Oracle® Communications

Diameter Signaling Router SCEF User's Guide

Release 8.3

E93572

September 2018

Oracle Communications Diameter Signaling Router SCEF User's Guide, Release 8.3

E93572

Contents

1 Introduction

2 SCEF Functional Summary

List of Figures

List of Tables

1

Introduction

This chapter describes the Oracle Communications DSR Service Capability Exposure Function (SCEF) product, which interacts with, and implements controls on, Internet of Things (IoT) devices.

Machine Type Communication (MTC) is the communication between wired and wireless devices. It can enable a sensor or meter to communicate data (such as temperature, inventory level, etc.) to software at another location for its use. For example, sending the number of kilowatts of power used by an individual's home to the billing software at a utility company; or a refrigerator sending a user's smart phone information about what may be needed at the grocery store. The expansion of IP networks around the world has made machine-to-machine communication quicker and easier and it uses less power. These networks also allow new business opportunities for consumers and suppliers.

The end-to-end communications (between the user's equipment and the network), uses services provided by a 3rd Generation Partnership Project (3GPP) system, and optionally services provided by a Services Capability Server (SCS). The MTC application in the external network is typically hosted by an Application Server (AS) and may make use of an SCS for additional value-added services. The 3GPP system provides transport, subscriber management, and other communication services including various architectural enhancements motivated by, but not restricted to, MTC (for example, control plane device triggering).

The SCS connects to the 3GPP network to communicate with user equipment (UE) used for MTC and the MTC Interworking Function (MTC-IWF) and/or SCEF in the Home Public Land Mobility Network (HPLMN). The SCS offers capabilities for use by one or multiple MTC applications and the UE can host one or multiple MTC applications. The corresponding MTC applications in the external network are hosted on one or multiple ASs.

The SCEF is the key entity within the 3GPP architecture for service capability exposure that provides a means to securely expose the services and capabilities provided by 3GPP network interfaces. In certain deployments, the MTC-IWF may be co-located with SCEF in which case MTC-IWF functionality is exposed to the SCS/AS through the T8 interface (that is, the REST API). In deployments where MTC-IWF is not co-located with SCEF, interactions between MTC-IWF and SCEF are left up to the implementation.

SCEF allows services and capabilities to be securely used on 3GPP network interfaces by:

- providing a way to discover the exposed services and capabilities;
- providing access to network capabilities through homogenous network application programming interfaces (for example, network APIs) defined over the T8 interface; and

- abstracting the services from the underlying 3GPP network interfaces and protocols.

This document describes the how the configuration and administration of SCEF through a machine-to-machine interface (MMI) affects works with DSR and how various screens within DSR provide you with SCEF information.

Revision History

Date	Description
September 2018	Initial release of DSR SCEF

Intended Scope and Audience

This content is intended for personnel who plan to provision SCEF.

The content does not describe how to install, update, or replace software or hardware.

Content Organization

This content is organized as follows:

- Introduction contains general information about SCEF including an overview and logic information, the organization of this content, and how to get technical assistance.
- DSR SCEF Architecture describes how SCEF is configured within DSR.
- Configure SCEF describes how to access SCEF.
- Managed Objects describes the managed objects used to build the SCEF.
- SCEF MMI Attributes describes the MMI attributes used with the SCEF.

Understanding SCEF

DSR has been enhanced to support the capabilities of a Service Capability Exposure Function (SCEF). SCEF is a new network element that securely exposes the servers and capabilities provided by 3GPP network interfaces. Some functions included with SCEF include:

- Non-IP data delivery (NIDD) for low power devices

  Functions for NIDD are used to handle mobile originated (MO) and mobile terminated (MT) communication with UE, where the data used for the communication is considered unstructured from the Evolved Packet System (EPS) standpoint (which we refer to also as non-IP). The support of non-IP data is part of the Consumer Internet of Things (CIoT) EPS optimizations.

- Monitoring a device's state

  The Monitoring Events feature monitors specific events in the 3GPP system and makes the monitoring events information available using SCEF. It allows the identification of the 3GPP network element suitable for configuring the event, the event detection, and the event reporting to the authorized users, for use by applications or logging. If an event is detected, the network can be configured to perform special actions like limit the UE access.

- Device triggering performs application-specific action including communication with the Service Capability Server (SCS)

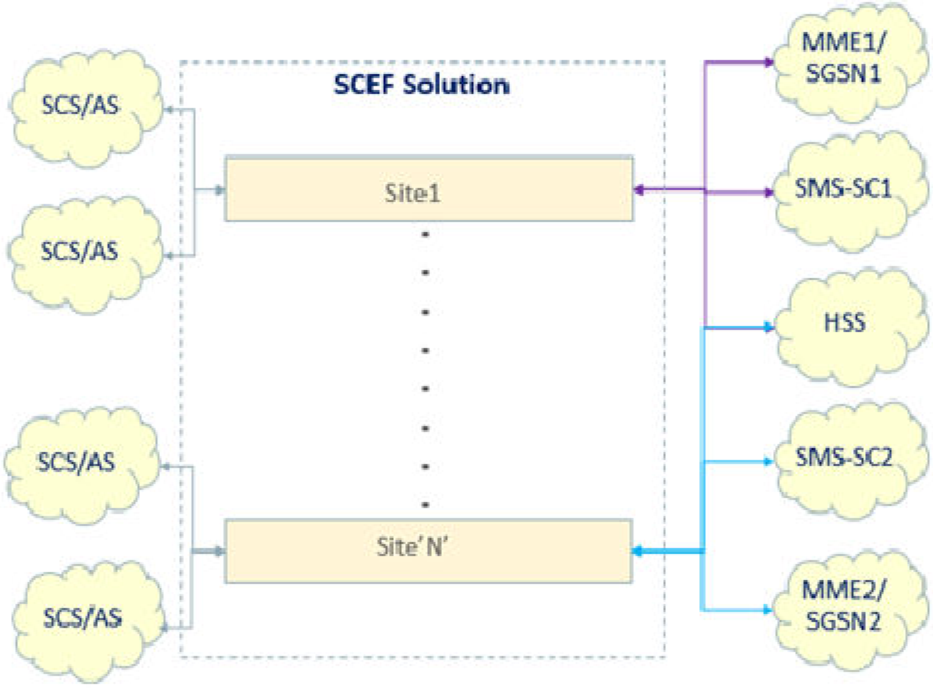
  Device Triggering allows SCS to send information to the UE through the 3GPP network to trigger the UE to perform application-specific actions that include initiating communication with SCS for the indirect model or an AS in the network for the hybrid model. Device Triggering is required when an IP address for the UE is not available or reachable by SCS/AS.

- Enhanced Coverage Restriction Control

  The support for Enhanced Coverage Restriction Control using SCEF enables 3rd party service providers to query status of enhanced coverage restriction, or enable/disable enhanced coverage restriction per individual UE.

The SCEF server interacts with Internet of Things (IoT) networks as a machine type communication inter-working function (MTC-IWF). Figure 1-1 shows how SCEF interacts with other DSR elements and an IoT network.

Figure 1-1 DSR SCEF Interactions



IoT devices have unique identifiers and can transmit data over a network. An IoT network can consist of numerous devices, characterized by simple design, low power consumption, brief and infrequent data transmissions, and infrequent machine transmissions (mostly they are not transmitting). The SCEF server supports IoT devices through non-IP data delivery (NIDD). An SCEF server can relay triggers from an SMS-SC function to IoT devices using Short Message Service (SMS) messages through the Diameter T4 interface. An SCEF server communicates with the home subscriber server (HSS) using the Diameter S6t and S6m interfaces. An SCEF server communicates with mobility management entity (MME) functions using the Diameter T6a and T6b interfaces. An SCEF server generates charging records and communicates with charging servers using the Diameter Ga interface. Table 1-1 provides a summary of these supported reference points.

Table 1-1 Supported Diameter Reference Points

Reference Point Name	Description
T4	Reference point used between SCEF and SMS-SC/GMSC/IWMSC
T6a	Reference point used between SCEF and serving MME
T6b	Reference point used between SCEF and serving SGSN
T8	Reference point used between SCEF and SCS/AS
S6t	Reference point used between SCEF and HSS
S6m	Reference point used between MTC-IWF and HSS

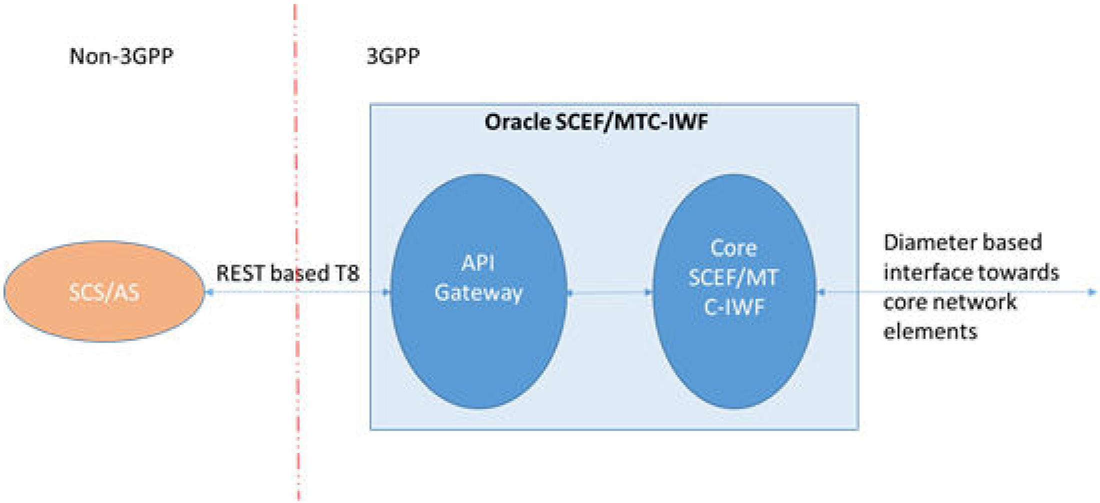
An SCEF network communicates with services capability server/application server (SCS/AS) functions using either the T8 otherwise known as the WebSocket representational state transfer (REST)ful application program interface (API) protocols using the DSR API gateway, which provides a proxy API gateway with trusted identity management, IP multimedia subsystem (IMS) access, quality of service (QoS) control, messaging services, and industry-standard security, authentication, accounting, and authorization. Configurable, extensible mechanisms support applying rate, volume, and other limits on a per-SCS/AS basis.

To support large network environments, an SCEF network can communicate with charging, home subscriber server (HSS), and mobility management entity (MME) servers using DSR.

Major Functional Components of an SCEF Network

DSR implements the functionality of both SCEF and MTC-IWF network elements. SCEF/MTC-IWF functionality at DSR can be split into two functional components as depicted in Figure 1-2:

Figure 1-2 SCEF/MTC-IWF functionality at DSR



- The API Gateway provides the northbound interface between SCEF and Services Capability Server/Application Server (SCS/AS) based on a T8 interface. The T8

APIs are a set of RESTful APIs defining the related procedures and resources for the interaction between the SCEF and the SCS/AS.

- The Core SCEF/MTC-IWF provides southbound interface toward core network elements like HSS, MME/SGSN (Serving GPRS Support Node), and Policy and Charging Rules Function (PCRF). The following MTC call procedures are implemented at the core SCEF/MTC-IWF component with DSR acting as SCEF and MTC-IWF network elements:
  - Device triggering function (MTC-IWF functionality)
  - Non-IP data delivery
  - Monitoring event
  - Enhance coverage restriction control

API Gateway

SCEF provides a means to securely expose the services and capabilities provided by 3GPP network interfaces. The API gateway provides access to network capabilities through homogenous application programming interfaces (that is, network RESTful APIs). The SCEF API gateway provides the secured gateway functionality implementing the northbound RESTful T8 interface toward SCS/AS. DSR has implemented the following functionalities at the API Gateway:

- Northbound T8 RESTful interface
- Authentication and authorization functionality for API requests from SCS/AS:
  - Identification of the API consumer (for example, SCS/AS)
  - Profile management of SCS/AS and management of service level agreements per SCS/AS
  - Support ACL (access control list) management for individual SCS/AS
- API firewall functionality to protect from security attacks through T8 interface:
  - Protection against malformed and oversized messages received from SCS/AS
  - Whitelist of IP addresses of SCS/AS

Core SCEF/MTC-IWF

Core SCEF/MTC-IWF implements the business logic of different MTC functional call procedures specific to SCEF and MTC-IWF network elements. Core SCEF/MTC-IWF interfaces with the API gateway to send or receive the T8 requests from SCS/AS.

DSR has implemented the following MTC functional procedures of SCEF/MTC-IWF network elements.

- Non-IP Data Delivery (NIDD) provides a path to exchange unstructured data between UE and SCS/AS without requiring the user equipment (UE) to support an IP stack.

  Eliminating the need to support IP results in the following benefits:
  - Reduces device complexity since there is no need to support TCP/IP
  - Reduces device cost due to lower complexity

  - Reduces device power consumption due to eliminating extra messaging and overhead related to TCP/IP
  - Compatibility with older devices not supporting IP

  NIDD, using the SCEF, is handled using a PDN connection to the SCEF. The UE may obtain a non-IP PDN connection to the SCEF during the attach procedure; using UE requested PDN; or using the PDP context activation procedure. An association between the SCS/AS and a PDN connection to the SCEF needs to be established to enable transfer of non-IP data between the UE and the SCS/AS. The SCEF determines the association based on provisioned policies that may be used to map an SCS/AS identity and user identity to an access point name (APN). SCEF supports both mobile terminated (MT) and mobile originated (MO) NIDD communication between UE and SCS/AS.

- The Monitoring Events feature monitors specific events in the 3GPP system and makes monitoring events information available through SCEF. This means you can identify the 3rd Generation Partnership Project (3GPP) network element suitable for configuring specific events, event detection, and event reporting to the authorized users, for example, for use by applications or logging. If such an event is detected, the network might be configured to perform special actions, for example, limit UE access.

  DSR supports the following monitoring events configuration and deletion using HSS:

  - LOSS_OF_CONNECTIVITY

    Notifies the AS when the UE loses connection and becomes offline, which signals device abnormality and need for troubleshooting.

  - UE_REACHABILITY

    Allows AS to know the status of the devices as reachable or not reachable.

  - LOCATION_REPORTING

    Allows the AS (enterprise) to track the location of the devices without GPS modules (cargo tracking).

  - CHANGE_OF_IMSI_IMEI(SV)_ASSOCIATION

    Allows AS to detect stolen devices.

  - ROAMING_STATUS

    Allows the SCS/AS to query the UE's current roaming status (the serving public land mobility network (PLMN) and/or whether the UE is in its home PLMN (HPLMN)) and notifies when that status changes.

  - UE_REACHABILITY

    Allows AS to know the status of the devices as reachable or not reachable with a status flag (idleStatusIndication flag = true).

  DSR supports both a single report event and a continuous event report for the requested monitoring events from SCS/AS. DSR supports both monitoring requests for a group of UE or single UE.

- The Enhanced Coverage Restriction Control enables 3rd party service providers (that is, SCS/AS) to query status, enhance coverage restriction, or enable/disable enhanced coverage restriction per individual UE.

- The Device Triggering feature allows the SCS/AS to deliver a specific device trigger to the UE through SCEF. The Device Trigger request is authenticated with HSS using the User Identifier received in the request. After successful authentication, SCEF forwards the Device Trigger request to the corresponding SMS-SC to be delivered to the UE.

Overview of Main Tasks

The major tasks involved with using SCEF and DSR, described in the remainder of this document, are:

- Configuring the SCEF and DSR topology
- Managing SCEF devices
- Configuring network protocols with which SCEF devices communicate
- Defining network elements with which SCEFdevices interact
- Monitoring the operation and performance of SCEF

Documentation Admonishments

Admonishments are icons and text throughout this manual that alert the reader to assure personal safety, to minimize possible service interruptions, and to warn of the potential for equipment damage.

Table 1-2 Admonishments

Icon	Description
DANGER	Danger: (This icon and text indicate the possibility of personal injury.)
WARNING	Warning: (This icon and text indicate the possibility of equipment damage.)
CAUTION	Caution: (This icon and text indicate the possibility of service interruption.)

Locate Product Documentation on the Oracle Help Center Site

Oracle Communications customer documentation is available on the web at the Oracle Help Center site, http://docs.oracle.com. You do not have to register to access these documents. Viewing these files requires Adobe Acrobat Reader, which can be downloaded at http://www.adobe.com.

1. Access the Oracle Help Center site at http://docs.oracle.com.
2. Click **Industries**.

3. Under the Oracle Communications subheading, click **Oracle Communications documentation** link.

   The Communications Documentation page displays. Most products covered by these documentation sets display under the headings Network Session Delivery and Control Infrastructure and Platforms.

4. Click on your product and then the release number.

   A list of the documentation set for the selected product and release displays.

5. To download a file to your location, right-click the **PDF** link, select **Save target as** (or similar command based on your browser), and save to a local folder.

Customer Training

Oracle University offers training for service providers and enterprises. Visit our web site to view, and register for, Oracle Communications training at http://education.oracle.com/communication

To obtain contact phone numbers for countries or regions, visit the Oracle University Education web site at www.oracle.com/education/contacts

My Oracle Support

My Oracle Support (https://support.oracle.com) is your initial point of contact for all product support and training needs. A representative at Customer Access Support can assist you with My Oracle Support registration.

Call the Customer Access Support main number at 1-800-223-1711 (toll-free in the US), or call the Oracle Support hotline for your local country from the list at http://www.oracle.com/us/support/contact/index.html. When calling, make the selections in the sequence shown below on the Support telephone menu:

1. Select **2** for New Service Request

2. Select **3** for Hardware, Networking and Solaris Operating System Support

3. Select one of the following options:

   - For Technical issues such as creating a new Service Request (SR), select **1**

   - For Non-technical issues such as registration or assistance with My Oracle Support, select **2**

You are connected to a live agent who can assist you with My Oracle Support registration and opening a support ticket.

My Oracle Support is available 24 hours a day, 7 days a week, 365 days a year.

Emergency Response

In the event of a critical service situation, emergency response is offered by the Customer Access Support (CAS) main number at 1-800-223-1711 (toll-free in the US), or by calling the Oracle Support hotline for your local country from the list at http://www.oracle.com/us/support/contact/index.html. The emergency response provides immediate coverage, automatic escalation, and other features to ensure that the critical situation is resolved as rapidly as possible.

A critical situation is defined as a problem with the installed equipment that severely affects service, traffic, or maintenance capabilities, and requires immediate corrective action. Critical situations affect service and/or system operation resulting in one or several of these situations:

- A total system failure that results in loss of all transaction processing capability
- Significant reduction in system capacity or traffic handling capability
- Loss of the system's ability to perform automatic system reconfiguration
- Inability to restart a processor or the system
- Corruption of system databases that requires service affecting corrective actions
- Loss of access for maintenance or recovery operations
- Loss of the system ability to provide any required critical or major trouble notification

Any other problem severely affecting service, capacity/traffic, billing, and maintenance capabilities may be defined as critical by prior discussion and agreement with Oracle.

2

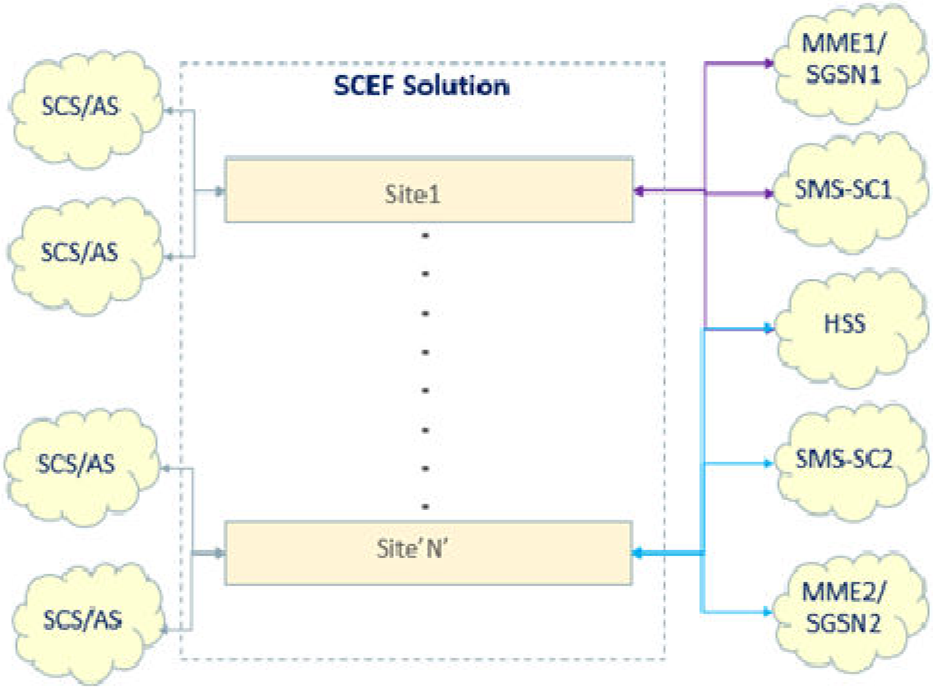
SCEF Functional Summary

This section provides a high-level summary of the SCEF functionality as it relates to DSR.

DSR SCEF Architecture

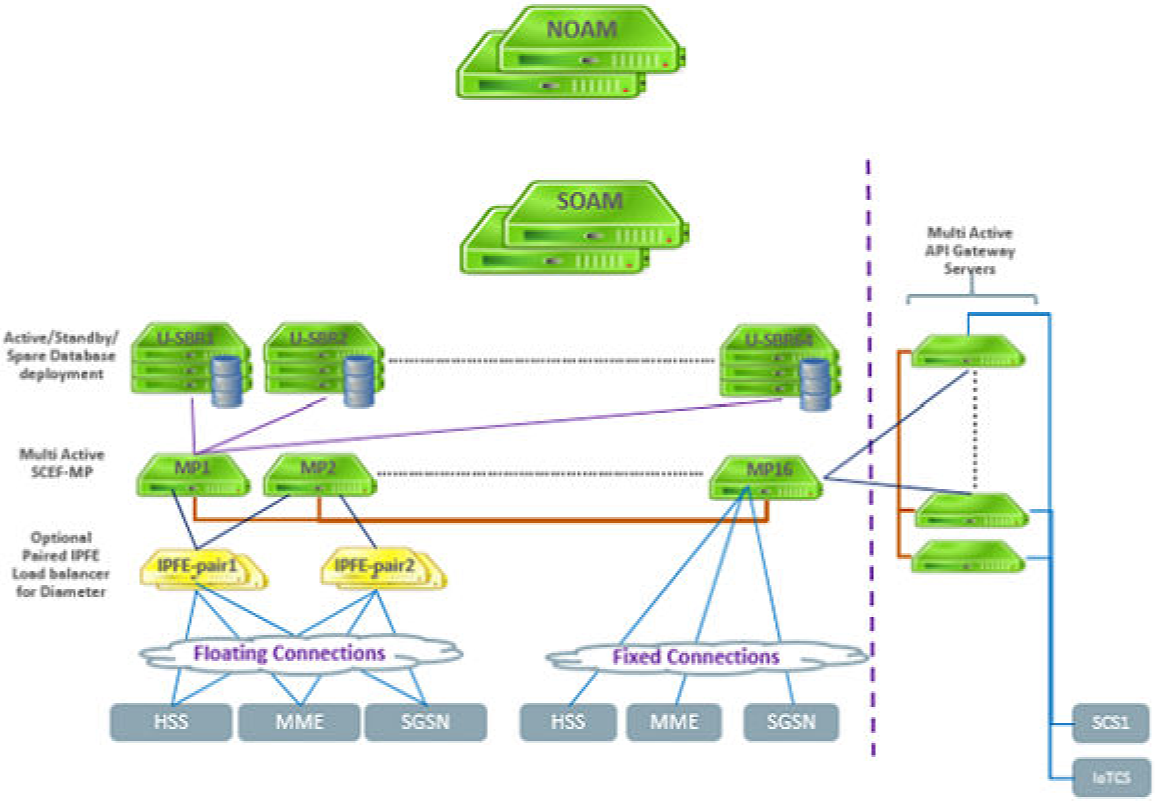
The SCEF solution, supported by a DSR network, contains one or more DSR nodes (sites). Each DSR node may be connected to 3GPP entities, like MME/SGSN, SMS-SC, and HSS, in the trusted domain; and the SCS/AS in the trusted and/or untrusted domain. The connectivity of these nodes with the DSR network is shown in Figure 2-1.

Figure 2-1 DSR-SCEF Interconnections



The DSR architecture is shown in Figure 2-2.

Figure 2-2 DSR SCEF Architecture



The solution has the following components:

- An API gateway to manage the REST interface(s) for the following:
  - Authentication of SCS/AS
  - Support for API lifecycle
  - Profile management
  - Quota and rate management
  - Load balance HTTP traffic among the DA-MP servers
- Network OAM servers deployed in active-standby redundancy model for configuration and maintenance of the DSR topology.
- Site OAM servers deployed in one, two, or three site redundancy model for provisioning of the SCEF administration data.
- IPFE servers (optional) to load balance the Diameter traffic.
- DA-MP server(s) for processing the HTTP (REST) and Diameter signaling according to the provisioning done through the site OAM servers. The DA-MPs receive the HTTP signaling traffic from the SCS/AS using the DSR API gateway application servers and the Diameter signaling traffic from the IPFE servers, if present, or from the connected Diameter peers directly. Diameter traffic generated from DA-MP servers is set to the Diameter peers directly and the HTTP traffic

generated from the DA-MP servers shall be routed to the SCS/AS using the DSR API gateway.

- U-SBR server(s) deployed in one, two, or three site redundancy model for caching context data. This data is volatile, that is, the data does not persist with a server reboot, therefore, it is important to plan an adequate redundancy model.

Each SCS/AS may have a configured quota and rate for T8 messages. For example, a quota of 1000 messages in 24 hours at a rate of no more than to 100 messages per hour. Such restrictions are enforced by the DSR API gateway. If the DSR API gateway determines the rate and/or quota to be exhausted, it responds with an appropriate error message to SCS/AS. If the quota and rate are found to be within limits, the DSR API gateway forwards the T8 message to one of the DSR MP servers chosen using a simple round-robin load-sharing algorithm.

For sending a T8 request message to the SCS/AS, the DSR MP servers forward the T8 message to one of the DSR API gateway servers chosen using a simple round-robin load-sharing algorithm.

The DSR MP servers provide the SCS/AS URL in an `X-callback-url` header and provide the callback type as defined in Table 2-1 in a `X-callback-type` header to the DSR API gateway.

Table 2-1 DSR API Gateway Callback Types

X-notification-type	Notification Description
1	Monitoring Event Notification
2	Device Triggering Delivery Report
3	NIDD Uplink Data Notification
4	NIDD Downlink Data Delivery Status Notification

HTTP Message Parsing

The SCEF application parses HTTP messages as defined in 3GPP TS 29.122 specifications, T8 Reference Point for Northbound Application Programming Interfaces (APIs). The Swagger templates for the T8 messages are available on the Oracle Help Center site. See Locate Product Documentation on the Oracle Help Center Site to go to the latest release of the Diameter Signaling Router and then open theService Capability Exposure Function (SCEF) YAML ZIP file.

The SCEF application receives and processes HTTP messages for Non-IP Data Delivery (NIDD), Monitoring Event, Enhanced Coverage Restriction Control, and Device Triggering APIs. The content of such messages is encoded in JSON format.

The API contained in the HTTP message is identified by a message URI prefix similar to that described in Table 2-2.

Table 2-2 Supported T8 APIs

API Name	URI Prefix
Non-IP Data Delivery	/3gpp_t8_nidd
Monitoring Events	/3gpp_t8_monitoring_event
Device Triggering	/3gpp_t8_device_triggering

Table 2-2 (Cont.) Supported T8 APIs

API Name	URI Prefix
Enhanced Coverage Restriction Control	/3gpp_ecr_control

Database Integrity Audits

Database Integrity Audits help SCEF identify and remove alternate key records that are stale and/or pointing to invalid context records. These audits are initiated when SCEF detects that a context record retrieved using an alternate key does not point to an appropriate context. This ability will be implemented in a future DSR release.

Error Reporting

HTTP Error Reporting

The SCEF application generates an error response when an HTTP request fails to get processed successfully. The SCEF application inserts an error cause whenever possible for easy of debugging. The error cause is contained in json format for requests of type POST/PATCH/PUT. The content of the detail attribute of the problem json structure is formatted as:

```
SCEF-ERR-XXX-YYY: <human readable text description of the error>
```

where XXX is the HTTP Response Code and YYY is a 3-digit problem code Table 2-3.

Table 2-3 Problem Codes for HTTP Error Reporting

Problem Code	Problem Details
099	Generic Error
100	The configuration sets were not found for the given SCS/AS.
101	Stack-Event deserialization failed
102	An internal database error was encountered
103	A Diameter response did not contain requisite parameters to complete the transaction
104	NIDD Authorization/Grant time received from the HSS is in the past
105	A Diameter error response was received due to which the current HTTP transaction cannot be processed
106	An unexpected response was received from the Database (USBR) server
107	A Database integrity error was detected for the (IMSI, APN) Alternate Key

Table 2-3 (Cont.) Problem Codes for HTTP Error Reporting

Problem Code	Problem Details
108	The HTTP message contains an invalid JSON content
109	The HTTP message contains a JSON content that failed schema validation
110	A context record was not found in the USBR database
111	The downlink data delivery packet was rejected as the data size exceeds the configured maximum limit
112	A USBR read request failed
113	A message or event was received that was not expected in the current state of the transaction context
114	An internal error was encountered while processing an NIDD transaction
115	A PDN connection was not found for the User Entity
116	A downlink data delivery message could not be buffered because it contains a T8 Transaction Identifier that is already in use by one of the buffered data messages
117	A downlink data delivery message was rejected as it did not contain any data
118	A downlink data delivery message could not be processed as the downlink data rate limit has been reached
119	The MME was not able to deliver the downlink data to the UE
120	A parameter value in the request message is not supported (in the current version)
121	The transaction could not be processed as the operation is not valid in the current transaction state
122	Unused
123	Unused
124	The downlink data delivery message could not be buffered as the packet size exceeds the maximum allowed size for a packet that can be buffered

Table 2-3 (Cont.) Problem Codes for HTTP Error Reporting

Problem Code	Problem Details
125	The downlink data delivery message could not be buffered as the maximum latency is too small
126	The downlink data delivery message could not be buffered as the number of currently buffered messages is at the configured maximum
127	The transaction request was failed as the UE is not authorized by Access Control
128	The transaction request as failed as the feature requested is not enabled for the requesting SCS/AS
129	The transaction request as failed as the feature requested is not enabled for the requesting UE
130	The API version requested is not supported
131	The HTTP message did not contain a mandatory parameter
132	The transaction was failed as the requesting SCS/AS is not configured

Diameter Error Reporting

Diameter error reporting problem codes will be introduced in a future DSR release.

Non-IP Data Delivery

Functions for NIDD may be used to handle mobile originated (MO) and mobile terminated (MT) communication with UEs, where the data used for the communication is considered unstructured from the EPS standpoint (which we refer to also as Non-IP). The support of Non-IP data is part of the CIoT EPS optimizations. The Non-IP data delivery to SCS/AS is accomplished by using SCEF.

NIDD via the SCEF is handled using a PDN connection to the SCEF. The UE may obtain a Non-IP PDN connection to the SCEF either during the Attach procedure or via UE requested PDN connectivity or via PDP Context Activation Procedure.

An association between the SCS/AS and the SCEF needs to be established to enable transfer of non-IP data between the UE and the SCS/AS.

