

**Oracle® Communications Session
Element Manager**

Web Services SOAP/XML Provisioning API Guide

Release 7.3

Formerly Net-Net Central

October 2013

Copyright ©2013, 2012 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

About This Guide	xi
Overview	xi
Deprecation	xi
Audience	xi
Related Documentation	xi
 1 About the Web Service Interface	 15
Introduction	15
Supported Configuration Elements	15
Terminology	15
Overview	15
Server-Side	16
Client-Side	16
Data Structures	17
DeviceInfoObject	17
IntegrityCheckResult	17
NNCDetails	17
NNCServerIPInfo	18
SaveDeviceTaskMessage	18
SBCDetails	19
WSBatch	19
WSBatchOperation	19
WSConfigAttribute	19
WSConfigAttributeMetaData	20
WSConfigAttributeMetaData, AttributeValueTypeInfo	20
WSConfigElement	21
WSConfigElementMetaData	21
WSConfigResult	21
WSDeviceResult	21
Exceptions/Faults	22
AcmeWSFault	22

AcmeAdminWSFault	22
AcmeConfigWSFault	22
AcmeDeviceWSFault	22
Build and Run the Sample Client Java Programs	22
Sample Work Flow	24
 2 Administrative Management Level	 25
Interface AdminMgmtIF	25
login	25
Input Parameters	25
Output Parameters	25
Throws	25
logout	25
Throws	25
Input Parameters	25
Output Parameters	25
Throws	25
getUserInfo	25
Input Parameters	26
Output Parameters	26
Throws	26
getAllUserInfo	26
Input Parameters	26
Output Parameters	26
Throws	26
getAccountManagementInfo	27
Input Parameters	27
Output Parameters	27
Throws	27
getLoginBanner	27
Input Parameters	27
Output Parameters	27
Throws	27
getTrapReceivers	27
Input Parameters	27
Output Parameters	27
Throws	27
 3 Device Management	 29
DeviceMgmtIF	29
addDevice	29

Input Parameters	29
Output Parameters	29
Throws	29
loadDevice	29
Input Parameters	29
Output Parameters	29
Throws	29
deleteDevice	30
Input Parameters	30
Output Parameters	30
Throws	30
saveConfig	30
Input Parameters	30
Output Parameters	30
Throws	30
activateConfig	31
Input Parameters	31
Output Parameters	31
Throws	31
saveAndActivateConfig	31
Input Parameters	31
Output Parameters	31
Throws	31
getAllManagedDevicesNames	32
Input Parameters	32
Output Parameters	32
Throws	32
getAllManagedDevicesbyDeviceGroup	32
Input Parameters	32
Output Parameters	32
Throws	32
getSBCDetails	33
Input Parameters	33
Output Parameters	33
Throws	33
getDevicePollingInterval	33
Input Parameters	33
Output Parameters	33
Throws	33
lockDevice	33
Input Parameters	33
Output Parameters	33
Throws	33

unlockDevice	34
Input Parameters	34
Output Parameters	34
Throws	34
getAllDeviceGroupList	34
Output Parameters	34
Throws	34
addDeviceGroup.	34
Input Parameters	34
Output Parameters	34
Throws	34
deleteDeviceGroup.	35
Input Parameters	35
Output Parameters	35
Throws	35
getLCVContentSaveSessionReport	35
Specified by	35
Input Parameters	35
Output Parameters	35
Throws	35
getAllManagedDevices	35
Specified by	36
Input Parameters	36
Output Parameters	36
Throws	36
getAllManagedDeviceTargetNames	37
Specified by	37
Input Parameters	37
Output Parameters	37
Throws	37
getNNCDetails	37
Specified by	37
Input Parameters	37
Output Parameters	37
Throws	37
getTopLevelElementCount	38
Specified By	38
Input Parameters	38
Output Parameters	38
Throws	38
getAllAssociatedDevicesInEMSLicense	38
Input Parameters	38
Output Parameters	38

Throws	38
addDeviceToEMSLicense	38
Input Parameters	38
Output Parameters	38
Throws	38
removeDeviceFromEMSLicense	39
Input Parameters	39
Output Parameters	39
Throws	39
4 Configuration Management Level	41
ConfigMgmtIF	41
getPrimaryKeyByElementType	41
Input Parameters	41
Output Parameters	41
Throws	41
getTopLevelConfigElementTypeNames	41
Input Parameters	41
Output Parameters	41
Throws	41
getSubElementTypesByElementType	42
Input Parameters	42
Output Parameters	42
Throws	42
getRequiredSubElementTypesByElementType	42
Input Parameters	42
Output Parameters	42
Throws	42
getAllSupportedAttributeInfoByElementType	43
Input Parameters	43
Output Parameters	43
Throws	43
deleteConfigElement	43
Input Parameters	43
Output Parameters	43
Throws	43
updateConfigElement	44
Input Parameters	44
Output Parameters	44
Throws	44
getConfigElement	45
Input Parameters	45

Output Parameters	45
Throws	45
getAllConfigElements	45
Input Parameters	45
Output Parameters	45
Throws	45
applyBatch	46
Input Parameters	46
Output Parameters	46
Throws	46
addConfigElement	46
Input Parameters	46
Output Parameters	46
Throws	46
replace	47
Input Parameters	47
Output Parameters	47
Throws	47
addSubElement	47
Input Parameters	47
Output Parameters	47
Throws	47
deleteSubElement	48
Input Parameters	48
Output Parameters	48
Throws	48
getConfigElementMetaData	48
Input Parameters	48
Output Parameters	48
Throws	48
getConfigAttributeMetaData	49
Input Parameters	49
Output Parameters	49
Throws	49
getValuesForReferenceAttribute	49
Input Parameters	49
Output Parameters	49
Throws	49
newConfigElement	50
Input Parameters	50
Output Parameters	50
Throws	50
encryptedPassword	50

Input Parameters	50
Output Parameters	50
Throws	50

Appendix AACP to ACLI Mappings 51

Physical Interface	57
Network Interface	58
Realm	59
Realm Media Address	61
Surrogate Agent	61
SIP Interface	62
SIP NAT	64
H.323 Stack	65
MGCP Config	66
DNS Config	67
Session Agent	68
Session Agent Group	70
Local Policy	71
Network Management Control	73
SIP Header Manipulation	74
Session Constraints	78
Session Translation	78
Translation Rules	78
RPH Profile	79
RPH Policy	79
Host Routes	79
SIP Local Map Entry	80
Codec Policy	80
Access Control	81
Media Profile	81
SIP Response Map	82
Diameter Director Agent	82
Diameter Director Configuration	82
Diameter Director Constraints	83
Diameter Director Group	83
Diameter Director Interface	84
Diameter Manipulation	85

Appendix BBackwards Compatibility 87

Running a Legacy SOAP Client API	87
---	-----------

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 Documentation

The following table lists related documents.

Document Name	Document Description
Installation Guide	Contains graphical and next mode installation information.
High Availability Guide	Describes Net-Net Central High Availability (HA) and the HA cluster, which is a network of tightly-linked servers. HA provides continuous management of the Net-Net Central 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 Net-Net Central graphical user interface (GUI), detailed information about managing devices in Net-Net Central, and Net-Net Central's licenses.

