# Oracle® Communications
# Evolved Communications Application Server

Security Guide

Release 7.1

**E66298-01**

May 2016

ORACLE®

Oracle Communications Evolved Communications Application Server Security Guide, Release 7.1

E66298-01

# Contents

# Preface

This document describes security features and configuration of Oracle Communications Evolved Communications Application Server (OCECAS).

## Audience

This document is intended for administrators who configure security for OCECAS.

## Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

### Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

## Accessing Oracle Communications Documentation

OCECAS documentation is available from the Oracle Documentation website: http://docs.oracle.com.

## Related Documents

For more information, see the following documents in the OCECAS documentation:

OCECAS is based on Oracle products, including the Oracle Communications Converged Application Server, Oracle WebLogic Server, and Oracle Databases. For more information, see the following documents:

- *Oracle Communications Evolved Communications Application Server System Administrator's Guide*
- *Oracle Communications Converged Application Server Security Guide* in the Converged Applications Server Version 7.0 documentation
- *Oracle Communications Converged Application Server Administrator's Guide* in the Converged Applications Server Version 7.0 documentation

- *Oracle Fusion Middleware Securing a Production Environment for Oracle WebLogic Server* in the Oracle WebLogic Server 12*g* documentation
- *Oracle Fusion Middleware Securing Oracle WebLogic Server* in the Oracle WebLogic Server 12*g* documentation
- *Oracle Fusion Middleware Application Security Guide* in the Oracle WebLogic Server 12*g* documentation
- *Oracle Application Server Security Guide*
- *Oracle Application Server Administrator's Guide*
- *Oracle Database Security Guide*

# 1

# OCECAS Security Overview

This chapter describes the Oracle Communications Evolved Communications Application Server (OCECAS) security features.

## Basic Security Considerations

The following principles are fundamental to using any application securely:

- **Keep software up to date**. This step includes the latest product release and any patches that apply to it.
- **Limit privileges as much as possible**. Only give user accounts the access necessary to perform their work. Review user privileges periodically to determine relevance to current work requirements.
- **Monitor system activity**. Establish who should access which system components, and how often, and monitor those components.
- **Install software securely**. For example, use firewalls, secure protocols using TLS (SSL) and secure passwords.
- **Learn about and use the OCECAS security.** See "Implementing OCECAS Security" for more information.
- **Keep up to date on security information**. Oracle regularly issues security-related patch updates and security alerts. You must install all security patches as soon as possible. See the “Critical Patch Updates and Security Alerts” website:

  `http://www.oracle.com/technetwork/topics/security/alerts-086861.html`

## Overview of OCECAS Security

OCECAS relies on, and benefits from the security features of the underlying WebLogic Server platform, including security realms, security monitoring features, and so on.

This guide describes the OCECAS security features. For WebLogic Server information, including information about implementing application security, see the Oracle WebLogic Server 12c documentation.

OCECAS includes these components that you install separately. Each has its own security considerations:

- Session Design Center User Interface
- OCECAS Session Control Features
- Universal Data Record (UDR) Server

### Session Design Center User Interface

Session Design Center runs in the OCECAS management environment. You use it to create and deploy the VoLTE multimedia services that your subscribers use. Session design Center requires the WebLogic server and an Oracle database, and it has these security considerations:

- The WebLogic server may use a local security realm, or integrate with an external security provider such as LDAP or Oracle Identity Manager. Configure the WebLogic server running the management application securely.
- Make sure the WebLogic Administration is only accessible to administrators; do not give any other user accounts access to it. On production environments, shut down the Administration Console when not in use.
- The Oracle Database must be secured to allow access by OCECAS only. Ensure that administrators are the only user accounts with access to it.
- Secure the interfaces provided to other servers, such as REST, JDBC, and Messaging.
- Event Data Records (EDRs) are collated on the Session Design Center server from the management environment and the UDR server. Restrict access to the EDRs to protect privacy.

### OCECAS Session Control Features

The session control features provide SIP Application Server functionality to operators. The session control features are based on WebLogic Server. Your deployment may also require a NoSQL database.

