

Oracle® Fusion Middleware
API Guide for Oracle Identity Analytics
11g Release 1, Patch Set 1 (11.1.1.5)
E23366-01

August 2011

Oracle Fusion Middleware API Guide for Oracle Identity Analytics, 11g Release 1, Patch Set 1 (11.1.1.5)

E23366-01

Copyright © 2010, 2011, Oracle and/or its affiliates. All rights reserved.

Primary Author: Kevin Kessler

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 RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle America, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

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

Preface	vii
Audience	vii
Documentation Accessibility	vii
Related Documents	viii
Conventions	viii
1 Enabling Web Services	
1.1 To Enable Web Services	1-1
1.2 To Disable Web Services	1-2
2 User Service	
2.1 Overview	2-1
2.2 WSDL Access	2-1
2.3 API Reference	2-2
2.3.1 Find All Users by Filter String	2-2
2.3.1.1 Method Signature	2-2
2.3.1.2 Request Parameters	2-2
2.3.1.3 Response	2-2
2.3.1.4 Exception	2-2
2.3.2 Find Roles For A User	2-2
2.3.2.1 Method Signature	2-2
2.3.2.2 Request Parameters	2-2
2.3.2.3 Response	2-3
2.3.2.4 Exception	2-3
2.3.3 Find Accounts for a User	2-3
2.3.3.1 Method Signature	2-3
2.3.3.2 Request Parameters	2-3
2.3.3.3 Response	2-3
2.3.3.4 Exception	2-3
2.3.4 Create a New User	2-3
2.3.4.1 Method Signature	2-3
2.3.4.2 Request Parameter	2-3
2.3.4.3 Response	2-4
2.3.4.4 Exception	2-4
2.3.5 Create a New User if Rule-Based Role Assignment	2-4

2.3.5.1	Method Signature	2-4
2.3.5.2	Request Parameters	2-4
2.3.5.3	Response	2-4
2.3.5.4	Exception	2-4
2.3.6	Update User	2-4
2.3.6.1	Method Signature	2-4
2.3.6.2	Request Parameter	2-5
2.3.6.3	Response	2-5
2.3.6.4	Exception	2-5
2.3.7	Update User if Rule-Based Role Assignment	2-5
2.3.7.1	Method Signature	2-5
2.3.7.2	Request Parameters	2-5
2.3.7.3	Response	2-5
2.3.7.4	Exception	2-5
2.3.8	Disable Users	2-5
2.3.8.1	Method Signature	2-5
2.3.8.2	Request Parameter	2-6
2.3.8.3	Response	2-6
2.3.8.4	Exception	2-6
2.3.9	Enable Users	2-6
2.3.9.1	Method Signature	2-6
2.3.9.2	Request Parameter	2-6
2.3.9.3	Response	2-6
2.3.9.4	Exception	2-6
2.3.10	Find Business Units for a User	2-6
2.3.10.1	Method Signature	2-6
2.3.10.2	Request Parameters	2-7
2.3.10.3	Response	2-7
2.3.10.4	Exception	2-7
2.3.11	Remove All Roles From User	2-7
2.3.11.1	Method Signature	2-7
2.3.11.2	Request Parameters	2-7
2.3.11.3	Response	2-7
2.3.11.4	Exception	2-7

3 Role Service

3.1	Overview	3-1
3.2	WSDL Access	3-1
3.3	API Reference	3-2
3.3.1	Find Roles in Business Units	3-2
3.3.1.1	Method Signature	3-2
3.3.1.2	Request Parameters	3-2
3.3.1.3	Response	3-2
3.3.1.4	Exception	3-2
3.3.2	Find Users in Role	3-2
3.3.2.1	Method Signature	3-2
3.3.2.2	Request Parameters	3-3

3.3.2.3	Response	3-3
3.3.2.4	Exception	3-3
3.3.3	Assign User to Role	3-3
3.3.3.1	Method Signatures	3-3
3.3.3.2	Request Parameters	3-3
3.3.3.3	Response	3-4
3.3.3.4	Exception	3-4
3.3.4	Remove User From Role	3-4
3.3.4.1	Method Signatures	3-4
3.3.4.2	Request Parameters	3-4
3.3.4.3	Response	3-4
3.3.4.4	Exception	3-4
3.3.5	Preview User to Role Assignment.....	3-4
3.3.5.1	Method Signature	3-5
3.3.5.2	Request Parameters	3-5
3.3.5.3	Response	3-5
3.3.5.4	Exception	3-5
3.3.6	Get Roles by Names	3-5
3.3.6.1	Method Signature	3-5
3.3.6.2	Request Parameters	3-5
3.3.6.3	Response	3-5
3.3.7	Update Roles	3-5
3.3.7.1	Method Signature	3-6
3.3.7.2	Request Parameters	3-6
3.3.7.3	Response	3-6

4 Business Unit Service

4.1	Overview	4-1
4.2	WSDL Access	4-1
4.3	API Reference	4-2
4.3.1	Add Roles to a Business Unit.....	4-2
4.3.1.1	Method Signature	4-2
4.3.1.2	Request Parameter.....	4-2
4.3.1.3	Response	4-2
4.3.1.4	Exception	4-2
4.3.2	Remove Roles From a Business Unit	4-2
4.3.2.1	Method Signature	4-2
4.3.2.2	Request Parameters	4-2
4.3.2.3	Response	4-3
4.3.2.4	Exception	4-3
4.3.3	Add Users to a Business Unit.....	4-3
4.3.3.1	Method Signature	4-3
4.3.3.2	Request Parameters	4-3
4.3.3.3	Response	4-3
4.3.3.4	Exception	4-3
4.3.4	Remove Users From a Business Unit	4-3
4.3.4.1	Method Signature	4-4