NIDD via SCEF uses the User Identity to identify which UE a particular T6a/T6b connection belongs to. The User Identity is the user's IMSI. The user's IMSI shall not be used on the interface between SCEF and SCS/AS. In order to perform NIDD configuration or to send or receive NIDD data, the SCS/AS shall use MSISDN or External Identifier to identify the user. In order to facilitate correlation of SCS/AS requests to T6a/T6b connection for a given UE, the HSS provides to the SCEF the user's IMSI, and if available, the MSISDN (when NIDD Configuration Request contains an External Identifier) or if available, External Identifier (when NIDD Configuration Request contains an MSISDN).

The NIDD procedure requested by SCS/AS is determined from the URI as described in .

Table 2-4 Supported NIDD Resources and Methods

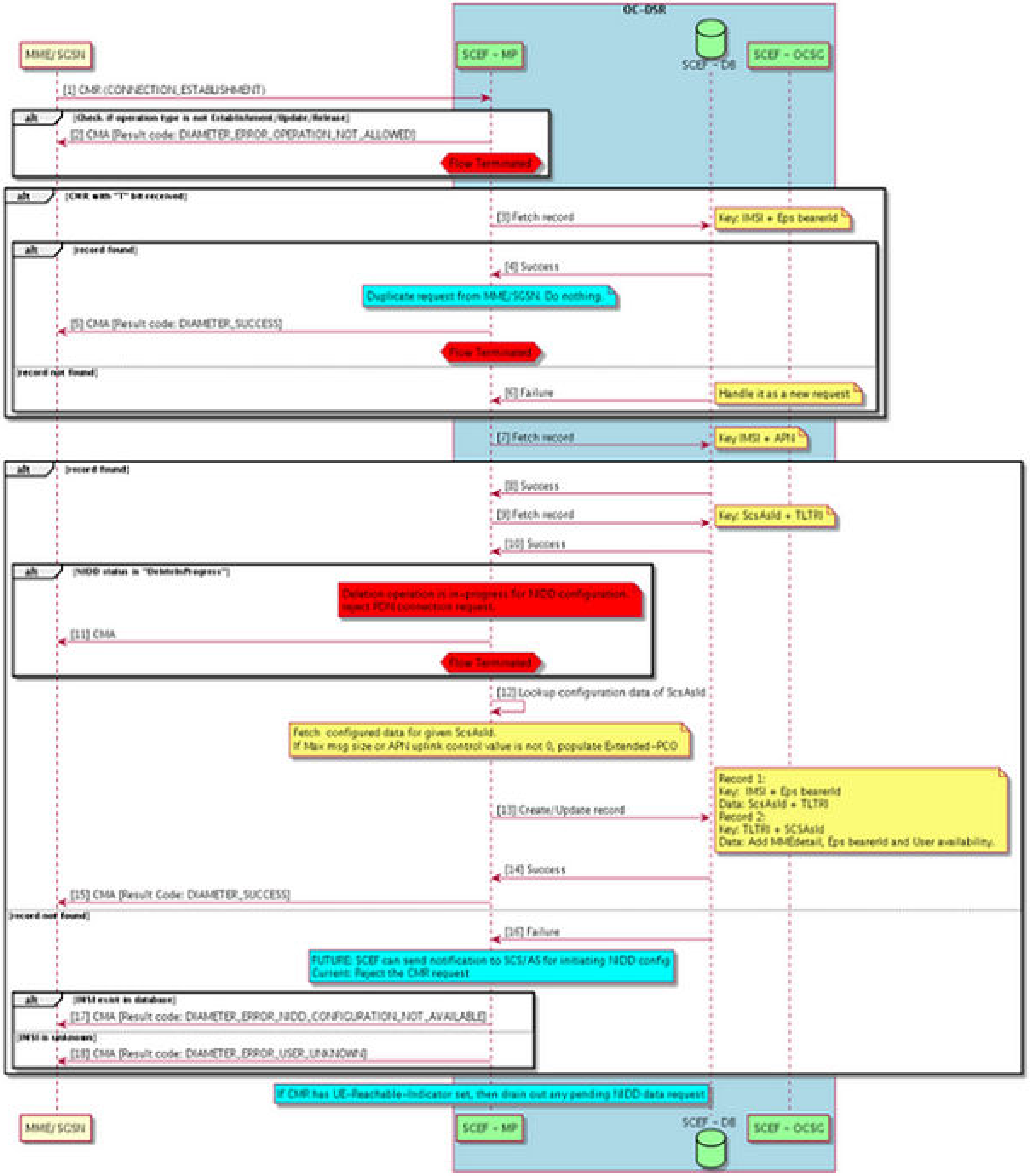
Resource Name	Resource URI	HTTP Method(s)	HTTP Initiator
NIDD Configurations	3gpp_t8_nidd/v1/{scsAsId}/configurations	POST	SCS/AS
Individual NIDD Configurations	3gpp_t8_nidd/v1/{scsAsId}/configurations/{tltrId}	PATCH, GET, DELETE	SCS/AS
NIDD Downlink Data Deliveries	3gpp_t8_nidd/v1/{scsAsId}/configurations/{tltrId}/downlink_data_deliveries	POST	SCS/AS
Individual NIDD Downlink Data Deliveries	3gpp_t8_nidd/v1/{scsAsId}/configurations/{tltrId}/downlink_data_deliveries/{ttrId}	PUT, GET	SCS/AS
NIDD Downlink Data Delivery Status Notification	{notification_destination_uri}	POST	SCEF
NIDD Uplink Data Notification	{notification_destination_uri}	POST	SCEF

PDN Connection

PDN Connection Establishment

Figure 2-3illustrates the procedure of PDN connection establishment. When the UE performs the EPS attach procedure with PDN type of Non-IP, and the subscription information corresponding to either the default APN for PDN type of Non-IP or the UE requested APN includes the **Invoke SCEF Selection** indicator, then the MME initiates a T6a/T6b connection toward the SCEF corresponding to the **SCEF ID** indicator for that APN.

Figure 2-3   PDN Connection Establishment


OC-DSR
MME/SGSN
SCEF - MP
SCEF - DB
SCEF - OCSG
[1] CMR (CONNECTION_ESTABLISHMENT)
alt
[Check if operation type is not Establishment/Update/Release]
[2] CMA [Result code: DIAMETER_ERROR_OPERATION_NOT_ALLOWED]
Flow Terminated
alt
[CMR with "T" bit received]
[3] Fetch record
Key: IMSI + Eps bearerId
alt
[record found]
[4] Success
Duplicate request from MME/SGSN. Do nothing.
[5] CMA [Result code: DIAMETER_SUCCESS]
Flow Terminated
[record not found]
[6] Failure
Handle it as a new request
[7] Fetch record
Key IMSI + APN
alt
[record found]
[8] Success
[9] Fetch record
Key: ScsAsId + TLTRI
[10] Success
alt
[NIDD status is "DeleteInProgress"]
Deletion operation is in-progress for NIDD configuration.
reject PDN connection request.
[11] CMA
Flow Terminated
[12] Lookup configuration data of ScsAsId
Fetch configured data for given ScsAsId.
If Max msg size or APN uplink control value is not 0, populate Extended-PCO
Record 1:
Key: IMSI + Eps bearerId
Data: ScsAsId + TLTRI
Record 2:
Key: TLTRI + SCSAsId
Data: Add MMEdetail, Eps bearerId and User availability.
[13] Create/Update record
[14] Success
[15] CMA [Result Code: DIAMETER_SUCCESS]
[record not found]
[16] Failure
FUTURE: SCEF can send notification to SCS/AS for initiating NIDD config
Current: Reject the CMR request
alt
[IMSI exist in database]
[17] CMA [Result code: DIAMETER_ERROR_NIDD_CONFIGURATION_NOT_AVAILABLE]
[IMSI is unknown]
[18] CMA [Result code: DIAMETER_ERROR_USER_UNKNOWN]
If CMR has UE-Reachable-Indicator set, then drain out any pending NIDD data request
MME/SGSN
SCEF - MP
SCEF - DB
SCEF - OCSG

The MME/SGSN creates a PDN connection toward the SCEF and allocates an EPS Bearer Identity (EBI) to that PDN connection. The MME/SGSN does so by sending a Create SCEF Connection Request (User Identity, EPS Bearer Identity, SCEF ID, APN, Serving PLMN Rate Control, Serving PLMN ID, IMEISV) message toward the SCEF. If the IWK-SCEF receives the Create SCEF Connection Request message from the MME/SGSN, it forwards it toward the SCEF.

The combination of EPS Bearer Identity, APN, and User Identity allows the SCEF to uniquely identify the PDN connection to the SCEF for a given UE. If no SCS/AS has performed the NIDD Configuration procedure with the SCEF for the User Identity, then the SCEF rejects the T6a/T6b connection setup with a cause **NIDD Configuration Not Available**.

The SCEF saves the EPS Bearer information in its Context for the user identified using User Identity and EBI. The SCEF sends a Create SCEF Connection Response (User Identity, EPS Bearer Identity, APN, PCO) message towards the MME/SGSN confirming establishment of the PDN connection to the SCEF for the UE. If the IWK-SCEF receives the Create SCEF Connection Response message from the SCEF, it forwards it toward the MME/SGSN.

PDN Connection Update

The MME/SGSN may update certain parameters that were provided in the T6a/T6b connection establishment request by sending a connection update message to SCEF. The MME/SGSN identifies the T6a/T6b connection by the IMSI and EPS Bearer Identifier. The MME/SGSN may update these parameters by a connection update message:

- Serving PLMN
- RAT Type
- Serving PLMN Rate threshold
- Origin Host and/or Origin-Realm of the MME/SGSN

The SCEF finds the context record in its database and if found updates the parameters provided in the connection update request. The SCEF then responds with the result of the update operation to the MME/SGSN.

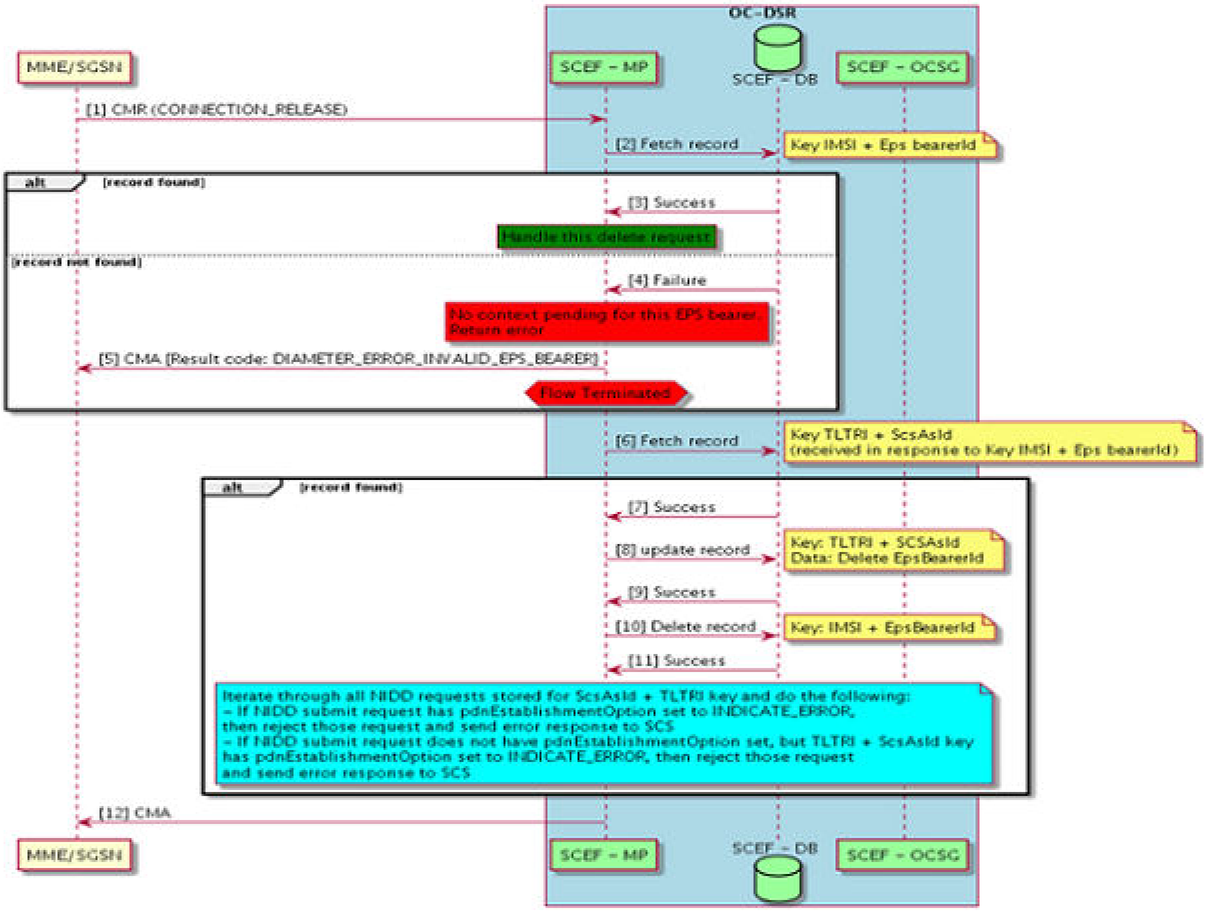
PDN Connection Release

The MME/SGSN releases the T6a/T6b connection(s) towards the SCEF(s) corresponding to the "SCEF ID" indicator for that APN when the UE or MME/SGSN or HSS initiates a detach procedure.

The SCEF releases the T6a/b connection(s) towards the MME/SGSN corresponding to PDN connections when an NIDD Authorization Update Request from the HSS indicates that the User is no longer authorized for NIDD, or the Granted Validity Time for the NIDD configuration provided by the HSS expires or based on a NIDD configuration deletion request from the SCS/AS.

Figure 2-4 illustrates the procedure of T6a/T6b connection release when initiated by the MME/SGSN.

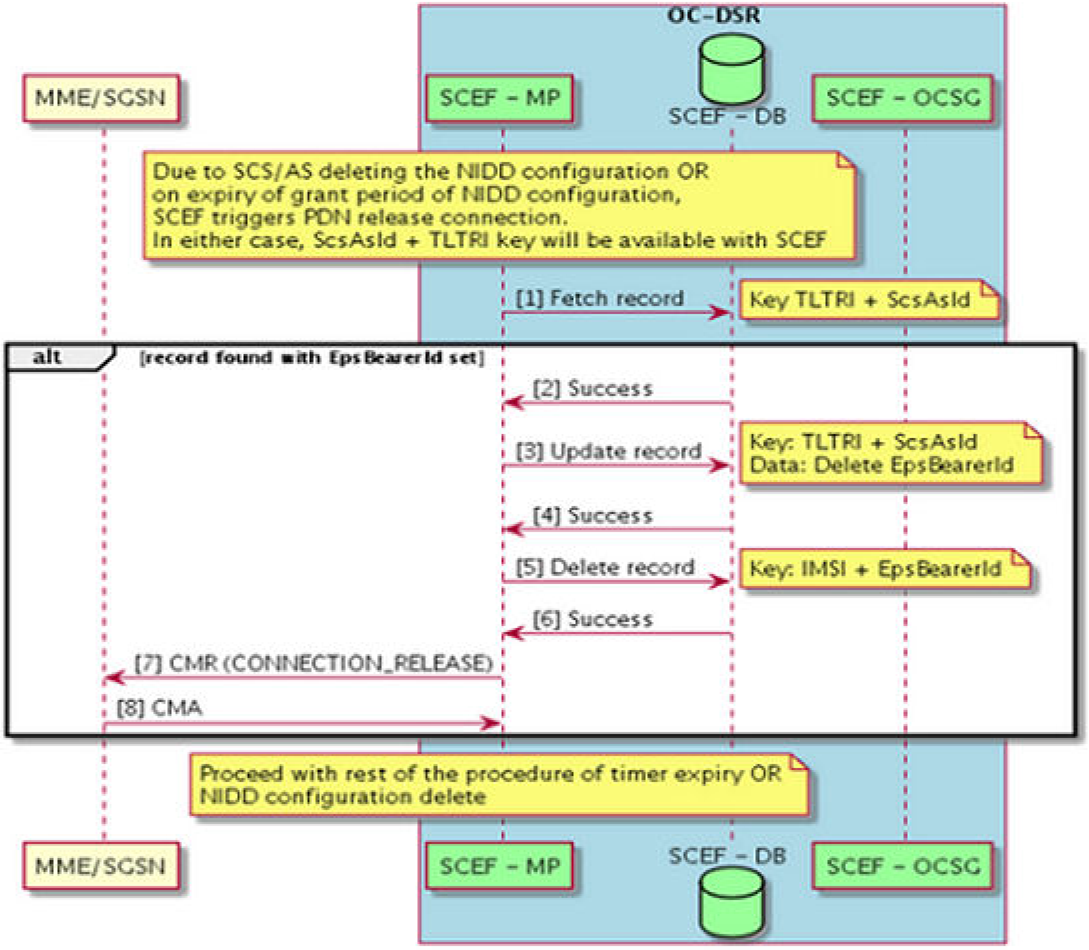
Figure 2-4 MME/SGSN Initiated PDN Connection Release



Iterate through all NIDD requests stored for ScsAsId + TLTRI key and do the following:
- If NIDD submit request has pdnEstablishmentOption set to INDICATE_ERROR,
then reject those request and send error response to SCS
- If NIDD submit request does not have pdnEstablishmentOption set, but TLTRI + ScsAsId key
has pdnEstablishmentOption set to INDICATE_ERROR, then reject those request
and send error response to SCS

Figure 2-5 illustrates the procedure of T6a/T6b connection release when initiated by the SCEF.

Figure 2-5 SCEF-Initiated PDN Connection Release


OC-DSR
MME/SGSN
SCEF - MP
SCEF - DB
SCEF - OCSG
Due to SCS/AS deleting the NIDD configuration OR on expiry of grant period of NIDD configuration, SCEF triggers PDN release connection. In either case, ScsAsId + TLTRI key will be available with SCEF
[1] Fetch record
Key TLTRI + ScsAsId
alt
[record found with EpsBearerId set]
[2] Success
[3] Update record
Key: TLTRI + ScsAsId Data: Delete EpsBearerId
[4] Success
[5] Delete record
Key: IMSI + EpsBearerId
[6] Success
[7] CMR (CONNECTION_RELEASE)
[8] CMA
Proceed with rest of the procedure of timer expiry OR NIDD configuration delete
MME/SGSN
SCEF - MP
SCEF - DB
SCEF - OCSG

Configuration Query by SCS Application Server

The SCS/AS may request the NIDD configuration data that is saved with the SCEF using a NIDD Configuration GET request. SCEF looks for the SCS/AS Identifier and the TLTRI provided in the request and, if found, includes these parameters stored in the SCEF's database in the response.

- User Identity (External Identifier or MSISDN)
- SCS AS Identifier
- TLTRI
- NIDD Duration
- NIDD Notification Destination Address
- List of buffered Downlink Data Delivery Packets

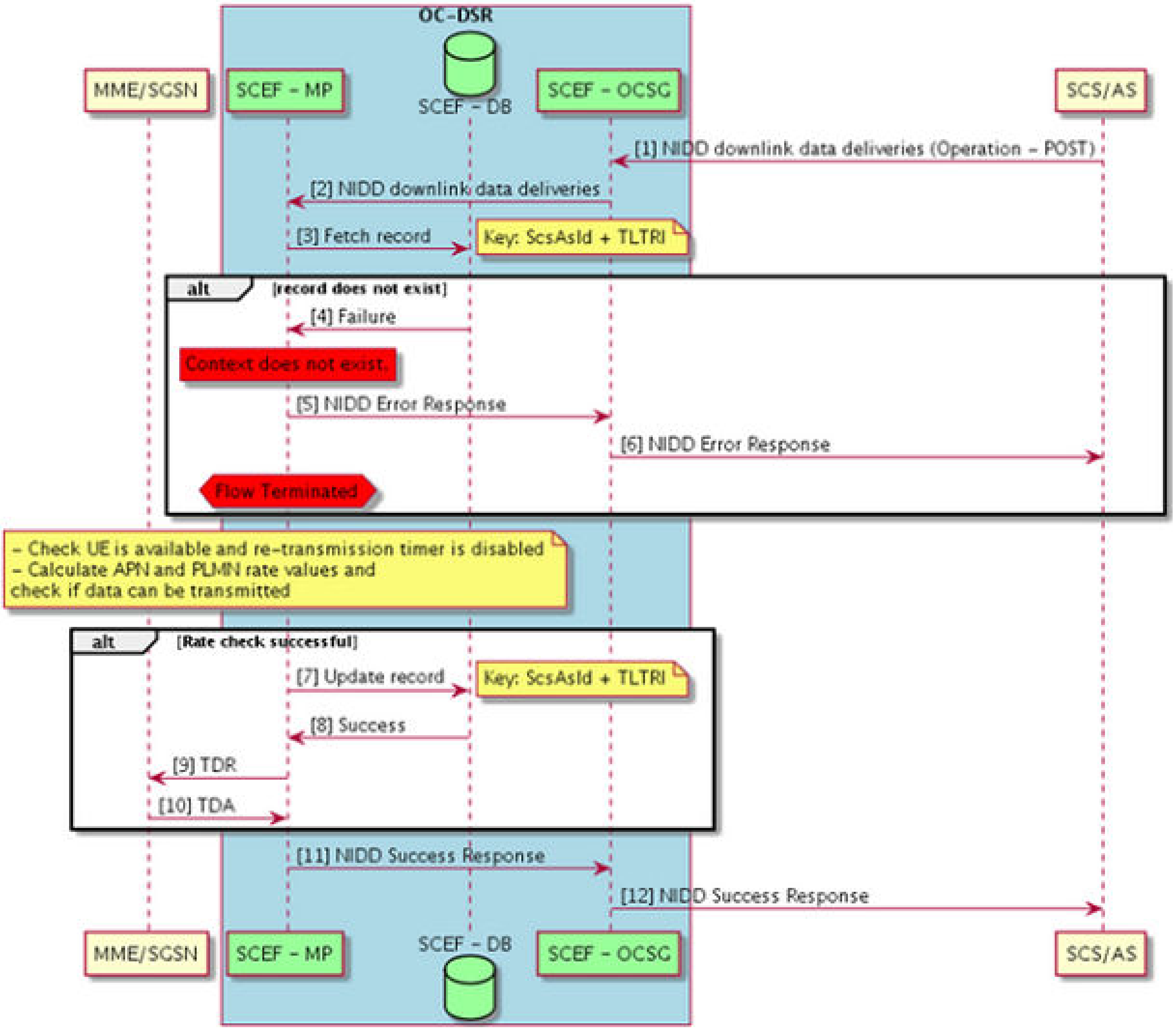
If the TLTRI requested by the SCS/AS is not found in the SCEF's database, the SCEF responds with the 404 Not Found error.

Downlink Data Delivery

Figure 2-6 illustrates the procedure SCS/AS uses to send non-IP data to a given user as identified using the External Identifier or MSISDN.

If SCS/AS has already activated the NIDD service for a given UE, and has downlink non-IP data to send to the UE, the SCS/AS sends an NIDD Submit Request containing the External Identifier or MSISDN and the non-IP data message to the SCEF.

Figure 2-6 Downlink Data Delivery



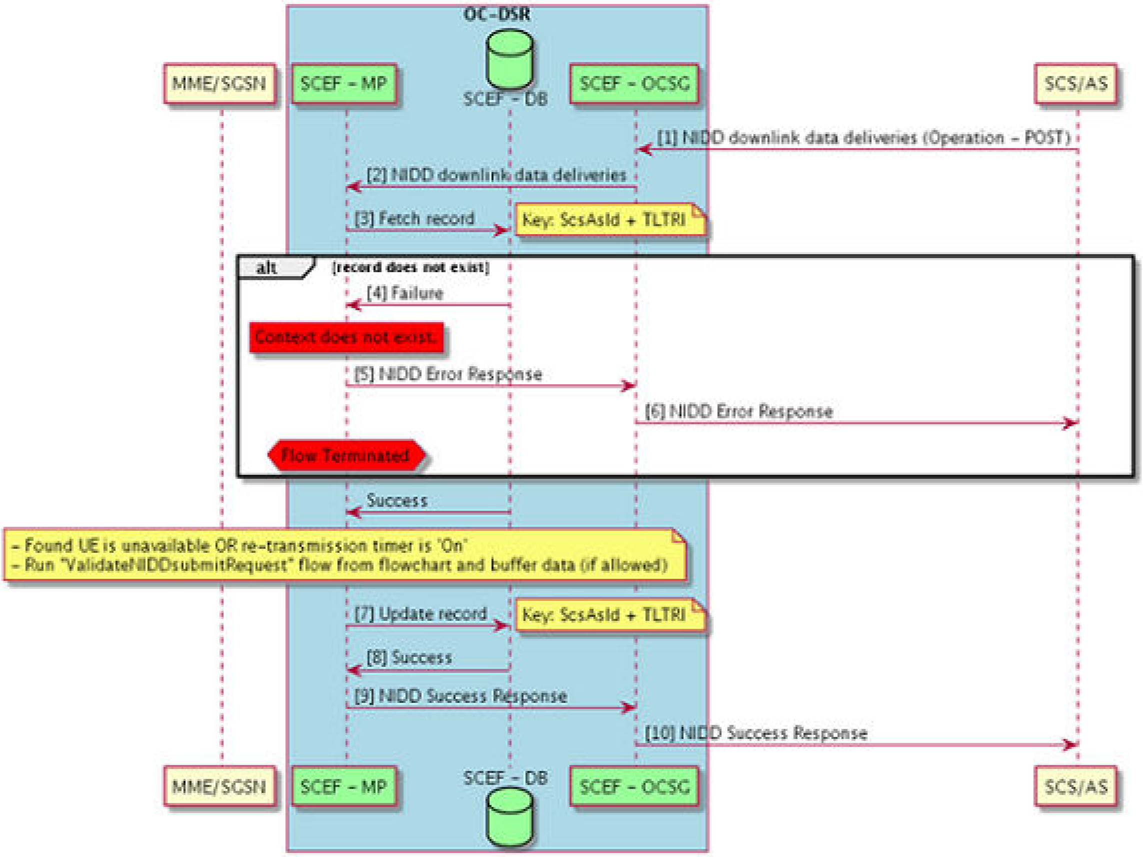
If an SCEF EPS bearer context corresponding to the External Identifier or MSISDN is found and the UE, according to the context of SCEF, is currently in a connected/ reachable state, the SCEF determines whether the data delivery message rate is within the configured APN downlink rate and the Serving PLMN rate as received in the T6a/T6b connection establishment request from the MME/SGSN. If the SCEF finds the downlink data delivery message is within the rate thresholds, it attempts to send a Mobile Terminating Data message to the MME/SGSN. The SCEF also informs the MME/SGSN of the duration of time that it can wait for a response from the MME/ SGSN and the duration of time up to which it can re-attempt to send the data messages. If the MME/SGSN finds the UE to be in a connected state, it attempts to deliver the data message to the UE. If the MME/SGSN cannot deliver the message within the time mentioned by the SCEF, it responds with an appropriate cause "UE temporarily not reachable." If the MME/SGSN knows when the UE is expected to be in connected state, it may inform the same to the SCEF in the Requested-Retransmission-Time parameter of the response. If the MME/SGSN is not aware when the UE may be reachable again, it stores, in its context, the SCEF Identity so that it can inform the SCEF when the UE becomes reachable.

If the SCEF does not a have an EPS bearer setup for the UE, the UE is not reachable, or the response from the MME/SGSN indicates the UE is not currently reachable, the SCEF tries to buffer the downlink data message. If the data message could be successfully buffered by SCEF, it responds with the 202 Accepted code to the SCS/AS and indicates the data is buffered. If the SCEF could not buffer the message, it indicates the cause of failure to the SCS/AS.

Data Buffering at SCEF

Figure 2-7 illustrates the procedure SCS/AS uses to send non-IP data to a given user as identified using the External Identifier or MSISDN, and the SCEF uses to decide to buffer the data message to deliver at a later point of time.

Figure 2-7 SCEF Buffering Downlink Data as UE is Not Available



Downlink data is buffered by SCEF under these conditions:

- There is no PDN connection with the MME/SGSN for the UE requested.

  When there is no PDN connection with the MME/SGSN, the PDN Establishment Option is considered in the following order of preference (most preferred first) to decide whether or not to buffer the data:

  - PDN Establishment Option received from SCS/AS in the downlink data message
  - PDN Establishment Option received from SCS/AS in the NIDD configuration message
  - As configured in the NIDD Configuration Set managed object

- A previous attempt to deliver a downlink data message was responded by the MME/SGSN with a cause of "UE temporarily" not reachable.

  The UE reachability status has not been updated further by the MME/SGSN. In this case the SCEF does not attempt to send a data delivery request to MME/SGSN, rather it tries to buffer the data as soon as it receives it from SCS/AS.

- The MME/SGSN has informed the status of the UE that it is not reachable using a T6a/T6b connection establish or update request.
- The current attempt to deliver the data message was responded by the MME/SGSN with a cause of "UE temporarily" not reachable.

The following conditions need to be met for the downlink data to be buffered by SCEF:

- Data buffering must not be disabled by setting the data message lifetime to zero.

  This data duration configuration can be found in the NIDD Configuration Set managed object and has a default value of 0, that is, Data Buffering is disabled by default.

- The maximum latency of the downlink data message must be at least two times the minimum time taken for retransmitting a buffered message.

  The minimum retransmission time can be configured in the NIDD Configuration Set managed object and has a default value of 5 seconds.

- The downlink data payload size must be less than the configuration maximum packet size allowed to be buffered.

  This configuration can be found in the APN Configuration Set managed object with the default value of 100 bytes.

- There must be room to fit the downlink data message in the buffer queue for the UE.

  The length of the queue is configurable in the NIDD Configuration Set managed object with a default value of 1.

While attempting to buffer a downlink data message, assuming that all other conditions listed are found to be satisfactory, however, the queue is found to be full and the data message attempting to getting buffered has a higher priority than any message already present in the queue, the higher priority data message takes the place of the lowest priority message. A data delivery status notification shall be sent to the SCS/AS with a cause of "FAILURE" for the message that exists in the queue.

Any downlink data message that is buffered at SCEF resides in the data delivery queue for a maximum time as indicated by Maximum Latency attribute of the message. The maximum time is further capped by the data duration configuration parameter in the NIDD Configuration Set managed object.

Data Retransmissions

SCEF attempts to retransmit data messages that it has buffered in these scenarios:

- On expiry of a re-transmission timer that was started when the SCEF received a requested retransmission time parameter from the MME/SGSN for a data delivery request that could not be delivered by the MME/SGSN as the UE was temporarily not reachable.
- On receiving a T6a/T6b connection update message indicating the UE is now reachable.

Downlink Data Delivery Status Notification

The downlink data messages that are buffered by SCEF are either retransmitted or they expire sitting in the delivery queue. In either case, a Downlink Data Delivery Status notification is generated by SCEF and sent to the SCS/AS using the DSR API Gateway. The SCEF used the Notification Destination Address provided by the SCS/AS at the time of NIDD configuration, if provided, or the configuration in the SCS/AS managed object.