Document Name	Document Description
Element Manager Guide	Contains detailed information pertaining to Net-Net Central's Element Manager application and describes the dashboard summary view, audit log, fault, and performance views.
Route Manager Guide	Contains detailed information about centrally automating the management and distribution of routing data.
Quick Start Guide (C-series, D-series, and E-series)	Contains a brief description of the GUI, along with information on how to add a device and perform basic configuration tasks.
Administration Guide (C-series, D-series, and E-series)	Contains information about security administration, which lets you create new users and new user groups, and set group-based authorization.
Configuration Guide Default View (C-series, D-series, and E-series)	Contains information about the administration and software configuration of the Net-Net SBCs using Net-Net Central's default configuration view.
SOAP XML Provisioning API User Guide	Contains information about how to use the Net-Net Central SOAP API.

Introduction

Net-Net Central Web Service is a SOAP/XML Provisioning Application Programming Interface (API) enabling users to write applications that automate the provisioning of Acme Packet 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 Net-Net Central.
- *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 Acme Packet Session Border Controller (SBC)

Overview

The Web Service Provisioning API is part of a client/server programming model. The Net-Net Central 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 Net-Net Central database or apply configurations to Net-Net SBCs in the same way in which it responds to requests from the Net-Net Central GUI client application.

The Provisioning API itself is defined by Acme Packet, 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 Net-Net Central 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 Net-Net Central 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 Net-Net Central Available SOAP services screen at *http://<ipAddress>:8080/ACMEWS/services*, where <ipAddress> is the IP address of a Net-Net Central 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 a 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.

- 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 aclName 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 ACME Packet 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 Net-Net Central 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.

- 5a. Use FTP to move a copy of a Net-Net Central 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 Net-Net Central Server.

- 5b. 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
```

- 5c. Edit *run.bat* by changing the TRUST_STORE variable to match the location of the JAVA keystore that contains the public certificates of associated Net-Net Central Servers.

- 5d. Edit *run.bat* by changing the TRUST_STORE_PASSWORD to match the password required to access the JAVA keystore containing the Net-Net Central Server certificates.

- 5e. 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.

- 5f. Repeat Steps 5a, 5b, and 5e to import additional Net-Net Central 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 Net-Net Central 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

2 Administrative Management Level

The Administration Management Level Interface (AdminMgmtIF) enables SOAP service between a SOAP Client (for example, a remote PC) and a SOAP Server (in this case, a Net-Net Central Server).

Interface AdminMgmtIF

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 passwordExpirationDate
- 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`

The Device Management Level Interface (ConfigDeviceIF) enables SOAP support of device management services.

DeviceMgmtIF

addDevice

addDevice adds device details to the configuration database.

```
public com.acmepacket.ems.ws.service.userobjects.WSDeviceResult
addDevice(com.acmepacket.ems.ws.service.userobjects.DeviceInfoObj
ect 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.SBC
Details>
getAllManagedDevicesByDeviceGroup(java.lang.String devicetGroupPa
th) throws com.acmepacket.ems.ws.service.fault.AcmeDeviceWSFault,
com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault
```

Input Parameters

- devicetGroupPath -- 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.WSC
onfigElement>
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	<code>com.acmepacket.ems.ws.service.fault.AcmeConfigWSFault</code> <code>com.acmepacket.ems.ws.service.fault.AcmeAdminWSFault</code>

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 WSConfigResult 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

4

Configuration Management Level

The Configuration Management Level Interface (ConfigMgmtIF) enables SOAP support of configuration/management services.

ConfigMgmtIF

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

getSubElementTypesByElementType 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 wsConfi
gElement)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 Net-Net Central 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

`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 wsConfi
gElement) 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 wsConfi
gElement) 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 wsConfi
gElement) 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 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 deleted

Output Parameters

WSConfigResult

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 WSConfigElementMetaData 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

WSConfigElementMetaData

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 WsConfigAttributeMetaData
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

WsConfigElementMetaData

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 WsConfigElement 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
```

Appendix A

ACP to ACLI Mappings

The first three tables in this appendix provide an SBC model-specific mapping of Acme Control Protocol (ACP) top-level element names to equivalent Acme Command Line Interface (ACLI) commands. ACP names are used to identify the target configuration element with the WSConfigElement data structure.

Table 1: Net-Net SBC 4250

ACP Element Name	ACLI Command Sequence
accessControl	session-router->access-control
acctConfig	session-router->account-config
auditLogging	
authConfig	security->authentication
authParamsConfig	security->auth-params
callRecordingServer	session-router->call-recording-server
captureReceiver	system->capture-receiver
certRecord	security->certificate-record
certStatusProfile	security->cert-status-profile
classPolicy	session-router->class-profile
codecPolicy	media-manager->codec-policy
dataFlow	security->ike->data-flow
dnsConfig	media-manager->dns-config
dpdParams	security->ike->dpd-params
earlyMediaGroup	
enforcementProfile	session-router->enforcement-profile
EnumConfig	session-router->enum-config
extBwManager	session-router->enforcement-profile
h248Config	session-router->h248-config
h248MgcConfig	session-router->h248->h248-mgc-config
h248MgConfig	session-router->h248->h248-mg-config
H323Config	session-router->h323
H323StackConfig	session-router->h323->h323-stacks
ikeCertificateProfile	security->ike->ike-certificate-profile
ikeConfig	security->ike->ike-config
ikeInterface	security->ike->ike-interface
ikeSaInfo	security->ike->ike-sainfo
imsAkaProfile	security->ims-aka-profile
ipRoute	system->host-route
ipsecGlobalConfig	security->ipsec->ipsec-global-config
IwfStackConfig	session-router->iwf-config
LiConfig	session-router->lawful-intercept
localAddressPool	security->ike->local-address-pool
localPolicy	session-router->local-policy
LocalRoutingConfig	session-router->local-routing-config
mediaPolicy	media-manager->media-policy
mediaProfile	session-router->media-profile
mediaRouter	media-manager->media-manager

Table 1: Net-Net SBC 4250

ACP Element Name	ACLI Command Sequence
mgcpConfig	session-router->mgcp-config
netMgmtCtrl	session-router->net-management-control
networkInterface	system->network-interface
NetworkParameters	system->network-parameters
ntpConfig	ntp-sync
passwordPolicy	security->password-policy
phyInterfaceConfig	system->phy-interface
qosConstraints	session-router->qos-constraints
q850SipMap	session-router->q850-sip-map
realmConfig	media-manager->realm-config
realmGroup	media-manager->realm-group
RedundancyConfig	system->redundancy
responseMap	session-router->sip-response-map
rphPolicy	session-router->rph-policy
rphProfile	session-router->rph-profile
saConfig	security->ipsec->security-association ->manual
sessionAgent	session-router->session-agent
sessionAgentGroup	session-router->session-group
sessionConstraints	session-router->session-constraints
sessionRouter	session-router->session-router-config
sessionTranslation	session-router->session-translation
sipConfig	session-router->sip-config
sipFeature	session-router->sip-feature
sipInterface	session-router->sip-interface
sipIsupProfile	session-router->sip-isup-profile
sipLocalMap	session-router->local-response-map
sipManipulation	session-router->sip-manipulation
sipNatConfig	session-router->sip-nat
sipProfile	session-router->sip-profile
sipQ850Map	session-router->sip-q850-map
snmpCommunity	system->snmp-community
spdConfig	security->ipsec->security-policy
sshPubKeyRecord	security->public-key
staticFlow	media-manager->static-flow
steeringPool	media-manager->steering-pool
surrogateAgent	session-router->surrogate-agent
sysACL	system->system-access-list
systemConfig	system->system-config
tlsGlobal	security->tls-global
tlsProfile	security->tls-profile
translationRules	session-router->translation-rules
trapReceiver	system->trap-receiver
tunnelOrigParams	security->ike->tunnel-orig-params