4.3.4.2	Request Parameter.....	4-4
4.3.4.3	Response	4-4
4.3.4.4	Exception	4-4
4.3.5	Get Parent Business Structure for a Business Unit	4-4
4.3.5.1	Method Signature	4-4
4.3.5.2	Request Parameter.....	4-4
4.3.5.3	Response	4-4
4.3.5.4	Exception	4-4
4.3.6	Get Children of a Business Unit	4-5
4.3.6.1	Method Signature	4-5
4.3.6.2	Request Parameter.....	4-5
4.3.6.3	Response	4-5
4.3.6.4	Exception	4-5
4.3.7	Update Business Unit.....	4-5
4.3.7.1	Method Signature	4-5
4.3.7.2	Request Parameter.....	4-5
4.3.7.3	Response	4-5
4.3.7.4	Exception	4-5

5 Identity Audit Service

5.1	Overview	5-1
5.2	WSDL Access.....	5-1
5.3	API Reference	5-2
5.3.1	Get All Open Exceptions	5-2
5.3.1.1	Method Signature	5-2
5.3.1.2	Request Parameters	5-2
5.3.1.3	Response	5-2
5.3.1.4	Exception	5-2
5.3.2	Get All Audit Exceptions Based on the State.....	5-2
5.3.2.1	Method Signature	5-2
5.3.2.2	Request Parameters	5-2
5.3.2.3	Response	5-3
5.3.2.4	Exception	5-3
5.3.3	Mitigate Exception.....	5-3
5.3.3.1	Method Signature	5-3
5.3.3.2	Request Parameters	5-3
5.3.3.3	Response	5-3
5.3.3.4	Exception	5-3
5.3.4	Check Audit Exceptions for Users	5-4
5.3.4.1	Method Signature	5-4
5.3.4.2	Request Parameters	5-4
5.3.4.3	Response	5-4
5.3.4.4	Exception	5-4

6 Web Service Exception Handling

6.1	Overview	6-1
6.2	Error Codes and Error Messages.....	6-1

Preface

This guide describes the Oracle® Identity Analytics Web Services application programming interface (API). Web Services support interactions between third-party systems and the user, role, business unit, and identity audit information inside Oracle Identity Analytics.

Audience

This guide is intended for system integrators who need to integrate Oracle Identity Analytics with other IT systems.

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 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.

Related Documents

For more information, see the following documents in the Oracle Identity Analytics Release 11g R1 PS1 documentation set:

- *Oracle Identity Analytics Release Notes*
- *Oracle Identity Analytics Installation and Upgrade Guide*
- *Oracle Identity Analytics Administrator's Guide*
- *Oracle Identity Analytics User's Guide*
- *Oracle Identity Analytics System Integrator's Guide*
- *Oracle Identity Analytics Database Administrator's Guide*

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
monospace	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

Enabling Web Services

This chapter contains the following sections:

- [Section 1.1, "To Enable Web Services"](#)
- [Section 1.2, "To Disable Web Services"](#)

For security reasons, Web Services in Oracle Identity Analytics are disabled by default.

1.1 To Enable Web Services

1. Download and deploy the WSDL4J JAR file as described in the *Installation and Upgrade Guide*.
2. In a text editor, open the `web.xml` file located in the `RBACX_Home/WEB-INF` folder.
3. Search for the string `WEB SERVICES`.

There are three occurrences that you will update in the following steps.

- a. Add the context configuration parameter value as follows:

Change:

```
<context-param>
  <param-name>contextConfigLocation</param-name>
  <param-value>
    /WEB-INF/conf-context.xml
    ...
    /WEB-INF/siem-context.xml
  </param-value>
</context-param>
```

to:

```
<context-param>
  <param-name>contextConfigLocation</param-name>
  <param-value>
    /WEB-INF/conf-context.xml
    ...
    /WEB-INF/siem-context.xml
    classpath:org/codehaus/xfire/spring/xfire.xml
  </param-value>
</context-param>
```

- b. Uncomment the servlet definition as follows:

Change:

```
<!--  
    <servlet>  
        <servlet-name>ws</servlet-name>  
        <servlet-class>org.springframework.web.servlet.DispatcherServlet  
    </servlet-class>  
    </servlet>  
-->
```

to:

```
<servlet>  
    <servlet-name>ws</servlet-name>  
    <servlet-class>org.springframework.web.servlet.DispatcherServlet  
    </servlet-class>  
</servlet>
```

- c. Uncomment the servlet mapping as follows:

Change:

```
<!--  
    <servlet-mapping>  
        <servlet-name>ws</servlet-name>  
        <url-pattern>/ws/*</url-pattern>  
    </servlet-mapping>  
-->
```

to:

```
<servlet-mapping>  
    <servlet-name>ws</servlet-name>  
    <url-pattern>/ws/*</url-pattern>  
</servlet-mapping>
```

4. Save the `web.xml` file and restart the application server to register your changes.

1.2 To Disable Web Services

If Web Services are enabled and you want to disable them, follow these steps.

