

**Oracle® Communications Session Delivery
Manager**
Web Services
Release 7.4

January 2016

Notices

Copyright ©2014, 2011, Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Contents

1 About the Web Service Interface.....	11
Introduction.....	11
Supported Configuration Elements.....	11
Terminology.....	11
Overview.....	11
Server-Side.....	12
Client-Side.....	12
Data Structures.....	12
DeviceInfoObject.....	12
IntegrityCheckResult.....	13
NNCDetails.....	13
NNCServerIPInfo.....	13
SaveDeviceTaskMessage.....	13
SBCDetails.....	13
WSBatch.....	14
WSBatchOperation.....	14
WSConfigAttribute.....	14
WSConfigAttributeMetaData.....	14
WSConfigAttributeMetaData. AttributeTypeInfo.....	14
WSConfigElement.....	15
WSConfigElementMetaData.....	15
WSConfigResult.....	15
WSDeviceResult.....	15
Exceptions Faults.....	15
AcmeWSFault.....	15
AcmeAdminWSFault.....	16
AcmeConfigWSFault.....	16
AcmeDeviceWSFault.....	16
Build and Run the Sample Client Java Programs.....	16
Sample Work Flow.....	17
2 Administrative Management Level.....	19
login.....	19
Input Parameters.....	19
Output Parameters.....	19
Throws.....	19
logout.....	19
Throws.....	19
Input Parameters.....	19
Output Parameters.....	20
Throws.....	20
getUserInfo.....	20
Input Parameters.....	20
Output Parameters.....	20
Throws.....	20
getAllUserInfo.....	20
Input Parameters.....	20
Output Parameters.....	20
Throws.....	21

getAccountManagementInfo.....	21
Input Parameters.....	21
Output Parameters.....	21
Throws.....	21
getLoginBanner.....	21
Input Parameters.....	21
Output Parameters.....	21
Throws.....	21
getTrapReceivers.....	21
Input Parameters.....	22
Output Parameters.....	22
Throws.....	22
alarmSync.....	22
Input Parameters.....	22
Output Parameters.....	22
Throws.....	22

3 Device Management.....23

addDevice.....	23
Input Parameters.....	23
Output Parameters.....	23
Throws.....	23
loadDevice.....	23
Input Parameters.....	23
Output Parameters.....	24
Throws.....	24
deleteDevice.....	24
Input Parameters.....	24
Output Parameters.....	24
Throws.....	24
saveConfig.....	24
Input Parameters.....	24
Output Parameters.....	24
Throws.....	24
activateConfig.....	24
Input Parameters.....	25
Output Parameters.....	25
Throws.....	25
saveAndActivateConfig.....	25
Input Parameters.....	25
Output Parameters.....	25
Throws.....	25
getAllManagedDevicesNames.....	25
Input Parameters.....	25
Output Parameters.....	25
Throws.....	26
getAllManagedDevicesbyDeviceGroup.....	26
Input Parameters.....	26
Output Parameters.....	26
Throws.....	26
getSBCDetails.....	26
Input Parameters.....	26
Output Parameters.....	26
Throws.....	26
getDevicePollingInterval.....	26

Input Parameters.....	27
Output Parameters.....	27
Throws.....	27
lockDevice.....	27
Input Parameters.....	27
Output Parameters.....	27
Throws.....	27
unlockDevice.....	27
Input Parameters.....	27
Output Parameters.....	27
Throws.....	27
getAllDeviceGroupList.....	28
Output Parameters.....	28
Throws.....	28
addDeviceGroup.....	28
Input Parameters.....	28
Output Parameters.....	28
Throws.....	28
deleteDeviceGroup.....	28
Input Parameters.....	28
Output Parameters.....	28
Throws.....	29
getLCVContentSaveSessionReport.....	29
Specified by.....	29
Input Parameters.....	29
Output Parameters.....	29
Throws.....	29
getAllManagedDevices.....	29
Specified by.....	29
Input Parameters.....	29
Output Parameters.....	29
Throws.....	29
getAllManagedDeviceTargetNames.....	30
Specified by.....	30
Input Parameters.....	30
Output Parameters.....	30
Throws.....	30
getNNCDetails.....	30
Specified by.....	30
Input Parameters.....	30
Output Parameters.....	30
Throws.....	30
getTopLevelElementCount.....	30
Specified By.....	31
Input Parameters.....	31
Output Parameters.....	31
Throws.....	31
getAllAssociatedDevicesInEMSLicense.....	31
Input Parameters.....	31
Output Parameters.....	31
Throws.....	31
addDeviceToEMSLicense.....	31
Input Parameters.....	31
Output Parameters.....	31
Throws.....	31
removeDeviceFromEMSLicense.....	32

Input Parameters.....	32
Output Parameters.....	32
Throws.....	32

4 Configuration Management Level.....33

getPrimaryKeyByElementType.....	33
Input Parameters.....	33
Output Parameters.....	33
Throws.....	33
getTopLevelConfigElementTypeNames.....	33
Input Parameters.....	33
Output Parameters.....	34
Throws.....	34
getSubElementTypesByElementType.....	34
Input Parameters.....	34
Output Parameters.....	34
Throws.....	34
getRequiredSubElementTypesByElementType.....	34
Input Parameters.....	34
Output Parameters.....	34
Throws.....	34
getAllSupportedAttributeInfoByElementType.....	34
Input Parameters.....	35
Output Parameters.....	35
Throws.....	35
deleteConfigElement.....	35
Input Parameters.....	35
Output Parameters.....	35
Throws.....	35
updateConfigElement.....	35
Input Parameters.....	36
Output Parameters.....	36
Throws.....	36
getConfigElement.....	36
Input Parameters.....	36
Output Parameters.....	36
Throws.....	36
getAllConfigElements.....	36
Input Parameters.....	37
Output Parameters.....	37
Throws.....	37
applyBatch.....	37
Input Parameters.....	37
Output Parameters.....	37
Throws.....	37
addConfigElement.....	37
Input Parameters.....	37
Output Parameters.....	37
Throws.....	38
replace.....	38
Input Parameters.....	38
Output Parameters.....	38
Throws.....	38
addSubElement.....	38
Input Parameters.....	38

Output Parameters.....	38
Throws.....	39
deleteSubElement.....	39
Input Parameters.....	39
Output Parameters.....	39
Throws.....	39
getConfigElementMetaData.....	39
Input Parameters.....	39
Output Parameters.....	39
Throws.....	39
getConfigAttributeMetaData.....	40
Input Parameters.....	40
Output Parameters.....	40
Throws.....	40
getValuesForReferenceAttribute.....	40
Input Parameters.....	40
Output Parameters.....	40
Throws.....	40
newConfigElement.....	40
Input Parameters.....	40
Output Parameters.....	41
Throws.....	41
encryptedPassword.....	41
Input Parameters.....	41
Output Parameters.....	41
Throws.....	41
deleteUserChanges.....	41
Input Parameters.....	41
Output Parameters.....	41
Throws.....	41

A— ACLI to ACP Mappings..... 43

Retrieving the ACLI to ACP Mapping.....	43
ACLI to ACP Mapping.....	44
Physical Interface.....	47
Network Interface.....	48
Realm.....	49
Realm Media Address.....	52
Surrogate Agent.....	52
SIP Interface.....	53
SIP NAT.....	56
H.323 Stack.....	56
MGCP Config.....	58
DNS Config.....	59
Session Agent.....	60
Session Agent Group.....	63
Local Policy.....	63
Network Management Control.....	65
SIP Header Manipulation.....	66
Session Constraints.....	70
Session Translation.....	71
Translation Rules.....	71
RPH Profile.....	72
RPH Policy.....	72
Host Routes.....	72

SIP Local Map Entry.....	73
Codec Policy.....	74
Access Control.....	74
Media Profile.....	75
SIP Response Map.....	76
Diameter Director Agent.....	76
Diameter Director Configuration.....	77
Diameter Director Constraints.....	77
Diameter Director Group.....	78
Diameter Director Interface.....	79
Diameter Manipulation.....	80

B— Running a Legacy SOAP Client API..... 81

About this Guide

Overview

Oracle Communications Session Element Manager Web Services SOAP/XML Provisioning API Guide is used to enable users to write applications that automate the provisioning of Session Border Controllers (SBCs). Web Services consists of operations that can be performed against SBCs managed by a NNC server, and data structures used as input and output parameters to those operations. The operations are invoked by a client application to provision SBCs.

Deprecation

With the current release of Net-Net Central Web Service, the SOAP-based provisioning applications distributed with prior Net-Net EMS releases are deprecated.

Users of these deprecated APIs should consult this manual to familiarize themselves with the new, generic API. These users should also refer to Appendix B for information on differences between legacy and new APIs, and for information regarding required changes in legacy client applications that enable compatibility with the new API.

Audience

This guide is written for network administrators and architects, and provides information about the SOAP/XML-based Web Services implementation.

Related Oracle Communications Session Delivery Manager Documentation

The following table lists related documents for the Oracle Communications Session Delivery Manager

Document Name	Document Description
Release Notes	Contains information about the administration and software configuration of the Oracle Communications Session Delivery Manager feature support new to this release.
Installation Guide	Contains graphical and next mode installation information.
High Availability Guide	Describes Oracle Communications Session Delivery Manager High Availability (HA) and the HA cluster, which is a network of tightly-linked servers. HA provides continuous management of the SDM system.
Web Services SOAP XML Provisioning API Guide	Provides a full description of the individual interface definitions that make up the Application Programming Interface (API).
Core Functionality Guide	Contains an overview of the Oracle Communications Session Delivery Manager graphical user interface (GUI), detailed information about managing devices in Net-Net Central, and Net-Net Central licenses.
Session Element Manager Guide	Contains detailed information pertaining to the Session Element Manager application and describes the dashboard summary view, audit log, fault, and performance views.

About this Guide

Document Name	Document Description
Session Route Manager Guide	Contains detailed information about centrally automating the management and distribution of routing data.
Quick Start Guide	Contains a brief description of the GUI, along with information on how to add a device and perform basic configuration tasks.
Administration Guide	Contains information about security administration, which lets you create new users and new user groups, and set group-based authorization.
Report Manager Installation Guide	Contains instructions for installing Report Manager's dependencies and registering BI Publisher.
Report Manager User Guide	Contains information about configuring collection groups and creating reports.

Revision History

Date	Description
April 2014	<ul style="list-style-type: none">Initial release
November 2014	<ul style="list-style-type: none">Added Northbound Alarm Sync API
April 2015	<ul style="list-style-type: none">Added instructions in Appendix A for generating the ACLI to ACP mapping.
January 2016	<ul style="list-style-type: none">Added missing deleteUserChanges API to <i>Configuration Management Level</i> chapter.

About the Web Service Interface

Introduction

Oracle Communications Session Delivery Manager Web Service is a SOAP/XML Provisioning Application Programming Interface (API) enabling users to write applications that automate the provisioning of Oracle Net-Net Session Border Controllers (SBCs). Net-Net Central Web Service consists of operations that can be performed against SBCs managed by a NNC server, and data structures used as input and output parameters to those operations. The operations are invoked by a client application to provision Net-Net SBCs.

This document provides a full description of the individual interface definitions that make up the API. Throughout the document, the Net-Net SBC is referred to as a managed device.

Supported Configuration Elements

The API supports all configuration elements available on the Net-Net SBC. Configuration elements and sub-elements are referred to in this document and in the API as managed objects and sub-objects.

Terminology

The following terms are used throughout the document:

- Active configuration – the configuration as currently present on the device. Active configurations are read-only within Oracle Communications Session Delivery Manager.
- Session ID – identifies the connection that is established between a SOAP client application instance and the NNC server. The session ID is used by the server to distinguish between multiple clients currently logged in, and is a required parameter for almost all operations supported by the API.
- Managed device – for the purpose of the API, the Oracle Session Border Controller (SBC)

Overview

The Web Service Provisioning API is part of a client/server programming model. The Oracle Communications Session Delivery Manager server accepts and responds to requests from clients coded to use the API. It acts on the requests to read and update configuration data in the Oracle Communications Session Delivery Manager database or apply configurations to Net-Net SBCs in the same way in which it responds to requests from the Oracle Communications Session Delivery Manager GUI client application.

The Provisioning API itself is defined by Oracle, and the messages exchanged between SOAP/XML clients and the server are encapsulated in the standard Simple Object Access Protocol (SOAP) format, as defined by the World Wide Web Consortium (W3C).

Server-Side

The Net-Net EMS server incorporates CXF technology to handle the processing of the SOAP messages received from the client. Apache CXF is a full-featured, open-source web services framework. For more information about SOAP-based Web Services, refer to <http://www.w3.org/TR/soap>.

The fully-specified API is published in the Net-Net EMS Provisioning 4000 and 9000 API Reference Guides. You can also obtain it from a running server in the structured form of Web Services Definition Language (WSDL) files. To access the WSDL files, point a browser to a running Oracle Communications Session Delivery Manager server, as follows:

```
http://<ip address>:8080/ACMEWS/services
```

For example:

```
http://192.0.2.01:8080/ACMEWS/services
```

Client-Side

Client applications are typically developed by users who want to integrate automated provisioning into their existing OSS infrastructures. The client application is custom-built to use the published API to read and set parameters in the configurations in the NNC server database, and to apply those settings to devices being managed by the NNC server.

The client application can be written in any language that supports SOAP/XML-based Web Services. Examples are provided for clients written in Java on the Net-Net EMS software distribution.

The Net-Net EMS server supports simultaneous connections from multiple SOAP clients and graphical user interface (GUI) clients. User profiles defined in the Oracle Communications Session Delivery Manager database are used to validate login requests from either type of client, but profiles can be defined only through the GUI interface. Audit trail entries are generated for operations performed by all clients, whether they are GUI or SOAP-based.

For more information about SOAP-based Web Services and examples of client code, see the CXF User's Guide at <http://cxf.apache.org/docs/index.html>.

Data Structures

This section describes the data structures used in the API.

The WSDL definitions for data structures can be found in the following WSDL files:

- AdminMgmtIFService.wsdl
- ConfigMgmtIFService.wsdl
- DeviceMgmtIFService.wsdl

You can access these files from the Oracle Communications Session Delivery Manager Available SOAP services screen at <http://<ipAddress>:8080/ACMEWS/services>, where <ipAddress> is the IP address of a Oracle Communications Session Delivery Manager Server. Use the links in the right-hand portion of the screen to display the WSDL files

DeviceInfoObject

DeviceInfoObject is a data structure that contains the information necessary to add a device to NNC system.

- ArrayList<String> deviceIPList
- String username
- String Password
- String communityName
- int snmpPort
- String deviceGroupName, which will be the full path of device group, if we have a device group (groupAA1) under group1/groupAA, the client need to pass group1/groupAA/groupAA1

- WebServicesProtocolEnum, which is an enumeration contains (HTTP or HTTPS), this piece information is only used by 2600 series device.
- WebServicesProtocolEnum webServicesProtocol, this piece information is only used by 2600 series device.

IntegrityCheckResult

IntegrityCheckResult contains top-level element count information as follows.

- String elementName; the name of the target element
- int ElementCount; the number of element instances

NNCDetails

NNCDetails contains product version information as follows.

- String version; The product version, for example, NNC7.0.0

NNCServerIPInfo

NNCServer contains NNC-Server-specific data as follows.

- String serverIPAddress;
- String serverName;
- String serverStatus;
- long inactivityCount;
- int heartBeatFailureMeter;
- int maxHeartBeatFailureMeter;
- long missedHBCount;
- long heartBeatCount;
- String lastHeartBeatTime;
- String downTime;
- String upTime;
- int resetCount;

SaveDeviceTaskMessage

SaveDeviceTaskMessage contains the result of save and/or activate operation as follows.

- String operation, operation type, Save or SaveAndActivate, Activate
- String tasked, which is task id which will be populated on devicetaskDetails schema once save or save\activate is fully done on server side.
- String username, which contains the user name who is going to perform save or activate operation
- String deviceName, device name the operation is performed.
- String isLockSuccess, specifies SBC Lock status
- isValidationSuccess, does the validation pass for the configuration elements integrity check
- String isEraseCacheSuccess, which is not applicable for granular save
- String isCreateSuccess specifies success or failure
- String isIntegrityCheckSuccess, if EMS count match to the SBC count
- String isRestoreConfigSuccess if Integrity check fail, we need to restore original SBC configuration data.
- String isSaveConfigSuccess, specifies success or failure of Save command
- isActivateConfigSuccess, specifies success or failure of Activate command
- isUnlockSuccess, specifies SBC Unlock status

SBCDetails

SBCDetails contains information describing the SBC configuration as follows.