The status notification may contain one of these codes:

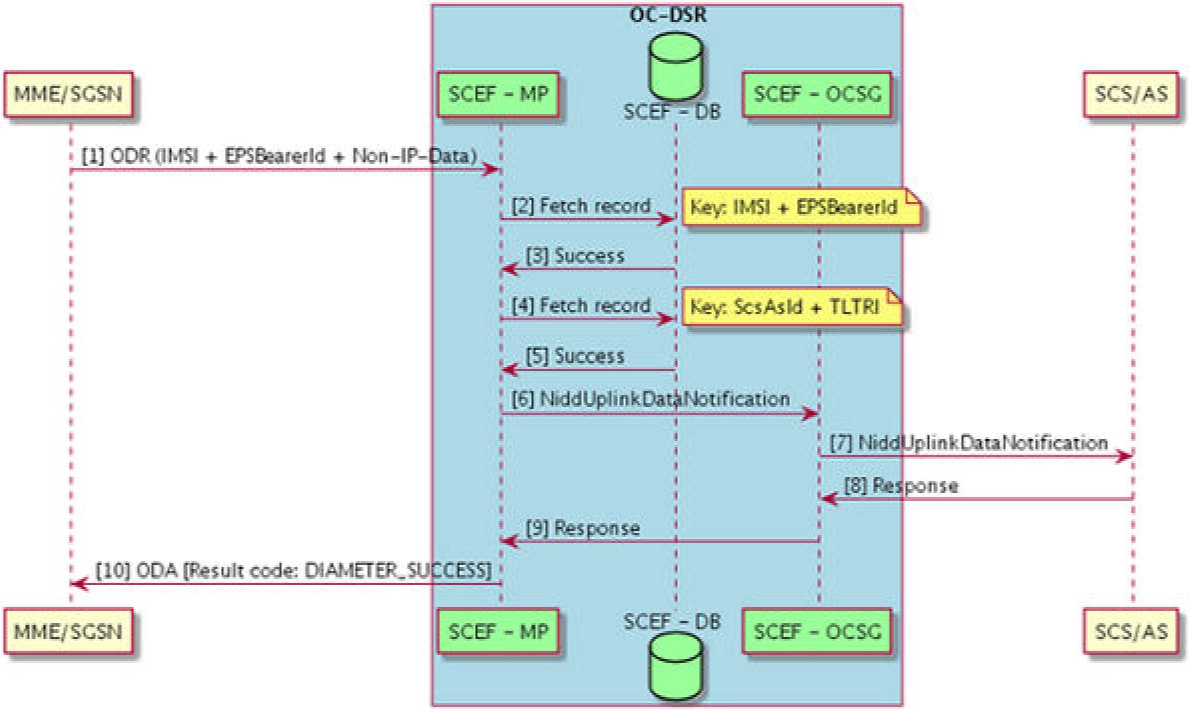
- FAILURE: When the retransmission attempt failed or the data lifetime expired.
- SUCCESS: When the data message could successfully be delivered by the MME/SGSN to the UE.

The SCEF included the TTRI of the data message in the notification for the SCS/AS to identify the same.

Uplink Data Notification

illustrates the procedure MME/SGSN uses to send non-IP uplink data to SCEF for delivery to SCS/AS.

Figure 2-8 Uplink Data Notification



The UE sends a NAS message with EPS bearer ID and non-IP data to the MME. The MME/SGSN sends the NIDD Mobile Originated Data Request containing User Identity (IMSI), EPS Bearer Identifier, and non-IP data message to SCEF. When SCEF receives the non-IP data on the T6a/T6b interface, and finds an SCEF context, it determines whether the uplink message rate is within the configured APN uplink rate. If SCEF finds the uplink data message is within the rate thresholds, it sends the non-IP data to the appropriate SCS/AS using the Notification Destination Address provided by the SCS/AS at the time of NIDD configuration, if provided, or the configuration in the SCS/AS managed object.

Note: The configured Uplink Data Rate is conveyed to the MME/SGSN and in turn to the UE in the Protocol Configuration Options IE in the T6a/T6b connection establishment answer, so it is not usually expected for the UE to send uplink data at a rate higher than that configured.

Monitoring Event

The Monitoring Events feature monitors specific events in the 3GPP system and makes monitoring event information available using SCEF. It identifies the 3GPP network element suitable for configuring specific events, event detection, and event reporting to the authorized users, for example, for use by applications or logging. If such an event is detected, the network can be configured to perform special actions, for example, limit UE access.

The Monitoring Event procedure requested by SCS/AS is determined from the URI as described in Table 2-5

Table 2-5 Supported Monitoring Event Resources and Methods

Resource Name	Resource URI	HTTP Method(s)	HTTP Initiator
Monitoring Event Subscriptions	3gpp_t8_monitoring_event/v1/{scsAsId}/subscriptions/	POST	SCS/AS
Individual Monitoring Event Subscription	3gpp_t8_monitoring_event/v1/{scsAsId}/subscriptions/{tltrId}	GET, DELETE	SCS/AS
Monitoring Event Notification	{notificationDestination}	POST	SCEF

Supported Monitoring Events include:

- LOSS_OF_CONNECTIVITY
- UE_REACHABILITY
- LOCATION_REPORTING
- CHANGE_OF_IMSI_IMEI_ASSOCIATION
- ROAMING_STATUS

Monitoring Event Subscription

To subscribe a new monitoring event configuration, the SCS/AS sends an HTTP POST message to the SCEF. The body of the HTTP POST message includes the SCS/AS Identifier, TLTRI, TTRI, and Monitoring Type; and may include External Identifier(s) or MSISDN(s) or External Group ID, Maximum Number of Reports, Monitoring Duration, T8 Destination Address, and Group Reporting Guard Time, wherein, the External Identifier or MSISDN indicates the subscription for an individual UE and the External Group ID indicates a group of UEs.

Note: SCEF always gives higher preference to an External Identifier when both Identifiers (External Identifier and MSISDN) are present in the Monitoring Event Configuration Request message.

The SCS/AS sends a Monitoring Request (External Identifier or MSISDN or External Group ID, SCS/AS Identifier, TTRI, TLTRI, Monitoring Type, Maximum Number of Reports, Monitoring Duration, T8 Destination Address, and Group Reporting Guard Time) message to the SCEF.

If the SCS/AS wants to configure Monitoring Event for the group of UEs, the SCS/AS can send a Monitoring Request message including External Group Identifier and Group Reporting Guard Time. A Group Reporting Guard Time is an optional parameter to indicate aggregated Monitoring Event Reporting(s), which has been detected for the UEs in a group, needs to be sent to the SCS/AS once the Group Reporting Guard Time is expired.

The SCEF stores the SCS/AS Identifier, T8 Destination Address, Monitoring Duration, and Maximum Number of Reports. The SCEF stores the received TLTRI and assigns it to an SCEF Reference ID.

The SCEF sends a Monitoring Request (External Identifier or MSISDN or External Group Identifier, SCEF ID, SCEF Reference ID, Monitoring Type, Maximum Number of Reports, and Monitoring Duration) message to the HSS to configure the given Monitoring Event on the HSS in Configuration-Information-Request (CIR) message.

After processing, HSS sends a Configuration-Information-Answer (CIA) message. Then according to the result code received in the CIA message, if the result code is Success (2001), the SCEF sends a Monitoring Response (TTRI, Cause, and Monitoring Event Report) message to the SCS/AS to acknowledge acceptance of the Monitoring Request; if the result code is not successful, then an error result code informs the SCS/AS about the error occurred/received

Monitoring Event Notification

Notification in Reporting-Information-Request (RIR) from HSS

This procedure is used between the HSS and the SCEF, whenever HSS needs to send a report in RIR.

When the procedure is invoked by the HSS, it is used for reporting the:

- Change of association of the UE and UICC and/or new IMSI-IMEI-SV;
- UE reachability for SMS; and
- Roaming status (Roaming or No Roaming) of the UE, and change in roaming status of the UE.

It is also used to:

- Update the SCEF with the suspend/resume/cancel status of an ongoing monitoring.

  Only **Cancel** is supported for current SCEF release.
- Convey reports and/or status indications for all or some UEs belonging to a group.

For group based configuration processing, if the Group Guard Timer was included in the CIR command, the HSS sends the RIR command before the Group Guard Timer expires and includes several reports and/or status indications in one or more Group-Monitoring Event Report AVPs.

Note: The HSS may divide the accumulated Monitoring Configuration Indications/immediate reports into multiple messages.

The HSS sends immediate reports and configuration indications for the group based configuration processing using the Group-Monitoring-Event-Report.

When the SCEF receives a RIR from the HSS, and at least one of the received Monitoring Event Reports has a SCEF-Reference-ID not known by the SCEF, it shall reply with DIAMETER_ERROR_SCEF_REFERENCE_ID_UNKNOWN. In that case, if the HSS had included multiple Monitoring Event Reports in the RIR command, the SCEF includes in the Reporting Information Answer command a list of Monitoring-Event-Report-Status AVPs where the status of multiple monitoring event reports is detailed. In that AVP list, the AVPs corresponding to event reports with a successful status may be omitted by the SCEF for efficiency.

SCEF compares the Monitoring type, User Identifier, and its value received in message with the context. If there is any mismatch, it replies with DIAMETER_ERROR_SCEF_REFERENCE_ID_UNKNOWN; otherwise, when the SCEF receives a RIR from the HSS, the SCEF sets the Experimental-Result to DIAMETER_SUCCESS in the Reporting Information Answer and handles it according to the procedures defined in 3GPP TS 23.682 Architecture enhancements to facilitate communications with packet data networks and applications.

For each successful report data in Group-Monitoring-Event-Report and the Monitoring Event Report AVPs, SCEF sends an HTTP post notification message to SCS/AS with details as received in RIR message.

Notification in Reporting-Information-Request (RIR) from MME/SGSN

When the SCEF receives a Reporting Information Request from the MME/SGSN and at least one of the Monitoring Event Report AVPs has a SCEF-Reference-ID not known by the SCEF, it replies with Experimental-Result-Code set to DIAMETER_ERROR_SCEF_REFERENCE_ID_UNKNOWN. In that case, if the HSS had included multiple Monitoring Event Reports in the RIR command, the SCEF includes in the Reporting Information Answer command a list of Monitoring-Event-Report-Status AVPs where the status of multiple monitoring event reports is detailed. In that AVP list, the AVPs corresponding to event reports with a successful status may be omitted by the SCEF, for efficiency; otherwise, when the SCEF receives a Reporting-Information-Request command from the MME/SGSN, the SCEF sets Result-Code to DIAMETER_SUCCESS in the Reporting-Information-Answer and handles it according to the procedures defined in 3GPP TS 23.682 Architecture enhancements to facilitate communications with packet data networks and applications.

SCEF compares the Monitoring type, User Identifier, and its value received in message with the context. If there is any mismatch, it replies with DIAMETER_ERROR_SCEF_REFERENCE_ID_UNKNOWN.

For each successful report data in the Monitoring Event Report AVPs, SCEF sends an HTTP post notification message to SCS/AS with details as received in RIR message.

Notification in Configuration-Information-Answer (CIA)

This procedure is used between the HSS and the SCEF. HSS can send an available report for the Monitoring Event for the subscription done in the Monitoring Event Report AVPs in the Configuration-Information-Answer (CIA) message itself.

For each successful report data in the Monitoring Event Report AVPs, SCEF sends an HTTP post notification message to SCS/AS with details as received in the RIR message.

Monitoring Event Deletion Initiated from SCS/AS

SCS/AS can send HTTP message using Individual Monitoring Event Subscription and DELETE method. SCS/AS includes the TLTRI in URI, which need to be deleted.

SCEF finds and deletes the context for Monitoring Event Subscription corresponding to SCS/AS and TLTRI received in HTTP message.

SCEF also sends Configuration-Information-Request (CIR) for deletion for SCEF Reference ID corresponding to SCS/AS and TLTRI received in HTTP message.

Monitoring Event Deletion Initiated from HSS

When a subscriber is deleted from the HSS while monitoring is active or the authorization for monitoring is revoked, the HSS sends an RIR command to the SCEF with the Event-Handling AVP set to the value CANCEL.

SCEF finds and deletes the context for Monitoring Event Subscription corresponding to SCEF Reference ID received in RIR message from HSS.

Monitoring Event Get

SCS/AS can send an HTTP message using the Individual Monitoring Event Subscription and GET method. SCS/AS includes the TLTRI in URI, which needs to be fetched.

SCEF finds and gets back the context data stored for the Monitoring Event Subscription corresponding to SCS/AS and TLTRI received in the HTTP message.

Enhanced Coverage Restriction Control

Enhanced Coverage Restriction Control support using SCEF enables 3rd party service providers to query status of, enhanced coverage restriction, or enable/disable enhanced coverage restriction per individual UEs.

The Enhanced Coverage Restriction Control procedure requested by SCS/AS is determined from the URI as described below.

Table 2-6 Supported Enhanced Coverage Restriction Control Resources and Methods

Resource Name	Resource URI	HTTP Method(s)	HTTP Initiator
Query	3gpp_ecr_control/v1 /{scsAsId}/query	POST	SCS/AS
Configure	3gpp_ecr_control/v1 /{scsAsId}/configure	POST	SCS/AS

1. The SCS/AS sends an Enhanced Coverage Request (External Identifier or the MSISDN, SCS/AS Identifier, Request Type, and Enhanced Coverage Restriction Data) message to SCEF. Request Type indicates if the request needs to query the status, enable, or disable the enhanced coverage restriction. Enhanced Coverage Restriction Data provides data related to the Enhanced Coverage Restriction. Enhanced Coverage Restriction Data is only present if the Request Type enables or disables the enhanced coverage restriction.

2. Based on operator policies, if the SCS/AS is not authorized to perform this request, or the Enhanced Coverage Request is malformed, or the SCS/AS has exceeded its quota or rate of submitting Enhanced Coverage requests, SCEF performs step 9 and provides a Cause value appropriately indicating the failure result.

3. SCEF sends an Enhanced Coverage Request (External Identifier or MSISDN Type) message to the HSS.

4. HSS examines the Enhanced Coverage Request message for the existence of an External Identifier or MSISDN, any included parameters in the acceptable range

for the operator, and the Enhanced Coverage restriction by the serving MME/SGSN. If this check fails, the HSS follows step 8 and provides a Cause value indicating the reason for the failure condition to the SCEF.

If the Request Type is to get the current status of enhanced coverage, HSS retrieves the value and follows the procedure at step 8; otherwise, if the Type is to enable or to disable the enhanced coverage, HSS sets the Enhanced Coverage Restricted parameter to the appropriate value and the procedure continues with step 5.

5. If required by the specific Enhanced Coverage Request Type and when Enhanced Coverage is supported by the serving MME/SGSN, HSS sends an Insert Subscriber Data Request (Type) message to the MME/SGSN.

6. Based on operator policies, MME/SGSN may reject the request (for example, for an overload or HSS has exceeded its quota or rate of submitting enhanced coverage requests defined by an SLA).

   The MME/SGSN updates Enhanced Coverage Restricted parameters in the MME/SGSN context.

   The MME/SGSN transfers the Enhanced Coverage Restricted parameters stored as part of its context information during the MME/SGSN change.

   Note: UE is informed of the updated Enhanced Coverage Restricted parameters value at the next TAU/RAU or, based on the local policy, the network can detach the UE indicating re-attach is required.

7. If the Enhanced Coverage restriction is updated successfully, the MME/SGSN sends an Insert Subscriber Data Answer (Cause) message to HSS. MME/SGSN may include the Enhanced Coverage Restricted parameter in the Insert Subscriber Data Answer message.

8. HSS sends an Enhanced Coverage Response (Cause) message to SCEF. HSS includes result = success/failure and in case of success may include Enhanced Coverage Restriction Data.

9. SCEF sends an Enhanced Coverage Response (Cause, Enhanced Coverage Restriction Data) message to the SCS/AS. Cause indicates success or failure. If, in step 1, the Enhanced Coverage Request message is sent to query the status of Enhanced Coverage Restricted, then the Enhanced Coverage Restriction Data is included (in case of success) in the Enhanced Coverage Response message.

Device Triggering

The Device Triggering feature allows the SCS/AS to deliver a specific device trigger to the UE through SCEF. The Device Trigger request is authenticated with HSS using the User Identifier received in request. After successful authentication SCEF forwards the Device Trigger Request to the corresponding SMS-SC to be delivered to the UE.

The Device Triggering procedure requested by SCS/AS is determined from the URI as described in Table 2-7.

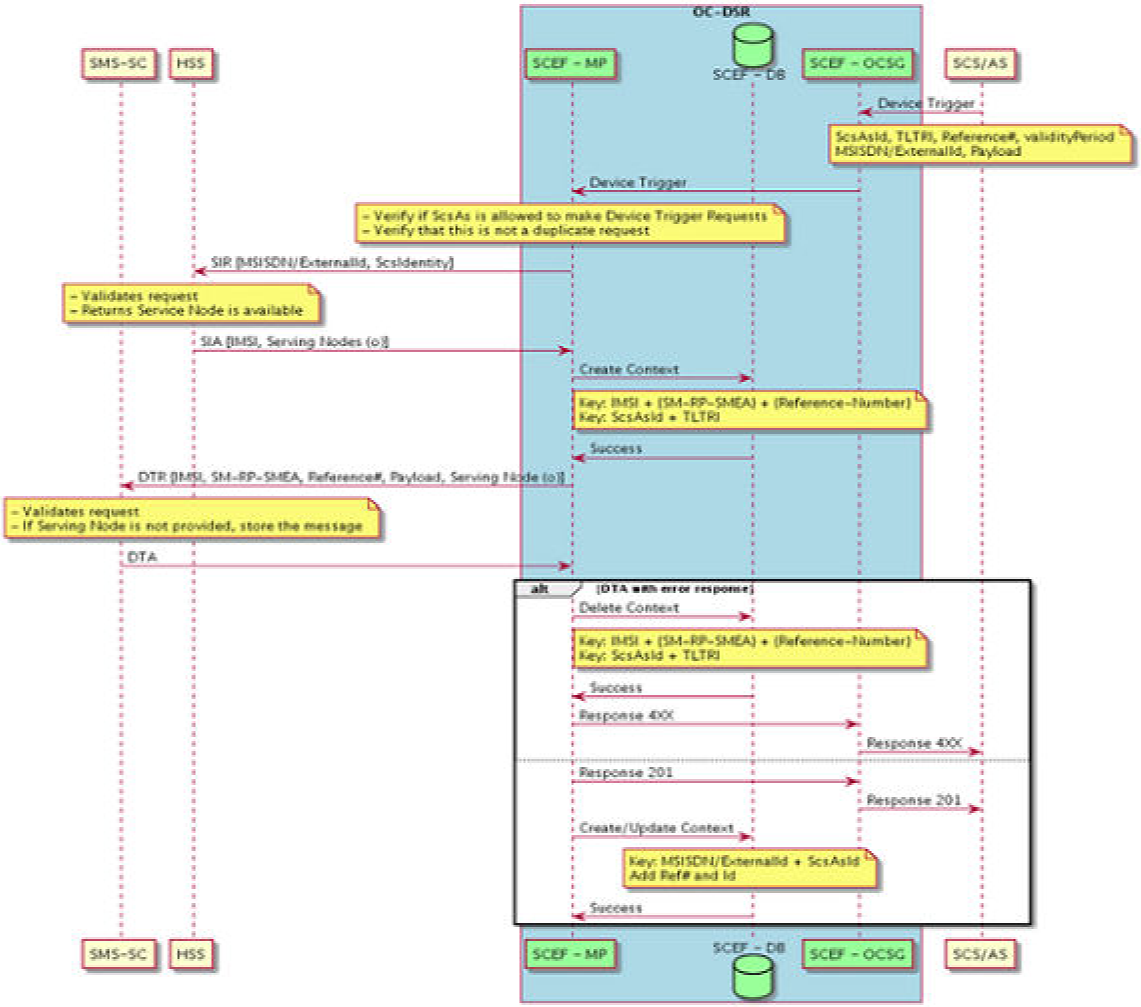
Table 2-7 Supported Device Triggering Resources and Methods

Resource Name	Resource URI	HTTP Method(s)	HTTP Initiator
Device Triggering Transactions	3gpp_t8_device_triggering/v1/{scsAsId}/transactions	POST	SCS/AS
Individual Device Triggering Transaction	3gpp_t8_device_triggering/v1/{scsAsId}/transactions/{tltrId}	GET	SCS/AS
Device Triggering Delivery Report Notification	{notification_uri}	POST	SCEF

Device Triggering Transaction

Figure 2-9 illustrates the procedure of creating a Device Trigger Transaction at the SCEF and SMS-SC.

Figure 2-9 Device Triggering Transaction Creation


OC-DSR
SMS-SC
HSS
SCEF - MP
SCEF - DB
SCEF - OCSG
SCS/AS
Device Trigger
ScsAsId, TLTRI, Reference#, validityPeriod
MSISDN/ExternalId, Payload
Device Trigger
- Verify if ScsAs is allowed to make Device Trigger Requests
- Verify that this is not a duplicate request
SIR [MSISDN/ExternalId, ScsIdentity]
- Validates request
- Returns Service Node is available
SIA [IMSI, Serving Nodes (o)]
Create Context
Key: IMSI + (SM-RP-SMEA) + (Reference-Number)
Key: ScsAsId + TLTRI
Success
DTR [IMSI, SM-RP-SMEA, Reference#, Payload, Serving Node (o)]
- Validates request
- If Serving Node is not provided, store the message
DTA
alt
[DTA with error response]
Delete Context
Key: IMSI + (SM-RP-SMEA) + (Reference-Number)
Key: ScsAsId + TLTRI
Success
Response 4XX
Response 4XX
Response 201
Response 201
Create/Update Context
Key: MSISDN/ExternalId + ScsAsId
Add Ref# and Id
Success
SMS-SC
HSS
SCEF - MP
SCEF - DB
SCEF - OCSG
SCS/AS

The SCS/AS sends a Device Triggering Transaction Request (External Identifier or MSISDN, SCS/AS Identifier, Trigger Reference Number, Payload, Validity Period, Destination Address) message to the SCEF.

> **Note:** SCEF always gives higher preference to the External Identifier when both Identifiers (External Identifier and MSISDN) are present in the Device Triggering Transaction Request message.

DSR SCEF stores the External Identifier or MSISDN, SCS/AS Identifier, Destination Address, and Validity Period. If the SCS/AS is not authorized to perform this request (for example, based on Access Control policies as described in Access Control, if the SLA does not allow for it), or the Device Triggering Transaction Request is malformed, the SCEF responds appropriately indicating the error.

The SCEF sends a Subscriber Information Request (External Identifier, MSISDN, APN) message to the HSS to authorize the Device Triggering request for the received External Identifier or MSISDN, and to receive other information like IMSI, serving entities of the user, which are necessary for Device Triggering request processing.

The HSS examines the Subscriber Information Request message regarding the existence of the External Identifier or MSISDN and maps the external identifier to IMSI and/or MSISDN. If this check fails, the HSS provides a result indicating the reason for the failure condition to the SCEF.

The HSS sends a Subscriber Information Response (IMSI and MSISDN; or External Identifier, Serving Nodes, and Result) message to the SCEF to Authorize Device Triggering Request. The IMSI and, if available, the MSISDN (when Device Triggering Transaction Request contains an External Identifier) or if available, the External Identifier(s) (when Device Triggering Transaction Request contains an MSISDN) are returned by the HSS in this message.

SCEF sends a Device Trigger Request (IMSI, SME-Address, Reference Number, Payload, Validity Time, Serving Node) message to the SMS-SC to transfer the Device Trigger received from SCS/AS and identities entities serving the user. The SCEF caps the Validity Period specified by the SCS/AS at a value configured at SCEF (in the Device Triggering Configuration Set Managed Object) before sending it to SMS-SC.

The SMSC validates the identity of the user, SME-Address, and the routing information of serving entities (if available), and checks for congestion in the system. If these checks fail, then SMS-SC sends a response with result indicating the reason for failure.

The SMS-SC sends a Device Trigger Answer (Result) message to SCEF with success result if the Device Triggering Request is accepted.

The SCEF sends a Device Triggering Transaction Response message to the SCS/AS to acknowledge acceptance of the Device Triggering Transaction Request.

Transaction Query by SCS/AS

The SCS/AS may request for the Device Triggering Transaction data that is saved with SCEF using a Device Triggering Transaction GET API. SCEF looks for the SCS/AS Identifier and the TLTRI provided in the request and if found, includes the following parameters stored in the SCEF's database in the response.

- User Identity (External Identifier or MSISDN)
- TLTRI

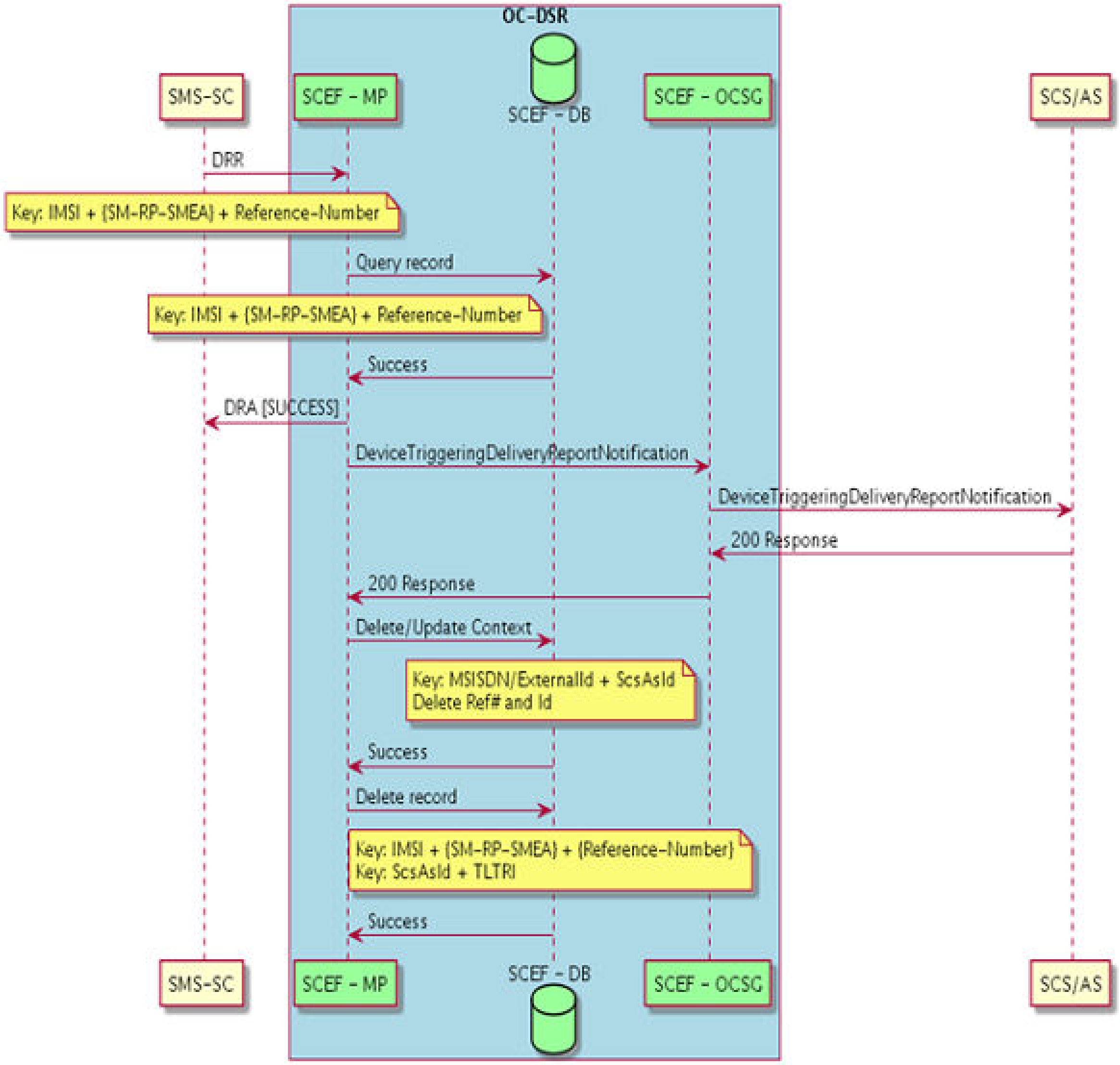
- Result

If the TLTRI requested by the SCS/AS is not found in the SCEF's database, SCEF responds with the 404 Not Found error.

Device Triggering Delivery Report Notification

Figure 2-10 illustrates the procedure of sending Device Triggering Delivery Report Notification to SCS/AS.

Figure 2-10 Device Triggering Delivery Report Notification


OC-DSR
SMS-SC
SCEF - MP
SCEF - DB
SCEF - OCSG
SCS/AS
DRR
Key: IMSI + (SM-RP-SMEA) + Reference-Number
Query record
Key: IMSI + (SM-RP-SMEA) + Reference-Number
Success
DRA [SUCCESS]
DeviceTriggeringDeliveryReportNotification
DeviceTriggeringDeliveryReportNotification
200 Response
200 Response
Delete/Update Context
Key: MSISDN/ExternalId + ScsAsId
Delete Ref# and Id
Success
Delete record
Key: IMSI + (SM-RP-SMEA) + (Reference-Number)
Key: ScsAsId + TLTRI
Success
SMS-SC
SCEF - MP
SCEF - DB
SCEF - OCSG
SCS/AS

SMS-SC sends the Device Report Request to report the success or failure of delivering the device trigger to the UE to the SCEF. SCEF verifies the context for this Device Trigger exists and sends the notification to SCS/AS with an appropriate delivery result. SCEF uses the Notification Destination Address provided by SCS/AS at the time of Device Triggering Transaction, if provided, or uses the configuration in the SCS/AS managed object.

Access Control

The SCEF application provides support for multi-tenancy of SCS. This is achieved by Access Control Logic (ACL).

ACL ensures UE (IOT devices) belonging to one SCS are not accessed by another SCS.

ACL performs this functionality on HTTP requests:

- Validate SCS: If SCS is not pre-configured in SCEF, it returns a 401 Unauthorized error. If SCS is configured, but the feature (requested in message) is not enabled for SCS, then it returns a 401 Unauthorized error or it displays “Validate SCS accessibility to UE.”
- Validate SCS accessibility to UE: Extract UE-Identifier from message and validate if SCS is allowed to access the UE for the specific requested feature, if not, then it returns a 401 Unauthorized error or it allows the message for further processing.

3

Managed Objects

SCEF works with Common (including SCS/AS and System Options), AppWorks, and NIDD (including NIDD and APN Configuration) managed objects described in this chapter.

SCS/AS

The SCS/AS managed objects exist for each SCS/AS that needs to communicate with DSR's SCEF application. This managed object allows the customer to configure an SCS/AS by specifying its properties and associate an APN to it. Attributes listed in SCS/AS are used to configure the SCS/AS managed object.

Table 3-1 SCS/AS Attribute Descriptions

Attribute	Description
scsAsId	The SCS/AS identifier.
niddCfgSetName	The NIDD Configuration Set managed object associated to this SCS/AS. When this attribute is populated, the NIDD feature is enabled.
apnCfgSetName	The APN Configuration Set managed object associated to this SCS/AS. This attribute must be populated if the niddConfigSetName is populated.
monitoringEventCfgSetName	The Monitoring Events Configuration Set managed object associated to this SCS/AS.
deviceTriggeringCfgSetName	The Device Triggering Configuration Set managed object associated to this SCS/AS.
aclId	Associated AclId with SCS.
callbackUrl	Destination URL for any notification messages for this SCS/AS.
smsScFqdn	FQDN of SMS-SC.
smsScRealm	Realm of SMS-SC.
scsAsIsdn	ISDN number of the SCS/AS in international ISDN number format.
isEcrAllowed	Value of this attribute decides if Enhanced Coverage Restriction Control is allowed or not.

System Options

The System Options managed objects allow the customer to specify routing configurations and system defaults that apply to a DSR node. Attributes listed Table 3-2in are used to configure the System Options managed object.

Table 3-2 System Options Attribute Descriptions

Attribute	Description
art	Application Routing Table instance used to route any Diameter request messages generated by the SCEF application.
prt	Peer Routing Table instance used to route any Diameter request messages generated by the SCEF application.
apiGwIpList	A comma separated list of IPv4 addresses of the DSR API Gateway application server. The SCEF application distributes the HTTP request messages toward the DSR API Gateway among the IP addresses listed in the *apiGwIpList* attribute.
priority	DRMP priority of NIR, ACR, and CMR messages originated by SCEF.
retryDbUpdate	The number of times the SCEF MP server may retry when an attempt to update a context in the USBR server fails due to concurrent update checksum mismatch.
servingPlmnRateControlEnabled	This option allows the customer to enable or disable Mobile Terminating PDU rate control based on the Serving PLMN Rate Control configuration requested by MME/SGSN.
scefWaitTime	The duration of time in seconds the SCEF application may wait for a Diameter Answer message for any request sent to the MME/ SGSN. The *scefWaitTime* attribute should match the Pending Answer Timeout value in the Diameter configuration.
binaryEncoder	The Binary-To-Text encoding scheme to use to transcode binary data while sending to or receiving from the SCS/AS in JSON-encoded HTTP message. Allowed values are: • Base2 • Base16 • Base64 • ASCII

AppWorks

The AppWorks managed objects (Server Groups, Resource Domains, Places, and Place Associations) require the following changes described in this section.