1. In a text editor, open the `web.xml` file located in the `RBACX_Home/WEB-INF` folder.
2. Search for the string `WEB SERVICES`.

There are three occurrences and in the following steps you must comment out two of them.

3. Comment out the servlet definition as follows:

Change:

```
<servlet>  
    <servlet-name>ws</servlet-name>  
    <servlet-class>org.springframework.web.servlet.DispatcherServlet  
    </servlet-class>  
</servlet>
```

to:

```
<!--  
  <servlet>  
    <servlet-name>ws</servlet-name>  
    <servlet-class>org.springframework.web.servlet.DispatcherServlet  
  </servlet-class>  
  </servlet>  
-->
```

4. Comment out the servlet mapping as follows:

Change:

```
<servlet-mapping>  
  <servlet-name>ws</servlet-name>  
  <url-pattern>/ws/*</url-pattern>  
</servlet-mapping>
```

to:

```
<!--  
<servlet-mapping>  
  <servlet-name>ws</servlet-name>  
  <url-pattern>/ws/*</url-pattern>  
</servlet-mapping>  
-->
```

5. Restart the application server to register your changes.

This chapter contains the following sections:

- [Section 2.1, "Overview"](#)
- [Section 2.2, "WSDL Access"](#)
- [Section 2.3, "API Reference"](#)

2.1 Overview

Oracle Identity Analytics Web Services (web services) have been designed to support interaction of third-party systems with the user information inside Oracle Identity Analytics 11gR1 software. A set of methods are provided to perform various operations inside Oracle Identity Analytics such as adding, deleting, or modifying users.

This chapter describes the web services used to get information from Oracle Identity Analytics regarding users in the Identity Warehouse.

Each web service has two parts: the request parameters and the response. For example, to obtain a list of roles allocated to a user, use the `username` request parameter. The `RoleVO` object is then returned with the information on the user's roles.

Note - The User Management Service has been deprecated and is superseded by the User Service. Implementations using methods from the User Management Service, however, can continue to use the older methods.

2.2 WSDL Access

WSDL stands for Web Services Description Language. It is an XML format for describing web services as a set of endpoints operating on messages, and contains information on how to access them. It specifies the location of each service and the operations (or methods) the service exposes.

Use the following to access User Service WSDL through a browser:

```
http://server-name:port-number/rbacx/ws/userService?wsdl
```

where *server-name* is the IP address or alias of the server on which Oracle Identity Analytics is running, and *port-number* is the port number of the application server that is listening to Oracle Identity Analytics calls.

Example:

```
http://localhost:8081/rbacx/ws/userService?wsdl
```

In the previous example, `localhost` is the server name, and `8081` is the port number of the application server.

2.3 API Reference

This section describes the web services used to get information from Oracle Identity Analytics regarding users.

2.3.1 Find All Users by Filter String

This web service finds all of the users filtered by a specific criteria. The request parameter is the String `filterString`, and an array of `UserVO[]` objects is returned with all of the users that satisfy the filter string.

2.3.1.1 Method Signature

```
public UserVO[] findUsers(String filterString) throws  
RbacxServiceException
```

2.3.1.2 Request Parameters

Name	Data Type	Description
<code>filterString</code>	<code>String</code>	The string to filter to find all of the users.

2.3.1.3 Response

Data Type	Description
<code>UserVO[]</code>	An array of <code>UserVO</code> objects is returned.

2.3.1.4 Exception

This method throws an `RbacxService` Exception that has an error code and an error message.

2.3.2 Find Roles For A User

This web service finds the roles allocated to a user. The request parameter is the string `username` whose roles need to be found, and an array of `RoleVO[]` objects is returned with the information on the user's roles.

2.3.2.1 Method Signature

```
public RoleVO[] getRolesForUser(String userName) throws  
RbacxServiceException
```

2.3.2.2 Request Parameters

Name	Data Type	Description
<code>userName</code>	<code>String</code>	The user name for which roles need to be checked.

2.3.2.3 Response

Data Type	Description
RoleVO[]	An array of RoleVO objects is returned.

2.3.2.4 Exception

This method throws an `RbacxServiceException` that has an error code and an error message.

2.3.3 Find Accounts for a User

This web service finds the accounts associated with a user. The request parameter is the `String` `username`, and an array of `AccountVO[]` objects is returned with a list of all the accounts that the user has.

2.3.3.1 Method Signature

```
public AccountVO[] getAccountsForUser(String username) throws
RbacxServiceException
```

2.3.3.2 Request Parameters

Name	Data Type	Description
<code>username</code>	<code>String</code>	The user name for which an account list needs to be generated.

2.3.3.3 Response

Data Type	Description
<code>AccountVO[]</code>	An array of <code>AccountVO</code> objects is returned.

2.3.3.4 Exception

This method throws an `RbacxServiceException` that has an error code and an error message.

2.3.4 Create a New User

This web service creates a new user. The request parameter is `UserVO` `user`, and a `Boolean` is returned indicating whether or not the user was created.

2.3.4.1 Method Signature

```
public boolean createUser(UserVO user) throws
RbacxServiceException
```

2.3.4.2 Request Parameter

Name	Data Type	Description
<code>user</code>	<code>UserVO</code>	The user object that needs to be created.