- Ensure that OCECAS is only accessible by administrators. Make sure that no other user accounts have access to it. Once you configure a production environment, shut down the Administration Console.
- Secure WebLogic server components that provide connections to the management server.
- Whenever possible, secure communication channels between OCECAS and the IMS, HSS, OCF, and other network nodes using transport layer security (TLS).

### Universal Data Record (UDR) Server

A UDR provides a repository for subscriber data, and can be installed on separate physical hardware, or on the same hardware as the OCECAS environments.

- Configure the server running the UDR application for secure access.
- Ensure that the administration console is only accessible to administrators, and shut it down on production environments.

## Understanding the OCECAS Environment

When planning your OCECAS implementation, consider the following:

- Which resources need to be protected?
  - Protect customer data, such as IP addresses.
  - Protect internal data, such as EDRs.

  - Protect system components from being disabled by external attacks or intentional system overloads.

- Who are you protecting data from?

  For example, you need to protect subscriber data from other subscribers, but someone in your organization might need to access that data to manage it. You can analyze your workflows to determine who needs access to the data. For example, a system administrator may be able to manage your system components without needing to access the system data.

- What happens if protections on a strategic resource fails?

  In some cases, a fault in your security scheme is nothing more than an inconvenience. In other cases, a fault might cause great damage to you or your customers. Understanding the security ramifications of each resource helps you protect it properly.

# 2

# Performing a Secure OCECAS Installation

This chapter presents planning information for your Oracle Communications Evolved Communications Application Server (OCECAS) system and describes recommended deployment topologies that enhance security.

For more information about installing OCECAS, see *Oracle Communications Evolved Communications Application Server Installation Guide.*

## Installing OCECAS Securely

When installing OCECAS, perform these steps for each domain:

- You have the option to perform a typical installation or a custom installation. Perform a custom installation to avoid installing options and products you do not need. If you perform a typical installation, remove or disable features that you do not need after the installation.
- Disable all non-SSL ports to secure all communication between components, such as with SIP, Diameter, and HTTP traffic.
- Make sure that you enable and use SSL ports for the administration servers for all OCECAS domains. Change the default port numbers.
- If installing OCECAS on a cluster of servers, configure the cluster addresses to use SSL ports.
- After you have created the WebLogic domains for OCECAS, start the administration server. Then, use t3s to start the managed servers:

  ```
  startManagerServer.sh ManagedServer_1 t3s://host_name
  ```

  where *ManagedServer_1* is the name of the first managed server, and *host_name* is the host name of the administration server.
- Using the Administration Console, configure certificate identity and trust store to use SSL. Do not use the default, demonstration certificate that comes with WebLogic Server. See the WebLogic Server security and system administration documentation for more information.

### About Access to Files Created During Installation

Access to files created during the installation is limited. The user account that installs OCECAS has write access to the files created during installation.

### About Password Policies

Oracle recommends having strong password policies for OCECAS. Consider enforcing the following password policies:

- Require that passwords have a minimum of eight characters.
- Passwords must contain at least one digit, one capital letter, and one special character.
- The user name must not be part of the password.

Stricter rules can be set for the authentication provider using the Administration Console. For details on authentication providers and their configuration, refer to the discussion on securing Oracle WebLogic Server in the WebLogic Server documentation.

See *Oracle Communications Evolved Communications Application Server System Administrator's Guide* for information about changing and setting OCECAS passwords.

## Post-Installation Configuration

This section explains security configurations to complete after OCECAS is installed.

### Setting Up User Accounts to Lock and Expire

Create OCECAS user accounts and configure them to lock after several failed login attempts, and to expire after a certain period of idle time.

See *Oracle Communications Evolved Communications Application Server System Administrator's Guide* for information about changing and setting OCECAS passwords.

### Enabling SSL for LDAP Authentication Providers

For secure communication between WebLogic Server and an external LDAP, enable SSL on both the external LDAP authentication provider and the corresponding WebLogic Security Provider. SSL on the WebLogic security provider is enabled from the Administration Console.

For information about secure communication between WebLogic Server and an external LDAP authentication provider, see *Oracle Fusion Middleware Securing Oracle WebLogic Server*.

# 3

# Implementing OCECAS Security

This chapter describes the specific security mechanisms provided by Oracle Communications Enhanced Communications Application Server (OCECAS).