About the Web Service Interface

- String `deviceName`, device name
- String `targetName`, target name
- String `domainName`, which is device group path
- String `SBCVersion`, SBC version, such as SC620
- String `snmpCommunityName`, snmp community name
- int `snmpPort`, snmp port
- String `primaryIP`, primary IP address
- String `secondaryIP`, secondary IP address
- String `hardwareVersion`, hard ware version, such as NN4500, NN4200

WSBatch

WSBatch contains information describing a SOAP batch operation as follows.

- `ArrayList < WSBatchOperation > operationsToApply`, Arraylist of WSBatchOperations, described in the following section.

WSBatchOperation

WSBatchOperation contains information describing SOAP batch operation content.

- String `operation`, operation can be ADD, UPDATE, DELETE
- `WSConfigElement configElement`, which is a data structure described before.

WSConfigAttribute

WSConfigAttribute contains configuration attribute information as follows.

- String `name`: the name of the attribute
- String `value`: the value of the attribute

WSConfigAttributeMetaData

WSConfigAttributeMetaData contains attribute-specific meta data as follows.

- String `name`; The name of the attribute
- String `acliName` The ACLI name for the attribute
- `AttributeValueTypeInfo valueTypeInfo`; The type information of attribute.
- String `delimiter`; null if the value of this attribute is not delimited string.
- boolean `isRequired`; true, if this attribute is a required to configure the parent element
- String `defaultValue`; The default value
- String `referred_Element_Type_Name`; If this attribute is referring to another element, that element's type name.
- `List<String> suggested_Values`; A list of suggested values for this attribute.
- `List<WSNumericRange> valid_Numeric_Range`; The valid range of numeric values for this attribute. Applicable only if `valueTypeInfo` is NUMERIC
- `List<String> enumerated_values`; A list of valid enumeration for this attribute

WSConfigAttributeMetaData. AttributeValueTypeInfo

WSConfigAttributeMetaData.AttributeValueTypeInfo contains Enumerations of valid attribute types as follows.

- This is an enumeration of valid types of an attribute.
- `numeric`
- `string`
- `delimited_string`
- `ipaddress`
- `boolean`

- date
- enumerated_value,
- reference_to_another_element
- ipaddress_and_portnumber

WSConfigElement

WSConfigElement is the generic data structure for all configuration elements. This data structure is used by add/update/delete/get functions to describe a new or modified configuration element. It contains the following data:

- String type: identifies the target configuration element. Top-level configuration elements are identified by their Acme Control Protocol (ACP) element names, for example sipManipulation.
- ArrayList<WSConfigAttribute> attributeList: an ArrayList of WSConfigAttributes
- ArrayList<WSConfigElement> children: ArrayList of WSConfigElements. that provides information on sub-elements.

Sub-elements type (children) are identified by a path expression rooted in an ACP element, for example, sipManipulation/headerRule/elementRule specifies an Element Rule

- String elementTypePath: which is used internally to specify the path expression (for example sipManipulation/headerRule/elementRule). Because this tag is generated internally, the client does not need to set this data.

WSConfigElementMetaData

WSConfigElementMetaData contains element-specific meta data as follows.

- private String type; The type name of the element
- private boolean isSingleInstance; true, if this element is a single instance
- private String elementTypePath; The full path of the element starting from the root configuration
- private List<String> subElementTypeNames; A list of subelement type names of this element
- private List<WSConfigAttributeMetaData> attributeMetaDataList; A list of attribute metadata for this element

WSConfigResult

WSConfigResult contains the result of an operation as follows.

- Boolean resultFlag: status of the operation
- String objectId: objectId value
- ArrayList of validation message string

WSDeviceResult

WSDeviceResult contains the result of network level (device) operation as follows.

- Boolean resultFlag: status of the operation
- ArrayList<String> validationMessage, the success or fail message

Exceptions Faults

The following exceptions may be generated by the server in attempting to process requests from a SOAP/XML interface client.

AcmeWSFault

AcmeWSFault is the base exception class for the web service interface.

AcmeAdminWSFault

AcmeAdminWSFault is the exception class for managing administrative level (AdminMgmtIF) APIs (login and logOut). In the event of an access error, AdminMgmtIF throws this exception.

AcmeConfigWSFault

AcmeConfigWSFault is the exception class for managing configuration level (DeviceConfigIF) APIs. In the event of a configuration error, AdminConfigIF throws this exception.

AcmeDeviceWSFault


AcmeDeviceWSFault is the exception class for managing device level (DeviceMgmtIF) APIs. In the event of a device-level error, AdminDeviceIF throws this exception.

Build and Run the Sample Client Java Programs

Use the following procedure to build and run the sample client Java code.

The sample client code is available at {CXFClient_HOME}/sampleSouce folder. You can make changes to any of the files contained in this folder and build an executable image using build.bat.

1. Extract CXFClient.zip (contained on the Oracle software distribution CD) to a folder on the client computer. This folder provides the {CXFClient_HOME}.
2. Go to {CXFClient_HOME}/bin.
3. Edit run.bat, the file that allows you to run the sample client code, by changing the JAVA_HOME path variable to match the JDK installation path. In addition, edit the SERVER_NAME and SERVER_PORT variables to match the IP address and port number of the Oracle Communications Session Delivery Manager Server.

 **Note:** The CXFClient requires JDK 1.6.0 or later; the latest update is recommended.

4. Edit build.bat by making the same change to the JAVA_HOME path variable.
5. The following procedure, which imports one or more server certificates to a specific JAVA keystore, is required only if the Web Service Interface will run over HTTPS. These steps (5a through 5f) can be safely ignored if client/server transactions will take place over unsecured HTTP.

1. Use FTP to move a copy of a Oracle Communications Session Delivery Manager Server public certificate to the JAVA_HOME location on the client computer.

The certificate is usually at opt/AcmePacket/NNC700/ssl/nncentral_server.cer on the Oracle Communications Session Delivery Manager Server.

2. Use the JAVA keytool utility to import the public certificate into a specified JAVA keystore. For example,

```
keytool -import -keystore trustedCerts -alias NNC-01 -file nnC01.cer
```

imports the certificate file, nnC01.cer, into the keystore named trustedCerts; the keystore file will be referenced by the NNC-01 alias.

Note that you will be prompted for the keystore password before the import operation is initiated.

For example:

```
Owner: EMAILADDRESS=test@test.com, CN=172.30.10.120, OU=NmsCore,
O=Acme Packet Inc., ST=Some-State, C=AU
Issuer: EMAILADDRESS=test@test.com, CN=172.30.10.120, OU=NmsCore,
O=Acme Packet Inc., ST=Some-State, C=AU
Serial number: 8b4d53819b6dfff1
Valid from: Tue Nov 14 16:04:53 EST 2006 until: Sat Jan 31 16:04:53 EST
2015
Certificate fingerprints:
MD5: 98:DA:F6:04:A8:A0:CA:D4:33:83:2A:3F:CE:C3:FB:CD
```



```
SHA1: F4:BB:72:7D:43:25:56:86:6A:70:55:27:63:96:D2:13:DF:89:B2:68
Trust this certificate? [no]: y
Certificate was added to keystore
```

3. Edit run.bat by changing the TRUST_STORE variable to match the location of the JAVA keystore that contains the public certificates of associated Oracle Communications Session Delivery Manager Servers.
4. Edit run.bat by changing the TRUST_STORE_PASSWORD to match the password required to access the JAVA keystore containing the Oracle Communications Session Delivery Manager Server certificates.
5. Use the JAVA keytool utility to confirm the presence of the key in the keystore. For example,

```
keytool -list -v -keystore trustedCerts
```

provide a verbose display of the contents of the designated JAVA keystore, in this case, trustedCerts.

Note that you will be prompted for the keystore password before the keystore contents are displayed.

6. Repeat Steps 5a, 5b, and 5e to import additional Oracle Communications Session Delivery Manager Server certificates to the same JAVA keystore.
6. Optionally edit {CXFClient_HOME}/conf/client.properties by changing the value of the session_timeout_ms property to specify a non-default session timeout value, expressed in milliseconds.
7. Use build.bat to compile the client application.
8. Use run.bat to run the client application.

Sample Work Flow

The following procedure illustrates the creation of a `WSConfigElement` that defines a specific network interface, and associated keep-alive mechanisms. Refer to [WSConfigElement](#) for details.

1. Use the login administrative API to access a Oracle Communications Session Delivery Manager server
2. Use the `newConfigElement` API to create a template (actually a `WSConfigElement` data structure with default attribute values) of the `networkInterface` Type.
3. Construct an `ArrayList` of `WSConfigAttribute` data structures to assign local attribute values to the default `networkInterface` template returned by `newConfigElement`. Refer to [WSConfigAttribute](#) for details.
4. Add this attribute `ArrayList` to the `networkInterface` `WSConfigElement` data structure. This step completes configuration of the top-level `networkInterface`.
5. Use the `newConfigElement` API to create a second-level (child) template of the `networkInterface/GWHeartbeat` Type.
6. Construct an `ArrayList` of `WSConfigAttribute` data structures to assign local attribute values to the default `networkInterface/GWHeartbeat` template returned by `newConfigElement`.
7. Add the attribute `ArrayList` to the child template. This step completes configuration of the second-level child.
8. Construct an `ArrayList` of child `WSConfigElements`; in this case the array contains only a single element.
9. Append this child `ArrayList` to the `WSConfigElement` data structure.
10. Use the `addConfigElement` API to commit the `WSConfigElement` to the configuration database.
11. Logout, using the `logOut` API

Administrative Management Level

login

login is used by the CXF client to login to the CXF Web service.

```
public java.lang.String login(java.lang.String username,  
java.lang.String password)  
throws com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault
```

Input Parameters

- username: username information
- password: password information

Output Parameters

- String sessionId

Throws

com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault

logOut

logOut ends the current user session.

```
public com.acmepacket.ems.ws.service.userobjects.WSDeviceResult logOut()  
throws com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault
```

Throws

```
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault
```

Input Parameters

None

Output Parameters

- WSDeviceResult data structure

Throws

com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault

getUserInfo

Retrieves the values for the following parameters for a user:

- Account Never Expires
- Account Expires
- Password Never Expires
- Password Expires (Days)

Public UserInfo getUserInfo(String userName) throws AcmeAdminWSFault

Input Parameters

- userName- - user name

Output Parameters

UserInfo

- Boolean accountExpire
- String accountExpirationDate
- Boolean passwordExpire
- String passwor ExpirationDate
- String userName

Throws

com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault

getAllUserInfo

Retrieves the values for the following parameters for all users added to the system:

- Account Never Expires
- Account Expires
- Password Never Expires
- Password Expires (Days)

Public UserInfo getUserInfo(String userName) throws AcmeAdminWSFault

Input Parameters

None

Output Parameters

UserInfo

- Boolean accountExpire
- String accountExpirationDate

- Boolean passwordExpire
- String passwordExpirationDate
- String userName

Throws

com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault

getAccountManagementInfo

Retrieves the values for the following parameters:

- Password Reuse Count
- Inactivity Time for admin user
- Inactivity Time for non-admin user

```
Public AccountManagementInfo getAccountManagementInfo() throws  
AcmeAdminWSFault
```

Input Parameters

None

Output Parameters

AccountManagementInfo

- Int passwordResuseCount
- Int adminUserInactivityTimeout
- Int nonAdminUserInactivityTimeout

Throws

com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault

getLoginBanner

Retrieves the the login banner.

```
Public String getLoginBanner() throws AcmeAdminWSFault
```

Input Parameters

None

Output Parameters

- String loginBanner

Throws

com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault

getTrapReceivers

Retrieves the trap receiver information.

```
Public String getLoginBanner() throws AcmeAdminWSFault
```

Input Parameters

None

Output Parameters

TrapReceiver

- String ipAddress
- Int udpPort
- String communityName

Throws

com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault

alarmSync

Forwards traps stored in OCSDM system to the network management system.

Trap receivers have a state for flag the Alarm Sync operation state. Possible states are:

- Enabled
- Disabled
- Suspended
- Syncing
- SyncSucceed
- SyncFailed

Once you add trap receiver to OCSDM, the initial trap receiver state will be set to Enabled. On following states (Enabled/SyncSucceed/SyncFailed) the AlarmSync request will be sent to the OCSDM server successfully, otherwise it will throw the AcmeWSAdminFault Exception.

```
alarmSync(String destTrapReceiverIP, String startTime, String endTime) throws AcmeAdminFault
```

Input Parameters

- destTrapReceiverIP: Destination trap receiver IP address
- startTime - Start sync time in MM/dd/yyyy HH:mm:ss (OCSDM server local time)
- endTime - End sync time in MM/dd/yyyy HH:mm:ss (OCSDM server local time)

Output Parameters

A WSAlarmSyncResult data structure.

The detail of this data structure is as follows:

1. Boolean resultFlag: either true or false
2. ArrayList<String > validationMessage: Validation message
3. Int numOfTrapsToBeSync: The number of the Alarms to be sync if successful

Throws

```
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault
```

Device Management

addDevice

addDevice adds device details to the configuration database.

```
public com.acmepacket.ems.ws.service.userobjects.WSDeviceResult
addDevice(com.acmepacket.ems.ws.service.userobjects.DeviceInfoObject deviceInfoObject)throws
com.acmepacket.ems.ws.service.fault.AcmeDeviceWSFault,
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault
```

Input Parameters

- deviceInfoObject - - DeviceInfoObject data structure

Output Parameters

WSDeviceResult

Throws

```
com.acmepacket.ems.ws.service.fault.AcmeConfigWSFault
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault
com.acmepacket.ems.ws.service.fault.AcmeDeviceWSFault
```

loadDevice

loadDevice adds a new managed device to the configuration database.

```
public com.acmepacket.ems.ws.service.userobjects.WSDeviceResult loadDevice(java.lang.String targetName) throws
com.acmepacket.ems.ws.service.fault.AcmeDeviceWSFault,
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault
```

Input Parameters

- targetName: target name information

Output Parameters

WSDeviceResult

Throws

com.acmepacket.ems.ws.service.fault.AcmeDeviceWSFault

com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault

deleteDevice

deleteDevice deletes configuration data from database, including device detail and configuration document information.

```
public com.acmepacket.ems.ws.service.userobjects.WSDeviceResult deleteDevice(java.lang.String targetName)
throws com.acmepacket.ems.ws.service.fault.AcmeDeviceWSFault,
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault
```

Input Parameters

- targetName---target name information

Output Parameters

WSConfigResult

Throws

com.acmepacket.ems.ws.service.fault.AcmeDeviceWSFault

com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault

saveConfig

saveConfig pushes configuration dataset changes to the SBC.

```
public com.acmepacket.ems.common.SaveDeviceTaskMessage saveConfig(java.lang.String targetName) throws
com.acmepacket.ems.ws.service.fault.AcmeDeviceWSFault,
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault
```

Input Parameters

- targetName - - target name information (for example, sd80_sd8)

Output Parameters

SaveDeviceTaskMessage, a data structure described as before.

Throws

com.acmepacket.ems.ws.service.fault.AcmeDeviceWSFault

com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault

activateConfig

activateConfig promotes a specified configuration to the running configuration area.

```
public com.acmepacket.ems.common.SaveDeviceTaskMessage activateConfig(java.lang.String targetName)throws  
com.acmepacket.ems.ws.service.fault.AcmeDeviceWSFault,  
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault
```

Input Parameters

- targetName - - target name information

Output Parameters

SaveDeviceTaskMessage, a data structure described as before.