- All DA-MP servers that running the SCEF application need to be configured in a Place Association of type *Application Region*.
- All DA-MP servers that running the SCEF application need to be configured in a Resource Domain with a *Application MPs* profile.

NIDD Configuration Set

An NIDD Configuration Set managed object exists for each instance of an NIDD Configuration Set. This managed object allows the customer to create instances of NIDD Configuration Sets as needed. The NIDD Configuration Set specifies various attributes that controls the processing of NIDD-related messages, and safeguards the operator from unacceptable request parameters, for example, an authorization duration that is too long.

Table 3-3 Non-IP Data Delivery Attribute Descriptions

Attribute	Description
name	A name that uniquely defines the NIDD Configuration Set.
maxAuthDuration	Maximum time in seconds the NIDD configuration is valid.
pdnEstablishmentOptionEnabled	Applicability of the PDN Establishment Option IE received in the NIDD Configuration message and/or Downlink Data Delivery messages received for this NIDD configuration.
pdnEstablishmentOption	Default PDN Establishment Option to apply if the NIDD Configuration message and any subsequent Downlink Data Delivery message(s) do not contain the PDN Establishment Option IE. **Note:** This attribute is applicable only if the *pdnEstablishmentOptionEnabled* is set to true. Any change made to the *pdnEstablishmentOptionEnabled* attribute takes effect when the next Downlink Data message is received from the SCS/AS.
dataDuration	The maximum time in seconds a Downlink Data Delivery message is considered to be valid when it is buffered by the SCEF application. If *dataDuration* is set to 0, then SCEF does not buffer NIDD MT data messages irrespective of *maxOnholdDataMsg* configuration and maximum message latency value received in the MT NIDD submit request from SCS/AS.
maxOnholdDataMsg	The maximum number of messages for each UE that can be buffered by SCEF application.

Table 3-3 (Cont.) Non-IP Data Delivery Attribute Descriptions

Attribute	Description
priority	The default priority associated to a Downlink Data Delivery message when the same is not present in the message received from the SCS/AS.
minRetransmissionTime	The minimum time in seconds the SCEF application requires to buffer a Downlink Data Delivery message in the USBR database and then retransmit it to the MME/SGSN. **Note:** This attribute becomes more significant in slow WAN networks.

APN Configuration

An APN Configuration Set managed object instance exists for each Access Point Name that the customer servers by SCEF signaling. An APN Configuration Set managed object is associated to an SCS/AS. This managed object specifies various attributes that controls the processing of signaling messages that belong to that APN, for example the rate control attributes.

Table 3-4 Access Point Name Attribute Descriptions

Attribute	Description
Name	Access Point Name.
maxPacketSize	Maximum Packet Size (Uplink or Downlink) in bytes that is allowed to be transmitted through the SCEF application.
maxPacketBufferSize	Maximum Packet Size (Downlink) in bytes that is allowed to be buffered by the SCEF application.
downlinkApnRateControlUnit	The unit for Downlink APN rate control. Any change made to the *downlinkApnRateControlUnit* attribute takes effect when the next Downlink Data message is received from the SCS/AS.
downlinkApnRateControlVal	Multiple of Downlink APN rate control unit. Any change made to the *downlinkApnRateControlVal* attribute takes effect when the next Downlink Data message is received from the SCS/AS.

Table 3-4 (Cont.) Access Point Name Attribute Descriptions

Attribute	Description
downlinkApnMessageRate	The maximum Downlink message rate for this APN. Any change made to the *downlinkApnMessageRate* attribute takes effect when the next Downlink Data message is received from the SCS/AS.
uplinkApnRateControlUnit	The unit for Uplink APN rate control. Any change made to the *uplinkApnRateControlUnit* attribute takes effect when the next Downlink Data message is received from the SCS/AS.
uplinkApnRateControlVal	Multiple of Uplink APN rate control unit. Any change made to the *uplinkApnRateControlVal* attribute takes effect when the next Downlink Data message is received from the SCS/AS.
uplinkApnMessageRate	The maximum Uplink message rate for this APN. Any change made to the *uplinkApnMessageRate* attribute takes effect when the next Downlink Data message is received from the SCS/AS.

4

Configure SCEF

All configurations and status reporting for the SCEF application is performed using machine-to-machine interfaces. To access the MMI API documentation through a direct URL access, without login, go to http://(IP address of NOAM or SOAM)/raml/mmi.html. Or the MMI API documentation can be accessed directly from the DSR GUI by clicking on the new MMI API Guide menu item.

The SCEF application is bundled with DSR. Once you have activated SCEF and start using it, SCEF changes, described in this section, are seen in the DSR GUI.

Alarms, Events, and KPI Changes

Alarms, Events, and KPIs have been added to the Alarms and KPI Reference Manual available on OHC at https://docs.oracle.com/en/industries/communications/diameter-signaling-router/index.html.

Measurement Changes

Measurements have been added to the Measurements Reference Manual available on OHC at https://docs.oracle.com/en/industries/communications/diameter-signaling-router/index.html.

MMI Changes

MMI changes are described in the SCEF MMI Attributes chapter of this manual.

Basic SCEF Configuration

Follow these steps first before beginning the SCEF MMI configuration.

1. Configure the Resource, Resource Domain, Place, and Place Association for SCEF.

   Refer to the Session Binding Repository (SBR) User's Guide on OHC.

2. Configure the Local Node, Peer Nodes, Connections, PRT, and ART (table and rules).

   Refer to the Diameter User's Guide on OHC.

3. (Optional) If Application Chaining is intended for Diameter routing:

   a. Configure ART (Table and Rules).

      Refer to the Diameter User's Guide on OHC.

   b. Configure RBAR/FABR.

      Refer to the Range Based Address Resolution User's Guide and Full Address Based Resolution User's Guide on OHC.

SCEF MMI Configuration

Once the basic configuration is complete, make these changes in MMI on the SO. The MMI API documentation can be found on OHC.

1. Configure ACL by configuring these managed objects:

    a. Access Control List

    An Access Control List (ACL) configuration entry consists of a name. The purpose of the ACL is to maintain a set of Access Control Rules that can be associated to one or more SCS/AS.

    ```
    Send a POST message to an active SOAM with a URL  scef/
    accesscontrollists  and content as specified in accesscontrollist.json
    ```

    For example,

    ```
    scef/accesscontrollists -v POST –r accesscontrollist.json
    ```

    b. Access Control Rules

    This MO maintains all rules for access control.

    ```
    Send a POST message to an active SOAM with a URL  scef/
    accesscontrolrules  and content as specified in accesscontrolrule_DN.json
    Send a POST message to an active SOAM with a URL  scef/
    accesscontrolrules  and content as specified in
    accesscontrolrule_Domain.json
    ```

    For example,

    ```
    scef/accesscontrolrules -v POST –r accesscontrolrule_DN.json
    scef/accesscontrolrules -v POST –r accesscontrolrule_Domain.json
    ```

    c. Access Control Associations

    This MO creates/maintains the association between the List and Rules. This is used to tell which rules are under a specific List.

    ```
    Send a POST message to an active SOAM with a URL  scef/
    accesscontrolassociations  and content as specified in
    accesscontrolassociation.json
    ```

    For example,

    ```
    scef/accesscontrolassociations -v POST –r accesscontrolassociation.json
    ```

2. Configure APN with the name and configuration-specific data for APN.

    ```
    scef/apnconfigurationsets -v POST –r apnconfigurationset.json
    ```

3. Configure the Options managed object with system configuration parameters for SCEF.

    ```
    scef/options -v POST –r options.json
    ```

4. Configure the NIDD configuration set.

    ```
    scef/niddconfigurationsets -v POST –r niddconfigurationset.json
    ```

5. Configure the Monitoringevent configuration set.

```
scef/monitoringeventconfigurationsets -v POST –r
monitoringeventconfigurationset.json
```

6. Configure the Device Triggering configuration set.

```
scef/devicetriggeringconfigurationsets -v POST –r
devicetriggeringconfigurationset.json
```

7. Configure SCS/AS to specify the SCSid and its associated configuration. Use JSON based on the features that need to be enabled.

```
/scef/scsapplicationservers -v POST –r  scsapplicationserver.json
```

8. Create JSON files MMI will use.

Example 4-1 scsapplicationserver.json

scsapplicationserver.json with NIDD enabled.

```
{
   "apnCfgSetName": "apn1.test.com",
   "callbackUrl": "https://test.xyz.com",
   "chargingEnabled": false,
   "interimInterval": 600,
   "niddCfgSetName": "NIDDCFG1",
   "scsAsId": "SCSAS1"
}
```

scsapplicationserver.json with MONITORING enabled.

```
{
   "apnCfgSetName": "apn1.test.com",
   "callbackUrl": "https://test.xyz.com",
   "chargingEnabled": false,
   "interimInterval": 600,
   "monitoringEventCfgSetName": "MONEVENT1",
   "scsAsId": "SCSAS1"
}
```

scsapplicationserver.json with ECR enabled.

```
{
     "callbackUrl": "https://test.xyz.com",
   “isEcrAllowed”:true,
   "scsAsId": "SCSAS1"
}
```

scsapplicationserver.json with DT enabled.

```
{
   "apnCfgSetName": "apn1.test.com",
   "callbackUrl": "https://test.xyz.com",
   "chargingEnabled": false,
   "interimInterval": 600,
   "deviceTriggeringCfgSetName": "DevTrig1",
   "scsAsId": "SCSAS1",
   "scsAsIsdn":"123456789",
   "smsScFqdn": "test.one",
   "smsScRealm":"oracle.com"
}
```

scsapplicationserver.json with ACL, NIDD, MON, and DT enabled.

```
{
   "apnCfgSetName": "apn1.test.com",
   "callbackUrl": "https://test.xyz.com",
   "aclName":"ACL1",
   "chargingEnabled": false,
   "interimInterval": 600,
   "niddCfgSetName": "NIDDCFG1",
   "monitoringEventCfgSetName": "MONEVENT1",
   "deviceTriggeringCfgSetName": "DevTrig1",
   "scsAsId": "SCSAS1",
   "scsAsIsdn":"123456789",
   "smsScFqdn": "test.one",
   "smsScRealm":"oracle.com"
}
```

Example 4-2   accesscontrolrule_DN.json

This creates a rule to allow NIDD and Monitoring for DOMAIN = test.oracle.com.

```
{
   "name": "RULE2",
   "domain": "test.oracle.com",
   "supportedFeatures": [
          "Monitoring",
          "Nidd"
   ],
   "userIdentifierType": "DOMAIN"
}
```

Example 4-3   accesscontrolassociation.json

Associates RULE1 with ACL1

```
{
   "aclName": "ACL1",
   "ruleName": "RULE1"
}
```

Example 4-4   apnconfigurationset.json

This managed object controls the APN level rate control parameters.

```
{
   "downlinkApnMessageRate": 300,
   "downlinkApnRateControlUnit": "Day",
   "downlinkApnRateControlVal": 3,
   "maxPacketBufferSize": 100,
   "maxPacketSize": 100,
   "name": "apn1.test.com",
   "value": "apn1.test.com",
   "uplinkApnMessageRate": 600,
   "uplinkApnRateControlUnit": "Week",
   "uplinkApnRateControlVal": 6
}
```

Example 4-5   options.json

```
{
   "apiGwIpList": [
       "20.20.20.2",
       "20.20.20.4"
   ],
   "art": "ART1",
   "prt": "PRT1",
   "scefId": "oracle.com",
   "scefWaitTime": 1200,
   "servingPlmnRateControlEnabled": true
}
```

Example 4-6 monitoringeventconfigurationset.json

```
{
   "monitoringType": [
          "UEReachability",
          "LocationReporting"
   ],
   "name": "MONEVENT1"
}
```

Example 4-7 devicetriggeringconfigurationset.json

```
{
   "defaultApplicationPort": 1000,
   "defaultPriority": "NonPriority",
   "maxValidityPeriod": 3600,
   "name": "DevTrig1",
   "mandateApplicationPort": false,
   "mandatePriority": false
}
```

Example 4-8 niddconfigurationset.json

```
{
   "dataDuration": 20,
   "maxAuthDuration": 86400,
   "maxOnholdDataMsg": 1,
   "minRetransmissionTime": 6,
   "name": "NIDDCFG1",
   "pdnEstablishmentOption": "INDICATE_ERROR",
   "pdnEstablishmentOptionEnabled": false
}
```

5

SCEF MMI Attributes

Access Control Associations

The Access Control Associations are used to associate the Rules with an Access Control List.

Table 5-1 Access Control Associations Attribute Details

Attribute	Type	Mandatory (Yes/No)
aclName	string	Yes
ruleName	string	Yes

Access Control Lists

The Access Control Lists are a set of configurations to support multi tenancy Access Control, DRMP, and Flow/Overload Control during notification callback.

Table 5-2 Access Control Lists Attribute Details

Attribute	Type	Mandatory (Yes/No)
name	string	Yes

Access Control Rules

Access Control Rules are a set of configurations to create an Access Control Rule.

Table 5-3 Access Control Rules Attribute Details

Attribute	Type	Enum Values	Mandatory (Yes/No)
name	string		Yes
userIdentifierType	string	MSISDN DOMAIN	Yes
startAddr	string		No
endAddr	string		No
domain	string		No

Table 5-3 (Cont.) Access Control Rules Attribute Details

Attribute	Type	Enum Values	Mandatory (Yes/No)
supportedFeatures[1]	string	Nidd Monitoring Triggering ECR	No

[1] Multiple values can be specified.

APN Configuration Sets

APN Configuration Sets are a set of configurations associated to an APN; the APN is associated to a SCS/AS Profile.

Table 5-4 APN Configuration Sets Attribute Details

Attribute	Type	Min/Max	Enum Values	Default Value	Mandatory (Yes/No)
name	string				Yes
value	string				Yes
maxPacketSize	integer	1/2500		100	No
maxPacketBufferSize	integer	1/500		100	No
downlinkApnRateControlUnit	string		Unrestricted Minute Hour Day Week	Unrestricted	No
downlinkApnRateControlVal[1]	integer	0/60			No
downlinkApnMessageRate[1]	integer	1/1000			No
uplinkApnRateControlUnit	string		Unrestricted Minute Hour Day Week	Unrestricted	No
uplinkApnRateControlVal[1]	integer	0/60			No
uplinkApnMessageRate[1]	integer	1/1000			No

[1] Has an unconfigured value of -1.

Device Triggering Configuration Sets

Device Triggering Configuration Sets are a set of configurations for Device Triggering to create a SCS/AS.

Table 5-5 Device Triggering Configuration Sets Attribute Details

Attribute	Type	Min/Max	Enum Values	Default Value	Mandatory (Yes/No)
name	string				Yes
defaultApplicationPort	integer	0/65535		1000	No
defaultPriority	string		NonPriority Priority	NonPriority	No
maxValidityPeriod	integer	-1/86400		3600	No
mandateApplicationPort	boolean		true false	false	No
mandatePriority	boolean		true false	false	No

Monitoring Event Configuration Sets

Monitoring Event Configuration Sets are a set of configurations for SCEF Monitoring Events.

Table 5-6 Monitoring Event Configuration Sets Attribute Details

Attribute	Type	Min/Max	Enum Values	Default Value	Mandatory (Yes/No)
name	string				Yes
monitoringType[2]	string		LossOfConnectivity UEReachability LocationReporting ChangeOfImsiImeiAssociation RoamingStatus		Yes
groupedMonitoringEnabled	boolean		true false	false	No
numberOfReports	integer	1/5000		1	No
maxMonitoringDuration	integer	1/86400		3600	No
maxUePerReport	integer	1/100		50	No

Table 5-6 (Cont.) Monitoring Event Configuration Sets Attribute Details

Attribute	Type	Min/Max	Enum Values	Default Value	Mandatory (Yes/No)
initiateRefDelete	boolean		true false	false	No
minTauRauTimerValue[1]	integer	0/2147483647			No
maxTauRauTimerValue[1]	integer	0/2147483647			No
minResponseTime[1]	integer	0/2147483647			No
maxResponseTime[1]	integer	0/2147483647			No
downlinkPackets	integer	1/100			No
enforceReportingInterval	boolean		true false	false	No
minReportingInterval[1]	integer	1/86400		120	No
locationAccuracy[3]	string		CgiEcgi eNB LaTaRa PLMN	eNB	No

[1] Has an unconfigured value of -1.

[2] User can select multiple values. Bit values are shown in Table 5-7.

[3] PLMN - Future Plan

Table 5-7 Bit Values

Enum	Bit Value
LossOfConnectivity	1
UEReachability	2
LocationReporting	4
ChangeOfImsiImeiAssociation	8
RoamingStatus	16
CommunicationFailure	32
AvailabilityAfterDdnFailure	64
NumberOfUEsInAnArea	128
EnhancedCoverage	256

NIDD Configuration Sets

NIDD Configuration Sets are a set of configurations that apply to NIDD call flows.

Table 5-8 NIDD Configuration Sets Attribute Details

Attribute	Type	Min/Max	Enum Values	Default Value	Mandatory (Yes/No)
name	string				Yes
dataDuration	integer	0/8600		0	No
maxAuthDuration	integer	60/604800		86400	No
maxOnholdDataMsg	integer	1/5		1	No
minRetransmissionTime	integer	0/10		5	No
pdnEstablishmentOptionEnabled	boolean		true false	false	No
pdnEstablishmentOption	string		WAIT_FOR_UE INDICATE_ERROR	INDICATE_ERROR	No
priority	integer	0/15		0	No

SCS/AS

SCS/AS enables applications to access and use functionality provided by Service Components over standardized interfaces (APIs).

Table 5-9 SCS/AS Attribute Details

Attribute	Type	Min/Max	Enum Values	Default Value	Mandatory (Yes/No)
scsAsId	string				Yes
niddCfgSetName	string				No
apnCfgSetName	string				No
monitoringEventCfgSetName	string				No
deviceTriggeringCfgSetName	string				No
callbackUrl	string				No
chargingEnabled	boolean		true false	false	No

Table 5-9 (Cont.) SCS/AS Attribute Details

Attribute	Type	Min/Max	Enum Values	Default Value	Mandatory (Yes/No)
isEcrAllowed	boolean		true false	false	No
interimInterval	integer	10/3600		600	No
chargingFeatureList	string		Nidd Monitoring Triggering		No
smsScFqdn[1]	string				No
smsScRealm[1]	string				No
scsAsIsdn[1]	string				No
aclName[2]	string				No

[1] Applicable only when Device Triggering is configured .

[2] Related to Access Control

System Options

SCEF related user configurable Options.

Table 5-10 System Options Attribute Details

Attribute	Type	Min/Max	Enum Values	Default Value	Mandatory (Yes/No)
art	string				No
prt	string				No
chargingFqdn	string				No
chargingRealm	string				No
apiGwIpList	string				Yes
priority	integer	0/15		0	No
retryDbUpdate	integer	0/5		2	No
servingPlmnRateControlEnabled	boolean		true false	false	No
longestSubdomainMatchEnabled	boolean		true false	false	No
scefWaitTime	integer	1/180		1	No
scefId[1]	string				No

Table 5-10 (Cont.) System Options Attribute Details

Attribute	Type	Min/Max	Enum Values	Default Value	Mandatory (Yes/No)
binaryEncoder	string		ASCII Base2 Base16 Base64	Base64	No

[1] This maps to the FQDN of Local Node

A

OCSG Introduction

This appendix describes how to configure Oracle Communications Services Gatekeeper (OCSG) and then provision it.

Custom Configuration

This section describes how to configure the OCSG.

Custom configuration of the OCSG involves these steps:

- Configure DSR MP IPs in DSR API GW
- Add SNMP Trap Receiver
- Change SNMP Version
- Generate MIB File
- Change General Logging Level
- Enable T8 Logging
- Change Statistics Storage Interval
- Enable CDRs
- Start/Restart Administrative Server
- Start/Restart Application Server
- Stop the Administrative and Application Servers
- Alarms
- Add New XSI to OCSG
- Change the Administrative Console Account Password
- Create User Account
- Change the Operator Account Password
- Purge Database Tables
- Set Up the Two-Way SSL Configuration, which includes:
  - Import Client Certificate
  - Import Server Certificate
- Change SSL Certificates and Private Keys

Configure DSR MP IPs in DSR API GW

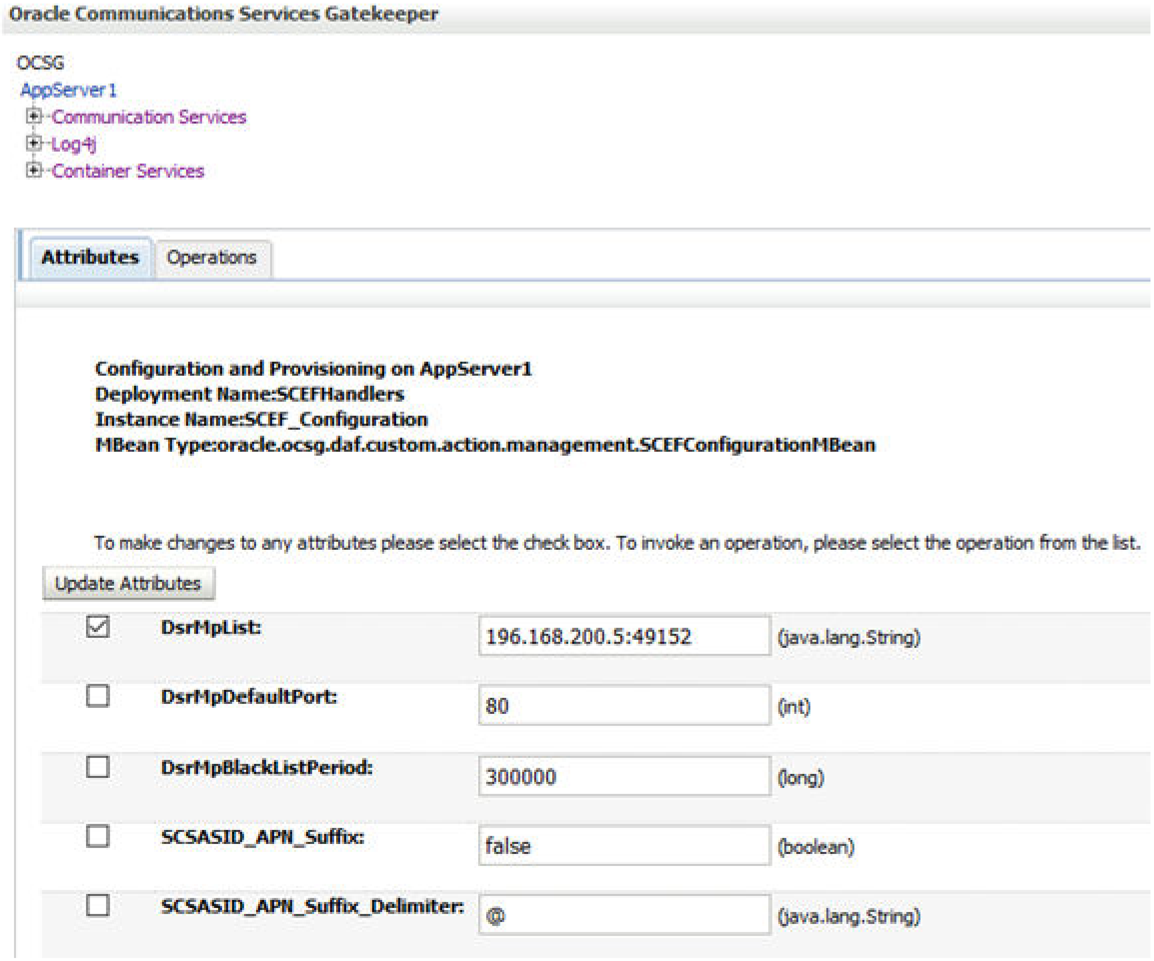
Any change made to an Application Server (AppServer) reflects to all other servers.

1. Access the DSR API GW Admin console using https://<Admin-Server-XMI-IP>:9002/console.
2. Login using the admin account created when configuring the API GW.

   The default username is weblogic.
3. Navigate to **OCSG** > **AppServerx** > **Communication Services** > **SCEF_Configuration** > **Attributes (tab)**.
4. If the ports are different for each MP server, change the *DsrMpList* to a ip1:port,ip2:port,ip3port... format.

   If the IPs are in a ip1,ip2, ip3... format, then provide the port in the *DsrMpDefaultPort*.
5. Mark the associated checkbox and click **Update Attributes**.

Figure A-1 Configure Communication Services Attributes



Configuration and Provisioning on AppServer1
Deployment Name:SCEFHandlers
Instance Name:SCEF_Configuration
MBean Type:oracle.ocsg.daf.custom.action.management.SCEFConfigurationMBean

To make changes to any attributes please select the check box. To invoke an operation, please select the operation from the list.

☑	**DsrMpList:**	196.168.200.5:49152	(java.lang.String)
☐	**DsrMpDefaultPort:**	80	(int)
☐	**DsrMpBlackListPeriod:**	300000	(long)
☐	**SCSASID_APN_Suffix:**	false	(boolean)
☐	**SCSASID_APN_Suffix_Delimiter:**	@	(java.lang.String)

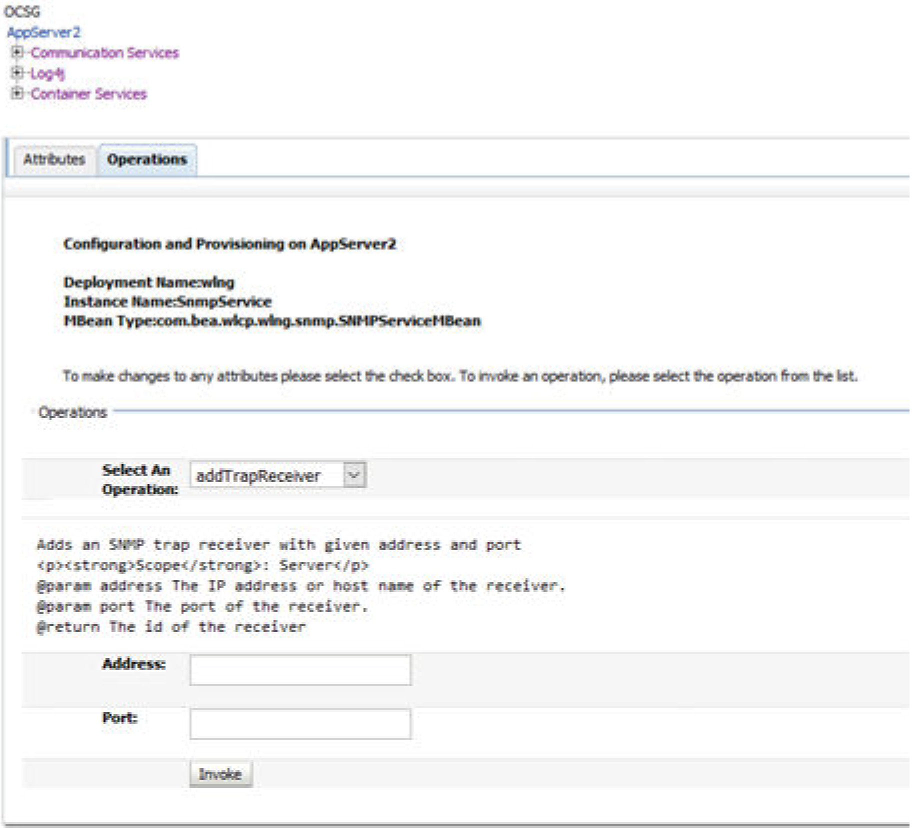
Add SNMP Trap Receiver

This procedure is performed on each AppServer.

1. Access the DSR API GW Admin console using https://<Admin-Server-XMI-IP>:9002/console.
2. Login using the admin account created when configuring the API GW.

   The default username is weblogic.
3. Navigate to **OCSG** > **AppServerx** > **Container Services** > **SnmpService** > **Operations (tab)**.
4. For the *AddTrapReceiver* operation, enter the **Address** (IP address of the SNMP trap receiver) and **Port** (Port to which the SNMP traps should be sent) information.

Figure A-2 Add SNMP Trap



Configuration and Provisioning on AppServer2

Deployment Name:wlng
Instance Name:SnmpService
MBean Type:com.bea.wlcp.wlng.snmp.SNMPServiceMBean

To make changes to any attributes please select the check box. To invoke an operation, please select the operation from the list.

Operations

Select An Operation: addTrapReceiver

```
Adds an SNMP trap receiver with given address and port
<p><strong>Scope</strong>: Server</p>
@param address The IP address or host name of the receiver.
@param port The port of the receiver.
@return The id of the receiver
```

Address:

Port:

Change SNMP Version

This procedure is performed on each AppServer.

1. Access the DSR API GW Admin console using https://<Admin-Server-XMI-IP>:9002/console.
2. Login using the admin account created when configuring the API GW.

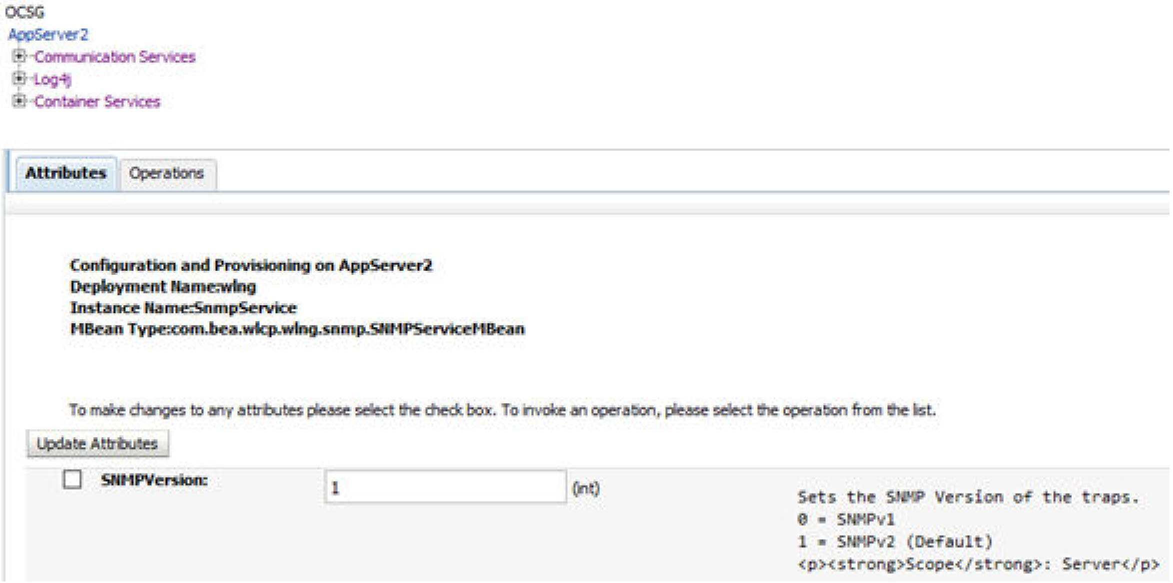
   The default username is weblogic.

3. Navigate to **OCSG** > **AppServerx** > **Container Services** > **SnmpService** > **Attributes (tab)**.

4. Change the *SNMPVersion* to 1 or 2.

5. Mark the associated checkbox and click **Update Attributes**.

Figure A-3 Change SNMP Version



Configuration and Provisioning on AppServer2
Deployment Name:wlng
Instance Name:SnmpService
MBean Type:com.bea.wlcp.wlng.snmp.SNMPServiceMBean

To make changes to any attributes please select the check box. To invoke an operation, please select the operation from the list.