Table 2: Net-Net SBC 4500

ACP Element Name	ACLI Command Sequence
accessControl	session-router->access-control
acctConfig	session-router->account-config
auditLogging	
authConfig	security->authentication
authParamsConfig	security->auth-params
BgfConfig	media-manager->bgf-config
callRecordingServer	session-router->call-recording-server
captureReceiver	system->capture-receiver
certRecord	security->certificate-record
certStatusProfile	security->cert-status-profile
classPolicy	session-router->class-profile
codecPolicy	media-manager->codec-policy
dataFlow	security->ike->data-flow
diamDirConstraints	session-router->diameter-director-constraints
diamDirAgent	session-router->diameter-director-agent
dnsConfig	media-manager->dns-config
dpdParams	security->ike->dpd-params
earlyMediaGroup	
enforcementProfile	session-router->enforcement-profile
EnumConfig	session-router->enum-config
extBwManager	media-manager->ext-policy-server
h248Config	session-router->h248-config
h248MgcConfig	session-router->h248->h248-mgc-config
h248MgConfig	session-router->h248->h248-mg-config
H323Config	session-router->h323
H323StackConfig	session-router->h323->h323-stacks
ikeCertificateProfile	security->ike->ike-certificate-profile
ikeConfig	security->ike->ike-config
ikeInterface	security->ike->ike-interface
ikeSaInfo	security->ike->ike-sainfo
imsAkaProfile	security->ims-aka-profile
ipRoute	system->host-route
ipsecGlobalConfig	security->ipsec->ipsec-global-config
IwfStackConfig	session-router->iwf-config
LiConfig	session-router->lawful-intercept
localAddressPool	security->ike->local-address-pool
localPolicy	session-router->local-policy
LocalRoutingConfig	session-router->local-routing-config
mediaPolicy	media-manager->media-policy
mediaProfile	session-router->media-profile
mediaRouter	media-manager->media-manager
mediaSecPolicy	security->media-security->media-sec-policy
mgcpConfig	session-router->mgcp-config
mikeyProfile	security->media-security->mikey-profile
netMgmtCtrl	session-router->net-management-control
networkInterface	system->network-interface
NetworkParameters	system->network-parameters
ntpConfig	ntp-sync
passwordPolicy	security->password-policy
phyInterfaceConfig	system->phy-interface
q850SipMap	session-router->q850-sip-map

Table 2: Net-Net SBC 4500

ACP Element Name	ACLI Command Sequence
qosConstraints	session-router->qos-constraints
realmConfig	media-manager->realm-config
realmGroup	media-manager->realm-group
RedundancyConfig	system->redundancy
responseMap	session-router->sip-response-map
rphPolicy	session-router->rph-policy
rphProfile	session-router->rph-profile
saConfig	security->ipsec->security-association ->manual
sdesProfile	security->media-security->sdes-profile
sessionAgent	session-router->session-agent
sessionAgentGroup	session-router->session-group
sessionConstraints	session-router->session-constraints
sessionRouter	session-router->session-router-config
sessionTranslation	session-router->session-translation
sipConfig	session-router->sip-config
sipFeature	session-router->sip-feature
sipInterface	session-router->sip-interface
sipIsupProfile	session-router->sip-isup-profile
sipLocalMap	session-router->local-response-map
sipManipulation	session-router->sip-manipulation
sipNatConfig	session-router->sip-nat
sipProfile	session-router->sip-profile
sipQ850Map	session-router->sip-q850-map
snmpCommunity	system->snmp-community
spdConfig	security->ipsec->security-policy
sshPubKeyRecord	security->public-key
staticFlow	media-manager->static-flow
steeringPool	media-manager->steering-pool
surrogateAgent	session-router->surrogate-agent
sysACL	system->system-access-list
systemConfig	system->system-config
tlsGlobal	security->tls-global
tlsProfile	security->tls-profile
translationRules	session-router->translation-rules
trapReceiver	system->trap-receiver
tunnelOrigParams	security->ike->tunnel-orig-params
VbgConfig	media-manager->vbg-config

Table 3: Net-Net SBC 9200

ACP Element Name	ACLI Command Sequence
accessControl	session-router->access-control
acctConfig	session-router->account-config
authConfig	security->authentication
captureReceiver	system->capture-receiver
certRecord	security->certificate-record
classPolicy	session-router->class-profile
collectConfig	system->collect
dnsConfig	media-manager->dns-config
enforcementProfile	session-router->enforcement-profile
EnumConfig	session-router->enum-config
extBwManager	media-manager->ext-policy-server
h323Config	session-router->h323
h323StackConfig	session-router->h323->h323-stacks
ipRoute	system->host-route
IwfStackConfig	session-router->iwf-config
liConfig	session-router->lawful-intercept
localPolicy	session-router->local-policy
LocalRoutingConfig	session-router->local-routing-config
mediaPolicy	media-manager->media-policy
mediaProfile	session-router->media-profile
mediaRouter	media-manager->media-manager
netMgmtCtrl	session-router->net-management-control
networkInterface	system->network-interface
networkParameters	system->network-parameters
ntpConfig	system->ntp
passwordPolicy	security->password-policy
phyInterfaceConfig	system->phy-interface
q850SipMap	session-router->q850-sip-map
realmConfig	media-manager->realm-config
realmGroup	media-manager->realm-group
responseMap	session-router->sip-response-map
rphPolicy	session-router->rph-policy
rphProfile	session-router->rph-profile
saConfig	security->ipsec->>manual-security-association
sessionAgent	session-router->session-agent
sessionAgentGroup	session-router->session-group
sessionConstraints	session-router->session-constraints
sessionRouter	session-router->session-router-config
sessionTranslation	session-router->session-translation
sipConfig	session-router->sip-config
sipFeature	session-router->sip-feature
sipInterface	session-router->sip-interface
sipManipulation	session-router->sip-manipulation
sipNatConfig	session-router->sip-nat
sipProfile	session-router->sip-profile
sipQ850Map	session-router->sip-q850-map
snmpCommunity	system->snmp-community
soapConfig	system->soap-config
spdConfig	security->ipsec->security-policy
sshPubKeyRecord	security->public-key
staticFlow	media-manager->static-flow
steeringPool	media-manager->steering-pool
surrogateAgent	session-router->surrogate-agent

Table 3: Net-Net SBC 9200

ACP Element Name	ACLI Command Sequence
sysACL	system->system-access-list
systemConfig	system->system-config
timezone	system->timezone
tlsGlobal	security->tls-global
tlsProfile	security->tls-profile
transcodingConfig	media-manager->transcoding-policy
translationRules	session-router->translation-rules
trapReceiver	system->trap-receiver

The following tables list the most commonly used top-level elements along with associated attributes and sub-elements associated. Tables also list default attribute values, valid ranges ACLI equivalencies and version information.

Note: The key values noted in the following tables are valid for pre-7.0 Net-Net Central releases. Key values for releases 7.0 (and later) may be different from the values listed below. You can use the `getPrimaryKeyByElementType` API to obtain primary key values in Net-Net Central 7.0 and later releases.