Throws

com.acmepacket.ems.ws.service.fault.AcmeDeviceWSFault - - throws exception while error occurs
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault

saveAndActivateConfig

saveAndActivateConfig first runs saveConfig, and then activateConfig.

```
public com.acmepacket.ems.common.SaveDeviceTaskMessage  
saveAndActivateConfig(java.lang.String targetname)throws  
com.acmepacket.ems.ws.service.fault.AcmeDeviceWSFault,  
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault
```

Input Parameters

- targetName- - name of the target device

Output Parameters

SaveDeviceTaskMessage, a data structure described as before

Throws

com.acmepacket.ems.ws.service.fault.AcmeDeviceWSFault
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault

getAllManagedDevicesNames

getAllManagedDevicesNames returns a list of all managed device names present in the configuration database.

```
public java.util.ArrayList<java.lang.String> getAllManagedDevicesNames()throws  
com.acmepacket.ems.ws.service.fault.AcmeDeviceWSFault,  
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault
```

Input Parameters

None

Output Parameters

an ArrayList of Managed device target name, such as 172.30.80.81, 172.30.80.150-172.30.80.131

Throws

com.acmepacket.ems.ws.service.fault.AcmeDeviceWSFault
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault

getAllManagedDevicesbyDeviceGroup

getAllManagedDevicesByDeviceGroup returns a membership list for a specified device group.

```
public java.util.ArrayList<com.acmepacket.ems.ws.service.userobjects.SBCDetails>  
getAllManagedDevicesByDeviceGroup(java.lang.String deviceGroupPath) throws  
com.acmepacket.ems.ws.service.fault.AcmeDeviceWSFault,  
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault
```

Input Parameters

- deviceGroupPath - -- device group full path, for example,group2/groupAC/groupAC1

Output Parameters

ArrayList of SBCDetails object

Throws

com.acmepacket.ems.ws.service.fault.AcmeDeviceWSFault
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault

getSBCDetails

getSBCDetails returns information of a specified SBC.

```
public com.acmepacket.ems.ws.service.userobjects.SBCDetails getSBCDetails(java.lang.String targetName) throws  
com.acmepacket.ems.ws.service.fault.AcmeDeviceWSFault,  
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault
```

Input Parameters

- targetName, - the target name

Output Parameters

SBCDetails object

Throws

com.acmepacket.ems.ws.service.fault.AcmeConfigWSFault
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault

getDevicePollingInterval

Retrieves the the polling interval set to poll SBCs.

```
Public int getDevicePollingInterval() throws AcmeAdminWSFault, AcmeDeviceWSFault
```

Input Parameters

None

Output Parameters

Device Polling Interval (seconds)

Throws

com.acmepacket.ems.ws.service.fault.AcmeDeviceWSFault
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault

lockDevice

lockDevice reserves a specified SBC.

```
public com.acmepacket.ems.ws.service.userobjects.WSDeviceResult lockDevice(java.lang.String targetName)throws  
com.acmepacket.ems.ws.service.fault.AcmeDeviceWSFault,  
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault
```

Input Parameters

- targetName, - the target name

Output Parameters

WSDeviceResult, a data structure described as before

Throws

com.acmepacket.ems.ws.service.fault.AcmeConfigWSFault
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault

unlockDevice

unlockDevice release a previously reserved SBC.

```
public com.acmepacket.ems.ws.service.userobjects.WSDeviceResult  
unlockDevice(java.lang.String targetName) throws  
com.acmepacket.ems.ws.service.fault.AcmeDeviceWSFault,  
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault
```

Input Parameters

- targetName, - the target name

Output Parameters

WSDeviceResult, a data structure described as before

Throws

com.acmepacket.ems.ws.service.fault.AcmeConfigWSFault
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault

getAllDeviceGroupList

getAllDeviceGroupList returns an array containing the names of all Device Groups.

```
public java.util.ArrayList<java.lang.String> getAllDeviceGroupList () throws  
com.acmepacket.ems.ws.service.fault.AcmeDeviceWSFault,  
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault
```

Output Parameters

ArrayList of devicegroup name

Throws

com.acmepacket.ems.ws.service.fault.AcmeConfigWSFault
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault

addDeviceGroup

addDeviceGroup adds a specified Device Group to the Configuration Database.

```
public boolean addDeviceGroup(java.lang.String deviceGroupPath)throws  
com.acmepacket.ems.ws.service.fault.AcmeDeviceWSFault,  
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFaultit will try to add the device group
```

Input Parameters

- deviceGroupPath: - device group path name(for example,group2/groupAC/groupAC1), which means that we are going to add groupAC1 to the device group group2/groupAC

Output Parameters

True or False

Throws

com.acmepacket.ems.ws.service.fault.AcmeConfigWSFault
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault

deleteDeviceGroup

deleteDeviceGroup deletes a specified Device Group from the Configuration Database.

```
public boolean deleteDeviceGroup(java.lang.String deviceGroupPath)throws  
com.acmepacket.ems.ws.service.fault.AcmeDeviceWSFault,  
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault
```

Input Parameters

- deviceGroupPath: - device group path name

Output Parameters

True or False

Throws

com.acmepacket.ems.ws.service.fault.AcmeConfigWSFault
 com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault

getLCVContentSaveSessionReport

getLCVContentSaveSessionReport returns a list if WSConfigElements created or modified by a specified user.

```
public java.util.ArrayList<com.acmepacket.ems.ws.service.userobjects.WSConfigElement>
getLCVContentSaveSessionReport(java.lang.String targetame, java.lang.String userName) throws
com.acmepacket.ems.ws.service.fault.AcmeDeviceWSFault,
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault
```

Specified by

getLCVContentSaveSessionReport java.lang.String in DeviceMgmtIF interface

Input Parameters

- targetName - -targetname
- userName - -user name

Output Parameters

A list of WSConfigElements

Throws

```
com.acmepacket.ems.ws.service.fault.AcmeConfigWSFault
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault
com.acmepacket.ems.ws.service.fault.AcmeDeviceWSFault
```

getAllManagedDevices

getAllManagedDevices returns a list of SBCDetails for all managed devices.

```
public ArrayList<SBCDetails> getAllManagedDevices() throws AcmeDeviceWSFault, AcmeAdminWSFault;
```

Specified by

getAllManagedDevices in DeviceMgmt interface

Input Parameters

None

Output Parameters

ArrayList<SBCDetails>: a List of SBCDetails object

Throws

com.acmepacket.ems.ws.service.fault.AcmeConfigWSFault
 com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault

getAllManagedDeviceTargetNames

getAllManagedDeviceTargetNames returns a list of the names of all managed devices.

```
public ArrayList<String> getAllManagedDeviceTargetNames() throws AcmeDeviceWSFault, AcmeAdminWSFault;
```

Specified by

getAllManagedDeviceTargetNames in the DeviceMgmtIF

Input Parameters

- None

Output Parameters

ArrayList< String >: a List of String

Throws

com.acmepacket.ems.ws.service.fault.AcmeConfigWSFault

com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault

getNNCDetails

getNNCDetails returns an NNCDetails data structure for the NNC server that contains software version, addressing, and cluster information.

```
public NNCDetails getNNCDetails() throws AcmeDeviceWSFault, AcmeAdminWSFault;
```

Specified by

getNNCDetails in DeviceMgmtIF interface.

Input Parameters

- None

Output Parameters

NNCDetails object

Throws

com.acmepacket.ems.ws.service.fault.AcmeConfigWSFault

com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault

getTopLevelElementCount

getTopLevelElementCount returns a count of all top-level configuration elements. To obtain the element count, it sums the local configuration copy and local configuration change values.

```
public ArrayList<IntegrityCheckResult> getTopLevelElementCount(String targetName) throws  
AcmeDeviceWSFault, AcmeAdminWSFault
```

Specified By

getTopLevelElementCount in interface DeviceMgmtIF

Input Parameters

- targetName- - device name

Output Parameters

An ArrayList of IntegrityCheckResult data structures

Throws

com.acmepacket.ems.ws.service.fault.AcmeConfigWSFault
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault

getAllAssociatedDevicesInEMSLicense

getAllAssociatedDevicesInEMSLicense returns an array list of all managed devices associated with an element manager license; it throws an exception in the absence of an element manager license.

```
public ArrayList<String> getAllAssociatedDevicesInEMSLicense() throws AcmeAdminWSFault,  
AcmeDeviceWSFault
```

Input Parameters

none

Output Parameters

none

Throws

com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault
com.acmepacket.ems.ws.service.fault.AcmeDeviceWSFault

addDeviceToEMSLicense

addDeviceToEMSLicense adds a specified device to the list of licensed devices; it throws an exception in the absence of an element manager license.

```
public WSConfigResult addDeviceToEMSLicense(@WebParam(name= targetName) String targetName) throws  
AcmeAdminWSFault, AcmeDeviceWSFault
```

Input Parameters

- targetName- - device name

Output Parameters

none

Throws

com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault
com.acmepacket.ems.ws.service.fault.AcmeDeviceWSFault

removeDeviceFromEMSLicense

removeDeviceFromEMSLicense removes a specified device from the list of licensed devices; it throws an exception in the absence of an element manager license.

```
public WConfigResult removeDeviceFromEMSLicense(@WebParam(name= targetName) String targetName)
throws AcmeAdminWSFault, AcmeDeviceWSFault
```

Input Parameters

- targetName - - device name

Output Parameters

none

Throws

com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault
com.acmepacket.ems.ws.service.fault.AcmeDeviceWSFault

Configuration Management Level

getPrimaryKeyByElementType

getPrimaryKeyByElementType returns a list of PrimaryKey information for a given element type.

```
public java.util.ArrayList<java.lang.String>
getPrimaryKeyByElementType(java.lang.String targetDevice,
java.lang.String elementType) throws
com.acmepacket.ems.ws.service.fault.AcmeConfigWSFault,
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault
```

Input Parameters

- elementType: element type information

For a sub-element, the elementType references the full path: for example, to obtain the primary key value for an elementRule, use the expression sipManipulation/headerRule/elementRule.

Output Parameters

An ArrayList of String primary key information, which will include path expression inside the string.

Throws

```
com.acmepacket.ems.ws.service.fault.AcmeConfigWSFault
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault
```

getTopLevelConfigElementTypeNames

getTopLevelConfigElementTypeNames returns a list of top-level configuration element names associated with a specific SBC

```
public java.util.ArrayList<java.lang.String>
getTopLevelConfigElementTypeNames(java.lang.String targetName) throws
com.acmepacket.ems.ws.service.fault.AcmeConfigWSFault,
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault
```

Input Parameters

- targetName: target device information, which is the target name

Output Parameters

An ArrayList of top configuration element names.

Throws

```
com.acmepacket.ems.ws.service.fault.AcmeConfigWSFault  
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault
```

getSubElementTypesByElementType

getSubElementTypesByElementTypes returns a list of sub element types for the given elementType.

```
public ArrayList<String> getSubElementTypesByElementType(String targetName,  
String elementType) throws AcmeConfigWSFault, AcmeAdminWSFault
```

Input Parameters

- targetName: target device information, which is the target name
- elementType: element type. For example, sipInterface

Output Parameters

A list of String (sub element types) for the given element type.

Throws

```
com.acmepacket.ems.ws.service.fault.AcmeConfigWSFault  
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault
```

getRequiredSubElementTypesByElementType

getRequiredSubElementTypesByElementType returns a list of required sub element types for the given elementType.

```
public ArrayList<String> getSubElementTypesByElementType(String targetName,  
String elementType) throws AcmeConfigWSFault, AcmeAdminWSFault
```

Input Parameters

- targetName: target device information, which is the target name
- elementType: element type. For example, sipInterface

Output Parameters

Returns a list of required sub element types for the given element type.

Throws

```
com.acmepacket.ems.ws.service.fault.AcmeConfigWSFault  
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault
```

getAllSupportedAttributeInfoByElementType

getAllSupportedAttributeInfoByElementType returns a list of WSConfigAttributeMetaData information for the given elementType.

```
public ArrayList<WSConfigAttributeMetaData>
getAllSupportedAttributeInfoByElementType(String targetName, String
elementType) throws AcmeConfigWSFault, AcmeAdminWSFault
```

Input Parameters

- targetName: target device information, which is the target name
- elementType: element type. For example, sipInterface

Output Parameters

Returns a list of WSConfigAttribute information for the given elementType.

Throws

```
com.acmepacket.ems.ws.service.fault.AcmeConfigWSFault
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault
```

deleteConfigElement

deleteConfigElement deletes a specified configuration element; any existing child elements of the element are also deleted.

```
public com.acmepacket.ems.ws.service.userobjects.WSConfigResult
deleteConfigElement(java.lang.String targetDevice,
com.acmepacket.ems.ws.service.userobjects.WSConfigElement wsConfigElement) throws AcmeConfigWSFault,
AcmeAdminWSFault
```

Input Parameters

- targetName: target device information, which is the target name
- wsConfigElement: wsConfigElement information

Output Parameters

WSConfigResult

Throws

```
com.acmepacket.ems.ws.service.fault.AcmeConfigWSFault
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault
```

updateConfigElement

updateConfigElement performs an incremental update of a single specified top-level WSConfigElement, or a single, specified sub-element. The user application does not need to provide a complete WSConfigElement; it needs to provide only the primaryKeys required to identify the target configuration element instance, and the attribute/value pairs that require update.

This Oracle Communications Session Delivery Manager release does not provide an API to update sub-elements; as a result, updateConfigElement is used to update both top-level and sub-elements. However, users cannot use updateConfigElement to add or delete sub-elements from an existing configuration element. Use addSubElement to add a sub-element, and deleteSubElement to delete a sub-element.

When updating a sub-element, users must provide an unambiguous path to the single target sub-element that requires update. The path consists of the ACP top-level identifier, followed by one or more sub-element types. At each path level, primaryKeys (such as ACLI object names) must be supplied to ensure unambiguous element identification. For example, the following path

Configuration Management Level

`sipManipulation(primaryKeys)/headerRule(primaryKeys)/elementRule(name="rule1")`

identifies a SIP element rule to be updated.

`public com.acmepacket.ems.ws.service.userobjects.WSConfigResult
updateConfigElement(java.lang.String targetDevice,
com.acmepacket.ems.ws.service.userobjects.WSConfigElement wsConfigElement) throws
com.acmepacket.ems.ws.service.fault.AcmeConfigWSFault,
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault`

Input Parameters

- `targetDevice` - - the target name
- `wsConfigElement` - - contains primarykey attributes to identify the target Configuration Element, and additional attributes which require update to new values

Output Parameters

`WSConfigResult`

Throws

`com.acmepacket.ems.ws.service.fault.AcmeConfigWSFault
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault`

getConfigElement

`getConfigElement` gets a specified configuration element from the configuration database.

`public com.acmepacket.ems.ws.service.userobjects.WSConfigElement
getConfigElement(java.lang.String targetDevice,
com.acmepacket.ems.ws.service.userobjects.WSConfigElement wsConfigElement) throws
com.acmepacket.ems.ws.service.fault.AcmeConfigWSFault,
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault`

Input Parameters

- `targetDevice` - - target name information
- `wsConfigElement`- - `WSConfigElement` you need to supply only the primary key attributes on this `wsConfigElement`.

Output Parameters

`WSConfigElement` data structure describing the requested configuration element

Throws

`com.acmepacket.ems.ws.service.fault.AcmeConfigWSFault
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault`

getAllConfigElements

`getAllConfigElements` gets a `ArrayList` of all configuration elements from the configuration database.

`public ArrayList<WSConfigElement> getAllConfigElement(String targetName, String elementType) throws
com.acmepacket.ems.ws.service.fault.AcmeConfigWSFault,
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault`

Input Parameters

- targetDevice - - target device information, which is the target name
- elementType---element type information, such as sipInterface.

Output Parameters

a list of WsConfigElements

Throws

com.acmepacket.ems.ws.service.fault.AcmeConfigWSFault
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault

applyBatch

applyBatch initiates a database batch operation; each individual requested operation is described by a wsBatch data structure.

```
public com.acmepacket.ems.ws.service.userobjects.WsConfigResult applyBatch(java.lang.String targetDevice,
com.acmepacket.ems.ws.service.userobjects.WsBatch wsBatch)throws
com.acmepacket.ems.ws.service.fault.AcmeConfigWSFault,
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault
```

Input Parameters

- targetDevice - - target device information
- wsBatch - - wsBatch data structure

Output Parameters

WsConfigResult

Throws

com.acmepacket.ems.ws.service.fault.AcmeConfigWSFault
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault

addConfigElement

addConfigElement adds a specified configuration element to the configuration database; sub-elements, if present, are also added to the database.

```
public com.acmepacket.ems.ws.service.userobjects.WsConfigResult
addConfigElement(java.lang.String targetDevice,
com.acmepacket.ems.ws.service.userobjects.WsConfigElement wsConfigElement) throws
com.acmepacket.ems.ws.service.fault.AcmeConfigWSFault,
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault
```

Input Parameters

- targetDevice - - target name information
- wsConfigElement - - configuration element to be added to database

Output Parameters

WsConfigResult

Throws

com.acmepacket.ems.ws.service.fault.AcmeConfigWSFault
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault

replace

replace replaces an existing configuration element in the configuration database; the user must define the replacement configuration element in its entirety, to include sub-elements if any exist.

```
public com.acmepacket.ems.ws.service.userobjects.WSConfigResult replace(java.lang.String targetDevice,  
com.acmepacket.ems.ws.service.userobjects.WSConfigElement wsConfigElement) throws  
com.acmepacket.ems.ws.service.fault.AcmeConfigWSFault,  
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault
```

Input Parameters

- targetDevice - - target name information
- wsConfigElement - - configuration element to be replaced

Output Parameters

WSConfigResult

Throws

com.acmepacket.ems.ws.service.fault.AcmeConfigWSFault
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault

addSubElement

addSubElement adds a new sub-element to a specified top-level configuration element.

