Oracle® Fusion Middleware

Security Guide for Oracle Business Intelligence Enterprise Edition

11*g* Release 1 (11.1.1)

E10543-02

July 2010

ORACLE®

Oracle Fusion Middleware Security Guide for Oracle Business Intelligence Enterprise Edition, 11*g* Release 1 (11.1.1)

E10543-02

Contents

3 Configuring Oracle BI to use Oracle Internet Directory

B Understanding the Default Security Configuration

C Troubleshooting Security in Oracle Business Intelligence

Preface

Oracle Business Intelligence Enterprise Edition is a comprehensive set of enterprise business intelligence tools and infrastructure, including a scalable and efficient query and analysis server, an ad-hoc query and analysis tool, interactive dashboards, proactive intelligence and alerts, real-time predictive intelligence, and an enterprise reporting engine. Oracle Business Intelligence is designed to bring greater business visibility and insight to a wide variety of users.

The components of Oracle Business Intelligence share a common service-oriented architecture, data access services, analytic and calculation infrastructure, metadata management services, semantic business model, security model and user preferences, and administration tools. Oracle Business Intelligence provides scalability and performance with data-source specific optimized request generation, optimized data access, advanced calculation, intelligent caching services, and clustering.

Oracle Fusion Middleware Security Guide for Oracle Business Intelligence Enterprise Edition is part of the documentation set for Oracle Business Intelligence. This guide contains information about system administration tasks and includes topics on enabling and managing a secure environment.

Audience

This guide is intended for system administrators who are responsible for managing Oracle Business Intelligence security.

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible to all users, including users that are disabled. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Accessibility standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For more information, visit the Oracle Accessibility Program Web site at `http://www.oracle.com/accessibility/`.

Accessibility of Code Examples in Documentation

Screen readers might not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, some screen readers might not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation

This documentation might contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/support/contact.html or visit http://www.oracle.com/accessibility/support.html if you are hearing impaired.

Related Documents

For more information, see the following documents in the Oracle Business Intelligence Enterprise Edition 11*g* Release 1 (11.1.1) documentation set:

- *Oracle Fusion Middleware Release Notes*
- *Oracle Fusion Middleware Installation Guide for Oracle Business Intelligence*
- *Oracle Fusion Middleware Upgrade Guide for Oracle Business Intelligence*
- *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*
- *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*
- *Oracle Fusion Middleware User's Guide for Oracle Business Intelligence Enterprise Edition*
- *Oracle Fusion Middleware Developer's Guide for Oracle Business Intelligence Enterprise Edition*
- *Oracle Fusion Middleware Integrator's Guide for Oracle Business Intelligence Enterprise Edition*

System Requirements and Certification

Refer to the system requirements and certification documentation for information about hardware and software requirements, platforms, databases, and other information. Both of these documents are available on Oracle Technology Network (OTN).

The system requirements document covers information such as hardware and software requirements, minimum disk space and memory requirements, and required system libraries, packages, or patches:

http://www.oracle.com/technology/software/products/ias/files/fusion_requirements.htm

The certification document covers supported installation types, platforms, operating systems, databases, JDKs, and third-party products:

http://www.oracle.com/technology/software/products/ias/files/fusion_certification.html

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.
italic	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.

New Features in Oracle Business Intelligence Security

This preface describes changes in securing Oracle Business Intelligence Enterprise Edition 11*g* Release 1 (11.1.1). If you are upgrading to Oracle Business Intelligence from a previous release, read the following information carefully, because there are significant differences in features, tools, and procedures.

New Features

New features for securing Oracle Business Intelligence include:

- Integrated with Fusion Middleware Security Model
- Direct Access to LDAP Servers
- Simplified SSL Configuration
- Improved Model for Managing Administrative Privileges
- Repository Protection and Encryption

Integrated with Fusion Middleware Security Model

All components of Oracle Business Intelligence are fully integrated with Oracle Fusion Middleware security architecture. Oracle Business Intelligence authenticates users using an Oracle WebLogic Server authentication provider against user information held in an identity store. User and group information is no longer held within the repository (RPD) and the upgrade process migrates repository users and groups to become users and groups in Oracle WebLogic Server embedded directory server, which is the default identity store. Oracle Business Intelligence defines it's security policy in terms of Application Roles held in a policy store and stores credentials in a credential store. For more information, see Chapter 1, "Introduction to Security in Oracle Business Intelligence".

Direct Access to LDAP Servers

Oracle BI Delivers now accesses information about users, their groups, and email addresses directly from the configured identity store. In many cases this completely removes the need to extract this information from your corporate directory into a database and configure SA Subject System Area to enable all Delivers functionality. SA System Subject Area is still supported for backward compatibility. For more information, see Chapter 2, "Managing Security Using the Default Security Configuration".

Simplified SSL Configuration

Configuring Oracle Business Intelligence to use SSL for communication between processes in the middle-tier has been greatly simplified. In addition, a trusted system identity, rather than the Administrator's identity, is used to establish trust between Oracle Business Intelligence processes. This allows an administrative user to change their password without any impact on middle-tier communications. For more information, see Chapter 5, "SSL Configuration in Oracle Business Intelligence" and Chapter 2, "Managing Security Using the Default Security Configuration".

Improved Model for Managing Administrative Privileges

In 11*g* any named user can be granted administrative permissions if desired. This compares to 10*g* where there was a single user with administrative permissions who was named Administrator. For more information, see Appendix B, "Understanding the Default Security Configuration".

Repository Protection and Encryption

The repository is protected by a password and the same password is used to encrypt its contents. For more information, see Section B.6.2, "Planning to Upgrade a 10g Repository".

1

Introduction to Security in Oracle Business Intelligence

This chapter introduces the Oracle Business Intelligence security model, discusses the tools used to configure security, and provides a detailed roadmap for configuring security in Oracle Business Intelligence.

> **Note:** For a high-level roadmap for setting up security, see Section 1.1, "High-level Roadmap for Setting Up Security In Oracle Business Intelligence".

This chapter contains the following sections:

1.1 High-level Roadmap for Setting Up Security In Oracle Business Intelligence

To set up security in Oracle Business Intelligence, you must do the following:

1. Read the rest of this chapter 'Introduction to Security in Oracle Business Intelligence' to get an overview of security concepts, tools, and terminology.

2. Learn about the default set of Users, Groups, and Application Roles that are installed out-of-the-box by reading the summary in Section 2.2, "Working with the default Users, Groups, and Application Roles Installed Out-Of-The-Box".
3. Decide which Authentication Provider to use to authenticate users.
4. Set up the required Users and Groups.
5. Set up the required Application Roles.
6. Map each Group to an appropriate Application Role.
7. Fine tune the permissions that Users and Groups have in the Oracle BI repository (that is, the RPD file).
8. Fine tune the permissions that Users and Groups have in the Oracle BI Presentation Catalog.
9. If required, configure Single Sign-On (SSO).
10. If required, configure Secure Sockets Layer (SSL).

For a detailed list of setup steps, see Section 1.8, "Detailed List of Steps for Setting Up Security In Oracle Business Intelligence".

1.2 Overview of Security in Oracle Business Intelligence

Oracle Business Intelligence 11*g* is tightly integrated with the Oracle Fusion Middleware Security architecture and delegates core security functionality to components of that architecture. Specifically, any Oracle Business Intelligence installation makes use of the following types of security providers:

- An **authentication provider** that knows how to access information about the users and groups accessible to Oracle Business Intelligence and is responsible for authenticating users.
- A **policy store provider** that provides access to Application Roles and Application Policies, which forms a core part of the security policy and determines what users can and cannot see and do in Oracle Business Intelligence.
- A **credential store provider** that is responsible for storing and providing access to credentials required by Oracle Business Intelligence.

By default, an Oracle Business Intelligence installation is configured with an authentication provider that uses the Oracle WebLogic Server embedded LDAP server for user and group information. The Oracle Business Intelligence default policy store provider and credential store provider store Credentials, Application Roles and Application Policies in files in the domain.

After installing Oracle Business Intelligence you can reconfigure the domain to use alternative security providers, if desired. For example, you might want to reconfigure your installation to use an Oracle Internet Directory, Oracle Virtual Directory, Microsoft Active Directory, or another LDAP server for authentication. You might also decide to reconfigure your installation to use Oracle Internet Directory, rather than files, to store Credentials, Application Roles, and Application Policies.

1.3 About Authentication

Each Oracle Business Intelligence 11*g* installation has an associated Oracle WebLogic Server domain. Oracle Business Intelligence delegates user authentication to the first authentication provider configured for that domain.

The default authentication provider accesses user and group information stored in the LDAP server embedded in the Oracle Business Intelligence's Oracle WebLogic Server domain. The Oracle WebLogic Server Administration Console can be used to create and manage users and groups in the embedded LDAP server.

You might choose to configure an authentication provider for an alternative directory. In this case, Oracle WebLogic Server Administration Console enables you to view the users and groups in your directory. However, you need to continue to use the appropriate tools to make any modifications to the directory. For example, if you reconfigure Oracle Business Intelligence to use OID, you can view users and groups in Oracle WebLogic Server Administration Console but you must manage them in OID Console.

For more information about managing users and groups in the embedded LDAP server, see Chapter 2, "Managing Security Using the Default Security Configuration".

For more information about Oracle WebLogic Server domains and authentication providers, see *Oracle Fusion Middleware Securing Oracle WebLogic Server*.

1.4 About Authorization

After a user has been authenticated, the next critical aspect of security is ensuring that the user can do and see what they are authorized to do and see. Authorization for Oracle Business Intelligence release 11*g* is controlled by a security policy defined in terms of applications roles.

1.4.1 About Application Roles

Instead of defining the security policy in terms of users in groups in a directory server, Oracle Business Intelligence uses a role-based access control model. Security is defined in terms of Application Roles that are mapped to directory server groups and users. For example, the Application Roles BIAdministrator, BIConsumer, and BIAuthor are installed out-of-the-box.

Application Roles represent a functional role that a User has, which gives that User the privileges required to perform that role. For example, having the Sales Analyst Application Role might grant a User access to view, edit and create reports on a company's sales pipeline.

This indirection between Application Roles and directory server users and groups allows the administrator for Oracle Business Intelligence to define the Application Roles and policies without creating additional users or groups in the corporate LDAP server. Instead, the administrator defines Application Roles that meet the authorization requirements and maps those roles to pre-existing users and groups in the corporate LDAP server.

In addition, the indirection afforded by Application Roles allows the artifacts of a business intelligence system to be easily moved between development, test and production environments. No change to the security policy is needed and all that is required is to map the Application Roles to the users and groups available in the target environment.

The Figure 1-1 shows an example using the default set of Users, Groups, Application Roles.

Figure 1–1 Example Users, Groups, Application Roles, and Permissions

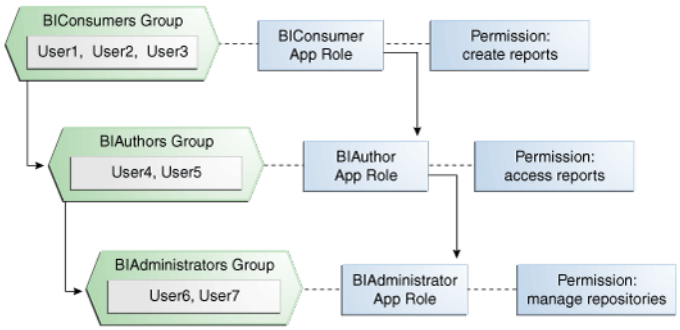


Figure 1-1 shows the following:

- The Group named 'BIConsumers' contains User1, User2, and User3. Users in the Group 'BIConsumers' are assigned the Application Role 'BIConsumer', which enables the users to view reports.
- The Group named 'BIAuthors' contains User4 and User5. Users in the Group 'BIAuthors' are assigned the Application Role 'BIAuthors', which enables the users to create reports.
- The Group named 'BIAdministrators' contains User6 and User7. Users in the Group 'BIAdministrators' are assigned the Application Role 'BIAdministrator', which enables the users to manage responsibilities.

1.4.2 About the Security Policy

In Oracle Business Intelligence release 11*g*, the security policy definition is split across the following components:

- Presentation Catalog – This defines the catalog objects and Oracle BI Presentation Services functionality that the Users with specific Application Roles can access. Access to functionality is defined in the Managing Privileges page in terms of Presentation Catalog privileges and access to presentation catalog objects is defined in the Permission dialog.
- Repository – This defines which Application Roles and users have access to which items of metadata within the repository. The Oracle BI Administration Tool is used to define this security policy.
- Policy Store – This defines which Oracle BI Server, BI Publisher, and Real Time Decisions functionality can be accessed by given users or users with given Application Roles. In the default Oracle Business Intelligence configuration, the policy store is managed using Oracle Enterprise Manager Fusion Middleware Control. For more information about the policy store, see *Oracle Fusion Middleware Security Guide*.

To find out about using these components, see Section 1.7, "Example: Looking at the Installed Users, Groups, and Application Roles".

1.5 About the Users, Groups, and Application Roles Installed Out-Of-The-Box

When you install Oracle Business Intelligence, you get a number of preconfigured Users, Groups, and Application Roles that you can use to deploy Oracle Business

Intelligence (for more information, see Section 2.2, "Working with the default Users, Groups, and Application Roles Installed Out-Of-The-Box").

1.6 What tools do I use to configure security in Oracle Business Intelligence?

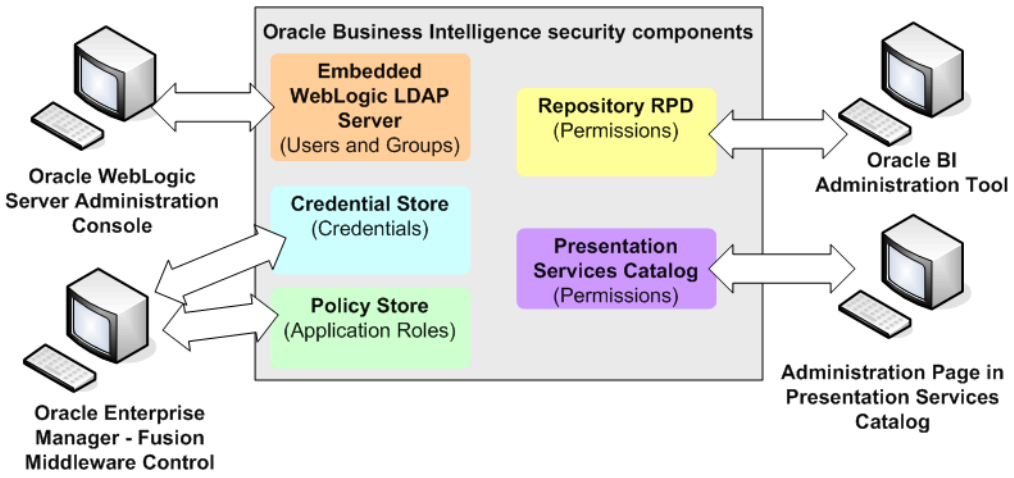
To configure security in Oracle Business Intelligence, you use the following tools:

- "Oracle WebLogic Server Administration Console"
- "Oracle Fusion Middleware Control"
- "Oracle BI Administration Tool"
- "Administration Page in Oracle BI Presentation Catalog"

> **Note:** To see an example of using the Oracle Business Intelligence tools to configure the installed Users, Groups, and Application Roles, see Section 2.3, "An Example Security Setup Using the Installed Groups and Application Roles".

The figure below summarizes the tools used to configure security in a default installation Oracle Business Intelligence using the embedded WebLogic LDAP Server.

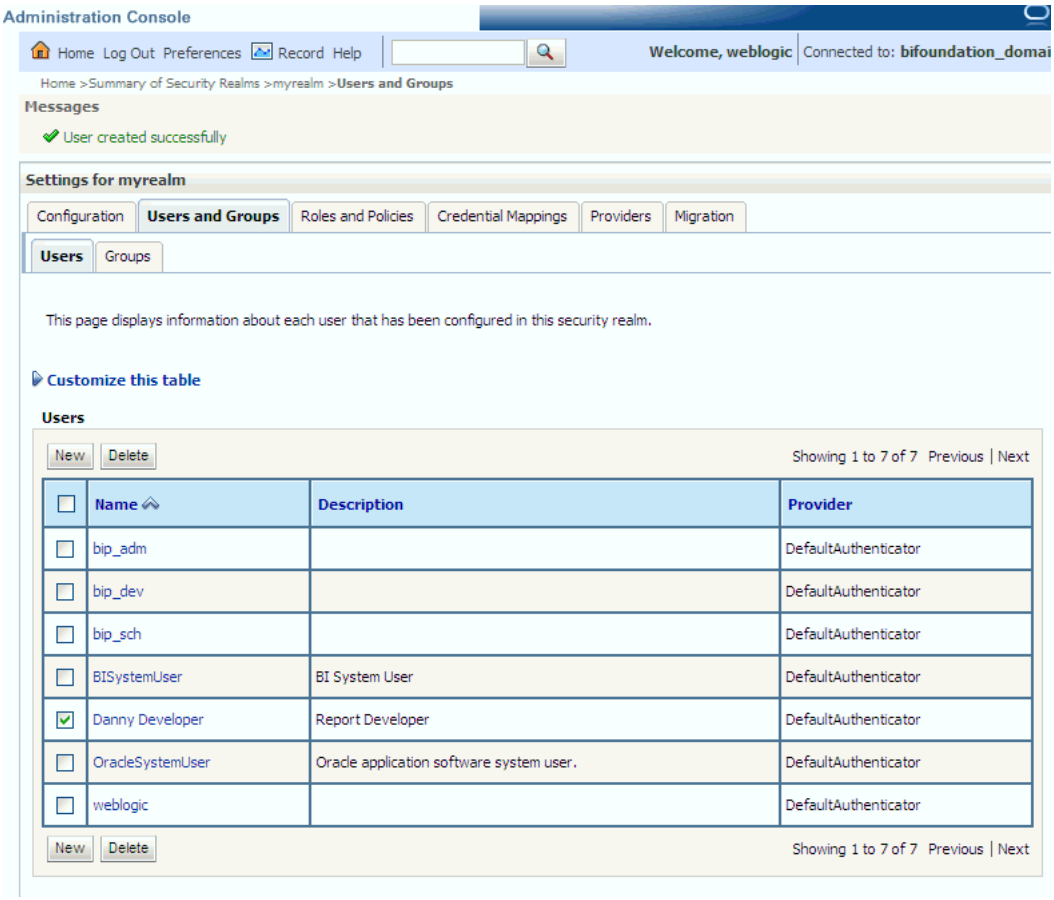
Figure 1–2 Summary of Tools for Configuring Security in a Default Installation



1.6.1 Oracle WebLogic Server Administration Console

You use Oracle WebLogic Server Administration Console to manage the embedded directory server that is used to authenticate Users and Groups.

The example screen shot below shows the **Users and Groups\Users** page in Oracle WebLogic Server Administration Console displaying a list of Users in Oracle Business Intelligence.

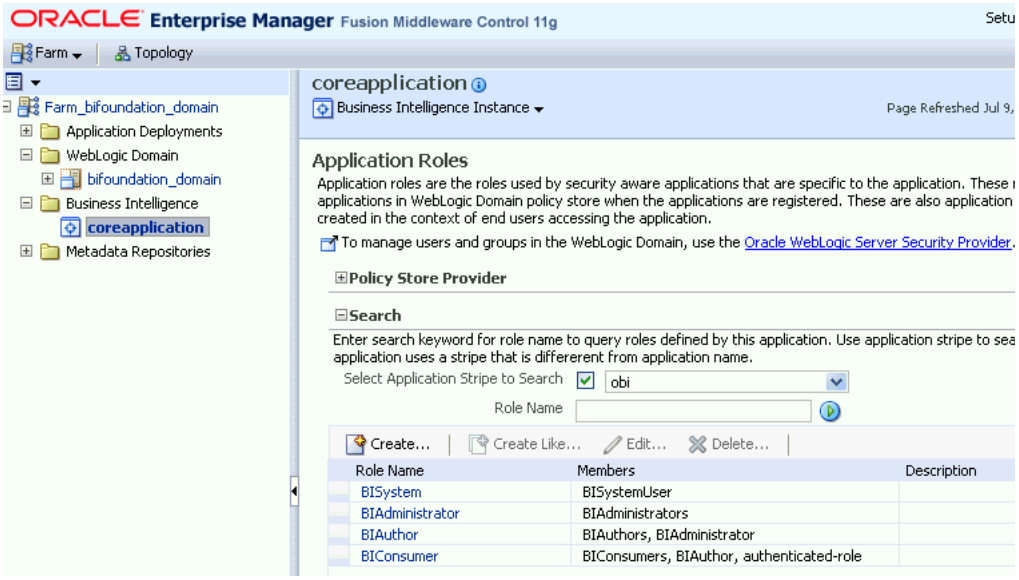


Note: If you use Oracle Internet Directory as the Authentication Provider instead of the default the embedded WebLogic LDAP Server, then you use OID Console to manage Users and Groups.

1.6.2 Oracle Fusion Middleware Control

You use Oracle Fusion Middleware Control to create and manage the Application Roles and Application Policies that control access to Oracle Business Intelligence resources.

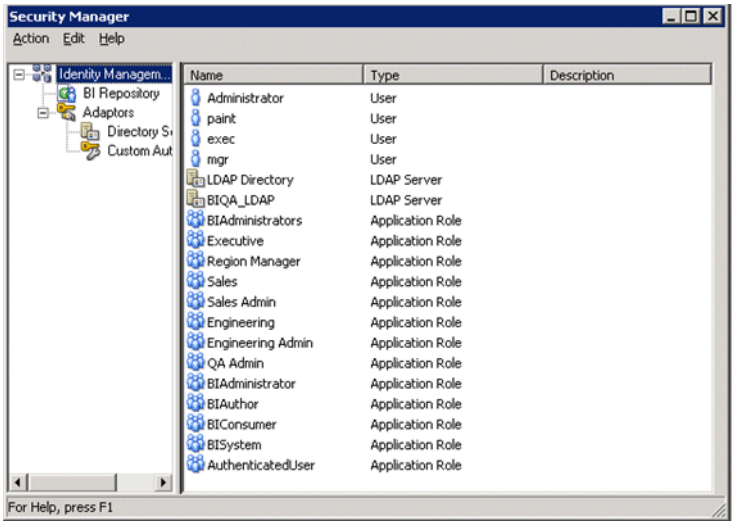
The example screen shot below shows the **Application Roles** page in Oracle Fusion Middleware Control displaying the default Application Roles named BIAdministrator, BIAuthor, and BIConsumer.



1.6.3 Oracle BI Administration Tool

You use the Oracle BI Administration Tool to configure privileges in the metadata repository (that is, the RPD file).

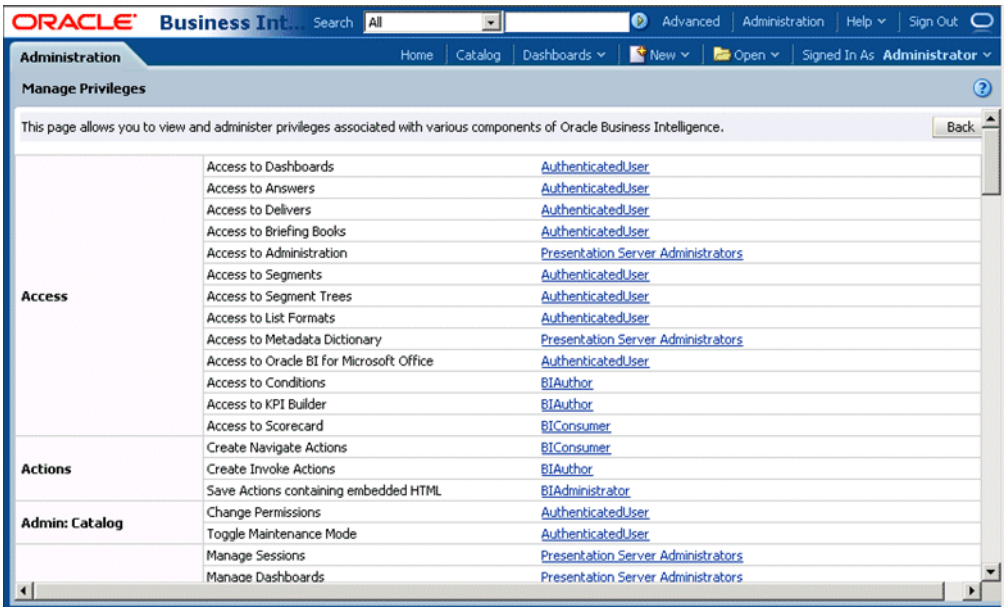
The screenshot below shows the Security Manager dialog, which enables you to manage Users and Application Roles.



1.6.4 Administration Page in Oracle BI Presentation Catalog

You use the Administration Page in Oracle BI Presentation Catalog to configure privileges for Users.

The screenshot below shows the Manage Privileges dialog, which enables you to manage privileges and associated Application Roles.



1.7 Example: Looking at the Installed Users, Groups, and Application Roles

This example takes a closer look at the installed Users, Groups, and Application Roles using the Oracle Business Intelligence tools. Follow the steps in this section to learn how to use the Oracle Business Intelligence tools to configure security options.

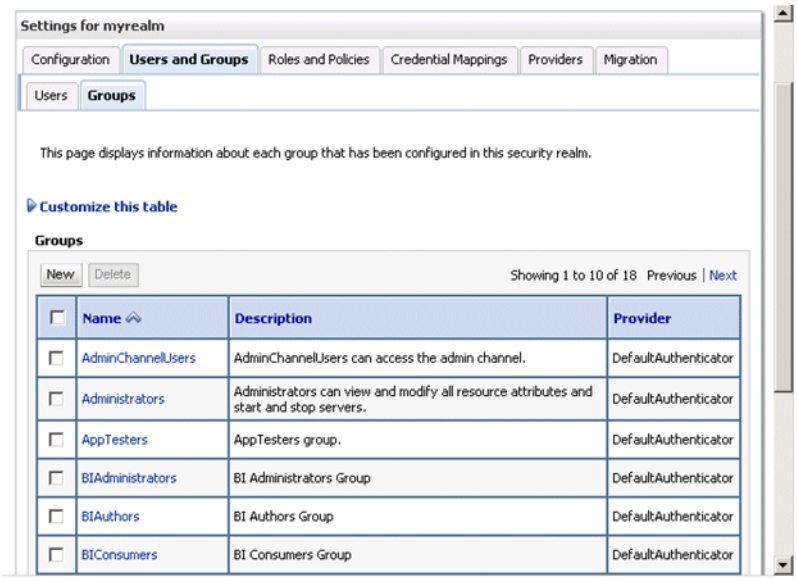
1.7.1 About Using Oracle WebLogic Server Administration Console

To display installed objects in Oracle WebLogic Server Administration Console:

1. Log in to Oracle WebLogic Server Administration Console.
2. In the Domain Structure tab at the left-hand side, select the **Security Realms** link.
3. In the list of Realms, select the realm that you are configuring.

   For example, myrealm.
4. Use the tabs and options on the Settings for <Realm name> dialog to configure Users and Groups.

   For example, display the Users and Groups tab to edit Users and Groups. In the example screenshot below, you can see the installed Groups named BIAdministrators, BIAuthors, and BIConsumers.

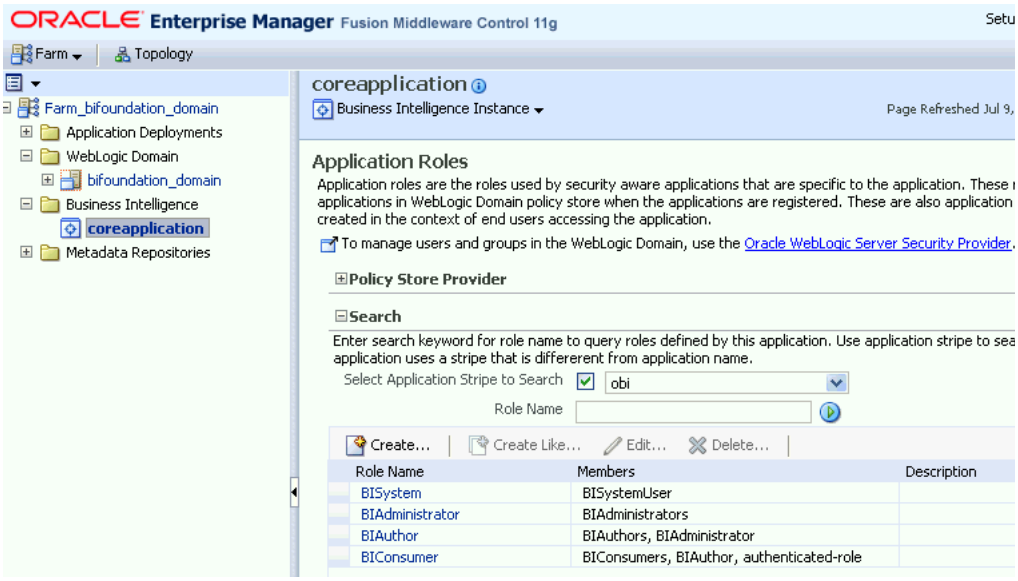


1.7.2 About using Oracle Enterprise Manager Fusion Middleware Control

To display installed objects in Oracle Enterprise Manager - Fusion Middleware Control:

1. Log in to Oracle Enterprise Manager - Fusion Middleware Control.
2. From the Home page, select the Business Intelligence link.
3. Select the coreapplication link.
4. Display the Security tab.
5. Select the **Configure and Manage Application Roles** link.

   In the example screenshot below, you can see the installed Application Roles BIAdministrator, BIAuthor, and BIConsumer.



1.7.3 About Using Oracle BI Administration Tool

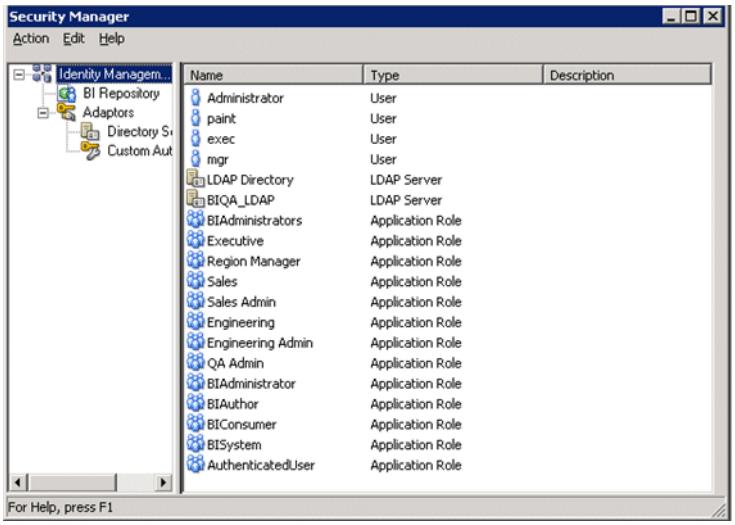
To display installed objects in Oracle BI Administration Tool:

1. Log in to BI Administration Tool.

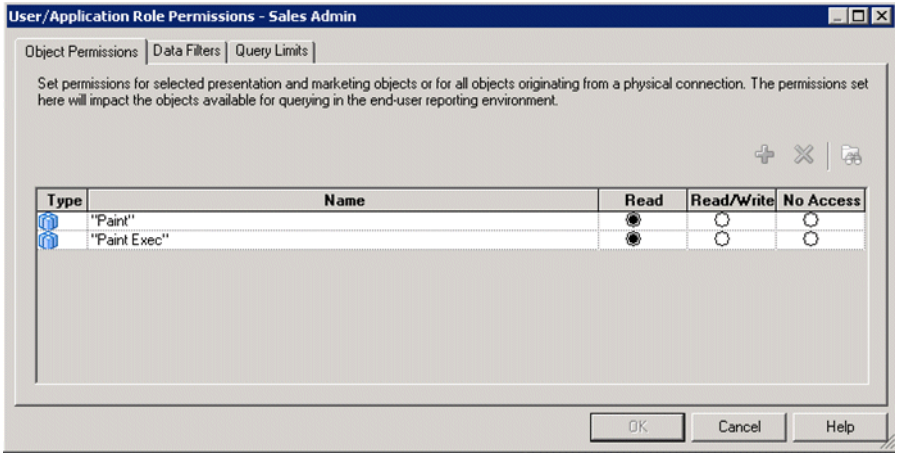
   Note: If you log in to BI Administration Tool in online mode, then you can view all users from the WebLogic Server. If you log in to BI Administration Tool in offline mode, then you can only view users that are stored in the catalog.

2. Choose Manage, then Identity to display the Security Manager dialog.

   In the example screenshot below you can see the installed Application Roles BIAdministrator, BIAuthor, and BIConsumer.

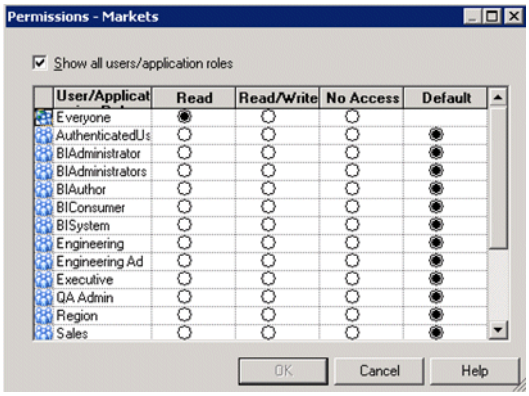


   If you double-click the Application Role named 'Sales Admin' to display the Application Role <*Name*> dialog, then click Permissions, you can use the Object Permissions tab to set Read and Write permissions for that Application Role on objects and folders in the catalog.



3. Close Security Manager.
4. In the Presentation pane, expand the Paint folder, then right-click Markets to display the Presentation Table <*Table name*> dialog.
5. Click Permissions to display the Permissions <*Table name*> dialog.

In the example screenshot below, you can see the installed Application Roles BIAdministrator, BIAuthor, and BIConsumer, and the radio buttons Read, Read/Write, No Access, and Default that are used to set the permissions for the Application Roles.

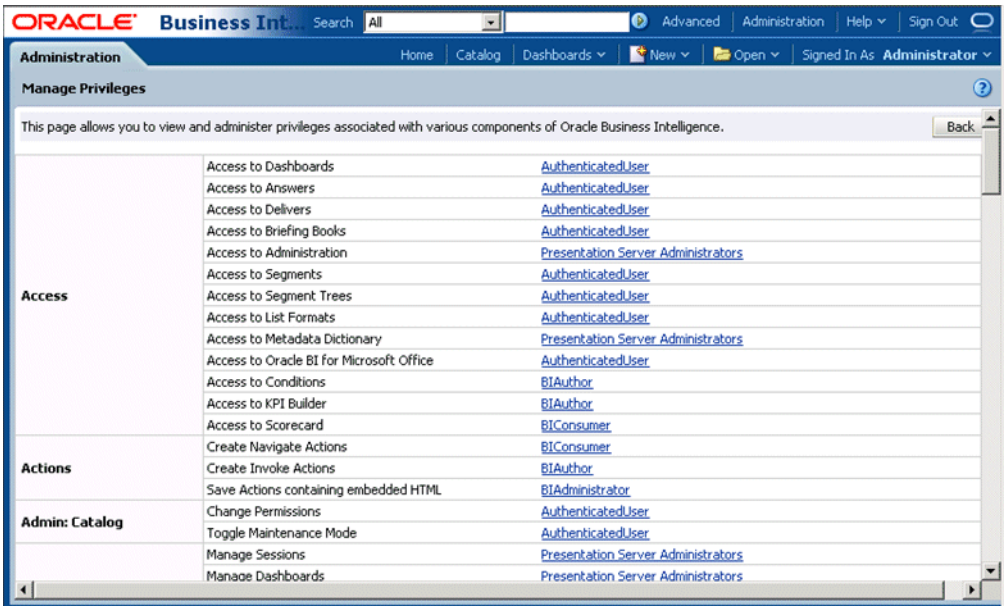


1.7.4 About Using Administration Page in Oracle BI Presentation Catalog

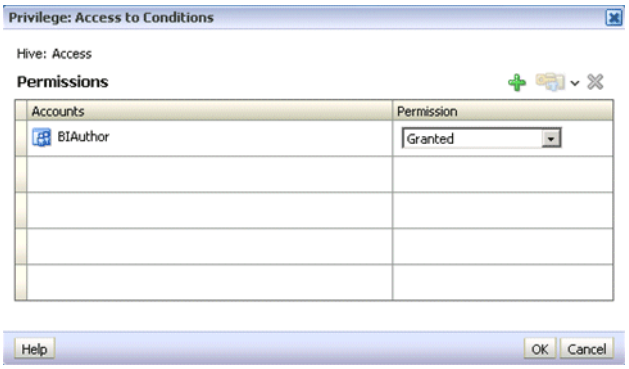
To display installed objects in Administration Page in Oracle BI Presentation Catalog:

1. Log in to BI EE with Administrator privileges.
2. Select the **Administration** link to display the Administration page.
3. Select the **Manage Privileges** link.

   In the example screenshot below, you can see the installed Application Roles BIAdministrator, BIAuthor, and BIConsumer listed against each of the privileges that they have been assigned.

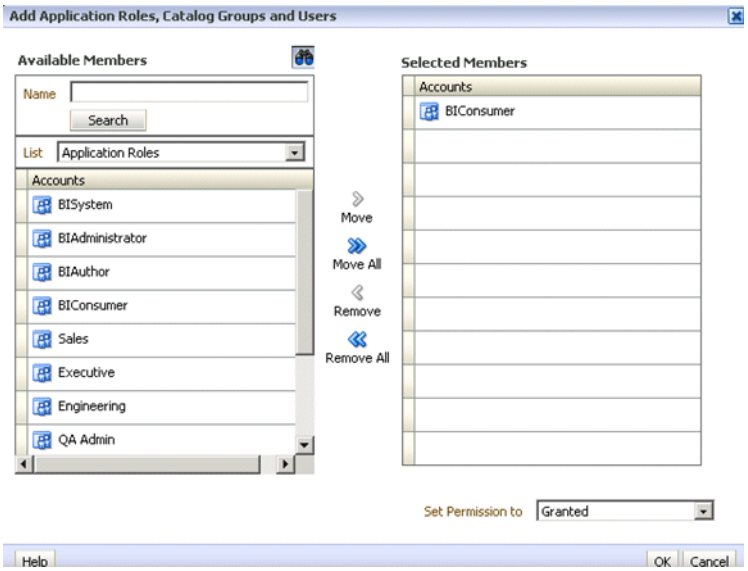


4. Select the BIAuthor link in the 'Access to Conditions' row, to display the Privilege <*Privilege name*> dialog.



5. Click the Add users/roles icon (+) to display the Add Application Roles, Catalog Groups, and Users dialog.

   In the example screenshot below you can see the installed Application Roles BIAdministrator, BIAuthor, and BIConsumer, which can be assigned to this privilege.



1.8 Detailed List of Steps for Setting Up Security In Oracle Business Intelligence

This section explains how to set up security in a new installation of Oracle Business Intelligence. Some tasks are mandatory, some are optional, and some are conditionally required depending on the configuration choices that you make. You might also refer to this section if you are maintaining an existing installation of Oracle Business Intelligence.

After you have installed Oracle Business Intelligence, you typically evaluate the product using the preconfigured Users, Groups, and Application Roles that are installed by default. Later, you typically create and develop your own Users, Groups, and Application Roles iteratively to meet your business requirements.

After you have installed Oracle Business Intelligence, Oracle recommends that you complete these tasks in the order listed below.

1. Read this chapter 'Introduction to Security in Oracle Business Intelligence' to get an overview of security concepts, tools, and terminology. In particular, you should familiarize yourself with the Oracle Business Intelligence components and tools for configuring security by reading Section 1.6, "What tools do I use to configure security in Oracle Business Intelligence?".
2. Learn about the default set of Users, Groups, and Application Roles that are installed out-of-the-box by reading the summary in Section 2.2, "Working with the default Users, Groups, and Application Roles Installed Out-Of-The-Box".
3. Decide which Authentication Provider to use to authenticate users, as follows:
   - If you want to use the default embedded WebLogic LDAP Server, then follow the tasks listed in Step 3 below.
   - If you want to reconfigure Oracle Business Intelligence to use a commercial authentication provider such as Oracle Internet Directory, then follow the tasks listed in Step 4 below.

     Tip: Oracle does not recommend using WebLogic Embedded LDAP Server in an environment with more than 1000 users. If you require a production environment with high-availability and scalability, then you should use a commercial directory server such as Oracle Internet Directory (OID) or a third-party directory server.

     For information about where to find the full list of supported Authentication Providers, see "System Requirements and Certification".

4. (Embedded WebLogic LDAP Server-specific) If you are using the default embedded WebLogic LDAP Server as the Authentication Provider, do the following:

   Tip: The simplest way to set up security is to create Users and map them to the default Groups (that is, BIConsumers, BIAuthors, and BIAdministrators) that are installed out-of-the-box. For detailed steps, see Section 2.4.1.1, "How to map a User to a Default Group".

   If you want to build a more complex security model using your own Groups, create new Groups and/or new Application Roles, then map your Users to the new Groups. For detailed steps, see Section 2.4.1.2, "How to create Your Own Groups and Application Roles".

   a. Set up the Users that you want to deploy as described in Section 2.4.3, "How to create a User in the Embedded WebLogic LDAP Server".

      For example, if you want to deploy business intelligence to 20 report consumers, you might create 20 Users.

   b. If you want to map Users to the default Groups that are installed out-of-the-box, (that is, BIConsumers, BIAuthors, and BIAdministrators), then follow the steps in Section 2.4.1.1, "How to map a User to a Default Group".

      For example, you might map a set of Users to the Group named BIConsumers, a set of Users to the Group named BIAuthors, and a set of Users to the Group named BIAdministrators.

   c. If you want to create new Groups, set up the Groups that you want to use as described in Section 2.4.4, "How to create a Group in the Embedded WebLogic LDAP Server".

For example, you might use the preconfigured Group named BIConsumers, or you might create your own Group with similar privileges.

d. Assign your Users to appropriate Groups, as described in Section 2.4.5, "How to add a User to a Group in the Embedded WebLogic LDAP Server".

For example, you might assign Users to the preconfigured Group named BIConsumers, or you might assign Users to a new Group that you have created.

5. (Oracle Internet Directory (OID) specific) If you are using OID as the Authentication Provider, do the following:

a. Configure OID as the Authentication Provider as described in Section 3.2.1, "How to Configure Oracle Internet Directory as an Authentication Store Provider".

b. (Optional) Configure OID as the Credential Store and Policy Store Provider as described in Section 3.3, "Configuring an Alternative Policy Store and Credentials Store".

c. Use your Authentication Provider tools (for example, OID Console) to create your Users and Groups as required.

6. Set up the Application Roles that you want to deploy as described in Section 2.5.2, "Creating Application Roles Using Fusion Middleware Control".

For example, you might use the default Application Roles named BIConsumer, BIAuthor, and BIAdministrator, or you might create your own Application Roles.