Table conventions are as follows:

- Attributes are in normal font
- Sub-elements are in bold
- Attributes marked with an asterisk (*) are keys values for the configuration element, used to uniquely identify it for get, update, and delete calls in the API
- Attributes marked with a pound sign (#) are the mandatory attributes required when creating the managed object with the add call of the API.

Physical Interface

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

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

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

SOAP Attributes/Sub-elements	ACLI system->network-interface	Default Values	Valid Values	SBC Version
aclObjectName * subPortId ** hostname ipAddress utilityAddress secondUtilityAddress netmask gateway gatewaySec NetworkInterfaceGWHeartbeat state # timeout retrycount retryTimeout healthDec domNameServer domNameServerB1 domNameServerB2 defDomainName HipIpList ip ** protocolParameters ftpAddress icmpAddress snmpAddress telnetAddress dnsTimeout	name sub-port-id hostname ip-address pri-utility-addr sec-utility-addr netmask gateway sec-gateway gw-heartbeat state heartbeat retry-count retry-timeout health-score dns-ip-primary dns-ip-backup1 dns-ip-backup2 dns-domain hip-ip-list	<phy> disabled 0 0 1 0	<phy> 0-4095 0-255 chars ipv4 ipv4 ipv4 ipv4 ipv4 enabled/disabled 0-65535 0-65535 1-65535 0-100 ipv4 ipv4 ipv4 list of IPs ipv4 empty or combo of (ftp, icmp, snmp, telnet) ftp HIP icmp HIP snmp HIP telnet HIP 0-4294967295	4.0.0 and above
description	description		255 chars	5.0.0 and above
IcmpIpList ip **	icmp-address		HIP	4.1.4; 5.1.0 and above
hostname ipAddress utilityAddress secondUtilityAddress netmask gateway gatewaySec domNameServer domNameServerB1 domNameServerB2 HipIpList ip ** protocolParameters IcmpIpList ip **	hostname ip-address pri-utility-addr sec-utility-addr netmask gateway sec-gateway dns-ip-primary dns-ip-backup1 dns-ip-backup2 hip-ip-list icmp-address		0-255 chars, ipv4, ipv6 ipv4, ipv6, ipv6/prefix ipv4, ipv6 ipv4, ipv6 ipv4, not allowed for ipv6 ipv4, ipv6 ipv4, ipv6 ipv4, ipv6 ipv4, ipv6 list of IPs ipv4, ipv6 empty or combo of (ftp, icmp, snmp, telnet)	CX6.2.0 and above

Realm

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

SOAP Attributes/Sub-elements	ACLI media-manager->realm- config	Default Values	Valid Values	SBC Version
id *# addrPrefix parent RealmNetworkInterfaceId acliObjectName *# subPortId *# maxBandwidth mrlInRealm mrlInNetwork isMSMRelease qosEnable mediaPolicy intranslationid outtranslationid dnsRealm classProfile trustLevel rateLimit errMsgThreshold maxMsgThreshold denyTimer symmetricLatching paiStrip trunkContext bwManager inManipulationId outManipulationId options maxMsgThresholdUntrusted	identifier addr-prefix parent-realm network-interfaces max-bandwidth mm-in-realm mm-in-network msm-release qos-enable media-policy in-translationid out-translationid dns-realm class-profile access-control-trust-level average-rate-limit invalid-signal-threshold maximum-signal-threshold deny-period symmetric-latching pai-strip trunk-context ext-policy-svr in-manipulationid out-manipulationid options untrusted-signal-threshold	0.0.0.0 0 disabled enabled disabled disabled none 0 0 0 30 disabled disabled 0	24 characters 0.0.0.0 or ipv4 or ipv4/mask empty or <realm> <phy:port-id> or lo0:0 0-999999999 enabled/disabled enabled/disabled enabled/disabled enabled/disabled empty or <QoS marking profile> empty or <trans-profile> empty or <realm> empty or <cls-profile> none, low, medium, high 0-999999999 0-999999999 0-999999999 0-999999999 enabled/disabled enabled/disabled empty or <pol-server> empty or <sip-manipulation> list of options separated by comma 0-999999999	4.0.0 and above
mmInSystem restrictedLatching restrictionMask mmSameIp earlyMediaAllow RealmAdditionalPrefixes prefix *#	mm-in-system restricted-latching restriction-mask mm-same-ip early-media-allow additional-prefixes	enabled none 32 enabled	enabled/disabled none, sdp, peer-ip 1-32 enabled/disabled empty, none, reverse, or both list of ipv4/mask ipv4/mask	4.0.1 and above
acctEnable	accounting-enable	enabled	enabled, disabled	4.1.0 and above
netMgmtCtrl userCacMode userCacBandwidth userCacSessions delayedMediaUpdate nonMmBwCAC codecPolicy codePolicyInRealm	net-management-control user-cac-mode user-cac-bandwidth user-cac-sessions delay-media-update bw-cac-non-mm codec-policy codec-manio-in-realm	disabled none 0 0 disabled disabled disabled	enabled or disabled none, aor, ip 0-999999999 0-999999999 enabled/disabled enabled/disabled empty or <codec-policy> enabled/disabled	4.1.1 and above
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 maxPriorityBandwidth fallbackBandwidth icmpDetectMultip icmpAdvInterval icmpTargetIp MatchMediaProfile acliObjectName *# natTrustThreshold stunEnable stunServerIp stunServerPort stunChangedIp stunChangedPort qosConstraintName	manipulation-string max-priority-bandwidth fallback-bandwidth icmp-detect-multiplier icmp-advertisement-interval icmp-target-ip match-media-profiles nat-trust-threshold stun-enable stun-server-ip stun-server-port stun-changed-ip stun-changed-port qos-constraint	0 0 0 0 0 disabled 0.0.0.0 3478 0.0.0.0 3479	0-999999999 0-999999999 0-999999999 0-999999999 ipv4 list of <media-profile>, <media-profile>::, <media-profile>::<subname>, *, *::, *::<subname> 0-65535 enabled, disabled ipv4 1025-65535 ipv4 1025-65535 empty or <qos constraint>	6.1.0 and above
sipProfile sipSupProfile 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 media-manager->steering- pool	Default Values	Valid Values	SBC Version
ipAddress *# startPort *# endPort # realmID RealmNetworkInterfaceId acliObjectName subPortId	ip-address start-port end-port realm-id network-interface	<realm>	Ipv4 1025-65535 1025-65535, endPort > startPort <realm> empty or <phy:port-id>	4.0.0 and above
ipAddress *#	ip-address		ipv4, ipv6	CX6.2.0and above