When adding a sub-element, users must provide an unambiguous path to the target sub-element to be added. The path consists of the ACP top-level identifier, followed by one or more followed by one or more sub-element types. At each path level, primaryKeys (such as ACLI object names) must be supplied to ensure unambiguous element identification. For example, the following path

```
sipManipulation(primaryKeys)/headerRule(primaryKeys)/elementRule(name="rule2")
```

identifies a SIP element rule to be added.

```
public WSConfigResult addSubElement(String targetName, WSConfigElement parent, WSConfigElement child)  
throws AcmeConfigWSFault, AcmeAdminWSFault
```

Input Parameters

- targetDevice - - target name information
- parent - - the parent configuration element
- child - - the child sub-element to be added

Output Parameters

WSConfigResult

Throws

com.acmepacket.ems.ws.service.fault.AcmeConfigWSFault
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault

deleteSubElement

deleteSubElement deletes an existing sub-element from a specified top-level configuration element.

When deleting a sub-element, users must provide an unambiguous path to the target sub-element to be deleted. The path consists of the ACP top-level identifier, followed by one or more followed by one or more sub-element types. At each path level, primaryKeys (such as ACLI object names) must be supplied to ensure unambiguous element identification. For example, the following path

```
sipManipulation(primaryKeys)/headerRule(primaryKeys)/elementRule(name="rule2")
```

identifies a SIP element rule to be deleted.

```
public WConfigResult addSubElement(String targetName, WConfigElement parent, WConfigElement child)
throws AcmeConfigWSFault, AcmeAdminWSFault;
```

Input Parameters

- targetDevice - - target name information
- parent - - the parent configuration element
- child - - the child sub-element to be deleted

Output Parameters

WConfigResult

Throws

com.acmepacket.ems.ws.service.fault.AcmeConfigWSFault
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault

getConfigElementMetaData

getConfigElementMetaData returns a configuration element's metadata to include its attributes.

```
public WConfigElementMetaData getConfigElementMetaData(String targetName, String elementType) throws
AcmeConfigWSFault, AcmeAdminWSFault
```

Input Parameters

- targetName - - target device information
- elementType - The type of the element for which the metadata to be returned

Output Parameters

WConfigElementMetaData

Throws

com.acmepacket.ems.ws.service.fault.AcmeConfigWSFault
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault

getConfigAttributeMetaData

getConfigAttributeMetaData returns metadata. for a specified attribute of a specified configuration element.

```
public WConfigAttributeMetaData getConfigAttributeMetaData(String targetName, String elementType, String attributeName) throws AcmeConfigWSFault, AcmeAdminWSFault
```

Input Parameters

- targetName: target device information
- elementType: type of the element for which the metadata to be returned
- attributeName: name of the attribute

Output Parameters

WConfigElementMetaData

Throws

com.acmepacket.ems.ws.service.fault.AcmeConfigWSFault
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault

getValuesForReferenceAttribute

getValuesForReferenceAttribute returns the values for a specified reference attribute.

```
public ArrayList<String> getValuesForReferenceAttribute(String targetName, String elementType, String attributeName) throws AcmeAdminWSFault, AcmeConfigWSFault
```

Input Parameters

- targetName: target device information
- elementType: type of the element for which the metadata to be returned
- attributeName: The name of the attribute

Output Parameters

ArrayList<String>

Throws

com.acmepacket.ems.ws.service.fault.AcmeConfigWSFault
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault

newConfigElement

newConfigElement creates a new default instance of a specified configuration element.

```
public WConfigElement newConfigElement(String targetName, String elementType) throws AcmeConfigWSFault, AcmeAdminWSFault
```

Input Parameters

- targetName - - target name information
- elementType – The type of the element to be returned

Output Parameters

WSConfigElement

Throws

com.acmepacket.ems.ws.service.fault.AcmeConfigWSFault
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault

encryptedPassword

encryptedPassword encrypts a specified user password.

public String encryptedPassword(string configurationPasswordInfo, String inputPassword) throws
AcmeConfigWSFault, AcmeAdminWSFault

Input Parameters

- configurationPasswordInfo - - an SBC constant
- inputPassword – the plaintext password to be encrypted

Output Parameters

a string containing the encrypted inputPassword

Throws

com.acmepacket.ems.ws.service.fault.AcmeConfigWSFault
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault

deleteUserChanges

The deleteUserChanges API is used to delete any changes that a logged-in user made to a managed device.

Input Parameters

- targetName—The target name of the device on which the logged-in user made changes.

Output Parameters

- WSDeviceResult—The name of the result of deleting changes that a logged-in user made to a managed device.

Throws

com.acmepacket.ems.ws.service.fault.AcmeDeviceWSFault
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault

ACLI to ACP Mappings

Retrieving the ACLI to ACP Mapping

The ACLI to ACP mapping varies between models. To dynamically generate the ACLI to ACP mapping for your SBC version, use the `getACLItoACPMapping` API call.

The SBCs must already be added in Device Manager and loaded in Configuration Manager. If an SBC is not added in Device Manager, or added in Device Manager but not loaded in Configuration Manager, this procedure returns the error message:

```
The target device names added to OC SDM are: [null]
```

1. Unzip the `CXFClientNNCMain.zip` included as part of the Oracle Communications Session Delivery Manager download.
2. In the `CXFClientNNCMain\bin\build.bat` file, set the following variable:
 - `JAVA_HOME`—The path to your local JDK installation
3. In the `CXFClientNNCMain\bin\run.bat` file, set the following variables:
 - `JAVA_HOME`—The path to your local JDK installation
 - `SERVER_NAME`—The host name of your Oracle Communications Session Delivery Manager server.
4. Also in the `CXFClientNNCMain\bin\run.bat` file, change `sampleSource.GenericClient` to `sampleSource.ClientExample`.
 - If using `https`, change the `sampleSource.GenericClient` in line 15.
 - 👉 **Note:** If using `https`, set values for the `TRUST_STORE` and `TRUST_STORE_PASSWORD` variables.
 - If using `http`, change the `sampleSource.GenericClient` in line 20.

```
:http
"%JAVA_HOME%\bin\java" -classpath .;%CLASSPATH% -DServerName=%SERVER_NAME
% -DServerPort=%SERVER_PORT% sampleSource.ClientExample
```
5. In the `CXFClientNNCMain\sampleSource\ClientExample.java` file, under the Class Variables section, set the following private static variables:
 - `serverName`—The host name or IP address of your Oracle Communications Session Delivery Manager server.
 - `serverPort`—The port of your Oracle Communications Session Delivery Manager server
 - `soapUser`—The administrator's user name.
 - `soapUserPwd`—The administrator's password.

ACLI to ACP Mappings

- targetDevice.—The target name of the SBC.



Note: To find the target name from within the Oracle Communications Session Delivery Manager GUI, click **Configuration Manager > Devices > Expand All**. The Target Name column contains the value for the targetDevice variable.

For example:

```
private static String serverName="1.2.3.4";
private static String serverPort="8080";
private static String soapUser="admin_user";
private static String soapUserPwd="admin_password";
private static String targetDevice = "sbc720";
```

6. In the runScenarios() function, uncomment the getACLtoACPMMapping API call.



Note: Because the getACLtoACPMMapping call needs to know the devices managed by the Oracle Communications Session Delivery Manager, the runScenario2 call must also be uncommented.

```
private void runScenarios() {
    try {
        //runScenario1();           // Summary view SOAP user
        runScenario2();           // Get managed devices
        //runScenario3();         // Get top level element names
        //runScenario4();         // Create Top level element
        getACLtoACPMMapping();    // Return a list of ACLI to ACP name
                                // mapping for Top-level Elements
    }
}
```

7. From the Command Prompt, re-build and run the SOAP client.

```
C:\CXFClientNNCMain\bin>build.bat
Note: ..\sampleSource\ClientExample.java uses unchecked or unsafe
operations.
Note: Recompile with -Xlint:unchecked for details.

C:\CXFClientNNCMain\bin>run.bat
```

The ACLI to ACP mapping is displayed.

```
The target device names added to NNC are : [sbc720]
[
Name : media-manager->codec-policy Value : codecPolicy,
Name : media-manager->dns-config Value : dnsConfig,
Name : media-manager->dnsalg-constraints Value : dnsAlgConstraints,
Name : media-manager->ext-policy-server Value : extBwManager,
. . . .
```

ACLI to ACP Mapping

Table 1: ACLI to ACP Mapping

ACP Element Name	ACLI Command Sequence
media-manager > codec-policy	codecPolicy
media-manager > dnsalg-constraints	dnsAlgConstraints
media-manager > dns-config	dnsConfig
media-manager > ext-policy-server	extBwManager
media-manager > media-manager	mediaRouter
media-manager > media-policy	mediaPolicy
media-manager > msrp-config	msrpConfig

ACP Element Name	ACLI Command Sequence
media-manager > playback-config	playbackConfig
media-manager > realm-config	realmConfig
media-manager > realm-group	realmGroup
media-manager > rtcp-policy	rtcpPolicy
media-manager > static-flow	staticFlow
media-manager > steering-pool	steeringPool
media-manager > tcp-media-profile	tcpMediaProfile
ntp-sync	ntpConfig
security > authentication	authConfig
security > auth-params	authParamsConfig
security > certificate-record	certRecord
security > cert-status-profile	certStatusProfile
security > ike > data-flow	dataFlow
security > ike > dpd-params	dpdParams
security > ike > ike-certificate-profile	ikeCertificateProfile
security > ike > ike-config	ikeConfig
security > ike > ike-interface	ikeInterface
security > ike > ike-sainfo	ikeSaInfo
security > ike > local-address-pool	localAddressPool
security > ike > tunnel-orig-params	tunnelOrigParams
security > ims-aka-profile	imsAkaProfile
security > ipsec > ipsec-global-config	ipsecGlobalConfig
security > ipsec > security-association > manual	saConfig
security > ipsec > security-policy	spdConfig
security > media-security > media-sec-policy	mediaSecPolicy
security > media-security > mikey-profile	mikeyProfile
security > media-security > sdes-profile	sdesProfile
security > media-security > sipura-profile	sipuraProfile
security > password-policy	passwordPolicy
security > public-key	sshPubKeyRecord
security > security-config	securityConfig
security > tls-global	tlsGlobal
security > tls-profile	tlsProfile
session-router > access-control	accessControl

ACLI to ACP Mappings

ACP Element Name	ACLI Command Sequence
session-router > account-config	acctConfig
session-router > allowed-elements-profile	allowedElementsProfile
session-router > call-recording-server	callRecordingServer
session-router > class-profile > policy	classPolicy
session-router > diameter-manipulation	diameterManipulation
session-router > enforcement-profile	enforcementProfile
session-router > enum-config	EnumConfig
session-router > filter-config	filterConfig
session-router > h248-config	h248Config
session-router > h248-config > h248-mgc-config	h248MgcConfig
session-router > h248-config > h248-mg-config	h248MgConfig
session-router > h323 > h323-config	H323Config
session-router > h323 > h323-stacks	H323StackConfig
session-router > home-subscriber-server	extHssManager
session-router > http-alg	httpAlg
session-router > iwf-config	IwfStackConfig
session-router > ldap-config	ldapConfig
session-router > local-policy	localPolicy
session-router > local-response-map	sipLocalMap
session-router > local-routing-config	LocalRoutingConfig
session-router > media-profile	mediaProfile
session-router > mgcp-config	mgcpConfig
session-router > net-management-control	netMgmtCtrl
session-router > q850-sip-map	q850SipMap
session-router > qos-constraints	qosConstraints
session-router > rph-policy	rphPolicy
session-router > rph-profile	rphProfile
session-router > service-health	serviceHealth
session-router > session-agent	sessionAgent
session-router > session-constraints	sessionConstraints
session-router > session-group	sessionAgentGroup
session-router > session-recording-group	sessionRecordinGroup
session-router > session-recording-server	sessionRecordingServer
session-router > session-router	sessionRouter

ACP Element Name	ACLI Command Sequence
session-router > session-timer-profile	sessionTimerProfile
session-router > session-translation	sessionTranslation
session-router > sip-advanced-logging	sipAdvancedLogging
session-router > sip-config	sipConfig
session-router > sip-feature	sipFeature
session-router > sip-interface	sipInterface
session-router > sip-isup-profile	sipIsupProfile
session-router > sip-manipulation	sipManipulation
session-router > sip-monitoring	sipMonitoring
session-router > sip-nat	sipNatConfig
session-router > sip-profile	sipProfile
session-router > sip-q850-map	sipQ850Map
session-router > sip-response-map	responseMap
session-router > surrogate-agent	surrogateAgent
session-router > survivability	survivability
session-router > translation-rules	translationRules
system > auto-config	autoConfig
system > capture-receiver	captureReceiver
system > host-route	ipRoute
system > ipt-config	iptConfig
system > network-interface	networkInterface
system > network-parameters	NetworkParameters
system > phy-interface	phyInterfaceConfig
system > redundancy	RedundancyConfig
system > snmp-community	snmpCommunity
system > snmp-user-entry	snmpUserEntry
system > spl-config	splConfig
system > system-access-list	sysACL
system > system-config	systemConfig
system > trap-receiver	trapReceiver
system > web-server-config	webServerConfig

Physical Interface

The following table lists SOAP attributes and sub-elements for the physical interface.

ACLI to ACP Mappings

SOAP Attributes/Sub-elements	ACLI	Default Values	Valid Values	SBC Version
	system->phy-interface			
acliObjectName *#	name	enabled	24 characters	4.0.0 and above
admin	admin-state	50	enabled/disabled	
operationType #	operation-type		maintenance or media	
port #	port		0-3	
slot #	slot		0-1	
ae_en	auto-negotiation		enabled/disabled	
duplex	duplex-mode		full or half	
speed	speed		100 or 10	
virtualMac	virtual-mac		empty or hh:hh:hh:hh:hh:hh	
wancomHealthScore	wancom-health-score		0-100	
overloadProtection	overload-protection	disabled	enabled, disabled	6.2.0 and above
AlarmThreshold	alarm-threshold	minor	minor, major, critical	
severity*#	severity	0	0-100	
value	value			

Network Interface

The following table lists SOAP attributes and sub-elements for the network interface.

SOAP Attributes/Sub-elements	ACLI	Default Values	Valid Values	SBC Version
	system->network-interface			
acliObjectName *	name	<phy>	<phy>	4.0.0 and above
subPortId *#	sub-port-id	disabled	0-4095	
hostname	hostname	0	0-255 chars	
ipAddress	ip-address	0	Ipv4	
utilityAddress	pri-utility-addr	1	Ipv4	
secondUtilityAddress	sec-utility-addr	0	Ipv4	
netmask	netmask		Ipv4	
gateway	gateway		Ipv4	
gatewaySec	sec-gateway		Ipv4	
NetworkInterfaceGWHeartbeat	gw-heartbeat		enabled/disabled	
state #	state		0-65535	
timeout	heartbeat		0-65535	
retrycount	retry-count		1-65535	

retryTimeout	retry-timeout		0-100	
healthDec	health-score		Ipv4	
domNameServer	dns-ip-primary		Ipv4	
domNameServerB1	dns-ip-backup1		Ipv4	
domNameServerB2	dns-ip-backup2		list of IPs	
defDomainName	dns-domain		ipv4	
HipIpList	hip-ip-list		empty or combo of (ftp, icmp, snmp, telnet)	
ip *#	ftp-address		ftp HIP	
protocolParameters	icmp-address		icmp HIP	
ftpAddress	snmp-address		snmp HIP	
icmpAddress	telnet-address		telnet HIP	
snmpAddress				
telnetAddress				
dnsTimeout	dns-timeout	11	0-4294967295	
description	description		255 chars	5.0.0 and above
IcmpIpList	icmp-address		HIP	4.1.4; 5.1.0 and above
ip *#				
hostname	hostname		0-255 chars, ipv4, ipv6	CX6.2.0 and above
ipAddress	ip-address		Ipv4, ipv6, ipv6/prefix	
utilityAddress	pri-utility-addr		Ipv4, ipv6	
secondUtilityAddress	sec-utility-addr		Ipv4, ipv6	
netmask	netmask		Ipv4, not allowed for ipv6	
gateway	gateway		Ipv4, ipv6	
gatewaySec	sec-gateway		Ipv4, ipv6	
domNameServer	dns-ip-primary		Ipv4, ipv6	
domNameServerB1	dns-ip-backup1		Ipv4, ipv6	
domNameServerB2	dns-ip-backup2		Ipv4, ipv6	
HipIpList	hip-ip-list		list of IPs	
ip *#	icmp-address		ipv4, ipv6	
protocolParameters			empty or combo of (ftp, icmp, snmp, telnet)	
IcmpIpList			list of ipv4, ipv6	
ip *#				

Realm

The following table lists SOAP attributes and sub-elements for the realm.

