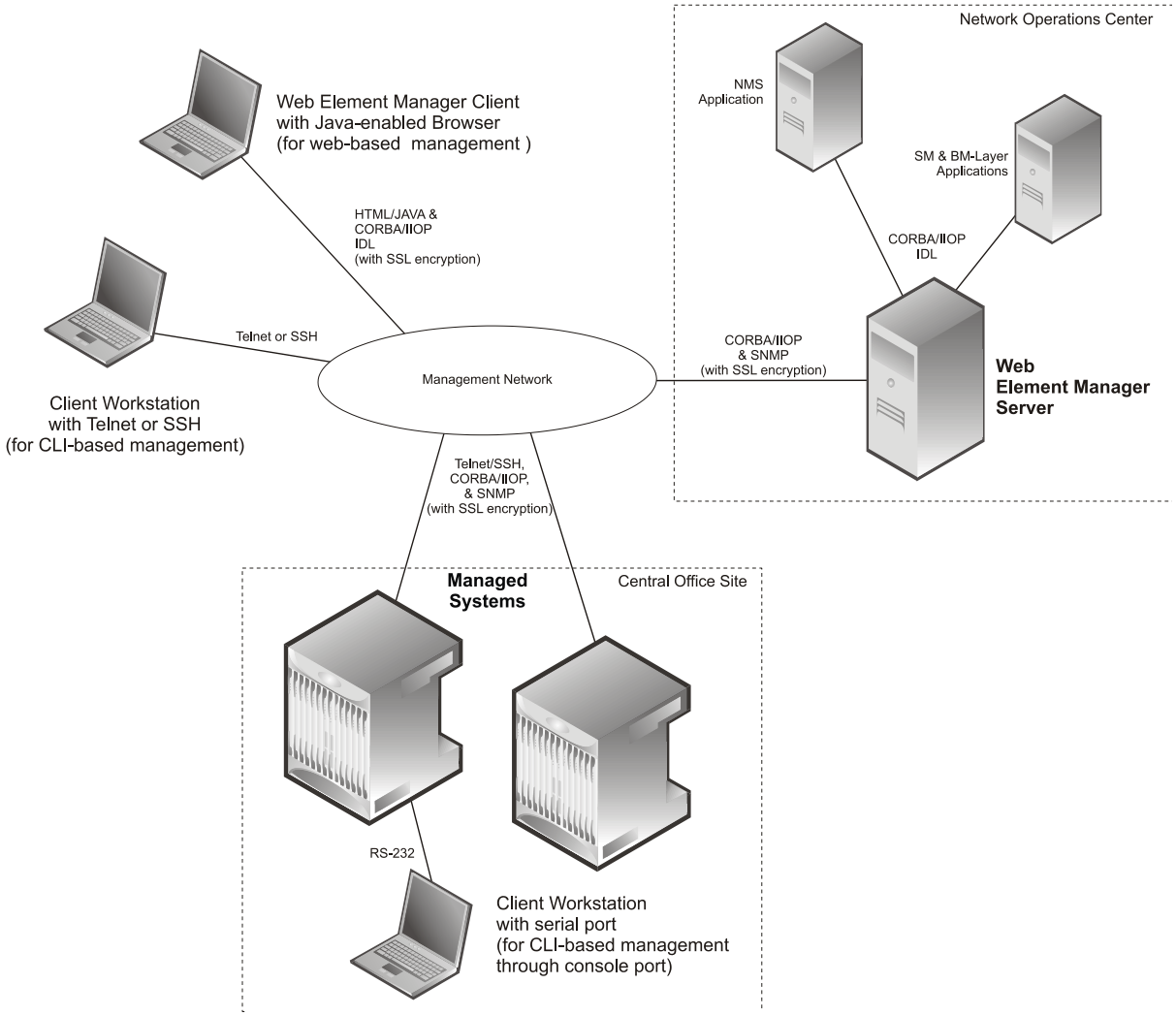
Cisco's O&M module offers comprehensive management capabilities to the operators and enables them to operate the system more efficiently. There are multiple ways to manage the system either locally or remotely using its out-of-band management interfaces.

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)

The following figure demonstrates these various element management options and how they can be utilized within the wireless carrier network.

Figure 9. Element Management Methods



Important: SCM management functionality is enabled by default for console-based access. For GUI-based management support, refer to the *Web Element Management System* section in this chapter.

Important: For more information on command line interface based management, refer to the *Command Line Interface Reference*.

MGCF Selection

MGCF selection is done based on the route configuration to select the next-hop-address, domain, or peer server.

Each record consists of one or more rules specifying the criteria that packets will be compared against. MGCF selection is based on subscriber prefix, ip-type, and accept-contact service-type criteria. While forwarding the message to external network element, the S-CSCF does the route lookup. S-CSCF applies the criteria configured to select the next-hop-

address. The criteria subscriber-ip-type will be matched for the Via address and subscriber-capability is applied for Accept-Contact header.

MSRP Support

The SCM supports Message Session Relay Protocol (MSRP) session and page modes.

NPDB Support

CSCF supports Local Number Portability (LNP), as per 3GPP standards, in which ENUM server is integrated with Number Portability Database (NPDB).

In addition, the S-CSCF supports a proprietary TCP/IP-based interface based on client server architecture to query an external NPDB.

Presence Enabled

With its high transaction setup rate, this is an ideal solution to handle a large number of messages generated by presence signaling. CSCF supports all the presence RFC extensions and signaling and interoperates with several presence servers.

Redirection

The SCM supports response to 3xx redirect messages. In addition to supporting redirection as per 3GPP, it supports call redirection to other chassis in the network (based on configuration) in case of system overload.

Redundancy and Session Recovery

When enabled, provides automatic failover of existing CSCF sessions due to hardware or software faults.

The system recovers from a single hardware or software fault with minimal interruption to the subscriber's service and maintains session information to rebuild sessions if multiple faults occur.

Registration Event Package

A set of event notifications used to inform SIP node of changes made to a registration.

Signaling Compression (SigComp)

SigComp compresses SIP call setup messages and is supported on the P-CSCF component. This reduces bandwidth demands on the RAN and reduces setup times.

SIP Denial of Service (DoS) Attack Prevention

The A-BG provides a scalable proxy network and a distributed Network Address Translation (NAT) network which effectively mitigates DoS attacks.

Prevents a variety of DoS attacks specific to CSCF and SIP technology.

SIP Intelligence at the Core

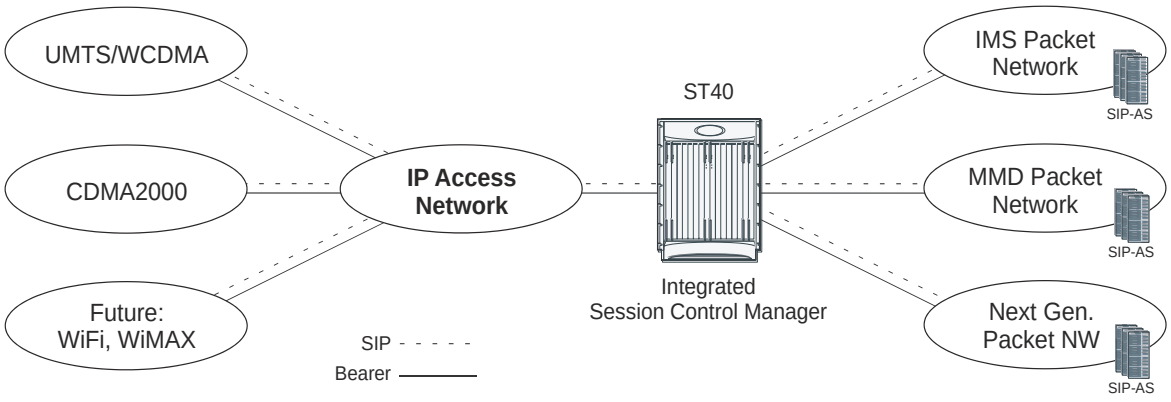
The SCM provides operators with an easy on-ramp for deploying SIP-based subscriber services while supporting various network control operations that provide the necessary intelligent control to insure a robust, carrier-class subscriber experience is achieved in this always changing multimedia environment.

When integrated into Cisco's session-aware Home Agent or GGSN platform, the SCM becomes the first SIP hop in the network, allowing operators to monitor and control all SIP-based sessions and execute additional value-added functions.

As the logical anchor point within the packet core, the SCM improves the user experience with device and location independence, and enhances subscriber control and policy enforcement with faster, more intelligent decisions for multimedia services.

Furthermore, as Fixed Mobile Convergence takes hold, it will be especially important to incorporate the SCM in the packet core in order to achieve mobility and voice continuity between multiple access networks (3G, WiFi, WiMAX, etc.).

Figure 10. Cisco Integrated Session Control Manager



SIP Large Message Support

Large notify contains information about multiple users in one message, which reduces the number of SIP messages in the network. Large SIP messages can be sent on UDP if the endpoint can support fragmentation; otherwise, UDP to TCP switching can be used to transport large messages intact.

If a request is within 200 bytes of the path MTU, or if it is larger than 1300 bytes and the path MTU is unknown, the request MUST be sent using TCP. This prevents fragmentation of messages over UDP and provides congestion control for larger messages. P-CSCF/A-BG is also able to handle messages up to the maximum datagram packet size. For UDP, this size is 65,535 bytes, including IP and UDP headers.

Large message support is needed for handling presence signaling traffic as the size of messages could be as large as 50K.

SIP Routing Engine

The SIP routing engine deploys SIP in a secure and controlled fashion.

Provides auto discovery of SIP elements, subscriber privacy, call fraud prevention, network security, and thwarting of network overload conditions.

Shared Initial Filter Criteria (SiFC)

If both the HSS and the S-CSCF support this feature, subsets of iFC may be shared by several service profiles. The HSS downloads the unique identifiers of the shared iFC sets to the S-CSCF. The S-CSCF uses a locally administered database to map the downloaded identifiers onto the shared iFC sets.

If the S-CSCF does not support this feature, the HSS will not download identifiers of shared iFC sets.

Telephony Application Server (TAS) Basic Supported

The following describe the local basic call features implemented on the S-CSCF:

- Abbreviated Dialing (AD)
- Call Forward Busy Line (CFBL)
- Call Forward No Answer (CFNA)
- Call Forward Not Registered (CFNR)
- Call Forward Unconditional (CFU)
- Call Transfer
- Call Waiting
- Caller ID Display (CID)
- Caller ID Display Blocked (CIDB)
- Feature Code Activation/De-activation
- Follow Me/Find Me
- Locally Allowed Abbreviated Dialing
- Outbound Call Restrictions/Dialing Permissions
- Short Code Dialing

TAS Basic provides basic voice call feature support in the SCM. In the IMS architecture, these telephony features are normally provided by an external application server. Providing these features with the S-CSCF:

- Reduces the need for an additional SIP AS
- Simplifies the network architecture
- Improves latency for call setup and feature invocation

The following describe the local basic call features implemented on the S-CSCF:

- **Abbreviated Dialing (AD)** - This feature allows the subscriber to call a Directory Number by entering less than the usual ten digits.Usually, the subscriber has four digit dialing to mimic PBX dialing privileges but these must be set up prior to use. When the SCM receives these numbers, it translates them and routes the call.

- **Call Forward Busy Line (CFBL)** - This feature forwards the call if busy line indication is received from the UE. If CFBL is enabled on both the AS and the S-CSCF, the call is forwarded by the S-CSCF on Busy Line indication. The feature detects and eliminates call forward loops if the History-Info header is present. It also terminates forwarding if forwarding causes the forward attempts to be more than the number specified in the Max-Forwards header.
- **Call Forward No Answer (CFNA)** - This feature forwards the call if no answer is received from the UE. If CFNA is enabled on both the AS and the S-CSCF, the call is forwarded by the S-CSCF on No Answer indication. The feature detects and eliminates call forward loops if the History-Info header is present. It also terminates forwarding if forwarding causes the forward attempts to be more than the number specified in the Max-Forwards header.
- **Call Forward Not Registered (CFNR)** - This feature forwards the call if the subscriber is not registered. If CFNR is enabled on both the AS and the S-CSCF, the call is forwarded by the S-CSCF on Not Registered indication. The feature detects and eliminates call forward loops if the History-Info header is present. It also terminates forwarding if forwarding causes the forward attempts to be more than the number specified in the Max-Forwards header.
- **Call Forward Unconditional (CFU)** - This feature unconditionally forwards the call. The check for local CFU is done prior to the filter criteria and before any AS interaction. Thus CFU is enabled on both the S-CSCF and the destination AS, the local CFU occurs and there is no AS interaction. The feature eliminates basic loop detection (A calls B which is forwarded to A) and if the History-Info header is present, enhanced loop detection is performed based on the contents of this header. It also terminates forwarding if forwarding causes the forward attempts to be more than the number specified in the Max-Forwards header.
- **Call Transfer** - This feature allows the subscriber to transfer a call.
- **Call Waiting** - This feature allows the subscriber to receive a second call while on the first call.
- **Caller ID Display (CID)** - This feature inserts P-Preferred-Identity which communicates the identity of the user within the trust domain. If this header is already present, the feature may not do anything different.
- **Caller ID Display Blocked (CIDB)** - This feature removes P-Preferred-Identity and P-Preferred-Asserted-Identity headers and inserts a Privacy header with the privacy value set to “id”.
- **Feature Code Activation/De-activation** - This feature allows for activating and de-activating certain features using a star (*) - number sequence (star code). Registered subscribers have the option of activating or deactivated call features using specified star codes. The SCM translates these codes and routes the call.
- **Follow Me/Find Me** - This feature invokes the incoming call to several configured destinations in parallel and connects the call to the first destination that responds, “tearing down” all the other calls. There are two possible implementations of this feature; one a sequential implementation in which each destination is attempted in sequence till a successful connection. The other is a parallel approach in which several destinations are tried simultaneously. The advantage of the parallel approach is a faster set up.
- **Locally Allowed Abbreviated Dialing** - This feature allows the subscriber to dial a local-only, legacy, short code such as *CG or *POL. The SCM translates these codes to a ten-digit directory number and routes the call.
- **Outbound Call Restrictions/Dialing Permissions** - This feature restricts subscribers from initiating certain outbound calls. For example, if a subscriber attempts to make an international call and is not permitted to, the S-CSCF rejects the call.
- **Short Code Dialing** - This feature allows the subscriber to dial a short code such as #PAY or #MIN. The SCM translates these codes and routes the call.

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.

The following thresholding models are supported by the system:

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

Thresholding reports conditions using one of the following mechanisms:

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

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

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

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

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

TPS (Transaction per Second) Based Overload Control Towards AS

S-CSCF can load balance among multiple AS nodes. Each AS serves a set of subscribers, and subscribers are assigned to AS based on prefix and capabilities. In spite of this distribution, there could be situations where AS might get more messages than it can handle during peak network traffic events and due to high performance of S-CSCF. To handle such situations, a rate control mechanism has been implemented in S-CSCF. The rate control is configured as TPS value per AS. S-CSCF is expected not to send more than the configured TPS messages to the node.

Trust Domain

Enables the identification of trusted network entities. This keeps subscriber information confidential when it is received.

Features and Functionality - External Application Support

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

- Web Element Management System

Web Element Management System

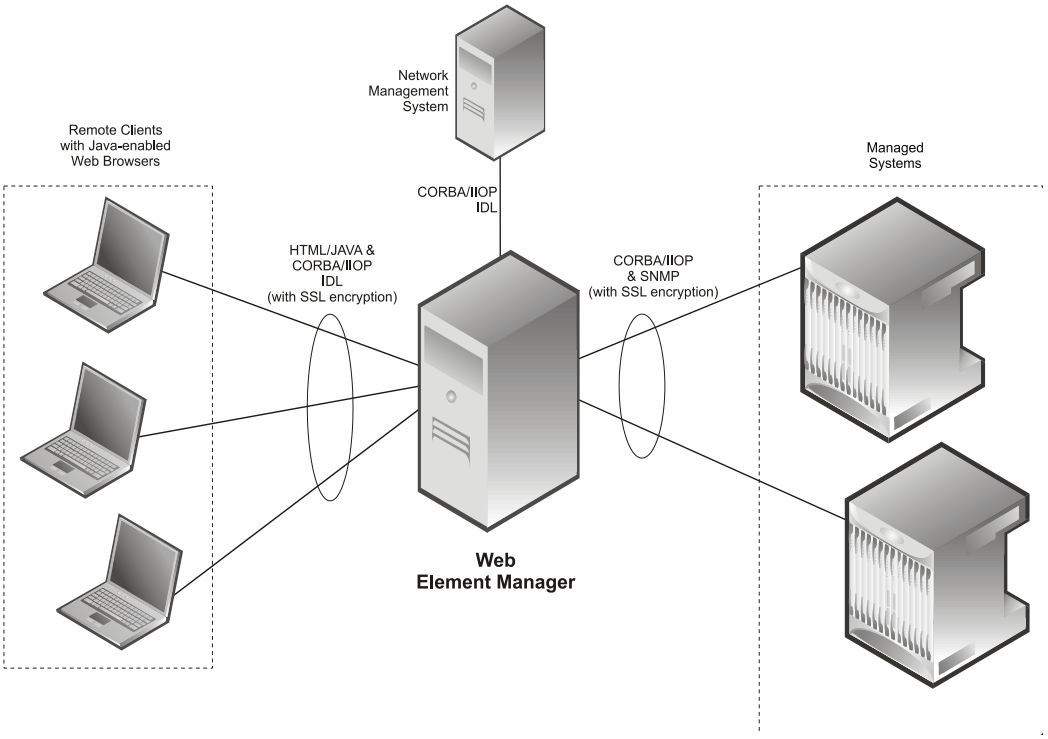
Provides a graphical user interface (GUI) for performing fault, configuration, accounting, performance, and security (FCAPS) management of the ASR 5000.

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.

The following figure demonstrates various interfaces between the Cisco Web Element Manager and other network components.

Figure 11. Web Element Manager Network Interfaces



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

Features and Functionality - Licensed Enhanced Feature Support

This section describes optional enhanced features and functions.

Each of the following optional enhanced features require the purchase of an additional license to implement the functionality with the SCM.

This section describes the following features:

- Interchassis Session Recovery
- IPSec Support
- IPv4-IPv6 Interworking
- Lawful Intercept
- Session Recovery Support
- TLS Support in P-CSCF

Interchassis Session Recovery

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

The ASR 5000 provides industry leading carrier class redundancy. The systems protects against all single points of failure (hardware and software) and attempts to recover to an operational state when multiple simultaneous failures occur.

The system provides several levels of system redundancy:

- Under normal N+1 PSC/PSC2/PSC3 hardware redundancy, if a catastrophic packet processing card failure occurs all affected calls are migrated to the standby packet processing card if possible. Calls which cannot be migrated are gracefully terminated with proper call-termination signaling and accounting records are generated with statistics accurate to the last internal checkpoint
- If the Session Recovery feature is enabled, any total packet processing card failure will cause a packet processing card switchover and all established sessions for supported call-types are recovered without any loss of session.

Even though Cisco Systems provides excellent intra-chassis redundancy with these two schemes, certain catastrophic failures which can cause total chassis outages, such as IP routing failures, line-cuts, loss of power, or physical destruction of the chassis, cannot be protected by this scheme. In such cases, the Interchassis Session Recovery feature provides geographic redundancy between sites. This has the benefit of not only providing enhanced subscriber experience even during catastrophic outages, but can also protect other systems such as the RAN from subscriber re-activation storms.

The Interchassis Session Recovery feature allows for continuous call processing without interrupting subscriber services. This is accomplished through the use of redundant chassis. The chassis are configured as primary and backup with one being active and one in recovery mode. A checkpoint duration timer is used to control when subscriber data is sent from the active chassis to the inactive chassis. If the active chassis handling the call traffic goes out of service, the inactive chassis transitions to the active state and continues processing the call traffic without interrupting the subscriber session. The chassis determines which is active through a propriety TCP-based connection called a redundancy link.

