Oracle® Fusion Middleware

Security Guide for Oracle Business Intelligence Enterprise Edition

11*g* Release 1 (11.1.1)

E10543-06

April 2012

Explains how to configure Oracle Business Intelligence Enterprise Edition security, including settings for SSO, SSL, external authentication, and Presentation Services privileges.

ORACLE

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

E10543-06

Primary Author: Nick Fry

Contributors: Trish Fuzesy, Oracle Business Intelligence development, product management, and quality assurance teams.

Contents

3 Using Alternative Authentication Providers

A Alternative Security Administration Options

B Understanding the Default Security Configuration

C Troubleshooting Security in Oracle Business Intelligence

D Managing Security for Dashboards and Analyses

Index

Preface

The Oracle Business Intelligence Foundation Suite is a complete, open, and integrated solution for all enterprise business intelligence needs, including reporting, ad hoc queries, OLAP, dashboards, scorecards, and what-if analysis. The Oracle Business Intelligence Foundation Suite includes Oracle Business Intelligence Enterprise Edition.

Oracle Business Intelligence Enterprise Edition (Oracle BI EE) 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, and an enterprise reporting engine.

The components of Oracle BI EE 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 BI EE provides scalability and performance with data-source specific optimized request generation, optimized data access, advanced calculation, intelligent caching services, and clustering.

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

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

Access to Oracle Support

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

Related Documents and Other Resources

See the Oracle Business Intelligence documentation library for a list of related Oracle Business Intelligence documents.

See also the following related document:

- *Oracle Fusion Middleware Application Security Guide*

In addition, go to the Oracle Learning Library for Oracle Business Intelligence-related online training resources.

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 BI EE from a previous release, read the following information carefully, because there might be significant differences in features, tools, and procedures. For more information about upgrading to Oracle BI EE 11*g*, see *Oracle Fusion Middleware Upgrade Guide for Oracle Business Intelligence Enterprise Edition*.

This preface contains the following topics:

New Features for Oracle BI EE 11*g* Release 1 (11.1.1.6)

This section describes new features for Oracle BI EE 11*g* Release 1 (11.1.1.6). It contains the following topics:

New Features for 11.1.1.6.2

New security features in Oracle BI EE 11*g* Release 1 (11.1.1.6.2) include:

Oracle BI Security Diagnostics Helper

A new Oracle BI Security Diagnostics Helper application has been added to help you diagnose possible configuration issues which may prevent your users from being able to log in to your Oracle BI system.

For more information, see "Using the Oracle BI Security Diagnostics Helper to Automatically Identify Security Issues".

New Features for 11.1.1.6.0

New security features in Oracle BI EE 11*g* Release 1 (11.1.1.6.0) include:

New Privileges

Several new privileges were added to the Oracle BI EE Administration page:

- **New Catalog and Home and Header Privileges** — These privileges determine which users can search the catalog, what functionality displays in the Oracle BI EE global header, who can access the Home Page and Catalog page, and who can see custom links in the global header and Getting Started area of the Home Page.
- **Access to BI Composer** — This privilege allows users to access the basic features of BI Composer.

For more information about these privileges, see "Managing Presentation Services Privileges".

New Features for Oracle BI EE 11*g* Release 1 (11.1.1.5)

There are no new security features in Oracle BI EE 11*g* Release 1 (11.1.1.5).

New Features for Oracle BI EE 11*g* Release 1 (11.1.1.3)

New security features in Oracle BI EE 11*g* Release 1 (11.1.1.3) include:

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

Integration 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 Oracle BI repository 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 its 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 his or her 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 Oracle BI 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 road map for configuring security in Oracle Business Intelligence.

> **Note:** For a high-level road map 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 to get an overview of security concepts, tools, and terminology.
2. Learn about the default set of users, groups, and application roles by reading the summary in Section 2.1, "Working with the Default Users, Groups, and Application Roles".
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. Assign each group to an appropriate application role.
7. Fine-tune the permissions that users and groups have in the Oracle BI repository.
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 1.7, "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.