7. (Optional) If you do not want to use the preconfigured Application Policies, set up the Application Policies that you want to deploy as described in Section 2.5.3, "Creating Application Policies Using Fusion Middleware Control".

For example, you might use the preconfigured Application Policies that are used by the preconfigured Application Roles named BIConsumer, BIAuthor, and BIAdministrator, or you might create your own Application Policies.

8. Map each Group to an appropriate Application Role, as follows:

- If you are using the default Groups (that is, BIConsumers, BIAuthors, and BIAdministrators) that are installed with the default embedded WebLogic LDAP Server, then these Groups are mapped to an appropriate Application Role (that is, BIConsumer, BIAuthor, or BIAdministrator). No additional steps are required to map the default Groups to Application Roles.

  If you have created new Groups, you must map the new Groups to appropriate Application Roles as described in Section 2.5.2.3, "How to map a Group to an Application Role".

- If you are using a commercial Authenticator Provider such as Oracle Internet Directory, then you must map the Groups to appropriate Application Roles as described in Section 2.5.2.3, "How to map a Group to an Application Role".

9. If you want to fine tune the permissions that Users and Groups have in the Oracle BI repository (that is, the RPD file), use Oracle BI Administration Tool to update the permissions as described in Section 2.6, "Managing Metadata Repository Privileges".

For example, you might want to enable an Application Role called BISuperConsumer to create reports, so you use BI Administration Tool to change the 'Read' access to a subject area to 'Read/Write' access.

10. If you want to fine tune the permissions that Users and Groups have in the Oracle BI Presentation Catalog, use the Administration Page in Oracle BI Presentation Catalog to the permissions as described in Section 2.7, "Managing Oracle BI Presentation Catalog Privileges Using Application Roles".

    For example, you might want to prevent an Application Role called BISuperConsumer from viewing scorecards, so you use Administration Page in Presentation Catalog to change the Scorecard\View Scorecard privileges for BISuperConsumer from 'Granted' to 'Denied'.

11. If you want to deploy Single Sign-On, follow the steps in Chapter 4, "Enabling SSO Authentication".

    Note: If you do not want to deploy Oracle Business Intelligence in a SSO environment, then no additional configuration steps are required to deploy the default configuration.

12. If you want to deploy secure sockets layer (SSL), follow the steps in Chapter 5, "SSL Configuration in Oracle Business Intelligence".

    Oracle Business Intelligence is installed with SSL turned off. If you want to deploy Oracle Business Intelligence in an SSL environment, follow the steps in Chapter 5, "SSL Configuration in Oracle Business Intelligence".

    Note: If you do not want to deploy Oracle Business Intelligence in an SSL environment, then no additional configuration steps are required to deploy the default configuration.

1.9 Comparing the Oracle Business Intelligence 10*g* and 11*g* Security Models

The release 10*g* and release 11*g* security models differ in the following ways:

- Defining users and groups - In Oracle Business Intelligence release 10*g* users and groups could be defined within a repository file using Oracle BI Administration Tool. In Oracle Business Intelligence release 11*g* users and groups can no longer be defined within a repository. The Oracle Business Intelligence Enterprise Edition Upgrade Assistant migrates users and groups from a release 10*g* repository into the embedded LDAP server in a release 11*g* installation.
- Defining security policies – In Oracle Business Intelligence release 10*g* security policies in the web catalog and repository could be defined to reference groups within a directory. In Oracle Business Intelligence release 11*g* a level of indirection is introduced whereby security policies are defined in terms of Application Roles, which are in turn are mapped to users and groups in a directory. This indirection allows an Oracle Business Intelligence release 11*g* system to be deployed without changes to the corporate directory and eases movement of artifacts between development, test and production environments.
- Use of the Administrator user – In an Oracle Business Intelligence release 10*g* installation, a special user named Administrator has full administrative permissions and is also used to establish trust between processes within that installation. In Oracle Business Intelligence release 11g there is no special significance to the name Administrator and there can be one or more users who are authorized to undertake different sets of administrative functions. In Oracle Business Intelligence release 11*g* the identity used to establish trust between processes in an installation is configurable and independent.

- Repository encryption – in Oracle Business Intelligence release 10*g* certain sensitive elements within a repository are encrypted. In Oracle Business Intelligence release 11g the entire repository is encrypted using a key derived from a user supplied password.

> **Caution:** A release 11*g* repository can only be opened with the password. There is no mechanism for recovering a lost password.

The following aspects of the Oracle Business Intelligence release 10*g* security model remain in release 11*g*:

- Oracle BI Server Initialization Blocks – Oracle BI Server in release 11*g* continues to support the use of initialization blocks for authentication and authorization. In release 10*g* Oracle BI Server falls back to use initialization blocks if a matching user cannot be found in the repository. In release 11*g* Oracle Business Intelligence falls back to use initialization blocks if the user cannot be authenticated by the installation's configured authentication provider.
- Presentation Catalog Groups – Oracle Business Intelligence release 11*g* continues to support the definition of catalog groups within the Presentation Catalog. These groups are only visible within Oracle BI Presentation Services. Oracle recommends that Oracle BI Presentation Catalog groups be used for backward compatibility only and that Application Roles be used instead for new installations.
- SA System Subject Area – Oracle Business Intelligence release 11*g* supports the use of SA System Subject Area, in combination with Oracle BI Server initialization blocks, to access user, group and profile information stored in database tables.

For more information, see *Oracle Fusion Middleware Upgrade Guide for Oracle Business Intelligence Enterprise Edition*.

1.10 Terminology

The following terms are used throughout this guide:

Application Policy
Oracle Business Intelligence permissions are granted by its Application Roles. In the default security configuration, each role conveys a predefined set of permissions. An Application Policy is a collection of Java EE and JAAS policies that are applicable to a specific application. The Application Policy is the mechanism that defines the permissions each Application Role grants. Permission grants are managed in the Application Policy corresponding to an Application Role.

Application Role
Represents a role a user has when using Oracle Business Intelligence. Is also the container used by Oracle Business Intelligence to grant permissions to members of a role. Application roles are managed in the policy store provider.

Authentication
The process of verifying identity by confirming the credentials presented during logon.

Authentication Provider
A security provider used to access user and group information and is responsible for authenticating users. Oracle Business Intelligence default authentication provider is

Oracle WebLogic Server embedded directory server and is named DefaultAuthenticator.

Authorization
The process of granting an authenticated user access to a resource in accordance to their assigned privileges.

Catalog Groups
A catalog group is defined locally in Oracle BI Presentation Services and is used to grant privileges in the Oracle Business Intelligence user interface in addition to granting Oracle BI Presentation Catalog permissions.

Credential Store
An Oracle Business Intelligence credential store is a file used to securely store system credentials used by the software components. This file is automatically replicated across all machines in the installation.

Credential Store Provider
The credential store is used to store and manage credentials securely that are used internally between Oracle Business Intelligence components. For example, SSL certificates are stored here.

Encryption
A process that enables confidential communication by converting plaintext information (data) to unreadable text which can be read only with the use of a key. Secure Sockets Layer (SSL) enables secure communication over TCP/IP networks, such as web applications communicating through the Internet.

Globally Unique Identifier (GUID)
A GUID is typically a 32-character hexadecimal string that is system-generated to form a unique identifier for an object. In Oracle Business Intelligence a GUID is used to refer to individual users and groups.

Impersonation
Impersonation is a feature used by Oracle Business Intelligence components to establish a session on behalf of a user without employing the user's password. For example, impersonation is used when Oracle BI Scheduler executes an Agent.

Oracle WebLogic Server Domain
A logically related group of Oracle WebLogic Server resources that includes an instance known as the Administration Server. Domain resources are configured and managed in the Oracle WebLogic Server Administration Console. During installation an Oracle WebLogic Server domain is created and Oracle Business Intelligence is installed into that domain. For more information, see Section B.2.2, "Oracle WebLogic Server Domain".

Identity Store
An **identity store** contains user name, password, and group membership information. In Oracle Business Intelligence, the identity store is typically a directory server and is what an authentication provider accesses during the authentication process. For example, when a user name and password combination is entered at log in, the authentication provider searches the identity store to verify the credentials provided. Oracle Business Intelligence can be reconfigured to use alternative identity stores. For a complete list, see *System Requirements and Supported Platforms for Oracle Fusion Middleware 11gR1*. For more information, see System Requirements and Certification.

Policy Store Provider

The policy store is the repository of system and application-specific policies. It holds the mapping definitions between the default Oracle Business Intelligence Application Roles, permissions, users and groups all configured as part of installation. Oracle Business Intelligence permissions are granted by mapping users and groups from the identity store to Application Roles and permission grants located in the policy store.

Policy Store

Contains the definition of Application Roles, Application Policies, and the members mapped (users, groups, and applications roles) to Application Roles. The default policy store is a file that is automatically replicated across all machines in an Oracle Business Intelligence installation. A policy store can be file-based or LDAP-based.

Presentation Catalog Permissions

These rights grant Presentation Services object level access. They are stored in the Presentation Catalog and managed by Oracle BI Presentation Server.

Presentation Catalog Privileges

These rights grant access to Presentation Catalog features. They are stored in the Presentation Catalog and managed by Oracle BI Presentation Server. These privileges are either granted or denied.

Secure Sockets Layer (SSL)

Provides secure communication links. Depending upon the options selected, SSL might provide a combination of encryption, authentication, and repudiation. For HTTP based links the secured protocol is known as HTTPS.

Security Policy

The security policy defines the collective group of access rights to Oracle Business Intelligence resources that an individual user or a particular Application Role have been granted. Where the access rights are controlled is determined by which Oracle Business Intelligence component is responsible for managing the resource being requested. A user's security policy is the combination of permission and privilege grants governed by the following elements:

- Presentation Catalog: defines which catalog objects and Oracle BI Presentation Services functionality can be accessed by users. Access to this functionality is managed in Oracle Business Intelligence user interface. These permissions and privileges can be granted to individual users or by membership in corresponding Application Roles.
- Repository File: defines access to the specified metadata within the repository file. Access to this functionality is managed in Oracle BI Administration Tool. These permissions and privileges can be granted to individual users or by membership in corresponding Application Roles.
- Policy Store: defines which Oracle Business Intelligence, Oracle BI Publisher, and Real Time Decisions functionality can be accessed. Access to this functionality is managed in Oracle Enterprise Manager Fusion Middleware Control. These permissions and privileges can be granted to individual users or by membership in corresponding Application Roles.

Security Realm

During installation an Oracle WebLogic Server domain is created and Oracle Business Intelligence is installed into that domain. Security for an Oracle WebLogic Server domain is managed in its **security realm**. A security realm acts as a scoping mechanism. Each security realm consists of a set of configured security providers, users, groups, security roles, and security policies. Only one security realm can be

active for the domain. Oracle Business Intelligence authentication is performed by the authentication provider configured for the default security realm for the WebLogic Server domain in which it is installed. Oracle WebLogic Server Administration Console is the administration tool for managing an Oracle WebLogic Server domain.

Single Sign-On

A method of authorization enabling a user to authenticate once and gain access to multiple software application during a single browser session.

Users and Groups

A **user** is an entity that can be authenticated. A user can be a person, such as an application user, or a software entity, such as a client application. Every user is given a unique identifier within in the identity store.

Groups are organized collections of users that have something in common. A group is a static identifier that is assigned by a system administrator. Users organized into groups facilitate efficient security management. There are two types of groups: an LDAP group and a catalog group. A *catalog group* is used to support the existing user base in Presentation Services to grant privileges in the Oracle Business Intelligence user interface. Using catalog groups is not considered a best practice and is available for backward compatibility in upgraded systems.

2

Managing Security Using the Default Security Configuration

This chapter explains how to deploy Oracle Business Intelligence using the default embedded WebLogic LDAP Server.

> **Note:** For a detailed list of security setup steps, see Section 1.8, "Detailed List of Steps for Setting Up Security In Oracle Business Intelligence".

By deploying the default embedded WebLogic LDAP Server, you can use the installed Users, Groups, and Application Roles that are installed and preconfigured out-of-the-box, as well as developing your own Users, Groups, and Application Roles. For more information about the installed Users, Groups, and Application Roles, see Section 2.2, "Working with the default Users, Groups, and Application Roles Installed Out-Of-The-Box".

This chapter contains the following sections:

- Section 2.1, "Common Tasks for Managing Security Using the Default Security Configuration"
- Section 2.2, "Working with the default Users, Groups, and Application Roles Installed Out-Of-The-Box"
- Section 2.3, "An Example Security Setup Using the Installed Groups and Application Roles"
- Section 2.4, "Creating Users and Groups in the Embedded WebLogic LDAP Server"
- Section 2.5, "Managing Application Roles and Application Policies Using Fusion Middleware Control"
- Section 2.6, "Managing Metadata Repository Privileges"
- Section 2.7, "Managing Oracle BI Presentation Catalog Privileges Using Application Roles"

2.1 Common Tasks for Managing Security Using the Default Security Configuration

Table 2–1 lists common authentication configuration tasks and provides links for obtaining more information.

Table 2–1 Task Map: Configuring Authentication for Oracle Business Intelligence

Task	Description	For Information
Add Users and Groups in Oracle WebLogic Server embedded directory server.	Add Users and Groups to the identity store, and map the Users to the appropriate Groups.	Section 2.4.3, "How to create a User in the Embedded WebLogic LDAP Server" Section 2.4.4, "How to create a Group in the Embedded WebLogic LDAP Server" Section 2.4.5, "How to add a User to a Group in the Embedded WebLogic LDAP Server"
Create Application Roles.	Create a new Application Role and define its permission grants in the corresponding Application Policy.	Section 2.5.2, "Creating Application Roles Using Fusion Middleware Control"
Modify the permissions for an Application Role.	Modify the permissions granted by an Application Role.	Section 2.5.4, "Modifying Application Roles Using Oracle Fusion Middleware Control"
Map each Group to an Application Role.	Map each Group to an appropriate Application Role to grant its permissions to group members.	Section 2.5.4, "Modifying Application Roles Using Oracle Fusion Middleware Control"
Modify permissions for a Repository Subject Area or Folder.	Modify permissions for a Repository Subject Area or Folder.	Section 2.6, "Managing Metadata Repository Privileges"
Modify Presentation Services catalog permission grants.	Modify the Oracle BI Presentation Catalog privilege grants for an Application Role.	Section 2.7, "Managing Oracle BI Presentation Catalog Privileges Using Application Roles"

2.2 Working with the default Users, Groups, and Application Roles Installed Out-Of-The-Box

When you install Oracle Business Intelligence, you get a number of preconfigured Users, Groups, and Application Roles that you can use to deploy Oracle Business Intelligence. For example, you get a User that is assigned to a BIAdministrators Group (with a name that is user-specified at installation time, for example Weblogic), a Group named 'BIAdministrators', and an Application Role named 'BIAdministrator'. The installed Users, Groups, and Application Roles are preconfigured to work together. For example, the installed BIConsumers Group is assigned to the BIConsumer Application Role. For a detailed description of the default security configuration, refer to Appendix B, "Understanding the Default Security Configuration".

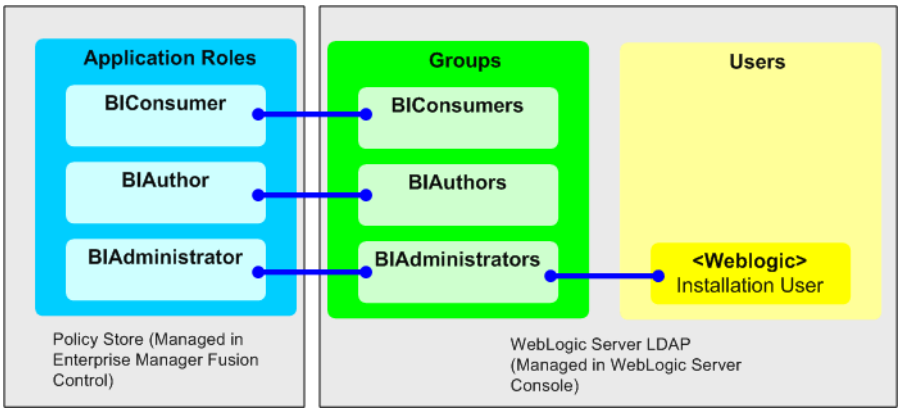
Caution: Oracle recommends that you do not modify the default Users, Groups, or Application Roles that are installed out-of-the-box, unless explicitly advised to do so by Oracle Support. Oracle recommends that you only modify copies that you have made of the installed Groups and Application Roles.

The installed Application Roles are preconfigured with appropriate permissions and privileges to enable them to work with the installed Oracle BI Presentation Catalog, BI

Repository (RPD), and Policy Store. For example, the Application Role named BIAuthors is preconfigured with permissions and privileges that are required to create dashboards, reports, actions, and so on.

The figure below shows the Users, Groups, and Application Roles that are installed and preconfigured.

Figure 2–1 Installed Application Roles, Groups, and Users



The following Groups are available:

- BIConsumers (preconfigured with the BIConsumer Application Role).
- BIAuthors (preconfigured with the BIAuthors Application Role).
- BIAdministrators (preconfigured with the BIAdministrators Application Role).

The User that is specified at installation time (for example, Weblogic), is automatically assigned to the WebLogic Administrators Group and the Group named 'BIAdministrators'. This User has permissions to log in to the Oracle Business Intelligence tools to create and administer other Users.

Note: Groups are organized hierarchically, and inherit privileges from parent Groups. In other words, the BIAdministrator Group automatically inherits privileges from the Groups BIAuthors and BIConsumer. Oracle recommends that you do not change this hierarchy.

You can use the installed Groups and Application Roles to deploy security, and if required you can develop your own Groups and Application Roles to meet your business needs. For example:

- If you want to enable an employee called Fred to create dashboards and reports, you might create a new User called 'Fred' and assign 'Fred' to the default BIAuthors Group.
- If you want to enable user Fred to perform BIAuthors and BIAdministrator type duties, you might create a new Application Role called 'BIManager', which has both BIAuthors privileges and BIAdministrators privileges
- If you want user Fred to be a Sales dashboard author, you might create an Application Role called 'Sales Dashboard Author' that has permissions to see Sales subject areas in the repository and edit Sales dashboards.

For detailed information about the installed Users, Groups, and Application Roles, see Appendix B, "Understanding the Default Security Configuration".

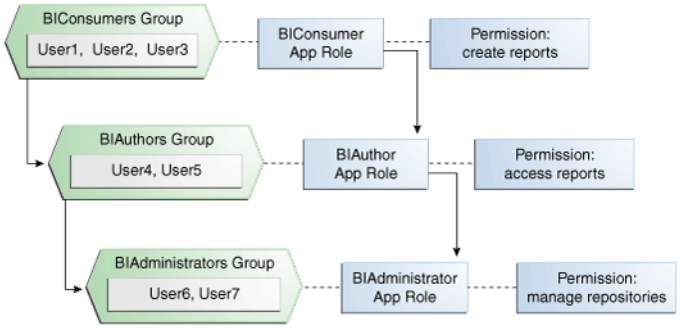
2.3 An Example Security Setup Using the Installed Groups and Application Roles

This example uses a small set of Users, Groups, and Application Roles to illustrate how you set up a security policy using the default Groups and Application Roles that are installed and pre-configured out-of-the-box. In this example, you want to implement the following:

- Three users named User1, User2, and User3, who need to view business intelligence reports.
- Two users named User4 and User5, who need to create business intelligence reports.
- Two users named User6 and User7, who administer Oracle Business Intelligence.

The figure below shows the Users, Groups, and Application Roles that you would deploy to implement this security model.

Figure 2–2 Example Groups, Application Roles, and Users



The example above shows the following:

- The Group named 'BIConsumers' contains User1, User2, and User3. Users in the Group 'BIConsumers' are assigned the Application Role named 'BIConsumer', which enables the users to view reports.
- The Group named 'BIAuthors' contains User4 and User5. Users in the Group 'BIAuthors' are assigned the Application Role named 'BIAuthors', which enables the users to create reports.
- The Group named 'BIAdministrators' contains User6 and User7. Users in the Group 'BIAdministrators' are assigned the Application Role named 'BIAdministrator', which enables the users to manage responsibilities.

To implement this example security model, you would do the following:

1. Create seven users named User1 to User 7, as described in Section 2.4.3, "How to create a User in the Embedded WebLogic LDAP Server".
2. Assign the users to the installed and preconfigured Groups, as follows:
   - Assign User1, User2, and User3 to the preconfigured Group named BIConsumers.
   - Assign User4 and User5 to the preconfigured Group named BIAuthors.
   - Assign User6 and User7 to the preconfigured Group named BIAdministrators.

   For more information, see in Section 2.4.5, "How to add a User to a Group in the Embedded WebLogic LDAP Server".

2.4 Creating Users and Groups in the Embedded WebLogic LDAP Server

This section explains how to create and manage Users and Groups in the Embedded WebLogic LDAP Server, and contains the following topics:

2.4.1 Overview to Setting Up Users, Groups, and Application Roles

This section summarizes what you must do to set up Users, Groups, and Application Roles.

The simplest way to set up security is to create Users and assign them to the default Groups (that is, BIConsumers, BIAuthors, or BIAdministrators). For example, you might create a user called Fred and assign Fred to the default Group named BIAuthors. The Group BIAuthors is pre-configured with the privileges it requires to access the other BI components, such as the metadata repository (RPD) and Presentation Catalog. For detailed steps, see Section 2.4.1.1, "How to map a User to a Default Group".

If the default Groups (that is, BIConsumers, BIAuthors, or BIAdministrators) do not meet your business requirements, you can extend the default security model by creating your own Groups and Application Roles. For example, you might want to create a user called Jim and assign Jim to a new Group called BIMarketingGroup that is mapped to a new Application Role named BIMarketingRole. For detailed steps, see Section 2.4.1.2, "How to create Your Own Groups and Application Roles".

2.4.1.1 How to map a User to a Default Group

To create a new User and assign that User to one of the installed Groups, do the following:

1. Launch WebLogic Administration Console as described in Section 2.4.2, "How to Launch Oracle WebLogic Server Administration Console".
2. Create a new User as described in Section 2.4.3, "How to create a User in the Embedded WebLogic LDAP Server".
3. Assign the new User to one of the installed Groups (that is, BIConsumers, BIAuthors, or BIAdministrators) as described in Section 2.4.5, "How to add a User to a Group in the Embedded WebLogic LDAP Server".

2.4.1.2 How to create Your Own Groups and Application Roles

If you want to create a new User and assign that User to a new Group that you have created, do the following:

1. Launch WebLogic Administration Console as described in Section 2.4.2, "How to Launch Oracle WebLogic Server Administration Console".

2. Create a new User as described in Section 2.4.3, "How to create a User in the Embedded WebLogic LDAP Server".
3. Create a new Group as described in Section 2.4.4, "How to create a Group in the Embedded WebLogic LDAP Server".
4. Create a new Application Role and assign it to the new Group as described in Section 2.5.2.2, "How to Create an Application Role".

   If you simply want to map a Group to an Application Role, follow the steps in Section 2.5.2.3, "How to map a Group to an Application Role".
5. Edit the repository (RPD file) and set up the privileges for the new Application Role as described in Section 2.6.2, "How to Set Repository Privileges for an Application Role".
6. Edit the Presentation Catalog and set up the privileges for the new User and Group as described in Section 2.7.3, "How to Set Catalog Privileges for an Application Role".

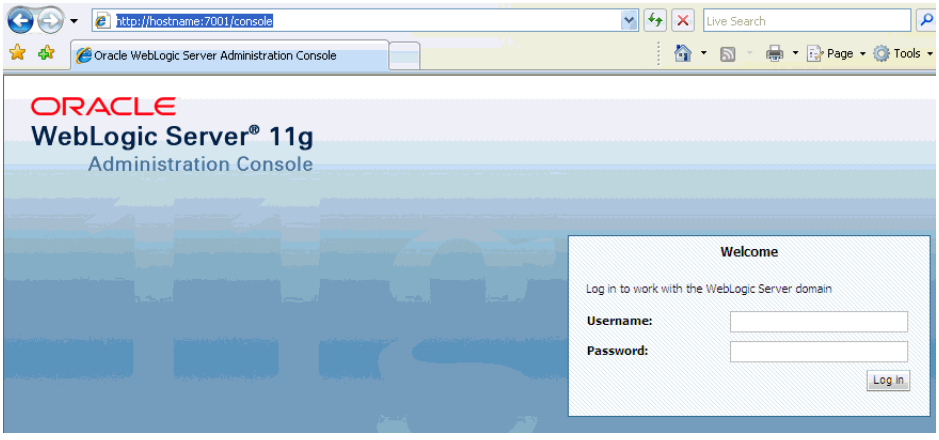
2.4.2 How to Launch Oracle WebLogic Server Administration Console

Oracle WebLogic Server is automatically installed and serves as the default administration server. The Administration Console is browser-based and is used to manage the embedded directory server that is configured as the default authenticator. It is launched by entering its URL into a web browser. The default URL takes the following form: http://*hostname:port_number*/console. The port number is the number of the administration server. By default, the port number is 7001. For more information about using the Administration Console, see *Oracle Fusion Middleware Oracle WebLogic Server Administration Console Online Help*.

To launch the Oracle WebLogic Server Administration Console:

1. Log in to Oracle WebLogic Serverr by entering its URL into a Web browser.

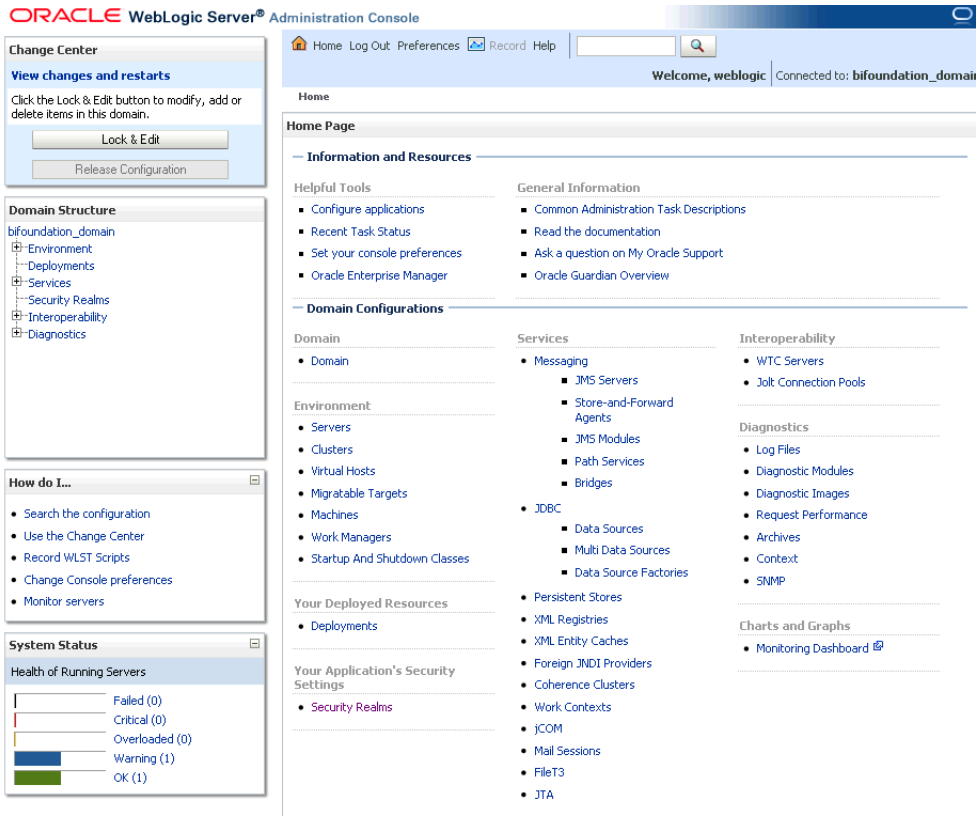
   For example, http://*hostname:7001*/console.



2. Log in using the Oracle Business Intelligencer administrative user and password and click **Login**.

   The user name and password combination is the one you supplied during the installation of Oracle Business Intelligence. If these values have been changed, then use the current administrative user name and password combination.

   The Administration Console displays.



2.4.3 How to create a User in the Embedded WebLogic LDAP Server

You typically create a separate User for each business user in your Oracle Business Intelligence environment. For example, you might plan to deploy 30 report consumers, three report authors, and 1 administrator. In this case, you would use Oracle WebLogic Server Administration Console to create 34 Users, which you would then assign to appropriate Groups (for example, you might use the preconfigured Groups named BIConsumers, BIAuthors, and BIAdministrators).

> **Tip:** For an example security model showing a set of Users, Groups, and Application Roles, see Section 2.3, "An Example Security Setup Using the Installed Groups and Application Roles".

Repeat this task for each User that you want to deploy

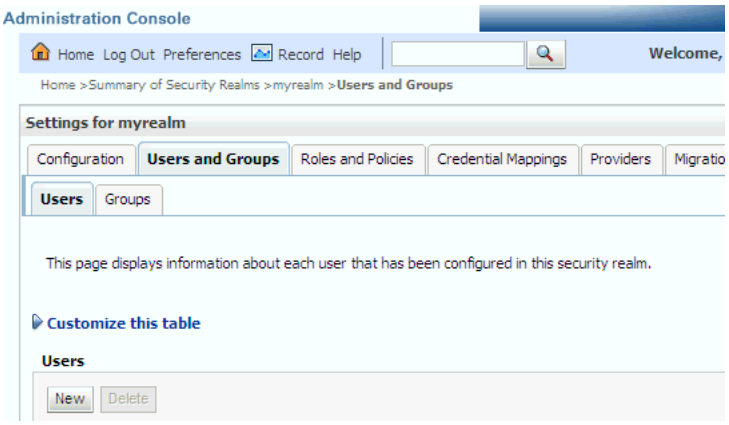
To create a user in the default directory server:

1. Launch Oracle WebLogic Server Administration Console.

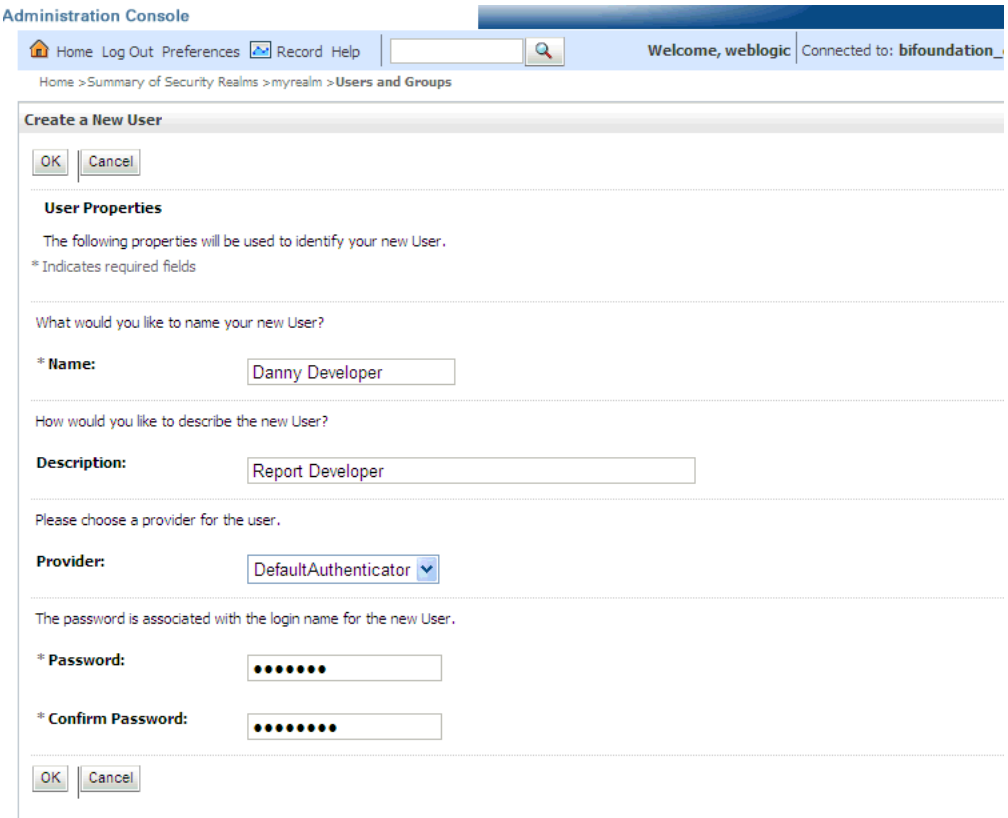
   For more information, see Section 2.4.2, "How to Launch Oracle WebLogic Server Administration Console".

2. In Oracle WebLogic Server Administration Console, select **Security Realms** from the left pane and click the realm you are configuring. For example, **myrealm**.

3. Select **Users and Groups** tab, then **Users**. Click **New**.



4. In the **Create a New User** page provide the following information:
   - **Name**: Enter the name of the user. See online help for a list of invalid characters.
   - (Optional) **Description**: Enter a description.
   - **Provider**: Select the authentication provider from the list that corresponds to the identity store where the user information is contained. DefaultAuthenticator is the name for the default authentication provider.
   - **Password**: Enter a password for the user that is at least 8 characters long.
   - **Confirm Password**: Re-enter the user password.



5. Click **OK**.

   The user name is added to the User table.

2.4.4 How to create a Group in the Embedded WebLogic LDAP Server

You typically create a separate Group for each functional type of business user in your Oracle Business Intelligence environment. For example, a typical deployment might require three Groups: BIConsumers, BIAuthors, and BIAdministrators. In this case, you could either use the preconfigured Groups named BIConsumers, BIAuthors, and BIAdministrators that are installed with Oracle Business Intelligence, or you might create your own custom Groups.

> **Tip:** For an example security model showing a set of Users, Groups, and Application Roles, see Section 2.3, "An Example Security Setup Using the Installed Groups and Application Roles".

Repeat this task for each Group that you want to deploy

To create a group in the default directory server:

1. Launch Oracle WebLogic Server Administration Console.

   For more information, see Section 2.4.2, "How to Launch Oracle WebLogic Server Administration Console".

2. In Oracle WebLogic Server Administration Console, select **Security Realms** from the left pane and click the realm you are configuring. For example, **myrealm**.

3. Select **Users and Groups** tab, then **Groups**. Click **New**

4. In the **Create a New Group** page provide the following information:

   - **Name**: Enter the name of the Group. Group names are case insensitive but must be unique. See online help for a list of invalid characters.
   - (Optional) **Description**: Enter a description.
   - **Provider**: Select the authentication provider from the list that corresponds to the identity store where the group information is contained. DefaultAuthenticator is the name for the default authentication provider.

5. Click **OK**

   The group name is added to the Group table.

2.4.5 How to add a User to a Group in the Embedded WebLogic LDAP Server

You typically add each User to an appropriate Group. For example, a typical deployment might require User IDs created for report consumers to be assigned to a Group named BIConsumers. In this case, you could either assign the Users to the default Group named BIConsumers, or you could assign the Users to your own custom Group that you have created.

> **Tip:** For an example security model showing a set of Users, Groups, and Application Roles, see Section 2.3, "An Example Security Setup Using the Installed Groups and Application Roles".

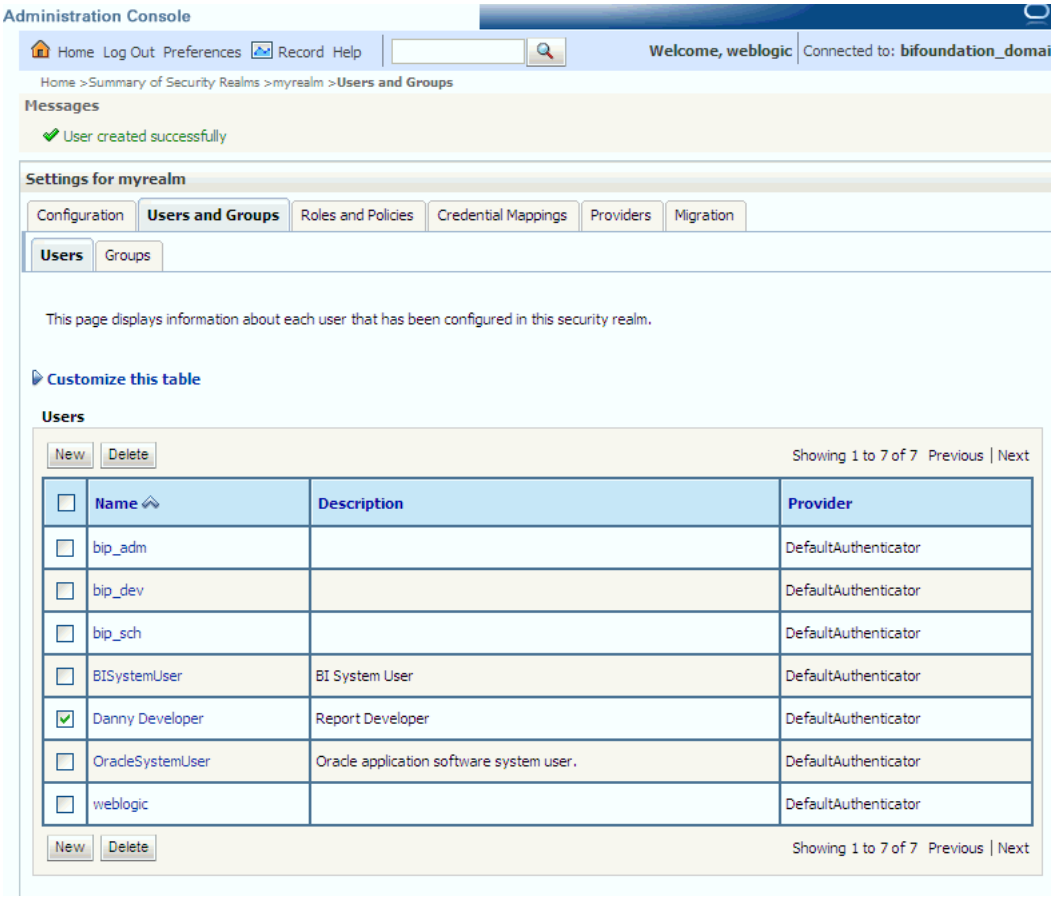
Repeat this task to assign each User to an appropriate Group.

To add a user to a group in the default directory server:

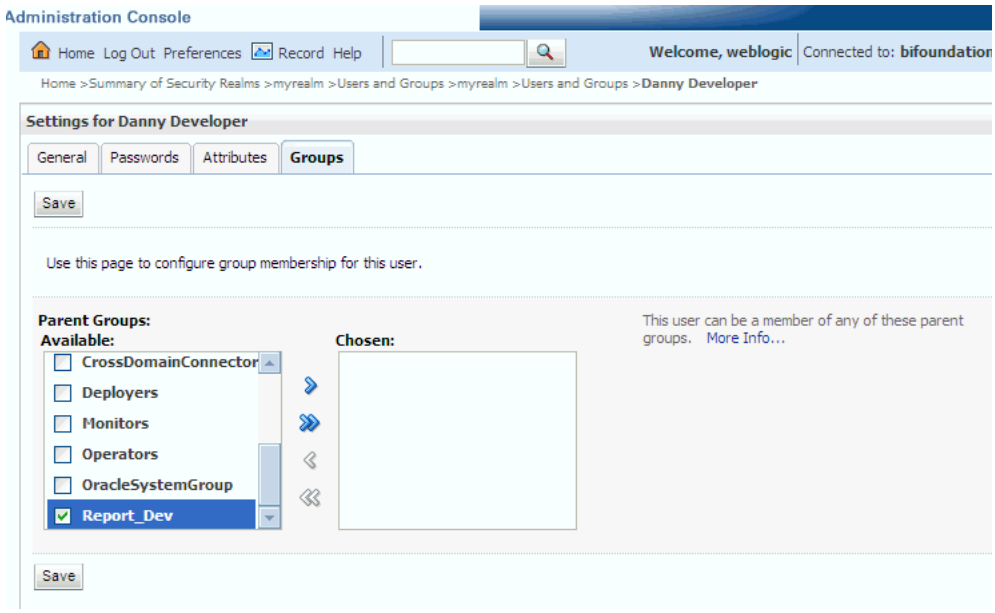
1. Launch Oracle WebLogic Server Administration Console.

   For more information, see Section 2.4.2, "How to Launch Oracle WebLogic Server Administration Console".

2. In Oracle WebLogic Server Administration Console, select **Security Realms** from the left pane and click the realm you are configuring. For example, **myrealm**.
3. Select **Users and Groups** tab, then **Users**.
4. In the Users table select the user you want to add to a group.



5. Select the **Groups** tab.
6. Select a group or groups from the **Available** list box.



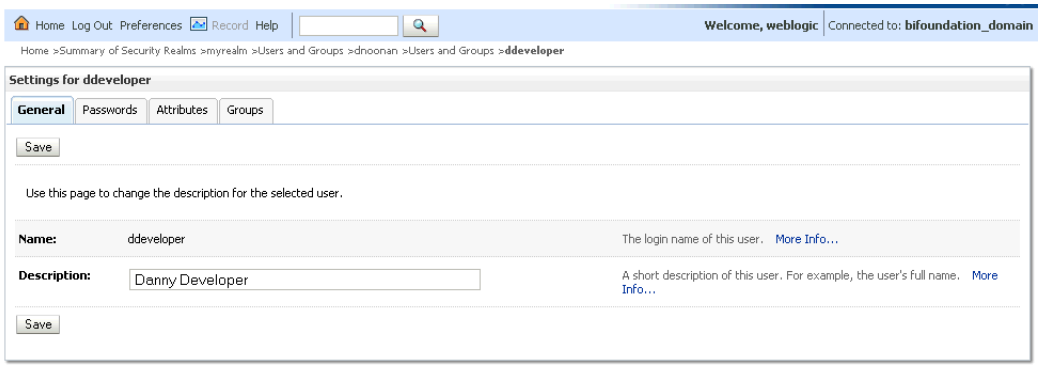
7. Click **Save**.

2.4.6 (Optional) How to change a User password in the Embedded WebLogic LDAP Server

Perform this optional task if you want to change the default password for a User.

To change a user password in the default directory server:

1. In Oracle WebLogic Server Administration Console, select **Security Realms** from the left pane and click the realm you are configuring. For example, **myrealm**.
2. Select **Users and Groups** tab, then **Users**
3. In the Users table select the user you want to change the password for. The user's **Settings** page displays.



4. Select the **Passwords** tab and enter the password in the **New Password** and **Confirm Password** fields.
5. Click **Save**.

2.5 Managing Application Roles and Application Policies Using Fusion Middleware Control

In Oracle Business Intelligence, you use Fusion Middleware Control to mange Application Roles and Application Policies that provide permissions for Users and Groups. For detailed information about using Fusion Middleware Control, see *Oracle Fusion Middleware Administrator's Guide*.