## About OCECAS Security

OCECAS is built upon the framework of Oracle Fusion Middleware WebLogic Server and Oracle Communications Converged Application Server. Session Design Center GUI users are authenticated against credentials stored in the WebLogic server, or against an external LDAP provider. Session Design Center and the REST API it uses must be deployed using SSL. OCECAS configuration settings prompt you for a user name and password, using the basic authentication method. JavaScript resources are not protected.

### About TLS (SSL)

Web browsers connect to OCECAS over an HTTP port or on HTTP with a TLS (SSL) port. You can use TLS connections to secure communications to these interfaces:

- Session Design Center UI interface (to browser)
- Session Design Center REST API interface (to browser)
- UDR REST API interfaces (to provisioning system)
- SIP interface (to IMS)
- Diameter Ro/Rf and Sh interfaces (to charging engines and HSS)
- NoSQL interface (to NoSQL repository)
- XCAP interface (to IMS)
- JDBC interface (to Oracle Database)

  For information on configuring a secure JDBC interface to Oracle database, see "Using SSL and Encryption with Data Sources and Oracle Drivers" in *Administering JDBC Data Sources for Oracle WebLogic Server*.

Using TLS, all network communication between the web browser and the server is encrypted, which means that sensitive information is never in clear text.

As a minimum authentication requirement, the server is required to present a digital certificate to the web browser client to prove its identity.

You enable TLS (SSL) by using the Administration Console.

# Deploying OCECAS Securely

This section describes recommended deployment configurations for OCECAS.

Deploy OCECAS and the UDR components it contains entirely on a private network, typically as part of an IMS Network. Do not allow direct access to any of these servers.

Event Data Records (EDRs) are saved in files on the Management server. Only the user created for installing OCECAS has access to the EDR files. See "Using EDRs for Testing and Troubleshooting" in *Evolved Communications Application Server System Administrator's Guide* for more information.

You can provision subscribers in a NoSQL database created as part of the UDR domain, using a RESTful API. Secure the RESTful API using basic authentication over TLS; secure NoSQL by configuring **kvStore** with TLS and authentication enabled.

Use the Administration Console to maintain and secure all configuration for OCECAS.

The OCECAS deployment pipeline defines a change-management architecture that isolates session state between runtime domains. Changes on, and actions toward a runtime domain cannot affect the other domains in the pipeline.

## Session Design Center GUI

The Session Design Center GUI is a web-based management application for multiple OCECAS domains. The application is deployed in the Management System domain. Configure this server to use SSL TLS.

You secure access to Session Design Center using a WebLogic security realm and a dedicated group. Only users with membership of the group are authorised to access the UI. The security realm can use a local LDAP directory, an external LDAP directory, or another security provider such as Oracle Identity Manager. Once configured, shut down the Administration server to prevent unauthorized access.

See *Oracle Fusion Middleware Securing a Production Environment for Oracle WebLogic Server* for further details.

Where required, enable application logging to diagnose problems with the Session Design Center GUI control flows. By default, only messages at the INFO level are output to log files. No sensitive information is logged at the INFO level. Oracle strongly recommends that you do not use DEBUG level on a production environment.

The Session Design Center is accessible to service designers. This part of the management server is typically accessible on a corporate network to allow service engineers access to the management application. Manage access to the management application locally using the WebLogic local LDAP security realm, or integrate it with a third-party security provider.

Once all applications are installed and configured, Oracle recommends that you shut down all Administration Servers to prevent unauthorized access.

# OCECAS Backend Security

This section explains security considerations for the software underlying OCECAS.

## Operating System Security

See these documents for information about securing the Linux operating system that OCECAS runs on:

- Guide to the Secure Configuration of Red Hat Enterprise Linux 5, available from the National Security Agency website here:

  https://www.nsa.gov/ia/_files/os/redhat/rhel5-guide-i731.pdf

- Hardening Tips for the Red Hat Enterprise Linux 5, available from the National Security Agency website here:

  https://www.nsa.gov/ia/_files/factsheets/rhel5-pamphlet-i731.pdf

## WebLogic Server Security

You can integrate Session Design Center GUI with an external security provider, allowing a single sign-on approach to accessing the web-based application.