ACLI to ACP Mappings

SOAP Attributes/Sub-elements	ACLI	Default Values	Valid Values	SBC Version
	media-manager->realm-config			
id *#	identifier	0.0.0.0	24 characters	4.0.0 and above
addrPrefix	addr-prefix	0	0.0.0.0 or ipv4 or ipv4/mask	
parent	parent-realm	disabled	empty or <realm>	
RealmNetworkInterfaceId	network-interfaces	enabled	<phy:port-id> or lo0:0	
acliObjectName *#	max-bandwidth	disabled	0-999999999	
subPortId *#	mm-in-realm	disabled	enabled/disabled	
maxBandwidth	mm-in-network	none	enabled/disabled	
mrInRealm	msm-release	0	enabled/disabled	
mrInNetwork	qos-enable	0	enabled/disabled	
isMSMRelease	media-policy	0	empty or <QoS marking profile>	
qosEnable	in-translationid	30	empty or <trans-profile>	
mediaPolicy	out-translationid		empty or <realm>	
intranslationid	dns-realm		empty or <cls-profile>	
outtranslationid	class-profile		none, low, medium, high	
dnsRealm	access-control-trust-level		0-999999999	
classProfile	average-rate-limit		0-999999999	
trustLevel	invalid-signal-threshold		0-999999999	
rateLimit	maximum-signal-threshold		0-999999999	
errMsgThreshold	deny-period			
denyTimer				
symmetricLatching	symmetric-latching	disabled	enabled/disabled	
paiStrip	pai-strip	disabled	enabled/disabled	
trunkContext	trunk-context	0	empty or <pol-server>	
bwManager	ext-policy-svr		empty or <sip-manipulation>	
inManipulationId	in-manipulationid		list of options separated by comma	
outManipulationId	out-manipulationid		0-999999999	
options	options			
maxMsgThresholdUntrusted	untrusted-signal-threshold			
mmInSystem	mm-in-system	enabled	enabled/disabled	4.0.1 and above
restrictedLatching	restricted-latching	none	none, sdp, peer-ip	
restrictionMask	restriction-mask	32	1-32	
mmSameIp	mm-same-ip	enabled	enabled/disabled	

earlyMediaAllow	early-media-allow		empty, none, reverse, or both	
RealmAdditionalPrefixes prefix *#	additional-prefixes		list of ipv4/mask ipv4/mask	
acctEnable	accounting-enable	enabled	enabled, disabled	4.1.0 and above
netMgmtCtrl userCacMode	net-management-control	disabled	enabled or disabled	4.1.1 and above
userCacBandwidth	user-cac-mode	none	none, aor, ip	
userCacSessions	user-cac-bandwidth	0	0-999999999	
delayedMediaUpdate	user-cac-sessions	0	0-999999999	
nonMmBwCAC	delay-media-update	disabled	enabled/disabled	
codecPolicy	bw-cac-non-mm	disabled	enabled/disabled	
codePolicyInRealm	codec-policy codec-manio-in-realm	disabled	empty or <codec-policy> enabled/disabled	
generateUDPCksum enforcementProfile	generate-udp-checksum enforcement-profile	disabled	empty or disabled empty or <enforcement-profile>	4.1.4; 5.1.0 and above
monthlyMinutes constraintName	monthly-minutes constraint-name	0	0-71582788 empty or <session-constraint>	4.1.4 and 5.1.1
referCallTransfer description	refer-call-transfer description	disabled	enabled, disabled	5.1.1 and above
callRecordingServerId	call-recording-server-id		empty, defined call recording server	6.0.0 and above
hmrString	manipulation-string	0	0-999999999	6.1.0 and above
maxPriorityBandwidth	max-priority-bandwidth	0	0-999999999	
fallbackBandwidth	fallback-bandwidth	0	0-999999999	
icmpDetectMultip	icmp-detect-multiplier	0	0-999999999	
icmpAdvInterval	icmp-advertisement-interval	0	ipv4	
icmpTargetIp	icmp-target-ip	disabled	list of <media-profile>, <media-profile>::, <media-profile>::<subname>, *, *::, *::<subname>	
MatchMediaProfile	match-media-profiles	0.0.0.0		
acliObjectName *#		3478		
natTrustThreshold	nat-trust-threshold	0.0.0.0	0-65535	
stunEnable	stun-enable	3479	enabled, disabled	
stunServerIp	stun-server-ip		ipv4	
stunServerPort	stun-server-port		1025-65535	
stunChangedIp	stun-changed-ip		ipv4	
stunChangedPort	stun-changed-port		1025-65535	
qosConstraintName	qos-constraint		empty or <qos constraint>	

ACLI to ACP Mappings

sipProfile sipIsupProfile referCallTransfer dynReferTerm cacFailThreshold untrustedCacFailThreshold manipPattern	sip-profile sip-isup-profile refer-call-transfer dyn-refer-term cac-failure-threshold untrust-cac-failure-threshold manipulation-pattern	disabled disabled 0 0	empty or <sip-profile> empty or <sip-isup-profile> disabled, enabled, dynamic enabled, disabled 0-999999999 0-999999999	6.2.0 and above
mediaSecPolicy addrPrefix xnqState hairpinId	media-sec-policy addr-prefix xnq-state hairpin-id	xnq-unknown	empty or <media-sec-policy> ipv4 or ipv4/mask, ipv6, ipv6/mask xnq-unknown, xnq-potential, xnq-remove 0-65535	CX6.2.0 and above

Realm Media Address

The following table lists SOAP attributes and sub-elements for the realm media address.ip-

SOAP Attributes/Sub-elements	ACLI	Default Values	Valid Values	SBC Version
	media-manager->steering-pool			
ipAddress *# startPort *# endPort # realmID	ip-address start-port end-port realm-id	<realm>	Ipv4 1025-65535 1025-65535, endPort > startPort <realm>	4.0.0 and above
RealmNetworkInterfaceId acliObjectName subPortId	network-interface		empty or <phy:port-id>	
ipAddress *#	ip-address		ipv4, ipv6	CX6.2.0 and above

Surrogate Agent

The following table lists SOAP attributes and sub-elements for the surrogate agent.

SOAP Attributes/Sub-elements	ACLI	Default Values	Valid Values	SBC Version
	session-router->surrogate-agent			

registerHost *#	register-host	enabled	Ipv6 or hostname	4.0.1 and above
registerUser *#	register-user	<realm>	enabled/disabled	
state	state	600000	<realm>	
realmID *	realm-id	disabled	Ipv4 or hostname	
description	description	enabled	SAG:<sag> or <sa> or ipv4 or hostname	
customerHost	customer-host	1		
customerRoute #	customer-next-hop		Ipv4 or hostname	
contactHost #	register-contact-host		0-999999999	
contactUser #	register-contact-user		enabled/disabled	
password	password		enabled/disabled	
expires	register-expires		0-999999999	
replaceContact	replace-contact		list of comma separated options	
routeToRegistrar	route-to-registrar			
count	aor-count			
authUser	auth-user			
options	options			
maxRegisterAttempts	max-register-attempts	3	0-10	4.1.4; 5.1.0 and above
registerRetryTime	register-retry-time	300	30-3600	
countStart	count-start	1	0-999999999	

SIP Interface

The following table lists SOAP attributes and sub-elements for the SIP interface.

SOAP Attributes/Sub-elements	ACLI session-router->sip-interface	Default Values	Valid Values	SBC Version
state	state	enabled	enabled/disabled	4.0.0 and above
RealmID *	realm-id	<realm>	<realm>	
trustMode	trust-mode	all	all, agents-only, relam-prefix, registered	
SIPConfigurationSIPPort	sip-port	5060	ipv4	
address *#	address	UDP	1025-65535	
port *#	port	all	UDP or TCP	
transProtocol *#	transport-protocol	none	all, agents-only, realm-prefix, registered, register-prefix	
anonMode	allow-anonymous	disabled	empty, proxy, redirect, record-route, stateless	
proxyMode	proxy-mode	none	empty, proxy, recurse	
redirectAct	redirect-action	30		
contactSip	contact-mode	300		

ACLI to ACP Mappings

telUri	teluri-scheme	3600	none, maddr, strict, loose	
fqdnDomain	uri-fqdn-domain	disabled	enabled, disabled	
natTraversal	nat-traversal	disabled	none, always, rport	
natInterval	nat-interval		0-999999999	
regMinExpire	min-reg-expire		1-999999999	
regInterval	registration-interval		0-999999999	
registrationCaching	registration-caching		enabled/disabled	
isRouteReg	route-to-registrar		enabled/disabled	
SIPConfigurationCarriers	carriers		list of carriers	
acliObjectName *# options	options		list of options seperated by comma	
natmaxInterval	max-nat-interval	3600	enabled, disabled	
natIntervalIncrement	nat-int-increment	10		
natTestIncrement	nat-test-increment	30		
sipdynamicHnt	sip-dynamic-hnt	disabled		
stopRecurse	stop-recurse	401,407	list of response codes, 300-599 seperated by comma	
portMapStart	port-map-start	0	0, 1025-65535	
portMapEnd	port-map-end	0	0, 1025-65535, end>start	
trustMode	trust-mode	disabled	all, agents-only, relam-prefix, registered, none	
extPolicyServer	ext-policy-server	None	empty or <pol-server>	
defaultLocationString	default-location-string	None	enable/disabled	
imsFeature	sip-ims-feature	pass	NONE (4.0.1), none (4.2), normal, non-urgent, urgent, emergency	
NetworkID	network-id	pass	none, iptel, egress-uri	
anonymousPriority	anonymous-priority		none, pass, delete, insert	
termTgrpMode	term-tgrp-mode		none, pass, delete, insert	
chargingVectorMode	charging-vector-mode		none, pass, delete, insert	
chargingFuncAddrMode	charging-function- address-mode		none, pass, delete, insert	
ecfAddress	ecf-address		Ipv4 or hostname	
ccfAddress	ccf-address		Ipv4 or hostname	
operatorIdentifier	operator-identifier		empty or <sip-manipulation>	
inManipulationId	in-manipulationid			
outManipulationId	out-manipulationid			
implicitServiceRoute	implicit-service-route	disabled	strict, enabled, disabled	4.0.1 and above

tcpNatInterval	tcp-nat-interval	90	0-99999999	4.1.0 and above
isSecure	secured-network	disabled	enabled, disabled	
rfc2833-mode	rfc2833-mode	transparent	transparent, preferred	
rfc2833-payload	rfc2833-payload	101	96-127	
maxConnPerPeer	per-src-ip-max-incoming-conns	0	0-20000	
maxConn	max-incoming-conns	0	0-20000	
idleConnTimeout	inactive-conn-timeout	0	0-999999999	
SIPConfigurationSIPPort	sip-port		UDP, TCP, TLS	
transProtocol	transport-protocol		<tls-profile> if TLS specified	
tlsProfile	tls-profile			
constraintName	constraint-name	disabled	empty or <session-constraint>	4.0.1; 4.1.1 and above
implicitServiceRoute	implicit-service-route		strict, enabled, disabled	
responseMap	response-map		empty or <sip-response-map>	4.1.1 and above
localresponseMap	local-response-map			
trans-expire	trans-expire		0-999999999	
invite-expire	invite-expire		0-999999999	
max-redirect-contacts	max-redirect-contacts		0-10	
rfc2833-mode	rfc2833-mode		transparent, preferred, dual	
untrustedConnTimeout	untrusted-conn-timeout	0	0-999999999	5.0.0 and above
tcp-keepalive	tcp-keepalive	none	none, disabled, enabled	4.1.4; 5.1.0 and above
chargingFuncAddrMode	charging-function-address-mode	pass	none, pass, delete, insert, delete-and-respond, insert-reg-cache	
enforcementProfile	enforcementProfile		empty or <enforcement profile>	
add-sdp-invite	add-sdp-invite	disabled	disabled, invite, reinvoke	4.1.4; 5.1.1 and above
SIPInterfaceMediaProfile	add-sdp-profile		list of media profiles	
referCallTransfer	refer-call-transfer	disabled	enabled, disabled	5.1.1 and above
routeUnauthorizedCalls	route-unauthorized-calls		empty, defined SA or SAG	
description	description			
implicitServiceRoute	implicit-service-route	disabled	strict, enabled, disabled, absent, replace	6.0.0 and above
chargingVectorMode	charging-vector-mode	pass	none, pass, delete, insert, delete-and-respond	6.0.0M1 and above
imsAkaFeature	ims-aka-feature	disabled	enabled, disabled	6.1.0 and above
imsAkaProfile	ims-aka-profile	UDP	<ims-aka-profile> or empty	
hmrString	manipulation-string		UDP, TCP, TLS, SCTP	

ACLI to ACP Mappings

SIPConfigurationSIPPort transProtocol	sip-port transport-protocol			
sipProfile sipIsupProfile manipPattern	sip-profile sip-isup-profile manipulation-pattern		empty or <sip-profile> empty or <sip-isup-profile>	6.2.0 and above
SIPConfigurationSIPPort address *#	sip-port address		ipv4, ipv6	CX6.2.0 and above

SIP NAT

The following table lists SOAP attributes and sub-elements for the SIP NAT.

SOAP Attributes/ Sub-elements	ACLI Session-router->sip- nat	Default Values	Valid Values	SBC Version
RealmID *	realm-id	<realm>	<realm>	4.0.0 and above
ProxyAddress	ext-proxy-address	5060	<0.0.0.0> or ipv4	
ProxyPort	ext-proxy-port	0	1025-65535	
ExternalAddress	ext-address	disabled	Ipv4	
HomeAddress	home-address	disabled	Ipv4	
HomeProxyAddress	home-proxy-address	-acme-	Ipv4	
HomeProxyPort	home-proxy-port	ACME-	1025-65535	
RouteHomeProxy	route-home-proxy	none	enabled, disabled, forced	
prefix	address-prefix	list of nat headers: Call-ID Contact f From I Join m rRecord-Route Refer-To Replaces Replay-To Route t To v Via	empty, *, 0.0.0.0, ipv4/ bitmask	
TunnelRedirect	tunnel-redirect		enabled/disabled	
UserNATTag	user-nat-tag		<.com>	
HostNATTag	host-nat-tag		none, from-to, all	
DomainSuffix	domain-suffix		list of nat headers: Call-ID Contact f From I Join m r Record-Route Refer-To Replaces Replay-To Route t To v Via with values = NAT, fqdn-ip-ext, fqdn-ip-tgt, ip- ip-ext, ip-ip-tgt	
ParamMode	use-url-parameter			
ParamName	parameter-name			
SipNatHeaders	headers			
acliObjectName*#				

H.323 Stack

The following table lists SOAP attributes and sub-elements for the H.323 stack.