- Section 2.5.1, "Starting Oracle Fusion Middleware Control and Locate the Pages for Managing Security"
- Section 2.5.2, "Creating Application Roles Using Fusion Middleware Control"
- Section 2.5.3, "Creating Application Policies Using Fusion Middleware Control"
- Section 2.5.4, "Modifying Application Roles Using Oracle Fusion Middleware Control"

> **Tip:** If you are using the default Groups (that is, BIConsumers, BIAuthors, and BIAdministrators) that are installed with the default embedded WebLogic LDAP Server, then these Groups are mapped to an appropriate Application Role (that is, BIConsumer, BIAuthor, or BIAdministrator). No additional steps are required to map the default Groups to Application Roles.

The simplest way to set up security is to map your Groups to the default Application Roles that are installed and pre-configured out-of-the-box, (that is, BIConsumer, BIAuthor, and BIAdministrator). Each default Group is pre-configured to use the appropriate default Application Role. For example, the default Group named BIAuthors is mapped to the default Application Role named BIAuthor. In other words, any Users that you add to the default Group named BIAuthors automatically have the privileges required to create reports and perform related duties.

If you want to create a more complex or fine grained security model, you might create your own Application Roles and Application Policies as described in this section. For example, you might want report authors in a Marketing department to only have write-access to the Marketing area of the metadata repository and Presentation Catalog. To achieve this, you might create a new Application Role called BIAuthorMarketing, and provide it with appropriate privileges.

> **Caution:** If you are deploying the default Policy Store, then Oracle recommends that you make a copy of the original system-jazn-data.xml policy file and place it in a safe location. Use the copy of the original file to restore the default policy store configuration, if needed. Changes to the default security configuration might lead to an unwanted state. The default location is *MW_HOME*/user_projects/domain/*your_domain*/config/fmwconfig.

To set up the Application Roles that you want to deploy, do the following:

- If required, create new Application Roles. For more information, see Section 2.5.2, "Creating Application Roles Using Fusion Middleware Control".

  Note: You can create Application Roles based on existing Application Policies that are installed and configured out-of-the-box, or you can create your own Application Policies. For more information about the default Users, Groups, and

Application Roles, see Section 2.2, "Working with the default Users, Groups, and Application Roles Installed Out-Of-The-Box".

- If required, create new Application Policies. For more information, see Section 2.5.3, "Creating Application Policies Using Fusion Middleware Control".
- (Optional) If required, modify the permission grants or membership for an Application Role. For more information, see Section 2.5.4, "Modifying Application Roles Using Oracle Fusion Middleware Control".

2.5.1 Starting Oracle Fusion Middleware Control and Locate the Pages for Managing Security

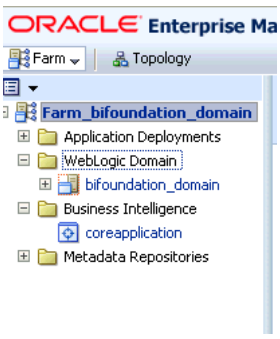
This section explains how to start Oracle Fusion Middleware Control and Locate the pages used to manage security components.

2.5.1.1 Overview

Fusion Middleware Control is a Web browser-based, graphical user interface that you can use to monitor and administer a farm. A farm is a collection of components managed by Fusion Middleware Control. It can contain Oracle WebLogic Server domains, one Administration Server, one or more Managed Servers, clusters, and the Oracle Fusion Middleware components that are installed, configured, and running in the domain. During installation an Oracle WebLogic Server domain is created and Oracle Business Intelligence is installed into that domain. If you performed a Simple or Enterprise installation type, this domain is named **bifoundation_domain** and is located under WebLogic Domain in the Fusion Middleware Control target navigation pane.

Launch Fusion Middleware Control by entering its URL into a Web browser. The URL includes the name of the host and the administration port number assigned during the installation. This URL takes the following form: http://*hostname:port_number*/em. The default port is 7001.

There are several methods available for accessing the common Fusion Middleware Control security pages used when managing the Oracle Business Intelligence security configuration. Depending upon the access point used in the target navigation pane, the obi application stripe is pre-selected for you. The access points are as follows:



- From **coreapplication** - You can reach the **Application Policies** and **Application Roles** pages using a shortcut menu. The obi application stripe is pre-selected and the Oracle Business Intelligence Application Policies or Application Roles are displayed. You cannot reach all Fusion Middleware Control Security menu options from this shortcut menu.

  For more information, see Section 2.5.1.2, "How to display the Security Menu from coreapplication".

- From **bifoundation_domain** - If you select either **Application Policies** or **Application Roles** from the **Security** menu, the obi application stripe must be selected and a search initiated. All Fusion Middleware Control Security menu options are available from this method.

  For more information, see Section 2.5.1.3, "How to display the Security Menu from bifoundation_domain".

For more information about using Fusion Middleware Control, see *Oracle Fusion Middleware Administrator's Guide*.

2.5.1.2 How to display the Security Menu from coreapplication

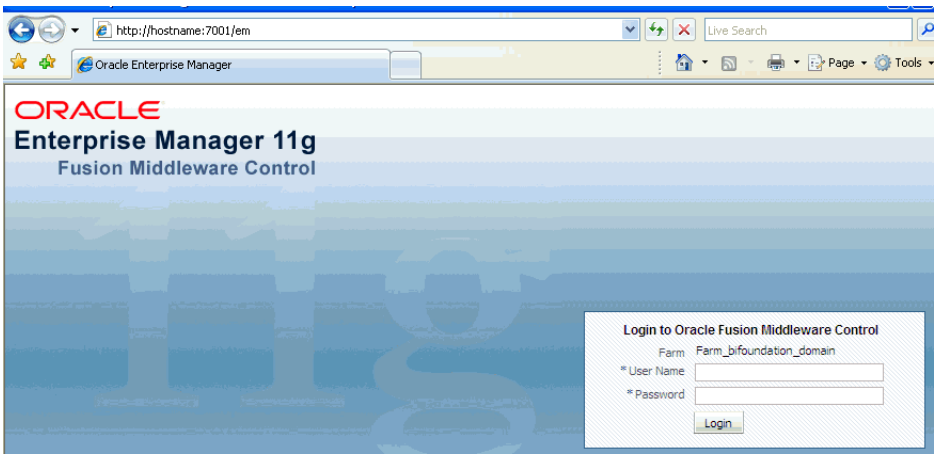
To display the Security menu in Fusion Middleware Control from coreapplication:

Using one of the following methods provides a shortcut for accessing the **Application Policies** or **Application Roles pages** with the obi application stripe pre-selected and the corresponding Oracle Business Intelligence policies or roles displaying.

1. Log in to Fusion Middleware Control by entering the URL in a Web browser.

   For example, http://*hostname:port_number*/em.

   The Fusion Middleware Control login page displays.

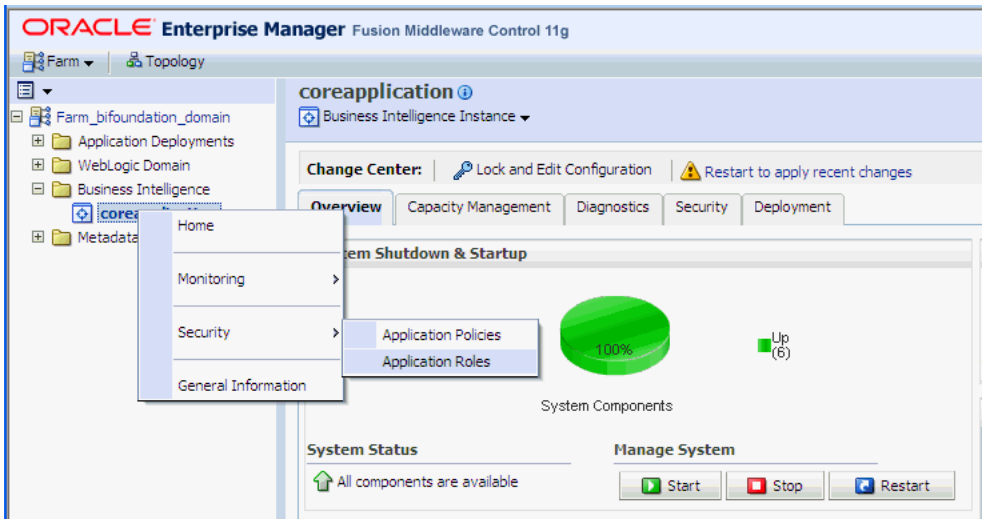


2. Enter the Oracle Business Intelligence administrative user name and password and click **Login**

   The password is the one you supplied during the installation of Oracle Business Intelligence. If these values have been changed, then use the current administrative user name and password combination.

3. From the target navigation pane, open **Business Intelligence** and select **coreapplication**. Display the **Security** menu by selecting one of the following methods:

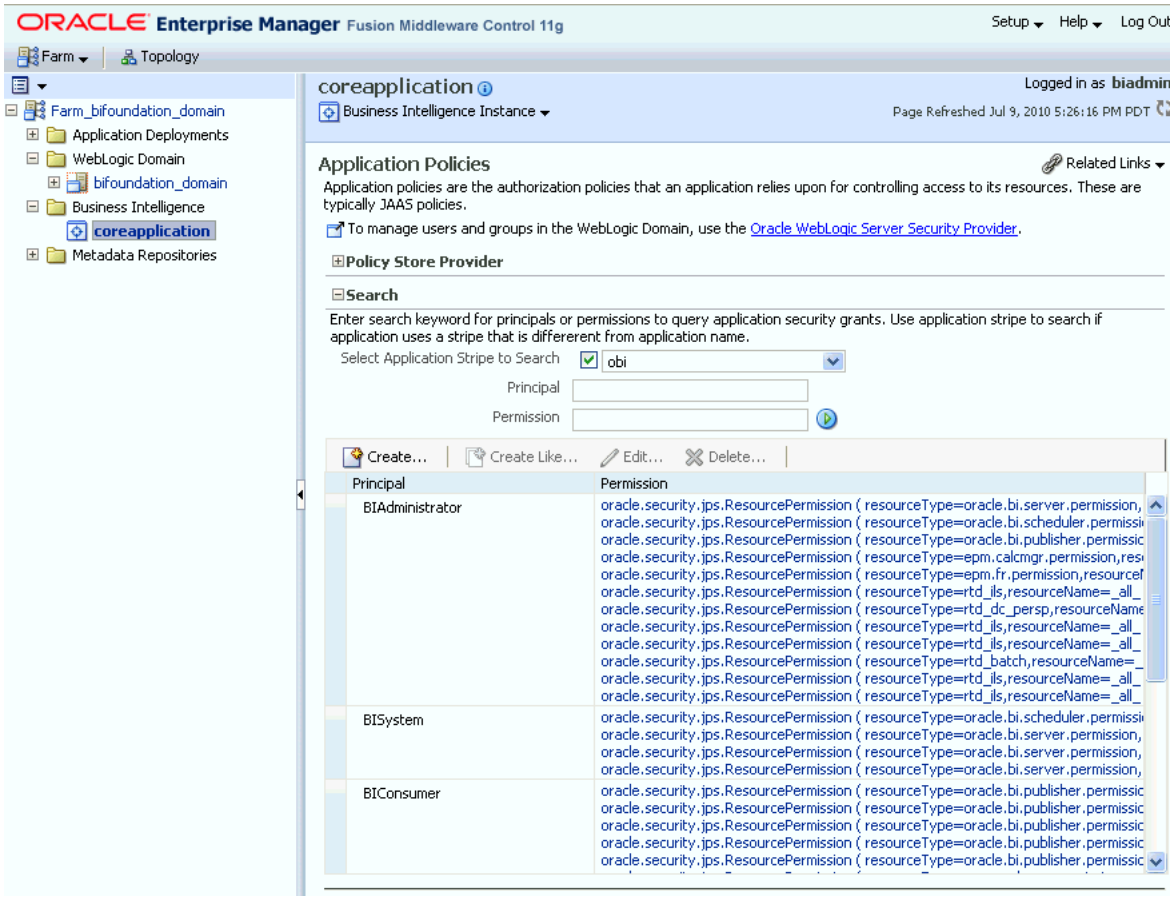
   - Right-click **coreapplication**, then select **Security** to display a submenu with **Application Policies** and **Application Roles** as options.



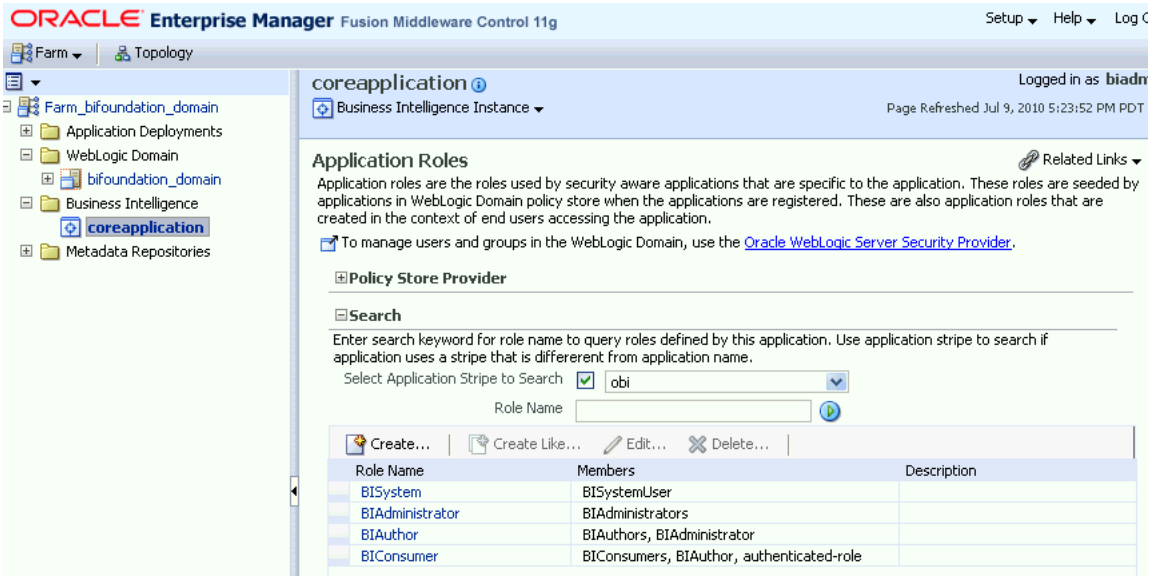
- From the content pane, select the **Business Intelligence Instance** menu, then select **Security** to display a submenu with **Application Policies** and **Application Roles** as options.

4. Select **Application Policies** or **Application Roles** as needed. The obi application stripe is selected and the corresponding Oracle Business Intelligence policies or roles are displayed.

   - The following figure shows an example of **Application Policies** page displaying the default Oracle Business Intelligence Application Policies.



- The following figure shows an example of **Application Roles** page displaying the default Oracle Business Intelligence Application Roles.



2.5.1.3 How to display the Security Menu from bifoundation_domain

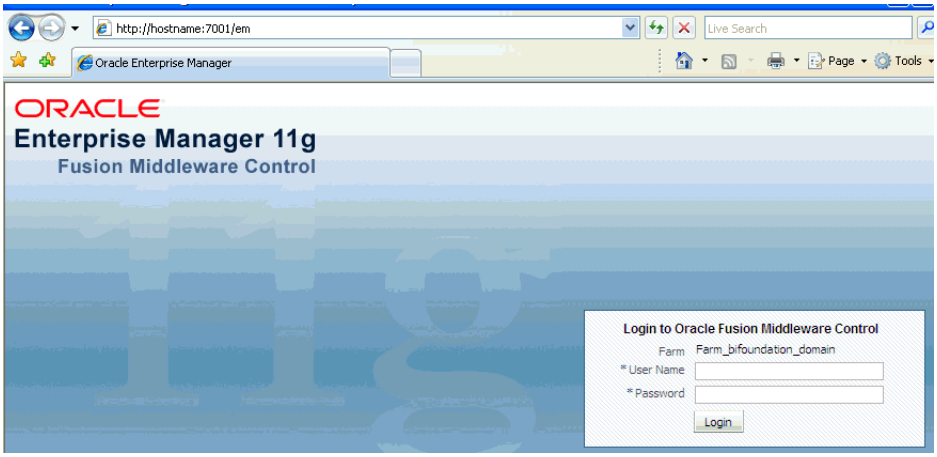
To display the Security menu in Fusion Middleware Control from bifoundation_domain:

Using one of the following methods requires you later select the obi application stripe to search for the Oracle Business Intelligence Application Policies or Application Roles.

1. Log in to Fusion Middleware Control by entering the URL in a Web browser.

   For example, http://*hostname:port_number*/em.

   The Fusion Middleware Control login page displays.

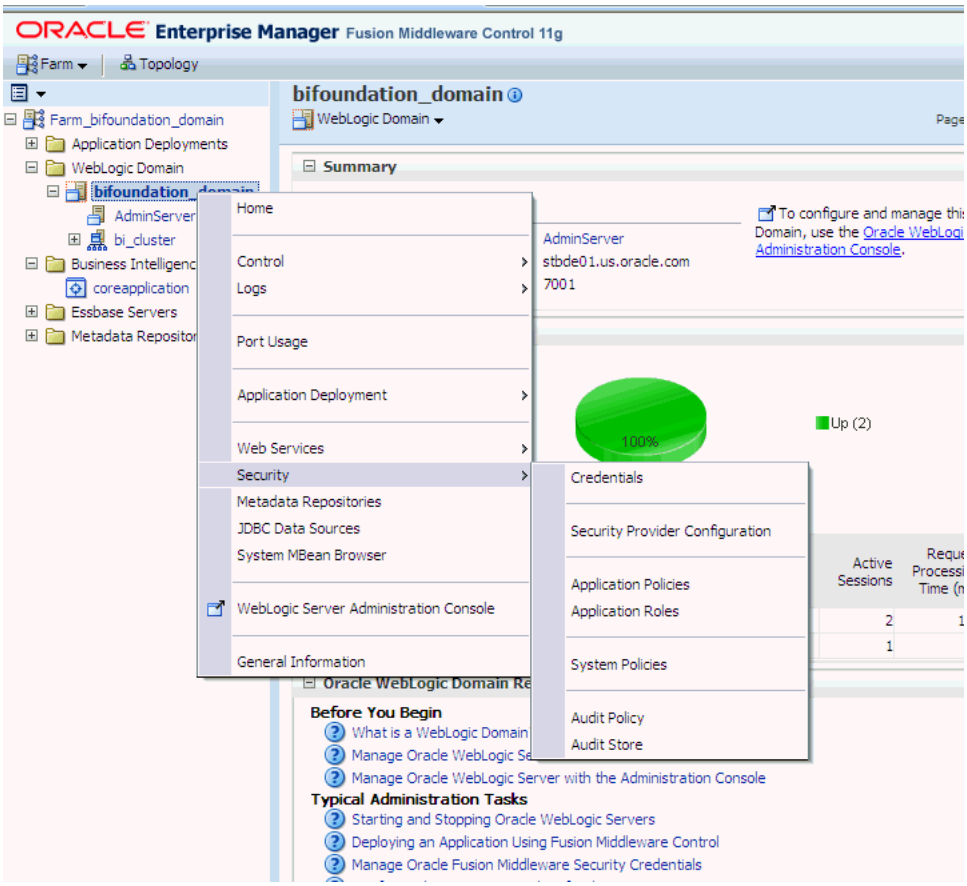


2. Enter the Oracle Business Intelligence administrative user name and password and click **Login**.

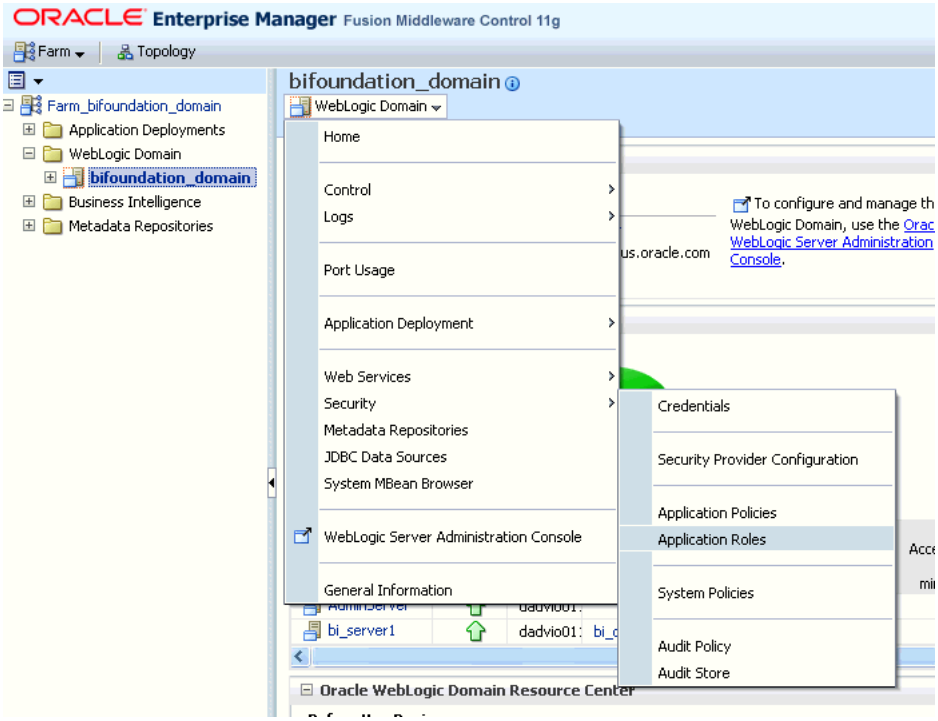
   The password is the one you supplied during the installation of Oracle Business Intelligence. If these values have been changed, then use the current administrative user name and password combination.

3. From the target navigation pane, open **WebLogic Domain** and select **bifoundation_domain**. Display the **Security** menu by selecting one of the following methods:

   - Right-click **bifoundation_domain**, then select **Security** to display a submenu.



   - From the content pane, select the **WebLogic Server** menu, then select **Security** to display a submenu.



2.5.2 Creating Application Roles Using Fusion Middleware Control

This section explains how to create and manage Application Roles using Oracle Fusion Middleware Control, and contains the following topics:

- Section 2.5.2.1, "Overview to creating and managing Application Roles"
- Section 2.5.2.2, "How to Create an Application Role"
- Section 2.5.2.3, "How to map a Group to an Application Role"

2.5.2.1 Overview to creating and managing Application Roles

In a new Oracle Business Intelligence deployment, you typically create a Application Roles for each type of business user activity in your Oracle Business Intelligence environment. For example, a typical deployment might require three Application Roles: BIConsumer, BIAuthors, and BIAdministrator. In this case, you could either use the preconfigured Application Roles named BIConsumer, BIAuthor, and BIAdministrator that are installed with Oracle Business Intelligence, or you could create your own custom Application Roles. For more information about the installed Application Roles that are available out-of-the-box, see Section 2.2, "Working with the default Users, Groups, and Application Roles Installed Out-Of-The-Box".

Oracle Business Intelligence Application Roles represent a role that a user has. For example, having the Sales Analyst Application Role might grant a user access to view, edit and create reports on a company's sales pipeline. You can create new Application Roles to supplement or replace the default roles configured during installation. Keeping Application Roles separate and distinct from the directory server groups enables you to better accommodate authorization requirements. You can create new Application Roles to match business roles for your environment without needing to change the groups defined in the corporate directory server. To control authorization requirements more efficiently, you can then map existing groups of users from the directory server to Application Roles.

Note: Before creating a new Application Role and adding it to the default Oracle Business Intelligence security configuration, familiarize yourself with how permission and group inheritance works. It is important when constructing a role hierarchy that circular dependencies are not introduced. For more information, see Section B.4.4, "How Permissions Are Granted Using Application Roles".

For more information about creating Application Roles, see "Managing Policies with Fusion Middleware Control" in Oracle Fusion Middleware Security Guide.

Note: For advanced-level information about using a BI repository in offline mode, see Section 2.6.3.1, "About Managing Application Roles in the Metadata Repository".

2.5.2.2 How to Create an Application Role

There are two methods for creating a new Application Role:

- **Create New** - A new Application Role is created. Members can be added at the same time or you can save the new role after naming it and add members later.
- **Copy Existing** - A new Application Role is created by copying an existing Application Role. The copy contains the same members as the original, and is made a Grantee of the same Application Policy as is the original. Modifications can be made as needed to the copy to further customize the new Application Role.

Membership in an Application Role is controlled using the **Application Roles** page in Fusion Middleware Control. Valid members of an Application Role are users, groups, and other Application Roles.

Permission grants are controlled in the **Application Policies** page in Fusion Middleware Control. The permission grant definitions are set in the Application Policy, then the Application Policy is *granted* to the Application Role. For more information, see Section 2.5.3, "Creating Application Policies Using Fusion Middleware Control".

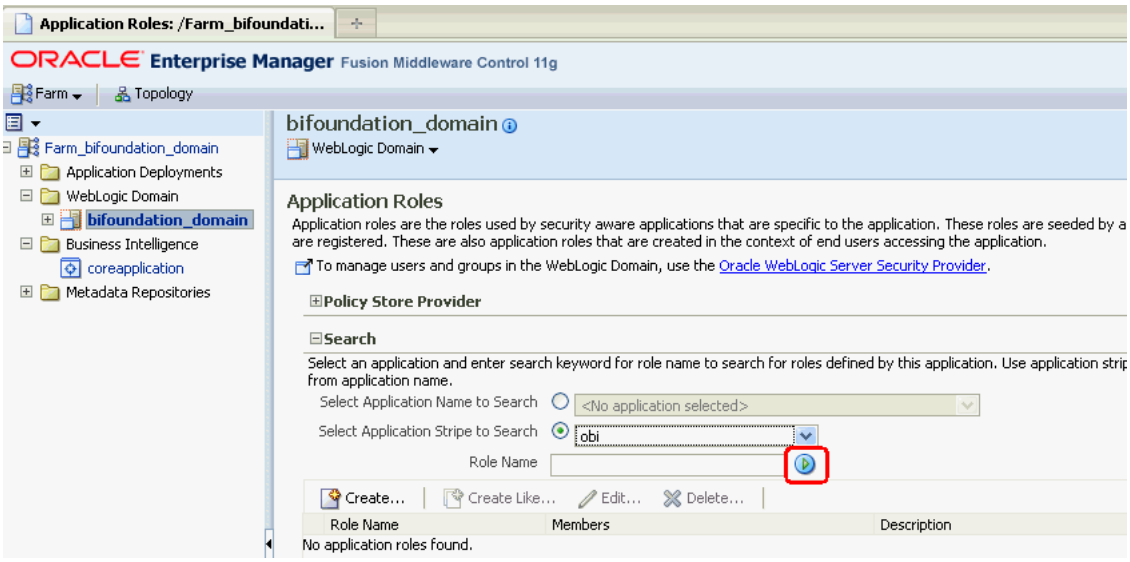
To create a new Application Role:

1. Log in to Fusion Middleware Control, navigate to **Security**, then select **Application Roles** to display the **Application Roles** page.

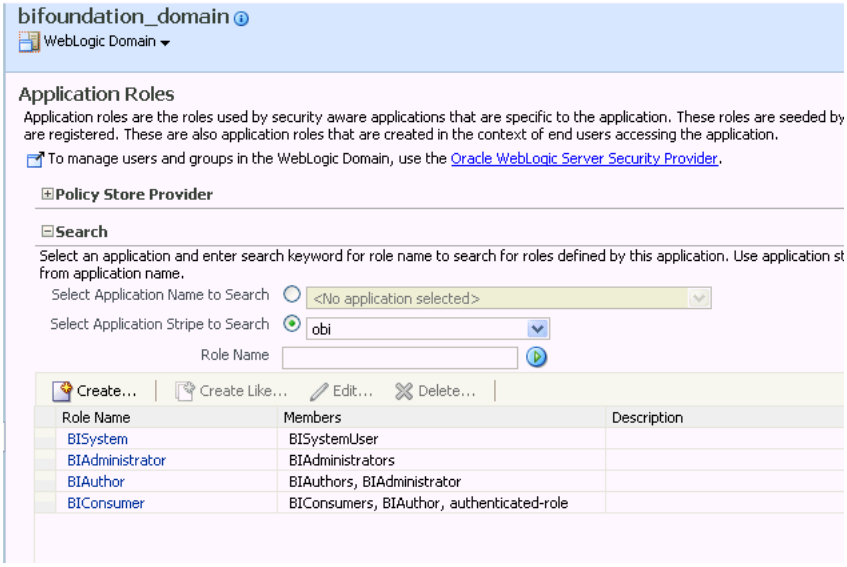
   For information, see Section 2.5.1, "Starting Oracle Fusion Middleware Control and Locate the Pages for Managing Security".

   Whether or not the obi application stripe is pre-selected and the Application Policies are displayed depends upon the method used to navigate to the **Application Roles** page.

2. If necessary, select **Select Application Stripe to Search**, then select **obi** from the list. Click the search icon next to **Role Name**.



The Oracle Business Intelligence Application Roles display. The following figure shows the default Application Roles.



3. Click **Create** to display the **Create Application Role** page. You can enter all information at once or you can enter a **Role Name**, save it, and complete the remaining fields later. Complete the fields as follows:

   In the **General** section:

   - **Role Name** - Enter the name of the Application Role
   - (Optional) **Display Name** - Enter the display name for the Application Role.
   - (Optional) **Description** - Enter a description for the Application Role.

   In the **Members** section, select the users, groups, or Application Roles to be mapped to the Application Role. Select **Add Application Role** or **Add Group** or **Add Users** accordingly. To search in the dialog box that displays:

   - Enter a name in **Name** field and click the blue button to search.
   - Select from the results returned in the **Available** box.

- Use the shuttle controls to move the desired name to the **Selected** box.
- Click **OK** to return to the **Create Application Role** page.
- Repeat the steps until all desired members are added to the Application Role.

4. Click **OK** to return to the **Application Roles** page.

   The Application Role just created displays in the table at the bottom of the page.

To create an Application Role based on an existing one:

1. Log in to Fusion Middleware Control, navigate to **Security**, then select **Application Roles** to display the **Application Roles** page.

   For information, see Section 2.5.1, "Starting Oracle Fusion Middleware Control and Locate the Pages for Managing Security".

   Whether or not the obi application stripe is pre-selected and the Application Policies are displayed depends upon the method used to navigate to the **Application Roles** page.

2. If necessary, select **Select Application Stripe to Search**, then select **obi** from the list. Click the search icon next to **Role Name**.

   The Oracle Business Intelligence Application Roles display.

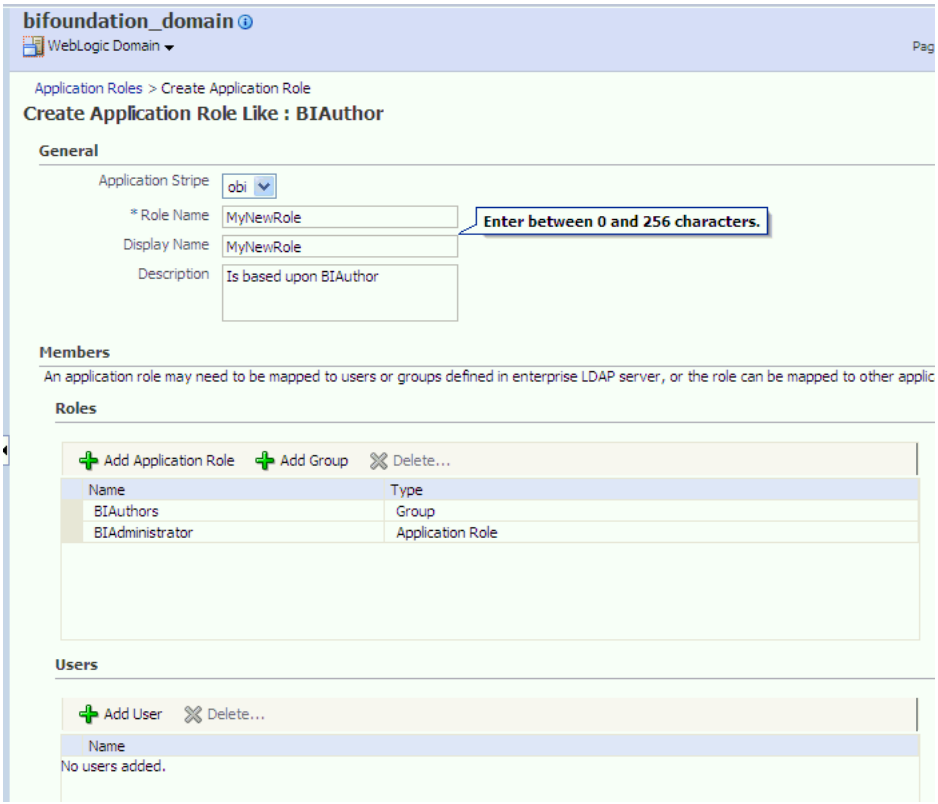
3. Select the Application Role you want to copy from the list to enable the action buttons.

4. Click **Create Like** to display the **Create Application Role Like** page.

   The **Members** section is completed with the same Application Roles, groups, or users that are mapped to the original role.

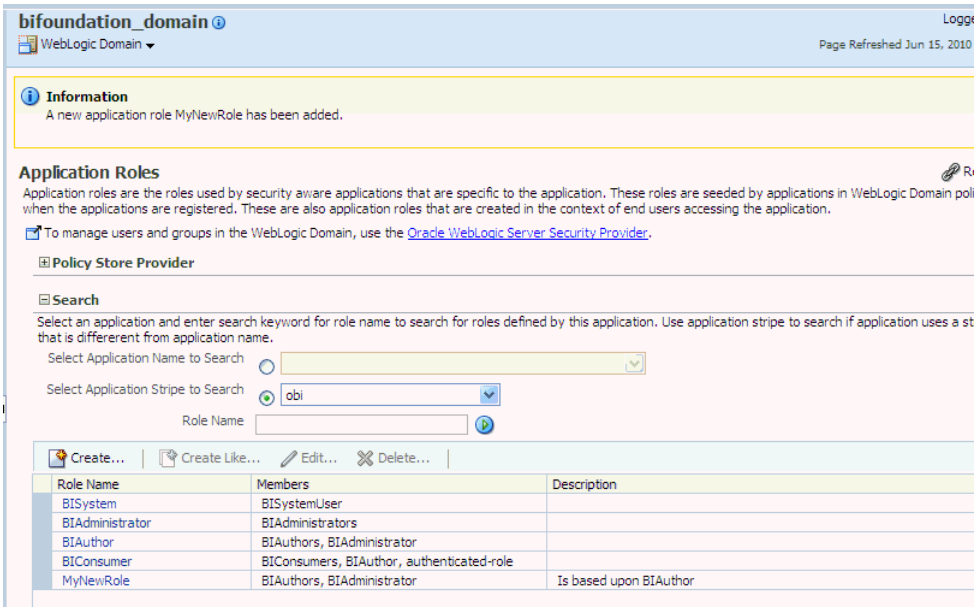
5. Complete the **Role Name**, **Display Name**, and **Description** fields.

   The following figure shows a new Application Role that is based upon the default BIAuthor Application Role and has been named **MyNewRole**.



6. Modify the members as appropriate and click **OK**.

   The just created Application Role displays in the table at the bottom of the page. The following figure shows the example **MyNewRole** that is based upon the default BIAuthor Application Role.



2.5.2.3 How to map a Group to an Application Role

You map a Group to an Application Role to provide Users in that Group with appropriate security privileges. For example, a Group for marketing report consumers

named BIMarketingGroup might require an Application Role called BIConsumerMarketing, in which case you map the Group named BIMarketingGroup to the Application Role named BIConsumerMarketing.

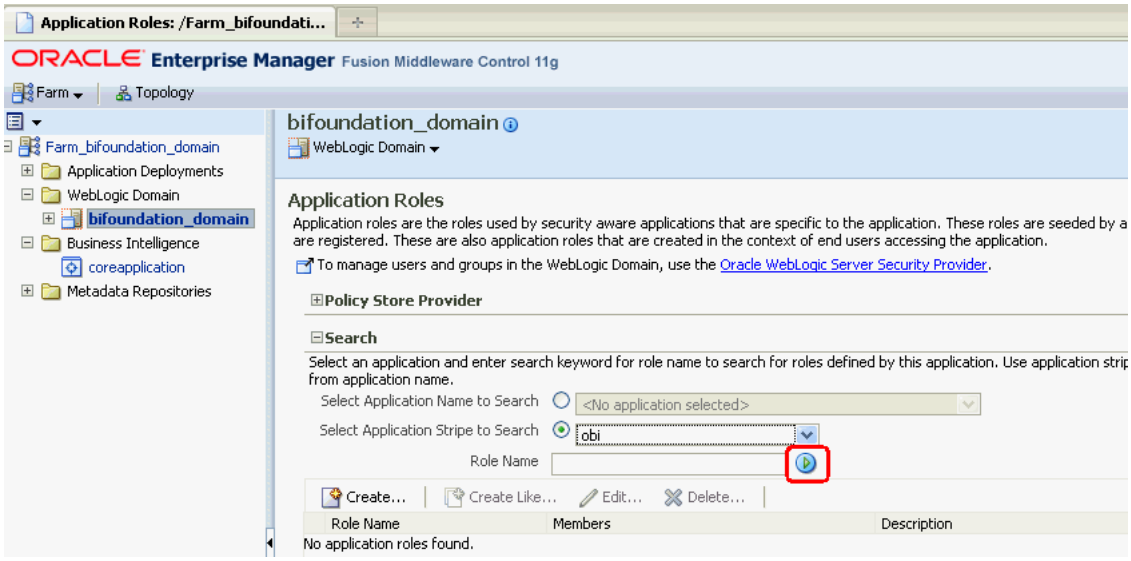
To map a Group to an Application Role:

1. Log in to Fusion Middleware Control, navigate to **Security**, then select **Application Roles** to display the **Application Roles** page.

   For information, see Section 2.5.1, "Starting Oracle Fusion Middleware Control and Locate the Pages for Managing Security".

   Whether or not the obi application stripe is pre-selected and the Application Policies are displayed depends upon the method used to navigate to the **Application Roles** page.

2. If necessary, select **Select Application Stripe to Search**, then select **obi** from the list. Click the search icon next to **Role Name**.



The Oracle Business Intelligence Application Roles display. The following figure shows the default Application Roles.

3. Select an Application Role in the list and click Edit to display an edit dialog, and complete the fields as follows:
4. In the **Members** section, use the **Add Group** option to add the Group that you want to map to the **Roles** list.

   For example, if a Group for marketing report consumers named BIMarketingGroup require an Application Role called BIConsumerMarketing, then add the Group named BIMarketingGroup to **Roles** list.
5. Click **OK** to return to the **Application Roles** page.

2.5.3 Creating Application Policies Using Fusion Middleware Control

You can create Application Roles based on existing Application Policies that are installed and configured out-of-the-box, or you can create your own Application Policies.

All Oracle Business Intelligencer permissions are provided after installation and you cannot create new permissions. The Application Policy is the mechanism that defines the permissions grants. Permission grants are controlled in the Fusion Middleware Control **Application Policies** page. The permission grants are defined in an Application Policy. An Application Role, user, or group, is then mapped to an Application Policy. This process makes the Application Role a **Grantee** of the Application Policy.

There are two methods for creating a new Application Policy:

- **Create New** - A new Application Policy is created and permissions are added to it.
- **Copy Existing** - A new Application Policy is created by copying an existing Application Policy. The copy is named and existing permissions are removed or permissions are added.

For more information about creating Application Policies, see "Managing Policies with Fusion Middleware Control" in Oracle Fusion Middleware Security Guide.

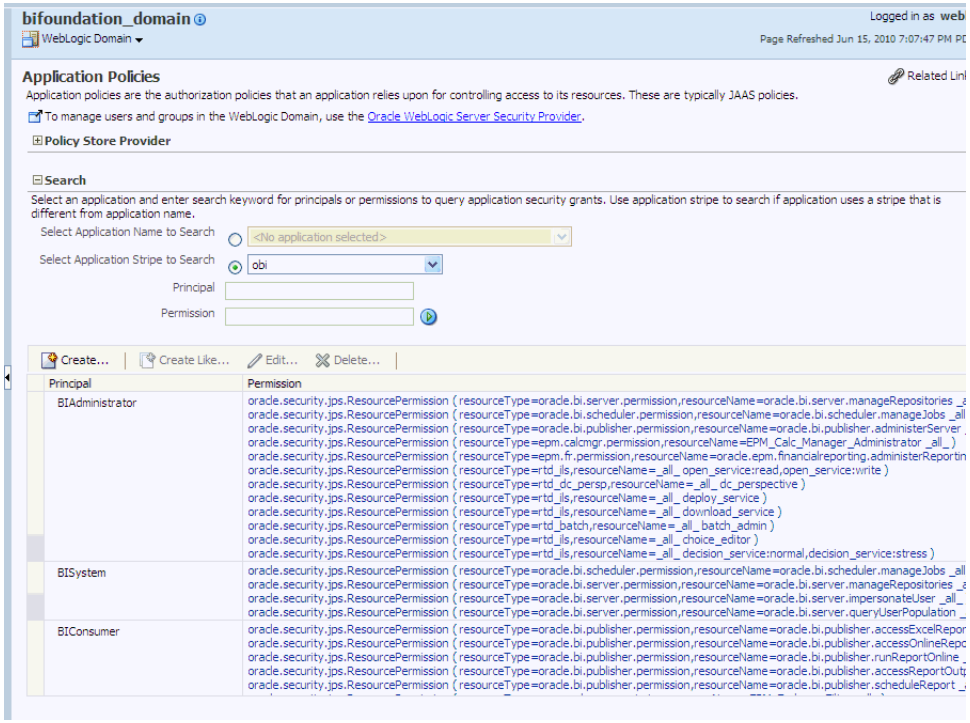
To create a new Application Policy:

1. Log in to Fusion Middleware Control, navigate to **Security**, then select **Application Policies** to display the **Application Policies** page.

   For information, see Section 2.5.1, "Starting Oracle Fusion Middleware Control and Locate the Pages for Managing Security".