Several Oracle Business Intelligence legacy authentication options are still supported for backward compatibility. The best practice is to perform authentication and authorization using an identity store and authentication provider through the default security model described in this chapter. However, there are certain scenarios where this is not possible or where certain aspects of the legacy approach to authentication and authorization are required. Typically the use of these alternative methods requires that your user population and groups are not held in the identity store referenced by the authentication provider configured in the Oracle WebLogic domain. Consequently, when using alternative authentication methods, several sections of this chapter are not relevant. Instead, refer to Appendix A, "Alternative Security Administration Options". Note that application roles described in this chapter are still used with alternative authentication and authorization mechanisms. Also note that the authentication

provider configured in the Oracle WebLogic domain is always used by the BI System User, even when using alternative methods for other users.

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 that is stored in the LDAP server that is embedded in the Oracle WebLogic Server domain for Oracle Business Intelligence. You can use the Oracle WebLogic Server Administration Console to create and manage users and groups in the embedded LDAP server.

You might choose to configure an authentication provider for an alternative directory. You can use the Oracle WebLogic Server Administration Console to view the users and groups in the directory. However, you must continue to use the appropriate tools to make any modifications to the directory. For example, if you reconfigure Oracle Business Intelligence to use Oracle Internet Directory (OID), you can view users and groups in Oracle WebLogic Server Administration Console but you must manage them using the OID Console. Oracle Internet Directory (OID) is an LDAP directory service, and is the preferred identity store for authentication purposes.

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*.

You can continue to use the external table authentication method, instead of the default authentication provided with this release. For more information, see Appendix A.1.2, "Setting Up External Table Authentication".

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 application 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 assigned to directory server groups and users. For example, the default application roles BIAdministrator, BIConsumer, and BIAuthor.

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 assigns those roles to preexisting 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 assign the application roles to the users and groups available in the target environment.

Figure 1-1 shows an example using the default set of users, groups, and application roles.

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

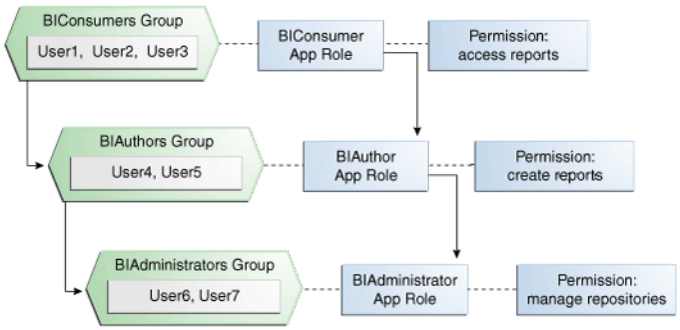


Figure 1-1 shows the following:

- 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 BIAuthor, 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

The security policy definition is split across the following components:

- Oracle BI Presentation Services – This defines which catalog objects and Oracle BI Presentation Services functionality can be accessed by which users with specific application roles. Access to functionality is defined in the Managing Privileges page in terms of Oracle BI Presentation Services privileges and access to Oracle BI Presentation Catalog objects is defined in the Permission dialog.
- Repository – This defines which metadata items within the repository can be accessed by which application roles and users. The Oracle BI Administration Tool is used to define this security policy.
- Policy Store – This defines which BI Server, Oracle BI Publisher, and Oracle 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 Application Security Guide*.

To find out about using these components, see Section 1.6, "Using Tools to Configure Security in Oracle Business Intelligence".

1.5 About Preconfigured Users, Groups, and Application Roles

When you install Oracle Business Intelligence, there are a number of preconfigured users, groups, and application roles that you can use to deploy Oracle Business Intelligence. For more information, see Section 2.1, "Working with the Default Users, Groups, and Application Roles".