The Session Design Center GUI application is deployed to a **mgmt1** server within WebLogic by default. **mgmt1** is the default name for the managed server running the Session Design Center UI and REST APIs in the management domain. Configure this server so that only SSL access is allowed, chiefly to prevent passwords being submitted in plain text on login.

Any users created in the WebLogic security realm must be added to the **EvolvedCommunicationUsers** group in order to access the application. See *Oracle Fusion Middleware Securing Resources Using Roles and Policies for Oracle WebLogic Server* for more information.

## JDBC Security

The default Session Design Center GUI implementation does not provide a secure configuration for JDBC connections to the database. To secure these connections requires you to configure secure client connections to the database.

See *Oracle Database Security Guide* for more information about securing client connections to the database.

## Securing Coherence

Oracle Coherence is used internally within OCECAS and its nodes. Coherence security includes securing both cluster members and extends clients. You enable security as required, based on your application or cluster implementation, and the security concerns and tolerances of your organization. For a brief discussion of each security feature, see *Oracle Coherence Security Guide.*

## Securing Ports

Configure firewalls to restrict access internally. Oracle recommends that you disable insecure port 7001 on the managed servers and use port 7002 instead. You can enable 7002 during domain installation. You remove the non-SSL port by using the Administration Console (**Environment**, then **Servers**, then **AdminServer**).

Table 3–1 lists the default ports, their names, and security considerations. Also see Table 3–2 for a list of the NoSQL ports that you must manage.

***Table 3–1*** OCECAS ***Default Ports***

| Domain | Value | Description | Security Considerations |
|---|---|---|---|
| All | 5556 | NodeManager Port | Make this port accessible to the machine running the AdminServer for the Node Manager domain. |
| All | 7001 | Insecure Administration Server HTTP port | Disable this port by using the Administration Console and use the SSL port (7002) instead. |
| All | 7002 | Secure Administration Server HTTPS port (admin/http/t3) | Make this port accessible to administrators during domain installation. Use instead of the insecure port (7001). |
| All | 8088 | Coherence port<br>(a unique port is required for each runtime domain. For example: 8088 for **scf_testing_domain**, 8188 for **scf_staging_domain**, and so on) | Make each port accessible to all other managed servers within the domain. |
| Management | 1521 | Oracle Database port | Make this port accessible to the managed servers in the runtime domains. |
| Management | 6002 | OCECAS Management Server | Make this port accessible to:<br>■ Runtime Managed Servers<br>■ UDR Managed Servers<br>■ Session Design Center GUI users |
| Runtime | 5060 | The SIP port<br>(unique per runtime managed server) | Allow access to the IMS network from this port. Restrict access to this port to trusted hosts only, for example S-CSCF for IMS traffic and OCCAS-SC, if supporting deployment with MSC Not Enhanced for ICS. |
| Runtime | 5061 | The SIPS port<br>(unique per runtime managed server) | Allow access to the IMS network from this port. Restrict access to this port to trusted hosts only, for example S-CSCF for IMS traffic and OCCAS-SC, if supporting deployment with MSC Not Enhanced for ICS. |
| UDR | 6052 | UDR Server HTTPS port | Make this port accessible to the system that provisions subscribers. |
| Runtime | 8001 | Insecure 3GPP HTTP port for XCAP interface | Disable the port using the Administration console. |
| Runtime | 8002 | Secure 3GPP HTTP port for XCAP interface. | Make this port accessible only to Authentication Proxy(s) if OCECAS is deployed in IMS using TS 3GPP 22.222 Generic Authentication Architecture (GAA). Otherwise, restrict access to proxies from which XCAP requests are expected. |

## Oracle Database Security

See *Oracle Database Security Guide* for details.

## Securing the NoSQL Database

OCECAS supports secure authenticated access to a NoSQL database, and Oracle recommends that you use this option. To configure secure authentication:

- Install and configure NoSQL securely. See *Oracle NoSQL Database Security Guide* for details.
- Use the Administration Console to configure credentials to access the runtime and UDR domains for accessing NoSQL. Both the runtime application and the UDR REST API retrieve these credentials.