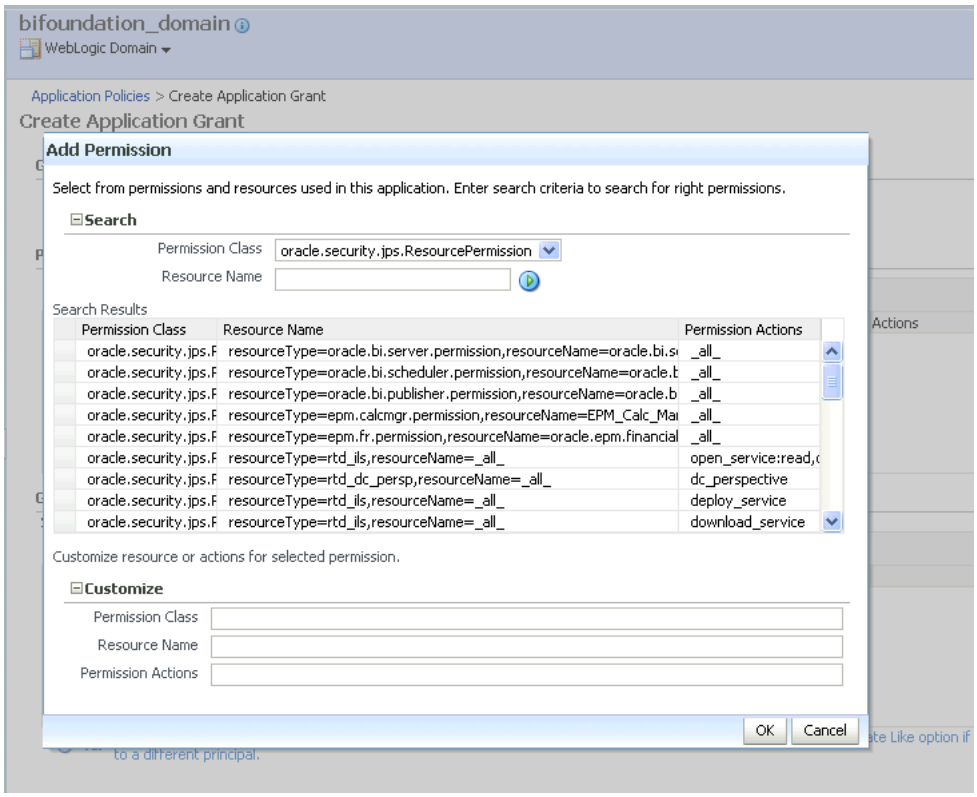
   Whether or not the obi application stripe is pre-selected and the Oracle Business Intelligence Application Policies are displayed depends upon the method used to navigate to the **Application Policies** page.
2. If necessary, select **Select Application Stripe to Search**, then select the **obi** from the list. Click the search icon next to **Role Name**.

   The Oracle Business Intelligence Application Policies are displayed. The **Principal** column displays the name of the policy **Grantee**.



3. Click **Create** to display the **Create Application Grant** page.
4. To add permissions to the policy being created, click **Add** in the **Permissions** area to display the **Add Permission** dialog.
   - Complete the **Search** area and click the blue search button next to the **Resource Name** field.

     All permissions located in the **obi** application stripe are displayed.

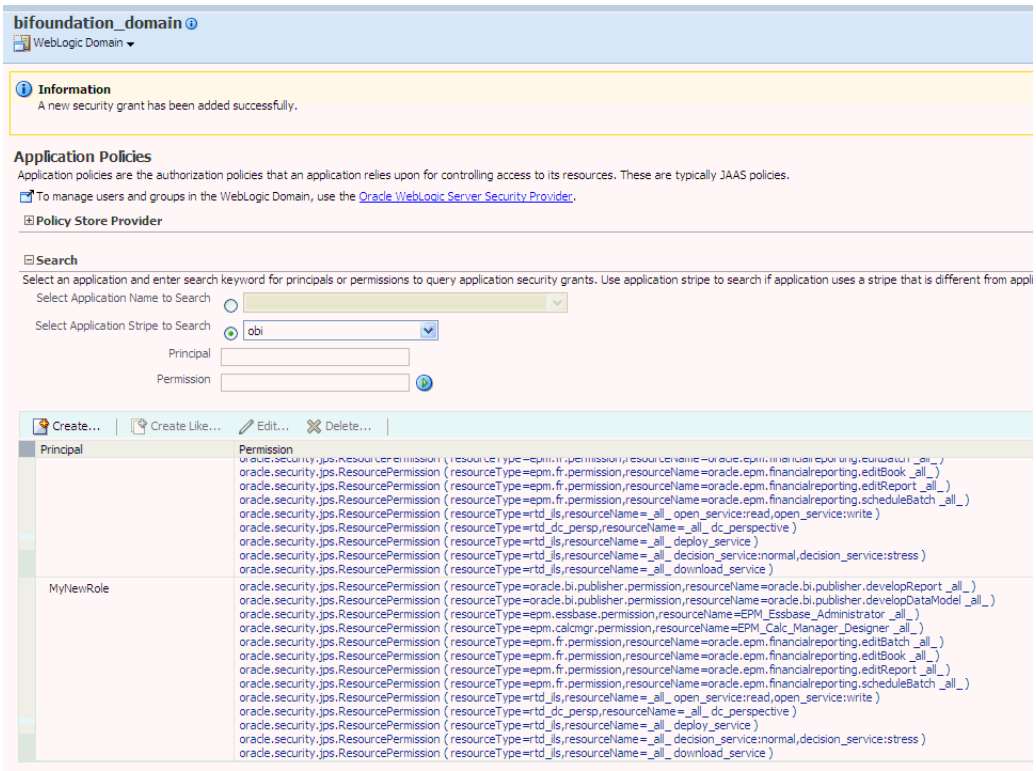


- Select the desired Oracle Business Intelligencer permission and click **OK**. Repeat until all desired permissions are selected. Selecting non-Oracle Business Intelligence permissions have no effect in the policy.
- To remove any items, select it and click **Delete**.

You are returned to the **Create Application Grant** page. The selected permissions display in the **Permissions** area.

5. To add an Application Role to the policy being created, click **Add Application Role** in the **Grantee** area to display the **Add Application Role** dialog.
   - Complete the **Search** area and click the blue search button next to the **Resource Name** field.
   - Select from the **Available Roles** list and use the shuttle controls to move it to **Selected Roles**.
   - Click **OK**.

   You are returned to the **Application Policies** page. The Principal and Permissions of the policy created are displayed in the table. The following figure shows the new Application Policy just created with MyNewRole Application Role as the Grantee (**Principal**).



To create an Application Policy based on an existing one:

1. Log in to Fusion Middleware Control, navigate to **Security**, then select **Application Policies** to display the **Application Policies** page.

   For information, see Section 2.5.1, "Starting Oracle Fusion Middleware Control and Locate the Pages for Managing Security".

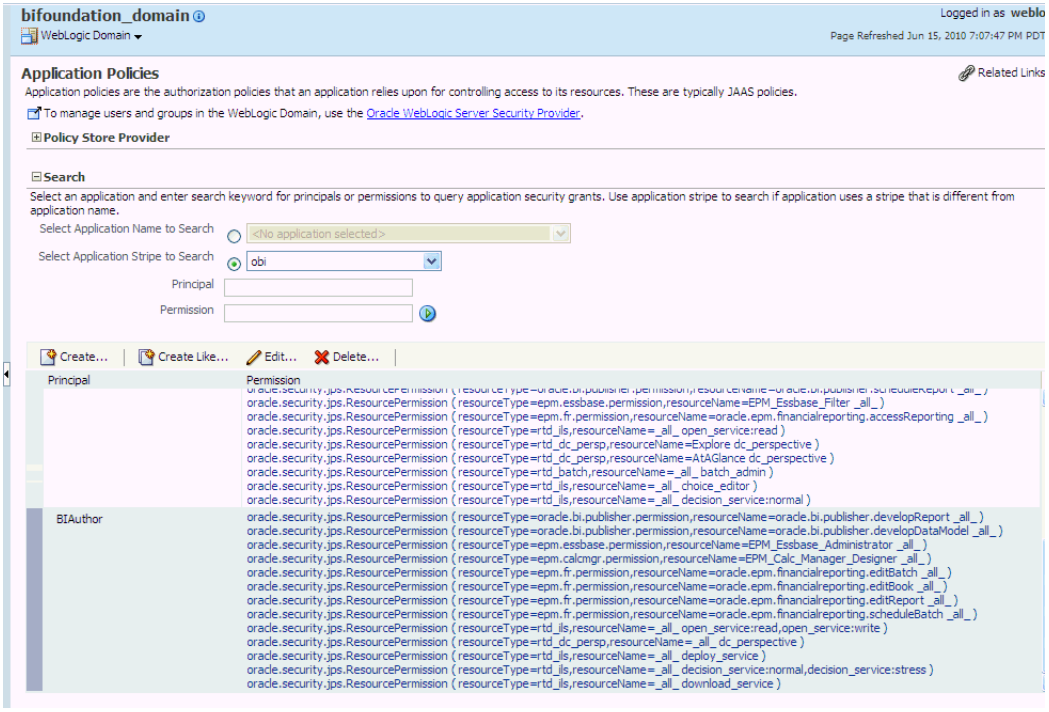
   Whether or not the obi application stripe is pre-selected and the Application Policies are displayed depends upon the method used to navigate to the **Application Policies** page.

2. If necessary, select **Select Application Stripe to Search**, then select the **obi** from the list. Click the search icon next to **Role Name**.

   The Oracle Business Intelligence Application Policies are displayed. The **Principal** column displays the name of the policy **Grantee**.

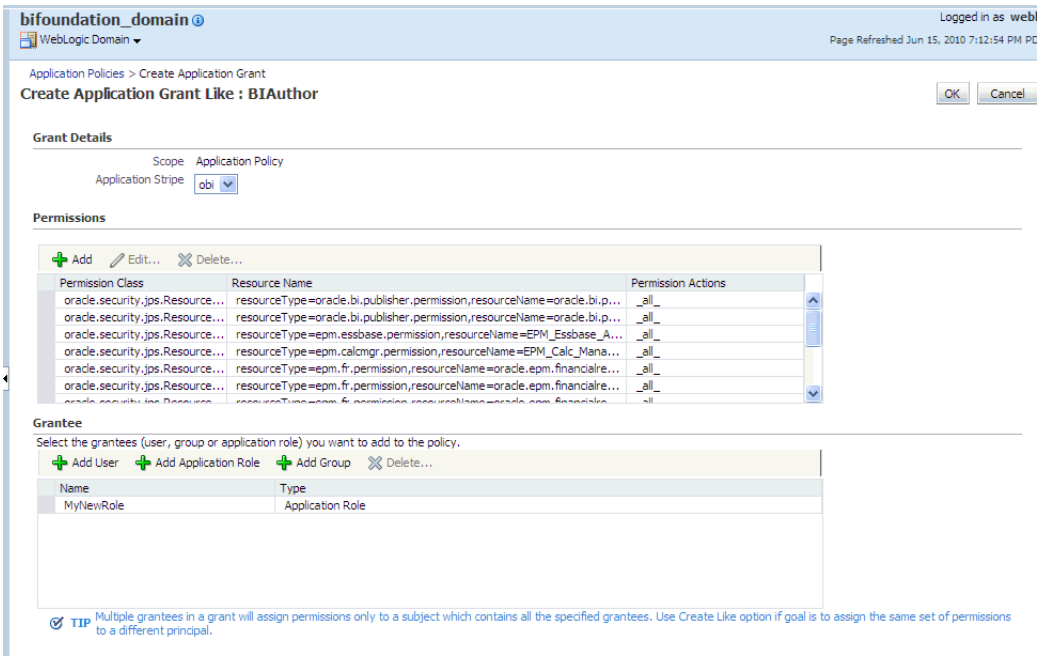
3. Select an existing policy from the table.

   The following figure shows the **BIAuthor** Principal selected with the **Create Like** button activated, which is used as an example in this procedure.



4. Click **Create Like** to display the **Create Application Grant Like** page. The Permissions table is automatically filled in with permissions granted by the policy selected.

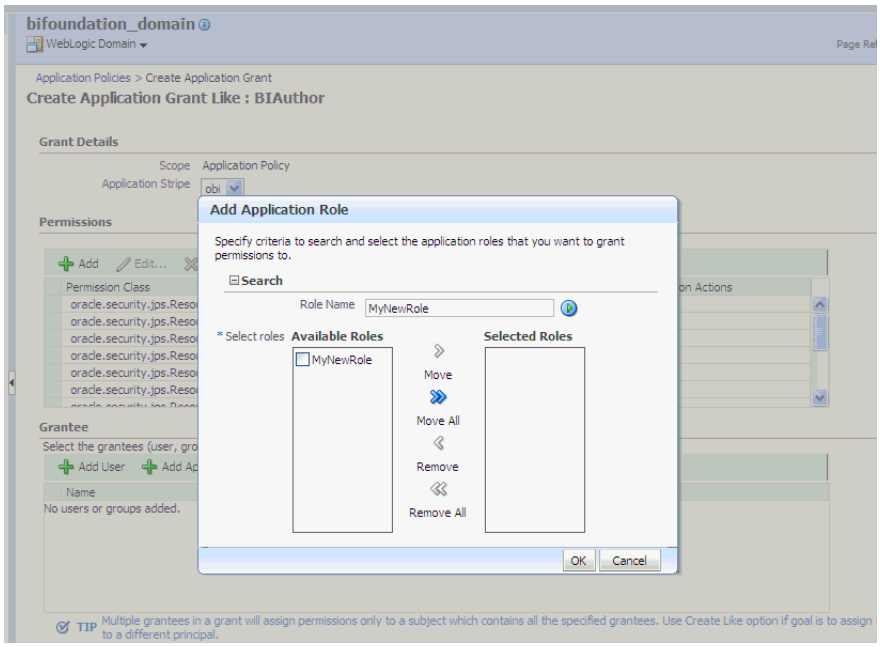
   The following figure shows the **Create Application Grant Like** dialog after the BIAuthor policy has been selected. Note that the Permissions section is completed with the permission grants for the BIAuthor policy.



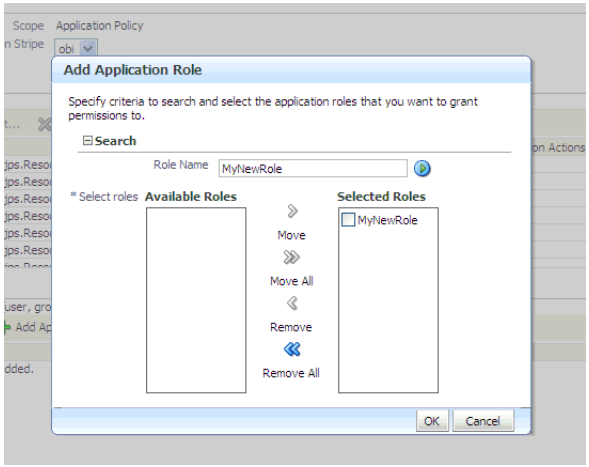
5. To remove any items, select it and click **Delete**.
6. To add Application Roles to the policy, click **Add Application Role** in the **Grantee** area to display the **Add Application Role** dialog.

The following figures use the **MyNewRole** Application Role as an example.

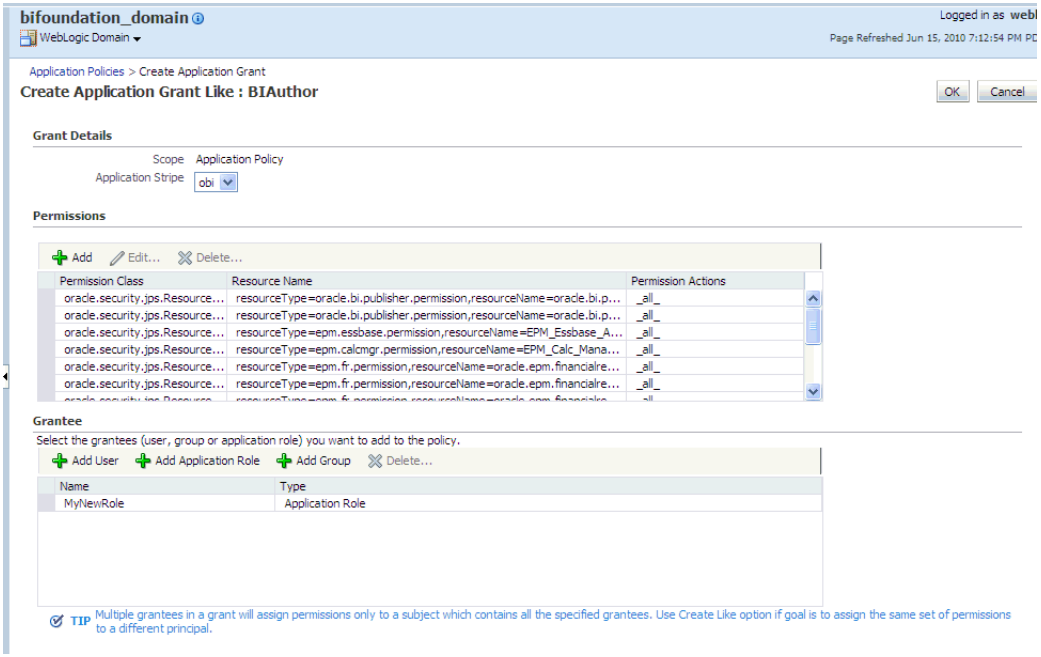
- Complete the **Search** area and click the blue search button next to the **Resource Name** field. The Application Roles matching the search are displayed.



- Select from the **Available Roles** list and use the shuttle controls to move it to **Selected Roles**. The **Create Application Grant Like** page displays with the selected Application Role added as **Grantee**.

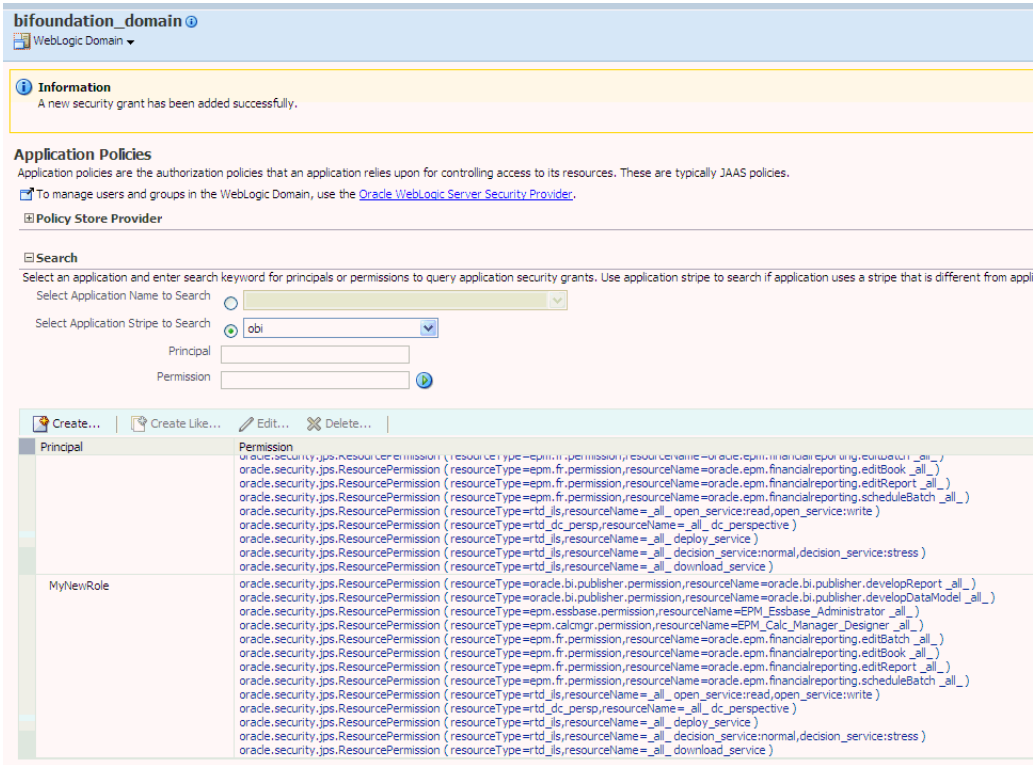


- Click **OK**. You are returned to the **Create Application Grant Like** dialog and the **Grantee** section is completed.



- Click **OK** to return to the **Application Policies** page.

  The Principal and Permissions of the Application Policy just created are displayed in the table.



2.5.4 Modifying Application Roles Using Oracle Fusion Middleware Control

The members of an Application Role can be changed using Oracle Fusion Middleware Control. If an Application Role is the Grantee of an Application Policy, the

permissions grants are changed by modifying the permission grants of the corresponding Application Policy.

> **Caution:** Oracle recommends that you do not change the permission grants and membership for the default Application Roles name BIConsumer, BIAuthor, and BIAdministrator.

For more information about managing Application Policies and Application Roles, see "Managing Policies with Fusion Middleware Control" in *Oracle Fusion Middleware Security Guide*.

2.5.4.1 Modifying the Permission Grants for an Application Role

Use this procedure if you want to change the permission grants for an Application Role This is done by modifying the permissions grants for the Application Policy the Application Role is a Grantee of.

To add or remove permission grants from an Application Policy:

1. Log in to Fusion Middleware Control, navigate to **Security**, then select **Application Policies** to display the **Application Policies** page.

   For more information, see Section 2.5.1, "Starting Oracle Fusion Middleware Control and Locate the Pages for Managing Security".

   Whether or not the **obi** stripe is pre-selected and the Application Policies are displayed depends upon the method used to navigate to the **Application Policies** page.

2. If necessary, select **Select Application Stripe to Search**, then select **obi** from the list. Click the search icon next to **Role Name**.

   The Oracle Business Intelligence Application Policies are displayed. The **Principal** column displays the name of the policy **Grantee**.

3. Select the Application Role from the Principal column and click **Edit**.

4. Add or delete permissions from the **Edit Application Grant** view and click **OK** to save the changes.

2.5.4.2 Modifying Membership of an Application Role

Members can be added or deleted from an Application Role using Fusion Middleware Control. You must perform these tasks while in the WebLogic Domain that Oracle Business Intelligence is installed in. For example, bifoundation_domain. Valid members of an Application Role are users, groups, or other Application Roles. Being mapped to an Application Role is to become a member of an Application Role. Best practice is to map groups instead of individual users to Application Roles.

> **Note:** Be very careful when changing the permission grants and membership for the default Application Roles. For example, the BISystem Application Role provides the permissions required for system communication and changes to it could result in an unusable system.

To add or remove members from an Application Role:

1. Log in to Fusion Middleware Control, navigate to **Security**, then select **Application Roles** to display the **Application Roles** page.

   For information about navigating to the **Security** menu, see Section 2.5.1, "Starting Oracle Fusion Middleware Control and Locate the Pages for Managing Security".

   Whether or not the obi application stripe is pre-selected and the Application Policies are displayed depends upon the method used to navigate to the **Application Roles** page

2. If necessary, select **Select Application Stripe to Search**, then select the **obi** from the list. Click the search icon next to **Role Name**.

   The Oracle Business Intelligence Application Roles are displayed.

3. Select the cell next to the Application Role name and click **Edit** to display the **Edit Application Role** page.

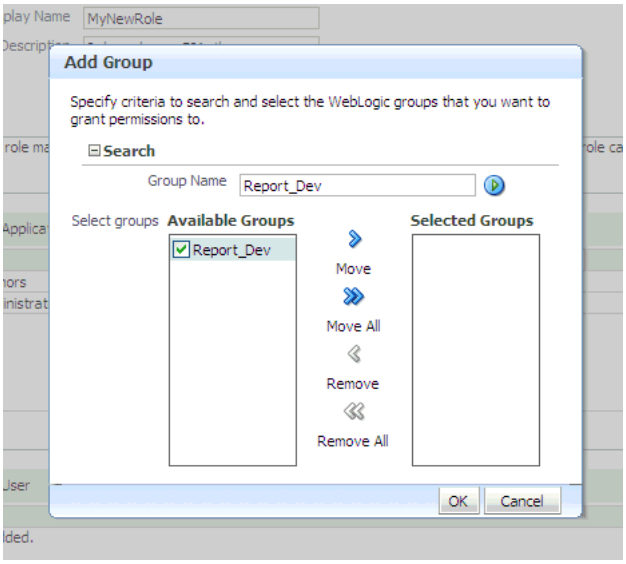
   You can add or delete members from the **Edit Application Role** page. Valid members are Application Roles, groups, and users.

4. From **Members**, select from the following options:

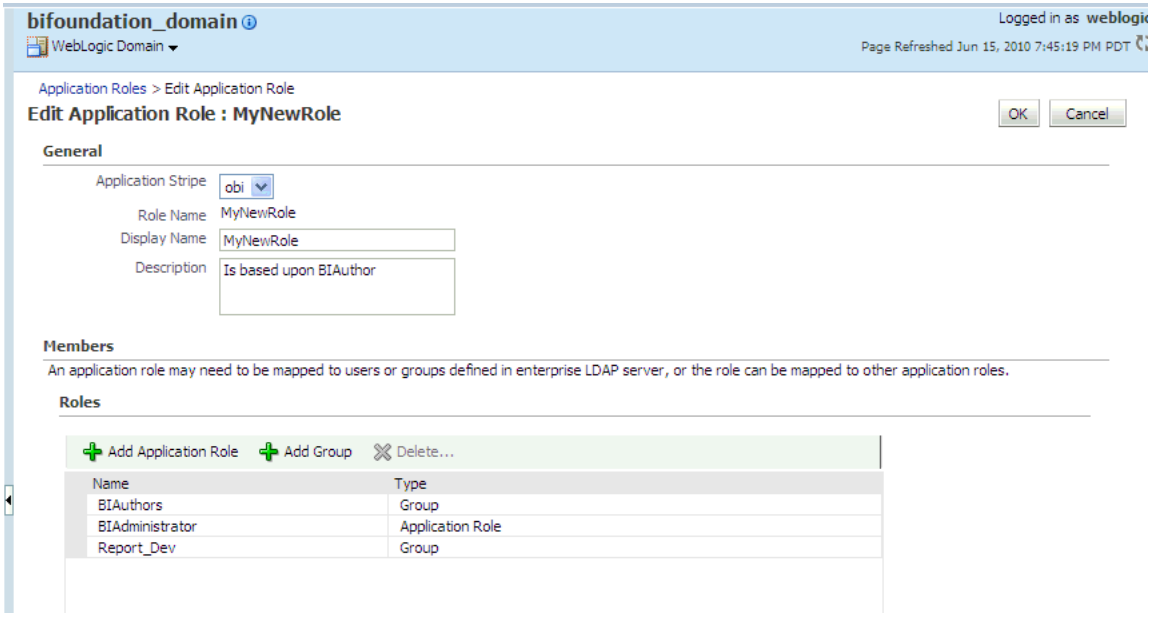
   - **To delete a member**: Select the **Name** of the member to activate the **Delete** button. Click **Delete**.
   - **To add a member**: Click the **Add** button that corresponds to the member type being added. Select from **Add Application Role**, **Add Group**, and **Add User**.

5. If adding a member, complete **Search** and select from the available list. Use the shuttle controls to move the member to the selected field. Click **OK**.

   For example, the following figure shows the **Add Group** dialog and after the **Report_Dev** group has been selected.



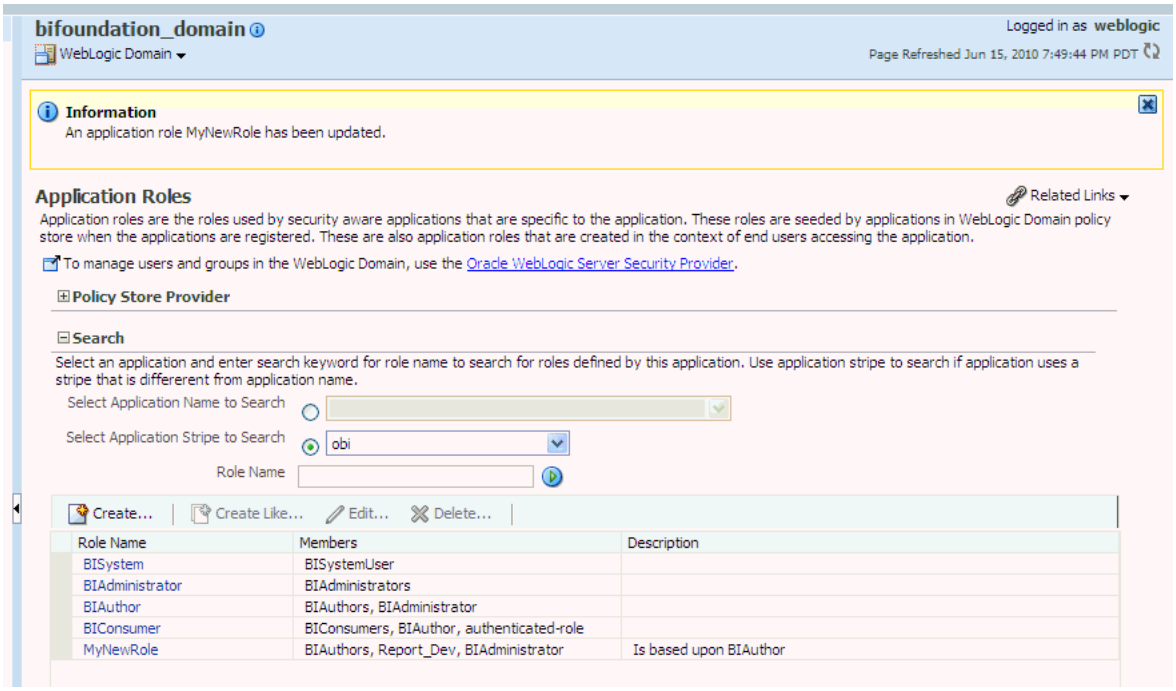
   The added member displays in the **Members** column corresponding to the Application Role modified in the **Application Roles** page. For example, the following figure shows the **Edit Application Role** page for the **MyNewRole** Application Role after the **Report_Dev** group has been added.



6. Click **OK** in the **Edit Application Role** page to return to the **Application Roles** page.

   The members just added to the Application Role display in the **Members** section. If members were deleted, they no longer display.

   The following figure shows the **MyNewRole** Application Role with the just added member **Report_Dev** group displaying.



For additional information, see "Managing Application Roles" in *Oracle Fusion Middleware Security Guide*.

2.6 Managing Metadata Repository Privileges

This section explains how to use Oracle BI Administration Tool to configure security in the metatdata repository (that is, the RPD file), and contains the following topics:

- Section 2.6.1, "Overview"
- Section 2.6.2, "How to Set Repository Privileges for an Application Role"
- Section 2.6.3, "Advanced Security Configuration Topics"

2.6.1 Overview

You use Security Manager in Oracle BI Administration Tool to manage permissions for Application Roles, and set access privileges for objects such as subject areas and tables. For an overview to using Oracle BI Administration Tool to configure security, see Section 1.7.3, "About Using Oracle BI Administration Tool".

Note: Oracle Business Intelligence Applications customers should read this section to understand the basics about security and setting up authentication, and then refer to the security and configuration information provided in the Oracle Business Intelligence Applications documentation.

2.6.2 How to Set Repository Privileges for an Application Role

The default Application Roles (that is, BIConsumer, BIAuthor, and BIAdministrator) are pre-configured with permissions for accessing the metadata repository. If you create a new Application Role, you must set appropriate repository permissions for the new Application Role, to enable that role to access the metadata repository (RPD).

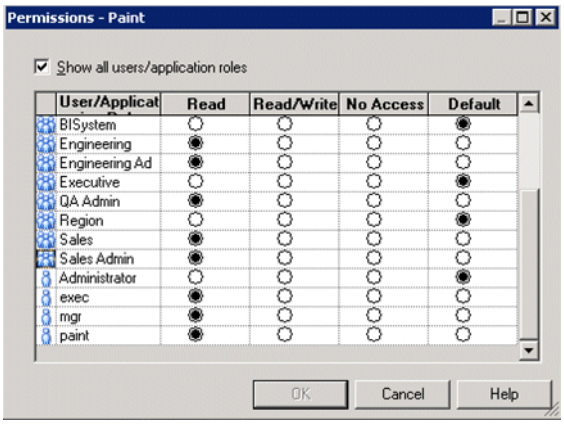
Note: In addition, you might map catalog privileges to a new Application Role in Presentation Catalog (for more information, see Section 2.7.3, "How to Set Catalog Privileges for an Application Role".

To set repository permissions for an Application Role:

1. Open the repository in the Oracle BI Administration Tool (in Online mode).
2. In the Presentation panel, navigate to the subject area or sub-folder for which you want to set permissions.
3. Right-click the subject area or sub-folder and choose Properties to display the properties dialog.

   For example, to provide access to the Paint subject area, right-click Paint.
4. Click Permissions to display the Permissions <*Name*> dialog.

   Note: Ensure that the **Show all users/application roles** check box is selected.



5. Use the Permissions <*Name*> dialog to change the security permissions for Application Roles in the **User/Application Role** list.

   For example, to enable Users to create dashboards and reports, you might change the repository permissions for an Application Role named BISalesAnalysis from 'Read' to 'Read/Write'.

   Note: Best practice is to modify permissions for Application Roles, not modify permissions for individual Users.

   Tip: To see all permissions for an object in the Presentation pane, right-click the object and choose Permission Report to display a list of Users and Application Roles and what permissions that have for the selected object.

2.6.3 Advanced Security Configuration Topics

This section contains advanced topics.

2.6.3.1 About Managing Application Roles in the Metadata Repository

Application Role definitions are maintained in the policy store and any changes must be made using the administrative interface. The repository maintains a *copy* of the policy store data to facilitate repository development. Oracle BI Administration Tool displays Application Role data from the repository's copy; you are not viewing the policy store data in real time. Policy store changes made while you are working with an offline repository are not available in the Administration Tool until the policy store next synchronizes with the repository. The policy store synchronizes data with the repository copy whenever BI Server restarts; if a mismatch in data is found, an error message is displayed.

While working with a repository in offline mode, you might discover that the available Application Roles do not satisfy the membership or permission grants needed at the time. A *placeholder for an Application Role* definition can be created in Administration Tool to facilitate offline repository development. But this is just a placeholder visible in Administration Tool and is not an actual Application Role. You cannot created an actual Application Role in Administration Tool. You can create an Application Role only in the policy store, using the administrative interface available for managing the policy store.

An Application Role must be defined in the policy store for each Application Role placeholder created using Administration Tool *before* bringing the repository back online. If a repository with role placeholders created while in offline mode is brought

online before valid Application Roles are created in the policy store, then the Application Role placeholder disappears from the Administration Tool interface. Always create a corresponding Application Role in the policy store before bringing the repository back online when using role placeholders in offline repository development.

For more information about how to create a placeholder for an Application Role during repository development, see *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

2.7 Managing Oracle BI Presentation Catalog Privileges Using Application Roles

This section explains how to manage Oracle BI Presentation Catalog privileges using Application Roles, and contains the following topics:

- Section 2.7.1, "Overview"
- Section 2.7.2, "About Oracle BI Presentation Catalog Privileges"
- Section 2.7.3, "How to Set Catalog Privileges for an Application Role"
- Section 2.7.4, "Advanced Security Configuration Topics"

2.7.1 Overview

BI Presentation Server uses Oracle BI Presentation Catalog privileges to control access to features such as Answers, Delivers, and BI Publisher. The default Oracle Business Intelligence Application Roles (BIAdministrator, BIAuthor, BIConsumer) are automatically configured with these privileges during installation, in addition to the Oracle Business Intelligence Application Policy permissions.

Systems upgraded from a previous release can continue to use catalog groups to grant these privileges, but this is not considered a best practice. Best practice is to use Application Roles to manage privileges, which streamlines the security management process. For example, using the same set of Application Roles throughout the system eliminates the need to manage a separate set of catalog groups and member lists. For more information regarding how to continue using upgraded catalog groups to manage Oracle BI Presentation Catalog privileges, see Section A.2.1, "Changes Affecting Security in Presentation Services".

Note: Mapping an Application Role to become a member of a catalog group creates complex group inheritance and maintenance situations and is not considered a best practice.

When Groups are mapped to Application Roles, the Group members are automatically granted associated Oracle BI Presentation Catalog privileges. This is in addition to the Oracle Business Intelligence permissions.

Tip: A list of Application Roles that a user is a member of is available from the Roles and Groups tab in the **My Account** dialog in Presentation Services.

2.7.2 About Oracle BI Presentation Catalog Privileges

Oracle BI Presentation Catalog privileges are maintained in BI Presentation Catalog. Presentation Services privileges control access only to Oracle BI Presentation Catalog features. These privileges grant or deny access rights to Presentation Services features and have no effect in other Oracle Business Intelligence components.

Being a member of a group mapped to a default Application Role grants Oracle BI Presentation Catalog privileges, in addition to the Oracle Business Intelligence permissions discussed in Section B.4.1.3, "Default Application Roles, Permission Grants, and Group Mappings". The Oracle BI Presentation Catalog privileges granted by a default Application Role can be modified by adding or removing default privilege grants using the **Manage Privileges** page.

Whenever a new catalog is created, it is populated with the default Application Role to Oracle BI Presentation Catalog privilege mappings. If you have changed the default mappings and want to see the default associations, create a new catalog by pointing to a file location where no catalog exists. When Oracle BI Presentation Server starts, a catalog is created as part of the initialization process.

Presentation Services privileges can be granted to users both explicitly and by inheritance. However, explicitly *denying* a Presentation Services privilege takes precedence over user access rights either granted or inherited as a result of group or Application Role hierarchy.

2.7.3 How to Set Catalog Privileges for an Application Role

If you create an Application Role, you must set appropriate privileges for the Application Role in the BI Presentation Catalog to enable that role to perform various functional tasks. For example, you might want Users with an Application Role named BISalesAdministrator to be able to create Actions in Oracle Business Intelligence. In this case, you would grant them a privilege named 'Create Invoke Action'.

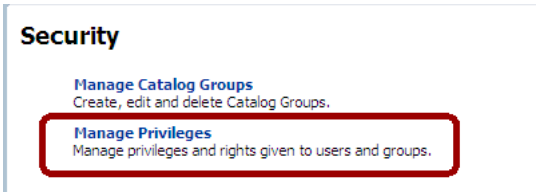
Oracle BI Presentation Catalog privileges are stored in the BI Presentation Server and cannot be accessed from the administrative interfaces used to manage the policy store. If you have created a new Application Role to grant Oracle Business Intelligence permissions, then you must the set Oracle BI Presentation Catalog privileges to that new role in addition to any Oracle Business Intelligence permissions.

Note: Oracle BI Presentation Catalog privileges can be mapped to a new Application Role programmatically using SecurityService Service. For more information, see "SecurityService Service" in *Oracle Fusion Middleware Integrator's Guide for Oracle Business Intelligence Enterprise Edition*

To set BI Presentation Catalog privileges for an Application Role:

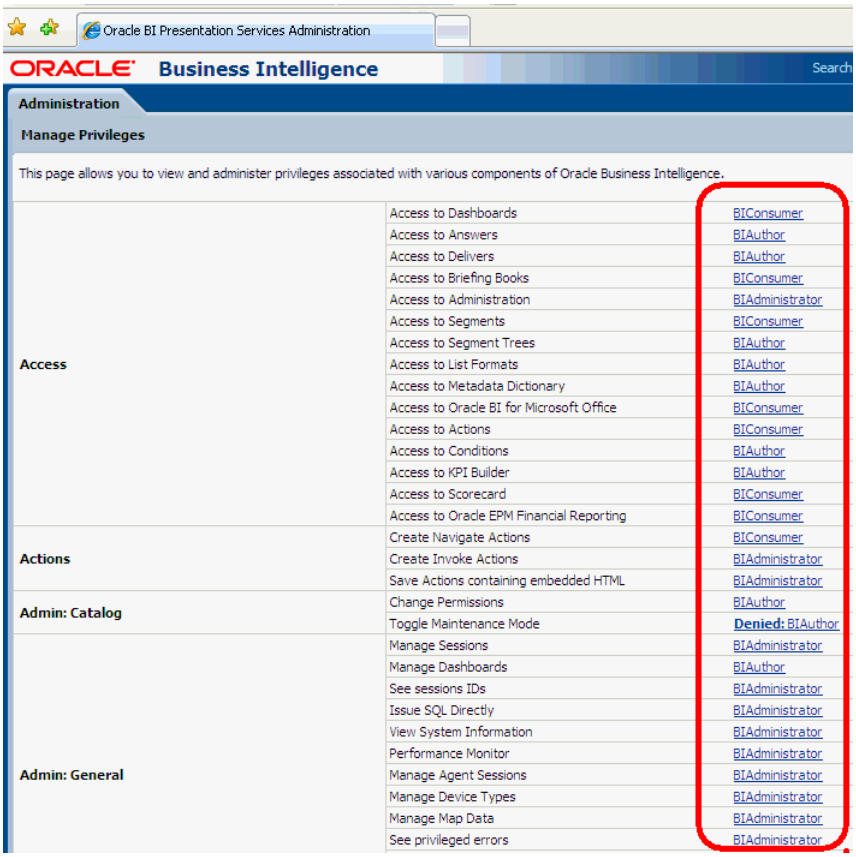
1. Log in to Oracle Business Intelligence as a User with Administrator privileges.
2. From the Home page in Presentation Services, select **Administration** to display the Administration page.

Note: If you log in as a User without Administrator privileges, the Administration option is not displayed.



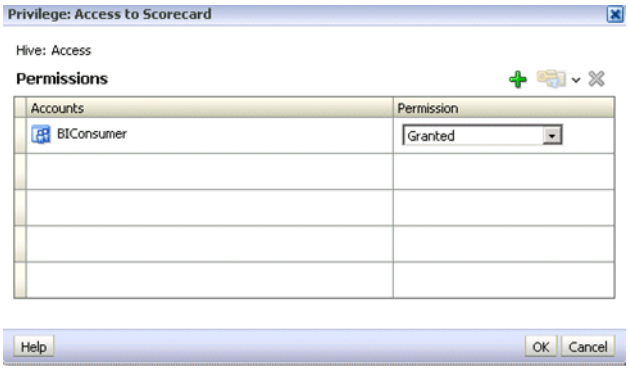
3. In the Security area, click **Manage Privileges** to display the Manage Privileges page.

   The screenshot below shows the Manage Privileges page with Application Roles highlighted for BI Presentation Catalog privileges.



4. Click an Application Role next to the privilege that you want to edit to display the Manage Privileges page.

   For example, to edit the privilege named 'Access to Scorecard' for the Application Role named BIConsumer, click the BIConsumer link next to Access\Access to Scorecard. The example screenshot below shows the Privilege dialog for the Access to Scorecard privilege.



Use the Privilege dialog to change permissions, grant privileges to Application Roles, and revoke privileges from an Application Role. For example, to grant the selected privilege to an Application Role, you must add the Application Role to the **Permissions** list.

5. To add an Application Role to the **Permissions** list, do the following:
   a. Click **Add Users/Roles**.
   b. Select **Application Roles** from the list and click **Search**.
   c. Select the Application Role from the results list.
   d. Use the shuttle controls to move the Application Role to the **Selected Members** list.
   e. Click **OK**.
6. Set the permission for the Application Role by selecting **Granted** or **Denied** in the **Permission** drop down list.

   Note: Explicitly *denying* a Presentation Services privilege takes precedence over user access rights either granted or inherited as a result of Group or Application Role hierarchy.
7. Save your changes.

Note: Existing catalog groups are migrated during the upgrade process. Moving an existing Oracle BI Presentation Catalog security configuration to the role-based Oracle Fusion Middleware security model based requires that each catalog group be replaced with a corresponding Application Role. To duplicate an existing Presentation Services configuration, replace each catalog group with a corresponding Application Role that grants the same Oracle BI Presentation Catalog privileges. You can then delete the original catalog group from Presentation Services.