2.3.4.3 Response

Data Type	Description
Boolean	True if the user has been successfully created; false otherwise.

2.3.4.4 Exception

This method throws an `RbacxService` Exception that has an error code and an error message.

2.3.5 Create a New User if Rule-Based Role Assignment

This web service creates a new user if there is a rule-based role assignment. The request parameters are `UserVO` `user` and a Boolean, `isRuleBasedRoleAssignment`. A Boolean is returned indicating whether or not the user was created.

2.3.5.1 Method Signature

```
public boolean createUser(UserVO user, boolean
isRuleBasedRoleAssignment) throws RbacxServiceException
```

2.3.5.2 Request Parameters

Name	Data Type	Description
<code>user</code>	<code>UserVO</code>	The user object that needs to be created.
<code>isRuleBasedRoleAssignment</code>	Boolean	True if it is a rule-based role assignment; false otherwise.

2.3.5.3 Response

Data Type	Description
Boolean	True if the user has been successfully created; false otherwise.

2.3.5.4 Exception

This method throws an `RbacxService` Exception that has an error code and an error message.

2.3.6 Update User

This web service provides the capability to update an existing user. The request parameter is `UserVO` `user`, and a Boolean is returned indicating whether or not the user was successfully updated.

2.3.6.1 Method Signature

```
public boolean updateUser(UserVO user) throws
RbacxServiceException
```

2.3.6.2 Request Parameter

Name	Data Type	Description
user	UserVO	The user object that needs to be updated.

2.3.6.3 Response

Data Type	Description
Boolean	True if the user has been successfully updated; false otherwise.

2.3.6.4 Exception

This method throws an `RbacxService` Exception that has an error code and an error message.

2.3.7 Update User if Rule-Based Role Assignment

This web service updates a user if there is a rule-based role assignment. The request parameter is `UserVO` `user` and a Boolean, `isRuleBasedRoleAssignment`. A Boolean is returned indicating whether or not the user was updated.

2.3.7.1 Method Signature

```
public boolean updateUser(UserVO user, boolean
isRuleBasedRoleAssignment) throws RbacxServiceException
```

2.3.7.2 Request Parameters

Name	Data Type	Description
user	UserVO	The user object that needs to be created.
isRuleBasedRoleAssignment	Boolean	True if it is a rule-based role assignment; false otherwise.

2.3.7.3 Response

Data Type	Description
Boolean	True if the user has been successfully updated; false otherwise.

2.3.7.4 Exception

This method throws an `RbacxService` Exception that has an error code and an error message.

2.3.8 Disable Users

This web service disables existing users. The request parameter is a list of usernames, and a Boolean is returned indicating whether or not all of the users were disabled.

2.3.8.1 Method Signature

```
public boolean disableUsers(String[] userNames) throws
RbacxServiceException
```

2.3.8.2 Request Parameter

Name	Data Type	Description
usernames	String[]	A list of usernames that have to be disabled.

2.3.8.3 Response

Data Type	Description
Boolean	True if all of the users have been disabled; false otherwise.

2.3.8.4 Exception

This method throws an `RbacxService` Exception that has an error code and an error message.

2.3.9 Enable Users

This web service provides the capability to enable existing users. The request parameter is a list of usernames, and a Boolean is returned indicating whether or not all of the users were enabled.

2.3.9.1 Method Signature

```
public boolean enableUsers(String[] usernames) throws
RbacxServiceException
```

2.3.9.2 Request Parameter

Name	Data Type	Description
usernames	String[]	A list of usernames that need to be enabled.

2.3.9.3 Response

Data Type	Description
Boolean	True if all of the users are enabled; false otherwise.

2.3.9.4 Exception

This method throws an `RbacxService` Exception that has an error code and an error message.

2.3.10 Find Business Units for a User

This web service finds the business units assigned to a user. The request parameter is the string `username` whose business units need to be found. An array of `BusinessUnitVO[]` objects is returned with the information on the user's business units.

2.3.10.1 Method Signature

```
public BusinessUnitVO[] getBusinessUnitsForUser(String userName)
throws RbacxServiceException
```

2.3.10.2 Request Parameters

Name	Data Type	Description
userName	String	The user name for which business units need to be checked.

2.3.10.3 Response

Data Type	Description
BusinessUnitVO[]	An array of BusinessUnitVO objects is returned.

2.3.10.4 Exception

This method throws an `RbacxService` Exception that has an error code and an error message.

2.3.11 Remove All Roles From User

This web service removes all the roles from a user. The request parameter is the String `username`, and a Boolean flag is returned to indicate the result.

2.3.11.1 Method Signature

```
public boolean removeAllRolesfromUser(String userName) throws
RbacxServiceException
```

2.3.11.2 Request Parameters

Name	Data Type	Description
userName	String	The name of the user whose roles need to be removed.

2.3.11.3 Response

Data Type	Description
Boolean	True if all the roles have been successfully removed from the user; false otherwise.

2.3.11.4 Exception

This method throws an `RbacxService` Exception that has an error code and an error message.

This chapter contains the following sections:

- [Section 3.1, "Overview"](#)
- [Section 3.2, "WSDL Access"](#)
- [Section 3.3, "API Reference"](#)

3.1 Overview

This chapter describes the web services that are used to get roles and policies information from Oracle Identity Analytics.