This link is used to exchange Hello messages between the primary and backup chassis and must be maintained for proper system operation.

- **Interchassis Communication**

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

  - router identifier
  - chassis priority
  - SPIO MAC address

- **Checkpoint Messages**

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

Important: For more information on interchassis session recovery support, refer to the *Interchassis Session Recovery* chapter in the *Cisco ASR 5000 Series System Administration Guide*.

IPSec Support

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

Encrypted IPSec tunnels are terminated and decrypted so that traffic coming from untrusted networks are secured before entering the secure operator network. This prevents eavesdropping, hijacking, and other intrusive behavior from occurring.

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.

Important: IPSec implementation is a mandatory part of IPv6, but it is optional to secure IPv4 traffic.

Important: For more information on IPSec support, refer to the *IP Security* appendix in the *Cisco ASR 5000 Series Session Control Manager Administration Guide*.

IPv4-IPv6 Interworking

Use of IPv4-IPv6 interworking requires that a valid license key be installed. Contact your local Sales or Support representative for information on how to obtain a license.

This feature allows the P-CSCF to provide IPv4-IPv6 interworking in the following scenarios:

- When UEs are IPv6-only and the IMS core network is IPv4-only
- When UEs are IPv4-only and the IMS core network is IPv6-only

In addition, IPv4-IPv6 interworking helps an IPv4 IMS network transition to an all-IPv6 IMS network.

The following interworking requirements are currently supported:

- MSRP support when IPv4-IPv6 interworking is enabled
- IPv4 TCP and IPv6 TCP
- Transport switching allowed based on size for both v4 and v6 network
- UDP fragmentation allowed for both v4 and v6 networks
- P-CSCF supports Mw and Gm interfaces on both v4 and v6
- KPIs for Mw and Gm interfaces are supported on both v4 and v6
- DNS supported for v4 and v6 networks
- Interworking supported for IM and presence
- Both v4 and v6 handsets are supported simultaneously on the same P-CSCF node

P-CSCF will provide IPv4-IPv6 interworking functionality between IPv6-only UEs and IPv4-only core network elements (I/S-CSCF) by acting as a dual stack. To achieve the dual-stack behavior, P-CSCF will be configured in two services with the first service (V6-SVC) listening on an IPv6 address and the second service (V4-SVC) listening on an IPv4 address. SIP messages coming from IPv6 UEs will come to V6-SVC and will be forwarded to the IPv4 core network through V4-SVC. Similarly, messages from the IPv4 core network come to V4-SVC and will be forwarded to IPv6 UEs via V6-SVC. P-CSCF also provides interworking functionality between IPV4-only UEs and IPv6-only core network elements.

P-CSCF handling different v4-v6 interworking scenarios is shown below.

Figure 12. Interworking Between IPv6 UE and IPv4 IMS Core Network

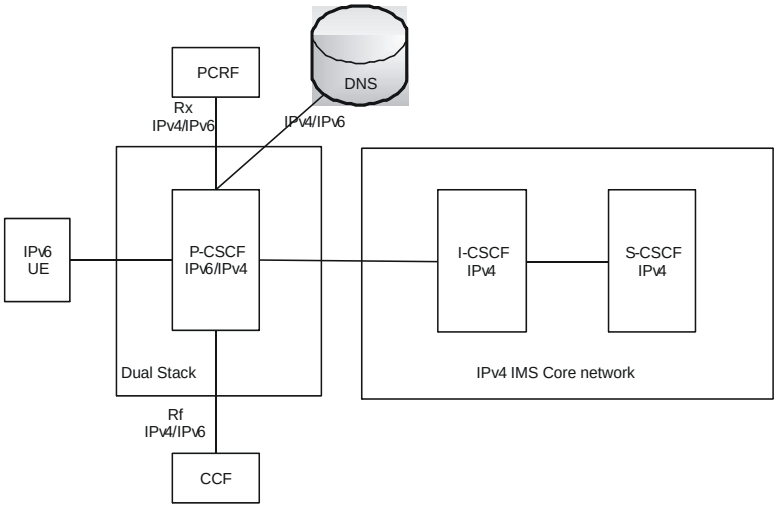
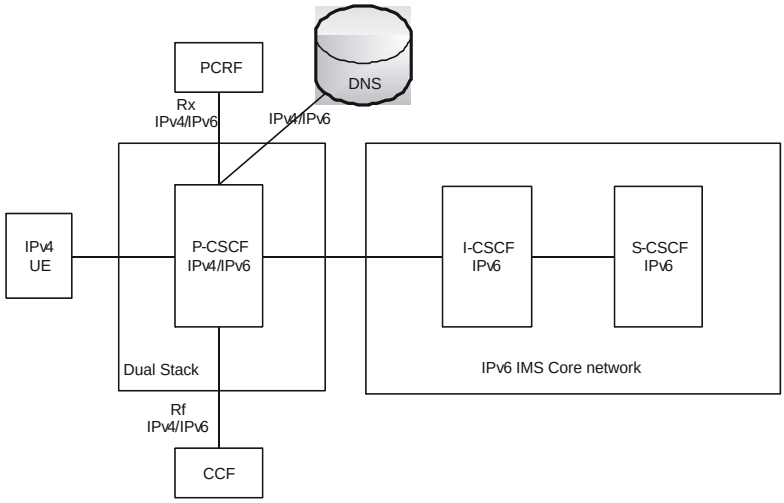


Figure 13. Interworking Between IPv4 UE and IPv6 IMS Core Network



To identify the need for IPv4-IPv6 interworking for a new incoming IPv6 REGISTER arriving at V6-SVC, a route lookup is performed based on the request-uri, first in V4-SVC context and then in V6-SVC context if the first lookup does not return any matching route entry. If a matching IPv4 next-hop route entry is found, then this indicates that interworking needs to be done. If no route entry is found, then a DNS query on request-uri domain is done for both A and AAAA type records. If DNS response yields only an IPv4 address, then this is also the case for performing IPv4-IPv6 interworking.

Headers (such as Via, Path, etc.) are automatically set to IPv4 bind address of P-CSCF V4-SVC. Remaining headers will be not be altered and sent as is toward the S-CSCF. The IPv4 address in a Path header received from S-CSCF in 200Ok of REGISTER will be replaced with V6-SVC's IPv6 address before forwarding to UE.

Lawful Intercept

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

The Cisco Lawful Intercept feature is supported on the SCM. Lawful Intercept is a licensed-enabled, standards-based feature that provides telecommunications service providers with a mechanism to assist law enforcement agencies in monitoring suspicious individuals for potential illegal activity. For additional information and documentation on the Lawful Intercept feature, contact your Cisco account representative.

Session Recovery Support

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

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.

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/PSC2/PSC3) 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/PSC2/PSC3 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 Management Card (SMC) and a standby PSC/PSC2/PSC3.

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 PSCs. The "standby-mode" task is renamed, made active, and is then populated using information from other tasks such as AAA manager.
- **Full PSC/PSC2/PSC3 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 PSCs/PSC2s/PSC3s to ensure task recovery.

Important: Session Recovery is supported for either IPv4 or IPv6 traffic.

Important: For more information on session recovery support, refer to the *Session Recovery* chapter in the *Cisco ASR 5000 Series System Administration Guide*.

TLS Support in P-CSCF

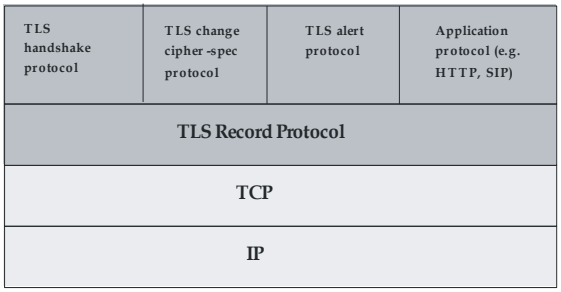
Use of SSL requires that a valid license key be installed. Contact your local Sales or Support representative for information on how to obtain a license.

Transport Layer Security (TLS) provides confidentiality and integrity protection for SIP signaling messages between the UE and P-CSCF/A-BG. TLS is a layered protocol that runs upon reliable transport protocols like TCP and SCTP.

The SCM supports the following two scenarios:

- TLS as a transport between UE and P-CSCF/A-BG, as per RFC 3261
- Use of TLS by Security Mechanism agreement between UE and P-CSCF/A-BG, as per RC 3329 and TS 33.203

The following figure shows the TLS protocol layers.



Important: For more information on TLS support, refer to the *TLS Support* appendix in the *Cisco ASR 5000 Series Session Control Manager Administration Guide*.

How the SCM Works

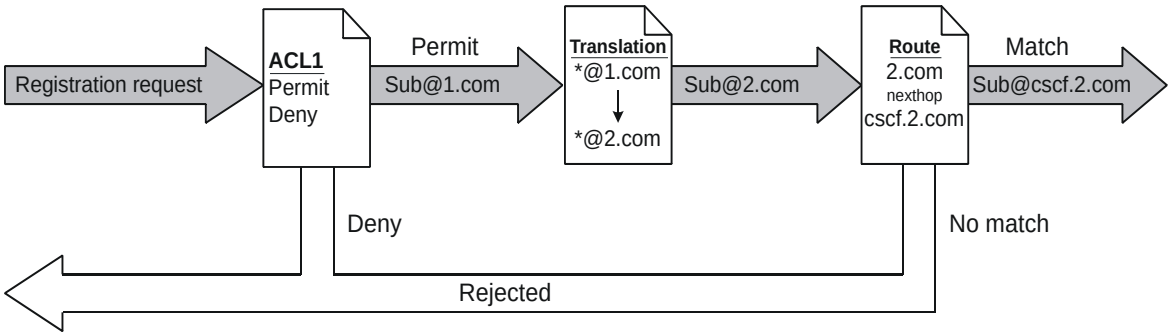
This section provides information on the function of the SCM in a CDMA2000 PDSN or UMTS GGSN network and presents call procedure flows for different stages of session setup.

Admission and Routing

Admission and routing of subscriber URIs is performed through a number of configurable lists in the SCM.

The following sections describe the main admission and routing techniques used in the SCM. The following figure presents the method and order for admitting and routing sessions within the SCM.

Figure 14. Admission and Routing Method



CSCF Access Control Lists

Access Control Lists (ACLs) are a set of rules that are applied during CSCF session establishment. A typical use of these rules is to accept or deny registration or session establishment requests. ACLs may be tied to subscribers and/or the whole service. Subscriber based ACLs can also be imported from an external ACL/policy server. In that event, the external policy server address would be configured with the service.

A complete explanation of the ACL configuration method is located in *Access Control Lists* appendix in the *Cisco ASR 5000 Series Session Control Manager Administration Guide*.

Translation Lists

Translation lists help modify request-uri (i.e. addressing of a CSCF session). One example is that E.164 numbers could be altered by adding prefixes and suffixes or the request-uri could be modified based on the registration database.

Route Lists

Route lists are service level lists that assist in finding the next CSCF/UA hop. These are static routes and will override any dynamic routes (based on DNS queries for FQDNs).

Signaling Compression

The Session Initiation Protocol (SIP) is a text-based protocol designed for higher bandwidth networks. As such, it is inherently less suited for lower bandwidth environments such as wireless networks. If a wireless handset uses SIP to set up a call, the setup time is significantly increased due to the high overhead of text-based signaling messages.

Signaling Compression (SigComp) is a solution for compressing/decompressing messages generated by application protocols such as SIP. The P-CSCF component of the SCM uses SigComp to reduce call setup times on the access network, typically between the P-CSCF and the UE. The following features are supported:

- **SigComp Detection** - P-CSCF detects if the UE supports SigComp and compresses messages it sends to the UE. The P-CSCF also detects if messages it receives are compressed and decompresses them.
- **SigComp Parameter Configuration** - P-CSCF allows the configuration of Decompression Memory Size (DMS), State Memory Size (SMS), and Cycles Per Bit (CPB).
- **Failure Acknowledgement** - P-CSCF replies with NACK on decompression failure.
- **SIP/SDP Static Dictionaries** - P-CSCF supports the Session Initiation Protocol/Session Description Protocol Static Dictionary for Signaling Compression.

Supported Standards

The SCM service complies with the following standards for CDMA2000 PDSN, UMTS GGSN, and LTE network wireless data services.

Release 9 3GPP References

Important: The SCM currently supports the following Release 9 3GPP specifications. Most 3GPP specifications are also used for 3GPP2 support; any specifications that are unique to 3GPP2 would be listed under 3GPP2 References.

- TS 23.167 IP Multimedia Subsystem (IMS) emergency sessions
- TS 23.204 Support of Short Message Service (SMS) over generic 3GPP Internet Protocol (IP) access; Stage 2
- TS 23.207 End-to-end Quality of Service (QoS) concept and architecture
- TS 23.228 IP Multimedia Subsystem (IMS); Stage 2
- TS 23.981 Interworking aspects and migration scenarios for IPv4-based IP Multimedia Subsystem (IMS) implementations
- TS 24.229 Internet Protocol (IP) multimedia call control protocol based on Session Initiation Protocol (SIP) and Session Description Protocol (SDP); Stage 3
- TS 24.341 Support of SMS over IP networks; Stage 3
- TS 29.208 End-to-end Quality of Service (QoS) signalling flows
- TS 29.214 Policy and charging control over Rx reference point
- TS 29.228 IP Multimedia (IM) Subsystem Cx and Dx interfaces; Signalling flows and message contents
- TS 29.229 Cx and Dx interfaces based on the Diameter protocol; Protocol details
- TS 32.240 Telecommunication management; Charging management; Charging architecture and principles
- TS 32.260 Telecommunication management; Charging management; IP Multimedia Subsystem (IMS) charging
- TS 33.203 3G security; Access security for IP-based services
- TS 33.978 Security aspects of early IP Multimedia Subsystem (IMS)

Release 8 3GPP References

Important: The SCM currently supports the following Release 8 3GPP specifications. Most 3GPP specifications are also used for 3GPP2 support; any specifications that are unique to 3GPP2 are listed under 3GPP2 References.

- TR 23.806 Voice call continuity between Circuit Switched (CS) and IP Multimedia Subsystem (IMS) Study
- TR 23.808 Supporting Globally Routable User Agent URI (GRUU) in IMS; Report and conclusions
- TR 23.816 Identification of Communication Services in IMS
- TS 24.229 IP multimedia call control protocol based on Session Initiation Protocol (SIP) and Session Description Protocol (SDP); Stage 3

- TR 24.930 IP Multimedia core network Subsystem (IMS) based on Session Initiation Protocol (SIP) and Session Description Protocol (SDP); Stage 3
- TR 29.847 Conferencing based on SIP, SDP, and other protocols; Functional models, information flows and protocol details
- TR 33.978 Security aspects of early IP Multimedia Subsystem (IMS)
- TS 22.101 Service principles
- TS 23.003 Numbering, addressing and identification
- TS 23.107 Quality of Service (QoS) concept and architecture
- TS 23.125 Overall high level functionality and architecture impacts of flow based charging; Stage 2
- TS 23.141 Presence service; Architecture and functional description; Stage 2
- TS 23.167 IP Multimedia Subsystem (IMS) emergency sessions
- TS 23.203 Policy and charging control architecture
- TS 23.204 Support of Short Message Service (SMS) over generic 3GPP Internet Protocol (IP) access; Stage 2
- TS 23.207 End-to-end Quality of Service (QoS) concept and architecture
- TS 23.218 IP Multimedia (IM) session handling; IM call model; Stage 2
- TS 23.221 Architectural Requirements
- TS 23.228 IP Multimedia Subsystem (IMS); Stage 2
- TS 23.271 Functional description of Location Services (LCS)
- TS 23.981 Interworking aspects and migration scenarios for IPv4-based IP Multimedia Subsystem (IMS) implementations
- TS 24.141 Presence service using the IP Multimedia (IM) Core Network (CN) subsystem; Stage 3
- TS 24.228 Signalling flows for the IP multimedia call control based on Session Initiation Protocol (SIP) and Session Description Protocol (SDP); Stage 3
- TS 24.229 Internet Protocol (IP) multimedia call control protocol based on Session Initiation Protocol (SIP) and Session Description Protocol (SDP); Stage 3
- TS 24.341 Support of SMS over IP networks; Stage 3
- TS 26.114 IP Multimedia Subsystem (IMS); Multimedia telephony; Media handling and interaction
- TS 26.141 IP Multimedia System (IMS) Messaging and Presence; Media formats and codecs
- TS 26.234 Transparent end-to-end Packet-switched Streaming Service (PSS); Protocols and codecs
- TS 26.235 Packet switched conversational multimedia applications; Default codecs
- TS 26.236 Packet switched conversational multimedia applications; Transport protocols
- TS 29.207 Policy control over Go interface
- TS 29.208 End-to-end Quality of Service (QoS) signalling flows
- TS 29.209 Policy control over Gq interface
- TS 29.213 Policy and charging control signalling flows and Quality of Service (QoS) parameter mapping
- TS 29.214 Policy and charging control over Rx reference point
- TS 29.228 IP Multimedia (IM) Subsystem Cx and Dx interfaces; Signalling flows and message contents
- TS 29.229 Cx and Dx interfaces based on the Diameter protocol; Protocol details

- TS 29.328 IMS Sh interface: signalling flows and message content
- TS 29.329 IMS Sh interface based on the Diameter protocol; Protocol details
- TS 31.103 Characteristics of the IMS Identity Module (ISIM) application
- TS 32.225 Telecommunication management; Charging management; Charging data description for the IP Multimedia Subsystem (IMS)
- TS 32.240 Telecommunication management; Charging management; Charging architecture and principles
- TS 32.260 Telecommunication management; Charging management; IP Multimedia Subsystem (IMS) charging
- TS 32.299 Telecommunication management; Charging management; Diameter charging applications
- TS 33.102 3G security; Security architecture
- TS 33.178 Security aspects of early IP Multimedia Subsystem (IMS)
- TS 33.203 3G security; Access security for IP-based services
- TS 33.978 Security aspects of early IP Multimedia Subsystem (IMS)

3GPP2 References

- S.R0079-A v1.0 Support for End-to-End QoS - Stage 1 Requirements
- S.R0086-A v1.0 IMS Security Framework
- X.S0013-000-A v1.0 All-IP Core Network Multimedia Domain - Overview
- X.S0013-002-A v1.0 All-IP Core Network Multimedia Domain - IP Multimedia Subsystem Stage 2
- X.S0013-003-0 v2.0 All-IP Core Network Multimedia Domain - IP Multimedia (IMS) Session Handling; IP Multimedia (IM) Call Model - Stage 2
- X.S0013-004-A v1.0 All-IP Core Network Multimedia Domain - IP Multimedia Call Control Protocol Based on SIP and SDP Stage 3
- X.S0013-005-0 All-IP Core Network Multimedia Domain: IP Multimedia Subsystem Cx Interface Signaling Flows and Message Contents
- X.S0013-006-0 All-IP Core Network Multimedia Domain - Cx Interface Based on the Diameter Protocol; Protocol Details
- X.S0013-007-0 All-IP Core Network Multimedia Domain: IP Multimedia Subsystem - Charging Architecture
- X.S0013-007-A v1.0 All-IP Core Network Multimedia Domain - IP Multimedia Subsystem - Charging Architecture
- X.S0013-008-0 All-IP Core Network Multimedia Domain: IP Multimedia Subsystem - Accounting Information Flows and Protocol
- X.S0013-008-A All-IP Core Network Multimedia Domain - IP Multimedia Subsystem - Offline Accounting Information Flows and Protocol
- X.S0013-010-0 v1.0 All-IP Core Network Multimedia Domain: IP Multimedia Subsystem Sh Interface; Signaling Flows and Message Contents - Stage 2
- X.S0013-011-0 v1.0 All-IP Core Network Multimedia Domain: Sh Interface Based on Diameter Protocols Protocol Details - Stage 3
- X.S0013-012-0 v1.0 All-IP Core Network Multimedia Domain - Service Based Bearer Control - Stage 2