Surrogate Agent

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

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

SOAP Attributes/Sub-elements	ACLI session-router->slp- interface	Default Values	Valid Values	SBC Version
state RealmID * trustMode	state realm-id trust-mode	enabled <realm> all	enabled/disabled <realm> all, agents-only, relam-prefix, registered	4.0.0 and above
SIPConfigurationSIPPort address *# port *# transProtocol *# anonMode proxyMode redirectAct contactSip telUri fqdnDomain natTraversal natInterval regMinExpire regInterval registrationCaching isRouteReg	slp-port address port transport-protocol allow-anonymous proxy-mode redirect-action contact-mode teluri-scheme uri-fqdn-domain nat-traversal nat-interval min-reg-expire registration-interval registration-caching route-to-registrar	5060 UDP all none disabled none 30 300 3600 disabled disabled	ipv4 1025-65535 UDP or TCP all, agents-only, realm-prefix, registered, register-prefix empty, proxy, redirect, record-route, stateless empty, proxy, recurse none, maddr, strict, loose enabled, disabled none, always, rport 0-999999999 1-999999999 0-999999999 enabled/disabled enabled/disabled list of carriers list of options seperated by comma	
SIPConfigurationCarriers acliObjectName *# options	carriers options			
natmaxInterval natIntervalIncrement natTestIncrement sipdynamicHnt stopRecurse	max-nat-interval nat-int-increment nat-test-increment sip-dynamic-hnt stop-recurse	3600 10 30 disabled 401,407	enabled, disabled list of response codes, 300-599 seperated by comma 0, 1025-65535 0, 1025-65535, end>start all, agents-only, relam-prefix, registered, none empty or <pol-server>	
portMapStart portMapEnd trustMode extPolicyServer defaultLocationString imsFeature NetworkID anonymousPriority termTgrpMode chargingVectorMode chargingFuncAddrMode	port-map-start port-map-end trust-mode ext-policy-server default-location-string sip-ims-feature network-id anonymous-priority term-tgrp-mode charging-vector-mode charging-function-address- mode	0 0 disabled None None pass pass	enable/disabled NONE (4.0.1), none (4.2), normal, non- urgent, urgent, emergency none, iptel, egress-uri none, pass, delete, insert none, pass, delete, insert	
ecfAddress ccfAddress operatorIdentifier inManipulationId outManipulationId	ecf-address ccf-address operator-identifier in-manipulationid out-manipulationid		Ipv4 or hostname Ipv4 or hostname empty or <sip-manipulation> strict, enabled, disabled	
implicitServiceRoute	implicit-service-route	disabled		4.0.1 and above

tcpNatInterval isSecure rfc2833-mode rfc2833-payload maxConnPerPeer maxConn idleConnTimeout SIPConfigurationSIPPort transProtocol tlsProfile	tcp-nat-interval secured-network rfc2833-mode rfc2833-payload per-src-ip-max-incoming- conns max-incoming-conns inactive-conn-timeout sip-port transport-protocol tls-profile	90 disabled transparent 101 0 0 0	0-99999999 enabled, disabled transparent, preferred 96-127 0-20000 0-20000 0-99999999 UDP, TCP, TLS <tls-profile> if TLS specified	4.1.0 and above
constraintName implicitServiceRoute	constraint-name implicit-service-route	disabled	empty or <session-constraint> strict, enabled, disabled	4.0.1; 4.1.1 and above
responseMap localresponseMap trans-expire invite-expire max-redirect-contacts rfc2833-mode	response-map local-response-map trans-expire invite-expire max-redirect-contacts rfc2833-mode		empty or <sip-response-map> 0-999999999 0-999999999 0-10 transparent, preferred, dual	4.1.1 and above
untrustedConnTimeout	untrusted-conn-timeout	0	0-999999999	5.0.0 and above
tcp-keepalive chargingFuncAddrMode	tcp-keepalive charging-function-address- mode	none pass	none, disabled, enabled none, pass, delete, insert, delete-and- respond, insert-reg-cache empty or <enforcement profile>	4.1.4; 5.1.0 and above
enforcementProfile add-sdp-invite SIPInterfaceMediaProfile	enforcementProfile add-sdp-invite add-sdp-profile	disabled	disabled, invite, reinvoke list of media profiles	4.1.4; 5.1.1 and above
referCallTransfer routeUnauthorizedCalls description	refer-call-transfer route-unauthorized-calls description	disabled	enabled, disabled empty, defined SA or SAG	5.1.1 and above
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 imsAkaProfile hmrString SIPConfigurationSIPPort transProtocol	ims-aka-feature ims-aka-profile manipulation-string slp-port transport-protocol	disabled UDP	enabled, disabled <ims-aka-profile> or empty UDP, TCP, TLS, SCTP	6.1.0 and above
sipProfile siplsupProfile	sip-profile sip-isup-profile		empty or <sip-profile> empty or <sip-isup-profile>	6.2.0 and above
manipPattern SIPConfigurationSIPPort address *#	manipulation-pattern slp-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 * ProxyAddress ProxyPort ExternalAddress HomeAddress HomeProxyAddress HomeProxyPort RouteHomeProxy prefix TunnelRedirect UserNATTag HostNATTag DomainSuffix ParamMode ParamName SipNatHeaders acliObjectName*#	realm-id ext-proxy-address ext-proxy-port ext-address home-address home-proxy-address home-proxy-port route-home-proxy address-prefix tunnel-redirect user-nat-tag host-nat-tag domain-suffix use-url-parameter parameter-name headers	<realm> 5060 0 disabled disabled -acme- ACME- none list of nat headers: Call-ID Contact f From I Join m rRecord-Route Refer-To Replaces Replay-To Route t To v Via	<realm> <0.0.0.0> or ipv4 1025-65535 Ipv4 Ipv4 Ipv4 1025-65535 enabled, disabled, forced empty, *, 0.0.0.0, ipv4/bitmask enabled/disabled <.com> none, from-to, all 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	4.0.0 and above

H.323 Stack

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

SOAP Attributes /Sub-elements	ACLI Session-router->h323- >h323-stack	Default Values	Valid Values	SBC Version
acliObjectName *# state realm-id * assoc-stack local-ip ras-port q931-port H323AlternateTransport ipAddress *# q931-max-calls max-calls max-channels q931-start-port q931-number-ports dynamic-start-port dynamic-number-ports tcp-keeplive isgateway AnonMode filename	name state realm-id assoc-stack local-ip ras-port q931-port alternate-transport q931-max-calls max-calls max-channels q931-start-port q931-number-ports dynamic-start-port dynamic-number-ports tcp-keepalive isgateway allow-anonymous filename	enabled <realm> 0.0.0.0 1719 1720 200 200 6 0 0 0 0 disabled enabled all	24 characters enabled/disabled <realm> empty or <h323-stack> empty or <HIP> 1025-65535 1025-65535 list of <ipv4:port> lpv4:port >0 >0, must > q931-max-calls >0 0-65535 0, 1024, 2048, 4096, 8192, 16384, 32768 0-65535 0, 1024, 2048, 4096, 8192, 16384, 32768 enabled/disabled enabled/disabled all, agents-only, realm-prefix	4.0.0 and above
H323TerminalAlias e164 *# url *# ipAddress *# email *# h323-ID *# H323Prefixes e164 *# url *# ipAddress *# email *# h323-ID *# registration-ttl processRegistration proxy-mode h245-stage	terminal-alias prefixes registration-ttl process-registration proxy-mode h245-stage	 connect	list of e164, url, h323-ID, email, ipAddress list of e164, url, h323-ID, email, ipAddress for gateway only NONE, H225, H245 CONNECT, SETUP, ALERTING, CALL PROCEEDING, DYNAMIC, FACILITY, SETUP or CONNECT, NONE enabled/disabled list of options seperated by comma enabled, disabled ipAddress=ipv4:port for gateway only ipAddress=ipv4:port	
h245-tunneling stack-options auto-gk-discovery multicast gatekeeper gk-identifier callStartFast callStartSlow H323MediaProfile acliObjectName *# fs-in-first-msg	h245-tunneling options auto-gk-discovery multicast gatekeeper gk-identifier call-start-fast call-start-slow media-profiles fs-in-first-msg	disabled disabled enabled disabled disabled	enabled/disabled enabled/disabled, each time only fast or only slow enabled 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 severity*# value	alarm-threshold severity value	minor 0	minor, major, critical 0-100	6.2.0 and above