□ **SNMPVersion:** 1 (int)

```
Sets the SNMP Version of the traps.
0 = SNMPv1
1 = SNMPv2 (Default)
<p><strong>Scope</strong>: Server</p>
```

Generate MIB File

1. Access the DSR API GW Admin console using https://<Admin-Server-XMI-IP>:9002/console.

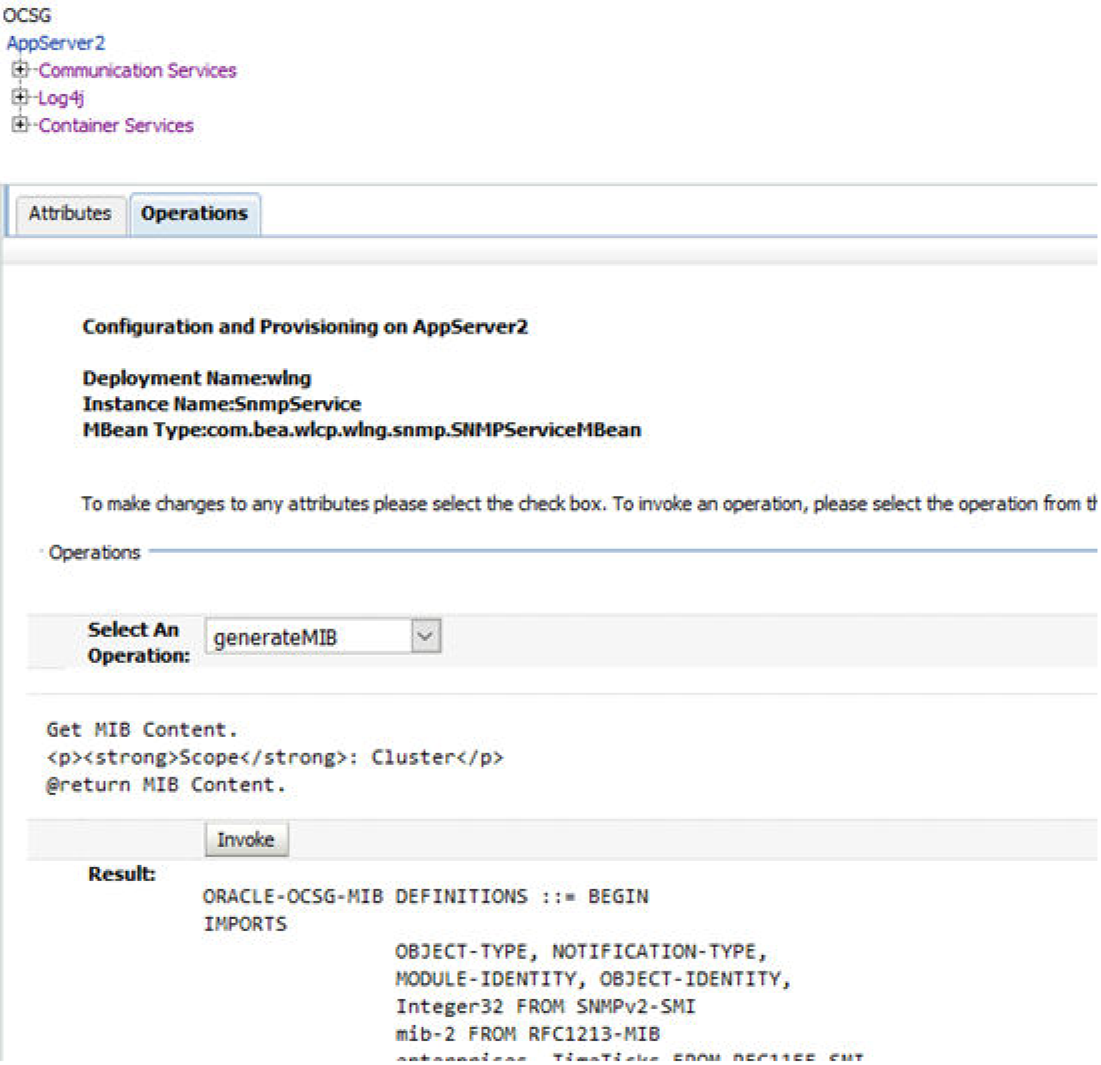
2. Login using the admin account created when configuring the API GW.

   The default username is weblogic.

3. Navigate to **OCSG** > **AppServerx** > **Container Services** > **SnmpService** > **Operations (tab)**.

4. Select the *generateMIB* operation and click **Invoke**.

Figure A-4 Generate MIB File



Configuration and Provisioning on AppServer2

Deployment Name:wlng
Instance Name:SnmpService
MBean Type:com.bea.wlcp.wlng.snmp.SNMPServiceMBean

To make changes to any attributes please select the check box. To invoke an operation, please select the operation from th

Operations

Select An Operation: generateMIB

```
Get MIB Content.
<p><strong>Scope</strong>: Cluster</p>
@return MIB Content.
```

Result:

```
ORACLE-OCSG-MIB DEFINITIONS ::= BEGIN
IMPORTS
                OBJECT-TYPE, NOTIFICATION-TYPE,
                MODULE-IDENTITY, OBJECT-IDENTITY,
                Integer32 FROM SNMPv2-SMI
                mib-2 FROM RFC1213-MIB
                enterprises, TimeTicks FROM RFC1155-SMI
```

Change General Logging Level

This procedure is performed on each AppServer.

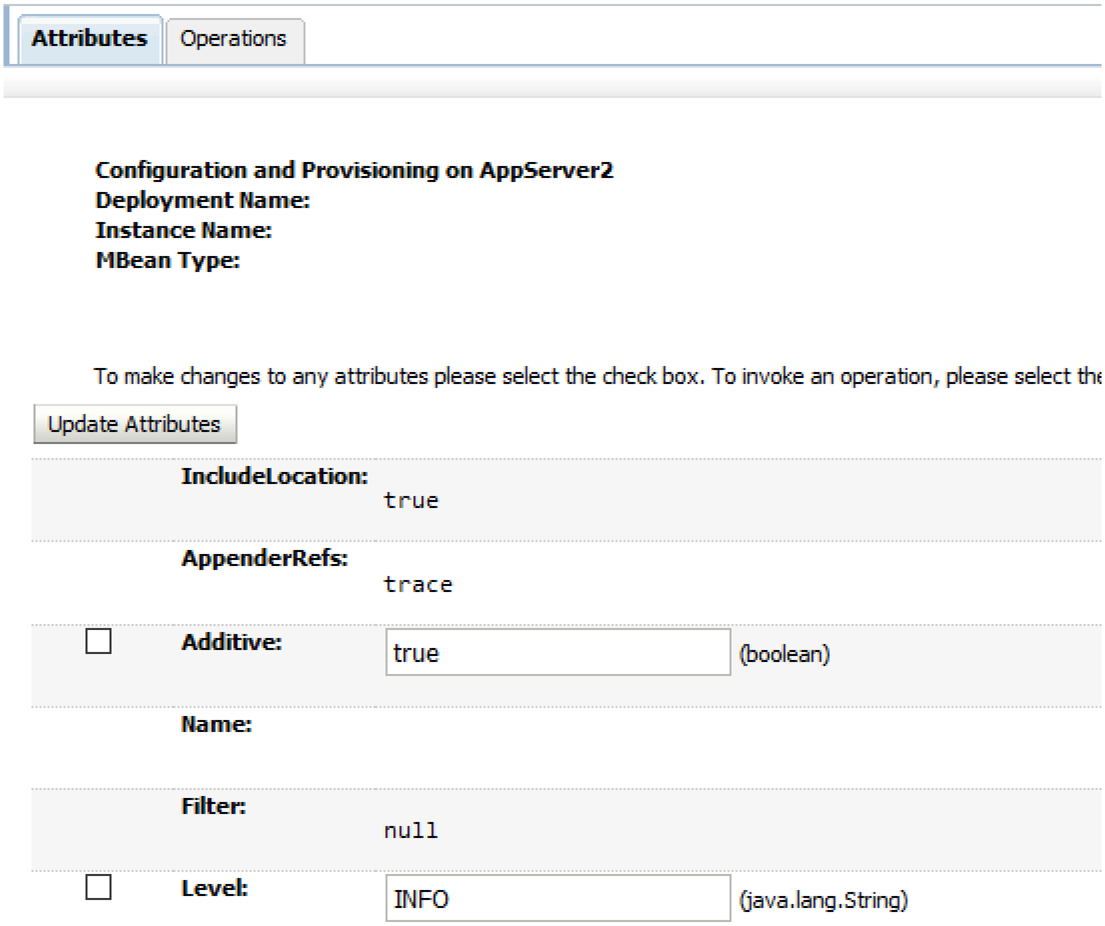
1. Access the DSR API GW Admin console using https://<Admin-Server-XMI-IP>:9002/console.
2. Login using the admin account created when configuring the API GW.

   The default username is weblogic.
3. Navigate to **OCSG** > **AppServerx** > **Log4J** > **log4j:Location=AppServerx,hierarchy=default,logger=root** > **Operations (tab)**.
4. Change the Level to one of these values
   - ALL
   - DEBUG
   - ERROR

- FATAL
- INFO
- OFF
- TRACE
- WARN

5. Mark the associated checkbox and click **Update Attributes**.

Figure A-5 Change Log Level



Configuration and Provisioning on AppServer2
Deployment Name:
Instance Name:
MBean Type:

To make changes to any attributes please select the check box. To invoke an operation, please select the

	IncludeLocation:	true	
	AppenderRefs:	trace	
□	**Additive:**	true	(boolean)
	Name:		
	Filter:	null	
□	**Level:**	INFO	(java.lang.String)

Enable T8 Logging

T8 Logging can be enabled per API (NIDD/ME/DT/ECR) through DSR API GW Partner and API management Portal.

The T8 Request and Responses are logged into the /u03/app/oracle/ocsg-x.x.x/user_projects/domains/ services-gatekeeper-domain/scef/t8.log file.

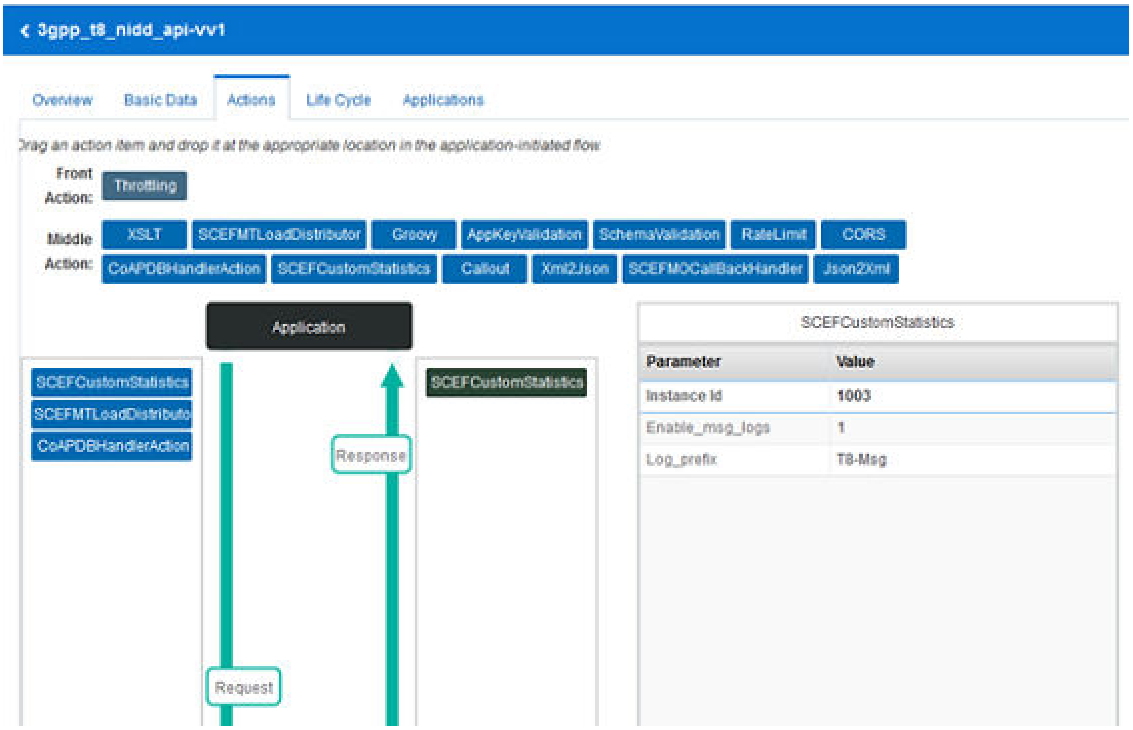
1. Access PRM portal using https://<Appserverx-XMI-IP>:9002/portal/partner-manager/index/login.html url.

2. Login using the admin account created when configuring the API GW.

   The default username is weblogic.

3. Click on APIs, and click on API for which T8 Logging should be enabled.
4. Select the Actions tab.
5. Select *SCEFCustomStatistics* on request path and update *Enable_msg_logs* to 1.
6. Click **Save**.
7. Repeat this procedure for the response path.
8. To disable T8 logging, repeat this procedure but change the *Enable_msg_logs* value to 0.

Figure A-6 Enable T8 Logging



Parameter	Value
Instance Id	1003
Enable_msg_logs	1
Log_prefix	T8-Msg

Change Statistics Storage Interval

DSR API GW generates various types of SCEF-related statistics that exist within the DSR API GW database table as slee_statistics_data. Each statistics is represented by a numerical ID.

Table A-1 NIDD Statistics

Statistic Name	Statistic ID
NIDD Configuration create	
- Request received	1021
- Successful response sent	1022
- Error response sent	1023
NIDD Configuration read	

Table A-1 (Cont.) NIDD Statistics

Statistic Name	Statistic ID
- Request received	1026
- Successful response sent	1027
- Error response sent	1028
Individual NIDD Configuration request	
- Request received	1031
- Successful response sent	1032
- Error response sent	1033
Individual NIDD Configuration delete request	
- Request received	1036
- Successful response sent	1037
- Error response sent	1038
Individual NIDD configuration read request	
- Request received	1041
- Successful response sent	1042
- Error response sent	1043
NIDD Download Link data deliveries create	
- Request received	1046
- Successful response sent	1047
- Error response sent	1048
NIDD Download Link data deliveries read	
- Request received	1051
- Successful response sent	1052
- Error response sent	1053
Individual NIDD Download Link data deliveries create	
- Request received	1056
- Successful response sent	1057
- Error response sent	1058
Individual NIDD Download Link data deliveries read	
- Request received	1061
- Successful response sent	1062

Table A-1 (Cont.) NIDD Statistics

Statistic Name	Statistic ID
- Error response sent	1063
NIDD Configuration update notification	
- Request sent	1066
- Successful response received	1067
- Error response received	1068
NIDD downlink data delivery status notification	
- Request sent	1071
- Successful response received	1072
- Error response received	1073
NIDD uplink data delivery status notification	
- Request sent	1076
- Successful response received	1077
- Error response received	1078

Table A-2 Event Monitoring Statistics

Statistic Name	Statistic ID
Monitoring event subscriptions read	
- Request received	1111
- Successful response sent	1112
- Error response sent	1113
Monitoring event subscriptions create	
- Request received	1116
- Successful response sent	1117
- Error response sent	1118
Individual monitoring event subscriptions read	
- Request received	1121
- Successful response sent	1122
- Error response sent	1123
Individual monitoring event subscriptions create	
- Request received	1126

Table A-2 (Cont.) Event Monitoring Statistics

Statistic Name	Statistic ID
- Successful response sent	1127
- Error response sent	1128
Individual monitoring event subscriptions delete	
- Request received	1131
- Successful response sent	1132
- Error response sent	1133
Monitoring event notification	
- Request sent	1136
- Successful response received	1137
- Error response received	1138

Table A-3 Device Triggering Statistics

Statistic Name	Statistic ID
Device Triggering transaction	
- Request received	1081
- Successful response sent	1082
- Error response sent	1083
Device Triggering transaction	
- Request received	1086
- Successful response sent	1087
- Error response sent	1088
Individual Device Triggering transaction	
- Request received	1091
- Successful response sent	1092
- Error response sent	1093
Individual Device Triggering transaction	
- Request received	1096
- Successful response sent	1097
- Error response sent	1098
Individual Device Triggering transaction	
- Request received	1101

Table A-3 (Cont.) Device Triggering Statistics

Statistic Name	Statistic ID
- Successful response sent	1102
- Error response sent	1103
Device Triggering delivery report notification	
- Request received	1106
- Successful response sent	1107
- Error response sent	1108

Table A-4 Enhanced Coverage Restriction Statistics

Statistic Name	Statistic ID
Configure	
- Request received	1141
- Successful response sent	1142
- Error response sent	1143
Query	
- Request received	1146
- Successful response sent	1147
- Error response sent	1148

Statistics are stored in the database at a configurable interval, which is configurable using the Admin console. Statistics are collected for the configured interval and stored in database in different records as one per Application per AppServer.

To change the statistics storage interval, follow this procedure:

1. Access the DSR API GW Admin console using https://<Admin-Server-XMI-IP>: 9002/console.
2. Login using the admin account created when configuring the API GW.

   The default username is weblogic.
3. Navigate to **OCSG** > **AppServerx** > **Container Services** > **Statistics Service** > **Attributes (tab)**.
4. Change the *StoreInterval* to desired seconds.
5. Mark the associated checkbox and click **Update Attributes**.

Enable CDRs

Any change made to an Application Server (AppServer) reflects to all other servers.

1. Access the DSR API GW Admin console using https://<Admin-Server-XMI-IP>: 9002/console.

2. Login using the admin account created when configuring the API GW.

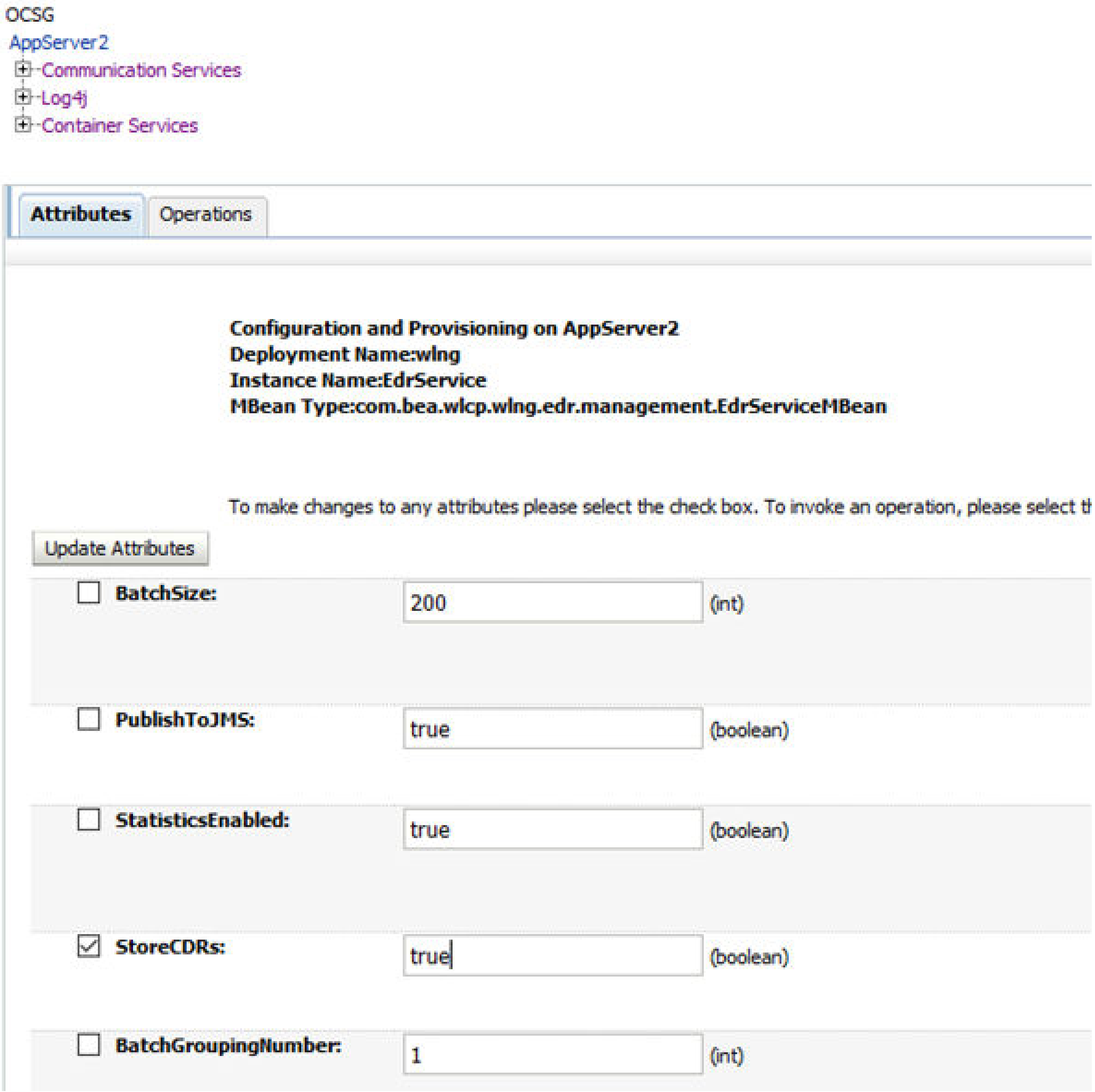
   The default username is weblogic.

3. Navigate to **OCSG** > **AppServerx** > **Container Services** > **Edr Service** > **Attributes (tab)**.

4. Change the *StoreCdrs* parameter to true.

5. Mark the associated checkbox and click **Update Attributes**.

Figure A-7 Enable CDRs



Configuration and Provisioning on AppServer2
Deployment Name:wlng
Instance Name:EdrService
MBean Type:com.bea.wlcp.wlng.edr.management.EdrServiceMBean

To make changes to any attributes please select the check box. To invoke an operation, please select th

□	**BatchSize:**	200	(int)
□	**PublishToJMS:**	true	(boolean)
□	**StatisticsEnabled:**	true	(boolean)
☑	**StoreCDRs:**	true	(boolean)
□	**BatchGroupingNumber:**	1	(int)

Start/Restart Administrative Server

The procedure to start and restart the server is the same.

1. Make sure the nodemgr service is running on the Admin server by with this command:

```
sudo service nodemgr status
```

If the nodemgr service is not running, start the service:

```
sudo service nodemgr start
```

2. cd to /u03/app/oracle/ocsg-x.x.x/user_projects/domains/services-gatekeeper-domain.
3. Execute

   ```
   source ../../../wlserver/server/bin/setWLSEnv.sh
   ```

4. Execute

   ```
   java weblogic.WLST”, this will start a WLST prompt
   ```

5. Execute this command at the WLDT prompt

   ```
   Wls:/offiline: nmConnect(‘<nodemanager-user-name', ‘nodemanager-password',
   'adminserver-imi-ip', '5556', 'services-gatekeeper-domain', '/u03/app/oracle/
   ocsg-x.x.x/user_projects/domains/services-gatekeeper-domain','plain')
   ```

   nodemanager-user-name - user name of node manager provided while configuring DSR API GW

   nodemanager-password - password of node manager provided while configuring DSR AI GW

   ocsg-x.x.x. - current DSR API GW version installed

6. Execute

   ```
   wls:/nm/services-gatekeeper-domain> nmStart(‘AdminServer’)
   ```

7. Make sure the Admin server has successfully started by accessing the console URL at https://<Admin-Server-XMI-IP>:9002/console.

Start/Restart Application Server

The procedure to start and restart the server is the same.

1. Make sure the nodemgr service is running on the Admin server by with this command:

   ```
   sudo service nodemgr status
   ```

   If the nodemgr service is not running, start the service:

   ```
   sudo service nodemgr start
   ```

2. Access the DSR API GW Admin console using https://<Admin-Server-XMI-IP>:9002/console.
3. Login using the admin account created when configuring the API GW.

   The default username is weblogic.

4. Navigate to **Environment** > **Servers** > **Control (tab)**.
5. Click **Start** and confirm.

Stop the Administrative and Application Servers

1. Access the DSR API GW Admin console using https://<Admin-Server-XMI-IP>:9002/console.
2. Login using the admin account created when configuring the API GW.

   The default username is weblogic.
3. Navigate to **Environment** > **Servers** > **Control (tab)**.
4. Select the server to stop and click **Stop**.
5. Select **Force shutdown now** and confirm.

Alarms

Alarms are raised by DSR API GW for different events. If SNMP is configured, alarms are sent as SNMP traps. OCSG alarms related to SCEF are described in the Alarms and KPIs reference guide.

Add New XSI to OCSG

This procedure adds a new External Signaling Interface (XSI) to the Oracle Communications Services Gatekeeper (OCSG).

1. Attach XSI interface to VMs.
2. Configure ifcfg files so the network is configured and the IP address is picked up by the VM.
3. Run these command to open ports in the firewall for the new XSI:

   ```
   sudo iptablesAdm append
                     --type=rule --protocol=IPv4 --domain=01dsrapigw --
   table=filter --chain=INPUT
                     --match='-m state --state NEW -m tcp -p tcp --dport 10001 -
   d <XSI-IP> -j
                     ACCEPT' --persist=yes

   sudo iptablesAdm append
                     --type=rule --protocol=IPv4 --domain=01dsrapigw --
   table=filter --chain=INPUT
                     --match='-m state --state NEW -m tcp -p tcp --dport 10002 -
   d <XSI-IP> -j
                     ACCEPT' --persist=yes
   ```
4. Access the DSR API GW Admin console using https://<Admin-Server-XMI-IP>:9002/console.
5. Login using the admin account created when configuring the API GW.

   The default username is weblogic.
6. Navigate to **Environment** > **Servers** > **AppServerx** > **Protocols** > **Channels**.
7. Click **Lock & Edit**.

8. Add new channels.

   Each channel name should be unique.

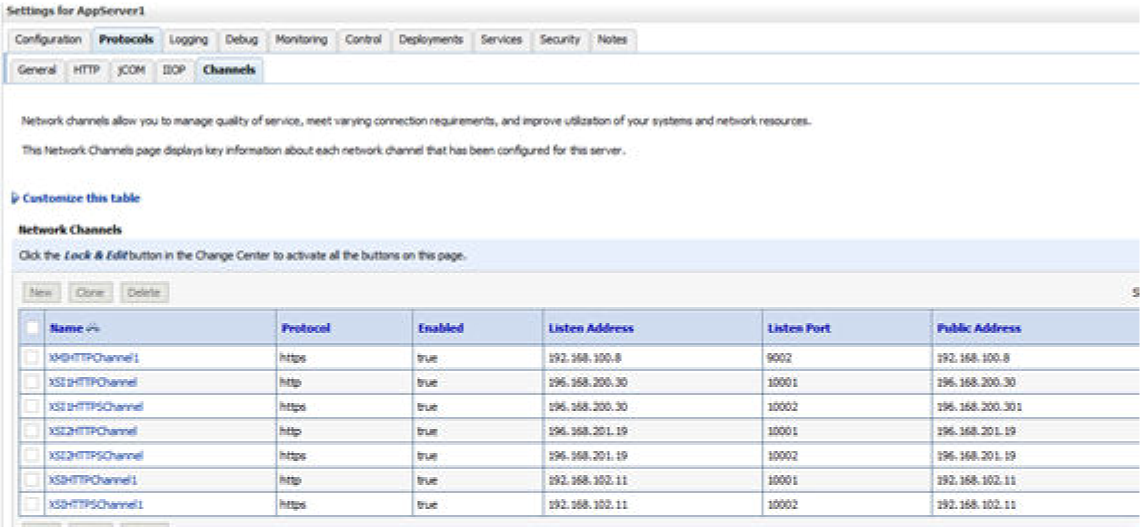
   Change the name, protocol, IP and port.

   Leave the rest of the options as default.

9. Add a channel for the new XSI IP and 10001 for http protocol.

10. Add another channel for XSI and 10002 for https protocol.

Figure A-8 Add New XSI to OCSG



Network channels allow you to manage quality of service, meet varying connection requirements, and improve utilization of your systems and network resources.

This Network Channels page displays key information about each network channel that has been configured for this server.

Network Channels

Click the *Lock & Edit* button in the Change Center to activate all the buttons on this page.

□	Name	Protocol	Enabled	Listen Address	Listen Port	Public Address
□	XMIHTTPChannel1	https	true	192.168.100.8	9002	192.168.100.8
□	XSI1HTTPChannel	http	true	196.168.200.30	10001	196.168.200.30
□	XSI1HTTPSChannel	https	true	196.168.200.30	10002	196.168.200.301
□	XSI2HTTPChannel	http	true	196.168.201.19	10001	196.168.201.19
□	XSI2HTTPSChannel	https	true	196.168.201.19	10002	196.168.201.19
□	XSIHTTPChannel1	http	true	192.168.102.11	10001	192.168.102.11
□	XSIHTTPSChannel1	https	true	192.168.102.11	10002	192.168.102.11

Change the Administrative Console Account Password

1. Access the DSR API GW Admin console using https://<Admin-Server-XMI-IP>:9002/console.

2. Login using the admin account created when configuring the API GW.

   The default username is weblogic.

3. Navigate to **Security Realms** > **myrealm** > **User and Groups** > **Users** > **weblogic** > **passwords**.

4. Stop all the Administrative and Application servers.

   See the Stop the Administrative and Application Servers procedure.

5. In each server, navigate to the u03/app/oracle/ocsg-x.x.x/user_projects/domains/ services-gatekeeper-domain/servers/<Server-name>/security folder.

6. Delete the boot.properties file.

7. Recreate the boot.properties file with a new username and password.

   username=<user-name>

   Password=<password>

Note: These details are encrypted when the server starts successfully.

---

8. Stare all the Administrative and Application servers.

   See the Start/Restart Administrative Server and Start/Restart Application Server procedures.

Create User Account

1. Access the DSR API GW Admin console using https://<Admin-Server-XMI-IP>:9002/console.
2. Login using the admin account created when configuring the API GW.

   The default username is weblogic.
3. Navigate to **OCSG** > **AppServerx** > **Container Services** > **Management Users** > **ManagementUsers** > **Operations (tab)**.
4. Select *addUser*.
5. Provide the new username, password, Userlevel (1000), and Type (1) and click **Invoke**.

Change the Operator Account Password

1. Access the DSR API GW Admin console using https://<Admin-Server-XMI-IP>:9002/console.
2. Login using the admin account created when configuring the API GW.

   The default username is weblogic.
3. Navigate to **OCSG** > **AppServerx** > **Container Services** > **Management Users** > **ManagementUsers** > **Operations (tab)**.
4. Select *setUserPassword*.
5. Provide the new username and password and click **Submit**.

Purge Database Tables

This section lists the database tables you should purge periodically on the OCSG database. How often the tables are cleaned depends on the traffic capacity of the site. This section provides some recommendations.

About Cleaning Database Tables

Table A-5 lists the database tables that you must periodically clean to prevent them from growing too large and adversely affecting performance.

Table A-5 Database Table Cleaning Intervals

Table	Recommended Cleaning Interval
SLEE_ALARM	Every two months
SLEE_CHARGING (if cdrs is enabled)	Depends on the traffic capacity of the site
SLEE_STATISTICS_DATA	Every month

Set Up Two-Way SSL Configuration

Two-way SSL configuration mandates clients, opening an HTTPS connection to DSR API GW, present a client certificate (that is validated by DSR API GW) before opening a connection.

The trusted client certificate should be imported to DSR API GW server so the validation is successful.

Import Client Certificate

This procedure to import the client certificate is performed on each AppServer.

1. Access the DSR API GW Admin console using https://<Admin-Server-XMI-IP>: 9002/console.
2. Login using the admin account created when configuring the API GW.