- X.S0013-014-0 v1.0 All-IP Core Network Multimedia Domain - Service Based Bearer Control - Tx Interface Stage 3
- X.S0016-000-A v1.0 3GPP2 Multimedia Messaging System MMS Specification Overview, Revision A
- X.S0027-002-0 v1.0 Presence Security
- X.S0027-003-0 v1.0 Presence Stage 3
- X.S0029-0 v1.0 Conferencing Using the IP Multimedia (IM) Core Network (CN) Subsystem
- X.S0049-0 v1.0 All-IP Network Emergency Call Support

IETF References

- RFC 1594 (March 1994): "FYI on Questions and Answers to Commonly Asked "New Internet User" Questions"
- RFC 1889 (January 1996): "RTP: A Transport Protocol for Real-Time Applications"
- RFC 2246 (January 1999): "TLS protocol version 1.0"
- RFC 2327 (April 1998): SDP: Session Description Protocol
- RFC 2401 (November 1998): "Security Architecture for the Internet Protocol (IPSec)"
- RFC 2403 (November 1998): "The Use of HMAC-MD5-96 within ESP and AH"
- RFC 2404 (November 1998): "The Use of HMAC-SHA-1-96 within ESP and AH"
- RFC 2462 (December 1998): "IPv6 Address Autoconfiguration"
- RFC 2617 (June 1999): "HTTP Authentication: Basic and Digest Access Authentication"
- RFC 2753 (January 2000): "A Framework for Policy-based Admission Control"
- RFC 2833 (May 2000): "RTP Payload for DTMF Digits, Telephony Tones and Telephony Signals"
- RFC 2915 (September 2000): The Naming Authority Pointer (NAPTR) DNS Resource Record
- RFC 2976 (October 2000): "The SIP INFO Method"
- RFC 3041 (January 2001): "Privacy Extensions for Stateless Address Autoconfiguration in IPv6"
- RFC 3261 (June 2002): "SIP: Session Initiation Protocol"
- RFC 3262 (June 2002): "Reliability of provisional responses in Session Initiation Protocol (SIP)"
- RFC 3263 (June 2002): "Session Initiation Protocol (SIP): Locating SIP Servers"
- RFC 3264 (June 2002): "An Offer/Answer Model with Session Description Protocol (SDP)"
- RFC 3265 (June 2002): "Session Initiation Protocol (SIP) - Specific Event Notification"
- RFC 3280 (April 2002): "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile"
- RFC 3310 (September 2002): "Hypertext Transfer Protocol (HTTP) Digest Authentication Using Authentication and Key Agreement (AKA)"
- RFC 3311 (September 2002): "The Session Initiation Protocol (SIP) UPDATE Method".
- RFC 3312 (October 2002): "Integration of Resource Management and Session Initiation Protocol (SIP)"
- RFC 3313 (January 2003): "Private Session Initiation Protocol (SIP) Extensions for Media Authorization"
- RFC 3315 (July 2003): "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)"
- RFC 3320 (January 2003): "Signaling Compression (SigComp)"

- RFC 3321 (January 2003): "Signaling Compression (SigComp) - Extended Operations"
- RFC 3323 (November 2002): "A Privacy Mechanism for the Session Initiation Protocol (SIP)"
- RFC 3325 (November 2002): "Private Extensions to the Session Initiation Protocol (SIP) for Network Asserted Identity within Trusted Networks"
- RFC 3326 (December 2002): "The Reason Header Field for the Session Initiation Protocol (SIP)"
- RFC 3327 (December 2002): "Session Initiation Protocol (SIP) Extension Header Field for Registering Non-Adjacent Contacts"
- RFC 3329 (January 2003): "Security Mechanism Agreement for the Session Initiation Protocol (SIP)"
- RFC 3388 (December 2002): "Grouping of Media Lines in the Session Description Protocol (SDP)"
- RFC 3428 (December 2002): "Session Initiation Protocol (SIP) Extension for Instant Messaging"
- RFC 3455 (January 2003): "Private Header (P-Header) Extensions to the Session Initiation Protocol (SIP) for the 3rd-Generation Partnership Project (3GPP)"
- RFC 3485 (February 2003): "The Session Initiation Protocol (SIP) and Session Description Protocol (SDP) Static Dictionary for Signaling Compression (SigComp)"
- RFC 3486 (February 2003): "Compressing the Session Initiation Protocol (SIP)"
- RFC 3515 (April 2003): "The Session Initiation Protocol (SIP) Refer method"
- RFC 3556 (July 2003): "Session Description Protocol (SDP) Bandwidth Modifiers for RTP Control Protocol (RTCP) Bandwidth"
- RFC 3581 (August 2003): "An Extension to the Session Initiation Protocol (SIP) for Symmetric Response Routing"
- RFC 3588 (September 2003): "Diameter Base Protocol"
- RFC 3608 (October 2003): "Session Initiation Protocol (SIP) Extension Header Field for Service Route Discovery During Registration"
- RFC 3665 (December 2003): "Session Initiation Protocol (SIP) Basic Call Flow Examples"
- RFC 3680 (March 2004): "A Session Initiation Protocol (SIP) Event Package for Registrations"
- RFC 3761 (April 2004): "The E.164 to Uniform Resource Identifiers (URI) Dynamic Delegation Discovery System (DDDS) Application (ENUM)"
- RFC 3824 (June 2004): "Using E.164 numbers with the Session Initiation Protocol (SIP)"
- RFC 3840 (August 2004): "Indicating User Agent Capabilities in the Session Initiation Protocol (SIP)"
- RFC 3841 (August 2004): "Caller Preferences for the Session Initiation Protocol (SIP)"
- RFC 3842 (August 2004): "A Message Summary and Message Waiting Indication Event Package for the Session Initiation Protocol (SIP)"
- RFC 3856 (August 2004): "A Presence Event Package for the Session Initiation Protocol (SIP)"
- RFC 3857 (August 2004): "A Watcher Information Event Template-Package for the Session Initiation Protocol (SIP)"
- RFC 3858 (August 2004): "An Extensible Markup Language (XML) Based Format for Watcher Information"
- RFC 3861 (August 2004): "Address Resolution for Instant Messaging and Presence"
- RFC 3891 (September 2004): "The Session Initiation Protocol (SIP) "Replaces" Header"
- RFC 3892 (September 2004): "The Session Initiation Protocol (SIP) Referred-By Mechanism"
- RFC 3903 (October 2004): "Session Initiation Protocol (SIP) Extension for Event State Publication"

- RFC 3911 (October 2004): "The Session Initiation Protocol (SIP) "Join" Header"
- RFC 3966 (December 2004): "The tel URI for Telephone Numbers"
- RFC 3986 (January 2005): "Uniform Resource Identifier (URI): Generic Syntax"
- RFC 4028 (April 2005): "Session Timers in the Session Initiation Protocol (SIP)"
- RFC 4032 (March 2005): "Update to the Session Initiation Protocol (SIP) Preconditions Framework"
- RFC 4077 (May 2005): "A Negative Acknowledgement Mechanism for Signaling Compression"
- RFC 4244 (November 2005): "An Extension to the Session Initiation Protocol (SIP) for Request History Information"
- RFC 4317 (December 2005): "Session Description Protocol (SDP) Offer/Answer Examples"
- RFC 4353 (February 2006): "A Framework for Conferencing with the Session Initiation Protocol (SIP)"
- RFC 4475 (May 2006): "Session Initiation Protocol (SIP) Torture Test Messages"
- RFC 4566 (July 2006): "SDP: Session Description Protocol"
- RFC 4975 (September 2007): "Message Session Relay Protocol (MSRP)"
- RFC 5031 (January 2008): "A Uniform Resource Name (URN) for Emergency and Other Well-Known Services"
- RFC 5049 (December 2007): "Applying Signaling Compression (SigComp) to the Session Initiation Protocol (SIP)"
- RFC 5112 (January 2008): "The Presence-Specific Static Dictionary for Signaling Compression (Sigcomp)"
- RFC 5491 (March 2009): "GEOPRIV Presence Information Data Format Location Object (PIDF-LO) Usage Clarification, Considerations, and Recommendations"
- RFC 5626 (October 2009): "Managing Client Initiated Connections in the Session Initiation Protocol (SIP)"

Other

- Packet-Cable spec (PKT-TR-SEC-V02-061013)

Chapter 2
Configuration

This chapter provides configuration information for the SCM.

Important: Information about all commands in this chapter can be found in the *Command Line Interface Reference*.

Because each wireless network is unique, the system is designed with a variety of parameters allowing it to perform in various wireless network environments. In this chapter, only the minimum set of parameters are provided to make the system operational. Optional configuration commands specific to the SCM product are located in the *Command Line Interface Reference*.

The following procedures are located in this chapter:

- Configuring the System to Perform as a Proxy-CSCF
- Configuring the System to Perform as a Serving-CSCF
- Configuring the System to Perform as an Emergency-CSCF
- Configuring the System to Perform as an A-BG

Configuring the System to Perform as a Proxy-CSCF

This section provides a high-level series of steps and the associated configuration file examples for configuring the system to perform as a Proxy-CSCF in a test environment. For a more robust configuration example, refer to the *Sample Configuration Files* appendix.

To configure the system to perform as a Proxy-CSCF:

Step 1 Set system configuration parameters such as activating PSCs by applying the example configurations found in the *System Administration Guide*.

Step 2 Set initial configuration parameters such as creating the VPN context and CSCF service by applying the example configurations found in the Initial Configuration section.

Step 3 Configure the system to perform as a Proxy-CSCF and set basic CSCF parameters such as service configuration, session limits, default AoR domain, CSCF peer servers, access control, translation and route lists, CSCF policy, and session template by applying the example configurations presented in the Proxy-CSCF Configuration section.

Step 4 Configure additional P-CSCF context parameters by applying the example configuration found in the P-CSCF Context Configuration section.

Step 5 Log system activity by applying the example configuration found in the CSCF Logging Configuration section.

Step 6 Save the configuration by following the steps found in the Save the Configuration section.

Initial Configuration

Step 1 Set local system management parameters by applying the example configuration in the Modifying the Local Context section.

Step 2 Create the context where the P-CSCF service will reside by applying the example configuration in the Creating a P-CSCF VPN Context section.

Step 3 Create the P-CSCF service within the newly created context by applying the example configuration in the Creating the CSCF Service section.

Modifying the Local Context

Use the following example to set the default subscriber and configure remote access capability in the local context:

```
configure
   context local
            interface <interface_name>
               ip address <ip_address> <ip_mask>
               exit
            server ftpd
```

```
      exit
   server telnetd
      exit
   subscriber default
      exit
   administrator <name> encrypted password <password> ftp
   ip route <ip_addr/ip_mask> <next_hop_addr> <lcl_cntxt_intrfc_name>
   exit
port ethernet <slot#/port#>
   no shutdown
   bind interface <local_context_interface_name> local
   exit
end
```

Creating a P-CSCF VPN Context

Use the following example to create a P-CSCF VPN context and interface, and bind the VPN interface to a configured Ethernet port.

```
configure
   context <p-cscf_context_name> -noconfirm
      interface <p-cscf_interface_name>
         ip address <address>
         exit
      ip route 0.0.0.0 0.0.0.0 <next_hop_address> <s-cscf_interface_name>
      exit
   port ethernet <slot_number/port_number>
      no shutdown
      bind interface <p-cscf_interface_name> <p-cscf_context_name>
      end
```

Creating the CSCF Service

Use the following configuration example to create the CSCF service:

```
configure
    context <p-cscf_context_name>
        cscf service <p-cscf_service_name> -noconfirm
            end
```

Proxy-CSCF Configuration

Step 1 Set the system's role as a Proxy-CSCF and configure service settings by applying the example configuration in the Setting the Systems Role as a Proxy-CSCF and Configuring Service Settings section.

Step 2 Configure communication with CSCF peer servers by applying the example configuration in the Identifying CSCF Peer Servers section.

Step 3 Specify ACLs and route lists by applying the example configuration in the Configuring Access Control and Route Lists section.

Step 4 Configure the CSCF policy and session template by applying the example configuration in the Setting the CSCF Policy and CSCF Session Template section.

Setting the System's Role as a Proxy-CSCF and Configuring Service Settings

Use the following configuration example to set the system to perform as a Proxy-CSCF and configure the CSCF service:

```
configure
    context <p-cscf_context_name>
        cscf service <p-cscf_service_name>
            bind address <ip_address> port <port_num>
            session-timer session-expires <value>
            session-timer min-se <value>
            keepalive method crlf max-retry <value> expire-timer <value>
            keepalive method stun max-retry <value> expire-timer <value>
            recurse-on-redirect-resp
            subscription package reg
            default-aor-domain <name>
```

```
        subscriber-policy-override
        proxy-cscf
           allow rfc3261-ua-interworking
           end
```

Identifying CSCF Peer Servers

Use the following example to identify peer servers to the P-CSCF:

```
configure
      context <p-cscf_context_name>
         cscf peer-servers <name> type <type> -noconfirm
            server <name> address <ip_address> port <number>
            hunting-method sequential-on-failure
            end
```

Configuring Access Control and Route Lists

Use the following example to configure CSCF access control lists (ACLs), CSCF translation lists, and CSCF route lists:

```
configure
      context <p-cscf_context_name>
         cscf acl default
            permit source aor $.
            exit
         cscf routes default
            end
```

Setting the CSCF Policy and CSCF Session Template

Use the following example to configure CSCF policy and session templates:

```
configure
      context <p-cscf_context_name>
         cscf policy default
            exit
```

```
cscf session-template name <name>
   inbound-cscf-acl default
   outbound-cscf-acl default
   route-list default
   translation-list default
   policy-profile default
   cscf-urn-service-list default
   end
```

P-CSCF Context Configuration

Use the following example to configure additional P-CSCF context parameters such as local subscribers for SIP UAs, AAA groups, and IP network settings:

```
configure
   context <p-cscf_context_name>
      subscriber default
         exit
      aaa group <name>
         exit
      domain <name>
      ip domain-lookup
      ip name-servers <ip_addr>
      dns-client <name>
         bind address <ip_addr>
         cache ttl positive <sec>
         cache ttl negative <sec>
         end
```

CSCF Logging Configuration

Use the following example to configure logging for the CSCF application:

```
logging filter active facility sessmgr level critical
logging filter active facility cscfmgr level critical
logging filter active facility cscf level critical
logging active
```

Save the Configuration

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

Configuring the System to Perform as a Serving-CSCF

This section provides a high-level series of steps and the associated configuration file examples for configuring the system to perform as a Serving-CSCF in a test environment. For a more robust configuration example, refer to the *Sample Configuration Files* appendix.

To configure the system to perform as a Serving-CSCF:

Step 1 Set system configuration parameters such as activating PSCs by applying the example configurations found in the *System Administration Guide*.

Step 2 Set initial configuration parameters such as creating the VPN context and CSCF service by applying the example configurations found in the Initial Configuration section.

Step 3 Configure S-CSCF context parameters by applying the example configuration found in the S-CSCF Context Configuration section.

Step 4 Configure the system to perform as a Serving-CSCF and set basic CSCF parameters such as service configuration, default AoR domain configuration, CSCF peer servers, access control, translation and route lists, and session template by applying the example configurations presented in the Serving-CSCF Configuration section.

Step 5 *Optional:* Configure the S-CSCF to also perform as an Interrogating-CSCF by applying the example configurations presented in the Optional Interrogating-CSCF Configuration section.

Step 6 Configure accounting service by applying the example configuration found in the CDR Accounting Service Configuration section.

Step 7 Log system activity by applying the example configuration found in the CSCF Logging Configuration section.

Step 8 Save the configuration by following the steps found in the Save the Configuration section.

Initial Configuration

Step 1 Set local system management parameters by applying the example configuration in the Modifying the Local Context section.

Step 2 Create the context where the S-CSCF service will reside by applying the example configuration in the Creating an S-CSCF VPN Context section.

Step 3 Create the S-CSCF service within the newly created context by applying the example configuration in the Creating the CSCF Service section.

Modifying the Local Context

Use the following example to set the default subscriber and configure remote access capability in the local context:

```
configure
   context local
            interface <interface_name>
```

```
      ip address <ip_address> <ip_mask>
      exit
   server ftpd
      exit
   server telnetd
      exit
   subscriber default
      exit
   administrator <name> encrypted password <password> ftp
   ip route <ip_addr/ip_mask> <next_hop_addr> <lcl_cntxt_intrfc_name>
   exit
port ethernet <slot#/port#>
   no shutdown
   bind interface <local_context_interface_name> local
   exit
end
```

Creating an S-CSCF VPN Context

Use the following example to create an S-CSCF VPN context and interface, and bind the VPN interface to a configured Ethernet port.

```
configure
   context <s-cscf_context_name> -noconfirm
      interface <s-cscf_interface_name>
         ip address <address>
         exit
      ip route 0.0.0.0 0.0.0.0 <next_hop_address> <s-cscf_interface_name>
      exit
   port ethernet <slot_number/port_number>
      no shutdown
      bind interface <s-cscf_interface_name> <s-cscf_context_name>
```

```
end
```

Creating the CSCF Service

Use the following configuration example to create the CSCF service:

```
configure
    context <s-cscf_context_name>
        cscf service <s-cscf_service_name> -noconfirm
            end
```

S-CSCF Context Configuration

Use the following example to configure additional S-CSCF context parameters such as local subscribers for SIP UAs, AAA groups, and IP network settings:

```
configure
    context <s-cscf_context_name>
        ims-sh-service <name>
            diameter dictionary standard
            diameter endpoint <hss_host_name>
            exit
        subscriber default
            exit
        aaa group <name>
            radius dictionary custom2
            diameter authentication dictionary aaa-custom4
            diameter authentication endpoint <hss_host_name>
            diameter authentication server <host_name> priority 1
            exit
        domain <name>
        ip domain-lookup
        ip name-servers <ip_addr>
        diameter endpoint <hss_host_name>
```

```
      origin realm <realm_name>
      origin host <host_name> address <host_ip_addr>
      connection retry-timeout <duration>
      peer <name> realm <realm_name> address <peer_pee_ip_addr>
   dns-client <name>
      bind address <ip_addr>
      cache ttl positive <sec>
      cache ttl negative <sec>
      end
```

Serving-CSCF Configuration

Step 1 Set the system's role as a Serving-CSCF and configure service settings by applying the example configuration in the Setting the Systems Role as a Serving-CSCF and Configuring Service Settings section.

Step 2 Configure communication with CSCF peer servers by applying the example configuration in the Identifying CSCF Peer Servers section.

Step 3 Specify ACL, translation, and route lists by applying the example configuration in the Configuring Access Control, Translation, and Route Lists section.

Step 4 Configure the CSCF policy and session template by applying the example configuration in the Setting the CSCF Session Template section.

Step 5 Configure communication with Domain Name Servers by applying the example configuration in the Configuring DNS Connectivity section.