  Use these mapping details:

  - **protocol=NOSQL**
  - **remoteHost=localhost**
  - **remotePort=5000**
  - **path=** (not used)
  - **local user =** *local_username* (for example, OCCAS)
  - **method=connect**
  - **remote user =** *NoSQL_username*
  - **remote password =** *NoSQL_password*
- Configure SSL for NoSQL by specifying the location of the Java truststore in the **setUserOverrides.sh** file. The truststore contains the public certificate that validates the NoSQL server. You edit this file in the managed servers of every runtime and UDR domain. See "SSL Model" in *Oracle NoSQL Database Security Guide* for more information.
- Manage the NoSQL ports listed in Table 3–2 to secure OCECAS runtime domains that include NoSQL databases.

***Table 3–2 NoSQL Management Ports***

| Port No | Description | Security Considerations |
|---|---|---|
| 5000 | NoSQL Data Store Port | Make this port accessible to the managed servers in the runtime domains, and to the managed servers. |
| 5010-5020 | NoSQL replication ports | Replication Nodes use this range of ports to communicate among themselves. Each machine hosting a NoSQL store must have access to the other NoSQL store instances using this port range. See *Oracle NoSQL Database Administrator's Guide* for more information. |
| 5110-5120 | NoSQL service ports | Administrative servers running on a storage node use this range of ports to communicate with managed services. |

## SIP Security

Where possible, use SIPS for all SIP communication. Your P-CSCF authenticates SIP traffic against information stored in the HSS. Requests forwarded to OCECAS contain the P-Asserted-Identity header which OCECAS honors. For more information, see

"Overview of SIP Servlet Identity Assertion Mechanisms" in *Oracle Communications Converged Application Server Security Guide*.

## XCAP Security

Where possible, use HTTPS for all XCAP communications.

## OCECAS to SIP Security

Other ways of improving security for OCECAS include securing SIP and handling challenges from the IMS Core.

### Securing SIP

OCECAS offers secure SIP (SIPS) connections, using TLS to secure signalling. OCECAS also uses two-way SSL to verify the digital certificate supplied by the client. Ensure that a SIPS transport (SSL) has been configured in order to use client-certificate authentication. For more information about configuring SSL, see *Oracle Database Advanced Security Administrator's Guide.*

## Securing SNMP

Weblogic Server includes functions that map credentials for managing remote user names and passwords securely. It also has functions for retrieving those remote user names and passwords. After the remote SNMP trap manager changes the passwords, the credential mappings can be reconfigured by using the Administration Console.

To secure SNMP, complete these tasks:

- Creating SNMP Credential Mapping on Runtime Domains
- Configuring SNMP Privileges on the Management Domain
- Granting SNMP AlarmMX Permission
- Configuring SNMP Trap Details

### Creating SNMP Credential Mapping on Runtime Domains

OCECAS supports the SNMPv3 authentication and privacy features. For a secure installation, enable both using the Administration Console:

1. Open the Administration Console for a runtime domain.

   See *Evolved Communications Application Server System Administrator's Guide* for details.

2. From the Domain Structure, click **Security Realms**.

   The Summary of Security Realms page appears.

3. Select a security realm.

   The Settings for *RealmName* page appears.

4. Click the **Credential Mappings** tab, and then click the **Default** subtab.

5. Click **New**.

   The Create a New Security Credential Mapping page appears.

6. Create a credential mapping for storing the authentication pass-phrase, by specifying the following:

- **Protocol**. For example **SNMP**.
- **Remote Host**. Enter the SNMP management system remote host IP address.
- **Remote Port**. Enter the SNMP management system remote port.
- **Path**. The path to the SNMP credential store on the SNMP management system.
- **Method**. Enter the authentication method, for example **auth**.

7. Click **Next**.
8. Specify the following:
   - **Local User**. Enter the name of the WebLogic Server user for this credential mapping.
   - **Remote User**. Enter the name of the remote user to which the local user maps.
   - **Remote Host**. Enter the SNMP management system remote host IP address.
   - **Remote Password**. The authentication pass phrase (provided by the SNMP Management System).
9. Click **Finish**.

To create a credential mapping for storing the privacy pass-phrase, repeat the steps above with the substitutions shown here:

- **Method - priv**
- **Remote password** - The privacy pass phrase provided by the SNMP management system.