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 * addressPrivate *# portPrivate mode divisor unitPrefix dnsAuthentication dnsTranslation natTraversal auditInterval options caRedundancy caPingMethod caPingInterval hostGWPublic addrGWPublic portGWPublic portGWPublic2realmPublic pubCAHost addrCAPublic portCAPublic	private-realm private-address private-port mode divisor unit-prefix dns-authentication dns-translation nat-traversal audit-interval options ca-redundancy ca-ping-method ca-ping-interval public-gw-host public-gw-address public-gw-port second-public-gw-portpublic-realm public-ca-host public-ca-address public-ca-port	<realm> 2727 LineUnit 256 disabled disabled 0 disabled 0 0.0.0.0 2427 0<realm> 0.0.0.0 2727	<realm> Ipv4 1025-65535 Host, LineUnit, LinePrefix, FQDN, FQDN2, OnlyHost 256, 65536, 16777216, 4294967295enabled/disabled empty or <trans-pfl> enabled/disabled list of options enabled/disabled "NTFY 1 ping@host" Ipv4/mask 1025-65535 0, 1025-65535<realm> Ipv4 1025-65535	4.0.0 and above
portALG	alg-port	2427	1025-65535	4.1.4; 5.1.1 and below
MGCPConfigIpAddresses Addr *#	ca-fallover-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 media-manager->dns-config	Default Values	Valid Values	SBC Version
clientrealmID *# description ClientIpList IPAddress *# ServerDNSAttributes serverRealmID *# ServerDnsDomainSuffix acliObjectName * # ServerDnsAddressList IPAddress * # sourceAddress *# sourcePort *# transactionTimeout ServerDNSAddressTranslation serverprefix *# clientprefix *#	client-realm description client-address-list server-dns-attributes serverrealmID domain-suffix server-address-list source-address source-port transaction-timeout address-translation server-prefix client-prefix	<realm> 53 10	<realm> ipv4 <realm> domain ipv4 ipv4 0-65535 0-999999999 ipv4/mask ipv4/mask	4.0.0 and above

Version 7.3

codecPolicy inServiceCodes outServiceCodes rfc2833-mode rfc2833-payload	codec-policy in-service-response-codes out-service-response-codes rfc2833-mod rfc2833-payload	None 0	empty or <codec-policy> list of comma-separated response codes, 200-699 none, transparent, preferred, dual 0, 96-127	4.1.1 and above (for SIP only)
reuse-connections tcp-keepalive tcp-reconn-Interval enforcementProfile	reuse-connections tcp-keepalive tcp-reconn-interval enforcement-profile	NONE none 0	NONE, TCP none, disabled, enabled 0, 2-300 empty or <enforcement-profile>	4.1.4; 5.1.0 and above (for SIP only)
maxRegisterBurstRate registerBurstWindow	max-register-burst-rate register-burst-window	0 0	0-999999999 0-999999999	4.1.4, 5.1.1 and above (for SIP only)
referCallTransfer pingSendMode	refer-call-transfer ping-send-mode	disabled keep- alive	enabled, disabled keep-alive, continuous empty or <realm>	5.1.1 and above (for SIP only)
egressRealmID SessionAgentRateConstraints method maxInBurstRate maxOutBurstRate maxInSustainedRate maxOutSustainedRate	egress-realm-id rate-constraints method max-inbound-burst-rate max-outbound-burst-rate max-inbound-sustain-rate max-outbound-sustain-rate		INVITE, ACK, BYE, REGISTER, CANCEL, PRACK, OPTIONS, INFO, SUBSCRIBE, NOTIFY, REFER, UPDATE, MESSAGE, PUBLISH 0-999999999 0-999999999 0-999999999 0-999999999	5.1.1 and above
hmrString transMethod	manipulation-string transport-method	UDP	UDP, UDP+TCP, dynamicTCP, staticTCP, dynamicTLS, staticTLS, staticSCTP, NONE, TCP, SCTP	6.1.0 and above (for SIP only)
reuse-connections pingAllAddresses sipProfile siplsupProfile	reuse-connections ping-all-addresses sip-profile sip-isup-profile	NONE disabled	enabled, disabled empty or <sip-profile> empty or <sip-isup-profile>	6.2.0 above (for SIP only)
manipPattern referCallTransfer	manipulation-pattern refer-call-transfer	disabled	disabled, enabled, dynamic	
hostname *# ipAddress	hostname ip-address		fqdn, ipv4, ipv6 ipv4, ipv6	CX6.2.0 and above

Session Agent Group

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

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

Local Policy

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

SOAP Attributes/Sub-elements	ACLI Session-router->local-policy	Default Values	Valid Values	SBC Version
routeName * LocalPolicyFrom # addr ** LocalPolicyTo # addr ** LocalPolicySourceRealm # acliObjectName ** activateTime deactivateTime state anonymousPriority LocalPolicyAttribute policyName * state startTime endTime dow cost LocalPolicyMediaProfiles acliObjectName ** carrierName nextHop # destRealm # appProtocol replace	from-address to-address source-realm activate-time deactivate-time state policy-priority policy-attribute state start-time end-time days-of-week cost media-profiles carrier next-hop realm app-protocol replace-uri	* * * enabled none enabled 0000 2400 U-S 0 SIP enabled	Route0, Route1..... +number, number, *, fqdn, ipv4 or 0.0.0.0 same * or <realm> yyyy-mm-dd-hh:mm:ss yyyy-mm-dd hh:mm:ss enabled/disabled none, normal, non-urgent, urgent, emergency RP0, RP1..... enabled, disabled 0000-2400 0000-2400 M, T, W, R, F, S, U, H or any combination 0-999999999 list of media-profiles <media-profile> <carrier-code> SAG:<sag>, <sa>, ipAddress, FQDN empty or <realm> SIP or H323 enabled/disabled	4.0.0 and above
LocalPolicyAttribute startTime endTime dow cost LocalPolicyMediaProfiles acliObjectName ** carrierName nextHop # destRealm # appProtocol action isTermRoute replace	policy-attribute start-time end-time days-of-week cost media-profiles carrier next-hop realm app-protocol action terminate-recursion replace-uri	0000 0000 U-S 0 SIP none disabled	0000-2400 0000-2400 M, T, W, R, F, S, U, H or any combination 0-999999999 List of media-profiles <media-profile> <carrier> SAG:<sag>, <sa>, enum:<name> empty or <realm> SIP or H323 none, replace-uri, redirect enabled/disabled	4.0.1 and above
LocalPolicyFrom # addr ** LocalPolicyTo # addr **	from-address to-address		also support 12*34, **** for number (potsstar)	4.0.1, 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	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	* * disabled 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.