2.7.4 Advanced Security Configuration Topics

This section contains advanced topics.

2.7.4.1 About Encryption in BI Presentation Services

The Oracle BI Server and Oracle BI Presentation Services client support industry-standard security for login and password encryption. When an end user enters a user name and password in the Web browser, the Oracle BI Server uses the Hyper Text Transport Protocol Secure (HTTPS) standard to send the information to a

secure Oracle BI Presentation Services port. From Oracle BI Presentation Services, the information is passed through ODBC to the Oracle BI Server, using Triple DES (Data Encryption Standard). This provides a high level of security (168 bit), preventing unauthorized users from accessing data or Oracle Business Intelligence metadata.

At the database level, Oracle Business Intelligence administrative users can implement database security and authentication. Finally, a proprietary key-based encryption provides security to prevent unauthorized users from accessing the metadata repository.

3

Configuring Oracle BI to use Oracle Internet Directory

This chapter explains how Oracle Business Intelligence can be configured to use commercial directory servers for authentication. It covers configuring Oracle Business Intelligence to use OID for authentication, and configuring Oracle Business Intelligence to use OID as a policy store and credential store.

> **Note:** For a detailed list of security setup steps, see Section 1.8, "Detailed List of Steps for Setting Up Security In Oracle Business Intelligence".

This chapter contains the following sections:

- Section 3.1, "Common Tasks for Deploying Security With Oracle Internet Directory"
- Section 3.2, "Configuring an Alternative Authentication Provider"
- Section 3.3, "Configuring an Alternative Policy Store and Credentials Store"

3.1 Common Tasks for Deploying Security With Oracle Internet Directory

Table 3–1 contains common authorization configuration tasks and provides links for more information.

Table 3–1 Task Map: Configuring Authorization for Oracle Business Intelligence

Task	Description	Information
Re-configure Oracle BI to use an LDAP based Authentication Provider.	Re-configure Oracle BI to use an LDAP based Authentication Provider, such as Oracle Internet Directory.	Section 3.3, "Configuring an Alternative Policy Store and Credentials Store"
Re-configure Oracle BI to use an LDAP based Credential Store and Policy Store Provider.	Re-configure Oracle BI to use an LDAP based Credential Store and Policy Store Provider, such as Oracle Internet Directory.	Section 3.3, "Configuring an Alternative Policy Store and Credentials Store"

3.2 Configuring an Alternative Authentication Provider

When you use OID as the Authentication Provider, you use OID Console to set up your Users and Groups. You can then map these Users and Groups to the

preconfigured Application Roles (for example, BIConsumer, BIAuthors, and BIAdministrator), and any additional Application Roles that you create. For more information about mapping Users and Groups to Application Roles, see Section 2.5, "Managing Application Roles and Application Policies Using Fusion Middleware Control".

You continue to use the other Oracle Business Intelligence tools (i.e. BI Administration Tool, Enterprise Manager Fusion Middleware Control, and Administration Page in Oracle BI Presentation Catalog) to manage the other areas of the security model.

For a current list of supported authentication providers and directory servers to use with Oracle Business Intelligence, see the system requirements and certification documentation. For more information, see System Requirements and Certification.

If a directory server other than the default the Embedded WebLogic LDAP Server is being used, you can view the users and groups from that directory server in Oracle WebLogic Server Administration Console. However, you must continue to manage the users and groups in the interface for the directory server being used. For example, if you are using OID, you must use OID Console to create and edit Users and Groups.

3.2.1 How to Configure Oracle Internet Directory as an Authentication Store Provider

To configure OID as an Authentication Store Provider, do the following:

Prerequisite: Shut down all servers except Admin Server.

1. Configure Oracle Internet Directory as an authentication provider as described in Section 3.2.1.1, "How to Configure Oracle Business Intelligence to use Oracle Internet Directory as an Authentication Provider".
2. Configure the User Name Attribute in the Identity Store to match the User Name Attribute in the Authentication Provider as described in Section 3.2.1.2, "How to Configure the User Name Attribute in the Identity Store".
3. Use the myrealm\Users and Groups tab to verify that the Users and Group from OID are displayed correctly. If the Users and Groups are displayed correctly, then proceed to Step 4. Otherwise, re-set your configuration settings and re-try.
4. Configure a new BISystemUser account for a user in Oracle Internet Directory to match the account for DefaultAuthenticator as described in Section 3.2.1.3, "Configure a New Trusted User (BISystemUser)".
5. Update the user GUIDs to be the values in Oracle Internet Directory as described in Section 3.2.1.4, "Refresh the User GUIDs".

Notes

- After a new authentication provider is configured, Application Roles must be mapped again to the correct Groups (enterprise roles) in the new identity store.

  For more information, see Section 2.5.4.2, "Modifying Membership of an Application Role".

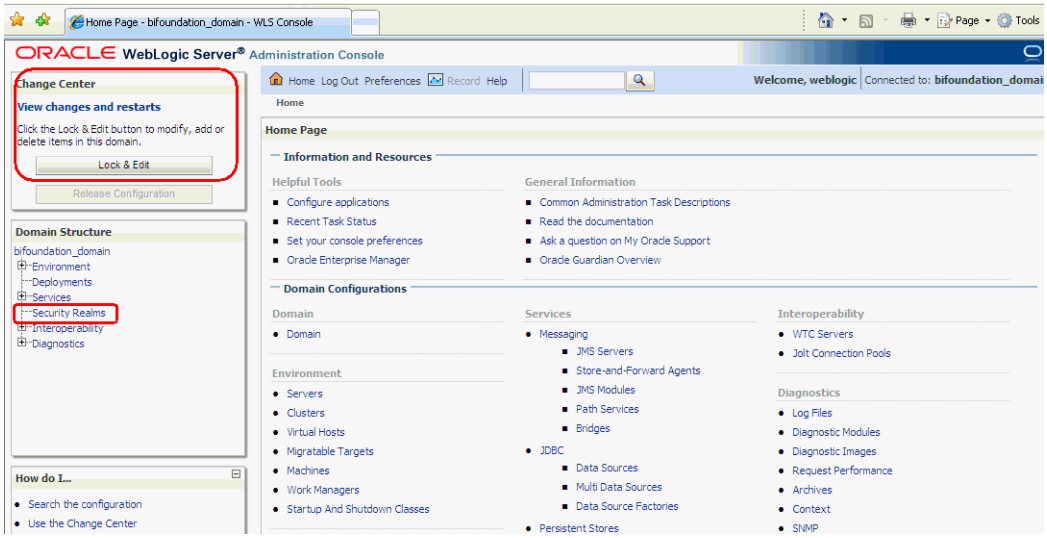
3.2.1.1 How to Configure Oracle Business Intelligence to use Oracle Internet Directory as an Authentication Provider

You perform this task to reconfigure your installation to use OID instead of the default Oracle WebLogic Administration Server.

To configure the OID authentication provider:

Note: MyOIDDirectory is used to represent the Oracle Internet Directory in the following procedure.

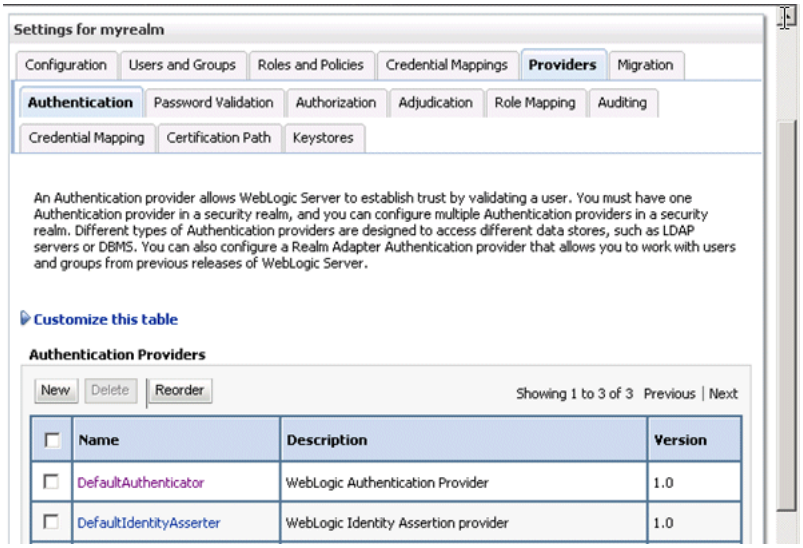
1. In Oracle WebLogic Server Administration Console, click **Lock & Edit** in the Change Center.



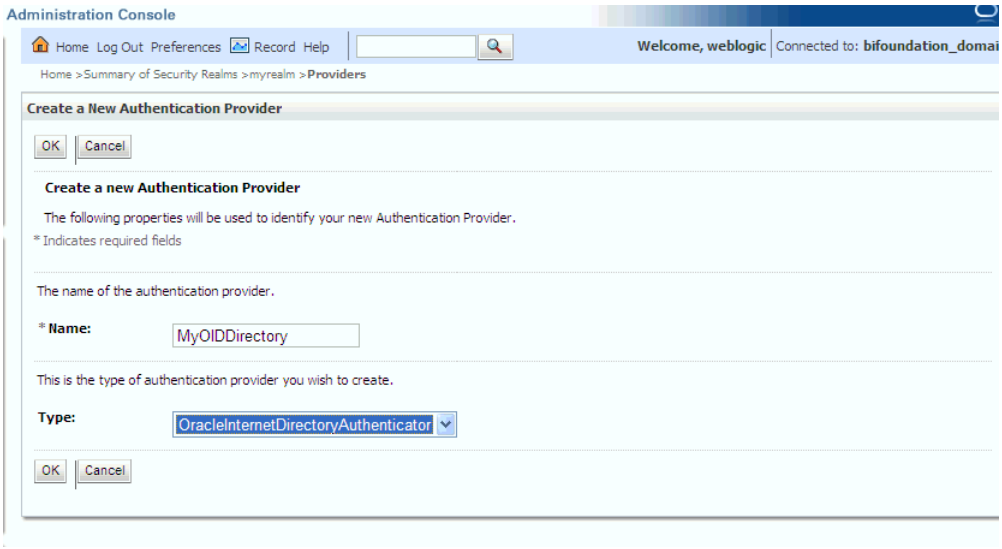
2. Select **Security Realms** from the left pane and click **myrealm**.

   The default Security Realm is named **myrealm**.

3. Display the **Providers** tab, then display the **Authentication** sub-tab.



4. Click **New** to launch the **Create a New Authentication Provider** page.

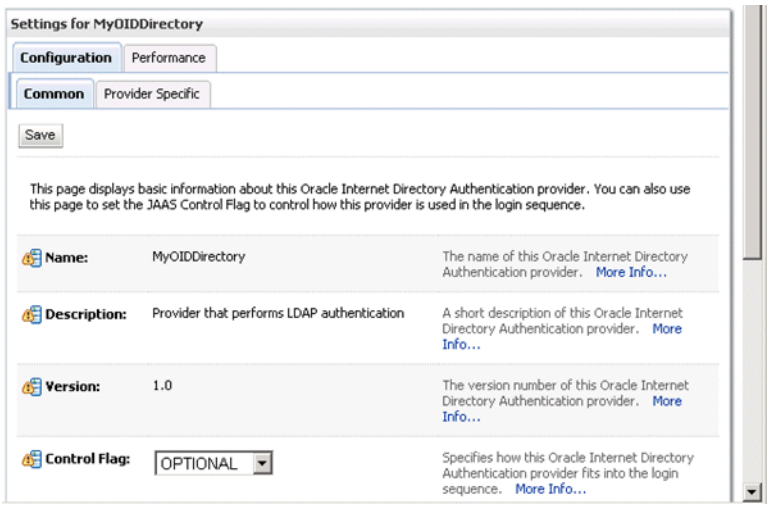


5. Enter values in the **Create a New Authentication Provider** page as follows:

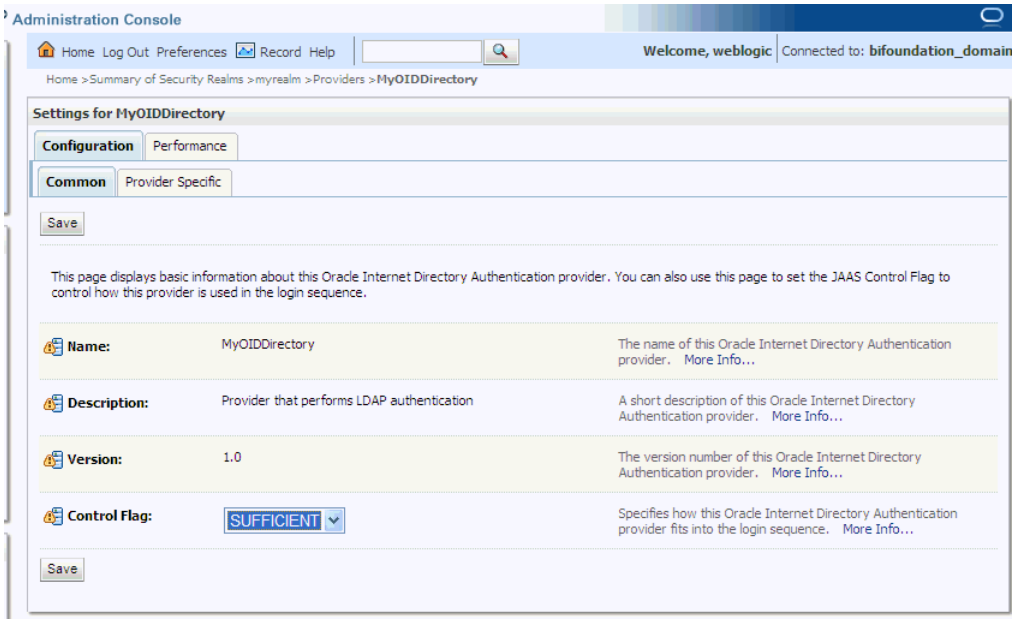
   - **Name**: Enter a name for the authentication provider. For example, MyOIDDirectory.
   - **Type**: Select OracleInternetDirectoryAuthenticator from the list.
   - Click **OK** to save the changes and display the Authentication Providers list updated with the new Authentication Provider.

6. Click the new Authenticator Provider in the **Name** column to display the Settings for <*Authentication Provider Name*> page.

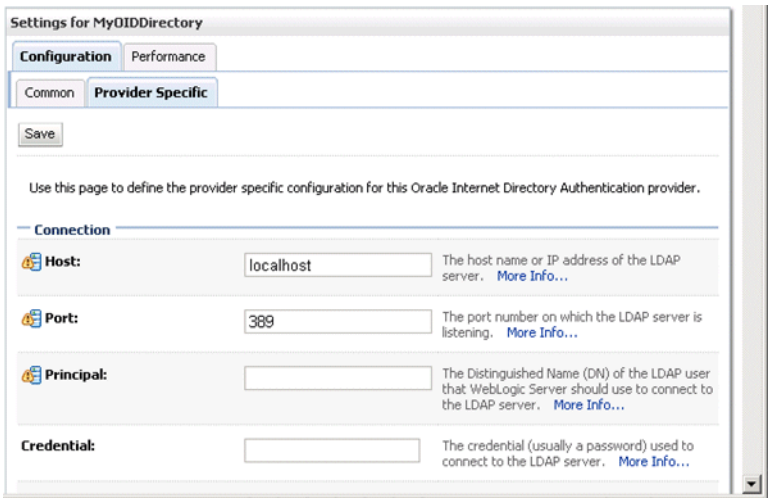
   For example, click MyOIDDirectory.



7. Display the **Configuration****Common** tab, and use the **Control Flag** drop down list to select 'SUFFICIENT', then click Save.



8. Display the **Provider Specific** tab.

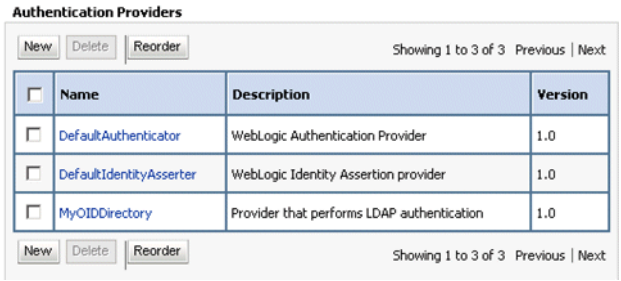


9. Use the Provider Specific tab to specify the details listed in the table below.

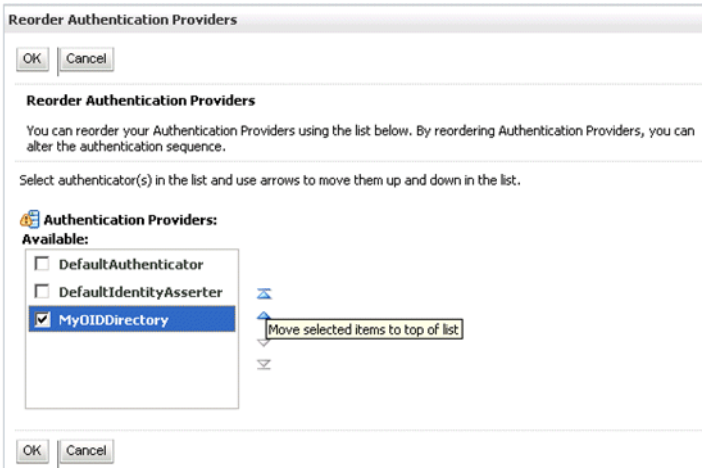
Section Name	Field Name	Description
Connection	Host	The host name of the Oracle Internet Directory server.
	Port	The port number on which the Oracle Internet Directory server is listening.
	Principal	The distinquished name (DN) of the Oracle Internet Directory user to be used to connect to the Oracle Internet Directory server. For example: cn=OIDUser,cn=users,dc=us,dc=mycompany,dc=com.
	Credential	Password for the Oracle Internet Directory user entered as the Principal.
Groups	Group Base DN	The base distinguished name (DN) of the Oracle Internet Directory server tree that contains groups.
Users	User Base DN	The base distinguished name (DN) of the Oracle Internet Directory server tree that contains users.
	All Users Filter	LDAP search filter. Click More Info... for details.
	User From Name Filter	LDAP search filter. Click More Info... for details.
	User Name Attribute	The attribute that you want to use to authenticate (for example, cn, uid, or mail). For example, to authenticate using a User's email address you set this value to 'mail'. **Note**: The value that you specify here must match the User Name Attribute that you are using in the Authentication Provider, as described in the next task Section 3.2.1.2, "How to Configure the User Name Attribute in the Identity Store".

The screenshot below shows the Users area of the Provider Specific tab.

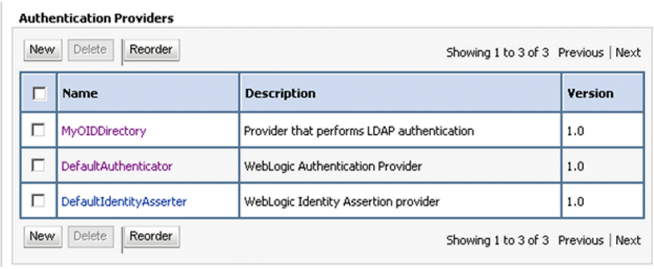
10. Click **Save**.
11. At the main **Settings for myrealm** page, display the **Providers** tab, then display the **Authentication** sub-tab.



12. Click **Reorder**. to display the Reorder Authentication Providers page.
13. Select the name of the Oracle Internet Directory authentication provider (for example, MyOIDDirectory) and use the arrow buttons to move it into the first position in the list, then click OK.



The screenshot below shows the re-ordered list of Authentication Providers.



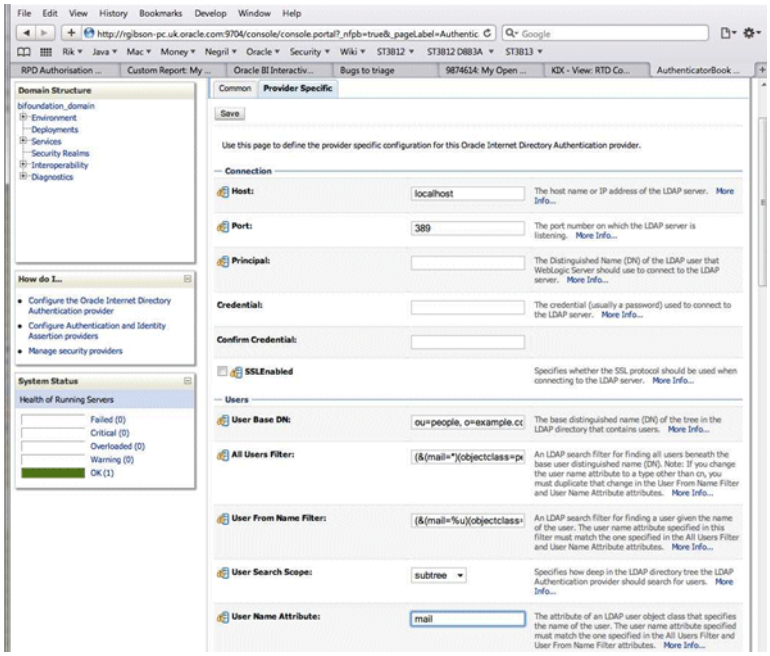
14. Click DefaultAuthenticator in the **Name** column to display the **Settings for DefaultAuthenticator** page.
15. Display the **Configuration****Common** tab, and use the **Control Flag** drop down list to select 'SUFFICIENT', then click Save.

3.2.1.2 How to Configure the User Name Attribute in the Identity Store

If you configure a different Authentication Provider such as OID, then you must ensure that the User Name Attribute that you use in the Identity Store matches the User Name Attribute that you use in the Authentication Provider.

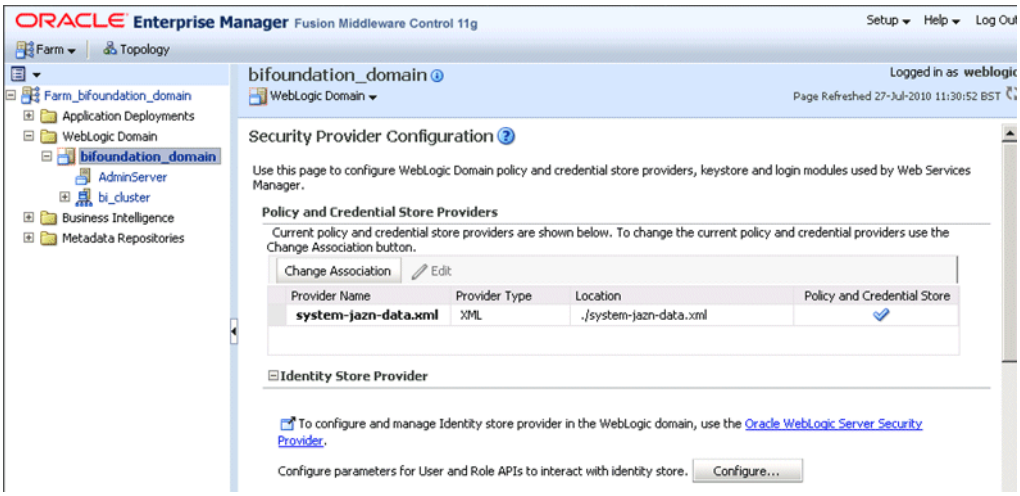
For example, to authenticate using a User's email address you might set the User Name Attribute to 'mail' in both the Identity Store and the Authentication Provider.

The screenshot below shows an example where the **User Name Attribute** in OID Authenticator has been set to 'mail'.

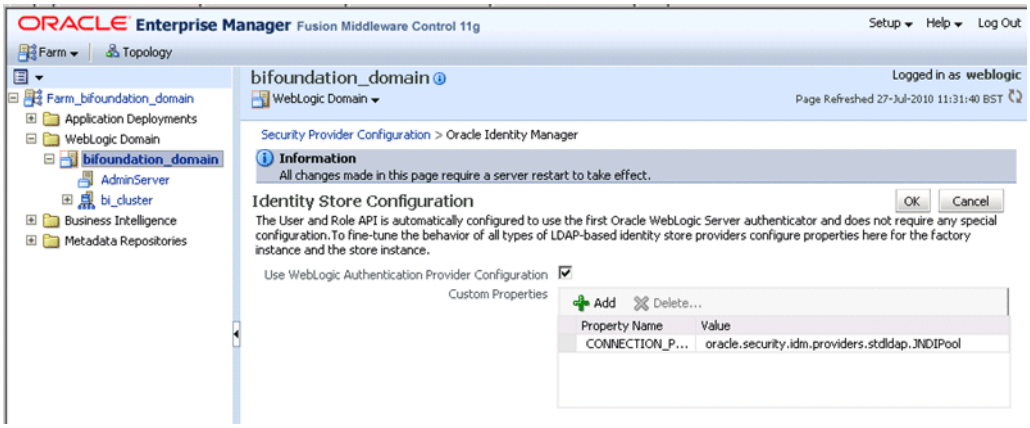


To configure the User Name Attribute:

1. In Oracle Enterprise Manager - Fusion Middleware Control, navigate to \Weblogic domain\bifoundation_domain in the navigation pane.
2. Right-click bifoundation_domain and select Security, then Security Provider Configuration to display the Security Provider Configuration page.



3. In the Identity Store Provider area, click Configure to display the Identity Store Configuration page.

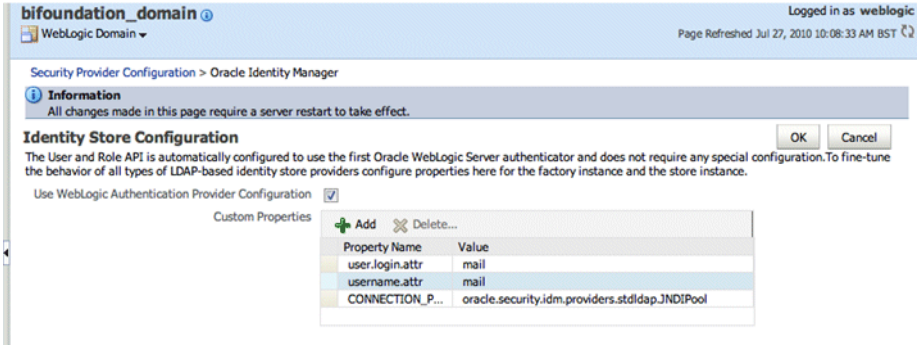


4. In the Custom Properties area, use the Add option to add the following two Custom Properties:

Table 3–2 Custom Properties

Property Name	Value
user.login.attr	Specify the User Name Attribute that is set in the Authentication Provider. For example, if the User Name Attribute is set to 'mail' in the Authentication Provider, then set this value to 'mail'.
username.attr	Specify the User Name Attribute that is set in the Authentication Provider. For example, if the User Name Attribute is set to 'mail' in the Authentication Provider, then set this value to 'mail'.

The screenshot below shows an example set of Custom Properties with the User Name Attribute set to 'mail'.



5. Click OK to save the changes.
6. Restart the Admin Server.

Note: Ensure that the Users and Groups in OID are displayed in WebLogic Console, as described in step 3 in Section 3.2.1, "How to Configure Oracle Internet Directory as an Authentication Store Provider".

3.2.1.3 Configure a New Trusted User (BISystemUser)

Oracle Business Intelligence uses a specific user for the configured authentication provider for internal communication. If you configure Oracle BI to use an OID authentication provider, then you must select a user from OID to use for this purpose and give that user the required permissions. You can create a new user in OID for this

purpose or use a pre-existing user. You give the chosen user the permission they need by making them a member of the pre-existing BISystem Application Role.

Pre-requisite: Delete the default 'BISystemUser' from the realm **myrealm**. Display the **Settings for myrealm** page, display the Users and Groups tab, then the Users sub-tab, and delete the user named BISystemUser with DefaultAuthenticator in the Provider column of the table (see example screenshot below).

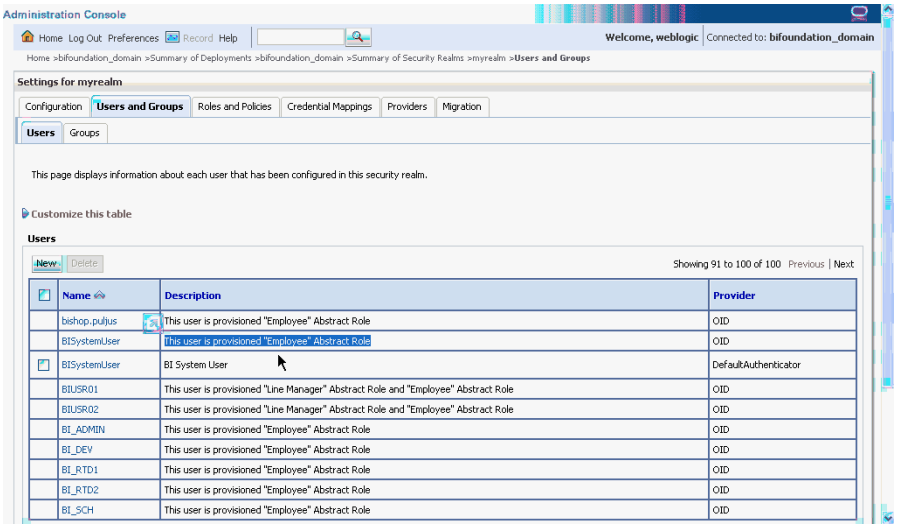
☑	BISystemUser	BI System User	DefaultAuthenticator
	cieuser		OVD/AD2008
	dadvmc023		OVD/AD2008
	dadvmc0237		OVD/AD2008
	devsrvsppt		OVD/AD2008

To create a new trusted user account with a user in OID:

1. In Oracle Internet Directory, create a user for the trusted user.

   Best practice is to name this trusted user **BISystemUser** to clarify its purpose, but you might choose any name you want.

   When you are finished, the **Users** table in Oracle WebLogic Server Administration Console should resemble the screenshot below.



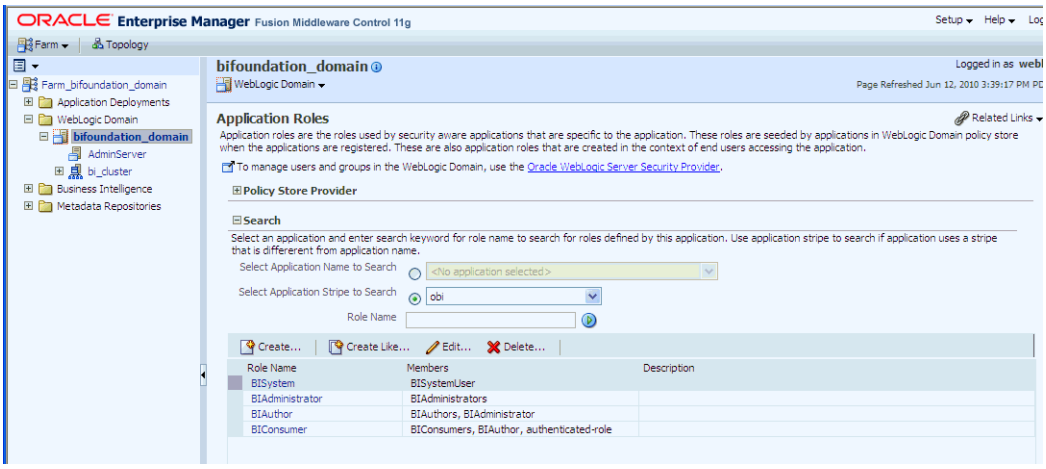
   Next you must make the new trusted user a member of the BISystem Application Role.

2. In Fusion Middleware Control target navigation pane, go to the Oracle WebLogic Server domain in which Oracle Business Intelligence is installed. For example, bifoundation_domain.

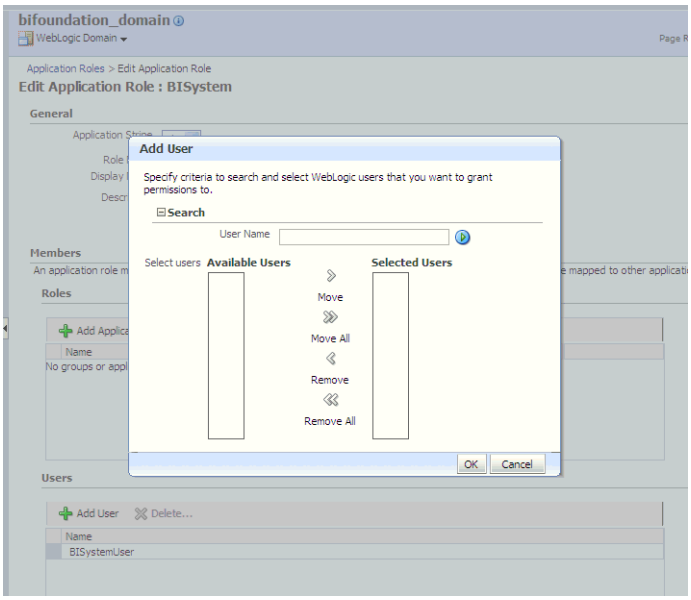
3. Go to the **Application Roles** page in Fusion Middleware Control.

4. In the **Select Application Stripe to Search** list, select **obi** from the list. Click the search arrow to the right of the **Role Name** field.

   The Oracle Business Intelligence Application Roles are displayed and should resemble the screenshot below.



5. Select the **BISystem** Application Role and click **Edit**.
6. In the **Edit Application Role** page, click **Add User**.
7. In the **Add User** dialog, search for the trusted user created in Oracle Internet Directory. Use the shuttle controls to move the trusted user name (**BISystemUser**) from the **Available Users** list to the **Selected Users** list.

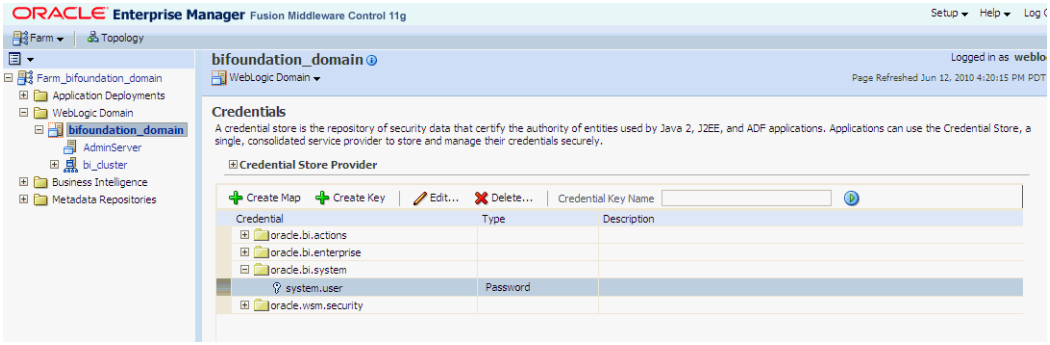


8. Click **OK**.

   The trusted user (**BISystemUser**) contained in Oracle Internet Directory is now a member of the BISystem Application Role.

   Next add the trusted user's credentials to the **oracle.bi.system** credential map.

9. From Fusion Middleware Control target navigation pane, expand the farm, then expand **WebLogic Domain**, and select **bifoundation_domain**.
   - From the WebLogic Domain menu, select **Security**, then **Credentials**.
   - Open the **oracle.bi.system** credential map, select **system.user** and click **Edit**.



- In the **Edit Key** dialog, enter **BISystemUser** (or name you selected) in the **User Name** field. In the **Password** field, enter the trusted user's password that is contained in Oracle Internet Directory.
- Click **OK**.

10. In WebLogic Console, click myrealm to display the Settings for <*Realm*> page, display the Roles and Policies tab, and add the new System user to the Global 'Admin Role'.
11. Start the Managed Servers.

The new trusted user from Oracle Internet Directory is configured for Oracle Business Intelligence

3.2.1.4 Refresh the User GUIDs

If you change the directory server used as the identity store for the authentication provider, then you must refresh the user GUIDs as described below. If you do not refresh the GUIDs and the same user name exists in both directory servers (original and new), then the original user GUID might conflict with the user GUID contained in new directory server, resulting in authentication errors.

To refresh the user GUIDs:

This task requires that you manually edit the configuration files to instruct Oracle BI Server and Oracle BI Presentation Server to refresh the GUIDs on restart. Once completed, you edit these files to remove the modification. For information about where to locate Oracle Business Intelligence configuration files, see "Where Configuration Files are Located" in *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*.

1. Update the `FMW_UPDATE_ROLE_AND_USER_REF_GUIDS` parameter in NQSConfig.INI:
   a. Open NQSConfig.INI for editing at:

      ORACLE_INSTANCE`/config/OracleBIServerComponent/coreapplication_obisn`

   b. Locate the `FMW_UPDATE_ROLE_AND_USER_REF_GUIDS` parameter and set it to `YES`, as follows:

      ```
      FMW_UPDATE_ROLE_AND_USER_REF_GUIDS = YES;
      ```

   c. Save and close the file.
2. Update the ps:Catalog element in instanceconfig.xml:
   a. Open instanceconfig.xml for editing at:

      ORACLE_INSTANCE`/config/OracleBIPresentationServicesComponent/`

```
coreapplication_obipsn
```

b. Locate the ps:Catalog element and update it as follows:

```
<ps:Catalog xmlns:ps="oracle.bi.presentation.services/config/v1.1">
<ps:UpgradeAndExit>false</ps:UpgradeAndExit>
<ps:UpdateAccountGUIDs>UpdateAndExit</ps:UpdateAccountGUIDs>
</ps:Catalog>
```

c. Save and close the file.

3. Start the Oracle Business Intelligence system components using opmnctl:

```
cd ORACLE_BASE/admin/instancen/bin
./opmnctl startall
```

4. Set the `FMW_UPDATE_ROLE_AND_USER_REF_GUIDS` parameter in NQSConfig.INI back to `NO`.

   Important: You must perform this step to ensure that your system is secure.

5. Update the ps:Catalog element in instanceconfig.xml to remove the <ps:UpdateAccount GUIDs> entry.

6. Restart the Oracle Business Intelligence system components using opmnctl:

```
cd ORACLE_BASE/admin/instancen/bin
./opmnctl stopall
./opmnctl startall
```

3.3 Configuring an Alternative Policy Store and Credentials Store

To re-configure Oracle Business Intelligence to use OID as a Credential Store and Policy Store Provider, follow the steps in Section 8.2 Reassociating the Domain Policy Store in *Oracle Fusion Middleware Security Guide*.

Notes

- The only LDAP server supported in this release is Oracle Internet Directory. The pre-requisites for using an LDAP-based credential store are the same as for using an LDAP-based policy store. For more information, see "Configuring a Domain to Use an LDAP-Based Policy Store" in *Oracle Fusion Middleware Security Guide*.

4 Enabling SSO Authentication

This chapter provides some general guidelines for configuring single sign-on (SSO) authentication for Oracle Business Intelligence.

> **Note:** For a detailed list of security setup steps, see Section 1.8, "Detailed List of Steps for Setting Up Security In Oracle Business Intelligence".

This chapter contains the following topics:

- Section 4.1, "Common SSO Configuration Tasks for Oracle Business Intelligence"
- Section 4.2, "Understanding SSO Authentication and Oracle Business Intelligence"
- Section 4.3, "SSO Implementation Considerations"
- Section 4.4, "Configuring SSO in an Oracle Access Manager Environment"

> **Note:** Oracle recommends using Oracle Access Manager as an enterprise-level SSO authentication provider with Oracle Fusion Middleware 11*g*. This chapter assumes that Oracle Access Manager is the SSO authentication provider being used unless stated otherwise. For more information about configuring and managing Oracle Access Manager with Oracle Fusion Middleware, see "Configuring Single Sign-On in Oracle Fusion Middleware" in *Oracle Fusion Middleware Security Guide*.

4.1 Common SSO Configuration Tasks for Oracle Business Intelligence

Table 4–1 contains common authentication configuration tasks and provides links for obtaining more information.

Table 4–1 Task Map: Configuring SSO Authentication for Oracle Business Intelligence

Task	Description	For More Information
Configure the SSO authentication provider.	Configure Oracle Access Manager to protect the Oracle Business Intelligence URL entry points.	Section 4.4, "Configuring SSO in an Oracle Access Manager Environment" "Configuring Single Sign-On in Oracle Fusion Middleware" in *Oracle Fusion Middleware Security Guide*

Table 4–1 (Cont.) Task Map: Configuring SSO Authentication for Oracle Business

Task	Description	For More Information
Configure HTTP proxy.	Configure the web proxy to forward requests from Oracle BI Presentation Server to the SSO provider.	"Configuring Single Sign-On in Oracle Fusion Middleware" in *Oracle Fusion Middleware Security Guide*
Configure a new authenticator for Oracle WebLogic Server.	Configure the Oracle WebLogic Server domain in which Oracle Business Intelligence is installed to use the new identity store.	Section 4.4.1, "Configuring a New Authenticator for Oracle WebLogic Server" Section 3.2, "Configuring an Alternative Authentication Provider" *Oracle Fusion Middleware Oracle WebLogic Server Administration Console Online Help*
Configure a new identity asserter for Oracle WebLogic Server.	Configure the Oracle WebLogic Server domain in which Oracle Business Intelligence is installed to use the SSO provider as an asserter.	Section 4.4.2, "Configuring a New Identity Asserter for Oracle WebLogic Server" Section 3.2, "Configuring an Alternative Authentication Provider" *Oracle Fusion Middleware Oracle WebLogic Server Administration Console Online Help*
Configure the new trusted system user to replace the default BISystemUser.	Add the new trusted system user name from Oracle Internet Directory to become a member of the BISystem Application Role.	Section 3.2, "Configuring an Alternative Authentication Provider" Section 3.2.1.3, "Configure a New Trusted User (BISystemUser)"
Refresh the user and group GUIDs.	Refresh the GUIDs of users and groups who migrated from the original identity store to the new identity store (authentication source).	Section 3.2.1.4, "Refresh the User GUIDs"
Enable Oracle Business Intelligence to accept SSO authentication.	Enable the SSO provider configured to work with Oracle Business Intelligence using Fusion Middleware Control.	Section 4.4.3, "Using Fusion Middleware Control to Enable SSO Authentication"

Note: For an example of an Oracle Business Intelligence SSO installation scenario, see *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Business Intelligence*.

4.2 Understanding SSO Authentication and Oracle Business Intelligence

Integrating a single sign-on (SSO) solution enables a user to log on (sign-on) and be authenticated once per browser session. Thereafter, the authenticated user is given access to system components or resources according to the permissions and privileges granted to that user. Oracle Business Intelligence can be configured to trust incoming HTTP requests authenticated by a SSO solution that is configured for use with Oracle Fusion Middleware and Oracle WebLogic Server. For more information about

configuring SSO for Oracle Fusion Middleware, see "Configuring Single Sign-On in Oracle Fusion Middleware" in *Oracle Fusion Middleware Security Guide*.