Each web service has two parts: the request parameters and the response. Request parameters such as `businessUnitName` and `roleId` are used to request a role by business unit name or role name, and the `policyId` request parameter is used to request a policy by policy ID. The typical responses received are the `RoleVO` and `AccessPolicyVO` objects. The Role Value Object provides the resultant role object on the request criteria of `roleId`.

Note - The Role Management Service has been deprecated and is superseded by the Role Service. Implementations using methods from the Role Management Service, however, can continue to use the older methods.

3.2 WSDL Access

WSDL stands for Web Services Description Language. It is an XML format that describes web services as a set of endpoints operating on messages, and that contains information on how to access them. It specifies the location of each service and the operations (or methods) the service exposes.

Use the following to access the Role Service WSDL through a browser:

```
http://server-name:port-number/rbacx/ws/roleService?wsdl
```

where *server-name* is the IP address or alias of the server on which Oracle Identity Analytics is running, and *port-number* is the port number of the application server that is listening to Oracle Identity Analytics calls.

Example:

```
http://localhost:8081/rbacx/ws/roleService?wsdl
```

In the previous example, `localhost` is the server name, and `8081` is the port number of the application server.

3.3 API Reference

This section describes the web services used to get information from Oracle Identity Analytics regarding roles and policies.

3.3.1 Find Roles in Business Units

This web service provides the capability to search for roles in a particular business unit. The request parameters are the business unit name (`businessUnitName`), `traverse`, and `stopWhenFound`, and the response is a `RoleVO[]` array associated with the business unit.

3.3.1.1 Method Signature

```
public RoleVO[] findRolesInBusinessUnit(String businessUnitName,
boolean traverse, boolean stopWhenFound) throws
RbacxServiceException
```

3.3.1.2 Request Parameters

Name	Data Type	Description
<code>businessUnitName</code>	String	The name of the business unit to be queried for roles.
<code>traverse</code>	Boolean	If <code>true</code> , traverse the business unit tree to find roles.
<code>stopWhenFound</code>	Boolean	If <code>true</code> , stop traversing the business unit tree as soon as roles are found. If <code>false</code> , continue traversing the entire tree.

3.3.1.3 Response

Data Type	Description
<code>RoleVO[]</code>	An array of <code>RoleVO</code> objects containing role information is returned from the requested business unit. An empty array is returned if no roles are found.

3.3.1.4 Exception

This method throws an `RbacxService` Exception that has an error code and an error message.

3.3.2 Find Users in Role

This web service searches for all users who have access to a particular role. The request parameter is the `String roleName`, and the response is a `UserVO[]` array that contains all users with the role specified.

3.3.2.1 Method Signature

```
public UserVO[] findUsersInRole(String roleName) throws
RbacxServiceException
```

3.3.2.2 Request Parameters

Name	Data Type	Description
roleName	String	The name of the role to be queried for users.

3.3.2.3 Response

Data Type	Description
UserVO[]	An array of UserVO containing user information. An empty array is returned if no users are found.

3.3.2.4 Exception

This method throws an `RbacxService` Exception that has an error code and an error message.

3.3.3 Assign User to Role

This web service assigns a role to a user. The request parameters are `roleName`, `username`, `startDateVO`, and `endDateVO`, and a Boolean flag is returned to indicate the result of the assignment. Include the `updateConnectedSystems` parameter to control if the user assignment should be propagated to any connected systems. If the `updateConnectedSystems` parameter *is not* included, the user assignment is always propagated to connected systems.

3.3.3.1 Method Signatures

```
public boolean assignUserRole(String roleName, String userName,
DateVO startDateVO, DateVO endDateVO) throws
RbacxServiceException
```

```
public boolean assignUserRole(String roleName, String userName,
DateVO startDateVO, DateVO endDateVO, boolean
updateConnectedSystems) throws RbacxServiceException
```

3.3.3.2 Request Parameters

Name	Data Type	Description
roleName	String	The name of the role that is being assigned.
username	String	The name of the user to whom the role needs to be assigned.
startDateVO	DateVO	The date that the role association should start.
endDateVO	DateVO	The date that the role association should end.
updateConnectedSystems	Boolean	Set to true if assignment is to be propagated to connected systems; false otherwise.

3.3.3.3 Response

Data Type	Description
Boolean	True if the role is successfully assigned to the user; false otherwise.

3.3.3.4 Exception

This method throws an `RbacxService` Exception that has an error code and an error message.

3.3.4 Remove User From Role

This web service removes a user from a role. The request parameters are `roleName` and `username`, and a Boolean flag is returned to indicate the result of the removal. Include the `updateConnectedSystems` parameter to control if the removal should be propagated to any connected systems. If the `updateConnectedSystems` parameter *is not* included, the user removal is always propagated to connected systems.

3.3.4.1 Method Signatures

```
public boolean removeUserFromRole(String roleName, String
username) throws RbacxServiceException
```

```
public boolean removeUserFromRole(String roleName, String
username, boolean updateConnectedSystems) throws
RbacxServiceException
```

3.3.4.2 Request Parameters

Name	Data Type	Description
<code>roleName</code>	String	The name of the role that needs to be removed from the user.
<code>username</code>	String	The name of the user from whom the role should be removed.
<code>updateConnectedSystems</code>	Boolean	Set to true if the removal should be propagated to connected systems; false otherwise.