SOAP Attributes/Sub- elements	ACLI	Default Values	Valid Values	SBC Version
----------------------------------	------	-------------------	--------------	----------------

	Session-router->h323->h323-stack			
acliObjectName *#	name	enabled	24 characters	4.0.0 and above
state	state	<realm>	enabled/disabled	
realm-id *	realm-id	0.0.0.0	<realm>	
assoc-stack	assoc-stack	1719	empty or <h323-stack>	
local-ip	local-ip	1720	empty or <HIP>	
ras-port	ras-port	200	1025-65535	
q931-port	q931-port	200	1025-65535	
H323AlternateTransport	alternate-transport	6	list of <ipv4:port>	
ipAddress *#	q931-max-calls	0	Ipv4:port	
q931-max-calls	max-calls	0	>0	
max-calls	max-channels	0	>0, must > q931-max-calls	
max-channels	q931-start-port	0	>0	
q931-start-port	q931-number-ports	disabled	0-65535	
q931-number-ports	dynamic-start-port	enabled	0, 1024, 2048, 4096, 8192, 16384, 32768	
dynamic-start-port	dynamic-number-ports	all	0-65535	
dynamic-number-ports	tcp-keepalive		0, 1024, 2048, 4096, 8192, 16384, 32768	
tcp-keepalive	isgateway		enabled/disabled	
isgateway	allow-anonymous		enabled/disabled	
AnonMode	filename		enabled/disabled	
filename	terminal-alias		all, agents-only, realm-prefix	
H323TerminalAial	prefixes		list of e164, url, h323-ID, email, ipAddress	
e164 *#			list of e164, url, h323-ID, email, ipAddress for gateway only	
url *#				
ipAddress *#				
email *#				
h323-ID *#				
H323Prefixes				
e164 *#				
url *#				
ipAddress *#				
email *#				
h323-ID *#				
registration-ttl	registration-ttl	120	>0 for gateway only	
processRegistration	process-registration	disabled	enabled/disabled for gatekeeper only	

ACLI to ACP Mappings

proxy-mode	proxy-mode	connect	NONE, H225, H245	
h245-stage	h245-stage	disabled	CONNECT, SETUP, ALERTING, CALL PROCEEDING, DYNAMIC, FACILITY, SETUP or CONNECT, NONE	
h245-tunneling	h245-tunneling	disabled		
stack-options	options	enabled		
auto-gk-discovery	auto-gk-discovery	disabled	enabled/disabled	
multicast	multicast	disabled	list of options seperated by comma	
gatekeeper	gatekeeper		enabled, disabled	
gk-identifier	gk-identifier		ipAddress=ipv4:port for gateway only	
callStartFast	call-start-fast		ipAddress=ipv4:port	
callStartSlow	call-start-slow		enabled/disabled	
H323MediaProfile	media-profiles		enabled/disabled, each time only fast or only slow enabled	
acliObjectName *#	fs-in-first-msg			
fs-in-first-msg			list of media-profiles enabled/disabled	
rfc2833-mode	rfc2833-mode	transparent	transparent, preferred	4.1.0 and above
description	description			5.1.1 and above
H323StackAlarmThreshold	alarm-threshold	minor	minor, major, critical	6.2.0 and above
severity*#	severity	0	0-100	
value	value			

MGCP Config

The following table lists SOAP attributes and sub-elements for the MGCP config.

SOAP Attributes/Sub-elements	ACLI Session-router->mgcp-config	Default Values	Valid Values	SBC Version
realmPrivate *	private-realm	<realm>	<realm>	4.0.0 and above
addressPrivate *#	private-address	2727	Ipv4	
portPrivate	private-port	LineUnit	1025-65535	
mode	mode	256	Host, LineUnit, LinePrefix, FQDN, FQDN2, OnlyHost	
divisor	divisor	disabled		
unitPrefix	unit-prefix	disabled	256, 65536, 16777216, 4294967295 enabled/disabled	
dnsAuthentication	dns-authentication	0	disabled	
dnsTranslation	dns-translation	disabled	empty or <trans-pfl>	
natTraversal	nat-traversal	0	enabled/disabled	

auditInterval	audit-interval	0.0.0.0	list of options	
options	options	2427	enabled/disabled	
caRedundancy	ca-redundancy	0<realm>	NTFY 1 ping@host	
caPingMethod	ca-ping-method	0.0.0.0	Ipv4/mask	
caPingInterval	ca-ping-interval	2727	1025-65535	
hostGWPublic	public-gw-host		0, 1025-65535<realm>	
addrGWPublic	public-gw-address		Ipv4	
portGWPublic	public-gw-port		1025-65535	
portGWPublic2realmPublic	second-public-gw-portpublic-realm			
pubCAHost	public-ca-host			
addrCAPublic	public-ca-address			
portCAPublic	public-ca-port			
portALG	alg-port	2427	1025-65535	4.1.4; 5.1.1 and below
MGCPConfigIpAddresses Addr *#	ca-failover-ip-addresses		List of ipv4 Ipv4	4.0.1 and above
rsipFailures	rsip-failures	500-509 511-519 522-599	500-599	5.1.0 and above
portMapStart	port-map-start	0	0 or 1025-65535	5.1.1
portMapEnd	port-map-end	0	0 or 1025-65535	
caPingRetries	ca-ping-retries	0	0-4294967295	SC6.1.0M1

DNS Config

The following table lists SOAP attributes and sub-elements for the DNS config.

SOAP Attributes/Sub-elements	ACLI	Default Values	Valid Values	SBC Version
	media-manager->dns-config			
clientrealmID *#	client-realm	<realm>	<realm>	4.0.0 and above
description	description	53	Ipv4	
ClientIpList	client-address-list	10	<realm>	
IPAddress *#	server-dns-attributes		domain	
ServerDNSAttributes	serverrealmID		ipv4	
serverRealmID *#	domain-suffix		ipv4	
ServerDnsDomainSuffix	server-address-list		0-65535	
acliObjectName * #	source-address		0-999999999	

ACLI to ACP Mappings

ServerDnsAddressList	source-port		ipv4/mask	
IPAddress * #	transaction-timeout		ipv4/mask	
sourceAddress *#	address-translation			
sourcePort *#	server-prefix			
transactionTimeout	client-prefix			
ServerDNSAddressTranslation				
serverprefix *#				
clientprefix *#				

Session Agent

The following table lists SOAP attributes and sub-elements for the session agent.

SOAP Attributes/Sub-elements	ACLI	Default Values	Valid Values	SBC Version
	session-router->session-agent			
hostname *#	hostname	5060	FQDN or ipv4	4.0.0 and above
ipAddress #	ip-address	enabled	0.0.0.0 or Ipv4	
port	port	UDP	0, 1025-65535	
state	state	disabled	enabled/disabled	
appProtocol #	app-protocol		SIP or H323	
appType	app-type	disabled	H323-GK or H323-GW for H323 only	
transMethod	transport-method	0	UDP, UDP+TCP, dynamicTCP, staticTCP	
realmID	realm-id	0	empty or <realm>	
description	description	0	list of comma seperated options	
options	options	0	List of <media-profiles> for H323 only	
SessionAgentMediaProfile	media-profiles	0	<media-profile>	
acliObjectName *#	carriers	0	List of carriers	
SessionAgentCarriers	allow-next-hop-ip	0	<carrier-code>	
acliObjectName *#	in-translationid	0	enabled/disabled	
allowNextHop	out-translationid	0	empty or <translation-profile>	
inTranslationId	constraints		enabled/disabled	
outTranslationId	max-sessions		empty or <translation-profile>	
useConstraints	max-outbound-sessions		enabled/disabled	
maxNumSessions	max-burst-rate		0-999999999	
maxOutbSessions	max-sustain-rate		0-999999999	
maxBurstRate	time-to-resume		0-999999999	
maxSustainedRate	ttr-no-response		0-999999999	

timeToResume	in-service-period		0-999999999	
noResponseTo	burst-rate-window		0-999999999	
inServicePeriod	sustain-rate-window		0-999999999	
burstWindow			0-999999999	
sustainedWindow			0-999999999	
trustMe	trust-me	enabled	enabled/disabled	4.0.0 and above (for SIP only)
proxyMode	proxy-mode	enabled	empty, proxy, redirect, record-route	
redirectAct	redirect-action	disabled	empty, proxy, recurse	
allocMedia	send-media-session	None	enabled/disabled	
responseMap	response-map	0	empty or <sip-response-code-profile>	
looseRouter	loose-routing		enabled, disabled	
inclCarrierAs	req-uri-carrier-mode		None, URI-param, Prefix	
pingMethod	ping-method		INFO, OPTIONS	
pingInterval	ping-interval		0-999999999	
localResponseMap	local-response-map		empty or <sip-response-code-profile>	
pingToUserPart	ping-to-user-part		list of headers	
pingFromUserPart	ping-from-user-part		list of response codes, 300-599	
RequesturiHeader	request-uri-headers			
acliObjectName *#	stop-recurse			
stopRecurse				
trustMeForLI	li-trust-me	disabled	enabled/disabled	
assertedID	p-asserted-id	0	sip:name@acme.com or tel:+1234	
SessionAgentTrunkGroup	trunk-group		list of trunk groups or group:context	
acliObjectName *#	in-manipulationid		empty or <sip-manipulation>	
inManipulationId	out-manipulationid		0-999999999	
outManipulationId	max-register-sustain-rate			
maxRegisterSustainedRate				
earlyMediaAllow	early-media-allow	disabled	empty, none, reverse, both	4.0.1 and above (for SIP only)
invalidateRegistrations	invalidate-registrations		enabled/disabled	
minSeizure	min-seizures	5	1-999999999	4.0.1 and above
minAnswerSeizureRatio	min-asr	0	0-100	
rfc2833-mode	rfc2833-mode	none	none, transparent, preferred	4.1.0 and above for H323
rfc2833-payload	rfc2833-payload	0	0, 96-127	

ACLI to ACP Mappings

maxInbSessions	max-inbound-sessions	0	0-999999999	4.0.1; 4.1.1 and above
maxInbBurstRate	max-inbounds-burst-rate	0	0-999999999	
maxOutbBurstRate	max-outbound-burst-rate	0	0-999999999	
maxInbSustainedRate	max-outbound-sustain- rate	0	0-999999999	
maxOutbSustainedRate		0	0-999999999	
codecPolicy	codec-policy		empty or <codec-policy>	4.1.1 and above (for SIP only)
inServiceCodes	in-service-response-codes		list of comma-separated response codes, 200-699	
outServicecodes	out-service-response- codes			
rfc2833-mode	rfc2833-mod	None	none, transparent, preferred, dual	
rfc2833-payload	rfc2833-payload	0	0, 96-127	
reuse-connections	reuse-connections	NONE	NONE, TCP	4.1.4; 5.1.0 and above (for SIP only)
tcp-keepalive	tcp-keepalive	none	none, disabled, enabled	
tcp-reconn-Interval	tcp-reconn-interval	0	0, 2-300	
enforcementProfile	enforcement-profile		empty or <enforcement-profile>	
maxRegisterBurstRate	max-register-burst-rate	0	0-999999999	4.1.4, 5.1.1 and above (for SIP only)
registerBurstWindow	register-burst-window	0	0-999999999	
referCallTransfer	refer-call-transfer	disable d	enabled, disabled	5.1.1 and above (for SIP only)
pingSendMode	ping-send-mode	keep- alive	keep-alive, continuous	
egressRealmID	egress-realm-id		empty or <realm>	
SessionAgentRateConstraint s	rate-constraints		INVITE, ACK, BYE, REGISTER, CANCEL, PRACK, OPTIONS, INFO, SUBSCRIBE, NOTIFY, REFER, UPDATE, MESSAGE, PUBLISH	5.1.1 and above
method	method			
maxInBurstRate	max-inbound-burst-rate		0-999999999	
maxOutBurstRate	max-outbound-burst-rate		0-999999999	
maxInSustainedRate	max-inbound-sustain-rate		0-999999999	
maxOutSustainedRate	max-outbound-sustain- rate		0-999999999	
hmrString	manipulation-string	UDP	UDP, UDP+TCP, dynamicTCP, staticTCP, dynamicTLS, staticTLS, staticSCTP, NONE, TCP, SCTP	6.1.0 and above (for SIP only)
transMethod	transport-method	NONE		
reuse-connections	reuse-connections			
pingAllAddresses	ping-all-addresses	disable d	enabled, disabled	6.2.0 above (for SIP only)
sipProfile	sip-profile		empty or <sip-profile>	
sipIsupProfile	sip-isup-profile	disable d	empty or <sip-isup-profile>	
manipPattern	manipulation-pattern		disabled, enabled, dynamic	

referCallTransfer	refer-call-transfer			
hostname *#	hostname		fqdn, ipv4, ipv6	CX6.2.0 and above
ipAddress	ip-address		ipv4, ipv6	

Session Agent Group

The following table lists SOAP attributes and sub-elements for the session agent group.

SOAP Attributes/Sub-elements	ACLI	Default Values	Valid Values	SBC Version
	session-router->session-agent-group			
acliObjectName *#	group-name	enabled	enabled/disabled	4.0.0 and above
description	description	SIP	SIP or H323	
state	state	hunt	hunt, roundrobin, leastbusy, propdist, lowsusrate	
protocol #	app-protocol		list of session-agents	
strategy	strategy		<sa name>	
SessionAgentGroupSipDest	dest			
hostname *#				
SessionAgentGroupTrunkGroup	trunk-group		List of trunk groups or group:context	4.0.0 and above (for SIP only)
acliObjectName *#			<trk> or <trk>:<contxt>	
sagRecurse	sag-recursion	disabled	enabled, disabled	4.1.1 and above (for SIP only)
stopRecurse	stop-sag-recursion	410, 407	list of comma-separated response codes, 300-599	

Local Policy

The following table lists SOAP attributes and sub-elements for local policy.

SOAP Attributes/Sub-elements	ACLI	Default Values	Valid Values	SBC Version
	Session-router->local-policy			
routeName *	from-address	*	Route0, Route1.....	4.0.0 and above
LocalPolicyFrom #	to-address	*	+number, number, *, fqdn, ipv4 or 0.0.0.0	
addr *#	source-realm	*	same	
LocalPolicyTo #	activate-time	enabled	* or <realm>	
addr *#	deactivate-time	none	yyyy-mm-dd-hh:mm:ss	
LocalPolicySourceRealm #	state	enabled	yyyy-mm-dd hh:mm:ss	
	policy-priority	0000		

ACLI to ACP Mappings

acliObjectName *#	policy-attribute	2400	enabled/disabled	
activateTime	state	U-S	none, normal, non-urgent, urgent, emergency	
deactivateTime	start-time	0	RP0, RP1.....	
state	end-time	SIP	enabled, disabled	
anonymousPriority	days-of-week	enabled	0000-2400	
LocalPolicyAttribute	cost		0000-2400	
policyName *	media-profiles		M, T, W, R, F, S, U, H or any combination	
state	carrier		0-999999999	
startTime	next-hop		list of media-profiles	
endTime	realm		<media-profile>	
dow	app-protocol		<carrier-code>	
cost	replace-uri		SAG:<sag>, <sa>, ipAddress, FQDN	
LocalPolicyMediaProfiles			empty or <realm>	
acliObjectName *#			SIP or H323	
carrierName			enabled/disabled	
nextHop #				
destRealm #				
appProtocol				
replace				
LocalPolicyAttribute	policy-attribute	0000	0000-2400	4.0.1 and above
startTime	start-time	0000	0000-2400	
endTime	end-time	U-S	M, T, W, R, F, S, U, H or any combination	
dow	days-of-week	0	0-999999999	
cost	cost	SIP	List of media-profiles	
LocalPolicyMediaProfiles	media-profiles	none	<media-profile>	
acliObjectName *#	carrier	disabled	<carrier>	
carrierName	next-hop		SAG:<sag>, <sa>, enum:<name>	
nextHop #	realm		empty or <realm>	
destRealm #	app-protocol		SIP or H323	
appProtocol	action		none, replace-uri, redirect	
action	terminate-recursion		enabled/disabled	
isTermRoute	replace-uri			
replace				
LocalPolicyFrom #	from-address		also support 12*34, ***** for number (potsstar)	4.0.1, 4.1.1 and above
addr *#	to-address			

LocalPolicyTo # addr *#				
LocalPolicyAttribute nextHop	policy-attribute next-hop		SAG:<sag>, <sa>, enum:<name>, lrt:<name>, enum:<name>;key=<value>, lrt:<name>;key=<value>, ipAddress, FQDN	4.1.1 and above
LocalPolicyAttribute nextHop	policy-attribute next-hop		SAG:<sag>, <sa>, enum:<name>, lrt:<name>, enum:<name>;key=<value>, lrt:<name>;key=<value>, ipAddress, FQDN, ldap:<name>	4.50 and 4.5.1
description	description			5.1.1 and above
LocalPolicyAttribute methods LocalPolicyFrom # addr *# LocalPolicyTo # addr *#	policy-attribute methods from-address to-address		space separated list of INVITE, REGISTER, PRACK, OPTIONS, INFO, SUBSCRIBE, NOTIFY, REFER, UPDATE, MESSAGE, PUBLISH also supports DS: 123#456*Ab (alpha-numeric-dtmf, a combination of A-D, a-d, 0-9, #, *) (potsstar)	6.1.0 and above
LocalPolicyFrom # addr *# LocalPolicyTo # addr *# LocalPolicyAttribute eLocStrLkup eLocStrMatch lookup nextKey	from-address to-address policy-attribute eloc-str-lkup eloc-str-match lookup next-key	* * disable d single	+number (e164), number(pots), num**num (potsstar), */fqdn/ipv4/0.0.0.0 (hostname), DS:[A-D][a-d][0-9]**(potsstar), urn:service:[sos, sos.fire, sos.animal-control] (hostname) enabled, disabled 24 chars such as noc, lac, line-code single, multi \$TO, \$FROM, \$PAI or any string	6.2.0 and above
LocalPolicyFrom # addr *# LocalPolicyTo # addr *# LocalPolicyAttribute nextHop	from-address to-address policy-attribute next-hop	* *	+number (e164), number(pots), num**num (potsstar), */fqdn/ipv4/ipv6/0.0.0.0 (hostname), DS:[A-D][a-d][0-9]**(potsstar), urn:service:[sos, sos.fire, sos.animal-control] (hostname) SAG:<sag>, <sa>, enum:<name>, lrt:<name>, enum:<name>;key=<cic rn value>, lrt:<name>;key=<cic rn value>, ipv4, ipv6, FQDN	CX6.2.0 and above

Network Management Control

The following table lists SOAP attributes and sub-elements for network management control.