Note the **Resource Identifier** of each mapping created by WebLogic, for example:

```
Auth: type=<remote>, protocol=SNMP, remoteHost=snmphost, remotePort=162,
method=auth
Priv: type=<remote>, protocol=SNMP, remoteHost=snmphost, remotePort=162,
method=priv
```

### Configuring SNMP Privileges on the Management Domain

On the management domain that triggers SNMP traps:

1. Open the Administration Console.

   See *Evolved Communications Application Server System Administrator's Guide* for details.
2. Navigate to **Evolved Communications**, then **Alarm**, then **Advanced**.
3. Configure the settings as follows:
   - **Security Level**: **AUTH_PRIV**
   - **Security Username**: The username to access the SNMP management system.
   - **Authorization Resource ID**: copy the Resource Identifier for method **auth** created by WebLogic in "Creating SNMP Credential Mapping on Runtime Domains".
   - **Privacy Resource ID**: Enter the **Resource Identifier** for method **auth** created by WebLogic in "Creating SNMP Credential Mapping on Runtime Domains".

### Granting SNMP AlarmMX Permission

To grant AlarmMX permission to invoke operations provided by the **DefaultCredientialMapperMBean**:

1. Log in to the Administration Console for the management domain.

   See *Evolved Communications Application Server System Administrator's Guide* for details.

2. Navigate to **Security Realms**, and then click *realm_name*.
3. Enable **Use Authorization Providers to Protect JMX Access**.
4. Activate the change.
5. Restart the Administration Server.
6. Navigate to **Roles and Policies**, then **Realm Policies**, and then **JMX Policy Editor**.
7. Ensure that **GLOBAL SCOPE** is selected.
8. Click **Next**.
9. Expand **weblogic.security.providers.credentials**.
10. Click **DefaultCredentialMapperMBean**.
11. Click **Next**.
12. Click **Lookup Operations: Permission to Invoke Expand Node Lookup Operations: Permission to Invoke**.
13. Click **Create Policy.**

    Create a policy for the **weblogic** user.

See *Oracle Communications Converged Application Server System Administrator's Guide* for more information.

### Configuring SNMP Trap Details

SNMP trap details and the management host & port are configured by using the Administration Console. Navigate to **Evolved Communications Configuration**, then **Alarm** to make changes.

## JMS Security

The Session Design Center GUI uses JMS Modules. The JMS Modules for EDR generation are secured by default. However, the JMS Module for **scfCompiler** uses a JDBC Connection to the database, which is not secure by default.

See *Oracle Fusion Middleware Securing a Production Environment for Oracle WebLogic Server* and *Oracle Database Security Guide* for more information about securing database clients.

## Securing the UDR Server

User credentials secure the UDR APIs for subscriber provisioning.

Also see:

- Operating System Security
- WebLogic Server Security

- Oracle Database Security

# Securing Client-to-OCECAS Authentication

OCECAS uses basic authentication to secure the Session Design Center GUI and its associated REST interface. Credentials are passed in clear-text. Therefore, configure the server to support only secured access, HTTP over SSL.

## About Basic Authentication

OCECAS provides basic authentication by default. Basic authentication uses HTTP headers to transmit the user name and password to OCECAS. Basic authentication is not recommended for production systems unless you can ensure that all connections between clients and the WebLogic SIP server instance are secure.

With basic authentication, a client requests access to a protected resource. The web server displays a login screen that requests the user name and password. The client then submits the user name and password to the server. The server validates the credentials and, if successful, returns the requested resource.

HTTP basic authentication is not secure. Basic authentication sends user names and passwords over the Internet as text that is uu-encoded (UNIX-to-UNIX encoded) but not encrypted. This form of authentication, which uses base64 encoding, can expose your user names and passwords unless all connections are over SSL. If someone can intercept the transmission, the user name and password information can be easily decoded.

# Implementing Authorization and Permissions

OCECAS supports fine-grained authorization of user actions through the assignment of roles, which have associated permissions. OCECAS uses the underlying Oracle Platform Security Services (OPSS) framework to implement the authorization of user actions. OPSS provides security to Oracle Fusion Middleware, including WebLogic Server.