When Oracle Business Intelligence is configured to use SSO authentication, it accepts authenticated users from whatever SSO solution Oracle Fusion Middleware is configured to use. If SSO is not enabled, then Oracle Business Intelligence challenges each user for authentication credentials. When Oracle Business Intelligence is configured to use SSO, a user is first redirected to the SSO solution's login page for authentication. After the user is authenticated the SSO solution forwards the user name to Oracle BI Presentation Services where this name is extracted. Next a session with the Oracle BI Server is established using the impersonation feature.

After a successful logon using SSO, users are still required to have the oracle.bi.server.manageRepositories permission to log in to Administration Tool using a valid user name and password combination. After installation, the oracle.bi.server.manageRepositories permission is granted by being a member of the default BIAdministration Application Role.

Configuring Oracle Business Intelligence to work with SSO authentication requires minimally that the following be done:

- Oracle Fusion Middleware and Oracle WebLogic Server is configured to accept SSO authentication. Oracle Access Manager is recommended in production environments.
- Oracle BI Presentation Services is configured to trust incoming messages.
- The HTTP header information required for identity propagation with SSO configurations (namely, user identity and SSO cookie) is specified and configured.

4.2.1 How an Identity Asserter Works

How an identity asserter works is described using Oracle Access Manager Identity Asserter for single sign-on. The Oracle Access Manager authentication provider works with Oracle WebLogic Server and provides the following features:

- **Identity Asserter for Single Sign-on**

  This feature uses the Oracle Access Manager authentication services and validates already-authenticated Oracle Access Manager users through the ObSSOCookie and creates a WebLogic-authenticated session. It also provides single sign-on between WebGates and portals. WebGate is a plug-in that intercepts Web resource (HTTP) requests and forwards them to the Access Server for authentication and authorization.

- **Authenticator**

  This feature uses Oracle Access Manager authentication services to authenticate users who access an application deployed in Oracle WebLogic Server. Users are authenticated based on their credentials, for example a user name and password.

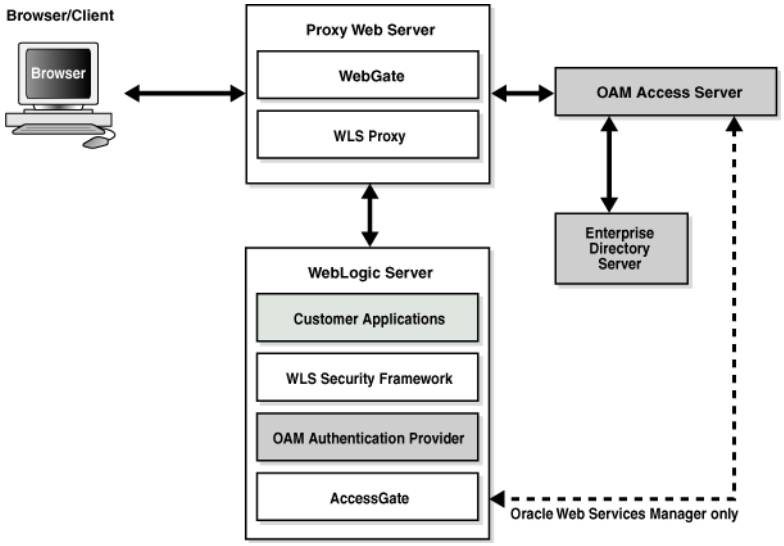
After the authentication provider for Oracle Access Manager is configured as the Identity Asserter for single sign-on, the Web resources are protected. Perimeter authentication is performed by WebGate on the web tier and by the ObSSOCookie to assert the identity of users who attempt access to the protected WebLogic resources.

All access requests are routed to a reverse proxy Web server. These requests are in turn intercepted by WebGate. The user is challenged for credentials based on the authentication scheme configured within Oracle Access Manager (form-based login recommended).

After successful authentication, WebGate generates an ObSSOCookie and the Web server forwards the request to Oracle WebLogic Server, which in turn invokes Oracle Access Manager Identity Asserter for single sign-on validation. The WebLogic Security Service invokes Oracle Access Manager Identity Asserter for single sign-on, which next gets the ObSSOCookie from the incoming request and populates the subject with the WLSUserImpl principal. The Identity Asserter for single sign-on adds the WLSGroupImpl principal corresponding to the groups the user is a member of. Oracle Access Manager then validates the cookie.

Figure 4–1 depicts the distribution of components and the flow of information when the Oracle Access Manager Authentication Provider is configured as an Identity Asserter for SSO with Oracle Fusion Middleware.

Figure 4–1 Oracle Access Manager Single Sign-On Solution for Web Resources Only



4.2.2 How Oracle Business Intelligence Operates With SSO Authentication

After SSO authorization has been implemented, Oracle BI Presentation Services operates as if the incoming web request is from a user authenticated by the SSO solution. Oracle BI Presentation Services next creates a connection to the Oracle BI Server using the impersonation feature and establishes the connection to the Oracle BI Server on behalf of the user. User personalization and access controls such as data-level security are maintained in this environment.

4.3 SSO Implementation Considerations

When implementing a SSO solution with Oracle Business Intelligence you should consider the following:

- When accepting trusted information from the HTTP server or servlet container, it is essential to secure the machines that communicate directly with the Oracle BI Presentation Server. This can be done by setting the Listener\Firewall node in the instanceconfig.xml file with the list of HTTP Server or servlet container IP addresses. Additionally, the Firewall node must include the IP addresses of all Oracle BI Scheduler instances, Oracle BI Presentation Services Plug-in instances and Oracle BI Javahost instances. If any of these components are co-located with Oracle BI Presentation Services, then address 127.0.0.1 must be added in this list as well. This setting does not control end-user browser IP addresses.

- When using mutually-authenticated SSL, you must specify the Distinguished Names (DNs) of all trusted hosts in the Listener\TrustedPeers node.

4.4 Configuring SSO in an Oracle Access Manager Environment

For information about how to configure Oracle Access Manager as the SSO authentication provider for Oracle Fusion Middleware with, see "Configuring Single Sign-On in Oracle Fusion Middleware" in *Oracle Fusion Middleware Security Guide*. For more information about managing Oracle Access Manager, see Oracle Fusion Middleware Administrator's Guide for Oracle Access Manager.

After the Oracle Fusion Middleware environment is configured, in general the following must be done to configure Oracle Business Intelligence:

- Configure the SSO provider to protect the Oracle Business Intelligence URL entry points.
- Configure the Web server to forward requests from the Oracle BI Presentation Server to the SSO provider.
- Configure the new identity store as the main authentication source for the Oracle WebLogic Server domain in which Oracle Business Intelligence has been installed. For more information, see Section 4.4.1, "Configuring a New Authenticator for Oracle WebLogic Server".
- Configure the Oracle WebLogic Server domain in which Oracle Business Intelligence is installed to use an Oracle Access Manager asserter. For more information, see Section 4.4.2, "Configuring a New Identity Asserter for Oracle WebLogic Server".
- After configuration of the SSO environment is complete, enable SSO authentication for Oracle Business Intelligence. For more information, see Section 4.4.3, "Using Fusion Middleware Control to Enable SSO Authentication".

4.4.1 Configuring a New Authenticator for Oracle WebLogic Server

After Oracle Business Intelligence is installed, the Oracle WebLogic Server embedded LDAP server is the default authentication source (identity store). The Oracle WebLogic Server domain in which Oracle Business Intelligence is installed must be configured to use the new identity store as the main authentication source. This topic uses Oracle Internet Directory as an example and you should adapt accordingly for the SSO provider being used.

Setting the Control Flag attribute for the authenticator provider determines the ordered execution of the Authentication providers. The possible values for the Control Flag attribute are:

- REQUIRED - This LoginModule must succeed. Even if it fails, authentication proceeds down the list of LoginModules for the configured Authentication providers. This setting is the default.
- REQUISITE - This LoginModule must succeed. If other Authentication providers are configured and this LoginModule succeeds, authentication proceeds down the list of LoginModules. Otherwise, return control to the application.
- SUFFICIENT - This LoginModule needs not succeed. If it does succeed, return control to the application. If it fails and other Authentication providers are configured, authentication proceeds down the LoginModule list.

- OPTIONAL - The user is allowed to pass or fail the authentication test of this Authentication providers. However, if all Authentication providers configured in a security realm have the JAAS Control Flag set to OPTIONAL, the user must pass the authentication test of one of the configured providers.

For more information about creating a new default authenticator in Oracle WebLogic Server, see *Oracle Fusion Middleware Oracle WebLogic Server Administration Console Online Help* or *Oracle Fusion Middleware Securing Oracle WebLogic Server*.

To configure a new authenticator in Oracle WebLogic Server**:**

1. Log in to Oracle WebLogic Server Administration Console.

   For more information, see Section 2.4.2, "How to Launch Oracle WebLogic Server Administration Console".

2. In Oracle WebLogic Server Administration Console, select **Security Realms** from the left pane and click the realm you are configuring. For example, **myrealm**. Select **Providers**.

3. Click **New**. Complete the fields as follows:
   - **Name**: *OID Provider*, or a name of your choosing.
   - **Type**: OracleInternetDirectoryAuthenticator
   - Click **OK**

4. In the **Authentication Providers** table, click the newly added authenticator.

5. Navigate to **Settings**, then select **Common**:
   - Set the Control Flag to **SUFFICIENT.**
   - Click **Save**.

6. Click the **Provider Specific** tab and enter the following required settings using values for your environment:
   - **Host**: Your LDAP host. For example: *localhost*.
   - **Port**: Your LDAP host listening port. For example: *6050*.
   - **Principal**: LDAP administrative user. For example: *cn=orcladmin*.
   - **Credential**: LDAP administrative user password.
   - **User Base DN**: Same searchbase as in Oracle Access Manager.
   - **All Users Filter**: For example, (&(uid=*) (objectclass=person))
   - **User Name Attribute**: Set as the default attribute for username in the directory server. For example: *uid*
   - **Group Base DN**: The group searchbase (same as User Base DN)
   - Do not set the All Groups filter as the default works fine as is.
   - Click **Save**.

7. **Default Authenticator**: Perform the following steps to set up the **Default Authenticator** for use with the Identity Asserter:
   a. From Providers tab, select **Authentication**, then select **DefaultAuthenticator** to display its configuration page.
   b. Select the **Common** tab and set the Control Flag to SUFFICIENT.
   c. Click **Save**.

8. In the **Providers** tab, perform the following steps to reorder **Providers**:
   a. Click **Reorder.**
   b. On the **Reorder Authentication Providers** page, select a provider name and use the arrows beside the list to order the providers as follows:
      – OID Authenticator (SUFFICIENT)
      – OAM Identity Asserter (REQUIRED)
      – Default Authenticator (SUFFICIENT)
   c. Click **OK** to save your changes.
9. **Activate Changes**: In the Change Center, click **Activate Changes**.
10. Restart Oracle WebLogic Server.

4.4.2 Configuring a New Identity Asserter for Oracle WebLogic Server

The Oracle WebLogic Server domain in which Oracle Business Intelligence is installed must be configured to use an Oracle Access Manager asserter.

For more information about creating a new asserter in Oracle WebLogic Server, see *Oracle Fusion Middleware Oracle WebLogic Server Administration Console Online Help*.

To configure a new asserter for Oracle WebLogic Server**:**

1. Log in to Oracle WebLogic Server Administration Console.

   For more information, see Section 2.4.2, "How to Launch Oracle WebLogic Server Administration Console".
2. In Oracle WebLogic Server Administration Console, select **Security Realms** from the left pane and click the realm you are configuring. For example, **myrealm**. Select **Providers**.
3. Click **New**. Complete the fields as follows:
   - **Name**: *OAM Provider*, or a name of your choosing.
   - **Type**: OAMIdentityAsserter.
   - Click **OK**.
   - Click **Save**.
4. In the **Providers** tab, perform the following steps to reorder **Providers**:
   a. Click **Reorder**
   b. On the **Reorder Authentication Providers** page, select a provider name and use the arrows beside the list to order the providers as follows:
      – OID Authenticator (SUFFICIENT)
      – OAM Identity Asserter (REQUIRED)
      – Default Authenticator (SUFFICIENT)
   c. Click **OK** to save your changes.
5. **Activate Changes**: In the Change Center, click **Activate Changes**.
6. Restart Oracle WebLogic Server.

You can verify that Oracle Internet Directory is the new identity store (default authenticator) by logging back into Oracle WebLogic Server and verifying the users and groups stored in the LDAP server appear in the console.

4.4.3 Using Fusion Middleware Control to Enable SSO Authentication

After Oracle Business Intelligence has been configured to use the SSO solution configured for use by Oracle Fusion Middleware, you enabled SSO authentication for Oracle Business Intelligence in Fusion Middleware Control from the **Security** tab.

To enable Oracle Business Intelligence **to use SSO authentication:**

1. Go to the Business Intelligence Overview page.

   For information, see "Logging In to Fusion Middleware Control" in *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*.

2. Go to the **Security** page.

   Click the **Help** button on the page to access the page-level help for its elements.

3. Click **Lock and Edit Configuration**.

4. Check **Enable SSO**.

   The SSO provider list becomes active.

5. Select the configured SSO provider from the list.

6. Click **Apply**, then **Activate Changes**.

7. Manually edit each instanceconfig.xml file for every Oracle BI Presentation Services process to configure the login and logout information. Inside the <Authentication> section, add the following:

```
<SchemaExtensions>
  <Schema name="SSO" logonURL="{your SSO logon URL}" logoffURL="{your logoff
URL}/>
</SchemaExtensions>
```

   Note: For the logout page, you must use the URL specified by the SSO provider for this purpose. Do not use a URL within the domain and port protected by the SSO provider, because the system does not log users out.

   For information about where to locate Oracle Business Intelligence configuration files, see "Where Configuration Files are Located" in *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*.

8. Save each instanceconfig.xml file.

9. Restart the Oracle Business Intelligence components using Fusion Middleware Control.

   For more information, see "Starting and Stopping the Oracle Business Intelligence Components" in *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*.

5

SSL Configuration in Oracle Business Intelligence

This chapter describes how to configure Oracle BI components to communicate over the Secure Socket Layer (SSL).

> **Note:** For a detailed list of security setup steps, see Section 1.8, "Detailed List of Steps for Setting Up Security In Oracle Business Intelligence".

The SSL Everywhere feature of Oracle Business Intelligence enables secure communications between the components. You can configure SSL communication between the Oracle Business Intelligence components and between Oracle WebLogic Server for secure HTTP communication across your deployment. This section does not cover configuring secure communications to external services, such as databases and Web servers. For information about how to configure SSL for Oracle WebLogic Server, see "SSL Configuration in Oracle Fusion Middleware" in *Oracle Fusion Middleware Administrator's Guide*.

This chapter contains the following sections:

5.1 Common SSL Configuration Tasks for Oracle Business Intelligence

Table 5–1 contains common SSL configuration tasks and provides links for obtaining more information.

Table 5–1 Task Map: Configuring SSL Communication for Oracle Business Intelligence

Task	Description	Information
Understand SSL communication in Oracle Business Intelligence.	Understand how SSL communication between components and the application server works.	Section 5.2, "About SSL"

Table 5–1 (Cont.) Task Map: Configuring SSL Communication for Oracle Business

Task	Description	Information
Configure SSL communication between the Oracle WebLogic Server Managed servers.	The Web server must be configured to use HTTPS *before* enabling SSL communication for Oracle Business Intelligence.	Section 5.3, "Configuring the Web Server to Use HTTPS Protocol" "SSL Configuration in Oracle Fusion Middleware" in *Oracle Fusion Middleware Administrator's Guide*
Configure SSL communication between components.	Configure SSL communication between Oracle Business Intelligence components.	Section 5.4, "Configuring SSL Communication Between Components"

5.2 About SSL

SSL is a cryptographic protocol that enables secure communication between applications across a network. Enabling SSL communication provides several benefits, including message encryption, data integrity, and authentication. An encrypted message ensures confidentiality in that only authorized users have access to it. Data integrity ensures that a message is received intact without any tampering. Authentication guarantees that the person sending the message is who they claim to be.

For more information about SSL concepts and public key cryptography, see "How SSL Works" in *Oracle Fusion Middleware Administrator's Guide*.

5.2.1 SSL in Oracle Business Intelligence

By default, Oracle Business Intelligence components communicate with each other using TCP/IP. Configuring SSL between the Oracle Business Intelligence components enables secured network communication.

Oracle Business Intelligence components can communicate only through one protocol at a time. It is not possible to use SSL between some components, while using simple TCP/IP communications between others. To enable secure communication, all instances of the following Oracle Business Intelligence components must be configured to communicate over SSL:

- Oracle BI Server
- Oracle BI Presentation Services
- Oracle BI JavaHost
- Oracle BI Scheduler
- Oracle BI Job Manager
- Oracle BI Cluster Controller
- Oracle BI Server Clients, such as Oracle BI ODBC Client

SSL requires that the server possess a public key and a private key for session negotiation. The public key is made available through a server certificate. The certificate also contains information that identifies the server. The private key is protected by the server.

The SSL Everywhere central configuration feature configures SSL throughout the Oracle Business Intelligence installation from a single centralized point. Certificates

are created for you and every Oracle Business Intelligence component is configured to use SSL. The following default security level is configured by the SSL Everywhere feature:

- SSL encryption is enabled.
- Mutual SSL authentication is not enabled. Since mutual SSL authentication is not enabled, clients do not need their own private SSL keys. All security sensitive inter-component communication links are authenticated by the BISystemUser credentials, or a user's credential.
- The default cipher suites are used. For information about how to use a non-default cipher suite, see Section 5.6, "Advanced SSL Configuration Options".
- When scaling out, the centrally managed SSL configuration is automatically propagated to any new components that are added.

If a higher level of security is required, manual configuration might be used to augment or replace the SSL Everywhere central configuration. This is considerably more complex. For more information about how to configure SSL manually, contact Oracle Support. For more information, see Access to Oracle Support.

5.2.2 Creating Certificates and Keys in Oracle Business Intelligence

Secure communication over SSL requires certificates signed by a certificate authority (CA). For internal communication, the SSL Everywhere feature creates both a private certificate authority and the certificates for you. The internal certificates cannot be used for the outward facing Web server because user Web browsers are not aware of the private certificate authority. The Web server must therefore be provided with a Web server certificate signed by an externally recognized certificate authority. The central SSL configuration must be given the external certificate authority's root certificate so that the Oracle Business Intelligence components can recognize the Web server certificate.

5.2.3 Credential Storage

The Oracle Business Intelligence credential store is used to store the SSL credentials, such as certificates, trusted certificates, certificate requests, and private keys. SSL-related credentials are stored in the oracle.bi.enterprise credential map. The supported certificate file formats are .der and .pem.

5.3 Configuring the Web Server to Use HTTPS Protocol

The Web server must be configured to use HTTPS before enabling SSL communication for Oracle Business Intelligence. For information about how to configure SSL for Oracle WebLogic Server, see "SSL Configuration in Oracle Fusion Middleware" in *Oracle Fusion Middleware Administrator's Guide*.

Some Oracle Business Intelligence Java components running in Oracle WebLogic Server invoke other web services running in Oracle WebLogic Server. Therefore, Oracle WebLogic Server must be configured to trust itself by setting the following Java properties:

- javax.net.ssl.trustStore
- javax.net.ssl.trustStorePassword

These properties are set by editing *MW_Home*/user/projects/domains/bifoundation_domain/bin/startManagedWebLogic.sh (or .bat), and adding the properties to the end

of the JAVA_OPTIONS value. Note that any \ character in a path must be escaped with another \ character.

For example, the following edits are made if using the demonstration Oracle WebLogic Server certificate:

```
set JAVA_OPTIONS=%JAVA_OPTIONS% -Djavax.net.ssl.trustStore="C:/biee/wlserver_
10.3/server/lib/DemoTrust.jks" -Djavax.net.ssl.trustStorePassword=""
```

If this step is omitted then Web Services for SOA and BI Search fail.

Best practice is to disable the HTTP listener and leave only the HTTPS listener. After disabling the HTTP listener you must restart Oracle WebLogic Server. If Oracle WebLogic Server is not restarted, then the log in attempts to Oracle Business Intelligence fail.

If the trust store location is given incorrectly, then Web Services for SOA display an error message similar to the following:

```
java.security.InvalidAlgorithmParameterException: the trustAnchors parameter must
be non-empty
```

5.4 Configuring SSL Communication Between Components

Table 5–2 contains the tasks for setting up SSL communication between components and provides links for obtaining more information.

> **Note:** You must configure SSL for the Web server before enabling SSL for Oracle Business Intelligence. For more information, see Section 5.3, "Configuring the Web Server to Use HTTPS Protocol".

Table 5–2 Task Map: Configuring SSL Communication Between Components

Task	Description	For Information
Lock the configuration.	Use the BIDomain Mbean to lock the domain configuration before making changes.	Section 5.4.1, "Locking the Configuration"
Generate the SSL certificate.	Use the BIDomain.BIInstance.SecurityConfiguration Mbean to generate the SSL certificate.	Section 5.4.2, "Generating the SSL Certificates"
Commit the SSL configuration changes.	Use the BIDomain Mbean to commit the SSL configuration changes.	Section 5.4.3, "Commit the SSL Configuration Changes"
Verify SSL certificates in credential store.	Verify that the SSL certificates are saved in the credential store.	Section 5.4.4, "Verifying the SSL Credentials in the Credential Store"
Enable the SSL configuration and restart Oracle Business Intelligence components.	Use the BIDomain.BIInstance.SecurityConfiguration Mbean to enable the SSL configuration between components, then restart the components so the changes take effect.	Section 5.4.5, "Enabling the SSL Configuration"

Table 5–2 (Cont.) Task Map: Configuring SSL Communication Between Components

Task	Description	For Information
Confirm that SSL communication is enabled between components.	Run the SSL report to confirm status.	Section 5.4.6, "Confirming SSL Status"
Configure SSL communication for the mail server.	Configure SSL communication for the mail server.	Section 5.4.7, "Configuring the SMTP Server"
Update expired SSL certificates.	Update expired SSL certificates and replace with new ones.	Section 5.4.8, "Updating Expired SSL Certificates"

Internal SSL communication between components is configured using Oracle Business Intelligence managed beans (MBeans). A Mbean is a Java object that represents a JMX manageable resource in a distributed environment, such as an application.

Use the Fusion Middleware Control System Mbean Browser to configure SSL communication between Oracle Business Intelligence components. The System Mbean Browser is accessed from the Oracle WebLogic Server domain in which Oracle Business Intelligence is installed in the Fusion Middleware Control target navigation pane. For example, bifoundation_domain.

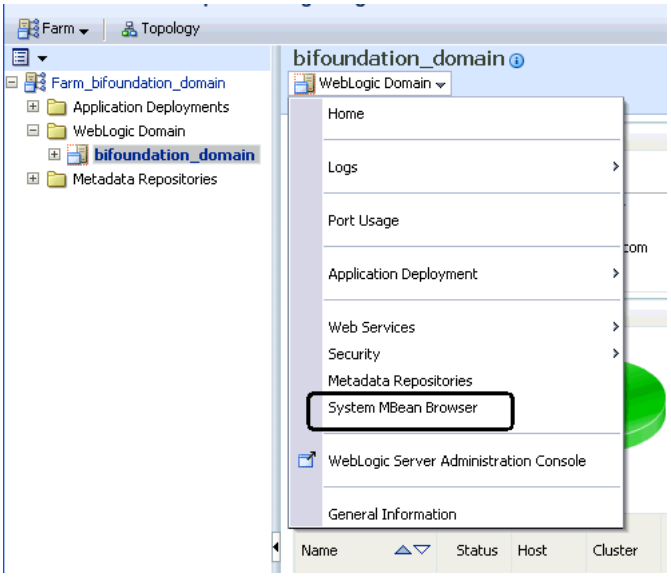
For more information about using and navigating within Fusion Middleware Control, see "Navigating Within Fusion Middleware Control" in *Oracle Fusion Middleware Administrator's Guide*.

5.4.1 Locking the Configuration

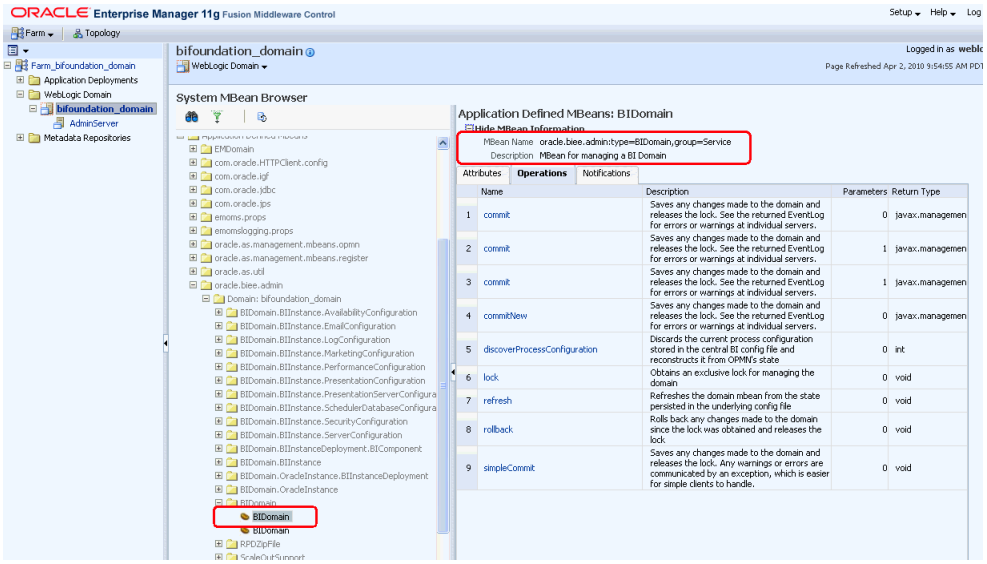
Configuring SSL between components requires that you lock the configuration before making changes. The BIDomain Mbean is used to lock the configuration.

To lock the configuration:

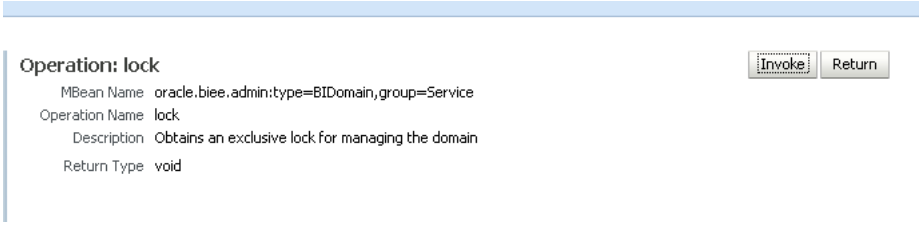
1. In Fusion Middleware Control target navigation pane, go to the Oracle WebLogic Server domain in which Oracle Business Intelligence is installed. Select this domain. For example, bifoundation_domain.
2. From the WebLogic Domain menu, select **System MBean Browser**.



3. Expand the Application Defined MBeans node in the MBean navigation tree, then expand the oracle.biee.admin node, then expand the bifoundation_domain node.
4. Locate and expand the BIDomain node to display two BIDomain Mbeans. Then either hover your cursor over each Mbean or click **Show MBean Information** to display their full names:
   - oracle.biee.admin:type=BIDomain, group=Service
   - oracle.biee.admin:type=BIDomain, group=Config
5. Select the BIDomain Mbean having the full name oracle.biee.admin:type=BIDomain, group=Service from the Mbean navigation tree.



6. Select the **Operations** tab, then **Lock**.
7. Click **Invoke**.



A confirmation displays to indicate that the configuration is locked. The next step to generate the SSL certificates. For more information, see Section 5.4.2, "Generating the SSL Certificates".

5.4.2 Generating the SSL Certificates

Internal SSL communication requires that server certificates, a server public key, and a private key be generated. Oracle Business Intelligence acts as a private CA for internal communication only. The BIDomain.BIInstance.SecurityConfiguration MBean is used to generate the SSL certificates.

Note: If you have existing certificates, best practice is to discard them and generate new certificates by following these steps. To use your existing certificates you must manually configure SSL.

To generate the SSL certificate:

1. Lock the configuration.

   For information, see Section 5.4.1, "Locking the Configuration".

2. From Fusion Middleware Control target navigation pane, expand the farm, then expand **WebLogic Domain**, and select **bifoundation_domain**.

3. From the WebLogic Domain menu, select **System MBean Browser**.

   The System MBean Browser page is displayed.

4. Expand the Application Defined MBeans node in the MBean navigation tree, then expand the oracle.biee.admin node, then expand the bifoundation_domain node.

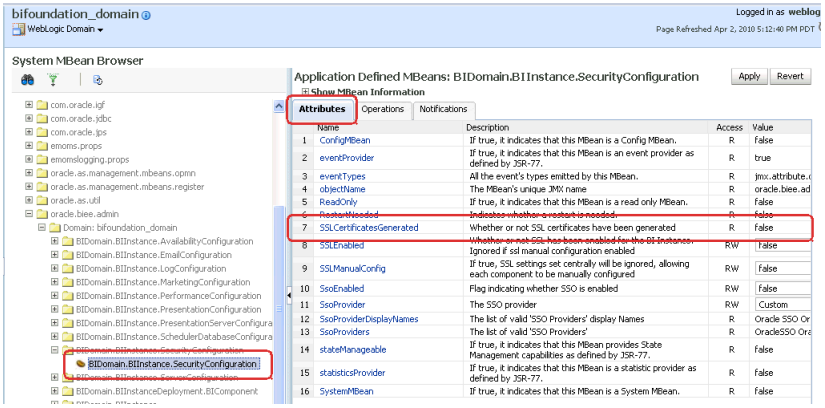
5. Locate and expand the BIDomain.BIInstance.SecurityConfiguration node.

   The BIDomain.BIInstance.SecurityConfiguration Mbean is displayed.

6. Select the BIDomain.BIInstance.SecurityConfiguration MBean.

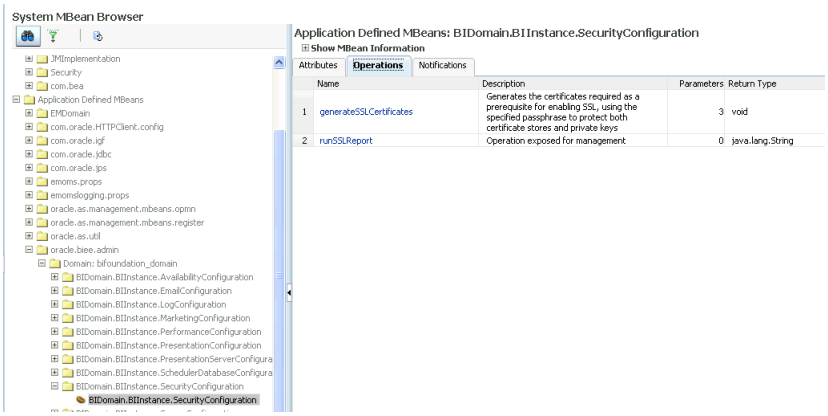
   Configuration options for the MBean display in the right pane.

7. Select the **Attributes** tab, then locate the SSLCertificatesGenerated attribute. A value of false indicates that SSL certificates have not been generated. If certificates have been previously generated, you can continue to replace them with new certificates.



8. Select the **Operations** tab, then select **generateSSLCertificates** operation.

   The parameters for the generateSSLCertificates attribute for the BIDomain.BIInstance.SecurityConfiguration Mbean displays.

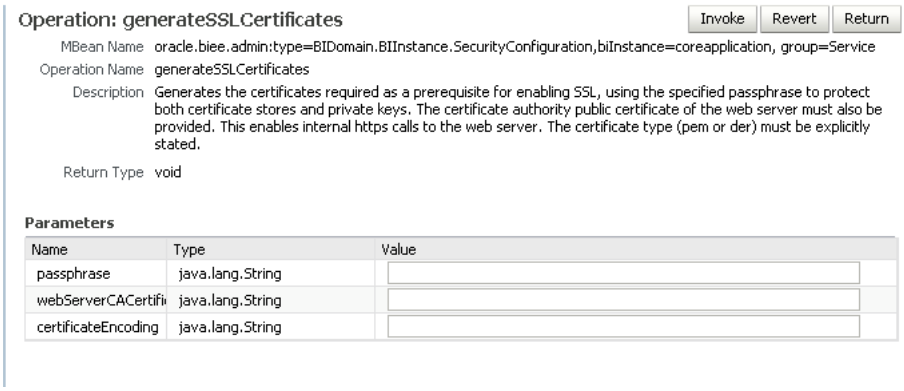


9. Provide values for the following parameters:

   - **passphrase**: Must be more than six characters. The SSL passphrase protects the various certificates and, most importantly, the private key. Remember this passphrase. For example, you need to use it to connect to a BI Server using command line tools that require the tool to verify the BI Server certificate.
   - **webServerCACertificatePath**: Enter the path for the Web server certificate. For Oracle WebLogic Server default demonstration certificate, enter /server/lib/CertGenCA.der. Supported types are .der. and .pem.

     Note: The recommended practice is to install a non-demonstration certificate in Oracle WebLogic Server, signed either by a recognized public certificate authority or your organization's certificate authority. You can obtain the root certificate direct from the certificate authority or by exporting it from your web browser.

   - **certificateEncoding**: Supported types are .der. and .pem. For Oracle WebLogic Server default, enter `der`



10. Click **Invoke**.

    A confirmation displays if the operation executed successfully. If successful, the input CA certificate has been validated and the certificate generation request is queued. The next step is to commit the changes, which completes certificate creation and distribution throughout the domain. For more information, see Section 5.4.3, "Commit the SSL Configuration Changes".

5.4.3 Commit the SSL Configuration Changes

The SSL configuration changes are committed using the BIDomain Mbean.

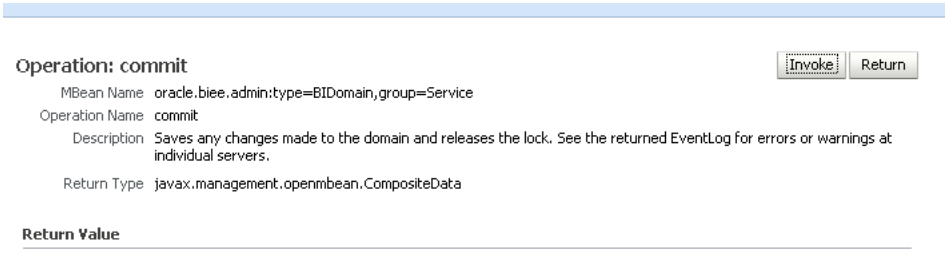
> **Note:** You must configure SSL for the Web server before enabling SSL for Oracle Business Intelligence. For more information, see Section 5.3, "Configuring the Web Server to Use HTTPS Protocol".

To commit the SSL configuration:

1. From the System MBean Browser, navigate to the BIDomain Mbean. You want the Mbean with the complete name of oracle.biee.admin:type=BIDomain, group=Service.

   For more information about navigating to the BIDomain MBean, follow steps 1 through 5 in Section 5.4.1, "Locking the Configuration".

2. Select the BIDomain MBean having the complete name oracle.biee.admin:type=BIDomain, group=Service.
3. Select the **Operations** tab, then **simpleCommit**.
4. Click **Invoke**.



A confirmation displays to indicate if the commit operation was successful.

The next step is to verify the SSL credentials are in the credential store. For more information, see Section 5.4.4, "Verifying the SSL Credentials in the Credential Store".

5.4.3.1 Troubleshooting Tip

If the commit operation fails you might see the following error message:

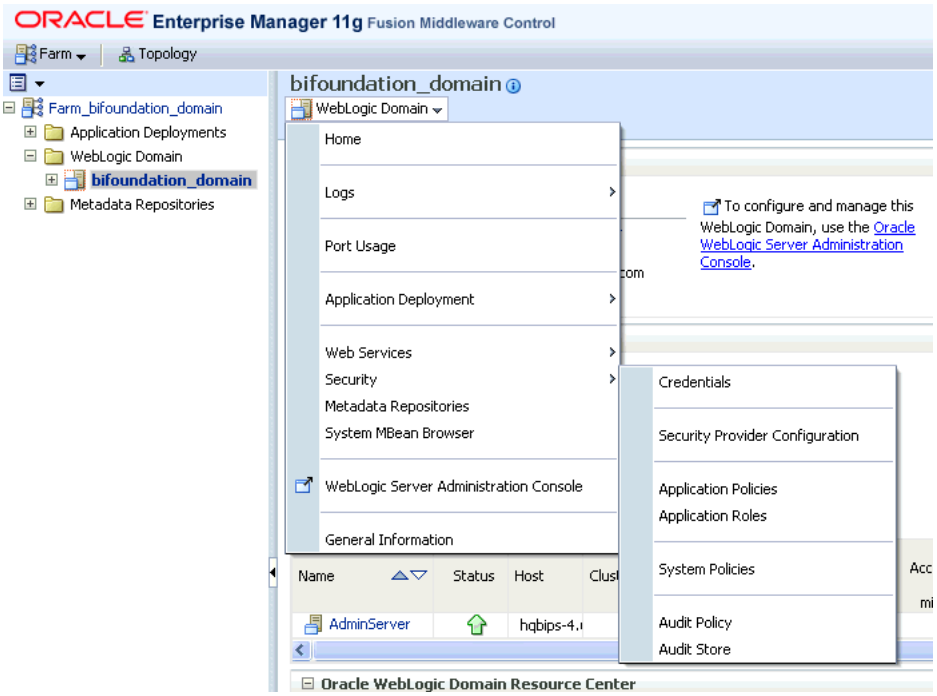
```
SEVERE: Element Type: DOMAIN, Element Id: null, Operation Result:
VALIDATION_FAILED, Detail Message: SSL must be enabled on AdminServer before
enabling on BI system; not set on server: AdminServer
```

This message indicates that SSL has not been enabled on the Oracle WebLogic Server Managed Servers, which is a pre-requisite step. For more information, see Section 5.3, "Configuring the Web Server to Use HTTPS Protocol". After this pre-requisite is completed you can repeat the commit operation.

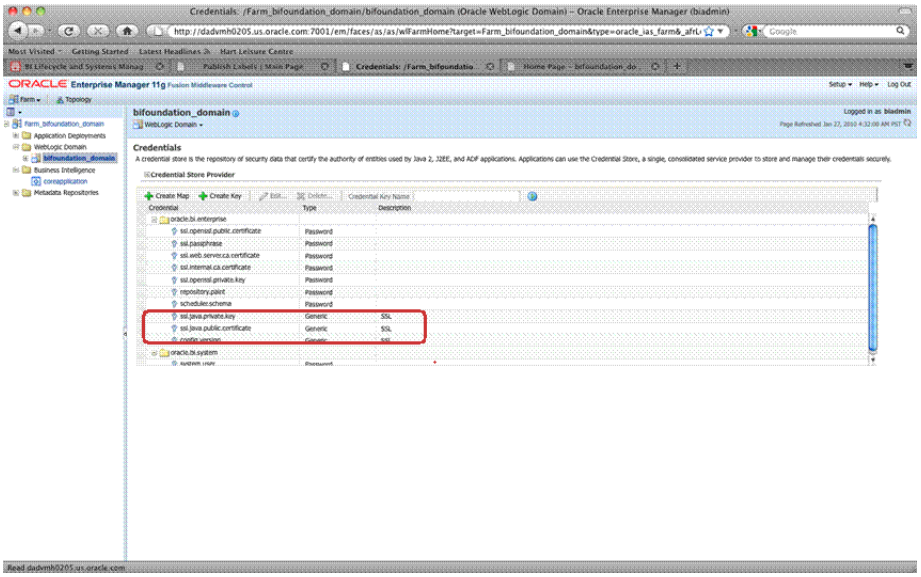
5.4.4 Verifying the SSL Credentials in the Credential Store

The SSL credentials are stored in the credential store for the Oracle Business Intelligence.

1. If necessary, from Fusion Middleware Control target navigation pane, expand the farm, then expand **WebLogic Domain**, and select **bifoundation_domain**.
2. From the WebLogic Domain menu, select **Security**, then **Credentials**.



3. Open oracle.bi.enterprise credential map and verify the SSL credentials have been saved to the credential store. If successful, the following SSL credentials display in the oracle.bi.enterprise credential map:
   - **ssl.java.private.key**
   - **ssl.java.public.certificate**
   - **config.version**



In addition, the certificates are also copied into each MW Home at *MW_HOME*\user_projects\domains\bifoundation_domain\config\fmwconfig\biinstances\coreapplication\ssl. The certificate files are:

- **cacert.pem**: The certificate of the private CA. Command line tools that want to verify the BI Server certificates points to this file.
- **webservercacert.pem**: The certificate of the public CA that signed the Web server certificate. This is a copy of the CA certificate registered in the **generateSSLCertificate** operation, in .pem format.
- **javaserver.keystore**: Contains all the certificates in a format suitable for use by Java clients. Contents include:

Alias	Certificate
javaservercert	Server
javaserverkey	Key
internalcacertificate	Private Key
webservercacertificate	Web server CA

- **server-key.pem**: Private key for the openssl servers.

The next step is to enable the SSL configuration changes. For more information, see Section 5.4.5, "Enabling the SSL Configuration".

5.4.5 Enabling the SSL Configuration

The configuration must be locked before you can enable SSL.

> **Note:** After the SSL configuration is enabled the Oracle Business Intelligence components must be restarted.

1. Verify that the Web server is configured to use HTTPS before enabling the SSL configuration. If necessary, configure the Web server before proceeding.

   For information about how to configure SSL for Oracle WebLogic Server, see Section 5.3, "Configuring the Web Server to Use HTTPS Protocol".

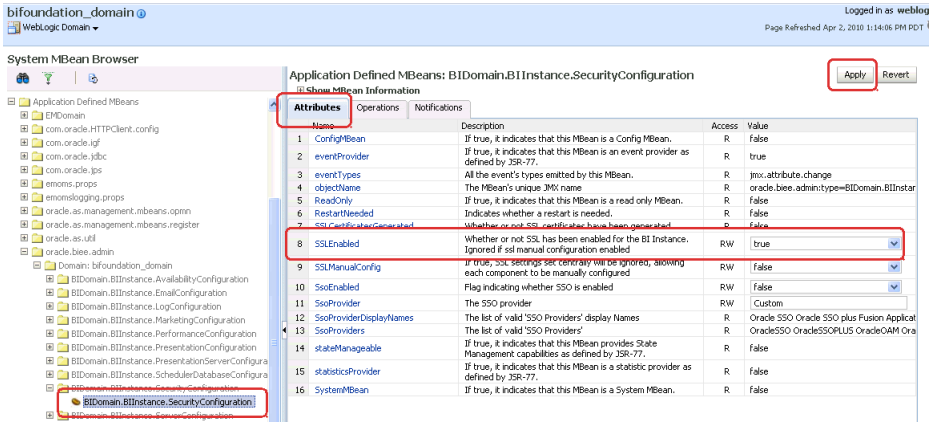
2. Lock the configuration.

   For information, see Section 5.4.1, "Locking the Configuration".

3. From the System Mbean Browser, select the BIDomain.BIInstanceSecurityConfiguration MBean.

   For information about how to navigate to the Mbean, see Section 5.4.2, "Generating the SSL Certificates".