ACLI to ACP Mappings

SOAP Attributes/ Sub-elements	ACLI	Default Values	Valid Values	SBC Version
	Session-router->net- management-control			
acliObjectName *#	name	enabled	24 characters	4.1.1 and above
state	state	0	enabled, disabled	
aclitype	type	0	empty, gap-rate, gap-percent, priority	
value	value	0	-1, 0-100 for gap-percent, 0-2147483647 for gap-rate	
treatment	treatment	503	empty, reject, divert	
nextHop	next-hop	63	empty, hostname:port, ipv4:port, sa, SAG:sa	
nextHopRealm	realm-next-hop		empty, <realm>	
nextHopProtocol	protocol-next-hop		empty, <realm>	
statusCode	status-code		empty, SIP, H323	
causeCode	cause-code		1-699	
gapRateMaxCount	gap-rate-max-count		0-999999999	
gapRateWindowSize	gap-rate-window-size		0-999999999	
DestinationId	destination-identifier		0-999999999	
destinationId *#			List of number(^as wildcard), prefix(^as wildcard), ipv4(^as wildcard) and fqdn	
rphFeature	rph-feature		disabled or enabled	4.1.4
rphProfile	rph-profile		empty or <rph-profile>	
rphPolicy	rph-policy		empty or <rph-policy>	
destinationId	destination-identifier		list of number, prefix, ipv4 or fqdn (^ as wildcard for digit), urn:service: (sos, sos.fire, sos.animal-control etc.)	6.2.0 and above
destinationId *#				

SIP Header Manipulation

The following table lists SOAP attributes and sub-elements for SIP header manipulation.

SOAP Attributes/ Subelements	ACLI	Default Values	Valid Values	SBC Version
	session-router- >sip- manipulation			
acliObjectName * #	name	none	add, delete, manipulate, none	4.0.0 and above
HeaderRule	header-rule	any	empty or <value>	
acliObjectName * #	name	none	any, request, reply	
action	action	any	empty or comma separated strings	
	match-value			
	msg-type			

matchValue	methods		header-value, header-param, uri-user, uri-host, uri-port, uri-param, uri-header, uri-user-param	
msgType	element-rule			
methods	name		add, replace, delete-header, delete-element, none	
ElementRule	type			
acliObjectName	action		IP, FQDN, ANY	
*#	match-val-type		empty or <255 characters>	
aclitype	match-value		empty or combination of \$ORIGINAL, \$LOCAL_IP, \$REMOTE_IP, \$REMOTE_VIA_HOST, \$STRUNK_GROUP, \$STRUNK_GROUP_CONTEXT, <any string>, +, -, +^, -^	
action	new-value			
matchValueType				
matchValue				
newValue				
acliObjectName *	name	none	255 characters	4.1.1 and above
#	description	case-sensitive	255 characters or @ status-line	
description	header-rule		add, delete, manipulate, store, none	
HeaderRule	name	any	case-sensitive, case-insensitive, pattern-rule	
acliObjectName *	header-name	none	empty or <reg-expr>	
#	action	ANY	any, request, reply	
headerName #	comparison-type	case-sensitive	empty or <reg-expr>	
action	match-value		empty or comma-seperated strings	
cmpType	msg-type		255 characters	
matchValue	new-value		255 characters	
msgType	methods		header-value, header-param-name, header-param, uri-display, uri-user, uri-user-param, uri-host, uri-port, uri-param-name, uri-param, uri-header-name, uri-header, status-code, reason-phrase,	
newValue	element-rule			
methods	name		add, replace, delete-header, delete-element, store, none	
ElementRule	parameter-name		IP, FQDN, ANY	
acliObjectName	type		case-sensitive, case-insensitive, pattern-rule	
*#	action		empty or <reg-expr-value>	
paramName	comparison-type		empty or <reg-expr-value> with pre0defined parameters:	
aclitype	match-value		\$ORIGINAL, \$LOCAL_IP, \$REMOTE_IP, \$REMOTE_VIA_HOST, \$STRUNK_GROUP, \$STRUNK_GROUP_CONTEXT	
action	new-value			
matchValueType				
cmpType				
matchValue				
newValue				
Header Rule	header-rule	none	add, delete, manipulate, store, none, sip-manip	4.1.4; 5.1.1 and above
action	action			
cmpType	comparison-type			

ACLI to ACP Mappings

newValue	new-value		case-sensitive, case-insensitive, pattern-rule, boolean, refer-case-sensitive, refer-case-insensitive	
ElementRule	element-rule			
cmpType	comparison-type		empty, <reg-expr>, or <sip-manipulation>	
newValue	new-value		case-sensitive, case-insensitive, pattern-rule, boolean, refer-case-sensitive, refer-case-insensitive	
action	action			
aclitype	type		empty or <reg-expr> with pre-defined parameters: \$ORIGINAL, \$LOCAL_IP, \$LOCAL_PORT, \$REMOTE_IP, \$REMOTE_PORT, \$REMOTE_VIA_HOST, \$TRUNK_GROUP, \$TRUNK_GROUP_CONTEXT add, replace, delete-header, delete-element, store, none, find-replace-all header-value, header-param-name, header-param, uri-display, uri-user, uri-user-param, uri-host, uri-port, uri-param-name, uri-param, uri-header-name, uri-header, statust-code, reason-phrase, mime	
HeaderRule	header-rule		empty or <regular-expr> with \$MANIP_STRING	6.1.0 and above
matchValue	match-value			
ElementRule	element-rule		header-value, header-param-name, header-param, uri-display, uri-user, uri-user-param, uri-host, uri-port, uri-param-name, uri-param, uri-header-name, uri-header, status-code, reason-phrase, mime, uri-user-only, uri-phone-number-only	
aclitype	type			
matchValue	match-value		empty or <regular-expr> with \$MANIP_STRING	
HeaderRule	header-rule	none	unique and ordered with mime-rule, mime-isup-rule	6.2.0 and above
acliObjectName *	name	any		
#	action	none	add, delete, manipulate, store, none, sip-manip, find-replace-all, reject, log	
action	msg-type		any, request, reply, out-of-dialog	
msgType	match-value		empty or <regular-expr> with \$MANIP_STRING, \$MANIP_PATTERN	
matchValue	new-value			
newValue	element-rule		empty or <regular-expr> or <sip-manipulation> or ACME_NAT_TO_FROM_IP	
ElementRule	action			
action	match-value			
matchValue	new-value		add, replace, delete-header, delete-element, store, none, find-replace-all, reject, log, sip-manip	
newValue				

			<p>empty or <regular-expr> with \$MANIP_STRING (\$M_STRING), \$MANIP_PATTERN</p> <p>empty or <sip-manipulation> or <reg-expr> with reserved words and operators: \$ORIGINAL, \$LOCAL_IP/PORT, \$REMOTE_IP/PORT, \$REMOTE_VIA_HOST, \$STRUNK_GROUP (\$T_GROUP), \$STRUNK_GROUP_CONTEXT (\$T_CONTEXT), \$REPLY_IP/PORT, \$TARGET_IP/PORT, \$TO/FROM/CONTACT/RURI/PAI/PPI/PCPID_USER/PHONE/HOST/PORT, \$TIMESTAMP_UTC, \$CALL_ID, &, , ==, ~=, !=, <=, >=, <, ></p>	
<p>MimeRules</p> <p>acliObjectName * #</p> <p>contentType</p> <p>action</p> <p>cmpType</p> <p>msgType</p> <p>format</p> <p>methods</p> <p>matchValue</p> <p>newValue</p> <p>MimeHeaderRule</p> <p>acliObjectName * #</p> <p>mimeHeaderName#</p> <p>action</p> <p>cmpType</p> <p>matchValue</p> <p>newValue</p>	<p>mime-rule</p> <p>name</p> <p>content-type</p> <p>action</p> <p>comparison-type</p> <p>msg-type</p> <p>format</p> <p>methods</p> <p>match-value</p> <p>new-value</p> <p>mime-header-rule</p> <p>name</p> <p>mime-header-name</p> <p>action</p> <p>comparison-type</p> <p>match-value</p> <p>new-value</p>	<p>none</p> <p>case-sensitive</p> <p>any</p> <p>ascii-string</p> <p>none</p> <p>case-sensitive</p> <p>any, request, reply, out-of-dialog</p> <p>ascii-string, hex-ascii, binary-ascii</p> <p>empty or comma seperated strings</p> <p>255 chars</p> <p>255 chars such as Content-Disposition</p> <p>add, replace, store, none, sip-manip, find-replace-all, reject, log</p> <p>case-sensitive, case-insensitive, pattern-rule, boolean, refer-case-sensitive, refer-case-insensitive</p>	<p>unique and ordered with header-rule, mime-isup-rule</p> <p>255 chars such as application/SDP, @preamble, @epilogue</p> <p>add, delete, manipulate, store, none, sip-manip, find-replace-all, reject, log</p> <p>case-sensitive, case-insensitive, pattern-rule, boolean, refer-case-sensitive, refer-case-insensitive</p>	<p>6.2.0 and above</p>
<p>MimeISUPRules</p> <p>acliObjectName * #</p> <p>contentType</p> <p>isupSpec</p> <p>isupMsgTypes</p> <p>action</p>	<p>mime-isup-rule</p> <p>name</p> <p>content-type</p> <p>isup-spec</p> <p>isup-msg-types</p> <p>action</p>	<p>ansi-2000</p> <p>none</p> <p>case-sensitive</p> <p>any</p> <p>0</p> <p>hex-ascii</p>	<p>unique and ordered with header-rule, mime-isup-rule</p> <p>255 chars such as application/ISUP</p> <p>ansi-2000, itu-99, gr-317, etsi-356</p> <p>empty or comma seperated list of 1-255</p> <p>add, delete, manipulate, store, none, sip-manip, find-replace-all, reject, log</p>	<p>6.2.0 and above</p>

ACLI to ACP Mappings

cmpType	comparison-type	none	case-sensitive, case-insensitive, pattern-rule, boolean, refer-case-sensitive, refer-case-insensitive any, request, reply, out-of-dialog empty or comma seperated strings 255 chars 0-255 number-param, hex-ascii, binary-ascii, ascii-string, bcd add, replace, store, none, sip-manip, find-replace-all, reject, log case-sensitive, case-insensitive, pattern-rule, boolean, refer-case-sensitive, refer-case-insensitive	
msgType	msg-type	case-sensitive		
methods	methods			
matchValue	match-value			
newValue	new-value			
MimeHeaderRule	mime-header-rule			
MimeISUPParamRule	isup-param-rule			
acliObjectName	name			
*#	type			
parameterType	format			
parameterFormat	action			
action	comparison-type			
cmpType	match-value			
matchValue	new-value			
newValue				
splitHdrList	split-headers		comma seperated list of header names such as "Allowed,P-Asserted-Identity", "Diversion,Allow"	6.2.0M1
joinHdrList	join-headers			

Session Constraints

The following table lists SOAP attributes and sub-elements for session constraints.

SOAP Attributes/Sub-elements	ACLI Session-router->session-constraints	Default Values	Valid Values	SBC Version
acliObjectname * #	name	disabled	24 characters	4.0.1; 4.1.1 above
useConstraints	state	0	enabled/ disabled	
maxNumSessions	max-sessions	0	0-999999999	
maxInbSessions	max-inbound-sessions	0	0-999999999	
maxOutbSessions	max-outbound-sessions	0	0-999999999	
maxBurstRate	max-burst-rate	0	0-999999999	
maxInbBurstRate	max-inbound-burst-rate	0	0-999999999	
maxOutbBurstRate	max-outbound-burst-rate	0	0-999999999	
maxSustainedRate	max-sustain-rate	0	0-999999999	
maxInbSustainedRate	max-inbound-sustain-rate	0	0-999999999	
maxOutbSustainedRate	max-outbound-sustain-rate	5	0-999999999	
minSeizure	min-seizures	0	0-999999999	
minAnswerSeizureRatio	min-asr	0	1-999999999	
timeRoResume	time-to-resume	0	0-100	

noResponseTo	ttr-no-response	0	0-999999999	
inServicePeriod	in-service-period	0	0-999999999	
burstWindow	burst-rate-window	0	0-999999999	
sustainedWindow	sustain-rate-window		0-999999999	
SessionConstraintRateConstraints	method		INVITE, ACK, BYE, REGISTER, CANCEL, PRACK, OPTIONS, INFO, SUBSCRIBE, NOTIFY, REFER, UPDATE, MESSAGE, PUBLISH	5.1.1 and above
method	max-inbound-burst-rate		0-999999999	
maxInBurstRate	max-outbound-burst-rate		0-999999999	
maxOutBurstRate	sustain-rate-max-outbound		0-999999999	
maxInSustainedRate	max-outbound-sustain-rate		0-999999999	
maxOutSustainedRate			0-999999999	

Session Translation

The following table lists SOAP attributes and sub-elements for session translation.

SOAP Attributes/Sub-elements	ACLI	Default Values	Valid Values	SBC Version
	session-router->session-translation			
Id *#	id		list of translation rules	4.0.0 and above
RuleCalling	rules-calling			
acliObjectName *#	rules-called			
RuleCalled				
acliObjectName *#				

Translation Rules

The following table lists SOAP attributes and sub-elements for translation rules.

SOAP Attributes/Sub-elements	ACLI	Default Values	Valid Values	SBC Version
	session-router->translation-rule			

ACLI to ACP Mappings

Id *#	id	none	add, delete, replace, none	4.0.0 and above
aclitype	type	0	<string>	
add_s	add-string	0	0-999999999, \$ for appending at the end	
add_indx	add-index		@ as wild char or <string>	
del_s	delete-string		0-999999999	
del_indx	delete-index			

RPH Profile

The following table lists the SOAP attributes and sub-elements for RPH profiles.

SOAP Attributes/ Sub-elements	ACLI	Default Values	Valid Values	SBC Version
	session-router->rph-profile			
acliObjectName *#	name	accept	24 characters	4.1.4; 5.1.0 and above
callTreatment	call-treatment		accept, reject, priority	
mediaPolicy	media-policy		empty or <QoS marking profile>	
RValues	r-values		list or r-values such as ets.0 or wps.1, ets.1	
rValue *#				

RPH Policy

The following table lists the SOAP attributes and sub-elements for RPH policies.

SOAP Attributes/ Sub-elements	ACLI	Default Values	Valid Values	SBC Version
	session-router->rph-policy			
acliObjectName *#	name		24 characters	4.1.4; 5.1.0 and above
OverrideRValues	override-r-values		One rValue	
rValue *#	insert-r-values			
InsertRValues				
rValue *#				

Host Routes

The following table lists the SOAP attributes and sub-elements for host routes.

SOAP Attributes/ Sub-elements	ACLI	Default Values	Valid Values	SBC Version
	system->host-route			
netAddress *#	dest-network		ipv4	4.0.0 and above
netmask	netmask			

gateway	gateway		Ipv4	
description	description			5.1.1 and above
netAddress *#	dest-network		Ipv4, ipv6, ipv6/prefix	CX6.2.0 and above
netmask	netmask		Ipv4, not allowed for ipv6	
gateway	gateway		Ipv4, ipv6	

SIP Local Map Entry

The following table lists the SOAP attributes and sub-elements for SIP local map entries.