   The default username is weblogic.
3. Navigate to **Environment** > **Servers** > **AdminServer or AppServerx** > **Configuration** > **KeyStore**.
4. Note the trust store file path.
5. SSH to the corresponding server and browse to the trust store file path.
6. Copy the client certificate(.cer file) to import to the current directory.
7. Execute this command to import the certificate to the trust store (trust store passphrase should be entered):

   ```
   keytool -import -alias <any-alias-name-for-cert> -file <certificate-file> -
   keystore <trust-store-name>
   ```
8. Restart the corresponding server.
9. Access the DSR API GW Admin console using https://<Admin-Server-XMI-IP>: 9002/console.
10. Login using the admin account created when configuring the API GW.

    The default username is weblogic.
11. Navigate to **Environment** > **Servers** > **AdminServer or AppServerx** > **Configuration** > **SSL**.
12. Click **Advanced**.
13. Click **Lock and Edit**.
14. Change *Two Way Client Cert Behavior* to `Client Certs Requested And Enforced`.
15. Click **Save** and **Active Changes**.

Import Server Certificate

This procedure imports the server certificates on the application servers, which the DSR API GW then sends to SCEF reports. The request is initiated by the DSR API GW towards the application server over HTTPS.

1. Access the DSR API GW Admin console using https://<Admin-Server-XMI-IP>: 9002/console.
2. Login using the admin account created when configuring the API GW.

   The default username is weblogic.
3. Navigate to **Environment** > **Servers** > **AdminServer or AppServerx** > **Configuration** > **KeyStore**.
4. Note the trust store file path.
5. SSH to the corresponding server and browse to the trust store file path.
6. Copy the client certificate(.cer file) to import to the current directory.
7. Execute this command to import the certificate to the trust store (trust store passphrase should be entered):

   ```
   keytool -import -alias <any-alias-name-for-cert> -file <certificate-file> -
   keystore <trust-store-name>
   ```
8. Restart the corresponding server.

Change SSL Certificates and Private Keys

DSR API GW shipped with a demo certificate and private key, which are not recommended for use in a production environment.

To change the demo certificate and private keys of DSR API GW, obtain:

- A CA signed digital certificate (.pem file) and private key for each DSR API GW server separately.
- A root certificate of CA and any other intermediate certificates used to sign the digital certificate.

This procedure is performed on each AppServer.

1. SSH to the server.
2. Browse to the /u03/app/oracle/ocsg-x.x.x/user_projects/domains/ services-gatekeeper-domain/security directory.

   Replace x.x.x with the DSR API GW version.
3. Copy the signed certificate, private key, CA root and intermediate certificates (if any) to the current directory.
4. Execute

   ```
   source ../../../../wlserver/server/bin/setWLSEnv.sh
   ```

5. Create a custom key store and import the private key and signed digital certificate with this command.

   ```
   java utils.ImportPrivateKey -keystore  SeverIdentity.jks -storepass
   <storepass>
    -storetype JKS -keypass <keypass> -alias <skey> -certfile <serverCert.pem> -
   keyfile <ServerKey.pem>
    -keyfilepass <keypass>
   ```

   Keystore: SeverIdentity.jks -JKS file in which the certificate and key will be imported.

   Storepass: storepass - This is the password of the keystore file severIdentity.jks

   Storetype: JKS - Java Key Store.

   Keypass: keypass - This password will be configured in server which will be used to read the Private Key from the keystore.

   Alias: skey - This is the alias used for reading the Private Key from the Keystore.

   Certfile: serverCert.pem - This is the certificate to be imported into the Keystore.

   Keyfile: ServerKey.pem - This is the Private Key to be imported into the Keystore.

   Keyfilepass: keypass - This is the Password required to read the Private Key from the ServerKey.pem file

6. Create a custom trust store (java key store) and import the CA root certificate with this command.

   ```
   keytool -import -file <ca.cert> -alias <firstCA> -keystore <ServerTrust.jks> -
   storepass <storepass>
   ```

   ca.cert - ca root certificate to be imported

   firstCA - an alias to the certificate

   ServerTrust.jks - trust store with the name will be created

   Storepass - trust store pass phrase

7. Access the DSR API GW Admin console using https://<Admin-Server-XMI-IP>: 9002/console.

8. Login using the admin account created when configuring the API GW.

   The default username is weblogic.

9. Navigate to **Environment** > **Servers** > **AdminServer or AppServerx** > **Configuration** > **KeyStore**.

10. Click **Advanced**.

11. Click **Lock and Edit**.

12. Change *Keystores* to `Custom Identity and Custom Trust`.

13. Provide these values:

    - Custom Identity Keystore
    - Custom Identity Keystore Type

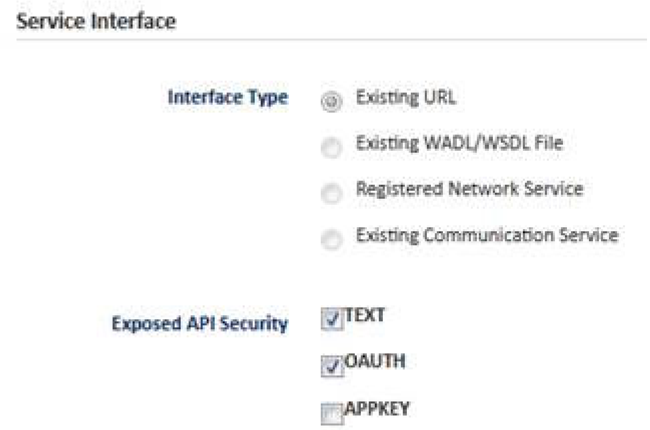
- Custom Identity Keystore Passphrase
- Confirm Custom Identity Keystore Passphrase
- Custom Trust Keystore
- Custom Trust Keystore Type
- Custom Trust Keystore Passphrase
- Confirm Custom Trust Keystore Passphrase

14. Select the SSL tab and provide these values:
    - Private Key Alias
    - Private Key Passphrase
    - Confirm Private Key Passphrase
15. Click **Activate Changes**.
16. Navigate to **Environment** > **Servers** > **Control (tab)** to restart the server.

Open Authorization Configuration Overview

Open Authorization or OAuth is an open standard for token-based authentication and authorization on the Internet. OAuth allows an end user's account information to be used by third-party services, such as Facebook, without exposing the user's password. This section describes an alternative configuration to modifying the APIs to authenticate with OCSG. The installation script automatically creates the APIs with support for OAuth as shown in Figure A-9.

Figure A-9 OAuth Installation Script



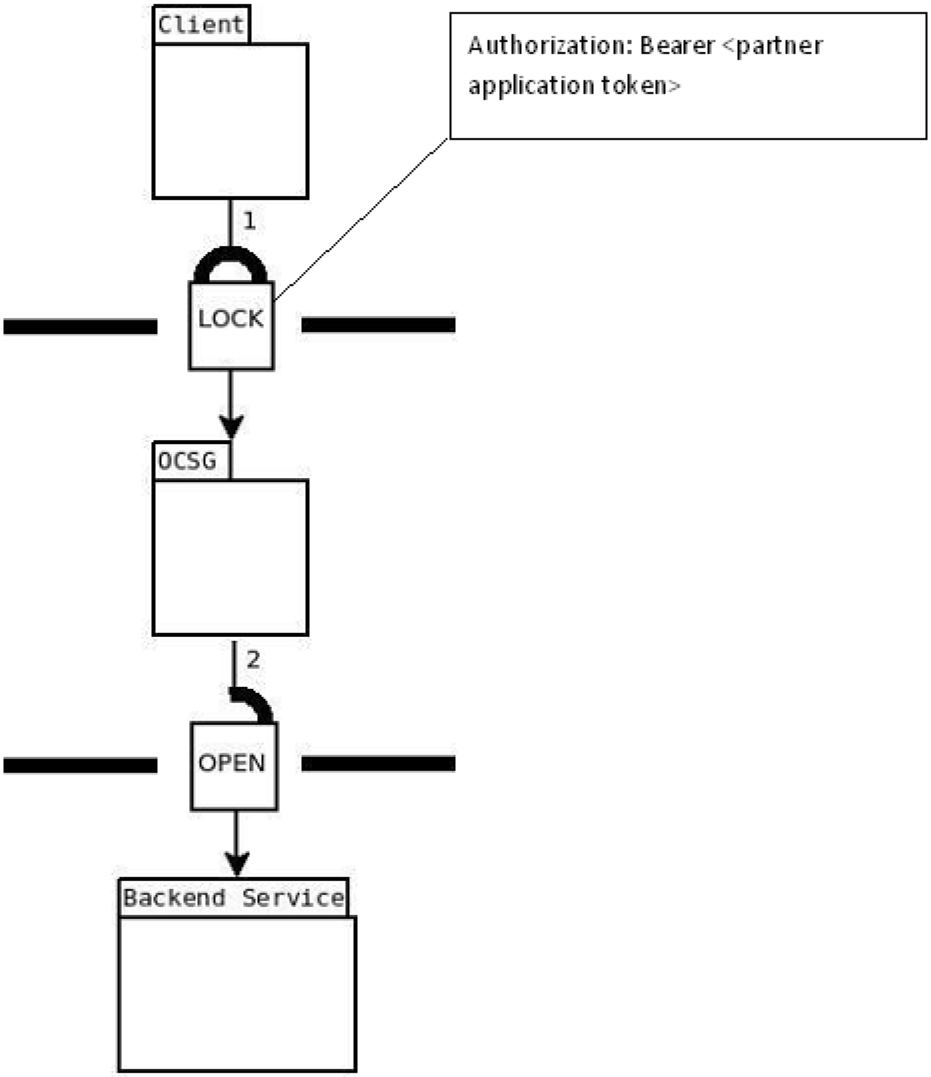
Service Interface

Interface Type
- ◉ Existing URL
- ○ Existing WADL/WSDL File
- ○ Registered Network Service
- ○ Existing Communication Service

Exposed API Security
- ☑ TEXT
- ☑ OAUTH
- □ APPKEY

Authorization take place after client has been created and between the two firewalls as shown in Figure A-10.

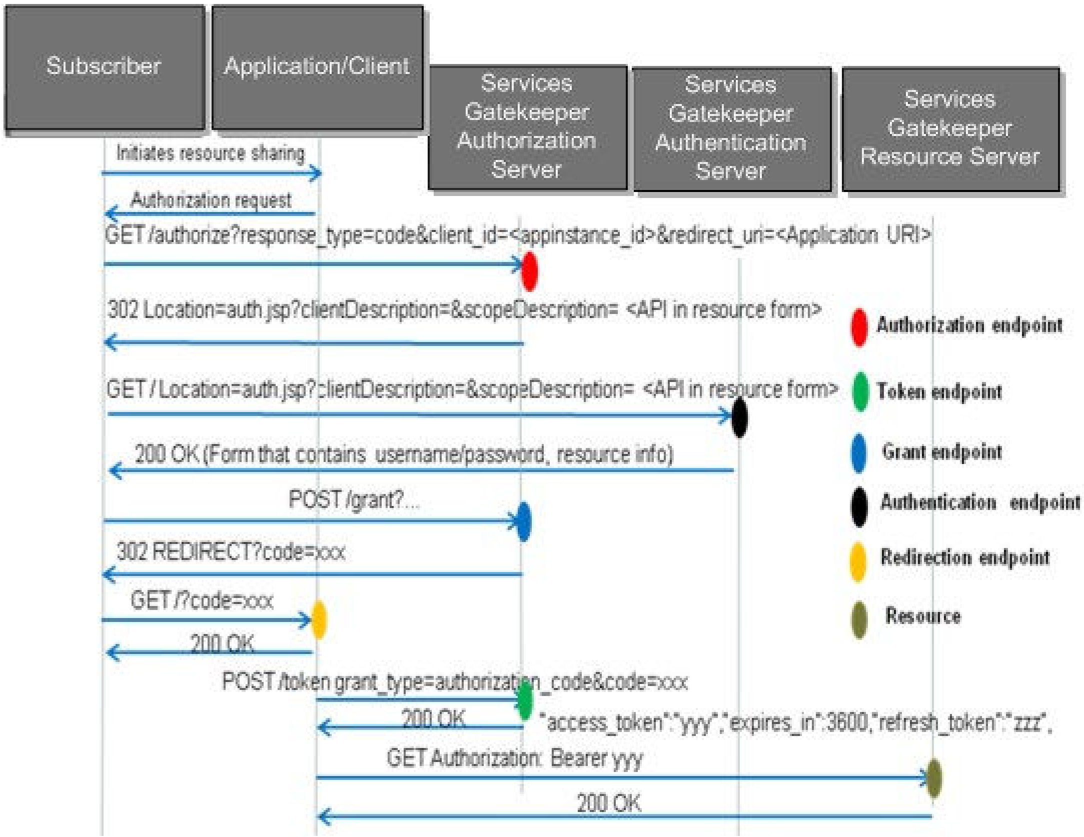
Figure A-10 Authorization Overview



This section assumes an API has been created and published and that the corresponding partner application has also been created. After the application has been created, assigned to a group, set up with the user account, set up the authorization as described in this section.

The Authorization Code grant type is used by confidential and public clients to exchange an authorization code for an access token. After the user returns to the client via the redirect URL, the application acquires the authorization code from the URL and uses it to request an access token. Figure A-11 shows this process using the resource owner authentication and code grant redirect.

Figure A-11 OAuth Code Grant

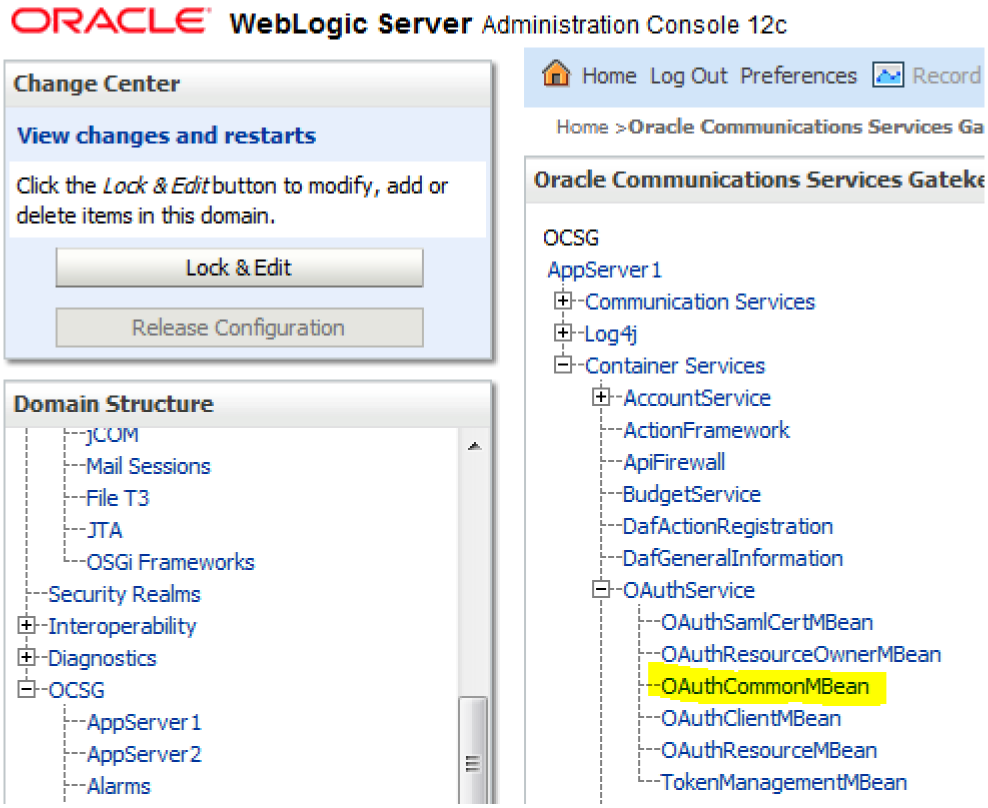

Subscriber
Application/Client
Services Gatekeeper Authorization Server
Services Gatekeeper Authentication Server
Services Gatekeeper Resource Server
Initiates resource sharing
Authorization request
GET /authorize?response_type=code&client_id=<appinstance_id>&redirect_uri=<Application URI>
302 Location=auth.jsp?clientDescription=&scopeDescription= <API in resource form>
GET / Location=auth.jsp?clientDescription=&scopeDescription= <API in resource form>
200 OK (Form that contains username/password, resource info)
POST /grant?...
302 REDIRECT?code=xxx
GET /?code=xxx
200 OK
POST /token grant_type=authorization_code&code=xxx
200 OK
"access_token":"yyy","expires_in":3600,"refresh_token":"zzz",
GET Authorization: Bearer yyy
200 OK
Authorization endpoint
Token endpoint
Grant endpoint
Authentication endpoint
Redirection endpoint
Resource

Set Up Authentication and Grant Redirect URLs

The first step to authorization is to set up the authentication and grant redirect URLs.

1. Navigate to **OCSG** > **AppServerx** > **Container Services** > **OAuthService** > **OAuthCommonMBean**.

Figure A-12 OAuthCommonMBean



ORACLE WebLogic Server Administration Console 12c

Home Log Out Preferences Record

Home >Oracle Communications Services Ga

Change Center

View changes and restarts

Click the *Lock & Edit* button to modify, add or delete items in this domain.

Lock & Edit

Release Configuration

Domain Structure

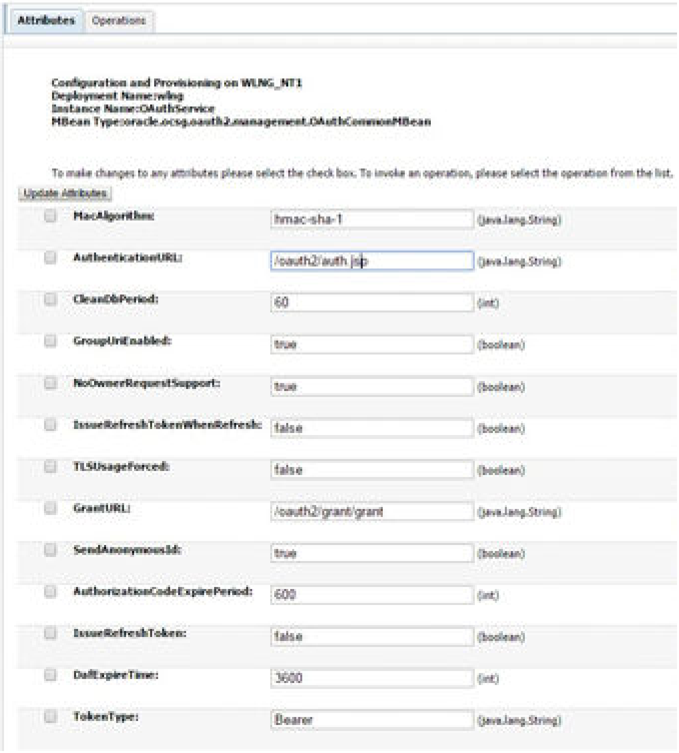
```
        ---JCOM
        ---Mail Sessions
        ---File T3
        ---JTA
        ---OSGi Frameworks
    ---Security Realms
  ⊞---Interoperability
  ⊞---Diagnostics
  ⊟---OCSG
        ---AppServer1
        ---AppServer2
        ---Alarms
```

Oracle Communications Services Gateke

```
OCSG
AppServer1
  ⊞--Communication Services
  ⊞--Log4j
  ⊟--Container Services
      ⊞--AccountService
        ---ActionFramework
        ---ApiFirewall
        ---BudgetService
        ---DafActionRegistration
        ---DafGeneralInformation
      ⊟--OAuthService
            ---OAuthSamlCertMBean
            ---OAuthResourceOwnerMBean
            ---OAuthCommonMBean
            ---OAuthClientMBean
            ---OAuthResourceMBean
            ---TokenManagementMBean
```

2. Set up the authentication and grand redirect URLs.

Figure A-13 Authentication and Grand Redirect URLs



Attributes | Operations

Configuration and Provisioning on WLNG_NT1
Deployment Name:wlng
Instance Name:OAuthService
MBean Type:oracle.ocsg.oauth2.management.OAuthCommonMBean

To make changes to any attributes please select the check box. To invoke an operation, please select the operation from the list.

Update Attributes

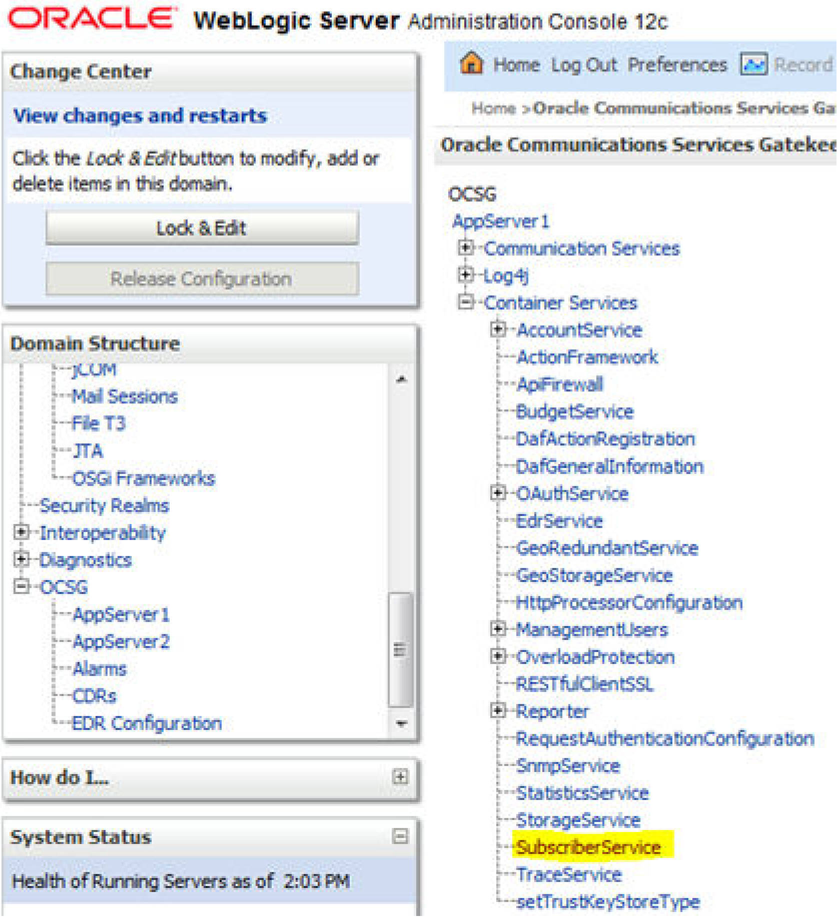
□	**MacAlgorithm:**	hmac-sha-1	(java.lang.String)
□	**AuthenticationURL:**	/oauth2/auth.jsp	(java.lang.String)
□	**CleanDbPeriod:**	60	(int)
□	**GroupUriEnabled:**	true	(boolean)
□	**NoOwnerRequestSupport:**	true	(boolean)
□	**IssueRefreshTokenWhenRefresh:**	false	(boolean)
□	**TLSUsageForced:**	false	(boolean)
□	**GrantURL:**	/oauth2/grant/grant	(java.lang.String)
□	**SendAnonymousId:**	true	(boolean)
□	**AuthorizationCodeExpirePeriod:**	600	(int)
□	**IssueRefreshToken:**	false	(boolean)
□	**DafExpireTime:**	3600	(int)
□	**TokenType:**	Bearer	(java.lang.String)

Subscriber

The Services Gatekeeper offers a built-in subscriber repository to authenticate subscribers. To use the default Subscriber Manager to authenticate subscribers, follow these steps:

1. Navigate to **OCSG** > **AppServerx** > **Container Services** > **SubscriberService**.

   Figure A-14 SubscriberService

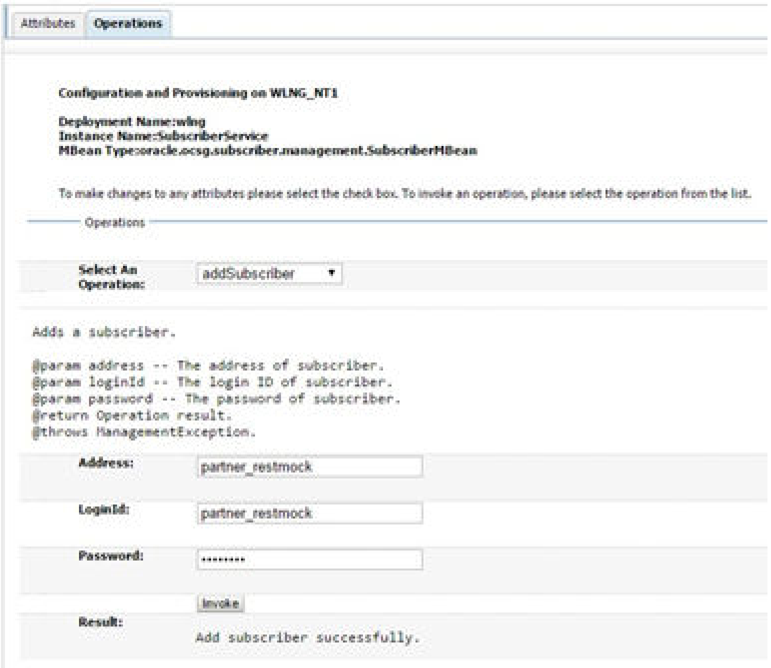


Click the *Lock & Edit* button to modify, add or delete items in this domain.

Health of Running Servers as of 2:03 PM

2. Create a subscriber account to use for authentication purposes.

Figure A-15 Subscriber



Configuration and Provisioning on WLNG_NT1

Deployment Name:wlng
Instance Name:SubscriberService
MBean Type:oracle.ocsg.subscriber.management.SubscriberMBean

To make changes to any attributes please select the check box. To invoke an operation, please select the operation from the list.

Operations

Select An Operation: addSubscriber

```
Adds a subscriber.

@param address -- The address of subscriber.
@param loginId -- The login ID of subscriber.
@param password -- The password of subscriber.
@return Operation result.
@throws ManagementException.
```

Address: partner_restmock

LoginId: partner_restmock

Password: ••••••••

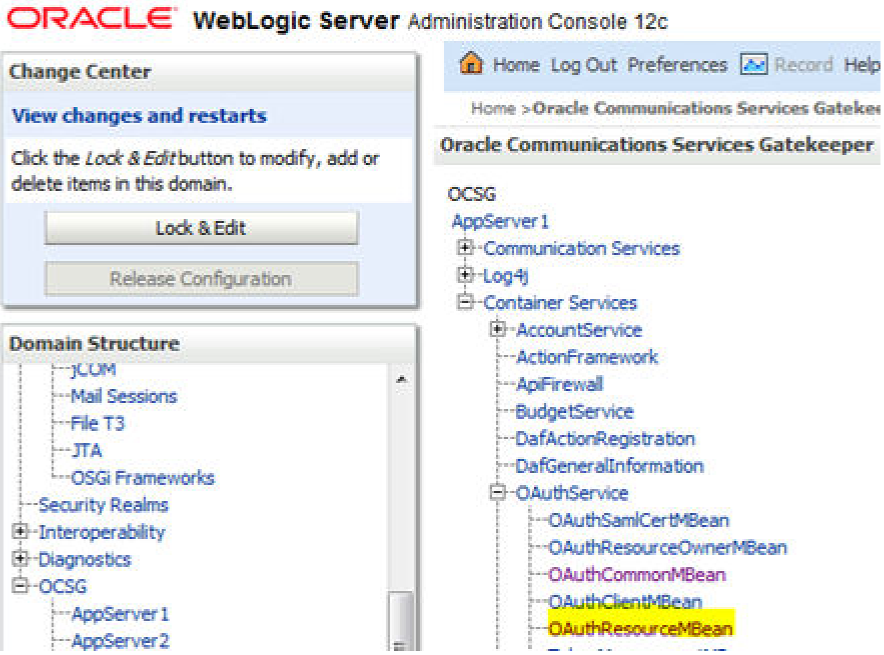
Result: Add subscriber successfully.

Resource Owner

The final step to open authorization is to set up the resource owner and associated resources.

1. Find the resource ID by navigating to **OCSG** > **AppServerx** > **Container Services** > **OAuthService** > **OAuthResourceMBean**.

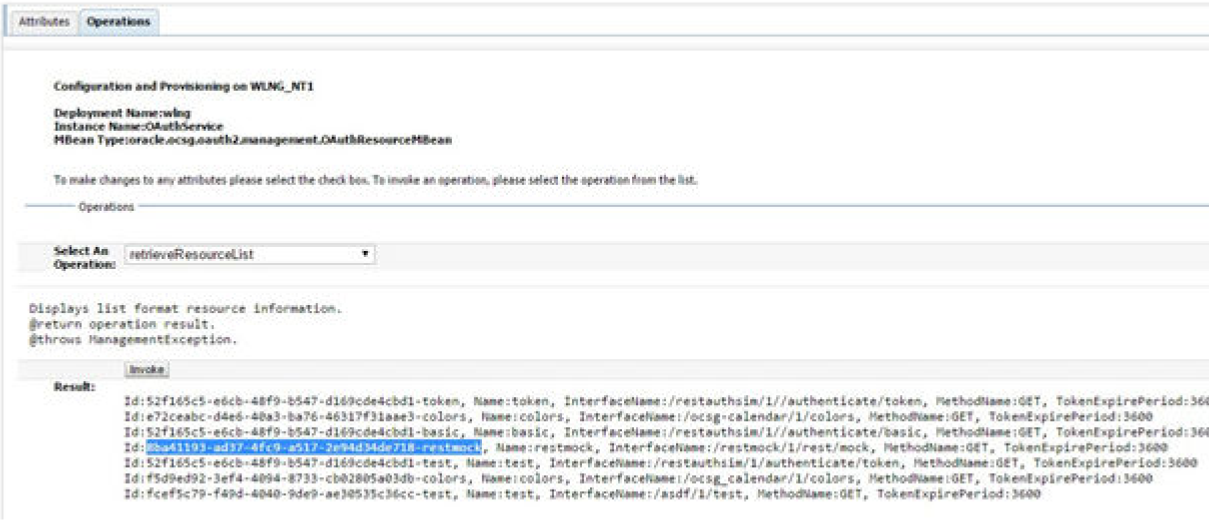
   Figure A-16 OAuthResourceMBean

   Click the *Lock & Edit* button to modify, add or delete items in this domain.

2. Find the resource ID corresponding to the application to be authenticated.

Figure A-17   Resource ID



Configuration and Provisioning on WLNG_NT1

Deployment Name:wlng
Instance Name:OAuthService
MBean Type:oracle.ocsg.oauth2.management.OAuthResourceMBean

To make changes to any attributes please select the check box. To invoke an operation, please select the operation from the list.