Setting the System's Role as a Serving-CSCF and Configuring Service Settings

Use the following configuration example to set the system to perform as a Serving-CSCF and configure the service:

```
configure
      context <s-cscf_context_name>
   cscf service <s-cscf_service_name>
      bind address <ip_address> port <port_num>
      serving-cscf
         authentication allow-noauth invite
         authentication allow-noipauth
         registration lifetime min <sec> max <sec> default <sec>
```

```
            allow rfc3261-ua-interworking
            tas
            tas-service <ims-sh-service_name>
            exit
         session-timer session-expires <value>
         session-timer min-se <value>
         default-aor-domain <name>
         subscription package reg
         trusted-domain-entity <domain_name>
         policy-name <s-cscf_policy_name>
         end
```

Identifying CSCF Peer Servers

Use the following example to identify peer servers to the S-CSCF:

```
configure
      context <s-cscf_context_name>
         cscf peer-servers <name> type <type> -noconfirm
            server <name> address <ip_address> port <number>
            hunting-method sequential-on-failure
            end
```

Configuring Access Control, Translation, and Route Lists

Use the following example to configure CSCF access control lists (ACLs), CSCF translation lists, and CSCF route lists:

```
configure
      context <s-cscf_context_name>
         cscf acl default
            permit any
            permit source aor $.
            exit
         cscf translation default
```

```
            uri-readdress type <tag> base-criteria destination aor <aor>
            exit
         cscf routes default
            end
```

Setting the CSCF Session Template

Use the following example to configure CSCF policy and session templates:

```
configure
      context <s-cscf_context_name>
         cscf session-template name <name>
            inbound-cscf-acl default
            outbound-cscf-acl default
            route-list default
            translation-list default
            policy-profile default
            end
```

Configuring DNS Connectivity

Use the following example to configure communication with a DNS and bind an interface to the server:

```
configure
      context <context_name>
         ip domain-lookup
         ip name-server <ip_address>
         dns-client <name>
            bind address <ip_address>
```

Optional Interrogating-CSCF Configuration

Use the following example to configure the S-CSCF service to also perform Interrogating-CSCF task including communicating with the HSS via a Diameter Cx interface:

```
configuration
```

```
context <s-cscf_context_name>
   cscf service <s-cscf_service_name>
      serving-cscf
         interrogating-cscf-role
         allow rfc3261-ua-interworking
         exit
         diameter policy-control <hss_host_name>
            origin endpoint <hss_host_name>
            peer-select peer <auth_srv_host> peer-realm <origin_realm_name>
            dictionary Rx-standard
         exit
      exit
   aaa group <name>
      radius dictionary custom2
      diameter authentication dictionary aaa-custom4
      diameter authentication endpoint <hss_host_name>
      diameter authentication server <host_name> priority 1
      exit
   diameter endpoint <hss_host_name>
      origin realm <realm_name>
      origin host <host_name> address <ip_address>
      connection retry-timeout 1
      peer <auth_srv_host> realm <origin_realm_name> address <ip_addr>
```

CDR Accounting Service Configuration

Use the following example to configure CDR accounting access for the CSCF application:

```
configure
   context <context_name>
      radius group default
```

```
radius attribute nas-ip-address address <primary_address>
radius dictionary <db>
radius server <ip_address key <value> port <number>
radius accounting server <ip_address> key <value> port <number>
end
```

CSCF Logging Configuration

Use the following example to configure logging for the CSCF application:

```
logging filter active facility sessmgr level critical
logging filter active facility cscfmgr level critical
logging filter active facility cscf level critical
logging active
```

Save the Configuration

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

Configuring the System to Perform as an Emergency-CSCF

This section provides a high-level series of steps and the associated configuration file examples for configuring the system to perform as an Emergency-CSCF in a test environment. For a more robust configuration example, refer to the *Sample Configuration Files* appendix.

To configure the system to perform as an Emergency-CSCF:

Step 1 Set system configuration parameters such as activating PSCs by applying the example configurations found in the *System Administration Guide*.

Step 2 Configure the system to perform as a Proxy-CSCF and set basic CSCF parameters by applying the example configurations presented in the Configuring the System to Perform as a Proxy-CSCF section.

Step 3 Set the system's role as an Emergency-CSCF and configure service settings by applying the example configuration in the section.

Step 4 *Optional:* Configure the system to perform as a Serving-CSCF and set basic CSCF parameters by applying the example configurations presented in the Configuring the System to Perform as a Serving-CSCF section.

Step 5 Log system activity by applying the example configuration found in the CSCF Logging Configuration section.

Step 6 Save the configuration by following the steps found in the Save the Configuration section.

Setting the System's Role as an Emergency-CSCF and Configuring Service Settings

Use the following configuration example to set the system to perform as an Emergency-CSCF and configure the CSCF service:

```
configure
        context <emergency_context_name>
           cscf service <emergency_service_name>
              emergency-cscf
                 privacy
                 exit
              default-aor-domain <name>
              keepalive method crlf max-retry <value> expire-timer <value>
              keepalive method stun max-retry <value> expire-timer <value>
              policy-name <emergency_policy_name>
              bind address <ip_address> port <port_num>
```

```
end
```

CSCF Logging Configuration

Use the following example to configure logging for the CSCF application:

```
logging filter active facility sessmgr level critical
logging filter active facility cscfmgr level critical
logging filter active facility cscf level critical
logging active
```

Save the Configuration

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

Configuring the System to Perform as an A-BG

This section provides a high-level series of steps and the associated configuration file examples for configuring the system to perform as an A-BG in a test environment. For a more robust configuration example, refer to the *Sample Configuration Files* appendix.

To configure the system to perform as an A-BG:

Step 1 Set system configuration parameters such as activating PSCs by applying the example configurations found in the *System Administration Guide*.

Step 2 Configure the system to perform as a Proxy-CSCF and set basic CSCF parameters by applying the example configurations presented in the Configuring the System to Perform as a Proxy-CSCF section.

> ***Important:*** The following commands must be added to the Proxy-CSCF Service: nat-pool name <*core_pool_name*> access-service name <*access_proxy_name*>

Step 3 Configure access context parameters by applying the example configuration found in the Access Context Configuration section.

Step 4 Set the system's role as an access-proxy and configure service settings by applying the example configuration in the Setting the Systems Role as an Access-Proxy and Configuring Service Settings section.

Step 5 *Optional:* Configure the system to perform as a Serving-CSCF and set basic CSCF parameters by applying the example configurations presented in the Configuring the System to Perform as a Serving-CSCF section.

Step 6 Log system activity by applying the example configuration found in the CSCF Logging Configuration section.

Step 7 Save the configuration by following the steps found in the Save the Configuration section.

Access Context Configuration

Use the following example to configure additional access context parameters, such as local subscribers for SIP UAs, AAA groups, and IP network settings:

```
configure
        context <access_context_name>
           ip pool <nat_pool> range <start_address> <end_address> nat 0
           interface <interface_name>
              ip address <ip_address> <ip_mask>
              exit
           cscf policy name <access_policy_name>
              service-policy-rules
```

```
      video-sessions
      exit
   exit
subscriber default
   exit
aaa group <name>
   exit
gtpp group default
   end
```

Setting the System's Role as an Access-Proxy and Configuring Service Settings

Use the following configuration example to set the system to perform as an access-proxy and configure the CSCF service:

```
configure
      context <access-proxy_context_name>
         cscf service <access-proxy_service_name>
            proxy-cscf
               allow rfc3261-ua-interworking
               exit
            core-service name <proxy_cscf>
            nat-pool name <nat_pool>
            default-aor-domain <name>
            keepalive method crlf max-retry <value> expire-timer <value>
            keepalive method stun max-retry <value> expire-timer <value>
            policy-name <access_policy_name>
            bind address <ip_address> port <port_num>
               end
```

CSCF Logging Configuration

Use the following example to configure logging for the CSCF application:

```
logging filter active facility sessmgr level critical
logging filter active facility cscfmgr level critical
logging filter active facility cscf level critical
logging active
```

Save the Configuration

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

Appendix A
Access Control Lists

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

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

Understanding ACLs

This section discusses concepts about how ACLs are created, ordered, and viewed on the system. The two main aspects to consider when creating an ACL are:

- Rule(s)
- Rule Order

Rule(s)

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

Important: Configured ACLs consisting of no rules imply a "permit any" rule. The **deny** action and **any** criteria are discussed later in this section.

Each rule specifies the action to take when a packet matches the specifies criteria. This section discusses the rule actions and criteria supported by the system.

Actions

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

- **Deny:** The packet is rejected.
- **Permit:** The packet is accepted and processed.
- **Log:** Enables logging for packets meeting the criteria specified in the ACL. The logs can be viewed by executing the **logging filter active facility acl-log** command in the system's Execute mode.

Important: Packet logging is not supported for context-level (policy) ACLs. Subscriber-level ACL logging can be performed using the Session Manager task (sessmgr) logging facility.

Permit and Deny use the following syntax:

```
{ permit | deny } [ log ] { <criteria> }
```

Keyword/Variable	Description
log	Enables logging for packets meeting the criteria specified in the ACL. ***Important:*** Logging is not supported for Policy ACLs (those applied to contexts).
criteria	The criteria to compare packets against as described in the section that follows.

Criteria

Each ACL consists of one or more rules specifying the criteria that packets will be compared against. The following criteria are supported:

- **Any:** Filters all packets
- **Source Address:** Filter packets based on one or more source IP addresses
- **Source AoR:** Filters packets based on the source address of record
- **Destination AoR:** Filters packets based on the destination address of record

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

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

Any

The rule applies to all packets.

The following syntax is used when configuring rule criteria that applies to all packets:

any

Source Address

The rule applies to specific packets originating from a specific source IP address or a group of source IP addresses.

The following syntax is used when configuring rule criteria that apply to one or more source IP addresses:

source address *<ip_address> <wildcard>*

Keyword/Variable	Description
ip_address	The IP address(es) from which the packet originated. This option is used to filter all packets from a specific IP address or a group of IP addresses. When specifying a group of addresses, the initial address is configured using this option. The range can then be configured using the *wildcard* parameter.
wildcard	This option is used in conjunction with the *ip_address* option to specify a group of addresses for which packets are to be filtered. The mask must be entered as a complement: Zero-bits in this parameter mean that the corresponding bits configured for the *ip_address* parameter must be identical. One-bits in this parameter mean that the corresponding bits configured for the *ip_address* parameter must be ignored. ***Important:*** The mask must contain a contiguous set of one-bits from the least significant bit (LSB). Therefore, allowed masks are 0, 1, 3, 7, 15, 31, 63, 127, and 255. For example, acceptable wildcards are 0.0.0.3, 0.0.0.255, and 0.0.15.255. A wildcard of 0.0.7.15 is **not** acceptable since the one-bits are not contiguous.

Source AoR

The rule applies to specific packets originating from a specific source address of record.

The following syntax is used when configuring rule criteria that apply to source AoRs:

source aor *<aor> <wildcard>*

Keyword/Variable	Description
aor	The address of record from which the packet originated.This option is used to filter all packets from a specific address of record or a group of AoRs. When specifying a group of addresses, the initial address is configured using this parameter. The range can then be configured using the *wildcard* parameter.
wildcard	This option is used in conjunction with the *aor* option to specify a group of addresses for which packets are to be filtered. The mask must be entered as a complement:Zero-bits in this parameter mean that the corresponding bits configured for the *aor* parameter must be identical. One-bits in this parameter mean that the corresponding bits configured for the *aor* parameter must be ignored. ***Important:*** The mask must contain a contiguous set of one-bits from the least significant bit (LSB). Therefore, allowed masks are 0, 1, 3, 7, 15, 31, 63, 127, and 255. For example, acceptable wildcards are 0.0.0.3, 0.0.0.255, and 0.0.15.255. A wildcard of 0.0.7.15 is **not** acceptable since the one-bits are not contiguous.

Destination AoR

The rule applies to specific packets sent to a specific destination address of record.

The following syntax is used when configuring rule criteria that apply to destination AoRs:

destination aor *<aor> <wildcard>*

Keyword/Variable	Description
aor	The address of record to which the packet is being sent.This option is used to filter all packets being sent to a specific address of record or a group of AoRs. When specifying a group of addresses, the initial address is configured using this parameter. The range can then be configured using the *wildcard* parameter.
wildcard	This option is used in conjunction with the *aor* option to specify a group of addresses for which packets are to be filtered. The mask must be entered as a complement:Zero-bits in this parameter mean that the corresponding bits configured for the *aor* parameter must be identical. One-bits in this parameter mean that the corresponding bits configured for the *aor* parameter must be ignored. ***Important:*** The mask must contain a contiguous set of one-bits from the least significant bit (LSB). Therefore, allowed masks are 0, 1, 3, 7, 15, 31, 63, 127, and 255. For example, acceptable wildcards are 0.0.0.3, 0.0.0.255, and 0.0.15.255. A wildcard of 0.0.7.15 is **not** acceptable since the one-bits are not contiguous.

Rule Order

A single ACL can consist of multiple rules. Each packet is compared against each of the ACL rules, in the order in which they were entered, until a match is found. Once a match is identified, all subsequent rules are ignored.

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

- Before
- After

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

```
[ before | after ] { <existing_rule> }
{ <new_rule> }
```

An example of an ACL is shown in the following section.

Viewing ACLs

ACLs can be viewed through the **show configuration** command executed from the context where the ACL resides. The following example was taken from the output of the **show configuration context** <*name*> command:

```
[test1]st40# show configuration context test1
config
 context test1
 subscriber default
 exit
 radius group default
 #exit
cscf acl name acl1
after permit criteria source address 1.2.3.4
after deny criteria destination aor *.bad.com
after permit criteria source aor *@test.com
after deny criteria source address 0.0.0.255
after deny criteria source aor user@test.com
 #exit
 #exit
```

end

Appendix B
IP Security

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

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

This chapter contains the following sections:

Overview

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

SIP deployments are exposed to a wide range of network security threats and attacks. Classic SIP threat models are:

- Registration hijacking
- Impersonating a server
- Tampering with message bodies
- Tearing down session on behalf of some other user
- Denial of service attacks
- SPIT attack (Spam over Internet Telephony)

To prevent such threats, the following network securities are required for SIP:

- Authentication and privacy of the call participants
- Prevent replay attacks and message spoofing
- Preserve integrity and confidentiality of the messages
- Application-level protection from SPIT attacks

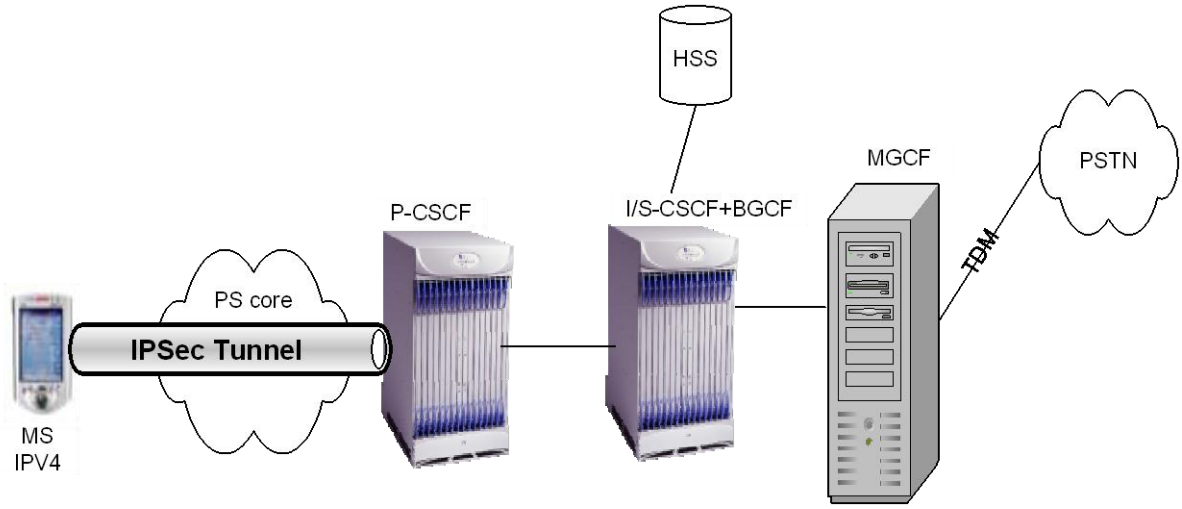
This chapter describes SIP/IMS security models for network- and access-side security, requirements, and CSCF and IPSEC subsystem interfaces.

IMS Security Network Scenarios

The following figures show supported network- and access-side security scenarios.