3.3.4.3 Response

Data Type	Description
Boolean	True if the user is successfully removed from the role; false otherwise.

3.3.4.4 Exception

This method throws an `RbacxService` Exception that has an error code and an error message.

3.3.5 Preview User to Role Assignment

This web service previews the results of a rule-based role assignment. The request parameters are `userVO[]` and `modifiedAccounts`, and the response is `RoleVO[]`.

3.3.5.1 Method Signature

```
public RoleVO[] userRoleAssignmentPreview(UserVO user,
AccountVO[] modifiedAccounts) throws RbacxServiceException
```

3.3.5.2 Request Parameters

Name	Data Type	Description
user	UserVO	An UserVO for whom the role assignment needs to be previewed.
modifiedAccounts	AccountVO[]	An array of AccountVO that is allocated to the user.

3.3.5.3 Response

Data Type	Description
RoleVO[]	An array of RoleVO containing role value objects is returned from the requested list of users. An empty array is returned if no roles are found.

3.3.5.4 Exception

This method throws an `RbacxServiceException` that has an error code and an error message.

3.3.6 Get Roles by Names

This web service returns an array of role value objects that match the specified array of role name strings.

3.3.6.1 Method Signature

```
public RoleVO[] getRolesByNames(String[] roleNames)
```

3.3.6.2 Request Parameters

Name	Data Type	Description
roleNames	String[]	The array of role names to retrieve.

3.3.6.3 Response

Data Type	Description
String[]	The array of role value objects matching the specified role names.

3.3.7 Update Roles

This web service updates a role using the specified Role Value Object (`RoleVO`). The `RoleVO` must have its role ID set to a valid role ID. If the `RoleVO` instance contains null values specified for any of its fields, those fields will be ignored and the current role's values will be retained. The specified flag `autoApprove` can be used to automatically approve the role update. Note that new owners must exist as Global Users or the update will fail.

3.3.7.1 Method Signature

```
public boolean updateRole(RoleVO role, boolean autoApprove)
```

3.3.7.2 Request Parameters

Name	Data Type	Description
role	RoleVO	The Role Value Object instance representing the role to be updated.
autoApprove	Boolean	Set to true if the role update should be automatically approved.

3.3.7.3 Response

Data Type	Description
Boolean	Returns true if the role has been updated.

Business Unit Service

This chapter contains the following sections:

- [Section 4.1, "Overview"](#)
- [Section 4.2, "WSDL Access"](#)
- [Section 4.3, "API Reference"](#)

4.1 Overview

This chapter describes the web services used to get business unit information from the Oracle Identity Analytics Identity Warehouse.

Each web service has two parts: request parameters and the response. Request parameters such as `businessUnitname` are used to request a business unit.

Note: The BusinessUnit Management Service has been deprecated and is superseded by the Business Unit Service. Implementations using methods from the BusinessUnit Management Service, however, can continue to use the older methods.

4.2 WSDL Access

WSDL stands for Web Services Description Language. It is an XML format that describes web services as a set of endpoints operating on messages, and that contains information on how to access them. It specifies the location of each service and the operations (or methods) the service exposes.

Use the following to access the Business Unit Service WSDL through a browser:

```
http://server-name:port-number/rbacx/ws/businessUnitService?wsdl
```

where *server-name* is the IP address or alias of the server on which Oracle Identity Analytics is running, and *port-number* is the port number of the application server that is listening to Oracle Identity Analytics calls.

Example:

```
http://localhost:8081/rbacx/ws/businessUnitService?wsdl
```

In the previous example, `localhost` is the server name, and `8081` is the port number of the application server.

4.3 API Reference

This section describes the web services used to get business unit information from the Oracle Identity Analytics Identity Warehouse.

4.3.1 Add Roles to a Business Unit

This web service adds roles to a business unit. The request parameters are the `businessUnitName` and the `roleNames` that should be added to the business unit. A Boolean flag is returned indicating if the roles were successfully added.

4.3.1.1 Method Signature

```
public boolean addRolesToBusinessUnit(String businessUnitName,
String[] roleNames) throws RbacxServiceException
```

4.3.1.2 Request Parameter

Name	Data Type	Description
<code>businessUnitName</code>	String	The name of the business unit to which one or more roles should be added.
<code>roleNames</code>	String[]	The list of roles that need to be added to the business unit.

4.3.1.3 Response

Data Type	Description
Boolean	True if the roles are successfully added to the business unit; false otherwise.

4.3.1.4 Exception

This method throws an `RbacxServiceException` that has an error code and an error message.

4.3.2 Remove Roles From a Business Unit

This web service removes roles from a business unit. The request parameters are `businessUnitName` and `roleNames`. A Boolean flag is returned indicating if the roles were successfully removed.

4.3.2.1 Method Signature

```
public boolean removeRolesFromBusinessUnit(String
businessUnitName, String[] roleNames) throws
RbacxServiceException
```

4.3.2.2 Request Parameters

Name	Data Type	Description
<code>businessUnitName</code>	String	The name of the business unit from which one or more roles should be removed.

Name	Data Type	Description
roleNames	String[]	The list of roles that need to be removed from a business unit.