4. Select the **Attributes** tab, then for the SSLEnabled attribute select **true** from the Value list, then click **Apply**. You must have the SSL listen port on for the Administration Server and Manager Servers. For more information, see Section 5.3, "Configuring the Web Server to Use HTTPS Protocol".



5. Navigate to the BIDomain MBean and commit the changes.

   For information, see Section 5.4.3, "Commit the SSL Configuration Changes".

   SSL communication is now enabled between the components. You must restart the Oracle Business Intelligence components for the changes to take effect.

6. Restart the Oracle Business Intelligence components from the Oracle Business Intelligence Overview page in Fusion Middleware Control.

   For information, see "Starting and Stopping Oracle Business Intelligence System Components" in *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*.

5.4.6 Confirming SSL Status

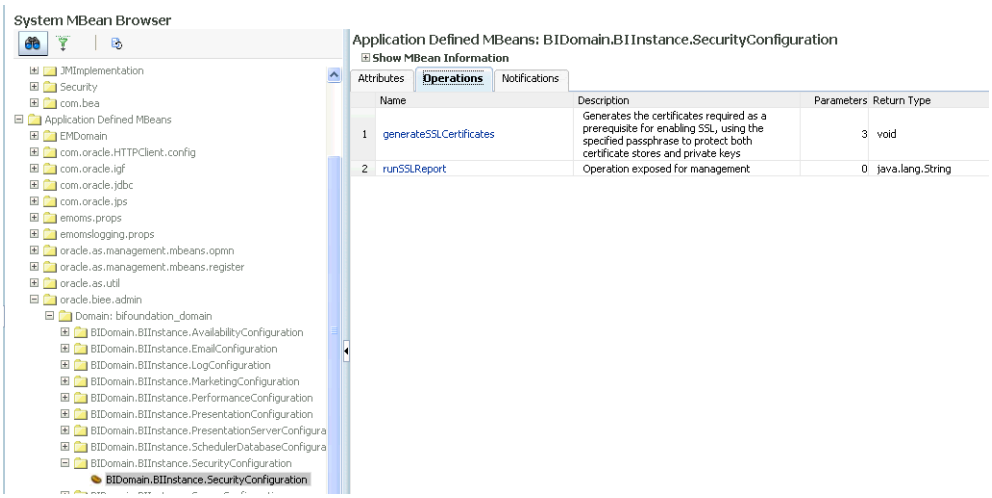
You can run a SSL report using the BIDomain.BIInstance.SecurityConfiguration Mbean to verify that SSL communication is operating between components.

To run the SSL report to confirm status:

1. From the System Mbean Browser, select the BIDomain.BIInstanceSecurityConfiguration MBean.

   For information about how to navigate to the Mbean, see Section 5.4.2, "Generating the SSL Certificates". You do not need to lock the configuration to run the SSL report.

2. Select the **Operations** tab, then select the **runSSLReport** option.



3. To run the report, click **Invoke**.

   The report indicating the status of SSL communication between components displays. See Example 5–1, "Sample SSL Report Output".

   If the SSL ping fails, check the following:

   - Verify the target component is running.
   - Verify that the component has been restarted since SSL was enabled. SSL configuration changes require a restart to take effect.
   - Verify that the SSLEnabled attribute for the BIDomain.BIInstanceSecurityConfiguration MBean is set to true. When changing SSL properties, both the apply and commit steps must be performed.

Example 5–1 Sample SSL Report Output

```
OracleBIPresentationServicesComponent
(1) <machine_name>:9710.  SSL ping OK.   peer: <machine_name> port: 9710 protocol:
SSLv3 cipher suite: SSL_RSA_WITH_RC4_128_MD5
 local certificates:  null
peer certificates: #18, expires Tue might 17 15:23:02 BST 2011 for CN=OBIEE
Installer  Openssl, OU=Business Intelligence, O=Oracle, C=US#9879704091745165219,
expires Tue might 17 15:23:02 BST 2011 for C=US, O=org, OU=unit, CN=OBIEE
Installer CA

OracleBIClusterControllerComponent
(No instances configured)

OracleBISchedulerComponent
(1) <machine_name>:9705.  SSL ping OK.   peer: <machine_name> port: 9705 protocol:
SSLv3 cipher suite: SSL_RSA_WITH_RC4_128_MD5
local certificates:  null
peer certificates: #18, expires Tue might 17 15:23:02 BST 2011 for CN=OBIEE
Installer Openssl, OU=Business Intelligence, O=Oracle, C=US

OracleBIJavaHostComponent
(1) <machine_name>:9810.  SSL ping OK.   peer: <machine_name> port: 9810 protocol:
SSLv3 cipher suite: SSL_RSA_WITH_RC4_128_MD5
 local certificates:  null
peer certificates: #19, expires Tue might 17 15:23:03 BST 2011 for CN=OBIEE
Installer Java, OU=Business Intelligence, O=Oracle, C=US

OracleBIServerComponent
(1) <machine_name>:9703.  SSL ping OK.   peer: <machine_name> port: 9703 protocol:
SSLv3 cipher suite: SSL_RSA_WITH_RC4_128_MD5
local certificates:  null
 peer certificates: #18, expires Tue might 17 15:23:02 BST 2011 for CN=OBIEE
Installer Openssl, OU=Business Intelligence, O=Oracle, C=US

SSL ok on 4 out of 4 components.
```

5.4.7 Configuring the SMTP Server

The server certificate from the SMTP server must be obtained.

To configure SSL for the SMTP server:

1. Go to the **Business Intelligence Overview** page.

For information, see "Logging In to Fusion Middleware Control" in *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*.

2. Display the **Mail** tab of the **Deployment** page.
3. Lock the configuring by clicking **Lock and Edit Configuration**.
4. Complete the fields under **Secure Socket Layer (SSL)** as follows:
   - Check **Use SSL to connect to mail server**. The other fields become active afterward.
   - Specify CA certificate source: select **Directory** or **File**.
   - **CA certificate directory**: Specify the directory containing CA certificates.
   - **CA certificate file**: Specify the file name for the CA certificate.
   - **SSL certificate depth**: Specify the verification level applied to the certificate
   - **SSL cipher list**: Specify the list of ciphers matching the cipher suite name that the SMTP server supports. For example, RSA+RC4+SHA.
5. Unlock the configuration.

5.4.8 Updating Expired SSL Certificates

Certificates generated by the SSL Everywhere central configuration expire after one year. The expiration date for a certificate is listed in the SSL status report. For more information about how to run an SSL report, see Section 5.4.6, "Confirming SSL Status". For an example of the certificate expiration message that is displayed, see Example 5–1, "Sample SSL Report Output".

To replace a certificate that is about to expire, generate new certificates by following the steps in Section 5.4.2, "Generating the SSL Certificates" and restart the Oracle Business Intelligence components.

5.5 Additional SSL Configuration Options

Additional configuration options are required for Oracle Business Intelligence components and tools.

5.5.1 Using SASchInvoke and SchShutdown When BI Scheduler is SSL-Enabled

When BI Scheduler is enabled for communication over SSL, you use the SASchInvoke and SchShutdown command line utilities, as specified below.

Caution: When you run SASchInvoke and SchShutdown commands, the password argument (that is, `[Admin Password]` or `<password>`) is optional. If you do not provide a password argument, you are prompted to enter a password when you run the command. To minimize the risk of security breaches, Oracle recommends that you do not provide a password argument either on the command line or in scripts. Note that the password argument is supported for backward compatibility only.

Use the following syntax for the SASchInvoke command:

```
SASchInvoke -u <Admin Name>/[Admin Password]  (-j <job id> | -i <iBot path>)  [-m
```

```
<machine name>[:<port>]]  [(-r <replace parameter filename> | -a <append parameter
filename>)] [-l [ -c SSL certificate filename> -k <SSL certificate private key
filename> [ -w <SSL passphrase>  | -q <passphrase file>  | -y ]] [-h <SSL cipher
list>] [-v [-e <SSL verification depth>] [-d <CA certificate directory>] [-f <CA
certificate file>] [-t <SSL trusted peer DNs>] ] ]
```

Use the following syntax for the SchShutdown command:

```
SchShutdown -s <machine:port> -u <username> -p <password> [ -l [-c <ssl
certificate file path>-k <ssl private key file path> [-q <ssl private key
passphrase file path> | -w <ssl private key passphrase> | -y ] [-h <ssl cipher
list> ]-v [ -e <ssl verification depth> ] -d <CA Certificate Directory path> | [-f
<CA Certificate File path>][-t <SSL Trusted Peer DNs ]]
```

5.5.2 Configuring Oracle BI Job Manager

To successfully connect to BI Scheduler that has been enabled for SSL, Oracle BI Job Manager must also be configured to communicate over SSL.

Oracle BI Job Manager is a Java based component and the keys and certificates that it uses must be stored in a java keystore database.

Use this procedure to configure Oracle BI Job Manager to communicate with the BI Scheduler server over SSL.

To configure Oracle BI Job Manager:

1. From the **File** menu, select **Oracle BI Job Manager**, then select **Open Scheduler Connection**.
2. In the Secure Socket Layer section of the dialog box, select the **SSL** check box. If you are using the central SSL configuration, which does not set up mutual authentication, you do not need to provide any additional values in this dialog box. Click **OK** to exit.
3. If BI Scheduler has been set to "Require Client Certificate", then Key Store and Key Store Password must be set as follows:
   - Key Store=*MW_HOME*\user_projects\domains\bifoundation_domain\config\fmwconfig\biinstances\coreapplication\ssl\javaserver.keystore.
   - Key Store Password = passphrase entered in the generateSSLCertificates operation. See Step 9 of Section 5.4.2, "Generating the SSL Certificates"
4. Select the **Verify Server Certificat**e check box. When this is checked, the trust store file must be specified. This trust store contains the CA that verifies the Scheduler server certificate.
5. In the **Trust Store** text box, enter the path and file name of the keystore that contains the Certificate Authority file. In the example provided previously, the CA certificate was stored in the same keystore that contains the certificate and private key, javaserver.keystore.
6. In the **Trust Store Password** text box, enter the password of the keystore entered in step 5.
7. Copy the keystore and trust store files to the locations specified in the parameters above.

5.5.3 Online Catalog Manager

The online Catalog Manager might fail to connect to Oracle BI Presentation Services when the HTTP Web server for Oracle BI is enabled for SSL. You must import the SSL server certificate or CA certificate from the Web server into the Java Keystore of the JVM that is specified by the system JAVA_HOME variable.

To import the exported Web server certificate to Java's default truststore:

1. Navigate to Java's default trust store located at JAVA_HOME/ jre/lib/security.

   For example, `<MW_HOME>\jrocket_160_17_R28.0.0-679\jre\lib\security`.

   The default trust store is named cacerts.

2. Copy the certificate exported from the Web server to the same location as Java's default truststore.

3. Execute the command to import the certificate to the default truststore:

   ```
   keytool -import -trustcacerts -alias bicert -file $WebServerCertFilename
   -keystore cacerts -storetype JKS
   ```

   where the Web server certificate file $WebserverCertFilename is imported into Java's default trust store named cacerts under an alias of bicert.

   For example if using the Oracle WebLogic Server default demonstration certificate, then use the full path to the certificate located in <WLS_HOME>/server/lib/CertGenCA.der.

   Note: The default password for the Java trust store is "changeit".

4. Restart Catalog Manager.

   Note: You must start Catalog Manager using the secure HTTPS URL.

5.5.4 Configuring Oracle BI Administration Tool

To successfully connect to Oracle BI Server that has been enabled for SSL, Administration Tool must also be configured to communicate over SSL. The DSN for the Oracle BI Server data source is required.

To configure Administration Tool that is part of a cluster:

1. Determine the Oracle BI Server data source DSN being used by logging into the Presentation Services **Administration** page as an administrative user.

   For more information, see *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*.

2. Locate the **Oracle BI Server Data Source** field in the upper left corner. The DSN is listed in the following format: coreapplication_OH<*DSNnumber*>.

3. In Oracle BI Administration Tool, enter the DSN number by selecting **File**, then **Open**, then **Online**. Select the DSN from the list.

4. Enter the repository password, user name, and password.

   Administration Tool is now connected to BI Server using SSL.

5.5.5 Configuring an ODBC DSN for Remote Client Access

You can create an ODBC DSN for the Oracle BI Server to enable remote client access. For more information about how to enable SSL communication for an ODBC DSN, see "Integrating Other Clients with Oracle Business Intelligence" in *Oracle Fusion Middleware Integrator's Guide for Oracle Business Intelligence Enterprise Edition*.

5.6 Advanced SSL Configuration Options

The default SSL configuration uses default cipher suite negotiation. You can configure the system to use a different cipher suite if your organization's security standards do not allow for the default choice. The default choice can be viewed in the output from the SSL status report.

This advanced option is not configured by the SSL Everywhere central configuration. Instead, individual components must be manually configured. If new components are added by scaling out, each additional component must be manually configured. Manual configuration involves editing of the configuration files (.ini and .xml). Be careful to observe the syntactic conventions of these file types. If the files are incorrect, the corresponding component logs an error in its log file and will not start up.

A manually configured SSL environment can co-exist with a default SSL configuration.

To manually configure SSL cipher suite:

1. Configure SSL Everywhere by following the instructions in Section 5.4, "Configuring SSL Communication Between Components".

   Note: Before making manual changes, invoke the SSLManualConfig MBean under BIDomain.BIInstance.SecurityConfiguration with the usual lock/commit cycle.

2. Select the desired Java Cipher Suite name from the options located at `http://java.sun.com/j2se/1.5.0/docs/guide/security/jsse/JSSERefGuide.html#AppA`.

3. Create an Open SSL Cipher Suite Name that matches the cipher suite chosen, using the list at `http://www.openssl.org/docs/apps/ciphers.html#CIPHER_LIST_FORMAT`.

   For example, Java Cipher Suite name SSL_RSA_WITH_RC4_128_SHA maps to Open SSL: RSA+RC4+SHA.

4. Edit the JavaHost configuration file config.xml (for example, located in \instances\instance1\config\OracleBIJavaHostComponent\coreapplication_obijh1\) and add following sub-element to JavaHost/Listener/SSL element. For example:

   ```
   <EnabledCipherSuites>SSL_RSA_WITH_RC4_128_SHA</EnabledCipherSuites>
   ```

5. If in a clustered environment, edit the Cluster Controller configuration file located at ORACLE_INSTANCE/config/OracleBIApplication/coreapplication/NQClusterConfig.INI and set the SSL_CIPHER_LIST value, as in the following example:

   ```
   SSL_CIPHER_LIST = "RSA+RC4+SHA";
   ```

6. Edit the BI Presentation configuration file located at ORACLE_INSTANCE/config/OracleBIPresentationServicesComponent/coreapplication_obips1/instanceconfig.xml and add the attribute cipherSuites="RSA+RC4+SHA" to the sub-elements WebConfig/ServerInstance/ps:Listener and WebConfig/ServerInstance/ps:JavaHostProxy.

7. Edit the BI Scheduler configuration file located at ORACLE_INSTANCE/config/OracleBISchedulerComponent/coreapplication_obisch1/instanceconfig.xml add following sub-element to scheduler/ServerInstance/SSL. For example:

   ```
   <CipherList>RSA+RC4+SHA</CipherList>
   ```

8. If in a clustered environment, edit the Cluster Controller configuration file located at ORACLE_INSTANCE/config/OracleBIApplication/coreapplication/NQClusterConfig.INI and set the SSL_CIPHER_LIST value, as in the following example:

   ```
   SSL_CIPHER_LIST = "RSA+RC4+SHA";
   ```

9. Restart all the Oracle Business Intelligence components.

   For information, see "Starting and Stopping Oracle Business Intelligence System Components" in *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*.

10. Run a SSL status report to confirm SSL is enabled by following the steps in Section 5.4.6, "Confirming SSL Status".

A

Alternative Security Administration Options

This appendix describes alternative security administration options included for backward compatibility with upgraded systems and are not considered a best practice. This appendix contains the following sections:

- Section A.1, "Alternative Authentication Options"
- Section A.2, "Alternative Authorization Options"

A.1 Alternative Authentication Options

Several Oracle Business Intelligence legacy authentication options are still supported for backward compatibility. The best practice for upgrading systems is to begin implementing authentication using an identity store and authentication provider as provided by the default security model. An embedded directory server is configured as the default identity store and authentication provider during installation or upgrade and is available for immediate use. For more information about the default security model, see Chapter 1, "Introduction to Security in Oracle Business Intelligence" and Appendix B, "Understanding the Default Security Configuration".

Authentication is the process by which the user name and password presented during log in is verified to ensure the user has the necessary credentials to log in to the system. Oracle BI Server authenticates each connection request it receives. The following legacy authentication methods are supported by BI Server for backward compatibility in this release:

- External LDAP-based directory server
- External initialization block authentication
- Table-based

This section contains the following topics:

- Section A.1.1, "Setting Up LDAP Authentication"
- Section A.1.2, "Setting Up External Table Authentication"
- Section A.1.3, "About Oracle BI Delivers and External Initialization Block Authentication"
- Section A.1.4, "Order of Authentication"
- Section A.1.5, "Authenticating by Using a Custom Authenticator Plug-In"
- Section A.1.6, "Managing Session Variables"
- Section A.1.7, "Managing Server Sessions"

A.1.1 Setting Up LDAP Authentication

You can set up BI Server to pass user credentials to an external LDAP server for authentication.

The legacy LDAP authentication method uses Oracle Business Intelligence session variables that you define using the Variable Manager in Oracle BI Administration Tool. For more information about the session variables, see "Using Variables in the Oracle BI Repository" in *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

To set up LDAP authentication:

1. Create an LDAP Server as follows:
   a. Select **Manage** then **Identity** in the Oracle BI Administration Tool to launch the Security Manager.
   b. Select **Directory Servers** from the left pane in Security Manager.
   c. Right-click in the right pane in Security Manager and select **New LDAP Server**. The LDAP Server dialog is displayed.
   d. Create the LDAP server by completing the fields.
2. Create an LDAP initialization block and associate it with an LDAP server. For more information. see "Creating Initialization Blocks" in *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.
3. Define a system variable named USER and map the USER variable to an LDAP attribute (uid or sAMAccountName).

   Session variables get their values when a user begins a session by logging on. Certain session variables, called system session variables, have special uses. The system session variable USER is used with authentication. For more information about the USER system session variable, see "Defining a USER Session Variable for LDAP Authentication". For more information about system session variables, see "About System Session Variables" in *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.
4. If applicable, delete users from the repository file.
5. Associate the USER system variable with the LDAP initialization block. For more information, see "Defining a USER Session Variable for LDAP Authentication" and "Associating Variables with Initialization Blocks" in *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

> **Note:** When using secure LDAP you must restart Oracle BI Administration Tool before testing if you have done the following: set the key file name and password, tested the LDAP parameter setting successfully in the Oracle BI Administration Tool, and then changed the key file name and password again.

A.1.1.1 Setting Up an LDAP Server

For instances of Oracle Business Intelligence that use ADSI as the authentication method, the following options should be used when setting up the AD instance:

- In **Log On To**, select **All Computers**, or if you list some computers, include the AD server as a Logon workstation.
- Ensure that **User must change password at next logon** is not selected.

In the Oracle BI Administration Tool, the CN user used for the BIND DN in the LDAP Server section must have both ldap_bind and ldap_search authority.

> **Note:** BI Server uses cleartext passwords in LDAP authentication. Make sure your LDAP Servers are set up to allow this.

To set up LDAP authentication for the repository:

1. Open a repository in the Oracle BI Administration Tool in either offline or online mode.
2. From **Identity Manager**, select **Action**, then **New**, then **LDAP Server**.
3. In the LDAP Server dialog, in the General tab, complete the necessary fields. The following list of options and descriptions contain additional information to help you set up the LDAP server:
   - **Host name**. The name of your LDAP server.
   - **Port number**. The default LDAP port is 389.
   - **LDAP version**. LDAP 2 or LDAP 3 (versions). The default is LDAP 3.
   - **Base DN**. The base distinguished name (DN) identifies the starting point of the authentication search. For example, if you want to search all of the entries under the o=Oracle.com subtree of the directory, o=Oracle.com is the base DN.
   - **Bind DN and Bind Password**. The optional DN and its associated user password that are required to bind to the LDAP server.

     If these two entries are blank, anonymous binding is assumed. For security reasons, not all LDAP servers allow anonymous binding.

     These fields are optional for LDAP V3, but required for LDAP V2, because LDAP V2 does not support anonymous binding.

     These fields are required if you select the **ADSI** option. If you leave these fields blank, a warning message appears asking if you want to leave the password empty anyway. If you click **Yes**, anonymous binding is assumed.
   - **Test Connection**. Use this button to verify your parameters by testing the connection to the LDAP server.
4. Click the **Advanced** tab, and type the required information. BI Server maintains an authentication cache in memory that improves performance when using LDAP to authenticate large numbers of users. Disabling the authentication cache can slow performance when hundreds of sessions are being authenticated.

   The following list of fields and descriptions contain additional information to help you set up the LDAP server:
   - **Connection timeout**. When BI Server attempts to connect to an LDAP server for user authentication, the connection times out after the specified interval.
   - **Domain identifier** (Optional). Typically, the identifier is a single word that uniquely identifies the domain for which the LDAP object is responsible. This is especially useful when you use multiple LDAP objects. If two different users have the same user ID and each is on a different LDAP server, you can designate domain identifiers to differentiate between them. The users log in to the BI Server using the following format:

     domain_id/user_name

If a user enters a user name without the domain identifier, then it is authenticated against all available LDAP servers in turn. If there are multiple users with the same name, then only one user can be authenticated.

- **ADSI**. (Active Directory Service Interfaces) A type of directory server. If you select the **ADSI** option, **Bind DN** and **Bind password** are required.
- **SSL**. (Secure Sockets Layer) Select this option to enable SSL.
- **User Name Attribute Type**. This parameter uniquely identifies a user. In many cases, this is the attribute used in the RDN (relative distinguished name). Typically, you accept the default value. For most LDAP servers, you would use the user ID. For ADSI, use sAMAccountName.

A.1.1.2 Defining a USER Session Variable for LDAP Authentication

To set up LDAP authentication, you define a system session variable called USER and associate it with an LDAP initialization block that is associated with an LDAP server. When a user logs in to the Oracle BI Server, the user name and password is passed to the LDAP server for authentication. After the user is authenticated successfully, other session variables for the user could also be populated from information returned by the LDAP server.

Note: If the user exists in both an external LDAP server using the legacy method and in an LDAP-based identity store based on Oracle Platform Security Services, the user definition in the identity store takes precedence and LDAP authentication fails.

The information in this section assumes that an LDAP initialization block has been defined.

For users not defined in an LDAP-based identity store, the presence of the defined system variable USER determines that external authentication is performed. Associating USER with an LDAP initialization block determines that the user is authenticated by LDAP. To provide other forms of authentication, associate the USER variable with an initialization block associated with an external database or XML source.

To define the USER session system variable for LDAP authentication:

1. Select **Manage**, then **Variables** from the Oracle BI Administration Tool menu.
2. Select the **System** leaf of the tree in the left pane.
3. Right-click in the right pane and select **New USER**.
4. In the Session Variable - USER dialog box, select the appropriate LDAP initialization block from the **Initialization Block** drop-down list.

   The selected initialization block provides the `USER` session system variable with its value.
5. Click **OK** to create the `USER` variable.

A.1.1.3 Setting the Logging Level

Use the system variable LOGLEVEL to set the logging level for users who are authenticated by an LDAP server.

A.1.2 Setting Up External Table Authentication

You can maintain lists of users and their passwords in an external database table and use this table for authentication purposes. The external database table contains user names and passwords, and could contain other information, including group membership and display names used for Oracle BI Presentation Services users. The table could also contain the names of specific database catalogs or schemas to use for each user when querying data.

> **Note:** If a user belongs to multiple groups, the group names should be included in the same column, separated by semicolons.

External table authentication uses session variables that you define using the Variable Manager in the Oracle BI Administration Tool. For more information about the Variable Manager, see "Using Variables in the Oracle BI Repository" in *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

Session variables get their values when a user begins a session by logging on. Certain session variables, called system variables, have special uses. The variable USER is a system variable that is used with external table authentication.

To set up external table authentication, you define a system variable called USER and associate it with an initialization block that is associated with an external database table. Whenever a user logs in, the user ID and password is authenticated using SQL that queries this database table for authentication. The initialization block uses the database connection in the physical layer to connect to the database. The connection in the physical layer contains the log in information. After the user is authenticated successfully, other session variables for the user could also be populated from the results of this SQL query.

The presence of the defined system variable USER determines that external authentication is performed. Associating USER with an external database table initialization block determines that the user is authenticated using the information in this table. To provide other forms of authentication, associate the USER system variable with an initialization block associated with a LDAP server or XML source. For more information, see "Setting Up LDAP Authentication".

To set up external table authentication:

1. Import information about the external table into the Physical layer.
2. Select **Manage**, then **Variables** in the Oracle BI Administration Tool to open the Variable Manager.
3. Select **Initialization Blocks** in the left pane.
4. Right-click in the right pane and select **New Initialization Block**.
5. In the Initialization Block dialog box, type a name for the initialization block.
6. Select **Database** from the **Data Source Connection** drop-down list.
7. Click **Browse** to search for the name of the connection pool this block uses.
8. In the **Initialization String** area, type the SQL statement that is issued at authentication time.

   The values returned by the database in the columns in your SQL is assigned to variables. The order of the variables and the order of the columns determines

which columns are assigned to which variables. Consider the SQL in the following example:

```
SELECT username, grp_name, SalesRep, 2 FROM securitylogons WHERE username =
':USER' and pwd = ':PASSWORD'
```

This SQL contains two constraints in the `WHERE` clause:

– :USER (note the colon) equals the name the user entered when logging on.

– `:PASSWORD` (note the colon) equals the password the user typed.

The query returns data only if the user name and password match values found in the specified table.

You should test the SQL statement outside of the Oracle BI Server, substituting valid values for :USER and :PASSWORD to verify that a row of data returns.

9. If this query returns data, then the user is authenticated and session variables are populated. Because this query returns four columns, four session variables are populated. Create these variables (USER, GROUP, DISPLAYNAME, and LOGLEVEL) by clicking **New** in the Variables tab.

   If a variable is not in the desired order, click the variable you want to reorder and use the **Up** and **Down** buttons to move it.

10. Click **OK** to save the initialization block.

A.1.3 About Oracle BI Delivers and External Initialization Block Authentication

Oracle BI Scheduler Server runs Delivers jobs for users without accessing or storing their passwords. Using a process called impersonation, Oracle BI Scheduler uses one user name and password with Oracle Business Intelligence administrative privileges that can act on behalf of other users. Oracle BI Scheduler initiates an Agent by logging on to Oracle BI Presentation Services with the Oracle Business Intelligence administrative name and password.

For Delivers to work, all database authentication must be performed in only one connection pool, and that connection pool can only be selected in an initialization block for the USER system session variable. This is typically called the Authentication Initialization Block. When impersonation is used, this initialization block is skipped. All other initialization blocks must use connection pools that do not use database authentication.

> **Caution:** An authentication initialization block is the only initialization block in which it is acceptable to use a connection pool where :USER and :PASSWORD are passed to a physical database.

For other initialization blocks, SQL statements can use :USER and :PASSWORD. However, because Oracle BI Scheduler Server does not store user passwords, the WHERE clause must be constructed as shown in the following example:

```
SELECT username, groupname, dbname, schemaname FROM users
WHERE username=':USER'
NQS_PASSWORD_CLAUSE(and pwd=':PASSWORD')NQS_PASSWORD_CLAUSE
```

When impersonation is used, everything in the parentheses is extracted from the SQL statement at runtime.

For more information, see the Oracle BI Delivers examples in *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

A.1.4 Order of Authentication

The Oracle BI Server populates session variables using the initialization blocks in the desired order that are specified by the dependency rules defined in the initialization blocks. If the server finds the session variable USER, it performs authentication against an LDAP server or an external database table, depending on the configuration of the initialization block with which the USER variable is associated.

Authentication against the identity store configured in Oracle WebLogic Server Administration Console occurs first, and if that fails, then initialization block authentication occurs.

A.1.5 Authenticating by Using a Custom Authenticator Plug-In

You can create a customized authentication module using initialization blocks. An **authenticator** is a dynamic link library (DLL), or shared object on UNIX, written by a customer or developer that conforms to the Oracle BI Authenticator API Specification and can be used by BI Server to perform authentication and other tasks at run time. The dynamically loadable authentication module is a BI Server module with a cache layer that uses the authenticator to perform authentication and related tasks at run time.

Two sample authenticator plug-ins are installed when you install Oracle Business Intelligence. One is available only for the Microsoft Windows platform. The other one uses a text file for user information storage and is available to all platforms. A header file is provided for all types that are used in the dynamically loadable authenticator. You can find the header files at *ORACLE_HOME*`\server\SDK\CustomAuthenticatorSamples`.

An administrative user can ask a developer to implement a dynamically loadable authentication module according to the Oracle BI Authenticator API specification. For more information about this specification, see *Oracle Fusion Middleware Integrator's Guide for Oracle Business Intelligence Enterprise Edition*.

After you create an authentication object (authenticator plug-in) and specify a set of parameters for the authentication module (such as configuration file path, number of cache entries, and cache expiration time), you must associate the authentication object with an initialization block. You can associate the USER variable (required) and other variables with the initialization blocks.

When a user logs in, if the authentication is successful, this populates a list of variables, as specified in the initialization block.

A custom authenticator is an object in the repository that represents a custom C authenticator plug-in. This object is used with an authentication init block to enable the BI Server component to authenticate users against the custom authenticator. The recommended method for authentication is to use Oracle WebLogic Server's embedded LDAP server. However, the practice of using custom authenticators can continue to be used.

To add a custom authenticator:

1. In Oracle BI Administration Tool, select **Manage**, then **Identity**. Select **Custom Authenticators** from the navigation tree. Select from the following options:
   - To create new a new custom authenticator: Right-click in the right pane and select **New Custom Authenticator**.

- To edit a custom authenticator: Double-click the name.

2. In the **Custom Authenticator** dialog, complete the necessary fields.
   - **Authenticator plug-in**: The path and name of the authenticator plug-in DLL.
   - **Configuration parameters**: Lists any parameters for this custom authenticator that have been explicitly exposed for configuration.
   - **Encrypted parameter**: Lists any parameters for this custom authenticator that have been encrypted, such as passwords.
   - **Cache persistence time**: The interval at which the authentication cache entry for a logged on user is refreshed for this custom authenticator.
   - **Number of cache entries**: The maximum number of entries in the authentication cache for this custom authenticator, preallocated when the Oracle BI Server starts. If the number of users exceeds this limit, cache entries are replaced using the LRU algorithm. If this value is 0, then the authentication cache is disabled.
3. Click **OK**.

A.1.6 Managing Session Variables

System session variables obtain their values from initialization blocks and are used to authenticate Oracle Business Intelligence users against external sources such as LDAP servers or database tables. Every active BI Server session generates session variables and initializes them. Each session variable instance can be initialized to a different value. For more information about how session variable and initialization blocks are used by Oracle Business Intelligence, see "Using Variables in the Oracle BI Repository" in *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

A.1.7 Managing Server Sessions

The Oracle BI Administration Tool Session Manager is used in online mode to monitor activity. The Session Manager shows all users logged in to the session, all current query requests for each user, and variables and their values for a selected session. Additionally, an administrative user can disconnect any users and terminate any query requests with the Session Manager.

How often the Session Manager data is refreshed depends on the amount of activity on the system. To refresh the display at any time, click **Refresh**.

A.1.7.1 Using the Session Manager

The Session Manager contains an upper pane and a lower pane:

- The top pane, the Session pane, shows users currently logged in to BI Server. To control the update speed, from the **Update Speed** list, select **Normal**, **High**, or **Low**. Select **Pause** to keep the display from being refreshed.
- The bottom pane contains two tabs:
  - The Request tab shows active query requests for the user selected in the Session pane.
  - The Variables tab shows variables and their values for a selected session. You can click the column headers to sort the data.

Table A–1 and Table A–2 describe the columns in the Session Manager dialog.

Table A–1 Fields in the Session Manager Dialog

Column Name	Description
Client Type	The type of client connected to the server.
Last Active Time	The time stamp of the last activity on the session.
Logon Time	The time stamp that shows when the session initially connected to the BI Server.
Repository	The logical name of the repository to which the session is connected.
Session ID	The unique internal identifier that the Oracle BI Server assigns each session when the session is initiated.
User	The name of the user connected.

Table A–2 Some Fields in the Request Tab of the Session Manager Dialog

Column Name	Description
Last Active Time	The time stamp of the last activity on the query.
Request ID	The unique internal identifier that the BI Server assigns each query when the query is initiated.
Session ID	The unique internal identifier that the BI Server assigns each session when the session is initiated.
Start Time	The time of the individual query request.

To view the variables for a session:

1. In the Oracle BI Administration Tool, open a repository in online mode and select **Manage** then **Sessions**.
2. Select a session and click the **Variables** tab.

   For more information about variables, see "Using Variables in the Oracle BI Repository" in *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.
3. To refresh the view, click **Refresh**.
4. To close Session Manager, click **Close**.

To disconnect a user from a session:

1. In the Oracle BI Administration Tool, open a repository in online mode and select **Manage** then **Sessions**.
2. Select the user in the Session Manager top pane.
3. Click **Disconnect**.

   The user session receives a message that indicates that the session was terminated by an administrative user. Any currently running queries are immediately terminated, and any outstanding queries to underlying databases are canceled.
4. To close the Session Manager, click **Close**.

To terminate an active query:

1. In the Oracle BI Administration Tool, open a repository in online mode and select **Manage** then **Sessions**.

2. Select the user session that initiated the query in the top pane of the Session Manager.

   After the user is highlighted, any active query requests from that user are displayed in the bottom pane.

3. Select the request that you want to terminate.

4. Click **Kill Request** to terminate the selected request.

   The user receives a message indicating that the query was terminated by an administrative user. The query is immediately terminated, and any outstanding queries to underlying databases are canceled.

   Repeat this process to terminate any other requests.

5. To close the Session Manager, click **Close**.

A.2 Alternative Authorization Options

This release supports for backward compatibility the ability to manage Presentation Catalog object privileges using catalog groups.

A.2.1 Changes Affecting Security in Presentation Services

If you have upgraded from a previous release, the best practice is to begin managing Presentation Catalog privileges and catalog objects using Application Roles maintained in the policy store.

Oracle Business Intelligence uses the Oracle Fusion Middleware security model and its resources are protected by a role-based system. This has significance for upgrading users as the following security model changes affect Presentation Services Catalog privileges:

- Authorization is now based on fine-grained JAAS permissions. Users are granted permissions by membership in corresponding Application Roles.
- Users and groups are maintained in the identity store and are no longer maintained in BI Server. Members of BI Server groups are no longer automatically made members of Presentation Catalog groups having the same name, as was the practice in earlier releases.
- Presentation Catalog privileges continue to be stored on the BI Presentation Server and cannot be accessed from the administrative interfaces used to manage the policy store.
- The Everyone Presentation Catalog group is no longer available and has been replaced by the AuthenticatedUser Application Role. Members of the Everyone catalog group automatically become members of AuthenticatedUser role after upgrade.
- Presentation Catalog groups can no longer be password protected. All catalog groups migrated during upgrade no longer have a password.

A.2.2 Managing Presentation Catalog Privileges Using Catalog Groups

Existing catalog groups are migrated during upgrade and available for your use. You can continue to create new catalog groups. For information about how to create, edit, or delete catalog groups, see *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*.

You can grant these privileges by mapping other catalog groups, users, or Application Roles to a catalog group.

> **Note:** Mapping catalog groups to become members of an Application Role creates complex group inheritance and maintenance situations, and is not considered a best practice.

To grant privileges using a catalog group:

1. From the Home page in Presentation Services, select **Administration**.
2. Click the **Manage Privileges** link to access the **Manage Privileges** dialog.
3. Click the link for the privilege from the Manage Privileges dialog.
4. To map the privilege to the catalog group:
   - Click **Add Users/Roles**.
   - Select **Catalog Groups** from the list and click **Search**.
   - Select the catalog group from the results list.
   - Use the shuttle controls to move the catalog group to **Selected Members**.
5. Click **OK**.
6. Set the permission for the catalog group by selecting **Granted** or **Denied** in the Privileges dialog.

   Explicitly *denying* a Presentation Services privilege takes precedence over user access rights either granted or inherited as a result of group or Application Role hierarchy.
7. Click **OK**.
8. Repeat steps 3 through 8 until the privileges have been granted or denied as needed.

B

Understanding the Default Security Configuration

Controlling access to system resources is achieved by requiring users to authenticate at log in (**authentication**) and by restricting users to only the resources for which they are authorized (**authorization**). The Oracle Business Intelligence default security configuration is automatically configured during installation and is available for use afterwards. The default configuration includes preconfigured security providers for managing user identities, credentials, and permission grants.

This chapter contains the following sections:

> **Note:** Unless otherwise stated, the permissions discussed in this chapter are those maintained in the policy store provider, such as the Oracle Business Intelligence permissions. Presentation Catalog privileges and permissions are distinct because they are maintained in Oracle BI Presentation Server. For more information about Presentation Catalog privileges and permissions, see Chapter 3, "Configuring Oracle BI to use Oracle Internet Directory".

B.1 About Securing Oracle Business Intelligence

Securing Oracle Business Intelligence can be broken down into two broad areas:

- System access security: Controlling access to the components and features that make up Oracle Business Intelligence.
- Data access security: Controlling access to business source data and metadata used by Oracle Business Intelligence.

System access security is discussed in this guide and topics include how to limit system access to authorized users, control software resources based on permission grants, and enable secure communication among components.

Data access security is discussed in *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

B.2 About the Security Framework

The Oracle Fusion Middleware security model is built upon the Oracle Fusion Middleware platform, which incorporates the Java security model. The Java model is a role-based, declarative model that employs container-managed security where resources are protected by roles that are assigned to users. However, extensive knowledge of the Java-based architecture is unnecessary when using the Oracle Fusion Middleware Security model. By being based upon this security model, Oracle Business Intelligence can furnish uniform security and identity management across the enterprise.

Oracle Business Intelligence is installed into a Oracle WebLogic Server domain during installation, which is a logically related group of resources that are managed as a unit. During a Simple installation type, an Oracle WebLogic Server domain named bifoundation_domain is created and Oracle Business Intelligence is installed into this domain. This name might vary depending upon the installation type performed. One instance of Oracle WebLogic Server in each domain is configured as an Administration Server. The Administration Server provides a central point for managing an Oracle WebLogic Server domain. The Administration Server hosts the Administration Console, which is a Web application accessible from any supported Web browser with network access to the Administration Server. Oracle Business Intelligence uses the active security realm configured for the Oracle WebLogic Server domain into which it is installed. For more information, see Section B.2.2, "Oracle WebLogic Server Domain".

For more information about the Oracle Fusion Middleware platform and the common security framework, see *Oracle Fusion Middleware Security Guide*. For more information about managing the Oracle WebLogic Server domain and security realm, see *Oracle Fusion Middleware Understanding Security for Oracle WebLogic Server* and *Oracle Fusion Middleware Securing Oracle WebLogic Server*.

B.2.1 Oracle Platform Security Services

Oracle Platform Security Services is the underlying platform on which the Oracle Fusion Middleware security framework is built. Oracle Platform Security Services is standards-based and complies with role-based-access-control (RBAC), Java Enterprise Edition (Java EE), and Java Authorization and Authentication Service (JAAS). Oracle Platform Security Services enables the shared security framework to furnish uniform security and identity management across the enterprise.

For more information about Oracle Platform Security Services, see *Oracle Fusion Middleware Security Guide*.

B.2.2 Oracle WebLogic Server Domain

An Oracle WebLogic Server administration domain is a logically related group of Java components. A domain includes a special WebLogic Server instance called the Administration Server, which is the central point from which you configure and manage all resources in the domain. You typically configure a domain to include additional WebLogic Server instances called Managed Servers. You deploy Java components, such as Web applications, EJBs, and Web services, and other resources to the Managed Servers and use the Administration Server for configuration and management purposes only.

Oracle WebLogic Server Administration Console and Oracle Enterprise Manager Fusion Middleware Control run in the Administration Server. Oracle WebLogic Server Administration Console is the Web-based administration console used to manage the resources in an Oracle WebLogic Server domain, including the Administration Server and Managed Servers. Fusion Middleware Control is a Web-based administration console used to manage Oracle Fusion Middleware, including the components that comprise Oracle Business Intelligence. For more information about the Oracle Business Intelligence individual components, see *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*.

Oracle Business Intelligence authentication is handled by the Oracle WebLogic Server authentication providers. An authentication provider performs the following functions:

- Establishes the identity of users and system processes
- Transmits identity information

Upon installation, Oracle Business Intelligence is configured to use the directory server embedded in Oracle WebLogic Server as both the default authentication provider and the repository for users and groups. Alternate authentication providers can be used if desired, and managed in the Oracle WebLogic Administration Console. For more information, see System Requirements and Certification.

B.3 Key Security Elements

The Oracle Fusion Middleware security platform depends upon the following key elements to provide uniform security and identity management across the enterprise. For more information about the Oracle Fusion Middleware security platform, see *Oracle Fusion Middleware Security Guide*.

Oracle Business Intelligence uses these security platform elements as follows:

Application Policy

Oracle Business Intelligence permissions are granted to members of its Application Roles. In the default security configuration, each role conveys a predefined set of permissions. Permission grants are defined and managed in an **Application Policy**. After an Application Role is associated with an Application Policy, that role becomes a **grantee** of the policy. An Application Policy is specific to a particular application.

An **application stripe** defines a subset of policies in the policy store. The Oracle Business Intelligence application stripe is named **obi**.

Application Role

An **Application Role** represents a *role* a user has in Oracle Business Intelligence and gives that user authorization to access system resources accordingly. For example, having the Sales Analyst Application Role can grant a user access to view, edit and create reports relating to a company's sales pipeline.The default security configuration provides four preconfigured roles that grant the permissions corresponding to the common types of work performed when using Oracle Business Intelligence. The Application Role is also the *container* used to grant permissions and access to its members. When members are mapped to an Application Role, that Application Role becomes the container used to convey access rights to its members. For example:

- Oracle Business Intelligence Permissions: These permission grants are defined in an Application Policy. After an Application Role is mapped to a policy, the permissions become associated with the Application Role through the relationship between policy and role. If groups of users have been mapped to that Application

Role, the corresponding permissions are in turn granted to all members equally. More than one user or group can be members of the same Application Role.

- Data Access Rights: Application roles can be used to control access rights to view and modify data in the repository file. Data filters can be applied to Application Roles to control object level permissions in the Business Model and Mapping layer and the Presentation layer. For more information about using Application Roles to apply data access security and control repository objects, see *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.
- Presentation Services Object-Level Access: Application roles can be used to grant access rights to reports and other objects in Oracle BI Presentation Services. For more information about using Application Roles to control access in Presentation Services, see *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*.