Access Security

Figure 15. IPSec Tunnel Between UE and P-CSCF



Access and Network Domain Security

Figure 16. Access and Network Domain Security with IPSec

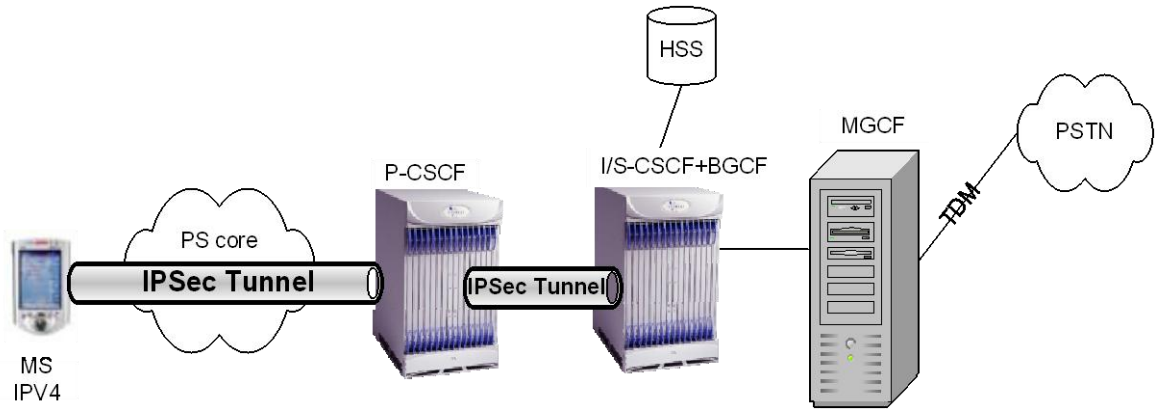


Figure 17. Access and Network Domain Security with IPSec and TLS

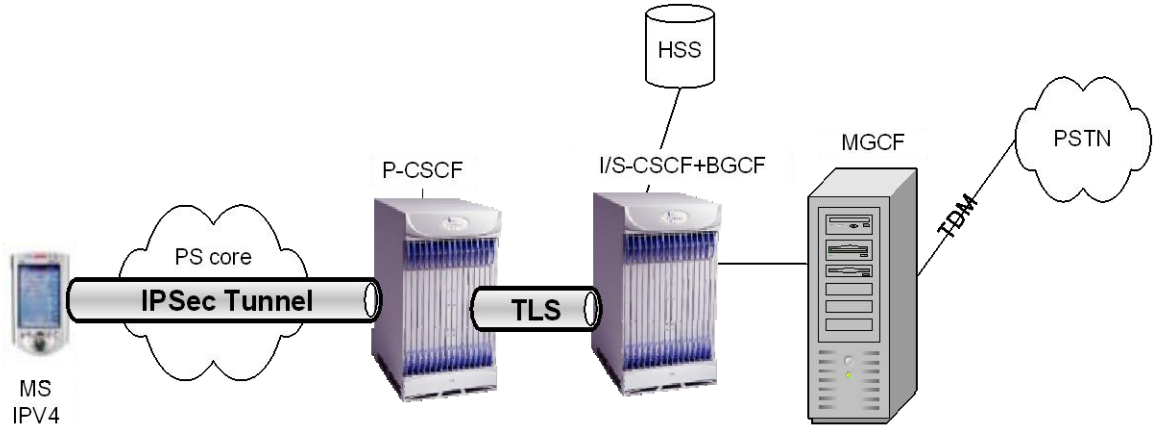


Figure 18. Secure Connection Towards HSS

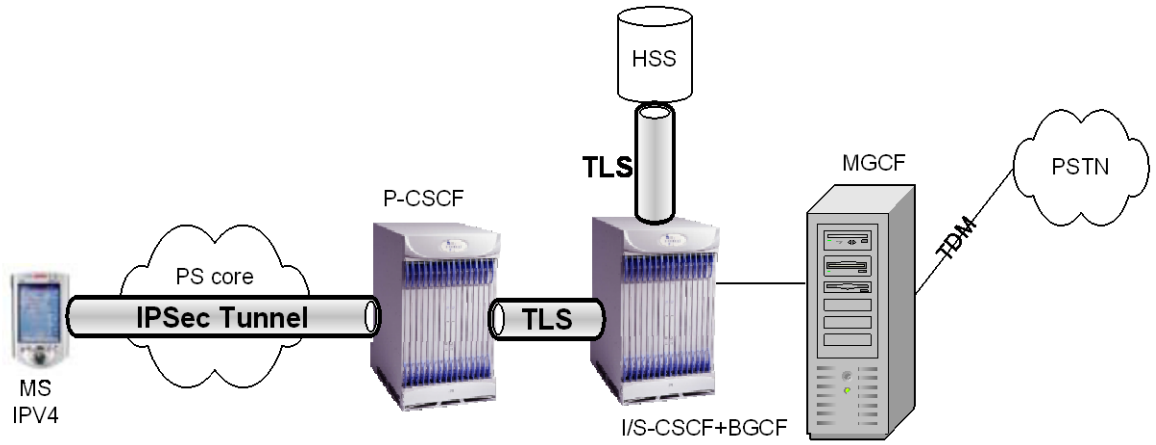


Figure 19. Secure Connection Towards SIP AS Outside Operator's Domain

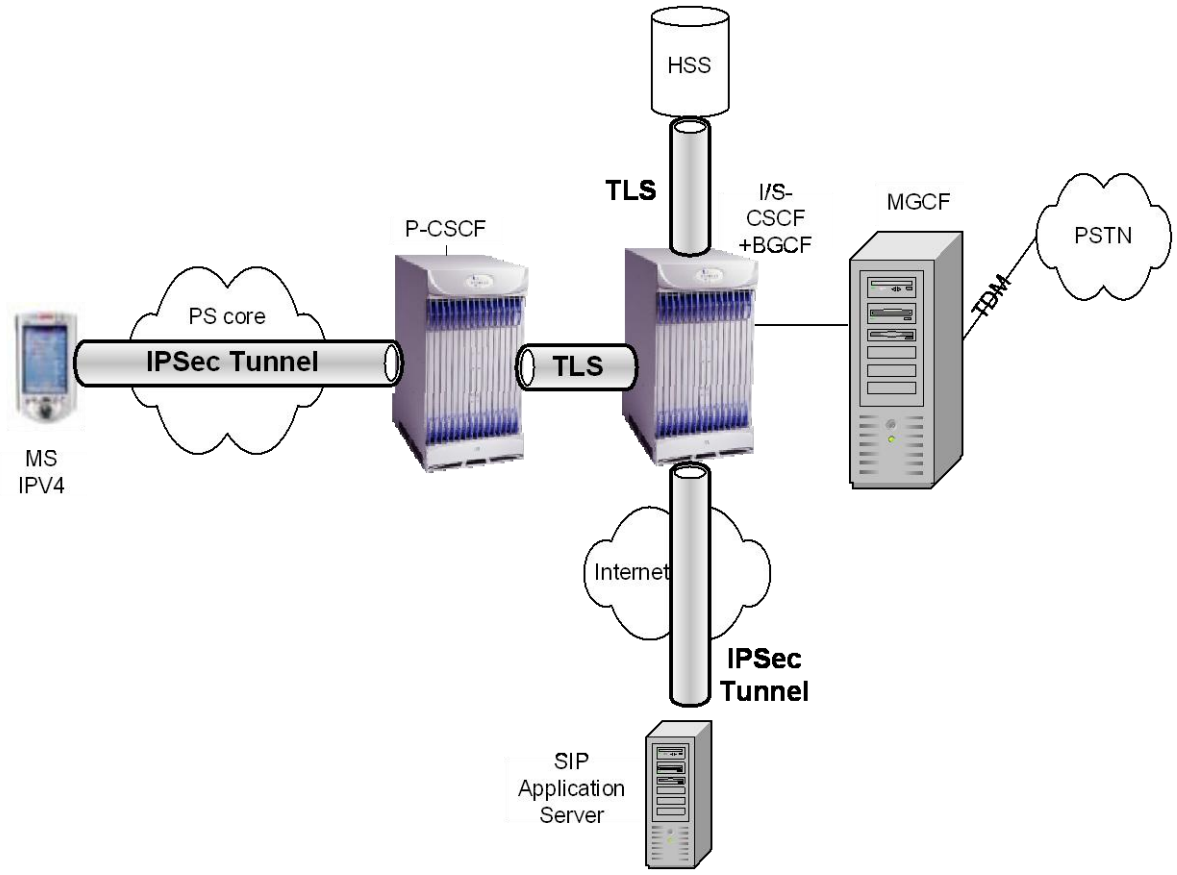


Figure 20. Secure Connection Using TLS with SIP Client

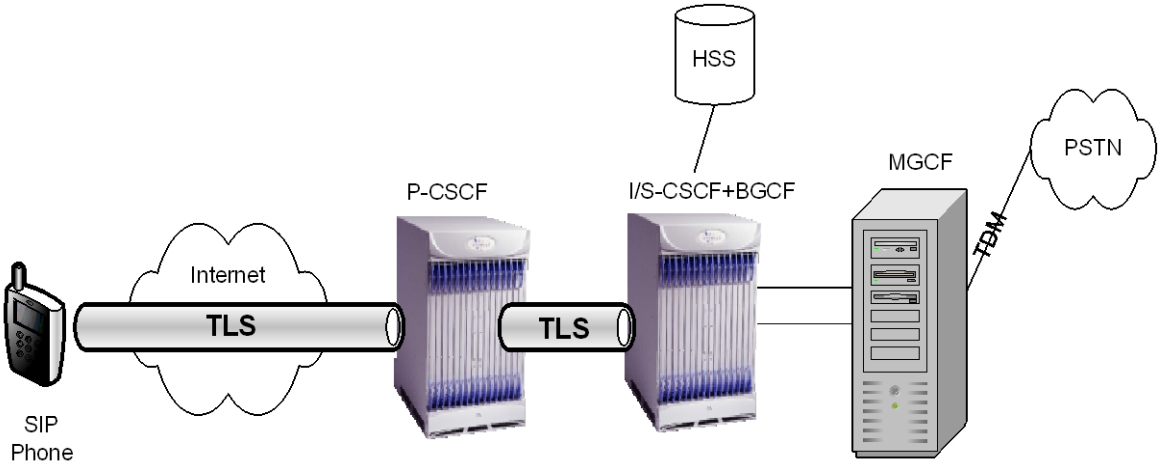
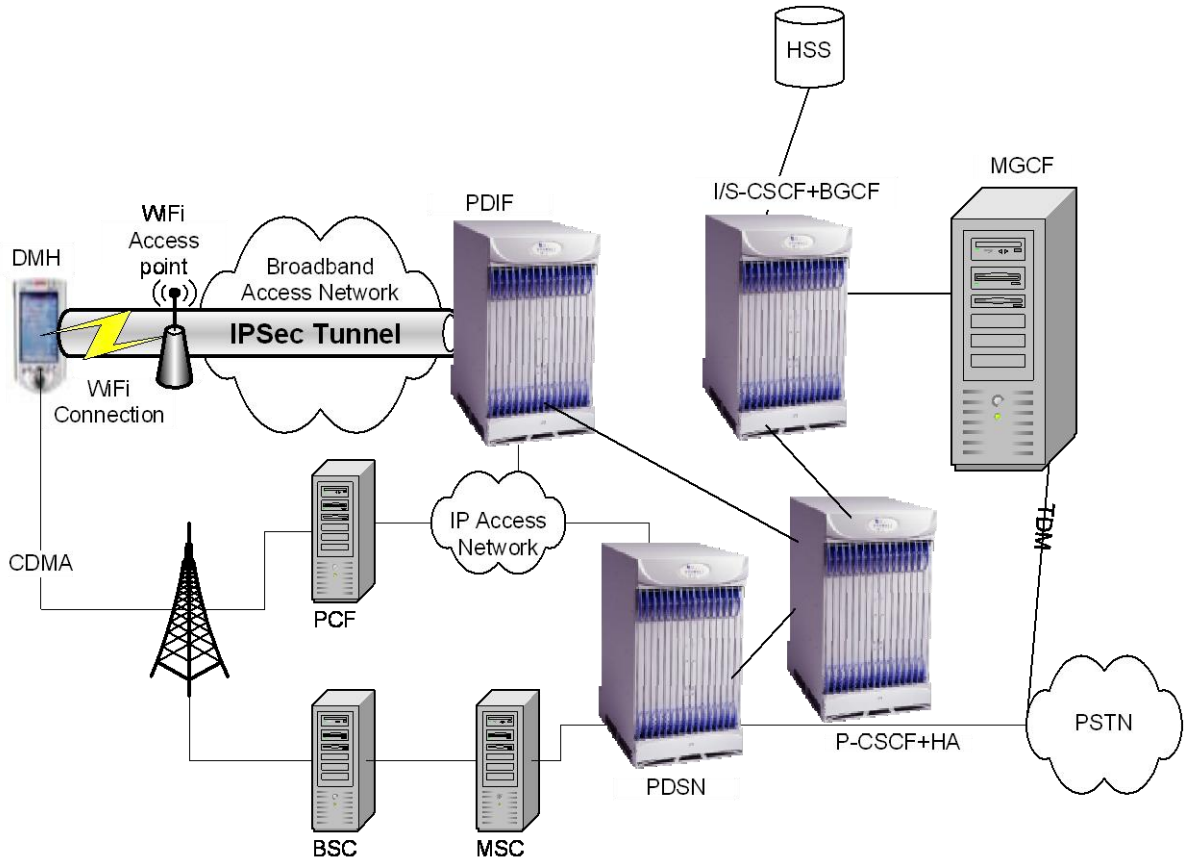


Figure 21. Dual Mode Handset with IPSec Termination on PDIF



P-CSCF Security Support

P-CSCF supports both ipsec-3gpp and TLS for SIP security as follows:

- P-CSCF and the terminal agree on the set of parameters to establish two IPSec security associations between them as specified in RFC 3329 and 3GPP 33.203.
- Access security includes authentication of users and the network, and protection of traffic between IMS terminals and the network.
- P-CSCF allows activation and de-activation of access security.
- Both P-CSCF and S-CSCF support coexistence of IMS and non-IMS/Early IMS endpoints. They support various authentication methods on the same network element.
- Both UDP and TCP are supported on the same SA pair.
- P-CSCF supports recovery of the security association.
- P-CSCF supports ESP **transport mode** for clients that are not behind NAT and supports ESP **tunnel mode** for clients behind NAT.
- UDP encapsulation of ESP in tunnel mode is supported.
- P-CSCF supports authentication algorithm HMAC-MD5-96 and HMAC-SHA-1-96, and encryption algorithm DES EDE3 CBC, as specified in RFC 2451, and AES CBC, as specified in RFC 3602 with 128-bit key.

As per 3GPP 33.203, P-CSCF supports negotiation of the following SA parameters between UE and P-CSCF using SIP:

- Encryption algorithm
- Integrity algorithm
- SPI (Security Parameter Index)

The following security parameters shall not be negotiated:

- Lifetime
- SA duration
- Key length of integrity key (The length of the integrity key IKESP depends on the integrity algorithm. It is 128 bits for HMAC MD5 96 and 160 bits for HMAC SHA 1 96.)
- Key length of encryption key
- Mode

Two pairs of (unilateral) security associations (SAs) are established between the UE and the P-CSCF. The subscriber may have several IMPUs associated with one IMPI. These may belong to the same or different service profiles. Only two pairs of SAs shall be active between the UE and the P-CSCF. These two pairs of SAs shall be updated when a new successful authentication of the subscriber has occurred.

P-CSCF monitors the expiry time of registrations without an authentication and if necessary increases the lifetime of SAs created by the last successful authentication such that it will expire shortly after the registration timer in the message. P-CSCF deletes any SA whose lifetime is exceeded. The P-CSCF deletes all SAs it holds that are associated with a particular IMPI once all the associated IMPUs are de-registered.

The SIP application at the P CSCF checks upon receipt of a protected REGISTER message that the source IP address in the packet headers coincide with the UE's IP address inserted in the Via header of the protected REGISTER message. If the Via header does not explicitly contain the UE's IP address, but rather a symbolic name, then the P-CSCF first resolves the symbolic name using DNS to obtain an IP address.

For each unidirectional SA which has been established and has not expired, the SIP application at the P-CSCF stores the following data: (UE_IP_address, UE_protected_port, P-CSCF_protected_port, SPI, IMPI, IMPU1, ... , IMPUn, lifetime, mode) in an "SA_table". The pair (UE_protected_port, P-CSCF_protected_port) equals either (port_uc, port_ps) or (port_us, port_pc).

P-CSCF does not accept registration attempts from UEs with the same address and protected server port in order to ensure unambiguous addressing of SIP messages sent towards the UE, using the protected server port.

P-CSCF checks that, for any one IMPI, no more than six SAs per direction are stored at any one time. If these checks are unsuccessful, the registration is aborted and an error message is sent to the UE.

The SIP application at the P CSCF checks upon receipt of an initial REGISTER message or a re-REGISTER message that the pair (UE_IP_address, UE_protected_client_port), where the UE_IP_address is the source IP address in the packet header and the protected client port is sent as part of the security mode set-up procedure, has not yet been associated with entries in the "SA_table".

The P-CSCF associates two ports, called port_ps and port_pc, with each pair of security associations established in an authenticated registration. The ports port_ps and port_pc are different from the standard SIP ports 5060 and 5061. No unprotected messages shall be sent from or received on the ports port_ps and port_pc. From a security point of view, unprotected messages may be received on any port which is different from the ports port_ps and port_pc. The number of the ports port_ps and port_pc are communicated to the UE during the security mode set-up procedure/SIP registration procedure.

Security Association Setup for Subscriber Session

Security association is established during SIP REGISTER processing.

First SIP register comes on unprotected port and will be received by P-CSCF and does the following processing for the first REGISTER from an IMS client:

1. Remove the Security-Client header and extract the following parameters from Security-Client header in the REGISTER message and store them in the local cache:
    - UE IMPI
    - UE IMPU
    - SPI value pair
    - Port value pair
    - UE integrity and encryption algorithm list
    - UE IP address from the source IP address of the IP packet header
2. Remove Proxy-Require: sec-agree option tag
3. Remove Require: sec-agree header option tag
4. It should insert "integrity-protected" parameter with value "no" as this REGISTER is not received on a secure connection.

P-CSCF has a list of integrity and encryption algorithm configured with priority in the crypto map template. Crypto map template is obtained during crypto map binding with CSCF service. P-CSCF selects the first algorithm combination on its own list which is also supported by the UE. After algorithm selection it will send the REGISTER to I/S-CSCF.

Upon receiving the SIP register, S-CSCF checks if this is first time registration or not. If this is first time registration, it will send MAR(IMPI,m) to HSS to get the AV where IMPI is UE's private ID and m is the number of authentication vectors. Upon receipt of a request from the S-CSCF, the HSS sends an ordered array of n authentication vectors to the S-CSCF using MAA. The authentication vectors are ordered based on sequence number. Each authentication vector consists of the following components: a random number RAND, an expected response XRES, a cipher key CK, an

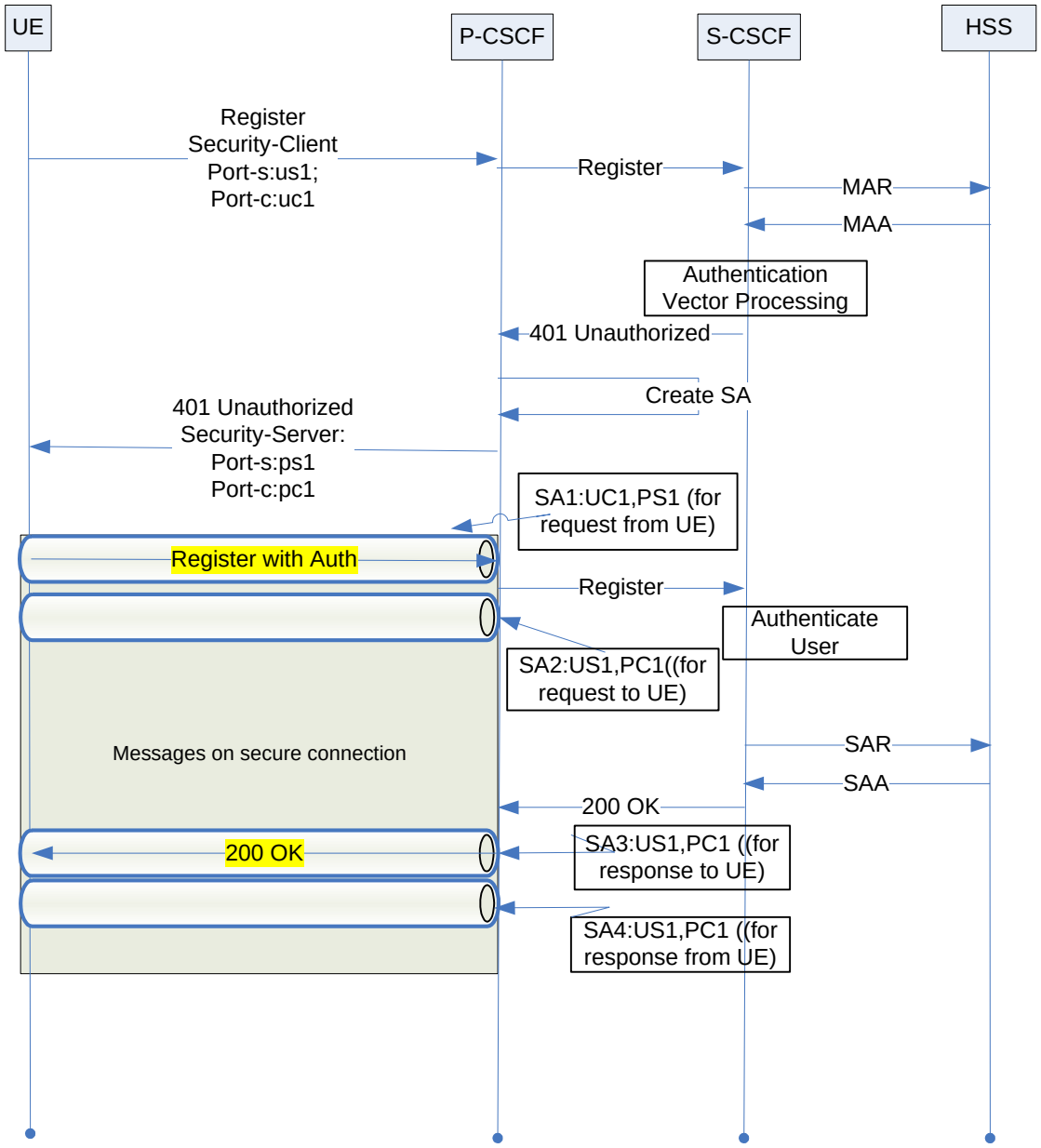
integrity key IK, and an authentication token AUTN. Each authentication vector is good for one authentication and key agreement between the S-CSCF and the IMS user. S-CSCF will store this information in local cache and sends 401 response to UE. P-CSCF will store and remove IK and CK from 401 in local cache and forwards the message to UE. P-CSCF will do the following processing on 401 received from S-CSCF:

1. Add list of algorithms, port value pair, and SPI value pair in the Security-server header in the 401 response. P-CSCF will keep a flag to indicate that this subscriber is using secure connection.
2. Remove ik and ck from WWW-authenticate header and store them locally.
3. Forward modified 401 to UE and complete security association establishment.

The second REGISTER from UE will come on secured connection. P-CSCF will get the second register on the port_ps. It will do the following processing on second REGISTER:

1. It will check for the presence of Security-Verify header and remove it. P-CSCF will make sure it contains the same parameters sent in Security-Server header. If the two parameters do not match, it will reject the REGISTER with 403 and delete the security association. If there is no Security-Verify header, then P-CSCF will send 403 response and delete the SA.
2. Remove Security-Client header and compare the contents of Security-Client header with the Security-Client header received in the first REGISTER. If Security-Client header is not present, or the content does not match, then the REGISTER will be rejected with 403 and SA should be deleted.
3. Remove Proxy-Require: sec-agree option tag
4. Remove Require: sec-agree header option tag
5. It will insert “integrity-protected” parameter with value “yes” as this REGISTER is received on a secure connection.