4.3.2.3 Response

Data Type	Description
Boolean	True if the roles are successfully removed from the business unit; false otherwise.

4.3.2.4 Exception

This method throws an `RbacxService` Exception that has an error code and an error message.

4.3.3 Add Users to a Business Unit

This web service adds users to a business unit. The request parameters are the `businessUnitName` and `String[] userNames` that need to be associated. A Boolean flag is returned indicating the result of the businessunit-users association.

4.3.3.1 Method Signature

```
public boolean addUsersToBusinessUnit(String businessUnitName,
String[] userNames) throws RbacxServiceException
```

4.3.3.2 Request Parameters

Name	Data Type	Description
businessUnitName	String	The name of the business unit to be queried for users.
userNames	String[]	The user names that need to be associated with the business unit.

4.3.3.3 Response

Data Type	Description
Boolean	True if a businessunit-role membership is successfully established; false otherwise.

4.3.3.4 Exception

This method throws an `RbacxService` Exception that has an error code and an error message.

4.3.4 Remove Users From a Business Unit

This web service removes users from a business unit. The request parameters are the `businessUnitName` and `userNames`. A Boolean flag is returned indicating the results of the businessunit-users disassociation.

4.3.4.1 Method Signature

```
public boolean removeUsersFromBusinessUnit(String
businessUnitName, String[] usernames) throws
RbacxServiceException
```

4.3.4.2 Request Parameter

Name	Data Type	Description
businessUnitName	String	The name of the business unit to be queried for users.
userNames	String[]	The user names that need to be removed from the business unit.

4.3.4.3 Response

Data Type	Description
Boolean	True if the users are removed from the business unit; false otherwise.

4.3.4.4 Exception

This method throws an `RbacxServiceException` that has an error code and an error message.

4.3.5 Get Parent Business Structure for a Business Unit

This web service gets the parent for a particular business unit. The request parameter is the string `businessUnitName`, and an array of `BusinessUnitVO[]` is returned.

4.3.5.1 Method Signature

```
public BusinessUnitVO[] getParentBusinessUnits(String
businessUnitName) throws RbacxServiceException
```

4.3.5.2 Request Parameter

Name	Data Type	Description
businessUnitName	String	Name of the business unit for which the parent business unit name needs to be queried.

4.3.5.3 Response

Data Type	Description
BusinessUnitVO[]	An array of <code>BusinessUnit</code> value objects is returned.

4.3.5.4 Exception

This method throws an `RbacxServiceException` that has an error code and an error message.

4.3.6 Get Children of a Business Unit

This web service gets the children of a particular business unit. The request parameter is the string `businessUnitName`, and `BusinessUnitVO[]` is returned.

4.3.6.1 Method Signature

```
public BusinessUnitVO[] getChildrenBusinessUnits(String
businessUnitName) throws RbacxServiceException
```

4.3.6.2 Request Parameter

Name	Data Type	Description
<code>businessUnitName</code>	String	The name of the business unit for which the child business units need to be queried.

4.3.6.3 Response

Data Type	Description
<code>BusinessUnitVO[]</code>	The names of the child business units are returned in a value object.

4.3.6.4 Exception

This method throws an `RbacxServiceException` that has an error code and an error message.

4.3.7 Update Business Unit

This web service provides the capability to update an existing business unit. The request parameter is `businessUnitVO`, and a Boolean is returned indicating whether the business unit was successfully updated.

4.3.7.1 Method Signature

```
public boolean updateBusinessUnit(BusinessUnitVO businessUnitVO)
throws RbacxServiceException
```

4.3.7.2 Request Parameter

Name	Data Type	Description
<code>businessUnitVO</code>	<code>BusinessUnitVO</code>	The name of the business unit that needs to be updated.

4.3.7.3 Response

Data Type	Description
Boolean	True if the update was successful; false otherwise.

4.3.7.4 Exception

This method throws an `RbacxServiceException` that has an error code and an error message.

Identity Audit Service

This chapter contains the following sections:

- [Section 5.1, "Overview"](#)
- [Section 5.2, "WSDL Access"](#)
- [Section 5.3, "API Reference"](#)

5.1 Overview

This chapter describes the web services used to get information from Oracle Identity Analytics regarding audit violations. In particular, the Identity Audit web service can be used to prevent and report Segregation of Duties (SoD) violations. A *Segregation of Duties violation* is a violation whereby a user account, a user attribute, or a role has been assigned two entitlements that should not be held in combination.

Using the Identity Audit Web Service, SoD violations can be reported. In addition, this web service can be used to implement a preventative SoD whereby a user requesting a role from a third-party system can first be verified against Oracle Identity Analytics for any SoD.

Each web service has two parts: request parameters and the response. Request parameters such as `SodState` are used to request the corresponding Audit SoD, which returns a list of exceptions as a response.

5.2 WSDL Access

WSDL stands for Web Services Description Language. It is an XML format that describes web services as a set of endpoints operating on messages, and that contains information on how to access them. It specifies the location of each service and the operations (or methods) the service exposes.

Use the following to access the Identity Audit Web Service WSDL through a browser:

```
http://server-name:port-number/rbacx/ws/auditService?wsdl
```

where *server-name* is the IP address or alias of the server on which Oracle Identity Analytics is running, and *port-number* is the port number of the application server that is listening to Oracle Identity Analytics calls.