Authorization policy is established when a grant is made that links principals with permissions. Principals are application roles to which users or groups are mapped. Application roles are fixed for a version of OCECAS and are created from templates during installation. Application roles are assigned to users through the administration console. For information on assigning roles and administering authorization and permissions, see *Oracle Fusion Middleware Securing Resources Using Roles and Policies for Oracle WebLogic Server*.

An application role is defined by its permissions to take action on OCECAS resources. Table 3–3 describes the application roles that OCECAS implements:

***Table 3–3 OCECAS Application Roles***

| Resource | Create | Delete | Modify | View |
|---|---|---|---|---|
| **Configuration** | NA | NA | NA | NA |
| Application | Designer | Designer | Editor | Viewer |
| Change Set | Designer | Designer | Editor | Viewer |
| Charging Template | Designer | Designer | Editor | Viewer |
| Control Flow | Designer | Designer | Editor | Viewer |

*Table 3–3 (Cont.) OCECAS Application Roles*

| Resource | Create | Delete | Modify | View |
|---|---|---|---|---|
| External Concept | Designer | Designer | Editor | Viewer |
| Locale | Designer | Designer | Editor | Viewer |
| Media Resource | Designer | Designer | Editor | Viewer |
| Media Server | Designer | Designer | Editor | Viewer |
| Notification | Designer | Designer | Editor | Viewer |
| Schema | Designer | Designer | Editor | Viewer |
| Statistic | Designer | Designer | Editor | Viewer |
| Private resource methods | Restricted | Restricted | Restricted | Restricted |
| **Environments** | NA | NA | NA | NA |
| Testing type | Tester | Tester | Tester | Viewer |
| Production type | Production Administrator | Production Administrator | Production Administrator | Viewer |
| **Service Provider** | NA | NA | NA | NA |
| Service Provider | Service Provider | Service Provider | Service Provider | Service Provider |
| **Security** | NA | NA | NA | NA |
| Application Roles | Security Administrator | Security Administrator | Security Administrator | Security Administrator |
| Permissions | Security Administrator | Security Administrator | Security Administrator | Security Administrator |

Permissions carry an implied inheritance of lower permissions. For example, the permission to modify carries the implied permission to view the same resource. Similarly, the permissions to Create and Delete carry the implied Modify and View permissions for the same resource.

# 4

# Developing Secure OCECAS Applications

This chapter provides information for developers about how to create secure applications for Oracle Communications Evolved Communications Application Server (OCECAS), and how to extend them without compromising security.

## About Developing Secure Applications for OCECAS

This section explains security considerations for modifying the OCECAS VoLTE and VoWiFi application.

### Securing UDR Communication

In order to access subscriber profile data, the OCECAS VoLTE and VoWiFi application requires that you create a database view file. The default location for this file is: *domain_home*/*domain_name*/**config/custom/csp.xml**. This file contains:

- Configuration for providers required to connect to external data sources.
- An embedded Groovy script for federating the data together for use by the application.

Strictly control access to **csp.xml**; the UNIX user created for OCECAS installation is the only account that should have access.

### Securing Control Flows

When creating or modifying control flows, consider security in their design. The **RemoteCopy** node allows data from another call context to be copied into the current call context. Give careful consideration when using this particular Activity.

# A

# OCECAS Secure Deployment Checklist

This appendix serves as a checklist to help you secure Oracle Communications Evolved Communications Application Server (OCECAS) and its components.

## OCECAS Security Checklist

To secure OCECAS:

- Ensure that SSL is configured for the management domain server, and that the default, non-secure port is disabled.
- Ensure that SSL is configured for the UDR application server, and that the default, non-secure port is disabled.
- Configure the WebLogic security realm and providers as required.
- During installation, choose a secure password for the **weblogic** user.
- During installation, ensure that the Oracle database in the Management domain has a secure password for the **SDC** and **SDC_ADMIN** users.
- During operating system installation, choose a secure password for the UNIX superuser.
- Configure Secure Client Connections for Oracle Database, and JDBC Connection Pools accordingly.
- Ensure that the WebLogic **AdminServer** is shut down on production environments.
- Ensure that only the installation user account has access to the **csp.xml** configuration file.
- Ensure that Diameter interfaces are securely configured.
- Ensure that NoSQL Interfaces (where applicable) are securely configured.
- Ensure that SNMP Credentials are properly and securely configured.