The following call flow shows a successful SA setup with REGISTER.



Re-registration Handling

Every reregistration that includes a user authentication attempt creates new security associations. Old set of security association is deleted and new set is created. According to the spec port_ps stays the same for new SA but port_pc and port_uc shall change. If UE sends an unprotected register, P-CSCF shall assume that old SAs have been deleted on UE and should proceed with registration procedure and delete the old SAs associated with that UE.

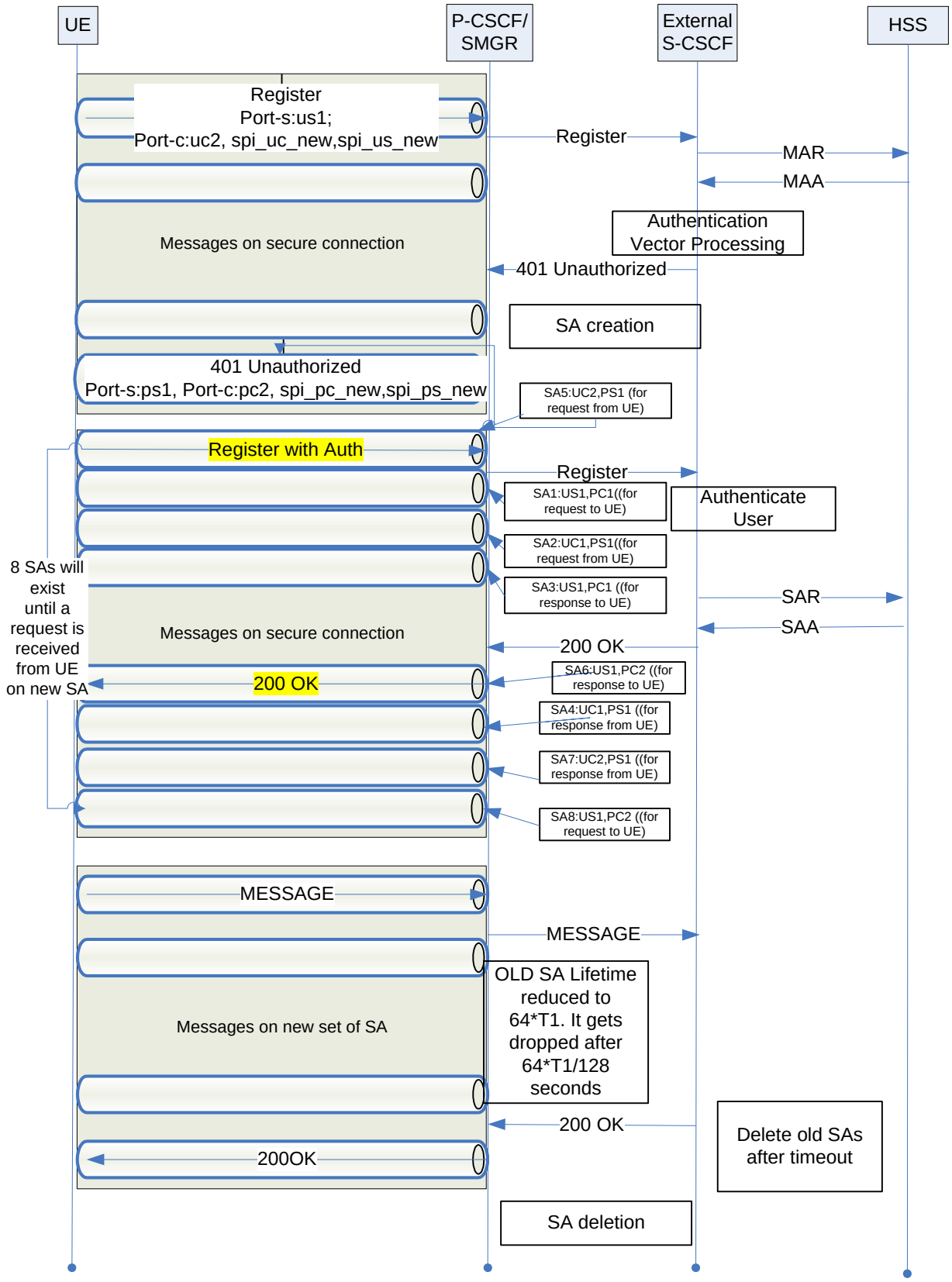
The initial REGISTER message for re-registration will come on the old set of security associations. A new set of SA will be created during 401 processing, with new client port and SPI values sent by UE. P-CSCF will keep the old set of SAs until it receives a request from the UE on new SA. P-CSCF is not sure if UE received 200OK for second REGISTER or not. If the UE has not received the 200OK response for second REGISTER, it will not start using the

new set of SAs. As long as P-CSCF does not receive a request from the UE on the new set of SA, it will do the following:

- Send incoming requests to the UE on the old SA with protected port_pc1.
- Keep both sets of SAs active until one or both of them either expire or a new request is received from the UE.

P-CSCF will mark the locally stored entry of SAs as the temporary entry until a message is received from the UE.When P-CSCF receives a new request from the UE on the new SA, P-CSCF will mark the new set of SAs as permanent SA and reduce the lifetime of the old set of SAs to be 64*T1 (default to 128 seconds for IMS UE). The old set of SAs are not dropped immediately, as the UE might have sent or received a request over it. After SA lifetime timer expiry, P-CSCF will delete the old SA.

The following call flow shows an SA setup during re-registration.


UE
P-CSCF/ SMGR
External S-CSCF
HSS
Register
Port-s:us1;
Port-c:uc2, spi_uc_new,spi_us_new
Register
MAR
MAA
Authentication Vector Processing
Messages on secure connection
401 Unauthorized
SA creation
401 Unauthorized
Port-s:ps1, Port-c:pc2, spi_pc_new,spi_ps_new
SA5:UC2,PS1 (for request from UE)
Register with Auth
Register
SA1:US1,PC1((for request to UE)
Authenticate User
SA2:UC1,PS1((for request from UE)
SA3:US1,PC1 ((for response to UE)
8 SAs will exist until a request is received from UE on new SA
SAR
SAA
Messages on secure connection
200 OK
SA6:US1,PC2 ((for response to UE)
200 OK
SA4:UC1,PS1 ((for response from UE)
SA7:UC2,PS1 ((for response from UE)
SA8:US1,PC2 ((for request to UE)
MESSAGE
MESSAGE
OLD SA Lifetime reduced to 64*T1. It gets dropped after 64*T1/128 seconds
Messages on new set of SA
200 OK
Delete old SAs after timeout
200OK
SA deletion

SA Lifetime Management

SA lifetime management will be done by P-CSCF.

During authentication, SA lifetime is set to four minutes. After authentication process is complete, the SA lifetime will be set to expiration time of the registration plus 30 seconds if there is no existing set of security associations (i.e., the "for the first time" association from the UE).

If there is already an existing set of SA (during re-registration), the lifetime of the newly established SA will be set to the lifetime of already existing SA as long as it is longer than the expiration time in reregistration plus 30 seconds.

If the lifetime received in re-registration is larger then the lifetime of existing SA, then lifetime of the old SA will be updated to new expiration time plus 30 seconds.

P-CSCF will set the SA lifetime to be 64*T1 in case of network-initiated de-registration. Expired SAs will be deleted using the timer maintained by P-CSCF.

IMS Registration with USIM

When the IMS terminal is equipped with a UICC that does not contain an ISIM application but contains an USIM application, there are a few issues with registration.

USIM does not contain private user identity and the home domain information. Both of these parameters are stored in ISIM not in USIM. USIM contains IMSI and IMSI is never used to route calls in the network; it is only used for authentication/authorization purposes.

When an IMS terminal has UICC with USIM, it uses IMSI to build a temporary private user ID and home network domain. These temporary IDs are only used during registration and de-registration. Once the user is registered, it gets a set of public user IDs from HSS that it can use for regular call processing and session setup. P-CSCF supports USIM registration.

The security association will have no change other then using temporary private ID. The temporary private ID will be of the form *imsi@mnc.mcc.imsi.3gppnetwork.org*. Home network domain name will be of the form *mnc.mcc.imsi.3gppnetwork.org*.

IPSec Configuration

Important: Commands used in the configuration samples in this section provide base functionality to the extent that the most common or likely commands and/or keyword options are presented. In many cases, other optional commands and/or keyword options are available. Refer to the *Cisco ASR 5000 Series Command Line Interface Reference* for complete information regarding all commands.

To configure the system for IPSec support:

1. Configure an IPSec transform set by applying the example configuration in the Creating and Configuring an IPSec Transform Set section. Transform sets are used to define IPSec security associations (SAs). IPSec SAs specify the IPSec protocols to use to protect packets.
2. Configure a crypto template by applying the example configuration in the Creating and Configuring a Crypto Template section.
3. Bind an IP address to the crypto template by applying the example configuration in the Binding an IP Address to the Crypto Template section.
4. Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Creating and Configuring an IPSec Transform Set

The following example configures an IPSec transform set, which is used to define the security association that determines the protocols used to protect the data on the interface:

```
configure
   context <p-cscf_context_name> -noconfirm
      ipsec transform-set <ipsec_transform-set_name>
         encryption <3des-cbc | aes-cbc-128 | aes-cbc-256 | des-cbc | null>
         group <1 | 2 | 5 | 14 | none>
         hmac <md5-96 | sha1-96 | null>
         mode <transport | tunnel>
         end
```

Notes:

- The encryption algorithm **aes-cbc-128**, or Advanced Encryption Standard Cipher Block Chaining, is the default algorithm for IPSec transform sets configured on the system.
- The **group** command configures the appropriate key exchange cryptographic strength and activates Perfect Forward Secrecy by applying a Diffie-Hellman group. The keyword **none** specifies that no crypto strength is

included and that Perfect Forward Secrecy is disabled. This is the default setting for IPSec transform sets configured on the system.

- The **hmac** command configures the Encapsulating Security Payload (ESP) integrity algorithm. The **sha1-96** keyword uses a 160-bit secret key to produce a 160-bit authenticator value. This is the default setting for IPSec transform sets configured on the system.
- The **mode** command onfigures the security of IP datagrams based on header placement. The default **tunnel** mode specifies that the entire packet is to be encapsulated by the IPSec header, including the IP header. The **transport** mode specifies that the IPSec header is applied only over the IP payload, not over the IP header in front of it.

Creating and Configuring a Crypto Template

The following example configures a crypto template:

```
configure
   context <p-cscf_context_name> -noconfirm
      crypto template <crypto_template_name> ipsec-3gpp-cscf
         ipsec transform-set list <list_name1> . . . <list_name4>
            end
```

Notes:

- The IPSec CSCF crypto template should be configured in the same context in which the P-CSCF is configured.
- The **ipsec transform-set list** command specifies up to four IPSec transform sets.

Binding an IP Address to the Crypto Template

The following example defines the domain name of the CSCF service and configures the binding of a logical IP interface to the crypto template:

```
configure
   context <p-cscf_context_name> -noconfirm
      cscf service <p-cscf_service_name> -noconfirm
         default-aor-domain <alias>
         bind address <ip_address> ipsec-crypto-template <template>
         exit
         end
```

Appendix C
TLS Support

This chapter describes the system's support for Transport Layer Security (TLS) and explains how it is configured. The product administration guides provide examples and procedures for configuration of basic services on the system. It is recommended that you select the configuration example that best meets your service model and configure the required elements for that model, as described in the *Cisco ASR 5000 Series Session Control Manager Administration Guide*, before using the procedures in this chapter.

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

This chapter includes following sections:

Overview

When enabled through a feature license key, TLS provides confidentiality and integrity protection for SIP signaling messages between the UE and P-CSCF/A-BG. TLS is a layered protocol that runs upon reliable transport protocols like TCP and SCTP.

Important: All future references in this chapter to P-CSCF imply support by the A-BG as well.

TLS Session Renegotiation

TLS handshake protocol creates a TLS session identified by a session id at both client and server. The TLS session contains all the security parameters selected for the connection. The lifetime of the TLS sessions can be configured in P-CSCF service. The default value is one hour. When the TLS session is about to expire, P-CSCF will initiate a TLS session renegotiation procedure over the existing TLS connection by sending the HelloRequest message. This will negotiate new security parameter for the connection. If the UE fails to start the renegotiation by sending clientHello, P-CSCF will terminate the TLS connection. UE may also initiate a TLS session renegotiation by sending clientHello message over the existing TLS connection.

TLS Session Setup

The setup of a TLS session between a UE and P-CSCF is coupled with the initial registration procedure. In IMS, the authentication of the users is performed during the registration procedure. Subsequent signaling messages between a UE and P-CSCF will be integrity protected based on the TLS session that was established during the authentication process. P-CSCF also supports TLS session setup, as per RFC 3261.

TLS Session Tear Down

When the user authentication fails, both the UE and P-CSCF will send a close_notify message on the TLS connection and delete the associated TLS session. Receiving an alert message on the TLS connection with severity “fatal” will cause the TLS connection and the session to be deleted. When the UE cannot verify the P-CSCF server certificate during the handshaking process, it sends an alert message and closes the TLS connection. When all the public user ids associated with the private user id of the UE is deregistered, P-CSCF will close the TLS connection by sending a close_notify message.

P-CSCF Server Certificate

The P-CSCF server certificate used in the TLS handshake for server authentication is the X.509v3 digital certificate. The Common Name value of the Subject field in the certificate contains the P-CSCF fully qualified domain name (FQDN). As part of the certificate verification process, UE verifies it against the known host names of the P-CSCF. Existing CLIs to input certificates are used to configure P-CSCF TLS certificate. The certificate is configured in the Global Configuration Mode and managed by vpnctrl. Either PEM encoded X.509v3 certificate can be configured or a URL to the certificate can be configured.

Important: Only RSA-based certificates are currently supported.

Use of TLS as Transport Between UE and P-CSCF

This section specifically outlines the use of TLS between UE and PCSCF.

P-CSCF supports two methods for TLS connection setup:

- TLS as a transport between UE and P-CSCF, as per RFC 3261
- Use of TLS by Security Mechanism agreement between UE and P-CSCF, as per RC 3329 and TS 33.203

TLS Setup Using 3GPP Approach

The setup of a TLS session between UE and P-CSCF is coupled with the initial registration procedure, as per 3GPP 33.203. In IMS, the authentication of the users is performed during registration procedure. Subsequent signaling messages between UE and P-CSCF will be integrity protected based on the TLS session that was established during the authentication process.

The sip-sec-agree negotiation is used by UE and P-CSCF to negotiate the choice of security mechanism. The UE sends the list of the security mechanisms it supports and the parameters required for the mechanisms in the Security-Client header in initial register request. Upon receiving the register request, P-CSCF selects one security-mechanism from the UE list (based on the configuration) and sends it in the Security-Server header in 401 response towards UE. If TLS was selected by P-CSCF, UE starts the TLS handshake procedure with P-CSCF. TLS handshake protocol authenticates the peers and establishes the security parameters (keys, secrets) required for the connection.

Once the TLS handshake completes and the TLS session is setup, UE sends the challenge response register over the established TLS connection. This contains the Security-Verify header that mirrors the Security-Server header received by UE in 401 response. P-CSCF, on receiving this register over the TLS connection, verifies the security-verify header and adds a TLS integrity-protection indicator with value “tls-pending” before forwarding it to S-CSCF. Upon receiving 200 OK from S-CSCF, P-CSCF forwards it over the established TLS connection and associates the UE's IP address and port of the TLS connection with the TLS session ID, the private user identity, and all the successfully registered public user identities related to the private user identity. This completes the successful TLS session setup between UE and P-CSCF. After this point, both UE and PCSCF exchange messages over the established TLS connection. See TLS Register Call Flow for a detailed call flow example.

During the TLS session setup, only P-CSCF is authenticated by the UE by presenting a valid server certificate. The authentication of the UE is done by the home network using SIP digest authentication mechanism.

UE and P-CSCF follow the procedures defined in RFC5626 to keep the TLS connection active. This is required because P-CSCF cannot initiate TLS connection towards UE and any terminating request for the UE requires an existing TLS connection.

By default, P-CSCF will listen on port 5061 for TLS connections. A configuration option will be provided so that the operator can configure any port for TLS connection. It is possible to configure both TLS and IPSec access security mechanisms in P-CSCF. When UE supports TLS and IPSec (indicated by the Security-Client header), P-CSCF will use the access profile configuration, if configured to select the access security mechanism; otherwise, IPSec is given preference over TLS.

P-CSCF will add the integrity-protection indicator for the REGISTER request received over the TLS connection. During initial registration, for the challenge response register request received over the established TLS connection, P-CSCF will add the integrity-protected value “tls-pending” while forwarding the register to S-CSCF.

For re-register/refresh register request received over the existing TLS connection, P-CSCF will add the integrity-protected value “tls-yes”.

TLS Setup Using RFC3261 Approach

When the DNS SRV records for P-CSCF return a sips URI, TLS is used to send SIP signaling messages toward P-CSCF. All the SIP signaling messages from UE will be sent via TLS to ensure confidentiality. UE will set up a TLS connection with P-CSCF before sending any SIP signaling messages. This scenario also assumes only server-based certificates are used. The UE is assumed to have the public key of the CA, who issued the P-CSCF certificate. See TLS 3GPP Approach Call Flow for a detailed call flow example.

The SYN packet is processed by P-CSCF. P-CSCF responds with SYN-ACK and installs a 4-tuple TCP flow (ue ip, ue port, pcscf ip, and tls port) for receiving further packets on the connection. Once the TCP connection is established, UE starts the TLS handshaking process and a TLS session is established between UE and sessmgr. All SIP signaling message exchanges between UE and P-CSCF are sent over the established TLS connection. The UE has to use a suitable keep-alive mechanism to keep this TLS connection active. This is required for forwarding any incoming request towards the UE on the existing TLS connection. When there is an active TLS connection between UE and P-CSCF, P-CSCF will use the same connection to send any mobile terminating requests towards the UE, bypassing the normal SIP routing rules.

Session Recovery

On session task crash or PSC failure TCP/TLS connection is not recovered. This will result in the UE detecting the flow failure. The UE will initiate the registration procedure again to establish the new TCP and TLS session. This is similar to the initial registration procedure.