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

SIP Header Manipulation

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

SOAP Attributes /Subelements	ACLI session-router-> sip-manipulation	Default Values	Valid Values	SBC Version
aclObjectName * # HeaderRule aclObjectName * # action matchValue msgType methods ElementRule aclObjectName * # aclitype action matchValueType matchValue newValue	name header-rule name action match-value msg-type methods element-rule name type action match-val-type match-value new-value	none any none any	add, delete, manipulate, none empty or <value> any, request, reply empty or comma separated strings header-value, header-param, uri-user, uri-host, uri-port, uri-param, uri-header, uri-user-param add, replace, delete-header, delete-element, none IP, FQDN, ANY empty or <255 characters> empty or combination of \$ORIGINAL, \$LOCAL_IP, \$REMOTE_IP, \$REMOTE_VIA_HOST, \$TRUNK_GROUP, \$TRUNK_GROUP_CONTEXT, <any string>, +, -, +^, -^	4.0.0 and above
aclObjectName * # description HeaderRule aclObjectName * # headerName # action cmpType matchValue msgType newValue methods ElementRule aclObjectName * # paramName aclitype action matchValueType cmpType matchValue newValue	name description header-rule name header-name action comparison-type match-value msg-type new-value methods element-rule name parameter-name name type action match-val-type comparison-type match-value new-value	none case-sensitive any none ANY case-sensitive	255 characters 255 characters or @ status-line add, delete, manipulate, store, none case-sensitive, case-insensitive, pattern-rule empty or <reg-expr> any, request, reply empty or <reg-expr> empty or comma-separated strings 255 characters 255 characters 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, add, replace, delete-header, delete-element, store, none IP, FQDN, ANY case-sensitive, case-insensitive, pattern-rule empty or <reg-expr-value> empty or <reg-expr-value> with pre0defined parameters: \$ORIGINAL, \$LOCAL_IP, \$REMOTE_IP, \$REMOTE_VIA_HOST, \$TRUNK_GROUP, \$TRUNK_GROUP_CONTEXT	4.1.1 and above

HeaderRule action cmpType newValue ElementRule cmpType newValue action acltype	header-rule action comparison-type new-value element-rule comparison-type new-value action type	none	add, delete, manipulate, store, none, sip-manip case-sensitive, case-insensitive, pattern-rule, boolean, refer-case-sensitive, refer-case-insensitive empty, <reg-expr>, or <sip-manipulation> case-sensitive, case-insensitive, pattern-rule, boolean, refer-case-sensitive, refer-case-insensitive 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	4.1.4; 5.1.1 and above
HeaderRule matchValue ElementRule acltype matchValue	header-rule match-value element-rule type match-value		empty or <regular-expr> with \$MANIP_STRING 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 empty or <regular-expr> with \$MANIP_STRING	6.1.0 and above
HeaderRule aclObjectName * # action msgType matchValue newValue ElementRule action matchValue newValue	header-rule name action msg-type match-value new-value element-rule action match-value new-value	none any none	unique and ordered with mime-rule, mime-isup-rule add, delete, manipulate, store, none, sip-manip, find- replace-all, reject, log any, request, reply, out-of-dialog empty or <regular-expr> with \$MANIP_STRING, \$MANIP_PATTERN empty or <regular-expr> or <sip-manipulation> or ACME_NAT_TO_FROM_IP add, replace, delete-header, delete-element, store, none, find-replace-all, reject, log, sip-manip empty or <regular-expr> with \$MANIP_STRING (\$M_STRING), \$MANIP_PATTERN empty or <sip-manipulation> or <reg-expr> with reserved words and operators: \$ORIGINAL, \$LOCAL_IP/PORT, \$REMOTE_IP/PORT, \$REMOTE_VIA_HOST, \$TRUNK_GROUP (\$T_GROUP), \$TRUNK_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, &, , ==, !=, <=, >=, <, >	6.2.0 and above

MimeRules aclObjectName * #	mime-rule name		unique and ordered with header-rule, mime-isup-rule	6.2.0 and above
contentType	content-type		255 chars such as application/SDP, @preamble, @epilogue	
action	action	none		
cmpType	comparison-type	case-sensitive	add, delete, manipulate, store, none, sip-manip, find-replace-all, reject, log	
msgType	msg-type	any	case-sensitive, case-insensitive, pattern-rule, boolean, refer-case-sensitive, refer-case-insensitive	
format	format	ascii-string	any, request, reply, out-of-dialog	
methods	methods		ascii-string, hex-ascii, binary-ascii	
matchValue newValue	match-value new-value		empty or comma seperated strings	
MimeHeaderRule aclObjectName *#	mime-header-rule name mime-header-name		255 chars	
mimeHeaderName #	action	none	255 chars such as Content-Disposition	
action	comparison-type	case-sensitive	add, replace, store, none, sip-manip, find-replace-all, reject, log	
cmpType	match-value new-value		case-sensitive, case-insensitive, pattern-rule, boolean, refer-case-sensitive, refer-case-insensitive	
matchValue newValue				

MimeISUPRules acliObjectName * #	mime-isup-rule name		unique and ordered with header-rule, mime-isup-rule	6.2.0 and above
contentType	content-type		255 chars such as application/ISUP	
isupSpec	isup-spec	ansi-2000	ansi-2000, itu-99, gr-317, etsi-356	
isupMsgTypes	isup-msg-types		empty or comma seperated list of 1-255	
action	action	none	add, delete, manipulate, store, none, sip-manip, find-replace-all, reject, log	
cmpType	comparison-type	case-sensitive	case-sensitive, case-insensitive, pattern-rule, boolean, refer-case-sensitive, refer-case-insensitive	
msgType	msg-type	any	any, request, reply, out-of-dialog	
methods	methods		empty or comma seperated strings	
matchValue newValue	match-value new-value			
MimeHeaderRule	mime-header-rule			
MimeISUPParamRule	isup-param-rule			
acliObjectName *#	name		255 chars	
parameterType	type	0	0-255	
parameterFormat	format	hex-ascii	number-param, hex-ascii, binary-ascii, ascii-string, bcd	
action	action	none	add, replace, store, none, sip-manip, find-replace-all, reject, log	
cmpType	comparison-type	case-sensitive	case-sensitive, case-insensitive, pattern-rule, boolean, refer-case-sensitive, refer-case-insensitive	
matchValue newValue	match-value new-value			
splitHdrList joinHdrList	split-headers join-headers		comma seperated list of header names such as "Allowed,P-Asserted-Identity", "Diversion,Allow"	6.2.0M1

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 * # useConstraints maxNumSessions maxInbSessions maxOutbSessions maxBurstRate maxInbBurstRate maxOutbBurstRate maxSustainedRate maxInbSustainedRate maxOutbSustainedRate minSeizure minAnswerSeizureRatio timeRoResume noResponseTo inservicePeriod burstWindow sustainedWindow	name state max-sessions max-inbound-sessions max-outbound-sessions max-burst-rate max-inbound-burst-rate max-outbound-burst-rate max-sustain-rate max-inbound-sustain-rate max-outbound-sustain-rate min-seizures min-asr time-to-resume ttr-no-response in-service-period burst-rate-window sustain-rate-window	disabled 0 0 0 0 0 0 0 0 0 0 5 0 0 0 0 0	24 characters enabled/disabled 0-999999999 0-999999999 0-999999999 0-999999999 0-999999999 0-999999999 0-999999999 0-999999999 0-999999999 1-999999999 0-100 0-999999999 0-999999999 0-999999999 0-999999999	4.0.1; 4.1.1 above
SessionConstraintRateConstraints method maxInBurstRate maxOutBurstRate maxInSustainedRate maxOutSustainedRate	method max-inbound-burst-rate max-outbound-burst-rate sustain-rate-max-outbound max-outbound-sustain-rate		INVITE, ACK, BYE, REGISTER, CANCEL, PRACK, OPTIONS, INFO, SUBSCRIBE, NOTIFY, REFER, UPDATE, MESSAGE, PUBLISH 0-999999999 0-999999999 0-999999999 0-999999999	5.1.1 and above