Select An Operation: retrieveResourceList

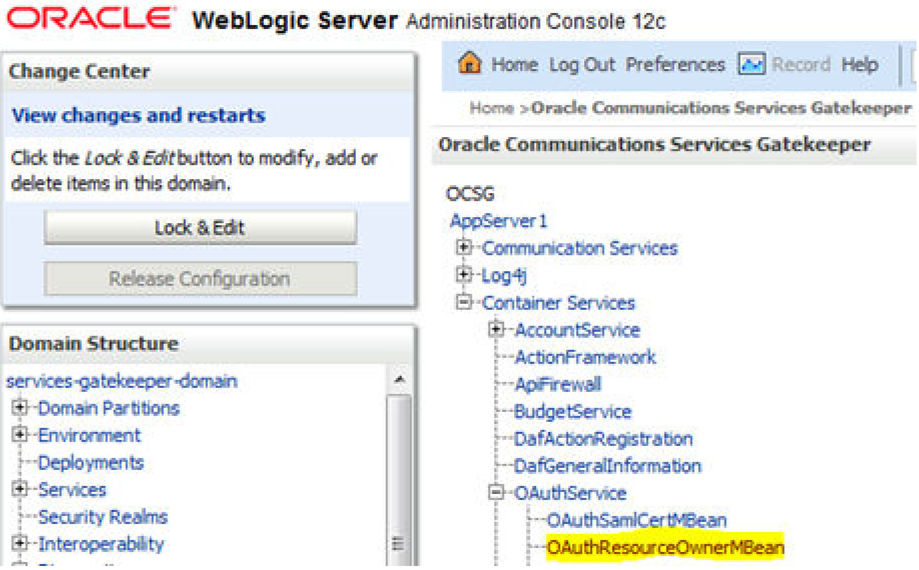
```
Displays list format resource information.
@return operation result.
@throws ManagementException.
```

Result:

```
Id:52f165c5-e6cb-48f9-b547-d169cde4cbd1-token, Name:token, InterfaceName:/restauthsim/1//authenticate/token, MethodName:GET, TokenExpirePeriod:36
Id:e72ceabc-d4e6-40a3-ba76-46317f31aae3-colors, Name:colors, InterfaceName:/ocsg-calendar/1/colors, MethodName:GET, TokenExpirePeriod:3600
Id:52f165c5-e6cb-48f9-b547-d169cde4cbd1-basic, Name:basic, InterfaceName:/restauthsim/1//authenticate/basic, MethodName:GET, TokenExpirePeriod:36
Id:8ba41193-ad37-4fc9-a517-2e94d34de718-restmock, Name:restmock, InterfaceName:/restmock/1/rest/mock, MethodName:GET, TokenExpirePeriod:3600
Id:52f165c5-e6cb-48f9-b547-d169cde4cbd1-test, Name:test, InterfaceName:/restauthsim/1/authenticate/token, MethodName:GET, TokenExpirePeriod:3600
Id:f5d9ed92-3ef4-4094-8733-cb02805a03db-colors, Name:colors, InterfaceName:/ocsg_calendar/1/colors, MethodName:GET, TokenExpirePeriod:3600
Id:fcef5c79-f49d-4040-9de9-ae30535c36cc-test, Name:test, InterfaceName:/asdf/1/test, MethodName:GET, TokenExpirePeriod:3600
```

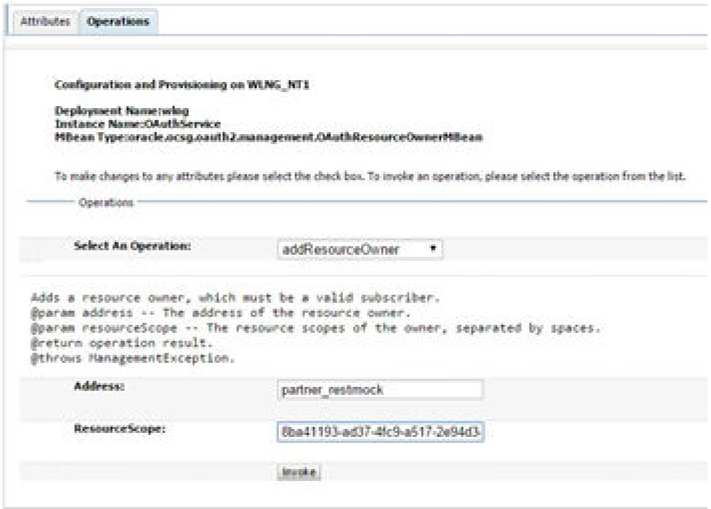
3. Navigate to **OCSG** > **AppServerx** > **Container Services** > **OAuthService** > **OAuthResourceOwnerMBean**..

Figure A-18   OAuthResourceOwnerMBean



4. Add the subscriber as resource owner of the application ID previously identified.

Figure A-19 Add Subscriber as Resource Owner



Configuration and Provisioning on WLNG_NT1

Deployment Name:wlng
Instance Name:OAuthService
MBean Type:oracle.ocsg.oauth2.management.OAuthResourceOwnerMBean

To make changes to any attributes please select the check box. To invoke an operation, please select the operation from the list.

Select An Operation: addResourceOwner

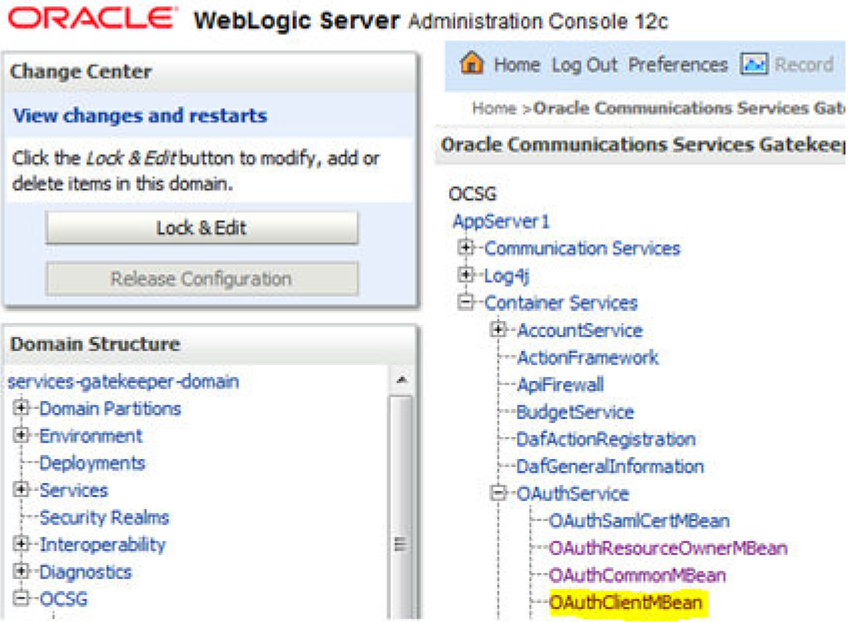
```
Adds a resource owner, which must be a valid subscriber.
@param address -- The address of the resource owner.
@param resourceScope -- The resource scopes of the owner, separated by spaces.
@return operation result.
@throws ManagementException.
```

Address: partner_restmock

ResourceScope: 8ba41193-ad37-4fc9-a517-2e94d3

5. Navigate to **OCSG** > **AppServerx** > **OAuthService** > **OAuthClientMBean**.

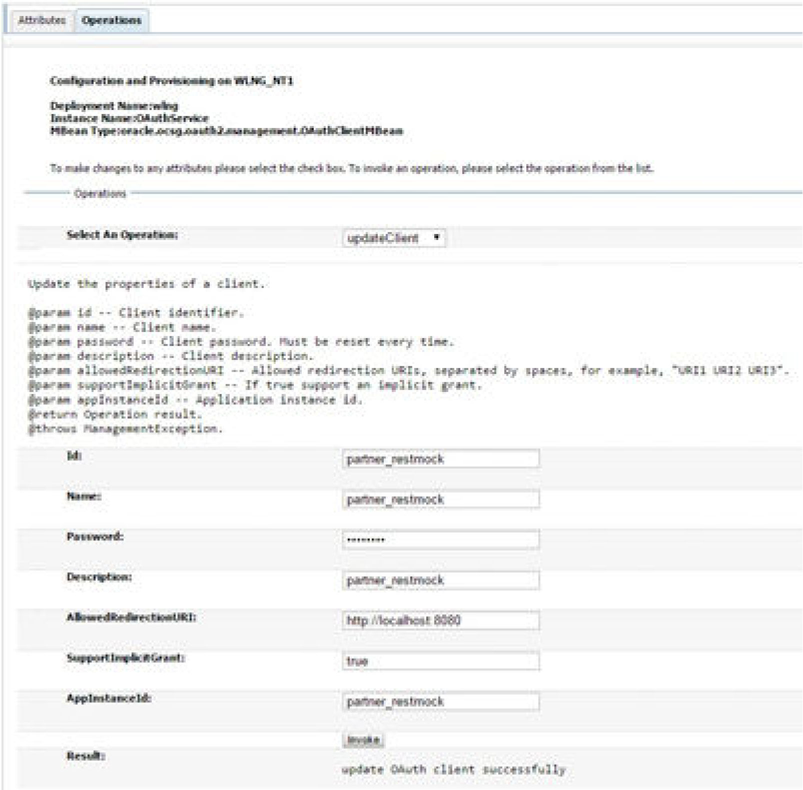
Figure A-20 OAuthClientMBean



Click the *Lock & Edit* button to modify, add or delete items in this domain.

6. Add the application traffic user as a client allowed redirect.

Figure A-21 Traffic User



Configuration and Provisioning on WLNG_NT1

Deployment Name:wlng
Instance Name:OAuthService
MBean Type:oracle.ocsg.oauth2.management.OAuthClientMBean

To make changes to any attributes please select the check box. To invoke an operation, please select the operation from the list.

Operations

Select An Operation: updateClient

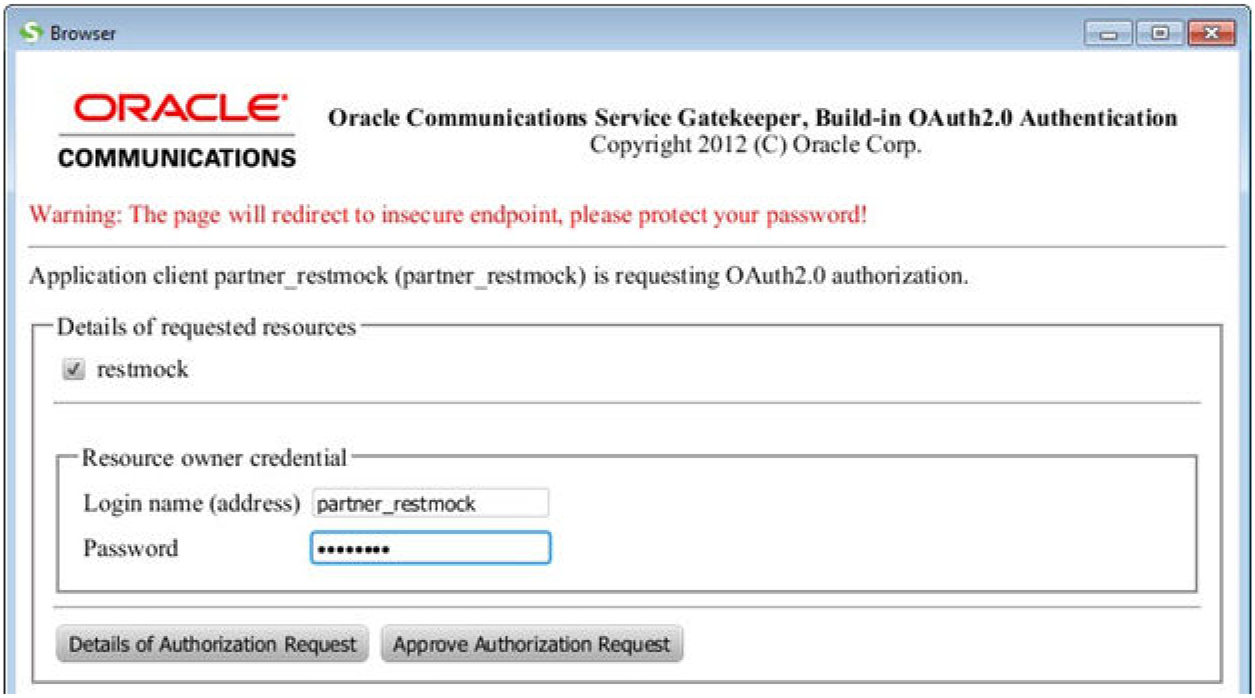
```
Update the properties of a client.

@param id -- Client identifier.
@param name -- Client name.
@param password -- Client password. Must be reset every time.
@param description -- Client description.
@param allowedRedirectionURI -- Allowed redirection URIs, separated by spaces, for example, "URI1 URI2 URI3".
@param supportImplicitGrant -- If true support an implicit grant.
@param appInstanceId -- Application instance id.
@return Operation result.
@throws ManagementException.
```

Id:	partner_restmock
Name:	partner_restmock
Password:	·········
Description:	partner_restmock
AllowedRedirectionURI:	http://localhost 8080
SupportImplicitGrant:	true
AppInstanceId:	partner_restmock

Result: update OAuth client successfully

Now when trying to access the application, this screen displays to request authentication.

Figure A-22 Authentication Request



Oracle Communications Service Gatekeeper, Build-in OAuth2.0 Authentication
Copyright 2012 (C) Oracle Corp.

Warning: The page will redirect to insecure endpoint, please protect your password!

Application client partner_restmock (partner_restmock) is requesting OAuth2.0 authorization.

Details of requested resources

☑ restmock

Resource owner credential

Login name (address) partner_restmock
Password •••••••••

Details of Authorization Request
Approve Authorization Request

Provisioning OCSG

This section describes how to provision the Oracle Communications Services Gatekeeper (OCSG).

Provisioning the OCSG involves the following five steps:

- On Boarding a Partner
- Register a Partner Account
- Approve (or Reject) a Partner Account
- Create a Partner Group
- Expose API URLs

Expose API URLs

The DSR API GW exposes 3GPP T8 resource URLs over the XMI subnet and port 10002 for HTTPS traffic. The deployment topology mandates using an external load balance, which should be owned and maintained by the customer. The load balancer is configured to send HTTPS traffic to all DSR API GW AppServers belonging to the site over XMI IP and port 10002.

Each resource URL format of the T8 API specification is prefixed with /<apiroot-{nidd/me/dt/ecr}>/v1.

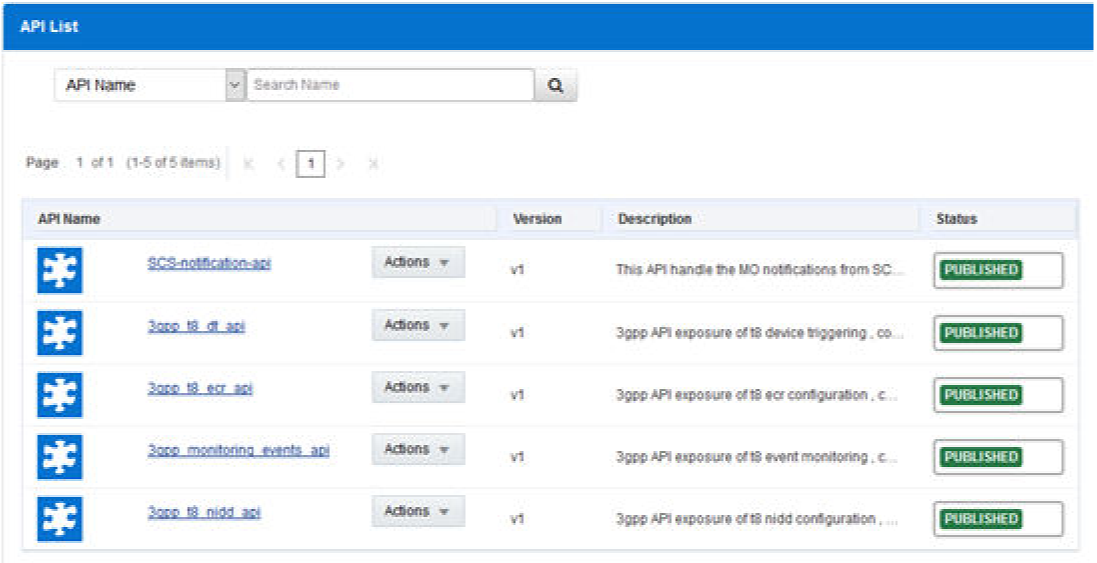
Apiroot is the property provided while configuring DSR API GW.

For example, the NIDD configuration URL will be where the apiroot provided is *operator1*:

```
/operator1-nidd/v1/3gpp_t8_nidd/v1/{scsAsId}/configurations
```

The T8 APIs for NIDD, Monitoring Events, Device Triggering, and ECR are defined on the Partner and API Management portal, which can be accessed using the operator account.

Figure A-23 Expose API URLs



API Name		Version	Description	Status
SCS-notification-api	Actions ▾	v1	This API handle the MO notifications from SC...	PUBLISHED
3gpp_t8_dt_api	Actions ▾	v1	3gpp API exposure of t8 device triggering , co...	PUBLISHED
3gpp_t8_ecr_api	Actions ▾	v1	3gpp API exposure of t8 ecr configuration , c...	PUBLISHED
3gpp_monitoring_events_api	Actions ▾	v1	3gpp API exposure of t8 event monitoring , c...	PUBLISHED
3gpp_t8_nidd_api	Actions ▾	v1	3gpp API exposure of t8 nidd configuration , ...	PUBLISHED

On Boarding a Partner

This procedure on-boards an SCS/AS into SCEF-OCSG.

1. Register a partner account.
2. Approve partner account creation.
3. Create a partner group.
4. Assign the partner to a specific group.
5. Create a partner application.
6. Approve the application creation request.
7. Set the traffic password for the application.

The procedures that follow in this section explain each step in more detail.

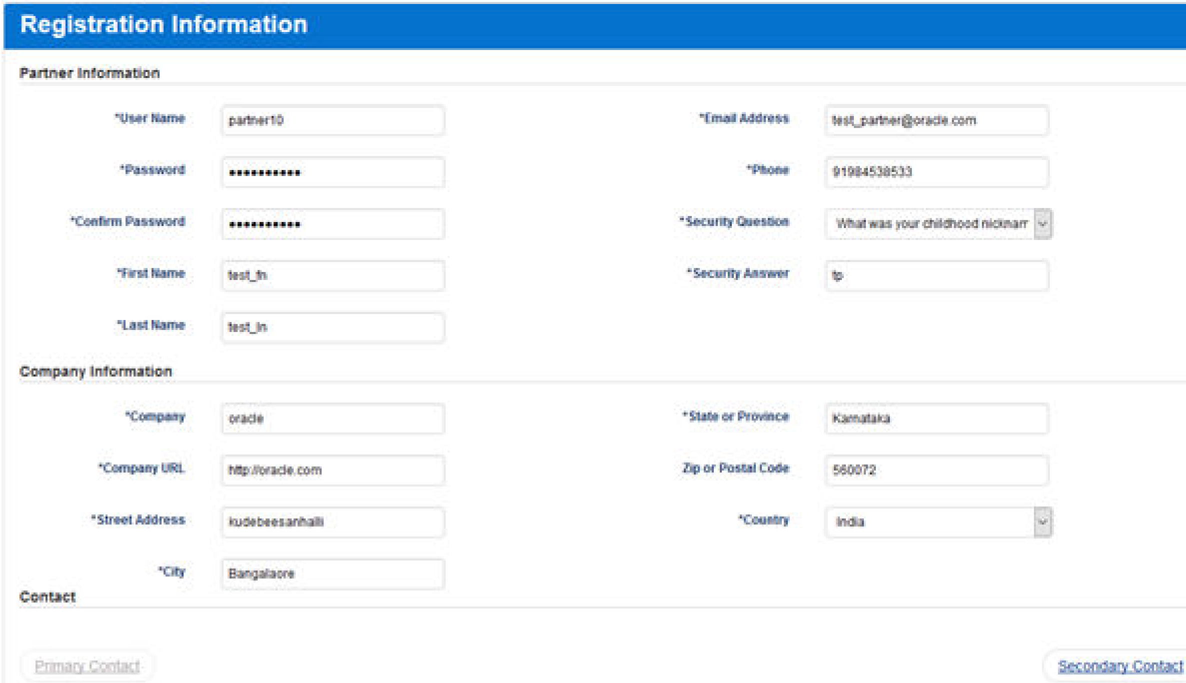
Register a Partner Account

A partner account can be registered using a self registration process or you can register an account for the partner. Both processes can also be done using a GUI interface or the REST interface. All procedures are described in this section.

Self Registration Using the GUI

1. Access the partner and API management portal at https://<AppServerx-XMI-IP>:9002/portal/partner/index/partnerLogin.html.
2. Register the account by clicking **Create new Account**.
3. Provide required details.
4. Mark the **Agreement** checkbox and click **Register**.

Figure A-24 Partner Self Registration



Registration Information

Partner Information

*User Name	partner10	*Email Address	test_partner@oracle.com
*Password	●●●●●●●●●●●	*Phone	91984538533
*Confirm Password	●●●●●●●●●●●	*Security Question	What was your childhood nicknam
*First Name	test_fn	*Security Answer	tp
*Last Name	test_ln		

Company Information

*Company	oracle	*State or Province	Karnataka
*Company URL	http://oracle.com	Zip or Postal Code	560072
*Street Address	kudebeesanhalli	*Country	India
*City	Bangalaore		

Contact

Self Registration Using REST

To self register as a partner, use the POST method from the /prm_pm_rest/services/prm_pr/services/register/Register/registers resource URL.

For more details on the request and response formats, see https://docs.oracle.com/communications/E81149_01/doc.70/e96582/resource_Partner_Account.html#resource_Partner_Account_registerSP_POST.

An example of a self registration request and response follow:

Request:

```
POST /prm_pm_rest/services/prm_pr/services/register/Register/registerSP HTTP/1.1
Host: 10.178.254.224:9001
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; rv:52.0) Gecko/20100101 Firefox/
52.0
Accept: application/json, text/javascript, */*; q=0.01
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/json
x-requested-with: XMLHttpRequest
Referer: http://10.178.254.224:9001/portal/partner/index/register.html
Content-Length: 650
Connection: keep-alive

{"registerSP":{"spInfo":
{"userName":"test_user1","emailAddr":"test_user1@oracle.com",
"password":"password123”,"phone":"91984538533","securityAnswerChoice":0,
"securityAnswer":"tp1","firstName":"test_fn","lastName":"test_ln","company":"orac
le",
"companyURL":"http://
oracle.com","stateOrProvince":"Karnataka","zipOrPostalCode":"560072",
"streetAddress":"kudebeesanhalli","city":"Bangalaore","country":"India",
"contacts":
```

```
[{"city":"Bangalore","contactTimeFrom":"09:00","contactTimeTo":"17:00",
"country":"India","emailAddress":"test_partner@oracle.com","firstName":"test_fn",
"lastName":"test_ln"}],"userType":"PRM_SP"}}}
```

Response:

```
HTTP/1.1 200 OK
Date: Thu, 30 Aug 2018 08:40:13 GMT
Content-Length: 25
Content-Type: application/json

{"registerSPResponse":{}}
```

Register an Account for a Partner Using the GUI

1. Access the partner and API management portal at https://<AppServerx-XMI-IP>:9002/portal/partner/index/partnerLogin.html.
2. Login with the operator account.
3. Click on the Partners tab.
4. Click **Create Partner Account**.
5. Provide required details.
6. Click **Create Partner**.

Register an Account for a Partner Using REST

To register a partner, use the POST method from the /portal/prm/prm_pm_rest/services/accountmanage/AccountManagement/createUser resource URL.

To authorize the request, use the operator username and password in the header of the request.

For more details on the request and response formats, see https://docs.oracle.com/communications/E81149_01/doc.70/e96582/resource_Partner_Manager_Account.html.

An example of a registration request and response follow:

Request:

```
POST /portal/prm/prm_pm_rest/services/accountmanage/AccountManagement/createUser
HTTP/1.1
Host: 10.75.244.188:9001
User-Agent: Mozilla/5.0 (Windows NT 6.1; Win64; x64; rv:57.0) Gecko/20100101
Firefox/57.0
Accept: application/json, text/javascript, */*; q=0.01
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://10.75.244.188:9001/portal/partner-manager/index/main.html
Content-Type: application/json
AuthorizationX: Basic
b3A6e0FFU310bnp2bmliSEpPbWNBdlo5QXZ0d3pzSFoxZVBBZnhNWEY3eG9CZ1kzci9ZPQ==
X-Requested-With: XMLHttpRequest
Content-Length: 671
Cookie:
ADMINCONSOLESESSION=645Fdnqjm2LmBV4xf2yqeM9Tk3YngS0_q7GwCrZMWfNPiXsLe9xI!-6760276
30
Connection: keep-alive
```

```
{"createUser":{"userInfo":
{"userName":"testuser","emailAddr":"user1@usercompany.com",
"password":"password1234","phone":"9845398765","secureityAnswerChoice":0,
"secureityAnswer":"user1","firstName":"userfirstname","lastName":"userlastname",
"company":"usercompany","companyURL":"http://
usercompany.com","stateOrProvince":"state1",
"zipOrPostalCode":"560037","streetAddress":"street1","city":"bangalore","country"
:"India",
"contacts":
[{"city":"bangalore","contactTimeFrom":"08:00","contactTimeTo":"17:00",
"country":"India","emailAddress":"user1@usercompany.com","firstName":"userfirstna
me",
"lastName":"userlastname"}],"userType":"PRM_SP"}}}
```

Response:

```
HTTP/1.1 200 OK
Content-Length: 0Server: Jetty(8.0.1.0)
```

Approve (or Reject) a Partner Account

When a partner account is created using the self-registration process on the partner portal, the operator needs to approve (or reject) the request. This can be done using either the GUI interface or REST interface.

Approve a Partner Account Using the GUI

1. Access the partner and API management portal at https://<AppServerx-XMI-IP>:9002/portal/partner/index/partnerLogin.html.

2. Login with the operator account.

3. Click on the red circle on top right corner.

   The screen displays all requests pending an approval.

4. Select the partner request and right click to View Details.

5. Review the details and approve or reject the request.

Approve a Partner Account Using REST

The partner approval using REST is a three step process involving getting the notification, approving the request, and updating the notification status.

1. To get the notification, use the GET method from the /portal/prm/prm_pm_rest/services/partner_manager/notification/PartnerManagerNotification/listNotificationsByStatus/UNREAD resource URL.

   To authorize the request, use the operator username and password in the header of the request.

   For more details on the request and response formats, see https://docs.oracle.com/communications/E81149_01/doc.70/e96582/resource_Partner_Manager_Notification.html.

   An example of the notification request and response follow:

   Request:

```
GET /portal/prm/prm_pm_rest/services/partner_manager/notification/
PartnerManagerNotification/listNotificationsByStatus/UNREAD HTTP/1.1
Host: 10.75.244.188:9001
User-Agent: Mozilla/5.0 (Windows NT 6.1; Win64; x64; rv:58.0) Gecko/20100101
Firefox/58.0
Accept: application/json, text/javascript, */*; q=0.01
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://10.75.244.188:9001/portal/partner/index/partnerMain.html
AuthorizationX: Basic
cGFydG5lcjE6e0FFU301Z2Q1UFpwcDZVTmJHdkxrRnBPSXpuMDNMNGxNeDRDZUpCcnBQTjJYaXZrP
Q==
X-Requested-With: XMLHttpRequest
Connection: keep-alive
```

Response:

```
HTTP/1.1 200 OK

Date: Wed, 31 Jan 2018 05:50:57 GMT
Content-Length: 9447
Content-Type: application/json
X-FRAME-OPTIONS: DENY

{"ListNotificationsByStatusResponse":{"return":
[{"id":"113e1904-8419-4cd2-958a-659a950aba6c",
"content":"Partner Create Application
Task","date":"01/31/2018","receiver":"PM",
"sender":"partner1","senderCompany":"oracle","redirectObject":
{"type":"ns4:application",
"notificationId":"113e1904-8419-4cd2-958a-659a950aba6c","applicationID":"fa78
9e8d-9b97-455a-b556-491ed5253da5",
"applicationName":"mmitestapp1","partnerName":"partner1","partnerCompany":"or
acle",
"description":"mmi testing application 1","applicationAPIs":
[{"apiDisplayName":"3gpp_t8_nidd",
"apiName":"3gpp_t8_nidd","accessURL":"http://10.10.10.9:8001/3gpp_t8_nidd/
v1\nhttps://10.10.10.9:7002/3gpp_t8_nidd/v1",
"apiVersion":"v1","apiDescription":"dsr nidd test api
","needReadContract":false}],
"trafficUser":"partner1_mmitestapp1","trafficPassword":
"{AES}UEYOH2WTIo5Kwgodl7uFtmMCr5oLLBTz3H6jQ4jK5j9AoPdBhjYJiqtqpvB86IuYwNybnTP
+x3hTAgn/UTLrUw==",
"submitDate":"2018-01-31-05:00","effectiveFrom":"2018-01-31-05:00","effective
To":"2018-03-28-04:00",
"status":"CREATE PENDING APPROVAL","lockStatus":"UNLOCKED","quota":{"days":
1,"limitExceedOK":true,
"qtaLimit":10000},"rate":{"reqLimit":10,"timePeriod":1},"icon":"expressive/
app.png"},
"status":"UNREAD"},{"id":"9bdc365a-40fd-496c-9cff-
dd1ec805dd18","content":"Partner Delete Pending Application Task",
"date":"01/31/2018","receiver":"PM","sender":"partner1","senderCompany":"orac
le",
"redirectObject":
{"type":"ns4:application","notificationId":"9bdc365a-40fd-496c-9cff-
dd1ec805dd18",
"applicationID":"60c5e194-ad46-4935-
ba78-4777bab65eaf","applicationName":"mmitestapp1",
"partnerName":"partner1","partnerCompany":"oracle","description":"mmi
testing application 1",
"applicationAPIs":[{"apiDisplayName":"3gpp_t8_nidd","apiName":"3gpp_t8_nidd",
"accessURL":"http://10.10.10.9:8001/3gpp_t8_nidd/v1\nhttps://
10.10.10.9:7002/3gpp_t8_nidd/v1",
"apiVersion":"v1","apiDescription":"dsr nidd test api
","applicationMethodSLAs":[{"methodName":"",
"interfaceName":"57cf5ce0-a175-43d2-a1f4-53fb3ebae851",
"quota":{"days":0,"limitExceedOK":false,"qtaLimit":0},"rate":{"reqLimit":
0,"timePeriod":0},
"methodGuarantee":{"reqLimitGuarantee":0,"timePeriodGuarantee":
0}}],"needReadContract":false}],
"trafficUser":"partner1_mmitestapp1","submitDate":"2018-01-31-05:00","effecti
veFrom":"2018-01-31-05:00",
"effectiveTo":"2018-04-24-04:00","status":"CREATE PENDING
APPROVAL","lockStatus":"UNLOCKED",
"quota":{"days":1,"limitExceedOK":true,"qtaLimit":10000},"rate":{"reqLimit":
10,"timePeriod":1},
"icon":"expressive/app.png"},"status":"UNREAD"},{"id":"b012db05-caac-421f-
ad3b-95d5350fc72a",
"content":"Partner Create Application
Task","date":"01/31/2018","receiver":"PM","sender":"partner1",
"senderCompany":"oracle","redirectObject":{"type":"ns4:application",
"notificationId":"b012db05-caac-421f-
ad3b-95d5350fc72a","applicationID":"60c5e194-ad46-4935-ba78-4777bab65eaf",
"applicationName":"mmitestapp1","partnerName":"partner1","partnerCompany":"or
acle",
"description":"mmi testing application 1","applicationAPIs":
[{"apiDisplayName":"3gpp_t8_nidd",
"apiName":"3gpp_t8_nidd","accessURL":"http://10.10.10.9:8001/3gpp_t8_nidd/
v1\nhttps://10.10.10.9:7002/3gpp_t8_nidd/v1",
"apiVersion":"v1","apiDescription":"dsr nidd test api
","needReadContract":false}],
"trafficUser":"partner1_mmitestapp1","trafficPassword":
"{AES}pqXqICn4W4IJq/u8kitcbn8w82RJKQKZbI2WUaV9KzKOMOcxSUQhU1vd/
9hEsZcDBwqjP93HllvhoU41UwOCaw==",
"submitDate":"2018-01-31-05:00","effectiveFrom":"2018-01-31-05:00","effective
To":"2018-04-24-04:00",
"status":"CREATE PENDING APPROVAL","lockStatus":"UNLOCKED","quota":{"days":
1,"limitExceedOK":true,"qtaLimit":10000},
"rate":{"reqLimit":10,"timePeriod":1},"icon":"expressive/
app.png"},"status":"UNREAD"}]}}
```

2. To approve the partner account creation request, use the POST method from the /portal/prm/prm_pm_rest/services/accountmanage/AccountManagement/approve resource URL.

   Note the account creation notification ID from the previous step.

   To authorize the request, use the operator username and password in the header of the request.

   For more details on the request and response formats, see https://docs.oracle.com/communications/E81149_01/doc.70/e96582/resource_Partner_Manager_Account.html.