Authentication Provider

User authentication is performed by an authentication provider. The Oracle Business Intelligence default security configuration authenticates against the Oracle WebLogic Server embedded directory server using an authentication provider named DefaultAuthenticator.

B.4 Default Security Configuration

When operating in a development or test environment you might find it convenient to use the default security configuration because it comes preconfigured, then add user definitions and credentials specific to your business, and customize the default Application Roles and permission grants to meet your requirements. After the authentication, policy, and credential providers are fully configured and populated with data specific to your business, they provide all user, policy, and credential information needed by the Oracle Business Intelligence components during authentication and authorization.

The default security configuration provides you with three security providers that are integrated to ensure safe, controlled access to system and data resources. These security providers are configured during a Simple or Enterprise installation type as follows:

- The authentication provider is DefaultAuthenticator, which authenticates against Oracle WebLogic Server embedded directory server (identity store). The directory server is preconfigured with the default users and groups supplied by Oracle Business Intelligence, as well as a user group needed for the embedded directory server. The default identity store is managed using Oracle WebLogic Server Administration Console.
- The policy store provider is the system-jazn-data.xml file. It contains the default Application Role definitions with their corresponding Oracle Business Intelligence permission grants, and the mapping definitions between default groups and Application Roles. The mapping of a group to an Application Role serves to convey the corresponding permissions to members of the group. The default policy store provider is managed using Oracle Enterprise Manager Fusion Middleware Control.
- The credential store provider is the cwallet.sso file. It contains the passwords and other security-related credentials either supplied or system-generated. The default credential store is managed using Fusion Middleware Control.

Table B–1 summarizes the three default security providers and their initial state after installation.

Table B–1 Default Security Providers

Security Provider Type	Purpose	Default Provider	Options
Authentication provider	Used to control authentication.	▪ DefaultAuthenticator. Authenticates against the users and groups stored in Oracle WebLogic Server embedded directory server (identity store). ▪ Oracle WebLogic Server embedded directory server is managed with Oracle WebLogic Server Administration Console.	Oracle Business Intelligence can be reconfigured to use different authentication providers and directory servers. For more information, see System Requirements and Certification.
Policy store provider	▪ Used to control authorization. ▪ Contains the definition of Application Roles, Application Policies, and the members mapped to Application Roles.	▪ system.jazn-data.xml file. ▪ Managed with Fusion Middleware Control.	Oracle Business Intelligence can be configured to use Oracle Internet Directory.
Credential store provider	Trusted store for holding system passwords and other security-related credentials. The data stored here is used for connecting to external systems, opening repositories, or for SSL.	▪ cwallet.sso. ▪ File is automatically replicated across all machines in the Oracle Business Intelligence installation. ▪ Managed with Fusion Middleware Control.	Oracle Business Intelligence can be configured to use Oracle Internet Directory.

Figure B–1 shows the relationship between Oracle Business Intelligence and the authentication and policy store providers.

Figure B–1 Relationship with the Default Security Providers

B.4.1 Default Policy Store Provider

The policy store provider contains the Oracle Business Intelligence application-specific policies, Application Roles, permission grants, and membership mappings configured during installation. A policy store can be file-based or LDAP-based, but the installation default provides a policy store that is an XML file.

Presentation Catalog privileges and permissions are not maintained in the policy store provider. For more information about them, see Chapter 3, "Configuring Oracle BI to use Oracle Internet Directory".

B.4.1.1 Default Permissions

All Oracle Business Intelligence permissions are provided; you cannot create additional permissions. In the default configuration, the Application Policies and Application Roles are preconfigured to group these permissions according to the access requirements of the Oracle Business Intelligence common user types: administrator, author, and consumer. However, these default permission grants can be changed as needed using Fusion Middleware Control. For more information, see Section 3.3, "Configuring an Alternative Policy Store and Credentials Store".

Table B–2 and Table B–3 list the available permissions and resource types that are contained in the obi application stripe.

Table B–2 Default Permissions

Permission Name	Description
oracle.bi.publisher.administerServer	Enables the Administration link to access the Administration page and grants permission to set any of the system settings.
oracle.bi.publisher.developDataModel	Grants permission to create or edit data models.
oracle.bi.publisher.developReport	Grants permission to create or edit reports, style templates, and sub templates. This permission also enables connection to the BI Publisher server from the Template Builder.
oracle.bi.publisher.runReportOnline	Grants permission to open (execute) reports and view the generated document in the report viewer.
oracle.bi.publisher.scheduleReport	Grants permission to create or edit jobs and also to manage and browse jobs.
oracle.bi.publisher.accessReportOutput	Grants permission to browse and manage job history and output.

Table B–2 (Cont.) Default Permissions

Permission Name	Description
oracle.bi.publisher.accessExcelReportAnalyzer	Grants permission to download the Analyzer for Excel and to download data from a report to Excel using the Analyzer for Excel. Note that to enable a user to upload an Analyzer for Excel template back to the report definition, the permission oracle.bi.publisher.developReport must also be granted.
oracle.bi.publisher.accessOnlineReportAnalyzer	Grants permission to launch the Analyzer and manipulate the data. Note that to save an Analyzer template to a report definition, the permission oracle.bi.publisher.developReport must also be granted.
oracle.bi.server.impersonateUsers	This description is not available.
oracle.bi.server.manageRepositories	Grants permission to open, view, and edit repository files using Oracle BI Administration Tool.
oracle.bi.server.queryUserPopulation	Internal use only.
oracle.bi.scheduler.manageJobs	Grants permission to use Job Manager to manage scheduled Delivers jobs.
EPM_Calc_Manager_Designer	Grants permissions for EPM Calc Manager Designer.
EPM_Calc_Manager_Administrator	Grants permissions for EPM Calc Manager Administrator.
EPM_Essbase_Filter	Grants permissions for EPM Essbase Filter.
EPM_Essbase_Administrator	Grants permissions for EPM Essbase Administrator.
oracle.epm.financialreporting.accessReporting	Grants permissions for EPM Report Access.
oracle.epm.financialreporting.administerReporting	Grants permissions for EPM Report Administration.
oracle.epm.financialreporting.editBatch	Grants permissions for EPM Batch Edit.
oracle.epm.financialreporting.editBook	Grants permissions for EPM Book Edit.
oracle.epm.financialreporting.editReport	Grants permissions for EPM Report Edit.
oracle.epm.financialreporting.scheduleBatch	Grants permissions for EPM Batch Scheduling.

Oracle RTD controls authorization using *resources* defined in context of a Java class. The Java class oracle.security.jps.ResourcePermission can be used as the permission class within any grant to protect application or system resources. Oracle RTD uses this class to control access to three types of resource:

- Inline Service
- Decision Center Perspective
- Batch Job

Table B–3 lists the Oracle RTD resource types. For more information about Real-Time Decision (RTD) resources, see "Security for Oracle Real-Time Decisions" in *Oracle Fusion Middleware Administrator's Guide for Oracle Real-Time Decisions*

Table B–3 Oracle RTD Resource Types and Actions

Type of Resource	Resource Type Name Stored in Application Grants	Action[:Qualifier]	Comments
Inline Service	rtd_ils	choice_editor	might execute any methods of the **ExternalChoice** web service for the named Inline Service.
		decision_ service:normal	might execute any integration points (advisors and informants) for the named Inline Service. Action qualifier **normal** allows integration point requests to be executed in the server.
		decision_ service:stress	might execute any integration points (Advisors and Informants) for the named Inline Service. Action qualifier **stress** allows LoadGen to issue integration point calls. To be accepted by the server, the user also needs the **normal** action.
		open_service:read	Authorizes the use of Decision Center to open the named Inline Service for viewing. Also authorizes the External Rule Editor to access the named Inline Service, since the External Rule Editor does not need to update the content of the Inline Service.
		open_ service:write	Authorizes the use of Decision Center to open the named Inline Service for editing.
		deploy_service	Authorizes the deployment of the named Inline Service from Decision Studio.
		download_service	Authorizes the use of Decision Studio to download the named Inline Service from a server.
Decision Center Perspective	rtd_dc_persp	dc_perspective	Open the named Decision Center Perspective, to have Decision Center render its specialized set of UI elements or capabilities.
Registered Batch Job Type	rtd_batch	batch_admin	might execute any methods of the BatchManager web service to start, stop, or query the status of the registered batch job type name.

B.4.1.2 Default Application Roles

The default Application Roles are grouped into broad categories of functional usage: administrator (BIAdministrator), author (BIAuthor), and consumer (BIConsumer). These categories correspond to the typical roles that users of Oracle Business Intelligence assume: an *administrator*, an *author* who creates reports for others, and a *consumer* who reads (consumes) reports created by others (authors).

The default Oracle Business Intelligence Application Roles are as follows:

BIAdministrator Role

The BIAdministrator role grants administrative permissions necessary to configure and manage the Oracle Business Intelligence installation. Any member of the BIAdministrators group is explicitly granted this role and implicitly granted the BIAuthor and BIConsumer roles. See Table B–4 and Table B–5 for a list of the default role permissions.

BIAuthor Role

The BIAuthor role grants permissions necessary to create and edit content for other users to use, or to consume. Any member of the BIAuthors group is explicitly granted this role and implicitly granted the BIConsumer role. See Table B–4 and Table B–5 for a list of the default role permissions.

BIConsumer Role

The BIConsumer role grants permissions necessary to use, or to consume, content created by other users. See Table B–4 and Table B–5 for a list of the default role permissions.

BISystem Role

The BISystem role grants the permissions necessary to impersonate other users. This role is required by Oracle Business Intelligence system components for inter-component communication. See Table B–4 and Table B–5 for a list of the default role permissions.

Authenticated Role

The authenticated role is a special Application Role provided by the Oracle Fusion Middleware security model and is made available to any application deploying this security model. Oracle Business Intelligence uses the authenticated Application Role to grant permissions implicitly derived by the role and group hierarchy of which the authenticated role is a member. The authenticated role is a member of the BIConsumer role by default and, as such, all authenticated role members are granted the permissions of the BIConsumer role implicitly.

Every user who successfully logs in to Oracle Business Intelligence becomes a member of the authenticated role, and it is a replacement for the 10*g* Everyone Presentation Catalog group. The authenticated role is not stored in the obi application stripe and is not searchable in the Oracle Business Intelligence policy store. However, the authenticated role is displayed in the administrative interface for the policy store, is available in Application Role lists, and can be added as a member of another Application Role.

You can map the authenticated role to another user, group, or Application Role, but you cannot remove the authenticated role itself. Removal of the authenticated role would result in the inability to log in to the system and this right would need to be granted explicitly.

For more information about the Oracle Fusion Middleware security model and the authenticated role, see *Oracle Fusion Middleware Security Guide*.

B.4.1.3 Default Application Roles, Permission Grants, and Group Mappings

The default file-based policy store is configured with the Oracle Business Intelligence default Application Roles. Each Application Role is preconfigured with a set of permissions grants and one or more members. Members of an Application Role can include users, groups, or other Application Roles from the policy store.

Table B–4 and Table B–5 lists the default configuration of Application Roles, permission grants, and members. The default naming convention is that Application Role names are singular and group names are plural.

Table B–4 Default Application Role, Permission Grants, and Members

Role Name	Role Permissions	Members
BIAdministrator	▪ oracle.bi.server.manageRepositories ▪ oracle.bi.scheduler.manageJobs ▪ oracle.bi.publisher.administerServer ▪ EPM_Calc_Manager_Administrator ▪ oracle.epm.financialreporting.administerReporting	BIAdministrators group
BIAuthor	▪ oracle.bi.publisher.developReport ▪ oracle.bi.publisher.devlopDataModel ▪ EPM_Essbase_Administrator ▪ EPM_Calc_Manager_Designer ▪ oracle.epm.financialreporting.editBatch ▪ oracle.epm.financialreporting.editBook ▪ oracle.epm.financialreporting.editReport ▪ oracle.epm.financialreporting.scheduleBatch	▪ BIAuthors group ▪ BIAdministrator Application Role
BIConsumer	▪ oracle.bi.publisher.accessExcelReportAnalyzer ▪ oracle.bi.publisher.accessOnlineReportAnalyzer ▪ oracle.bi.publisher.runReportOnline ▪ oracle.bi.publisher.accessReportOutput ▪ oracle.bi.publisher.scheduleReport ▪ EPM_Essbase_Filter ▪ oracle.epm.financialreporting.acessReporting	▪ BIConsumers group ▪ BIAuthor Application Role

Table B–4 (Cont.) Default Application Role, Permission Grants, and Members

Role Name	Role Permissions	Members
BISystem	■ oracle.bi.scheulder.manageJobs ■ oracle.bi.server.manageRepositories ■ oracle.bi.server.impersonateUser ■ oracle.bi.server.queryUserPopulation	BISystemUser

Table B–5 lists the default Application Roles, Oracle RTD resource types, resource names, and actions in the default application grants after installation. For more information about Real-Time Decision (RTD) resource defaults, see "Security for Oracle Real-Time Decisions" in *Oracle Fusion Middleware Administrator's Guide for Oracle Real-Time Decisions*

Note: The resource name **_all** _ is a special name that matches any Oracle RTD resource name of the associated resource type.

Table B–5 Default Application Grants for Oracle RTD Users

Application Role	Resource Type	Resource Name	Action[:Qualifier]
BIAdministrator	rtd_ils	_all_	open_service:read
		all	open_service:write
		all	deploy_service
		all	download_service
		all	choice_editor
		all	decision_service:normal
		all	decision_service:stress
	rtd_dc_persp	_all_	dc_perspective
	rtd_batch	_all_	batch_admin
BIAuthors	rtd_ils	_all_	open_service:read
		all	open_service:write
		all	deploy_service
		all	download_service
		all	decision_service:normal
		all	decision_service:stress
	rtd_dc_persp	_all_	dc_perspective

Table B–5 (Cont.) Default Application Grants for Oracle RTD Users

Application Role	Resource Type	Resource Name	Action[:Qualifier]
BIConsumer	rtd_ils	_all_	open_service:read
		all	choice_editor
		all	decision_service:normal
	rtd_dc_persp	Explore	dc_perspective
		At a Glance	dc_perspective
	rtd_batch	_all_	batch_admin

B.4.2 Default Authentication Provider

An **authentication provider** accesses user and group information and is responsible for authenticating users. An **identity store** contains user name, password, and group membership information and in Oracle Business Intelligence is a directory server. The default security configuration authenticates against the Oracle WebLogic Server embedded directory server using an authentication provider named DefaultAuthenticator.

When a user logs in to a system with a user name and password combination, Oracle WebLogic Server validates identity based on the combination provided. During this process, a Java principal is assigned to the user or group that is undergoing authentication. The principal can consist of one or more users or groups and is stored within subjects. A **subject** is a JAAS element used to group and hold identity information.

Upon successful authentication, each principal is signed and stored in a subject. When a program call accesses a principal stored in a subject, the default authenticator provider verifies the principal has not been altered since signing, and the principal is returned to the program making the call. For example, in the Oracle WebLogic Server default authenticator, the subject contains a principal for the user (WLSUserPrincipal) and a principal for the group (WLSGroupsPrincipals) of which the user is a member. If an authentication provider other than the installation default is configured, consult that provider's documentation because how identity information is stored might differ.

B.4.2.1 Default Groups and Members

Groups are logically ordered sets of users. Creating groups of users who have similar system resource access needs enables easier security management. Managing a group is more efficient than managing a large number of users individually. Oracle recommends that you organize your users into groups for easier maintenance. Groups are then mapped to Application Roles to grant rights.

The default group names discussed here are provided as a convenience so you can begin using the Oracle Business Intelligence software immediately after installation, but you are not required to maintain the default names.

Table B–6 lists the group names and group members that are created during the installation process. These defaults can be changed to different values and additional group names can be added by an administrative user using Oracle WebLogic Server Administration Console.

Table B–6 Default Groups and Members

Purpose	Group Name and Members	Description
Contains the Oracle Business Intelligence administrative users.	**Name**: BIAdministrators **Members**: Any *administratror user*	■ Members of the BIAdministrators group are granted administrative permissions because this group is mapped to the BIAdministrator Application Role at installation. ■ All users requiring administrative permissions should be added to the BIAdministrators group when using the default security configuration.
Contains the Oracle Business Intelligence authors.	**Name**: BIAuthors **Members**: BIAdministrators Group	Members of the BIAuthors group have the permissions necessary to create content for other users to use, or to consume.
Contains the Oracle Business Intelligence consumers.	**Name**: BIConsumers **Members**: BIAuthors group and Oracle WebLogic Server LDAP server users group	■ Members of the BIConsumers group have the permissions necessary to use, or consume, content created by other users. ■ The BIConsumers group represents all users that have been authenticated by Oracle Business Intelligence. By default, every authenticated user is automatically added to this group. ■ Oracle WebLogic Server LDAP server users group members have the permissions necessary to log in to and use Oracle WebLogic Server Administration Console.

B.4.2.2 Default Users and Passwords

Oracle WebLogic Server embedded directory server contains Oracle Business Intelligence user names provided as part of the default security configuration. These default user names are provided as a convenience so you can begin using the Oracle

Business Intelligence software immediately after installation, but you are not required to maintain the default names.

Table B–7 lists the default user names and passwords in the Oracle WebLogic Server embedded directory server after installation.

Table B–7 Default Users and Passwords

Purpose	User Name and Password	Description
Administrative user	**Name**: *administrator user* **Password**: *user supplied*	▪ This user name is entered by the person performing the installation, it can be any desired name, and does not need to be named Administrator. ▪ The password entered during installation can be changed later using the administration interface for the identity store provider. ▪ An administrative user is a member of the BIAdministrators group and has all rights granted to the Oracle Business Intelligence Administrator user in earlier releases, except impersonation. The administrator user cannot impersonate other users. ▪ The single administrative user is shared by Oracle Business Intelligence and Oracle WebLogic Server. This user is automatically made a member of the Oracle WebLogic Server default Administrators group after installation. This enables this user to perform all Oracle WebLogic Server administration tasks, including the ability to manage Oracle WebLogic Server embedded directory server.

Table B–7 (Cont.) Default Users and Passwords

Purpose	User Name and Password	Description
▪ A fixed user created during installation for trusted communication between components. ▪ All Oracle Business Intelligence system components run as this user.	**Name**: BISystemUser **Password**: *system generated*	▪ This is a highly privileged user whose credentials should be protected from non-administrative users. ▪ Using a separate user for secure inter-component communication enables you to change the password for the system administrator account without affecting communication between components. ▪ The name of this user can be changed or a different user can be created for inter-component communication.

B.4.3 Default Credential Store Provider

A **credential store** is a repository of security data (credentials) that validates the authority of users, Java components, and system components. Oracle Business Intelligence system processes use these credentials to establish trusted communication.

B.4.3.1 Default Credentials

The Oracle Business Intelligence default credential store is file-based, also known as being *wallet-based*, and is represented by the file cwallet.sso. The default credential store is managed in Fusion Middleware Control.

Credentials are grouped into logical collections called maps. The default security configuration contains the following maps: oracle.bi.system and oracle. bi.enterprise. Each credential is accessed from a map using a key, such as system.user or repository.paint. A key is case sensitive. Each repository file has its own entry in the credential map.

The oracle.bi.actions credential map is created manually. For information about creating the oracle.bi.actions credential map, see "Adding and Maintaining Credentials for Use with Action Framework" in *Oracle Fusion Middleware Integrator's Guide for Oracle Business Intelligence Enterprise Edition*.

Table B–8 lists the credentials contained in the default credential store after installation.

Table B–8 Default Credentials

Description	Map and Key	User Name and Password
RPD password	**map**: oracle.bi.enterprise **key**: repository.*RPD name*	**Name**: Not Applicable **Password**: *user supplied*
BISystem user	**map**: oracle.bi.system **key**: system.user	**Name**: BISystemUser **Password**: system generated
Oracle Business Intelligence Scheduler Schema user	**map**: oracle.bi.enterprise **key**: scheduler.schema	**Name**: Name of Scheduler schema **Password**: *system generated*

B.4.4 How Permissions Are Granted Using Application Roles

Oracle Business Intelligence permissions are typically granted by becoming a member in an Application Role. LDAP groups become members by being mapped to Application Roles. In the default security configuration, each Application Role is preconfigured to grant a predefined set of permissions. The mapping of a group to a role conveys the role's permissions to all members of the group. In short, permissions are granted by Oracle Business Intelligence Application Roles by establishing the following relationships:

- A group defines a set of users having similar system access requirements. Users are added as members to one or more groups according to the level of access required.
- Application roles are defined to represent the role a user typically performs when using Oracle Business Intelligence. The default security configuration provides the following role types: administrator (BIAdministrator), author (BIAuthor), and consumer (BIConsumer).
- The groups of users are mapped to one or more Application Roles that match the type of access required by each group.
- Application policies are created with Oracle Business Intelligence permissions that grant a set of access rights corresponding to each role type.
- An Application Role is mapped to the corresponding Application Policy that grants the set of permissions required by the role type (administrator, author, consumer). Once done, the Application Role is the Grantee of the Application Policy.
- Group membership can be inherited by nature of the group hierarchy. Application roles mapped to inherited groups are also inherited, and those permissions are likewise conveyed.

How a user's permissions are determined by the system is as follows:

1. A user enters credentials into a Web browser at login. The user credentials are authenticated by the authentication provider against data contained the identity store.
2. After successful authentication, a Java subject and principal combination is issued, which is populated with the user name and a user's groups.

3. A list of the user's groups is generated and checked against the Application Roles. A list is created of the Application Roles that are mapped to each of the user's groups.
4. A user's permission grants are determined from knowing which Application Roles the user is a member of. The list of groups is generated only to determine what roles a user has, and is not used for any other purpose.

For example, the ability to open a repository file in online mode from Oracle BI Administration Tool requires the manage repository permission (oracle.bi.server.manageRepositories). In the default security configuration, this permission is granted by membership in the BIAdministrator Application Role. The BIAdministrator Application Policy contains the actual permission grant definitions, and in this example, the BIAdministrator Application Policy contains the manage repository permission definition. The default security configuration includes a preconfigured mapping between the BIAdministrator Application Role and the BIAdministrators group. To convey the manage repository permission to a user in your environment, add that user to the BIAdministrators group. Every user who needs to manage a repository in online mode should be added to the BIAdministrators group instead of granting the required permission to each user individually. If a user no longer requires the manage repository permission, you then remove the user from the BIAdministrators group. After removal from the BIAdministrators group, the user no longer has the BIAdministrator Application Role or the manage repository permission granted by role membership.

Users can also obtain permissions by inheriting group membership and Application Roles. For more information and an example of how this is accomplished, see Section B.4.4.1, "Permission Inheritance and Role Hierarchy".

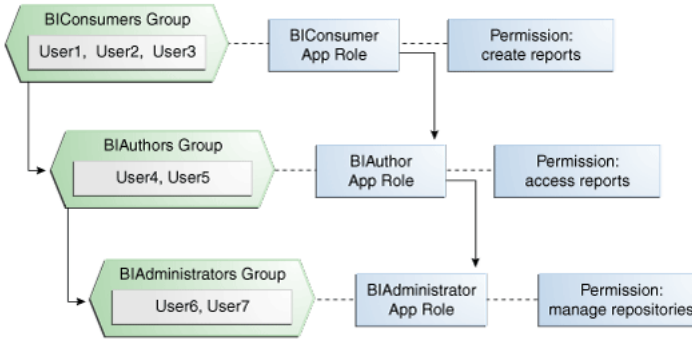
B.4.4.1 Permission Inheritance and Role Hierarchy

In Oracle Business Intelligence, the members of a default Application Role includes both groups and other Application Roles. The result is a hierarchical role structure where permissions can be inherited in addition to being explicitly granted. A group that is a member of a role is granted both the permissions of the role and the permissions for all roles descended from that role. It is important when constructing a role hierarchy that circular dependencies are not introduced.

The following figure provides an example of how the role hierarchy grants permissions using several of the Oracle Business Intelligence default groups and Application Roles. The default BIAdministrator role is a member the BIAuthor role, and BIAuthor role is a member of BIConsumer role. The result is members of the BIAdministrators group are granted *all* the permissions of the BIAdministrator role, the BIAuthor role, and the BIConsumer role. In this example only one of the permissions granted by each role is used for demonstration purposes.

Figure B–2 shows these relationship between the default Application Roles and how permissions are granted to members.

Figure B–2 Default Application Role Hierarchy Example



The result is that, by nature of the role hierarchy, the user who is a member of a particular group is granted both *explicit* permissions and any additional *inherited* permissions.

> **Note:** By themselves, groups and group hierarchies do not provide access rights to application resources. Privileges are conveyed by the permission grants defined in an Application Policy. A user, group, or Application Role becomes a Grantee of the Application Policy. The Application Policy grantee conveys the permissions and this is done by direct association (such as a user) or by becoming a member of the Grantee (such as a group or Application Role).

Table B–9 details the role and permissions granted to all group members (users) shown in Figure B–2.

Table B–9 Permissions Granted by The Role Hierarchy Example

User Name	Group Membership: Explicit/Inherited	Application Role Membership: Explicit/Inherited	Permission Grants: Explicit/Inherited
User1, User2, User3	BIConsumers: Explicit	BIConsumer: Explicit	Access reports: Explicit
User4, User5	BIAuthors: Explicit BIConsumers: Inherited	BIAuthor: Explicit BIConsumer: Inherited	Create reports: Explicit Access reports: Inherited
User6, User7	BIAdministrators: Explicit BIAuthors: Inherited BIConsumers: Inherited	BIAdministrator: Explicit BIAuthor: Inherited BIConsumer: Inherited	Manage repository: Explicit Create reports: Inherited Access Reports: Inherited

B.4.4.2 Presentation Catalog Groups and Precedence

If *catalog groups* and Application Roles are used in combination to manage Presentation Services Catalog permissions or privileges, the catalog groups take precedence. For example, if a user is a member of a catalog group that grants access to a Presentation Services object or feature and is also a member of an Application Role that denies access to the same object or feature, then this user has access. A Presentation Services Catalog group takes precedence over an Application Role. For

more information about Presentation Services permissions and privileges, see Chapter 3, "Configuring Oracle BI to use Oracle Internet Directory".

B.5 Common Security Tasks After Installation

The common security tasks performed after a successful Oracle Business Intelligence software installation are different according to purpose. Common reasons to install Oracle Business Intelligence are:

- Evaluate the product
- Implement the product

Implementation typically involves moving through the product lifecyle of using the product in one or more of the following environments:

- Development
- Test
- Production

B.5.1 Common Security Tasks to Evaluate Oracle Business Intelligence

Table B–10 contains common security tasks performed to evaluate Oracle Business Intelligence and provides links for more information.

Table B–10 Task Map: Common Security Tasks to Evaluate Oracle Business Intelligence

Task	Description	For Information
Understand the Oracle Fusion Middleware security model and the Oracle Business Intelligence default security configuration.	Familiarize yourself with the key elements of the Oracle Fusion Middleware security model and the Oracle Business Intelligence default security configuration after a successful installation.	Chapter 1, "Introduction to Security in Oracle Business Intelligence" Section B.4, "Default Security Configuration" *Oracle Fusion Middleware Security Guide*
Add users and groups to the default identity store.	Create new user and group definitions for the embedded directory server using Oracle WebLogic Server Administration Console.	Section 2.4.3, "How to create a User in the Embedded WebLogic LDAP Server" *Oracle Fusion Middleware Oracle WebLogic Server Administration Console Online Help*
Add a new member to a default Application Role.	Add a new user or group as a member to a default Application Role, such as BIConsumer.	Section 2.5.4, "Modifying Application Roles Using Oracle Fusion Middleware Control" Section B.4.1.3, "Default Application Roles, Permission Grants, and Group Mappings" *Oracle Fusion Middleware Security Guide*
Create a new Application Role based on an existing default Application Role.	Create a new Application Role based on an existing default Application Role by copying it and naming the copy.	Section 2.5.2, "Creating Application Roles Using Fusion Middleware Control" *Oracle Fusion Middleware Security Guide*

B.5.2 Common Security Tasks to Implement Oracle Business Intelligence

Table B–11 contains common security tasks performed when you implement Oracle Business Intelligence and provides links for more information. The following tasks are performed in addition to the tasks listed in Section B.5.1, "Common Security Tasks to Evaluate Oracle Business Intelligence".

Table B–11 Task Map: Common Security Tasks to Implement Oracle Business Intelligence

Task	Description	For Information
Transition to using your enterprise directory server as the authentication provider and identity store.	Configure your enterprise directory server to become the authentication provider and identity store.	Section 3.2, "Configuring an Alternative Authentication Provider" Appendix A, "Alternative Security Administration Options"
Create a new Application Role.	Create a new Application Role and make the role a Grantee of an Application Policy.	Section 2.5.2, "Creating Application Roles Using Fusion Middleware Control"
Map a group to a newly created Application Role.	Map a group to a newly created Application Role to convey the permission grants to group members.	Section 2.5.4, "Modifying Application Roles Using Oracle Fusion Middleware Control"
Decide whether to use SSL.	Decide whether to use SSL communication and devise a plan to implement.	Chapter 5, "SSL Configuration in Oracle Business Intelligence"
Decide whether to use an SSO provider in your deployment.	Decide whether to use SSO authentication and devise a plan to implement.	Chapter 4, "Enabling SSO Authentication"

B.6 About the Default Security Configuration After Upgrade

The Upgrade Assistant is a unified graphical user interface that enables you to selectively upgrade your Oracle Business Intelligence installation. For complete upgrade information, see *Oracle Fusion Middleware Upgrade Guide for Oracle Business Intelligence Enterprise Edition*.

Significant changes have been made to the security model regarding how and where users, groups, and credentials are defined and stored. The following is a summary of some of the changes that are made during the upgrade process by the Upgrade Assistant:

- Users, passwords, and groups are moved from the default release 10*g* repository file to the release 11*g* default identity store (Oracle WebLogic Server embedded LDAP server).
- Passwords for other repository objects, such as connection pools and LDAP servers, remain in the repository and are encrypted. The repository itself is encrypted as well.
- The Administrator user is migrated from the default release 10*g* repository file to the default identity store and becomes a member of the BIAdministrators group. The BIAdministrators group is granted the BIAdministrator role and by that association has system administrative rights.
- Presentation Catalog references to old groups and users are updated.

- The variable names ROLES, PERMISSIONS, USERGUID and ROLEGUIDS are reserved release 11*g* system variable names. Before upgrading a release 10*g* repository file, these variables must be renamed if they exist. Other references to these variable names, as in reports, also must be renamed for consistency.

Caution: Before upgrading, create a backup of the repository file and the Presentation Catalog to ensure that you can restore the originals if needed.

B.6.1 Security-Related Changes After Upgrading

The following is an overview of the security-related changes initiated by the Upgrade Assistant when upgrading an Oracle Business Intelligence installation. For information about upgrading a system, see *Oracle Fusion Middleware Upgrade Guide for Oracle Business Intelligence Enterprise Edition*.

In general, the standard upgrade process is as follows. The Upgrade Assistant is run on a system that has the Oracle Business Intelligence release 11*g* software installed. During this process the metadata from the release 10*g* repository file and Presentation Catalog is imported to the release 11*g* system. The release 10*g* system is left unchanged after the upgrade process completes. The imported metadata is upgraded as needed to function in the release 11*g* environment, such as moving users and groups defined in the repository to the Oracle WebLogic Server embedded LDAP server, and so on. However, configuration settings such as SSL settings are not carried over from the upgrade source.

Before running the Upgrade Assistant you must have the following available:

- The Oracle Business Intelligence release 10*g* installation, which is used as the upgrade source. This installation can be configured to use any combination of security mechanisms supported in the release 10*g*, including: repository users and groups, authentication initialization blocks, catalog groups, and SA System Subject Area.
- A default installation of Oracle Business Intelligence release 11*g* to be used as the target for the upgrade. This installation must not have been customized in any way.

The Upgrade Assistant prompts for details of the release 10*g* installation. The Upgrade Assistant migrates the existing security-related entries to the release 11*g* system, as explained in the following sections.

B.6.1.1 Changes Affecting the Identity Store

The Upgrade Assistant automatically creates the following entries in the Oracle WebLogic Server embedded LDAP server for the target system:

- An LDAP group corresponding to each group found in the repository. This does not include the Administrators group found in prior releases. Any users that were in this Administrators group are added to the BIAdministrators LDAP group.
- LDAP group hierarchies that match the repository group hierarchies.
- The Administrator user is migrated and made a part of the BIAdministrators group.

All users, other than the Administrator user, who are members of the Administrators group in the default repository are added to the BIAdministrators group in the embedded LDAP server. The release 11*g* Administrator user that is created from

information provided during installation is also added to the BIAdministrators group in the embedded LDAP server.

B.6.1.2 Changes Affecting the Policy Store

The Upgrade Assistant automatically creates the following entries in the file-based policy store for the target system:

- An Application Role that corresponds to each group in the default repository. This does not include the Administrators group found in prior releases. The Application Role is granted to the group with the same name.
- Application role hierarchies that match the repository group hierarchies.

B.6.1.3 Changes Affecting the Default Repository File

The upgrade assistant automatically upgrades the default repository in the source system and makes the following changes:

- All groups in the default release 10*g* repository are converted to Application Role references (placeholders) to Application Roles created in the policy store during upgrade.
- All users are removed from the default repository during upgrade and replaced with references (name and GUID) to LDAP users created in the embedded LDAP server on the target system.
- A numerical suffix is added to the name of an upgraded repository file. A number is added to indicate the number of times that file has been upgraded.

B.6.1.4 Changes Affecting the Oracle BI Presentation Catalog

The Upgrade Assistant automatically makes the following changes to the Presentation Catalog:

- The Presentation Catalog is scanned and the old security representations are converted to the new ones. Permissions and privileges that existed in 10*g* are migrated. Updates the internal representation of each user to the standard GUID being used across the environment. Users not found in the LDAP server are placed in the initialization block users folder until they have been added to the LDAP server, after which they are moved to the standard user folder. All references to old user and group representation are replaced by the GUID. The entire Presentation Catalog is reviewed.
- Leaves the release 10*g* catalog groups in the upgraded Presentation Catalog and assigns the same privileges, access, and membership.

B.6.2 Planning to Upgrade a 10*g* Repository

A release 10*g* repository can be opened and upgraded using the Upgrade Assistant. The following security-related changes are made to the repository upon upgrade:

- The upgraded repository is now protected and encrypted by the password entered during the upgrade.
- The repository file is upgraded to contain references to users it expects to be present in the identity store and references to Application Roles it expects to be present in the policy store.

The upgraded repository can be opened in the Oracle BI Administration Tool in offline mode as usual, and can be deployed to a server to be opened in online mode.

For more information about upgrading a release 10*g* repository, see *Oracle Fusion Middleware Upgrade Guide for Oracle Business Intelligence Enterprise Edition*.

B.6.3 Upgrading an Existing SSL Environment

Configuration settings such as SSL settings are not carried over from the upgrade source. For information regarding configuring SSL, see Chapter 5, "SSL Configuration in Oracle Business Intelligence".

B.6.4 Upgrading an Existing SSO Environment

Configuration settings such as single sign-on (SSO) settings are not carried over from the upgrade source. For information regarding configuring SSO, see Chapter 4, "Enabling SSO Authentication".

C

Troubleshooting Security in Oracle Business Intelligence

This appendix describes common problems that you might encounter when configuring and using Oracle Business Intelligence security, and explains how to solve them. It contains the following sections

C.1 Resolving Inconsistencies With the Identity Store

A number of inconsistencies can develop between a repository, the web catalog, and an identity store. The following sections describe the usual ways this can occur and how to resolve the inconsistencies.

C.1.1 User is Deleted From the Identity Store

Behavior

If a user is deleted from the identity store then that user can no longer log in to Oracle Business Intelligence. However, references to the deleted user remain in the repository until an administrator removes them.

Cause

References to the deleted user still remain in the repository but that user cannot log in to Oracle Business Intelligence. This behavior ensures that if a user was deleted by accident and re-created in the identity store, then the user's access control rules do not need to be entered again.

Action

An administrator can run the Consistency Checker in the Oracle BI Administration Tool in online mode identify inconsistencies.

C.1.2 User is Renamed in the Identity Store

Behavior

A user is renamed in the identity store and then cannot log in to the repository with the new name.

Cause

This can occur if a reference to the user under the original name still exists in the repository.

Action

An administrator must either restart the Oracle BI Server or run the Consistency Checker in the Oracle BI Administration Tool to update the repository with a reference to the user under the new name. Once this has been resolved the Oracle BI Presentation Server updates the Presentation Catalog to refer to the new user name the next time this user logs in.

C.1.3 User Name is Reused in the Identity Store

Behavior

If a user name is added that is identical to one previously used in the identity stored, the new user with the same name cannot log in.

Cause

This can occur if references to the user name exist in the repository.

Action

An administrator must remove existing references to the user name contained in the repository by either running Consistency Checker in Oracle BI Administration Tool or by changing the existing user references to use the new user's GUID. When the new user logs in with the reused name, a new home directory is created for them in the Presentation Catalog.

C.2 Resolving Inconsistencies With the Policy Store

A number of inconsistencies can develop between the Presentation Catalog and the policy store. The following sections describe the usual ways this can occur and how to resolve the inconsistencies.

C.2.1 Application Role Was Deleted From the Policy Store

Behavior

After an Application Role is deleted from the policy store the role name continues to appear in the Oracle BI Administration Tool when working in offline mode. But the role name no longer appears in Presentation Services and users are no longer granted the permissions associated with the deleted role.

Cause

References to the deleted role name persist in the repository enabling the role name to appear in the Administration Tool when working in offline mode.

Action

An administrator runs the Consistency Checker in the Oracle BI Administration Tool in online mode to remove references in the repository to the deleted Application Role name.

C.2.2 Application Role is Renamed in the Policy Store

Behavior

After an Application Role is renamed in the policy store the new name does not appear in the Administration Tool in offline mode. But the new name immediately appears in lists in Presentation Services and Administration Tool. Users continue to see the permissions the role grants them

Cause

References to the original role name persist in the repository enabling the role name to appear in the Administration Tool when working in offline mode.

Action

An administrator either restarts the BI Server or runs the Consistency Checker in the Administration Tool to update the repository with the new role name.

C.2.3 Application Role Name is Reused in the Policy Store

Behavior

An Application Role is added to the policy store reusing a name used for a previous Application Role. Users are unable to access Oracle Business Intelligence resources according to the permissions granted by the original role and are not granted permissions afforded by the new role.

Cause

The name conflict must be resolved between the original role and new role with the same name.

Action

An administrator resolves the naming conflict by either deleting references to the original role from the repository or by updating the repository references to use the new GUID.

C.2.4 Application Role Reference is Added to a Repository in Offline Mode

Behavior

An Application Role has a blank GUID. This can occur after an Application Role reference is added to the repository in offline mode.

Cause

The Administration Tool in offline mode does not have access to the policy store and cannot fill in the GUID when a reference to the Application Role is added to the repository.

Action

After start up, the Oracle BI Server fills in any blank GUIDs for Application Role references with the actual GUID.

C.3 Resolving SSL Communication Problems

Behavior

Communication error. A process (the client) cannot communicate with another process (the server).

Action

When there is an SSL communication problem the client typically displays a communication error. The error can state only "client refused" with no further information. Check the server log file for the corresponding failure error message which typically provides more information about the issue.

Behavior

The following error message is displayed after the commit operation is performed using the BIDomain MBean (oracle.biee.admin:type=BIDomain, group=Service).

```
SEVERE: Element Type: DOMAIN, Element Id: null, Operation
Result: VALIDATION_FAILED, Detail Message: SSL must be enabled
on AdminServer before enabling on BI system; not set on server:
AdminServer
```

Action

This message indicates that SSL has not been enabled on the Oracle WebLogic Server Managed Servers, which is a prerequisite step. For more information, see Section 5.3, "Configuring the Web Server to Use HTTPS Protocol" and Section 5.4.3, "Commit the SSL Configuration Changes".

C.4 Resolving Issues with BISystemUser Credentials

Issue: Users are unable to log in with their valid user names and passwords. Error message: Invalid User name or Password.

Example C–1 Example bifoundation_domain.log Output When BISystemUser Credentials Become Out-of Sync

```
####<DATE> <Error> <oracle.wsm.resources.enforcement> <Machine_Name> <bi_server1>
<[ACTIVE] ExecuteThread: '2' for queue: 'weblogic.kernel.Default (self-tuning)'>
<<anonymous>> <> <> <1273244079442> <WSM-07607> <Failure in execution of assertion
{http://schemas.oracle.com/ws/2006/01/securitypolicy}wss-username-token executor
class
oracle.wsm.security.policy.scenario.executor.WssUsernameTokenScenarioExecutor.>
####<DATE> <Error> <oracle.wsm.resources.enforcement> <Machine_Name> <bi_server1>
<[ACTIVE] ExecuteThread: '2' for queue: 'weblogic.kernel.Default (self-tuning)'>
<<anonymous>> <> <> <1273244079442> <WSM-07602> <Failure in WS-Policy Execution
due to exception.>
####<07-might-2010 15:54:39 o'clock BST> <Error>
<oracle.wsm.resources.enforcement> <ukp79330> <bi_server1> <[ACTIVE]
ExecuteThread: '2' for queue: 'weblogic.kernel.Default (self-tuning)'>
<<anonymous>> <> <> <1273244079442> <WSM-07501> <Failure in Oracle WSM Agent
processRequest, category=security, function=agent.function.service,
application=bimiddleware#11.1.1.2.0, composite=null, modelObj=SecurityService,
```

```
policy=oracle/wss_username_token_service_policy, policyVersion=null,
assertionName={http://schemas.oracle.com/ws/2006/01/securitypolicy}wss-username-to
ken.>
####<DATE> <Error> <oracle.wsm.agent.handler.wls.WSMAgentHook> <Machine_Name> <bi_
server1> <[ACTIVE] ExecuteThread: '2' for queue: 'weblogic.kernel.Default
(self-tuning)'> <<anonymous>> <> <> <1273244079442> <BEA-000000> <WSMAgentHook: An
Exception is thrown: FailedAuthentication : The security token cannot be
authenticated.>
####<DATE> <Error> <oracle.wsm.resources.security> <Machine_Name> <bi_server1>
<[ACTIVE] ExecuteThread: '5' for queue: 'weblogic.kernel.Default (self-tuning)'>
<<anonymous>> <> <> <1273244091113> <WSM-00008> <Web service authentication
failed.>
####<DATE> <Error> <oracle.wsm.resources.security> <Machine_Name> <bi_server1>
<[ACTIVE] ExecuteThread: '5' for queue: 'weblogic.kernel.Default (self-tuning)'>
<<anonymous>> <> <> <1273244091113> <WSM-00006> <Error in receiving the request:
oracle.wsm.security.SecurityException: WSM-00008 : Web service authentication
failed
```

Index

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

R

S

T

U

V

W