Example:

```
http://localhost:8081/rbacx/ws/auditService?wsdl
```

In the previous example, `localhost` is the server name, and `8081` is the port number of the application server.

5.3 API Reference

This section describes the web services used to get information from Oracle Identity Analytics regarding audit violations.

5.3.1 Get All Open Exceptions

This web service returns all open audit exceptions. An `AuditExceptionVO[]` array object containing the information on audit exceptions is returned.

5.3.1.1 Method Signature

```
AuditExceptionVO[] getAllOpenAuditExceptions() throws  
RbacxServiceException
```

5.3.1.2 Request Parameters

None

5.3.1.3 Response

Data Type	Description
<code>AuditExceptionVO[]</code>	An array of <code>AuditExceptionVO</code> value objects is returned.

5.3.1.4 Exception

This method throws an `RbacxService` Exception that has an error code and an error message.

5.3.2 Get All Audit Exceptions Based on the State

This web service returns all audit exceptions based on the state of the exception. The request parameter is the integer value indicating the state of the exception. An array of `AuditExceptionVO[]` objects containing audit exception information is returned.

5.3.2.1 Method Signature

```
public AuditExceptionVO[] getAuditExceptions(int state) throws  
RbacxServiceException
```

5.3.2.2 Request Parameters

Name	Data Type	Description
<code>state</code>	<code>int</code>	Integer value indicating the state of exception.

5.3.2.3 Response

Data Type	Description
<code>AuditExceptionVO[]</code>	An array of <code>AuditExceptionVO</code> value objects is returned.

5.3.2.4 Exception

This method throws an `RbacxServiceException` that has an error code and an error message.

5.3.3 Mitigate Exception

This web service marks the audit exception as "risk accepted." The request parameters are `exceptionID`, `comments`, and `expireDate`, and a Boolean flag is returned indicating the result of the action.

5.3.3.1 Method Signature

```
public boolean mitigateException(Long exceptionId, String
comments, Date expireDate) throws RbacxServiceException
```

5.3.3.2 Request Parameters

Name	Data Type	Description
<code>exceptionID</code>	<code>Long</code>	The audit exception that needs to be temporarily marked as risk accepted.
<code>comments</code>	<code>String</code>	Comments explaining why the exception is to be marked as risk accepted.
<code>expireDate</code>	<code>dateTime</code>	The date that the mitigation action expires.

5.3.3.3 Response

Data Type	Description
<code>Boolean</code>	True if the audit exception is successfully mitigated; false otherwise.

5.3.3.4 Exception

This method throws an `RbacxServiceException` that has an error code and an error message.

5.3.4 Check Audit Exceptions for Users

This web service checks for audit exceptions when changes are made to the accounts and roles of a user. The request parameters are `user`, `modifiedAccounts`, and `roleVOs`.

An `AuditExceptionVO[]` array is returned with the information on audit exceptions for that user.

5.3.4.1 Method Signature

```
public AuditExceptionVO[] checkForAuditExceptions(UserVO user,
AccountVO[] modifiedAccounts, RoleVO[] roleVOs) throws
RbacxServiceException
```

5.3.4.2 Request Parameters

Name	Data Type	Description
<code>user</code>	<code>UserVO</code>	User name for which the exception needs to be checked.
<code>modifiedAccounts</code>	<code>AccountVO[]</code>	An array of modified <code>AccountVO</code> objects.
<code>roleVOs</code>	<code>RoleVO[]</code>	An array of <code>RoleVO</code> objects.

5.3.4.3 Response

Data Type	Description
<code>AuditExceptionVO[]</code>	An array of <code>AuditException</code> value objects is returned.

5.3.4.4 Exception

This method throws an `RbacxServiceException` that has an error code and an error message.

Web Service Exception Handling

This chapter contains the following sections:

- [Section 6.1, "Overview"](#)
- [Section 6.2, "Error Codes and Error Messages"](#)

6.1 Overview

Exceptions are notifications that something has gone wrong with the normal flow of execution. A corresponding error message indicates the root cause of the exception.

In Oracle Identity Analytics 11gR1, web services are designed to return error codes and error messages for the following handled exceptions.

6.2 Error Codes and Error Messages

Error Code	Error Message
RBX0001	Generic Exception Occurred
RBX0002	Search Filter String cannot be null or empty
RBX1001	User Value Object cannot be null
RBX1002	Username/FirstName/LastName cannot be null or empty
RBX1003	Username already exists
RBX1004	User already exists
RBX1005	User(s) Not found
RBX2001	User already has the role assigned
RBX2002	Role not found
RBX2003	User doesn't have the role
RBX2004	Role name cannot be null or empty
RBX3001	BusinessUnit Name cannot be null or empty
RBX3002	BusinessUnit doesn't exist
RBX3003	Role already exist in BusinessUnit
RBX3004	BusinessUnit doesn't have the role(s) to be removed
RBX3005	User(s) already exist in BusinessUnit

RBX3006	BusinessUnit doesn't have the user(s) to be removed
RBX3007	Parent BusinessUnit doesn't exist
RBX3008	Child BusinessUnit doesn't exist
RBX3009	BusinessUnit Value Object cannot be null
RBX3010	Duplicate BusinessUnit Name