Session Translation

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

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

Translation Rules

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

SOAP Attributes /Sub-elements	ACLI session-router->translation-rule	Default Values	Valid Values	SBC Version
id * # aclitype add_s add_indx del_s del_indx	id type add-string add-index delete-string delete-index	none 0 0	add, delete, replace, none <string> 0-999999999, \$ for appending at the end @ as wild char or <string> 0-999999999	4.0.0 and above

RPH Profile

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

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

RPH Policy

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

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

Host Routes

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

SOAP Attributes /Sub-elements	ACLI system->host-route	Default Values	Valid Values	SBC Version
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 session-router->local-response-map->entries	Default Values	Valid Values	SBC Version
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, rcv-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, rcv-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, request-method-throttled empty, REGISTER 0-999999999	5.1.1 and above

Codec Policy

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

SOAP Attributes /Sub-elements	ACLI media manager->codec-policy	Default Values	Valid Values	SBC Version
acliObjectName *# CodecPolicyAllow acliObjectName *# CodecPolicyOrder acliObjectName *#	name allow-codecs order-codecs		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

Access Control

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

SOAP Attributes /Sub-elements	ACLI session-router->access-control	Default Values	Valid Values	SBC Version
inRealm * inSrc *# inDst *# appProtocol *# transProtocol *# access rateLimit trustLevel errMsgThreshold maxMsgThreshold denyTimer maxMsgThresholdUntrusted	realm-id source-address destination-address application-protocol transport-protocol access average-rate-limit trust-level invalid-signal-threshold maximum-signal-threshold deny-period untrusted-signal-threshold	0.0.0.0 0.0.0.0 all permit 0 none 0 0 30 0 or the same as average-rate-limit	<realm> ipv4/mask:port/mask ipv4/mask:port/mask for application-protocol, NONE NONE, SIP, MGCP TCP, UDP, all permit, deny 0-4294967295 none, low, medium, high 0-4294967295 0-4294967295 0-4294967295	4.0.0
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 untrustedCacFailThreshold	cac-fail-threshold untrusted-cac-failure-threshold	0 0	0-999999999 0-999999999	6.2.0 and above
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 *# mediaType payloadType transport reqBandwidth framesPerPacket parameters avgRate peakRate maxBurstSize	name media-type payload-type transport req-bandwidth frames-per-packet parameters average-rate-limit peak-rate-limit max-burst-size	audio 0 RTP/AVP 0 0 0 0 0 0	24 characters audio, video, data, application, control RTP/AVP, UDP 0-999999999 0-256 space separated <name=value> pair 0-125000000 0-125000000 0-125000000	4.0.0 and above
mediaType sdpRateLimit sdpBandwidth	media-type sdp-rate-limit-headroom sdp-bandwidth	audio 0 disabled	audio, video, data, application, control, imate, text 0-100 enabled or disabled	4.1.1 and above
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 session-router->slp-response-map	Default Values	Valid Values	SBC Version
aclObjectName *# SIPResponseMapEntry statusRcvd *# statusSend # reason	name entries rcv-code xmit-code reason		100-699 100-699	4.0.0 and above
method registerResponseExpires	method register-response-expires	0	empty, REGISTER 0-999999999	5.1.1 and above

Diameter Director Agent

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

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

redundancyPort	redndancy-port	1999	Valid port number	DD1.0.0 and above
IIAction	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	
IIExpResultCode	load-limit-exp-result-code			
statefulPolicy	stateful-policy			
IIResultCode	load-limit-result-code	3004	Valid 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 session-router->diameter-director-constraints	Default Values	Valid Values	SBC Version
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	
timeToResume	time-to-resume	0	0-999999	
name	name			
maxSustainedRate	max-sustained-rate	0	0-999999	
lastModifiedBy	last-modified-by			
resultCode	result-code	3004	1000-6000	
lastModifiedDate	last-modified-date			
messageRateConstraints	message-rate-constraints			
maxOutSustainedRate	max-outbound-sustain-rate	0	0-999999	
maxInSustainedRate	max-inbound-sustain-rate	0	0-999999	
maxInBurstRate	max-inbound-burst-rate	0	0-999999	
command	command			
maxOutBurstRate	max-outbound-burst-rate	0	0-999999	

Diameter Director Group

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

SOAP Attributes /Sub-elements	ACLI session-router->diameter-director-constraints	Default Values	Valid Values	SBC Version
-------------------------------	--	----------------	--------------	-------------

groupName description lastModifiedBy state lastModifiedDate strategy inManipulationId outManipulationId recursiveRouting recursionTimeout doRecursion transactionTimeout resultCodes expResultCodes diamDirApplication appld vendorId appType destination seqno name	group-name description last-modified-by state last-modified-date strategy in-manip-id out-manip-id recursive-routing recursion-timeout do-recursion transaction-timeout result-codes exp-result-codes diameter-director-application application-id vendor-id application-type destinations seqno name	enabled hunt 100000 disabled 32000 0 authentication	256 character string enabled or disabled hunt diameter-manipulation rule diameter-manipulation rule 1-999999 enabled or disabled 1-999999 valid diameter result code 32-bit hexadecimal or 32-bit integer 32-bit integer authentication or accounting	DD1.0.0 and above
---	--	---	--	-------------------

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 outManipId state routingPolicy realmId suppVendorIds originHostId description originHostFormat lastModifiedBy lastModifiedDate inManipId slpPort port tlsProfile transProtocol address anonMode imsAkaProfile sctpMultiHomeAddrs diamDirApplication appld vendorId appType	constraint-name out-manip-id state routing-policy realm-id supported-vendor-ids origin-host-identifier description origin-host-format last-modified-by last-modified-date in-manip-id diameter-director-ports port tls-profile transport-protocol address allow-anonymous ims-aka-profile multi-home-addrs diameter-director-application application-id vendor-id application-type	enabled none all 0 authentication	diameter director constraint diameter-manipulation rule disabled 256 character string 256 character string 256 character string none, identifier, identifier-with-realm diameter-manipulation rule number greater than 1023 TCP, SCTP IP address all, agents-only IP address 32-bit hexadecimal or integer 32-bit integer authentication, accounting	DD1.0.0 and above

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

Appendix B Backwards Compatibility

This Appendix provides a procedure which allows client applications created with deprecated provisioning APIs to be run with the generic SOAP XML Provisioning API introduced with Net-Net Central Version 7.0

Significant platform differences between older deprecated provisioning APIs and the new generic SOAP/XML Provisioning API are summarized below.

	Deprecated APIs	Preferred API
SOAP Engine	Axis 1.1	CFX
EMS Versions	pre-7.0	7.0 and later
Java SDK	1.4.2 or later	1.6.0 or later
	1.5.14 or later	
JRE	1.4.2 or 1.5	1.6.0 Development Package
	Development Package	

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 ACME Packet 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 Net-Net Central 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.

- 7a. Use FTP to move a copy of a Net-Net Central 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 Net-Net Central Server.

- 7b. 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
```

- 7c. Edit *run.bat* by changing the TRUST_STORE variable to match the location of the JAVA keystore that contains the public certificates of associated Net-Net Central Servers.

- 7d. Edit *run.bat* by changing the TRUST_STORE_PASSWORD to match the password required to access the JAVA keystore containing the Net-Net Central Server certificates.

- 7e. 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.

- 7f. 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.