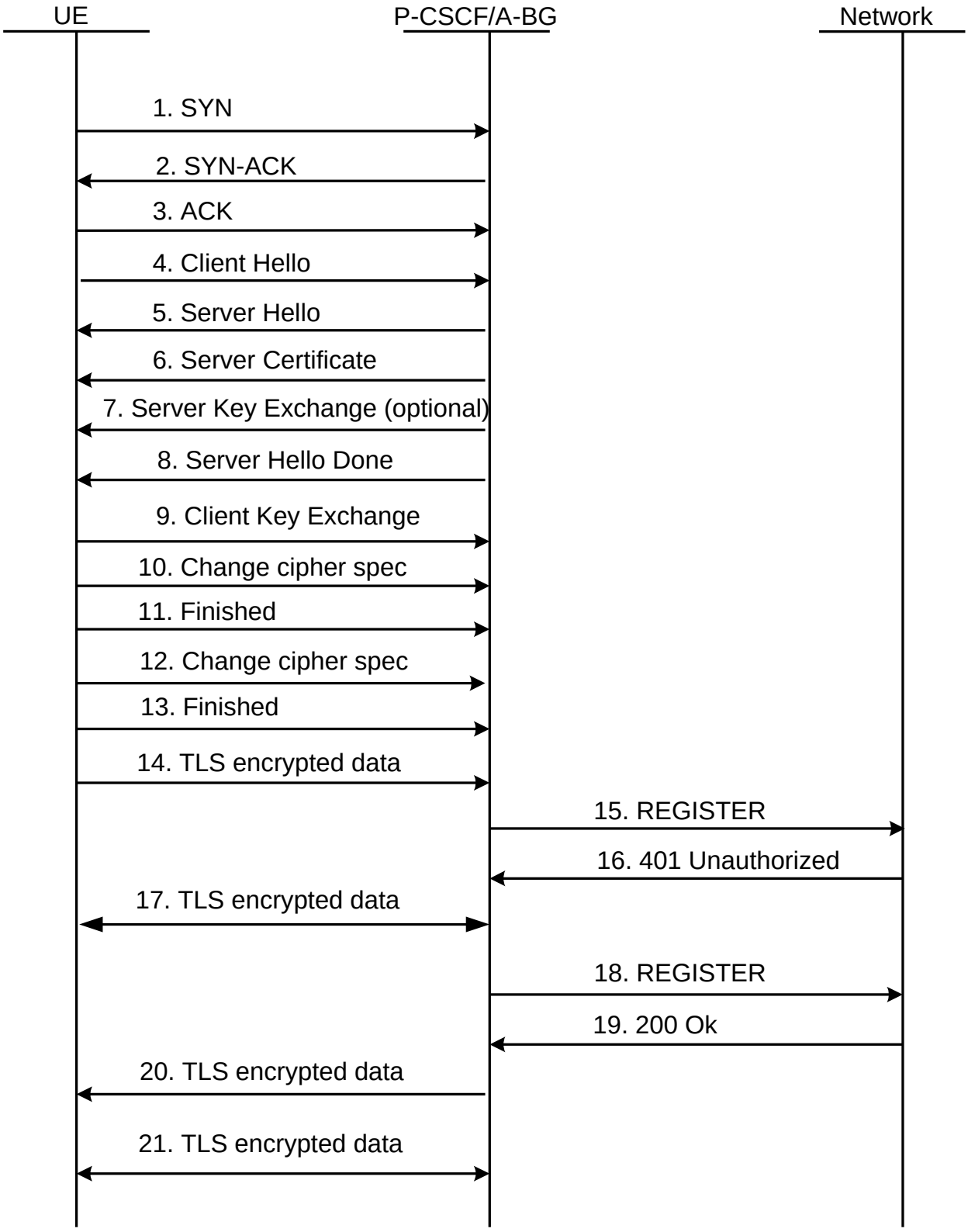
PSC Migration

TLS connection will be recovered on PSC migration.

Engineering Rules

- 500k current TLS sessions

TLS Register Call Flow

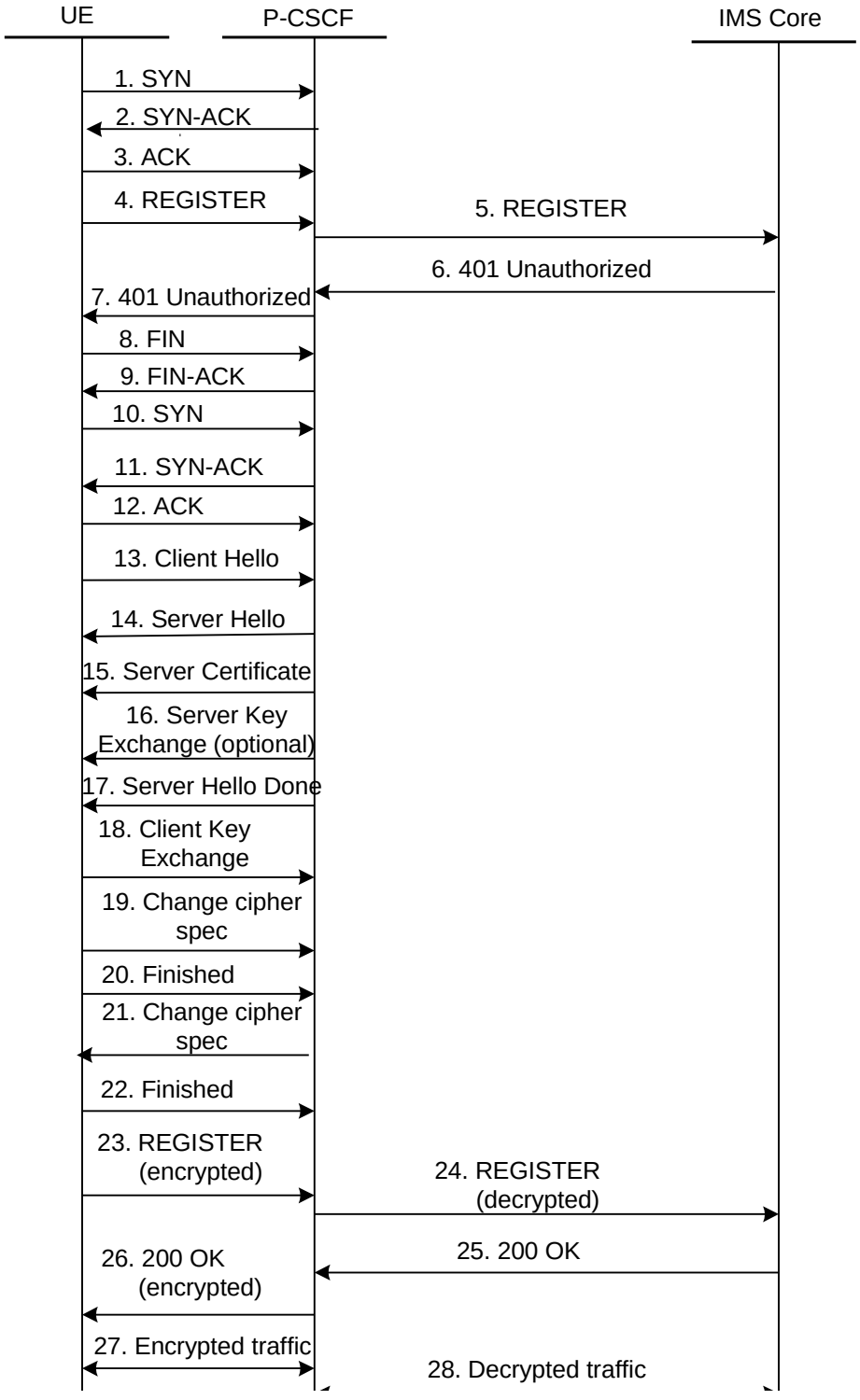


1. UE sends SYN packet to the P-CSCF TLS port.
2. P-CSCF upon receiving SYN packet sends SYN-ACK and installs a 4 tuple NPU flow for receiving future packets.
3. UE sends ACK. The TCP connection establishment is successful and it invokes SSL module. This involves creating a new TLS connection.
4. The UE sends the ClientHello message to initiate TLS handshake procedure. The ClientHello message contains the list of cipher-suites the UE supports, a random number and a session id field. The ClientHello message (TCP data), is processed by P-CSCF.
5. The P-CSCF upon receiving ClientHello responds with ServerHello message. The ServerHello message contains a single cipher suite selected from the client list.
6. TLS supports three authentication modes: authentication of both parties, server authentication with an unauthenticated client, and total anonymity. For TLS between UE and P-CSCF, server authentication is used. Client is authenticated by the home network using SIP digest. The P-CSCF sends the ServerCertificate message following the ServerHello message for server authentication. X.509 digital certificates are used for authentication.
7. ServerKeyExchange message is sent following the ServerCertificate message. This message is optional and is sent based on the key exchange algorithm selected in the above steps.
8. ServerHelloDone is sent to indicate the end of the server hello and associated messages.
9. UE sends the ClientKeyExchange message. This message contains the premaster secret generated by the UE. This premaster secret is used by both UE and P-CSCF for generating the keys required in the encryption and authentication process. The ClientKeyExchange message is processed by the P-CSCF and it computes the master secret, client write key, server write key, client write MAC secret, server write MAC secret for the TLS session.
10. ChangeCipherspec message is sent by UE to indicate that the subsequent messages will be protected under the negotiated cipher spec and keys.
11. UE sends the finished message to verify that the key exchange and authentication processes were successful. This is the first message protected with the just negotiated algorithms, keys and secrets. The P-CSCF verifies that the finished message is valid according to the negotiated session state.
12. In response to the finished message, the P-CSCF sends its own changecipherspec message.
13. Finally, the P-CSCF sends the finished message under the new cipher spec. This completes the TLS handshake process.
14. After this point, all the SIP signaling messages between UE and P-CSCF are exchanged over the established TLS connection.
15. Data exchange over TLS connection.
16. Data exchange over TLS connection.
17. Data exchange over TLS connection.
18. Data exchange over TLS connection.
19. Data exchange over TLS connection.
20. Data exchange over TLS connection.
21. Data exchange over TLS connection.


Important: Normally, UE is required to keep this TLS connection active by sending keep alives. This is required for the P-CSCF to forward any terminating request to the UE over TLS.The default idle timeout in

P-CSCF for TLS connection is one hour. P-CSCF stores the TLS connection parameter (source ip and source port) of the UE. It uses this information for sending any request towards the UE over the existing TLS connection.

TLS 3GPP Approach Call Flow



1. UE Initiates SYN.

2. P-CSCF responds with SYN ACK.
3. UE sends ACK. TCP connection establishment is complete.
4. Register from UE to P-CSCF.

   The REGISTER request is sent from UE to P-CSCF. It contains the security-client header indicating the support for TLS and headers related to RFC 5626. The request is sent over TCP.

   REGISTER sip:registrar.home1.net SIP/2.0

   Require: see-agree

   Proxy-Require: see-agree

   Security-Client: tls; q=0.1

   Supported: Outbound, Path

   Contact: <sip:xxx>;reg-id=1, +sip.instance="<urn:uuid:00000000-0000-1000-8000-000A95A0E128>"
5. Register is sent from P-CSCF to I/S-CSCF.

   P-CSCF upon receiving the REGISTER request examines the Security-Client header. If P-CSCF supports TLS, it removes the Security-Client header and the seg-agree option from Require and Proxy-require headers and forwards the REGISTER request to I/S-CSCF.
6. 401 unauthorized response is sent from S-CSCF to P-CSCF.
7. P-CSCF inserts the Security-Server header containing the value “tls” and forwards the response to UE over TCP.

   SIP/2.0 401 Unauthorized

   Security-Server: tls; q=0.1,
8. After receiving the 401 response from P-CSCF, UE now begins the TLS session setup procedures by performing the TLS handshake. UE initiates the TLS connection towards P-CSCF TLS default port 5061 or the configured port.
9. After receiving the 401 response from P-CSCF, UE now begins the TLS session setup procedures by performing the TLS handshake. UE initiates the TLS connection towards P-CSCF TLS default port 5061 or the configured port.
10. After receiving the 401 response from P-CSCF, UE now begins the TLS session setup procedures by performing the TLS handshake. UE initiates the TLS connection towards P-CSCF TLS default port 5061 or the configured port.
11. After receiving the 401 response from P-CSCF, UE now begins the TLS session setup procedures by performing the TLS handshake. UE initiates the TLS connection towards P-CSCF TLS default port 5061 or the configured port.
12. After receiving the 401 response from P-CSCF, UE now begins the TLS session setup procedures by performing the TLS handshake. UE initiates the TLS connection towards P-CSCF TLS default port 5061 or the configured port.
13. Once the TCP connection is established successfully, UE sends the ClientHello message to initiate TLS handshake procedure. The ClientHello message contains the list of cipher-suites the UE supports, a random number and a session id field. The ClientHello message (TCP data), is send to the SSL module for processing from the user tcp stack read call back function by calling sn_ssl_process_tcp_data() API.
14. P-CSCF upon receiving ClientHello responds with ServerHello message. The ServerHello message contains a single cipher suite selected from the client list. As per RFC 2246, cipher suites with NULL integrity protection or anonymous key exchange method are not allowed. Both UE and P-CSCF should support TLS_RSA_WITH_AES_128_CBC_SHA and TLS_RSA_WITH_3DES_EDE_CBC_SHA cipher suites. Other cipher suites mentioned in 33.203 are optional.

15. TLS supports three authentication modes: authentication of both parties, server authentication with an unauthenticated client, and total anonymity. For TLS between UE and P-CSCF, server authentication is used. Client is authenticated by the home network using SIP digest. Since the key exchange method selected in the above step cannot be anonymous, the P-CSCF sends the ServerCertificate message following the ServerHello message for server authentication. X.509 digital certificates are used for authentication.

16. ServerKeyExchange message is sent following the ServerCertificate message. This message is optional and is sent based on the key exchange algorithm selected in the above steps.

17. ServerHelloDone is sent to indicate the end of the server hello and associated messages.

18. ClientKeyExchange is the first message sent by the UE in the handshake process. This message contains the premaster secret generated by the UE. This premaster secret is used by both UE and P-CSCF for generating the keys required in the encryption and authentication process. The ClientKeyExchange message is processed by the P-CSCF and it computes the master secret, client write key, server write key, client write MAC secret, server write MAC secret for the TLS session.

19. ChangeCipherspec message is sent by UE to P-CSCF to indicate that the subsequent messages will be protected under the negotiated cipher spec and keys.

20. UE sends the finished message to verify that the key exchange and authentication process were successful. This is the first message protected with the just negotiated algorithms, keys and secrets. The P-CSCF verifies that the finished message is valid according to the negotiated session state.

21. In response to the finished message, the P-CSCF sends its own changecipherspec message.

22. Finally the P-CSCF sends the finished message under the new cipher spec. This completes the TLS handshake process.

23. Once the TLS handshake is complete, both UE and P-CSCF store the TLS session id. All further messages between UE and P-CSCF are sent over the established TLS connection. UE sends the challenge response register message over the TLS connection.

24. Once the TLS handshake is complete, both UE and P-CSCF store the TLS session id. All further messages between UE and P-CSCF are sent over the established TLS connection. UE sends the challenge response register message over the TLS connection.

25. Once the TLS handshake is complete, both UE and P-CSCF store the TLS session id. All further messages between UE and P-CSCF are sent over the established TLS connection. UE sends the challenge response register message over the TLS connection.

26. Once the TLS handshake is complete, both UE and P-CSCF store the TLS session id. All further messages between UE and P-CSCF are sent over the established TLS connection. UE sends the challenge response register message over the TLS connection.

27. Once the TLS handshake is complete, both UE and P-CSCF store the TLS session id. All further messages between UE and P-CSCF are sent over the established TLS connection. UE sends the challenge response register message over the TLS connection.

28. Once the TLS handshake is complete, both UE and P-CSCF store the TLS session id. All further messages between UE and P-CSCF are sent over the established TLS connection. UE sends the challenge response register message over the TLS connection.

REGISTER sip:registrar.home1.net SIP/2.0

Require: see-agree

Proxy-Require: see-agree

Security-Client: tls; q=0.1

Security-Verify: tls; q=0.1

Supported: outbound

Contact: <sip:xxx>;reg-id=1;+sip.instance="<urn:uuid:00000000-0000-1000-8000-000A95A0E128>"

The SSL module decrypts the REGISTER request (application data) and provides it to the user module via the registered callback. The register request is sent to the dc-sip stack for processing.

P-CSCF receives the REGISTER over the TLS connection and forwards it to SCSCF. It removes all the security related headers and adds an integrity-protected parameter in the Authorization header with the value “tls-pending”

REGISTER sip:registrar.home1.net SIP/2.0

Authorization: ..; integrity-protected="tls-pending"

Contact: <sip:xxx>;reg-id=1;+sip.instance="<urn:uuid:00000000-0000-1000-8000-000A95A0E128>"

All subsequent message exchanges (invite, subscribe etc) between UE and P-CSCF will happen over the established TLS connection. P-CSCF will not accept any sip message outside of the TLS connection except for the REGISTER request and INVITE request related to emergency calls.

TLS Configuration

Important: Commands used in the configuration samples in this section provide base functionality to the extent that the most common or likely commands and/or keyword options are presented. In many cases, other optional commands and/or keyword options are available. Refer to the *Cisco ASR 5000 Series Command Line Interface Reference* for complete information regarding all commands.

To configure the system for TLS:

1. Create a P-CSCF TLS certificate by applying the example configuration in the section Creating the P-CSCF TLS Certificate .
2. Create an X.509 CA root certificate to enable a the P-CSCF to perform certificate-based peer (client) authentication by applying the example configuration in the section Creating the Intermediate CAs in the Certificate Chain .
3. Create an SSL cipher suite for the SSL template by applying the example configuration in the section Creating the SSL Cipher Suite .
4. Create the SSL template and specify the associated SSL cipher suite by applying the example configuration in the section Creating the SSL Template .
5. Create the CSCF service for SSL access by applying the example configuration in the section Binding an SSL Template to a P-CSCF Service .
6. Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command **save configuration**. For additional information on how to verify and save configuration files, refer to the *System Administration Guide* and the *Command Line Interface Reference*.

Sample Configuration

```
configure
 orbem
 no siop-port
 no iiop-port
 default iop-address
 #exit
 card 17
 redundancy card-mode
 #exit
 card 1
 mode active
```

```
#exit
context local
subscriber default
exit
aaa group default
#exit
gtpp group default
gtpp egcdr lotdv-max-containers 0
gtpp egcdr losdv-max-containers 0
#exit
#exit
task facility sessmgr max 1
certificate name pcscftls pem url /flash/newcert.pem private-key pem url
/flash/newkey1.pem
ca-certificate name cacert pem url /home/psujithr/openssl_certs/demoCA/cacert.pem
context accessvpn
interface IP_17/1
ip address 191.168.10.10 255.255.255.0
#exit
ip route 191.168.20.0 255.255.255.0 next-hop 191.168.10.15 IP_17/1
ip route 191.168.30.0 255.255.255.0 next-hop 191.168.10.15 IP_17/1
ip route 191.168.40.0 255.255.255.0 next-hop 191.168.10.15 IP_17/1
ip route 60.0.0.0 255.0.0.0 next-hop 191.168.10.15 IP_17/1
ip route 10.5.2.0 255.255.255.0 next-hop 191.168.10.15 IP_17/1
ip pool pool_access range 191.168.40.153 191.168.40.250 napt-users-per-ip-address
20 port-chunk-size 6432
cipher-suite abc
encryption aes-128
#encryption null
#exit
```

```
ssl template pcscf ssl-subscriber
cipher-suites list abc
certificate pcscftls
ca-certificate list cacert
exit
subscriber default
exit
aaa group default
#exit
gtpp group default
#exit
end
config
context accessvpn
cscf service accesspcscf
proxy-cscf
allow rfc3261-ua-interworking
no store-session-path
network-id cisco.com
sip-param insert integrity-protected
sigcomp
#exit
#exit
media-bridging
nat-pool name pool_access
core-service name corepcscf
bind address 191.168.10.10 tls-crypto-template pcscf transport tcp
default-aor-domain 191.168.20.10
subscription package reg
```

```
#exit
keepalive method crlf max-retry 3 expire-timer 29
keepalive method stun max-retry 3 expire-timer 29
recurse-on-redirect-resp
strict-outbound
#exit
ip igmp profile default
#exit
#exit
```

Creating the P-CSCF TLS Certificate

Use this example to create and select an X.509 Trusted Author certificate:

```
configure
   certificate name <name> pem url <url> private-key pem url <url>
   end
```

Creating the Intermediate CAs in the Certificate Chain

Use this example to create selects an X.509 CA root certificate to enable the P-CSCF to perform certificate-based peer (client) authentication:

```
configure
   ca-certificate name <name> pem url <url>
   end
```

Creating the SSL Cipher Suite

Use this example to create the SSL cipher suite for the SSL template:

```
configure
   context <context_name> -noconfirm
      cipher-suite <cipher_suite_name>
         encryption rc4
```

```
hmac sha1

key-exchange rsa

end
```

A cipher suite contains the cryptographic algorithms supported by the client, and defines a key exchange and a cipher spec, which specifies the encryption and hash algorithms used during authentication. SSL cipher suites allow operators to select levels of security and to enable communication between devices with different security requirements.

This example shows default values.