SOAP Attributes/ Sub-elements	ACLI	Default Values	Valid Values	SBC Version
	session-router->local-response-map->entries			
localerror *# sipstatus cause sipreason causereason	local-error sip-status q850-cause sip-reason q850-reason	0	invalid-message, cpu-overload, media-released, media-not-allocated 100-699	4.0.0 and above
localerror *#	local-error		invalid-message, cpu-overload, media-released, media-not-allocated, enum-void-route	4.1.1 and above
localerror *#	local-error		invalid-message, cpu-overload, media-released, media-not-allocated, enum-void-route, monthly-minutes-exceed, next-hop-sa-oos, recv-sa-exc-constraints, revc-sip-int-exc-constraints, next-hop-sa-exc-constraints, next-hop-sip-int-exc-constraints, realm-bw-exc-poly-serv-reject, no-steering-pool-ports-available, allow-anonymous-rejection, sdp-address-mismatch,	4.1.4; 5.1.1 and above
localerror *# method registerResponseExpires	local-error method register-response-expires		invalid-message, cpu-overload, media-released, media-not-allocated, enum-void-route, monthly-minutes-exceed, next-hop-sa-oos, recv-sa-exc-constraints, revc-sip-int-exc-constraints, next-hop-sa-exc-constraints, next-hop-sip-int-exc-constraints, realm-bw-exc-poly-serv-reject, no-steering-pool-ports-available, allow-	5.1.1 and above

ACLI to ACP Mappings

			anonymous-rejection, sdp-address-mismatch, request-method-throttled empty, REGISTER 0-999999999	
--	--	--	---	--

Codec Policy

The following table lists the SOAP attributes and sub-elements for codec policies.

SOAP Attributes/ Sub-elements	ACLI	Default Values	Valid Values	SBC Version
	media manager->codec-policy			
acliObjectName *#	name		list of *, <media profile>, PCMU, G726-32, G723, PCMA, G722, G726, G729, telephone-event with appending exception :no or :force same values as in list above, but order matters	4.1.1
CodecPolicyAllow	allow-codecs			
acliObjectName *#	order-codecs			
CodecPolicyOrder				
acliObjectName *#				

Access Control

The following table lists the SOAP attributes and sub-elements for access control.

SOAP Attributes/Sub-elements	ACLI	Default Values	Valid Values	SBC Version
	session-router->access-control			
inRealm *	realm-id	0.0.0.0	<realm>	4.0.0
inSrc *#	source-address	0.0.0.0	lpv4/mask:port/mask	
InDst *#	destination-address	all	lpv4/mask:port/mask for application-protocol, NONE	
appProtocol *#	application-protocol	permit	NONE, SIP, MGCP	
transProtocol *#	transport-protocol	0	TCP, UDP, all	
access	access	none	permit, deny	
rateLimit	average-rate-limit	0	0-4294967295	
trustLevel	trust-level	0	none, low, medium, high	
errMsgThreshold	invalid-signal-threshold	30	0-4294967295	
maxMsgThreshold	maximum-signal-threshold	0 or the same as average-rate-limit	0-4294967295	
denyTimer	deny-period		0-4294967295	
maxMsgThresholdUntrusted	untrusted-signal-threshold		0-4294967295	

reservedBandwidth	minimum-reserved-bandwidth	0	0-999999999	4.1.4; 5.1.1 and above
description	description		255 characters	5.1.1 and above
natTrustThreshold	nat-trust-threshold	0	0-65535	6.1.0 and above
cacFailThreshold	cac-fail-threshold	0	0-999999999	6.2.0 and above
untrustedCacFailThreshold	untrusted-cac-failure-threshold	0	0-999999999	
inSrc *# inDst *#	source-address destination-address		ipAddress/mask:port/mask, ipAddress is either ipv4 or ipv6, mask is 32 for ipv4, 128 for ipv6	CX6.2.0 and above

Media Profile

The following table lists the SOAP attributes and sub-elements for media profile.

SOAP Attributes/ Sub-elements	ACLI session-router- >media-profile	Default Values	Valid Values	SBC Version
acliObjectName *#	name	audio	24 characters	4.0.0 and above
mediaType	media-type	0	audio, video, data, application, control	
payloadType	payload-type	RTP/AVP	RTP/AVP, UDP	
transport	transport	0	0-999999999	
reqBandwidth	req-bandwidth	0	0-256	
framesPerPacket parameters	frames-per-packet parameters		space separated <name=value> pair	
avgRate	average-rate-limit	0	0-125000000	
peakRate	peak-rate-limit	0	0-125000000	
maxBurstSize	max-burst-size	0	0-125000000	
mediaType	media-type	audio	audio, video, data, appkication, control, imate, text	4.1.1 and above
sdpRateLimit	sdp-rate-limit- headroom	0	0-100	
sdpBandwidth	sdp-bandwidth	disabled	enabled or disabled	
policeRate	police-rate	0	0-999999999	5.1.1 and above
subName	subname			6.1.0 and above

SIP Response Map

The following table lists the SOAP attributes and sub-elements for SIP response map.

SOAP Attributes/Sub-elements	ACLI	Default Values	Valid Values	SBC Version
	session-router->sip-response-map			
acliObjectName *#	name		100-699	4.0.0 and above
SIPResponseMapEntry	entries		100-699	
statusRcvd *#	recv-code			
statusSend #	xmit-code			
reason	reason			
method	method	0	empty, REGISTER	5.1.1 and above
registerResponseExpires	register-response-expires		0-999999999	

Diameter Director Agent

The following table lists the SOAP attributes and sub-elements for the DIAMETER Director agent.

SOAP Attributes/Sub-elements	ACLI	Default Values	Valid Values	SBC Version
	session-router->diameter-director-agent			
port	port	3868	Valid port number	DD1.0.0 and above
constraintName	constraint-name	30	0-65535	
appProtocol	protocol	enabled	enabled or disabled	
watchDogTimer	watchdog-timer	TCP	TCP or SCTP	
state	state	outbound	FQDN or IP address	
transportProtocol	transport-protocol	0	diameter-manipulation rule	
responseMap	response-map	authentication	Existing realm name	
hostname	hostname		diameter-manipulation rule	
inManipulationId	in-manip-ip		outbound, inbound, inbound-dynamic-ip	
realmId	realm-id		256-character string	
outManipulationId	out-manip-id		IP address	
connectionMode	connection-mode		32-bit hexadecimal or 32-bit integer	
description	description		32-bit integer	
options	options		32-bit integer	
ipAddress	ipAddress		authentication or accounting	
diamDirApplication	diameter-director-applications			
appId	application-id			

vendorId	vendor-id			
appType	application-type			
tosValue	tos-value			DD1.0.0M1 and DD2.0.0M1

Diameter Director Configuration

The following table lists the SOAP attributes and sub-elements for the DIAMETER Director agent.

SOAP Attributes/Sub-elements	ACLI	Default Values	Valid Values	SBC Version
	session-router->diameter-director-config			
redundancyPort	redndancy-port	1999	Valid port number	DD1.0.0 and above
llAction	load-limit-action	reject	reject or drop	
dynamicRouting	dynamic-routing	enabled	enabled or disabled	
state	state	enabled	enabled or disabled	
loadLimit	load-limit	85	0-100	
activeRedPort	active-redundancy-port	9000	Valid port number	
redNumTrans	red-max-transactions	50000	0-999999	
llExpResultCode	load-limit-exp-result-code	3004	Valid result code	
statefulPolicy	stateful-policy			
llResultCode	load-limit-result-code			
options	options			DD2.0.0 and above

Diameter Director Constraints

The following table lists the SOAP attributes and sub-elements for the DIAMETER Director constraints.

SOAP Attributes/Sub-elements	ACLI	Default Values	Valid Values	SBC Version
	session-router->diameter-director-constraints			
burstWindow	burst-rate-window	0	0-999999	DD1.0.0 and above
maxInbSustainedRAte	max-inbound-sustain-rate	0	0-999999	
maxOutbBurstRate	max-outbound-burst-rate	0	0-999999	
sustainedRate	sustain-rate-window	0	0-999999	
useConstraints	state	enabled	enabled or disabled	
maxBurstRate	max-burst-rate	0	0-999999	
maxInbBurstRate	max-inbound-burst-rate	0	0-999999	
maxOutbBurstRate	max-outbound-burst-rate	0	0-999999	

ACLI to ACP Mappings

timeToResume	time-to-resume	0	0-999999
name	name	0	0-999999
maxSustainedRate	max-sustained-rate	3004	1000-6000
lastModifiedBy	last-modified-by	0	0-999999
resultCode	result-code	0	0-999999
lastModifiedDate	last-modified-date	0	0-999999
messageRateConstraints	message-rate-constraints	0	0-999999
maxOutSustainedRate	max-outbound-sustain-rate		
maxInSustainedRate	max-inbound-sustain-rate		
maxInBurstRate	max-inbound-burst-rate		
command	command		
maxOutBurstRate	max-outbound-burst-rate		

Diameter Director Group

The following table lists the SOAP attributes and sub-elements for the DIAMETER Director group.

SOAP Attributes/Sub-elements	ACLI	Default Values	Valid Values	SBC Version
	session-router->diameter-director-constraints			
groupName	group-name	enabled	256 character string	DD1.0.0 and above
description	description	hunt	enabled or disabled	
lastModifiedBy	last-modified-by	100000	hunt	
state	state	disabled	diameter-manipulation rule	
lastModifiedDate	last-modified-date	32000	diameter-manipulation rule	
strategy	strategy	0	1-999999	
inManipulationId	in-manip-id	authentication	enabled or disabled	
outManipulationId	out-manip-id		1-999999	
recursiveRouting	recursive-routing		valid diameter result code	
recursionTimeout	recursion-timeout		32-bit hexadecimal or 32-bit integer	
doRecursion	do-recursion		32-bit integer	
transactionTimeout	transaction-timeout		authentication or accounting	
resultCodes	result-codes			
expResultCodes	exp-result-codes			
diamDirApplication	diameter-director-application			
appId	application-id			
vendorId	vendor-id			
appType	application-type			

destination	destinations			
seqno	seqno			
name	name			

Diameter Director Interface

The following table lists the SOAP attributes and sub-elements for the DIAMETER Director interface.

SOAP Attributes/Sub-elements	ACLI session-router->diameter-director-constraints	Default Values	Valid Values	SBC Version
constraintName	constraint-name	enabled	diameter director constraint	DD1.0.0 and above
outManipId	out-manip-id	none	diameter-manipulation rule	
state	state	3868	disabled	
routingPolicy	routing-policy	TCP	256 character string	
realmId	realm-id	all	256 character string	
suppVendorIds	supported-vendor-ids	0	256 character string	
originHostId	origin-host-identifier	authentication	none, identifier, identifier-with-realm	
description	description		diameter-manipulation rule	
originHostFormat	origin-host-format		number greater than 1023	
lastModifiedBy	last-modified-by		TCP, SCTP	
lastModifiedDate	last-modified-date		IP address	
inManipId	in-manip-id		all, agents-only	
sipPort	diameter-director-ports		IP address	
port	port		32-bit hexadecimal or integer	
tlsProfile	tls-profile		32-bit integer	
transProtocol	transport-protocol		authentication, accounting	
address	address			
anonMode	allow-anonymous			
imsAkaProfile	ims-aka-profile			
sctpMultiHomeAddrs	multi-home-addrs			
diamDirApplication	diameter-director-application			
appId	application-id			
vendorId	vendor-id			
appType	application-type			
tosValue	tos-value			DD1.0.0M1 and DD2.0.0M1

Diameter Manipulation

The following table lists the SOAP attributes and sub-elements for the DIAMETER manipulation.

SOAP Attributes/Sub-elements	ACLI session-router->diameter-director-constraints	Default Values	Valid Values	SBC Version
description	description	case-sensitive	256 character string	DD1.0.0 and above
name	name	none	256 character string	
lastModifiedBy	last-modified-by	0	AVP header-rule	
lastModifiedDate	last-modified-date	any	256 character string	
diameterManipRules	diameter-manip-rules	0	case-sensitive, case-insensitive, pattern-rule, boolean	
seqno	seqno	none	none, add, delete, store, diameter-manip, group-manip, find-replace-all, replace	
newValue	new-value	none	AVP code	
avpCodeDescr	descr-avp-code	none	any, request, reply	
name	name		diameter message code	
cmpType	comparison-type		none, octet-string, octet-hex, integer32, unsignedint32, address, diameteruri, enumerated	
action	action		avp-flag or avp-vendor-id	
matchValue	match-value		none, add, delete, replace	
avpCode	avp-code			
msgType	msg-type			
msgCmdCode	msg-cmd-code			
avpType	avp-type			
avpHeaderRule	avp-header-rule			
headerType	header-type			
newValue	new-value			
name	name			
action	action			
matchValue	match-value			

Running a Legacy SOAP Client API

Users who have created client applications with now deprecated APIs can run these applications after completing the following procedure.

1. Extract NNC700WSlassic.zip (contained on the Oracle software distribution CD) to a folder on the client computer. This folder provides the [WSClassicClient_HOME}.
2. Move existing applications, created with the now deprecated provisioning APIs, to the {WSClassicClient_HOME}/sampleSouce folder.
3. Delete the directory that previously contained client applications.
4. Go to {WSClassicClient_HOME}/bin.
5. Edit run.bat, the file that allows you to run client application code, by changing the JAVA_HOME path variable to match the JDK installation path. In addition, edit the SERVER_NAME and SERVER_PORT variables to match the IP address and port number of the Oracle Communications Session Delivery Manager Server.



Note: Client application code now requires JDK 1.6.0 or later; the latest update is recommended.

6. Edit build.bat by making the same change to the JAVA_HOME path variable.
7. The following Step, which imports one or more server certificates to a specific JAVA keystore, is required only if the client interface will run over HTTPS. This Step can be safely ignored if client/server transactions will take place over unsecured HTTP.

1. Use FTP to move a copy of a Oracle Communications Session Delivery Manager Server public certificate to the JAVA_HOME location on the client computer.
2. The certificate is usually at opt/AcmePacket/NNC700/ssl/nncentral_server.cer on the Net-Net Central Server.
3. Use the JAVA keytool utility to import the public certificate into a specified JAVA keystore. For example,

```
keytool -import -keystore trustedCerts -alias NNC-01 -file nnC01.cer
```

imports the certificate file, nnC01.cert, into the keystore named trustedCerts; the keystore file will be referenced by the NNC-01 alias.

Note that you will be prompted for the keystore password before the import operation is initiated.

For example:

```
Owner: EMAILADDRESS=test@test.com, CN=172.30.10.120, OU=NmsCore,
O=Acme Packet Inc., ST=Some-State, C=AU
Issuer: EMAILADDRESS=test@test.com, CN=172.30.10.120, OU=NmsCore,
O=Acme Packet Inc., ST=Some-State, C=AU
Serial number: 8b4d53819b6dfff1
Valid from: Tue Nov 14 16:04:53 EST 2006 until: Sat Jan 31 16:04:53 EST
```

```
2015
Certificate fingerprints:
MD5: 98:DA:F6:04:A8:A0:CA:D4:33:83:2A:3F:CE:C3:FB:CD
SHA1: F4:BB:72:7D:43:25:56:86:6A:70:55:27:63:96:D2:13:DF:89:B2:68
Trust this certificate? [no]: y
Certificate was added to keystore
```

4. Edit run.bat by changing the TRUST_STORE variable to match the location of the JAVA keystore that contains the public certificates of associated Oracle Communications Session Delivery Manager Servers.
5. Edit run.bat by changing the TRUST_STORE_PASSWORD to match the password required to access the JAVA keystore containing the Oracle Communications Session Delivery Manager Server certificates.
6. Use the JAVA keytool utility to conform the presence of the key in the keystore. For example,

```
keytool -list -v -keystore trustedCerts
```

provide a verbose display of the contents of the designated JAVA keystore, in this case, trustedCerts.

Note that you will be prompted for the keystore password before the keystore contents are displayed.

Repeat Steps 7a, 7b, and 7e to import additional Net-Net Central Server certificates to the same JAVA keystore.

8. If present, comment out the following code in your applications:

```
/* org.apache.axis.client.Stub yourStub = (Stub) emsLevelStub;// add this
line
yourStub._setProperty(org.apache.axis.MessageContext.HTTP_TRANSPORT_VERSION,
org.apache.axis.transport.http.HTTPConstants.HEADER_PROTOCOL_V11);// add
this line

//For the defect: SocketTimeoutException
org.apache.axis.client.Stub s = (Stub) networkLevelStub;
s.setTimeout(1800000);
//30 minutes, 30*60*1000

*/
```

9. Edit {WSCClassicClient_HOME}/conf/client.properties by changing the value of the session_timeout_ms property to specify a session timeout value, expressed in milliseconds.
10. Use build.bat to compile the client application.
11. Use run.bat to run the client application.