   An example of an approval request and response follow:

   Request:

   ```
   POST /portal/prm/prm_pm_rest/services/accountmanage/AccountManagement/
   approve HTTP/1.1
   Host: 10.178.254.224:9001
   User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; rv:52.0) Gecko/20100101
   Firefox/52.0
   Accept: application/json, text/javascript, */*; q=0.01
   ```

```
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/json
AuthorizationX: Basic
b3JhY2xlb3AxOntBRVN9ZnBTRHBaeWw0dGRqR0lob3c2SzZFOThGT2tKbGxyZXB5Y3RrbUx0MmhPW
T0=
x-requested-with: XMLHttpRequest
Referer: http://10.178.254.224:9001/portal/partner-manager/index/main.html
Content-Length: 935
Connection: keep-alive

{"approve":{"userInfo":
{"city":"Bangalaore","company":"oracle","companyURL":"http://oracle.com",
"contacts":
[{"city":"Bangalore","contactTimeFrom":"09:00","contactTimeTo":"17:00","count
ry":"India",
"emailAddress":"test_partner@oracle.com","firstName":"test_fn","lastName":"te
st_ln","phone":null}],
"country":"India","emailAddr":"test_user1@oracle.com","financial":
{"bankAccountNumber":"",
"bankAddress":"","bankName":"","bankRoutingNumber":"","city":"","country":"",
"invoiceTo":"",
"referenceAccount":"","stateOrProvince":"","taxID":"","zipOrPostalCode":""},"
firstName":"test_fn",
"lastName":"test_ln","password":"{AES}mhY96ryJA82JHEiChWJo3rDczngO/
YuMYN5tSxH4Oko=","phone":"91984538533",
"securityAnswer":"tp1","securityAnswerChoice":"0","stateOrProvince":"Karnatak
a","status":0,
"streetAddress":"kudebeesanhalli","userName":"test_user1","zipOrPostalCode":"
560072",
"userType":"PRM_SP","notificationId":"b1e706fb-4436-401f-972c-99821f052805"}}
}
```

Response:

```
HTTP/1.1 200 OK
Date: Thu, 30 Aug 2018 08:42:48 GMT
Content-Length: 22
Content-Type: application/json
X-Frame-Options: DENY

{"approveResponse":{}}
```

3. Once the request has been approved or rejected, change the notification status, using the POST method from the /portal/prm/prm_pm_rest/services/partner_manager/notification/PartnerManagerNotification/updateNotificationStatus resource URL.

   To authorize the request, use the operator username and password in the header of the request.

   For more details on the request and response formats, see https://docs.oracle.com/communications/E81149_01/doc.70/e96582/resource_Partner_Manager_Notification.html.

   An example of an updated notification status request and response follow:

   Request:

```
POST /portal/prm/prm_pm_rest/services/partner_manager/notification/
PartnerManagerNotification/updateNotificationStatus HTTP/1.1
Host: 10.178.254.224:9001
```

```
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; rv:52.0) Gecko/20100101
Firefox/52.0
Accept: application/json, text/javascript, */*; q=0.01
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/json
AuthorizationX: Basic
b3JhY2xlb3AxOntBRVN9ZnBTRHBaeWw0dGRqR0lob3c2SzZFOThGT2tKbGxyZXB5Y3RrbUx0MmhPW
T0=
x-requested-with: XMLHttpRequest
Referer: http://10.178.254.224:9001/portal/partner-manager/index/main.html
Content-Length: 102
Connection: keep-alive

{"updateNotificationStatus":
{"notificationId":"b1e706fb-4436-401f-972c-99821f052805","status":"READ"}}
```

Response:

```
HTTP/1.1 200 OK
Date: Thu, 30 Aug 2018 08:42:48 GMT
Content-Length: 39
Content-Type: application/json
X-Frame-Options: DENY

{"updateNotificationStatusResponse":{}}
```

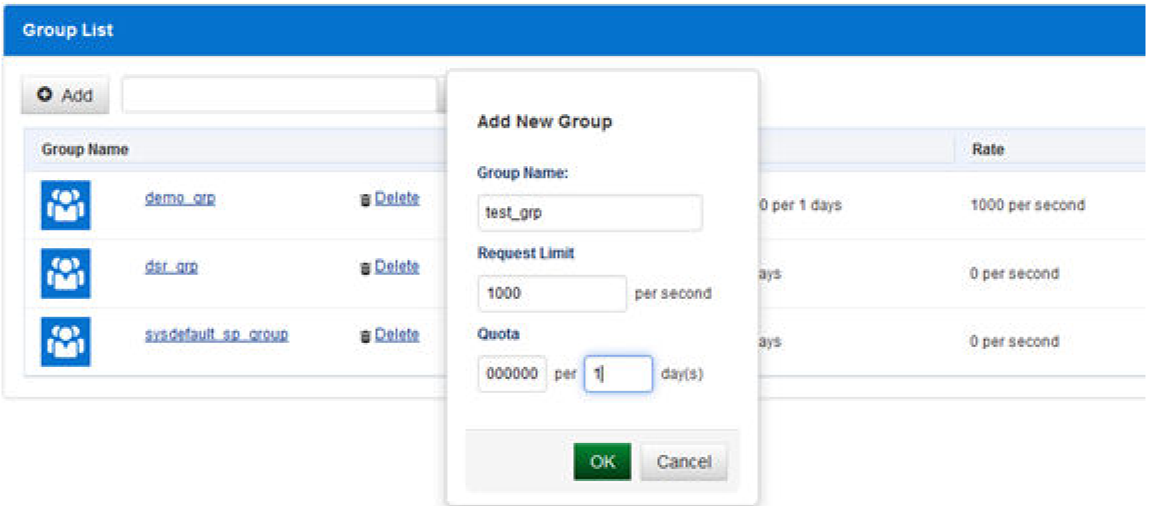
Create a Partner Group

Once a partner account is created and approved, it is added to a group. To create a group, use either the GUI interface or the REST interface. Both procedures are described in this section.

Create a Partner Group Using the GUI

1. Access the partner and API management portal at https://<AppServerx-XMI-IP>:9002/portal/partner/index/partnerLogin.html.
2. Login with the operator account.
3. Click on the Partners tab.
4. Click **Groups**.
5. Click **Add**.
6. Type the *Group Name*, *Request Limit*, and *Quota* allowed for the partner group.
7. Click **OK**.

Figure A-25 Create Partner Group



Group Name: test_grp

Request Limit 1000 per second

Quota 000000 per 1 day(s)

Create a Partner Group Using REST

To create a partner group, use the POST method from the /portal/prm/prm_pm_rest/services/partner_manager/group/PartnerManagerSlaGroup/createServiceProviderGroup resource URL.

To authorize the request, use the operator username and password in the header of the request.

For more details on the request and response formats, see https://docs.oracle.com/communications/E81149_01/doc.70/e96582/resource_Group.html.

An example to create a partner group request and response follow:

Request:

```
POST /portal/prm/prm_pm_rest/services/partner_manager/group/
PartnerManagerSlaGroup/createServiceProviderGroup HTTP/1.1
Host: 10.75.244.188:9001
User-Agent: Mozilla/5.0 (Windows NT 6.1; Win64; x64; rv:57.0) Gecko/20100101
Firefox/57.0
Accept: application/json, text/javascript, */*; q=0.01
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://10.75.244.188:9001/portal/partner-manager/index/main.html
Content-Type: application/json
AuthorizationX: Basic
b3A6e0FFU31UT0RyWXN5dE0yMWJMZ1VndVJTVTJWbk1XV3FSaFFNRlBLRDhaVG1RbHdJPQ==
X-Requested-With: XMLHttpRequest
Content-Length: 143
Cookie:
ADMINCONSOLESESSION=645Fdnqjm2LmBV4xf2yqeM9Tk3YngS0_q7GwCrZMWfNPiXsLe9xI!-6760276
30
Connection: keep-alive

{"createServiceProviderGroup":{"groupName":"user2_group","rate":
{"reqLimit":"10000","timePeriod":1},"quota":{"qtaLimit":"1000000","days":"1"}}}
```

Response:

```
HTTP/1.1 200 OK
Date: Tue, 30 Jan 2018 06:46:09 GMT
```

```
Content-Length: 41
Content-Type: application/json
X-FRAME-OPTIONS: DENY

{"createServiceProviderGroupResponse":{}}
```

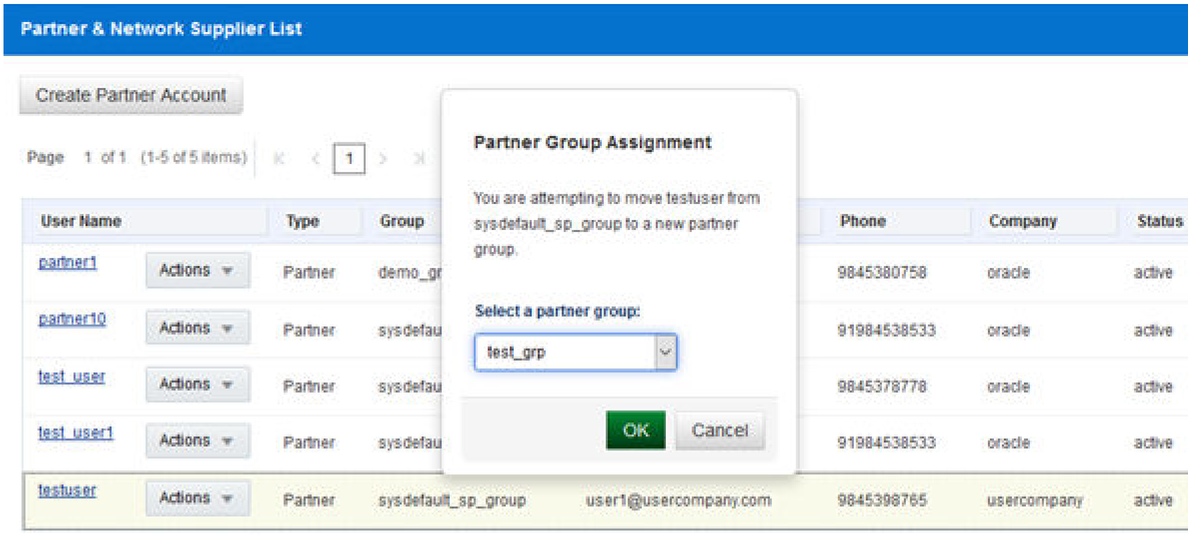
Assign a Partner to a Group

After a partner account and group have been created, the operator needs to add the account to the group. This can be done using either the GUI interface or REST interface.

Assign a Partner to a Group Using the GUI

1. Access the partner and API management portal at https://<AppServerx-XMI-IP>:9002/portal/partner/index/partnerLogin.html.
2. Login with the operator account.
3. Click on the Partners tab.
4. Select *Assign Group* from the Actions options.
5. Select the group to which the partner is to be assigned.
6. Click **OK**.

Figure A-26 Add Partner to Group



You are attempting to move testuser from sysdefault_sp_group to a new partner group.

Select a partner group: test_grp

User Name		Type	Group		Phone	Company	Status
partner1	Actions	Partner	demo_gr		9845380758	oracle	active
partner10	Actions	Partner	sysdefau		91984538533	oracle	active
test_user	Actions	Partner	sysdefau		9845378778	oracle	active
test_user1	Actions	Partner	sysdefau		91984538533	oracle	active
testuser	Actions	Partner	sysdefault_sp_group	user1@usercompany.com	9845398765	usercompany	active

Assign a Partner to a Group Using REST

To assign a partner to a group, use the POST method from the /portal/prm/prm_pm_rest/services/partner_manager/group/PartnerManagerSlaGroup/confirmMovePartnerToGroup resource URL.

To authorize the request, use the operator username and password in the header of the request.

For more details on the request and response formats, see https://docs.oracle.com/communications/E81149_01/doc.70/e96582/resource_Group.html.

An example of how to add a partner to a group request and response follow:

Request:

```
POST /portal/prm/prm_pm_rest/services/partner_manager/group/
PartnerManagerSlaGroup/confirmMovePartnerToGroup HTTP/1.1
Host: 10.75.244.188:9001
User-Agent: Mozilla/5.0 (Windows NT 6.1; Win64; x64; rv:57.0) Gecko/20100101
Firefox/57.0
Accept: application/json, text/javascript, */*; q=0.01
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://10.75.244.188:9001/portal/partner-manager/index/main.html
Content-Type: application/json
AuthorizationX: Basic
b3A6e0FFU31UT0RyWXN5dE0yMWJMZ1VndVJTVTJWbk1XV3FSaFFNRlBLRDhaVG1RbHdJPQ==
X-Requested-With: XMLHttpRequest
Content-Length: 107
Cookie:
ADMINCONSOLESESSION=645Fdnqjm2LmBV4xf2yqeM9Tk3YngS0_q7GwCrZMWfNPiXsLe9xI!-6760276
30
Connection: keep-alive

{"confirmMovePartnerToGroup":
{"partnerName":"testuser","newGroupName":"user1_group","action":"EXPAND_SLA"}}
```

Response:

```
HTTP/1.1 200 OK
Date: Tue, 30 Jan 2018 06:50:24 GMT
Content-Length: 40
Content-Type: application/json
X-FRAME-OPTIONS: DENY

{"confirmMovePartnerToGroupResponse":{}}
```

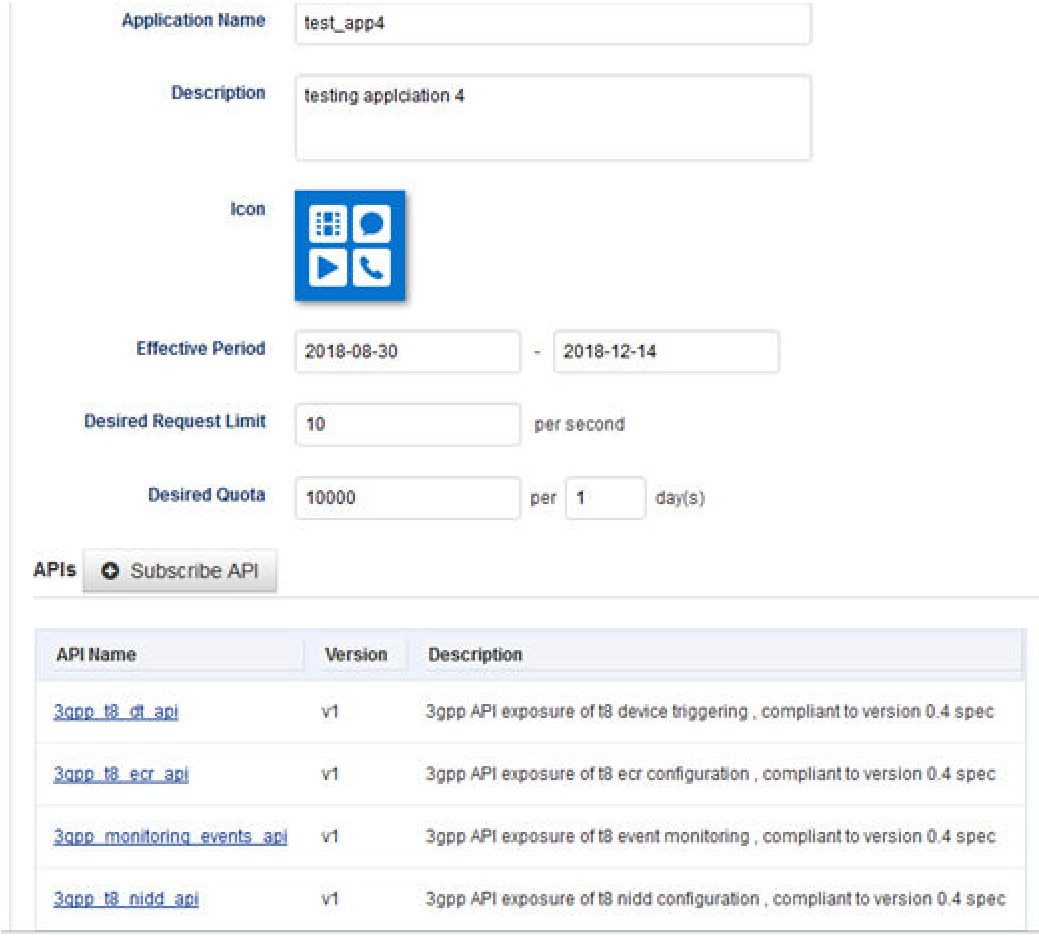
Create a Partner Application

To access APIs exposed by the DSR API GW, a logical unit partner application is created by the partner. This can be done using either the GUI interface or REST interface.

Create a Partner Application Using the GUI

1. Access the partner and API management portal at https://<AppServerx-XMI-IP>:9002/portal/partner/index/partnerLogin.html.
2. Log into the portal using the partner account.
3. Click on the Applications tab.
4. Click **Add**.
5. Provide required details and subscribe to the required T* APIs.

Figure A-27 Create Partner Application



Application Name	test_app4
Description	testing applciation 4
Icon	
Effective Period	2018-08-30 - 2018-12-14
Desired Request Limit	10 per second
Desired Quota	10000 per 1 day(s)

API Name	Version	Description
3gpp_t8_dt_api	v1	3gpp API exposure of t8 device triggering , compliant to version 0.4 spec
3gpp_t8_ecr_api	v1	3gpp API exposure of t8 ecr configuration , compliant to version 0.4 spec
3gpp_monitoring_events_api	v1	3gpp API exposure of t8 event monitoring , compliant to version 0.4 spec
3gpp_t8_nidd_api	v1	3gpp API exposure of t8 nidd configuration , compliant to version 0.4 spec

Create a Partner Application Using REST

To create a partner application, use the POST method from the /portal/prm/prm_pm_rest/services/prm_pm/services/partner/application/PartnerApplication/createApplication resource URL.

To authorize the request, use the partner username and password in the header of the request.

For more details on the request and response formats, see https://docs.oracle.com/communications/E81149_01/doc.70/e96582/resource_Partner_Application.html.

An example of how to create a partner application request and response follow:

Request:

```
POST /portal/prm/prm_pm_rest/services/prm_pm/services/partner/application/
PartnerApplication/createApplication HTTP/1.1
Host: 10.178.254.224:9001
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; rv:52.0) Gecko/20100101 Firefox/
52.0
Accept: application/json, text/javascript, */*; q=0.01
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/json
AuthorizationX: Basic
cGFydG5lcjE6e0FFU31GYjZmNWt2UmRRZEVyU2FzR08zL0kwYlc0aG1sdVE3SDZ4YktIN0pkZ2w4PQ==
x-requested-with: XMLHttpRequest
Referer: http://10.178.254.224:9001/portal/partner/index/partnerMain.html
Content-Length: 545
Connection: keep-alive

{"createApplication":{"application":
{"applicationName":"test_app_3","description":"testing application 3",
"trafficUser":"partner1_test_app1_user","trafficPassword":"password1234","effecti
veFrom":"2018-08-31",
"effectiveTo":"2019-04-19","partnerName":"partner1","quota":
{"days":"1","limitExceedOK":true,"qtaLimit":"100000"},
"rate":{"reqLimit":"10","timePeriod":"1"},"applicationAPIs":
[{"apiName":"80cec996-02a1-40ec-bb66-fece0b86b317"},
{"apiName":"36b271ee-0d37-46b7-96c2-72deec416bee"},{"apiName":"9d5c8500-
d5cc-4fda-8d8c-83bec26141a7"},
{"apiName":"19a44504-fb39-477f-b920-49f5bf85f443"}],"icon":"expressive/
app.png"}}}
```

Response:

```
HTTP/1.1 200 OK
Date: Thu, 30 Aug 2018 09:56:55 GMT
Content-Length: 86
Content-Type: application/json
X-Frame-Options: DENY

{"createApplicationResponse":{"applicationID":"ec8fa535-f45c-42fd-a4b6-
bcf1a584be82"}}
```

API Names are IDs generated by DSR API GW dynamically. A partner can retrieve the API IDs by using REST from the /portal/prm/prm_pm_rest/services/prm_pm/services/partner/api/PartnerApi/getAPIs resource URL.

Refer to https://docs.oracle.com/communications/E81149_01/doc.70/e96582/resource_Partner_API.html for more information.

Approve (or Reject) an Application Creation

When a partner application is created, the operator needs to approve (or reject) the request. This can be done using either the GUI interface or REST interface.

Approve a Partner Application Using the GUI

1. Access the partner and API management portal at https://<AppServerx-XMI-IP>:9002/portal/partner/index/partnerLogin.html.
2. Login with the operator account.
3. Click on the red circle on top right corner.

   The screen displays all requests pending an approval.
4. Select the application creation request and right click to View Details.
5. Review the details and approve or reject the application request.

Approve a Partner Application Using

The partner application approval using REST is a three step process involving getting the notification, approving the request, and updating the notification status.

1. To get the notification, use the GET method from the /portal/prm/prm_pm_rest/services/partner_manager/notification/PartnerManagerNotification/listNotificationsByStatus/UNREAD resource URL.

   To authorize the request, use the operator username and password in the header of the request.

   For more details on the request and response formats, see https://docs.oracle.com/communications/E81149_01/doc.70/e96582/resource_Partner_Manager_Notification.html.

   An example of the notification request and response follow:

   Request:

   ```
   GET /portal/prm/prm_pm_rest/services/partner_manager/notification/
   PartnerManagerNotification/listNotificationsByStatus/UNREAD HTTP/1.1
   Host: 10.75.244.188:9001
   User-Agent: Mozilla/5.0 (Windows NT 6.1; Win64; x64; rv:58.0) Gecko/20100101
   Firefox/58.0
   Accept: application/json, text/javascript, */*; q=0.01
   Accept-Language: en-US,en;q=0.5
   Accept-Encoding: gzip, deflate
   Referer: http://10.75.244.188:9001/portal/partner/index/partnerMain.html
   AuthorizationX: Basic
   cGFydG5lcjE6e0FFU301Z2Q1UFpwcDZVTmJHdkxrRnBPSXpuMDNMNGxNeDRDZUpCcnBQTjJYaXZrP
   Q==
   X-Requested-With: XMLHttpRequest
   Connection: keep-alive
   ```

   Response:

   ```
   HTTP/1.1 200 OK

   Date: Wed, 31 Jan 2018 05:50:57 GMT
   Content-Length: 9447
   Content-Type: application/json
   X-FRAME-OPTIONS: DENY

   {"ListNotificationsByStatusResponse":{"return":
   [{"id":"113e1904-8419-4cd2-958a-659a950aba6c",
   "content":"Partner Create Application
   Task","date":"01/31/2018","receiver":"PM",
   "sender":"partner1","senderCompany":"oracle","redirectObject":
   {"type":"ns4:application",
   "notificationId":"113e1904-8419-4cd2-958a-659a950aba6c","applicationID":"fa78
   9e8d-9b97-455a-b556-491ed5253da5",
   "applicationName":"mmitestapp1","partnerName":"partner1","partnerCompany":"or
   acle",
   "description":"mmi testing application 1","applicationAPIs":
   [{"apiDisplayName":"3gpp_t8_nidd",
   "apiName":"3gpp_t8_nidd","accessURL":"http://10.10.10.9:8001/3gpp_t8_nidd/
   v1\nhttps://10.10.10.9:7002/3gpp_t8_nidd/v1",
   "apiVersion":"v1","apiDescription":"dsr nidd test api
   ","needReadContract":false}],
   "trafficUser":"partner1_mmitestapp1","trafficPassword":
   "{AES}UEYOH2WTIo5Kwgodl7uFtmMCr5oLLBTz3H6jQ4jK5j9AoPdBhjYJiqtqpvB86IuYwNybnTP
   +x3hTAgn/UTLrUw==",
   "submitDate":"2018-01-31-05:00","effectiveFrom":"2018-01-31-05:00","effective
   To":"2018-03-28-04:00",
   "status":"CREATE PENDING APPROVAL","lockStatus":"UNLOCKED","quota":{"days":
   1,"limitExceedOK":true,
   ```

```
"qtaLimit":10000},"rate":{"reqLimit":10,"timePeriod":1},"icon":"expressive/
app.png"},
"status":"UNREAD"},{"id":"9bdc365a-40fd-496c-9cff-
dd1ec805dd18","content":"Partner Delete Pending Application Task",
"date":"01/31/2018","receiver":"PM","sender":"partner1","senderCompany":"orac
le",
"redirectObject":
{"type":"ns4:application","notificationId":"9bdc365a-40fd-496c-9cff-
dd1ec805dd18",
"applicationID":"60c5e194-ad46-4935-
ba78-4777bab65eaf","applicationName":"mmitestapp1",
"partnerName":"partner1","partnerCompany":"oracle","description":"mmi
testing application 1",
"applicationAPIs":[{"apiDisplayName":"3gpp_t8_nidd","apiName":"3gpp_t8_nidd",
"accessURL":"http://10.10.10.9:8001/3gpp_t8_nidd/v1\nhttps://
10.10.10.9:7002/3gpp_t8_nidd/v1",
"apiVersion":"v1","apiDescription":"dsr nidd test api
","applicationMethodSLAs":[{"methodName":"",
"interfaceName":"57cf5ce0-a175-43d2-a1f4-53fb3ebae851",
"quota":{"days":0,"limitExceedOK":false,"qtaLimit":0},"rate":{"reqLimit":
0,"timePeriod":0},
"methodGuarantee":{"reqLimitGuarantee":0,"timePeriodGuarantee":
0}}],"needReadContract":false}],
"trafficUser":"partner1_mmitestapp1","submitDate":"2018-01-31-05:00","effecti
veFrom":"2018-01-31-05:00",
"effectiveTo":"2018-04-24-04:00","status":"CREATE PENDING
APPROVAL","lockStatus":"UNLOCKED",
"quota":{"days":1,"limitExceedOK":true,"qtaLimit":10000},"rate":{"reqLimit":
10,"timePeriod":1},
"icon":"expressive/app.png"},"status":"UNREAD"},{"id":"b012db05-caac-421f-
ad3b-95d5350fc72a",
"content":"Partner Create Application
Task","date":"01/31/2018","receiver":"PM","sender":"partner1",
"senderCompany":"oracle","redirectObject":{"type":"ns4:application",
"notificationId":"b012db05-caac-421f-
ad3b-95d5350fc72a","applicationID":"60c5e194-ad46-4935-ba78-4777bab65eaf",
"applicationName":"mmitestapp1","partnerName":"partner1","partnerCompany":"or
acle",
"description":"mmi testing application 1","applicationAPIs":
[{"apiDisplayName":"3gpp_t8_nidd",
"apiName":"3gpp_t8_nidd","accessURL":"http://10.10.10.9:8001/3gpp_t8_nidd/
v1\nhttps://10.10.10.9:7002/3gpp_t8_nidd/v1",
"apiVersion":"v1","apiDescription":"dsr nidd test api
","needReadContract":false}],
"trafficUser":"partner1_mmitestapp1","trafficPassword":
"{AES}pqXqICn4W4IJq/u8kitcbn8w82RJKQKZbI2WUaV9KzKOMOcxSUQhU1vd/
9hEsZcDBwqjP93HllvhoU41UwOCaw==",
"submitDate":"2018-01-31-05:00","effectiveFrom":"2018-01-31-05:00","effective
To":"2018-04-24-04:00",
"status":"CREATE PENDING APPROVAL","lockStatus":"UNLOCKED","quota":{"days":
1,"limitExceedOK":true,"qtaLimit":10000},
"rate":{"reqLimit":10,"timePeriod":1},"icon":"expressive/
app.png"},"status":"UNREAD"}]}}
```

2. To approve the partner application creation request, use the POST method from the /portal/prm/prm_pm_rest/services/partner_manager/application/PartnerManagerApplication/updateCurrentSlaForApprove resource URL.

   Note the application creation notification ID from the previous step.

   To authorize the request, use the operator username and password in the header of the request.

For more details on the request and response formats, see https://docs.oracle.com/communications/E81149_01/doc.70/e96582/resource_Partner_Manager_Application.html.

An example of an approval request and response follow:

Request:

```
POST /portal/prm/prm_pm_rest/services/partner_manager/application/
PartnerManagerApplication/updateCurrentSlaForApprove HTTP/1.1
Host: 10.75.244.188:9001
User-Agent: Mozilla/5.0 (Windows NT 6.1; Win64; x64; rv:58.0) Gecko/20100101
Firefox/58.0
Accept: application/json, text/javascript, */*; q=0.01
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://10.75.244.188:9001/portal/partner-manager/index/main.html
Content-Type: application/json
AuthorizationX: Basic
b3A6e0FFU303cnFtSmc2MGNSMW95S3NvSkZ1V01VOTZ5MlI1cXp6alhCdHNTRkVZTGRvPQ==
X-Requested-With: XMLHttpRequest
Content-Length: 611
Connection: keep-alive

{"updateCurrentSlaForApprove":{"application":
{"notificationId":"92b78c6f-2f01-4ed6-8d01-75c43118d53e",
"applicationID":"b07e1e8e-0f85-4bdc-9422-1c4b88bbb51b","applicationName":"mmi
2","partnerName":"partner1",
"partnerCompany":"oracle","description":"mmi
2","trafficUser":"partner1_mmi2",
"trafficPassword":"{AES}XE1iY7gDD9sq3o7Ug1WAi
+dgAXmnYu7LsYsCUTQYvbQ=","submitDate":"2018-01-31-05:00",
"effectiveFrom":"2018-01-31-05:00","effectiveTo":"2018-04-10-04:00","status":
"CREATE PENDING APPROVAL",
"lockStatus":"UNLOCKED","quota":{"qtaLimit":
1000,"limitExceedOK":false,"days":1},"rate":{"reqLimit":10,"timePeriod":1}}}}
```

Response:

```
HTTP/1.1 200 OK
Date: Wed, 31 Jan 2018 05:58:20 GMT
Content-Length: 41
Content-Type: application/json
X-FRAME-OPTIONS: DENY

{"updateCurrentSlaForApproveResponse":{}}
```

3. Once the application request has been approved or rejected, change the notification status using the POST method from the /portal/prm/prm_pm_rest/services/partner_manager/notification/PartnerManagerNotification/updateNotificationStatus resource URL.

   To authorize the request, use the operator username and password in the header of the request.

   For more details on the request and response formats, see https://docs.oracle.com/communications/E81149_01/doc.70/e96582/resource_Partner_Manager_Notification.html.

   An example of an updated notification status request and response follow:

   Request:

```
POST /portal/prm/prm_pm_rest/services/partner_manager/notification/
PartnerManagerNotification/updateNotificationStatus HTTP/1.1
Host: 10.178.254.224:9001
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; rv:52.0) Gecko/20100101
Firefox/52.0
Accept: application/json, text/javascript, */*; q=0.01
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/json
AuthorizationX: Basic
b3JhY2xlb3AxOntBRVN9ZnBTRHBaeWw0dGRqR0lob3c2SzZFOThGT2tKbGxyZXB5Y3RrbUx0MmhPW
T0=
x-requested-with: XMLHttpRequest
Referer: http://10.178.254.224:9001/portal/partner-manager/index/main.html
Content-Length: 102
Connection: keep-alive

{"updateNotificationStatus":
{"notificationId":"b1e706fb-4436-401f-972c-99821f052805","status":"READ"}}
```

Response:

```
HTTP/1.1 200 OK
Date: Thu, 30 Aug 2018 08:42:48 GMT
Content-Length: 39
Content-Type: application/json
X-Frame-Options: DENY

{"updateNotificationStatusResponse":{}}
```

Set Application Password

Password for partner application needs to be set by the partner, which is sent in the T8 API request to the DSR API GW. This process is needed only if creating the application using the GUI interface. If the REST interface was used to create the application, the password was set during the creation of the application.

1. Access the partner and API management portal at https://<AppServerx-XMI-IP>:9002/portal/partner/index/partnerLogin.html.
2. Log into the portal using the partner account.
3. Click on the Applications tab.
4. Select the application.
5. Click on the key symbol next to the Traffic User property.
6. Set the traffic password and click **Update**.