Creating the SSL Template

Use this example to create the SSL template used to define the SSL cryptographic policy for the CSCF service for SSL access:

```
configure

   context <context_name> -noconfirm

      ssl template <ssl_template_name> ssl-subscriber

         cipher-suites list <name>

         certificate <name>

         ca-certificate list <name>

         version list tlsv1

         end
```

A P-CSCF service for SSL access will not function without a configured SSL template. The **ssl-subscriber** keyword in the **ssl template** command specifies that SSL protocol is used. The **certificate** command binds the specified X.509 trusted certificate to the SSL template.

Only one SSL template can be configured per P-CSCF service.

Binding an SSL Template to a P-CSCF Service

Use this example to bind an SSL template with a P-CSCF service. It also allows configuration of a non-default port for TLS.

```
configure

   context <context_name> -noconfirm

      cscf service <cscf_service_name> ssl-subscriber

         bind <ip_address> tls-crypto-template <tls_crypto_template_name> tls-port
<number>

         end
```

Appendix D
Sample Configuration Files

This appendix contains sample configuration files for the following SCM configurations:

- Proxy-CSCF Configuration
- Serving-CSCF Configuration
- A-BG Configuration

In each configuration example, commented lines are labeled with the number symbol (#) and variables are identified using italics within brackets (<*variable*>).

Proxy-CSCF Configuration

```
# Complete Configuration file for ASR 5000 in Proxy-CSCF role
#
# Send IMS licenses
configure /flash/flashconfig/<license_name>.cfg
end
#
# Set system to not require confirmation when creating new contexts and/or services.
Config file must end with "no autoconfirm" to return the CLI to its default setting.
#
configure
        autoconfirm
#
# Configure ASR 5000 cards
#
# Activate the PSCs
        card <slot_number>
           mode active psc
           exit
        card <slot_number>
           mode active psc
           exit
# Repeat for the number of PSCs in the system
#
# Modify the local context for local system management
        context local
           interface <interface_name>
              ip address <address> <mask>
```

```
            exit
         server ftpd
            exit
         server telnetd
            exit
         subscriber default
            exit
         administrator <name> encrypted password <password> ftp
#
#Set default IP route for local context
         ip route <ip_addr ip_mask> <next_hop_addr>
<local_context_interface_name>
         exit
#
# Configure Ethernet port for local context
      port ethernet <slot_number/port_number>
         no shutdown
         bind interface <local_context_interface_name> local
         exit
      end
#
# Create VPN context for P-CSCF service
configure
      context <p-cscf_context_name>
         interface <p-cscf_interface_name>
            ip address <address>
            exit
#
#Set the default subscriber for the P-CSCF context
```

```
            subscriber default
               exit
#
# Set default IP route for VPN context
            ip route 0.0.0.0 0.0.0.0 <next_hop_address> <vpn_interface_name>
            exit
#
# Configure Ethernet port for the VPN context
         port ethernet <slot_number/port_number>
            no shutdown
            bind interface <vpn_interface_name> <p-cscf_context_name>
            end
#
# Create the P-CSCF service in the P-CSCF VPN context
configure
         context <p-cscf_context_name>
            cscf service <p-cscf_service_name>
#
# Set the role of the service to P-CSCF
               proxy-cscf
                  allow rfc3261-ua-interworking
                  exit
#
# Bind an interface to the CSCF service
               bind address <ip_address> port <port_num>
#
# Enable Subscription package reg (reg-event package)
               subscription package reg
               exit
```

```
#
# Enable the session timers and set session-expires
               session-timer session-expires <sec>
#
# Set min-se
               session-timer min-se <sec>
#
# Set keepalive methods
               keepalive method crlf max-retry <value> expire-timer <value>
               keepalive method stun max-retry <value> expire-timer <value>
#
# Set default AoR domain
               default-aor-domain <alias>
#
# Set redirection recurse
               recurse-on-redirect-response
               exit
#
# Configure peer server list
            cscf peer-servers <name> type <type>
#
# Add a server to this list
               server <name> domain <domain_name> port <port_num>
               exit
#
# CSCF ACLs to permit or deny a CSCF session
            cscf acl default
               permit source aor $.
               exit
```

```
#
# CSCF route lists to define next-hop server address for a CSCF session
            cscf routes default
               exit
#
# CSCF policy to classify AoR policies
            cscf policy default
               exit
#
# CSCF session template to classify users/domains
            cscf session-template <name>
               inbound-cscf-acl default
               outbound-cscf-acl default
               route-list default
               translation-list default
               cscf-policy-profile default
               exit
#
#Configure additional P-CSCF Context parameters
#
# Configure domain name
            domain <name>
            end
#
# DNS client config
configure
         context <p-cscf_context_name>
            ip domain-lookup
            ip name-servers <ip_address>
```

```
            dns-client <name>
               bind address <ip_address>
               cache ttl positive <sec>
               cache ttl negative <sec>
               exit
            end
#
# Create local subscribers for SIP UAs
configure
         context <p-cscf_context_name>
            subscriber name <user_name>
               password <password>
               end
#
# Create AAA Group
configure
         context <p-cscf_context_name>
            aaa group default
               exit
#
# CDR Accounting service for calls over P-CSCF
            radius attribute nas-ip-address address <address>
            radius dictionary <dictionary_id>
            radius server <address> key <key> port <port_num>
            radius accounting server <address> key <key> port <port_num>
            end
#
# Configure Logging
logging filter active facility sessmgr level critical
```

```
logging filter active facility cscfmgr level critical

logging filter active facility cscf level critical

logging active

#

# Return system CLI to default setting of requiring confirmation when creating new
contexts and/or services.

#

configure

        no autoconfirm

        end

#
```

Serving-CSCF Configuration

```
# Complete Configuration file for ASR 5000 in Serving-CSCF role
#
# Send IMS licenses
configure /flash/flashconfig/<license_name>.cfg
end
#
# Set system to not require confirmation when creating new contexts and/or services.
Config file must end with "no autoconfirm" to return the CLI to its default setting.
#
configure
        autoconfirm
#
# Configure ASR 5000 cards
#
# Activate the PSCs
        card <slot_number>
           mode active psc
           exit
        card <slot_number>
           mode active psc
           exit
# Repeat for the number of PSCs in the system
#
# Modify the local context for local system management
        context local
           interface <interface_name>
              ip address <address> <mask>
```

```
            exit
         server ftpd
            exit
         server telnetd
            exit
         subscriber default
            exit
         administrator <name> encrypted password <password> ftp
#
# Set default IP route for local context
         ip route <ip_addr ip_mask> <next_hop_addr> <lcl_context_intfc_name>
         exit
#
# Configure Ethernet port for local context
      port ethernet <slot_number/port_number>
         no shutdown
         bind interface <local_context_interface_name> local
         exit
      end
#
# Create VPN context for S-CSCF service
configure
      context <s-cscf_context_name>
         interface <s-cscf_interface_name>
            ip address <address>
            exit
#
# Configure system access to an HSS
         ims-sh-service <name>
```

```
            diameter dictionary standard
            diameter endpoint <hss_host_name>
            exit
#
# Set the default subscriber for the S-CSCF context
            subscriber default
            exit
#
# Set default IP route for VPN context
            ip route 0.0.0.0 0.0.0.0 <next_hop_address> <vpn_interface_name>
            exit
#
# Configure Ethernet port for the VPN context
         port ethernet <slot_number/port_number>
            no shutdown
            bind interface <vpn_interface_name> <s-cscf_context_name>
            end
#
# Create the S-CSCF service in the S-CSCF VPN context
configure
         context <s-cscf_context_name>
            cscf service <s-cscf_service_name>
#
# Set the role of the service to S-CSCF
               serving-cscf
                  authentication allow-noauth invite
                  authentication allow-noipauth
                  registration lifetime min <sec> max <sec> default <sec>
                  allow rfc3261-ua-interworking
```

```
                exit
#
# Bind an interface to the CSCF service
             bind address <ip_address> port <port_num>
#
# Enable Subscription package reg (reg-event package)
             subscription package reg
#
# Set default AoR domain
             default-aor-domain <alias>
             exit
#
# Identify trusted network entities to the S-CSCF
             trusted-domain-entity <domain_name>
             trusted-domain-entity <domain_name>
                exit
#
# Configure CSCF Service access to the HSS for Call Features
             tas
             tas-service <ims-sh-service_name>
             exit
#
# Configure peer server list
          cscf peer-servers <name> type <type>
# Add a server to this list
             server <name> domain <domain_name> port <port_num>
             exit
#
# CSCF ACLs to permit or deny a CSCF session
```

```
            cscf acl default
               permit any
               permit source aor $.
               exit
#
# CSCF translation lists to re-address CSCF sessions
            cscf translation default
               uri-readdress type <tag> base-criteria destination aor <aor>
               exit
#
# CSCF route lists to define next-hop server address for a CSCF session
            cscf routes default
               exit
#
# CSCF session template to classify users/domains
            cscf session-template <name>
               inbound-cscf-acl default
               outbound-cscf-acl default
               route-list default
               translation-list default
               cscf-policy-profile default
               exit
#
# Optional: Configure integrated I-CSCF
            cscf service <s-cscf_service_name>
               proxy-cscf
                  interrogating-cscf-role
                  allow rfc3261-ua-interworking
                  exit
```

```
            exit
         aaa group default
            radius dictionary custom2
            diameter authentication dictionary aaa-custom4
            diameter authentication endpoint <hss_host_name>
            diameter authentication server <host_name> priority 1
            exit
         diameter endpoint <hss_host_name>
            origin realm <realm_name>
            origin host <host_name> address <ip_address>
            connection retry-timeout 1
            peer <auth_srv_host> realm <origin_realm_name> address <ip_addr>
#
# Configure additional S-CSCF Context parameters
#
# DNS client config
configure
      context <s-cscf_context_name>
         ip domain-lookup
         ip name-servers <ip_address>
         dns-client <name>
            bind address <ip_address>
            exit
         end
#
# Create AAA Group
configure
      context <s-cscf_context_name>
         aaa group default
```

```
            radius dictionary custom2
            diameter authentication dictionary aaa-custom4
            diameter authentication endpoint <hss_host_name>
            diameter authentication server <host_name> priority 1
            exit
#
# CDR Accounting service for calls over S-CSCF
            radius attribute nas-ip-address address <address>
            radius dictionary <dictionary_id>
            radius server <address> key <key> port <port_num>
            radius accounting server <address> key <key> port <port_num>
            end
#
# Configure Logging
logging filter active facility sessmgr level critical
logging filter active facility cscfmgr level critical
logging filter active facility cscf level critical
logging active
#
# Return system CLI to default setting of requiring confirmation when creating new
contexts and/or services.
#
configure
        no autoconfirm
        end
#
```

A-BG Configuration

```
# Complete Configuration file for ASR 5000 in Access-Proxy-CSCF role
#
# Send IMS licenses
configure /flash/flashconfig/<license_name>.cfg
end
#
# Set system to not require confirmation when creating new contexts and/or services.
Config file must end with "no autoconfirm" to return the CLI to its default setting.
#
configure
        autoconfirm
#
# Configure ASR 5000 cards
#
# Activate the PSCs
        card <slot_number>
           mode active psc
           exit
        card <slot_number>
           mode active psc
           exit
# Repeat for the number of PSCs in the system
#
# Modify the local context for local system management
        context local
           interface <interface_name>
              ip address <address> <mask>
```

```
            exit
         server ftpd
            exit
         server telnetd
            exit
         subscriber default
            exit
         administrator <name> encrypted password <password> ftp
#
#Set default IP route for local context
         ip route <ip_addr ip_mask> <next_hop_addr>
<local_context_interface_name>
         exit
#
# Configure Ethernet port for local context
      port ethernet <slot_number/port_number>
         no shutdown
         bind interface <local_context_interface_name> local
         exit
      end
#
# Create VPN context for P-CSCF service
configure
      context <p-cscf_context_name>
         interface <p-cscf_interface_name>
            ip address <address>
            exit
#
#Set the default subscriber for the P-CSCF context
```

```
            subscriber default
               exit
#
# Set default IP route for VPN context
            ip route 0.0.0.0 0.0.0.0 <next_hop_address> <vpn_interface_name>
            exit
#
# Configure Ethernet port for the VPN context
         port ethernet <slot_number/port_number>
            no shutdown
            bind interface <vpn_interface_name> <p-cscf_context_name>
            end
#
# Create the P-CSCF service in the P-CSCF VPN context
configure
         context <p-cscf_context_name>
            cscf service <p-cscf_service_name>
#
# Set the role of the service to P-CSCF
               proxy-cscf
                  allow rfc3261-ua-interworking
                  exit
## Bind an interface to the CSCF service
               bind address <ip_address> port <port_num>
#
# Enable Subscription package reg (reg-event package)
               subscription package reg
               exit
#
```

```
# Enable the session timers and set session-expires
               session-timer session-expires <sec>
#
# Set min-se
               session-timer min-se <sec>
#
# Configure nat-pool
               nat-pool name <core_pool_name>
#
# Set default AoR domain
               default-aor-domain <alias>
#
# Set redirection recurse
               recurse-on-redirect-response
#
# Configure access service
               access-service name <access_proxy_name>
               exit
#
# Configure peer server list
            cscf peer-servers <name> type <type>
#
# Add a server to this list
               server <name> domain <domain_name> port <port_num>
               exit
#
# CSCF ACLs to permit or deny a CSCF session
            cscf acl default
               permit source aor $.
```

```
            exit
#
# CSCF route lists to define next-hop server address for a CSCF session
            cscf routes default
               exit
#
# CSCF policy to classify AoR policies
            cscf policy default
               exit
#
# CSCF session template to classify users/domains
            cscf session-template <name>
               inbound-cscf-acl default
               outbound-cscf-acl default
               route-list default
               translation-list default
               cscf-policy-profile default
               exit
#
#Configure additional P-CSCF Context parameters
#
# Configure domain name
            domain <name>
            end
#
# DNS client config
configure
         context <p-cscf_context_name>
            ip domain-lookup
```

```
            ip name-servers <ip_address>
            dns-client <name>
               bind address <ip_address>
               cache ttl positive <sec>
               cache ttl negative <sec>
               exit
            end
#
# Create local subscribers for SIP UAs
configure
         context <p-cscf_context_name>
            subscriber name <user_name>
               password <password>
               end
#
# Create AAA Group
configure
         context <p-cscf_context_name>
            aaa group default
               exit
#
# CDR Accounting service for calls over P-CSCF
            radius attribute nas-ip-address address <address>
            radius dictionary <dictionary_id>
            radius server <address> key <key> port <port_num>
            radius accounting server <address> key <key> port <port_num>
            end
#
# Create context for access_proxy service
```

```
configure
        context <access_pcscf_context_name>
           ip pool <nat_pool> range <start_address> <end_address> napt-users-per-ip-
address <num_users> port-chunk-size <ports_per_user> nat-binding-timer <seconds>
           interface <p-cscf_interface_name>
              ip address <address>
              exit
#
#Set the default subscriber for the access-pcscf context
           subscriber default
              exit
#
# Set default IP route for access-pcscf context
           ip route 0.0.0.0 0.0.0.0 <next_hop_address> <vpn_interface_name>
           exit
#
# Configure Ethernet port for the access_pcscf context
        port ethernet <slot_number/port_number>
           no shutdown
           bind interface <access-pcscf_interface_name> <access-
pcscf_context_name>
           end
#
# Create the access_proxy service in the access-pcscf context
configure
        context <access_pcscf_context_name>
           cscf service <access_proxy_service_name>
#
# Set the role of the service to access_proxy
              proxy-cscf
```

```
                allow rfc3261-ua-interworking
                exit
#
# Bind an interface to the access_proxy service
                bind address <ip_address> port <port_num>
#
# Configure core service
                core-service name <proxy_cscf>
#
# Configure nat-pool
                nat-pool name <access_pool_name>
#
# Set default AoR domain
                default-aor-domain <alias>
#
# Set keepalive methods
                keepalive method crlf max-retry <value> expire-timer <value>
                keepalive method stun max-retry <value> expire-timer <value>
#
# CSCF policy to classify AoR policies
             cscf policy <access_policy>
                exit
#
# Configure Logging
logging filter active facility sessmgr level critical
logging filter active facility cscfmgr level critical
logging filter active facility cscf level critical
logging active
#
```

```
# Return system CLI to default setting of requiring confirmation when creating new
contexts and/or services.

#

configure

        no autoconfirm

        end

#
```

Appendix E
SCM Engineering Rules

This appendix provides SCM-specific engineering rules or guidelines that must be considered prior to configuring the ASR 5000 for your network deployment. General and network-specific rules are located in the appendix of the *System Administration and Configuration Guide* for the specific network type.

The following rules are covered in this appendix:

- SCM Context and Service Rules
- SCM Subscriber Rules
- AoR Regular Expression Rules
- Session Recovery Rules

SCM Context and Service Rules

- Multiple SCM services can be configured in the same context (the general rules of 256 maximum services per system and 64 maximum contexts per system apply)
- SCM services configured within the same context cannot communicate with each other
- When running collapsed with an access service such as the HA, the CSCF service correlates its call-line with the corresponding HA service call-line. If the HA service call goes down, the CSCF service aborts its call.

SCM Subscriber Rules

- When running collapsed with an access service such as the HA, the CSCF service correlates its call-line with the corresponding HA service call-line. If the HA service call goes down, the CSCF service aborts its call.

AoR Regular Expression Rules

Regular expressions can be used in **source aor** and **destination aor** keywords. Individual characters, sometimes referred to as wildcards or meta characters, can be used to create AoR ranges or broader groups to which rules or policies can be applied.

Meta Characters

Currently, the following meta characters are supported:

- "$." (dollar period): can be used in the username, domain, or sub-domain portion of the AoR. The following examples show how this character can be used:
  - $.@Provider.com - matches all users from the "Provider" domain
  - $.@$..com - matches all users with a ".com" domain only
  - mobile$.@Provider.com - matches "Provider" users who have an AoR starting with "mobile"
- "$" (dollar sign): use to substitute any single character. Example:
  - $11 matches 911, 411, etc.
- "%" (percent symbol): use to signify the start of a pattern such as add/delete/substitute for translations.

AoR Regular Expression Patterns

The **uri-readdress aor** keyword found in the Translation Configuration mode, supports the use of regular expression patterns. Individual characters, sometimes referred to as wildcards or meta characters, can be used to create AoR ranges or broader groups to which rules or policies can be applied. In a regular expression pattern, the meta character "%" is used to signify the beginning of an add, delete, or substitute command used for translations.

The syntax of a pattern is:

- **%-***num***p**
- **%+***num***s***sub*
- **%***num***t**
- **%+p***sub*

Character/Variable	Description
-	Delete
+	Add
num	Numeric character up to 32.
p	Prefix
s	Suffix
t	Truncate
sub	Substitute alpha and/or numeric string or "-" (hyphen) or "." (dot)

Syntax examples:

- **%-*num*p**: Removes (**-**) specified number (*num*) of characters from the prefix (**p**) of the username.
- **%+*num*s*sub***: Adds (**+**) specified number (*num*) of characters (*sub*) to suffix (**s**) of the username.
- **%*num*t**: Truncates (**t**) the username to a specified number (*num*) of characters.
- **%+p*sub***: Adds (+) specified number (*num*) to prefix of a dial number.

Practical examples:

- **%-*3*p**: Deletes first three characters from the prefix
- **%+*3*s*111***: Adds 111 as the suffix
- **%*10*t**: Truncates the username to 10 characters
- **%+p*sub***: Translation from number 23XY to 155588823XY using the following command:

```
uri-readdress user %+p1555888 base-criteria destination aor 23$.
```

Session Recovery Rules

RFC 3261 Proxy

- Only one call context in the call leg can be recovered. If the call leg is in multiple calls, only the active primary call context will be recovered after a sessmgr task failure.
- Session recovery should be enabled before the CSCF service creation.