1.6 Using Tools to Configure Security in Oracle Business Intelligence

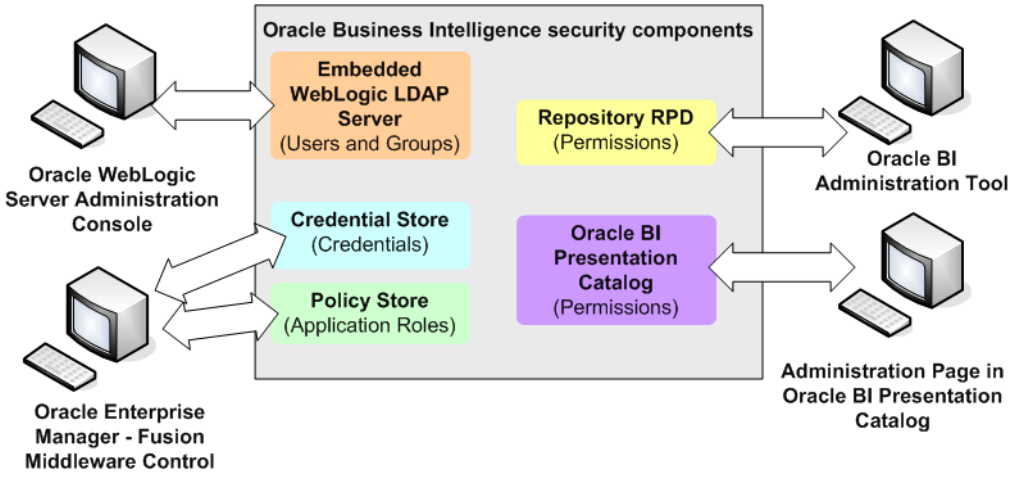
To configure security in Oracle Business Intelligence, you use the following tools:

- "Using Oracle WebLogic Server Administration Console"
- "Using Oracle Fusion Middleware Control"
- "Using Oracle BI Administration Tool"
- "Using Presentation Services Administration"

> **Note:** To see an example of using the Oracle Business Intelligence tools to configure the installed users, groups, and application roles, see Section 2.2, "An Example Security Setup Using the Default Groups and Application Roles".

Figure 1–2 summarizes the tools that you use to configure security in a default installation of Oracle Business Intelligence that uses the embedded WebLogic LDAP Server.

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



For more information about using managing the default security, see Chapter 2, "Managing Security Using the Default Security Configuration").

1.6.1 Using Oracle WebLogic Server Administration Console

You use Oracle WebLogic Server Administration Console to manage the default WebLogic LDAP Server that is used to authenticate users and groups.

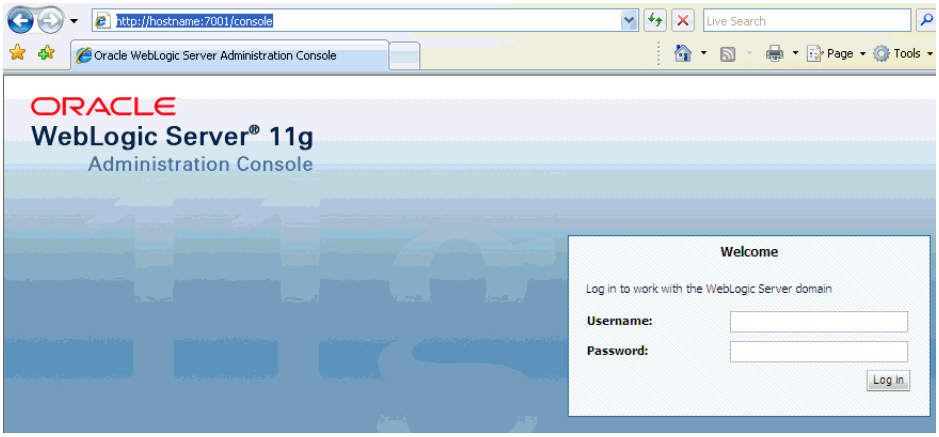
Oracle WebLogic Server is automatically installed and serves as the default administration server. The Oracle WebLogic Server Administration Console is browser-based and is used, among other things, to manage the embedded directory

server that is configured as the default authenticator. You launch the Oracle WebLogic Server Administration Console 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 same as used for the Administration Server; 7001 is the default port. For more information about using the Oracle WebLogic Server Administration Console, see *Oracle Fusion Middleware Oracle WebLogic Server Administration Console Online Help*.

To log in to the Oracle WebLogic Server Administration Console:

1. Display the Oracle WebLogic Server login page by entering its URL into a Web browser.

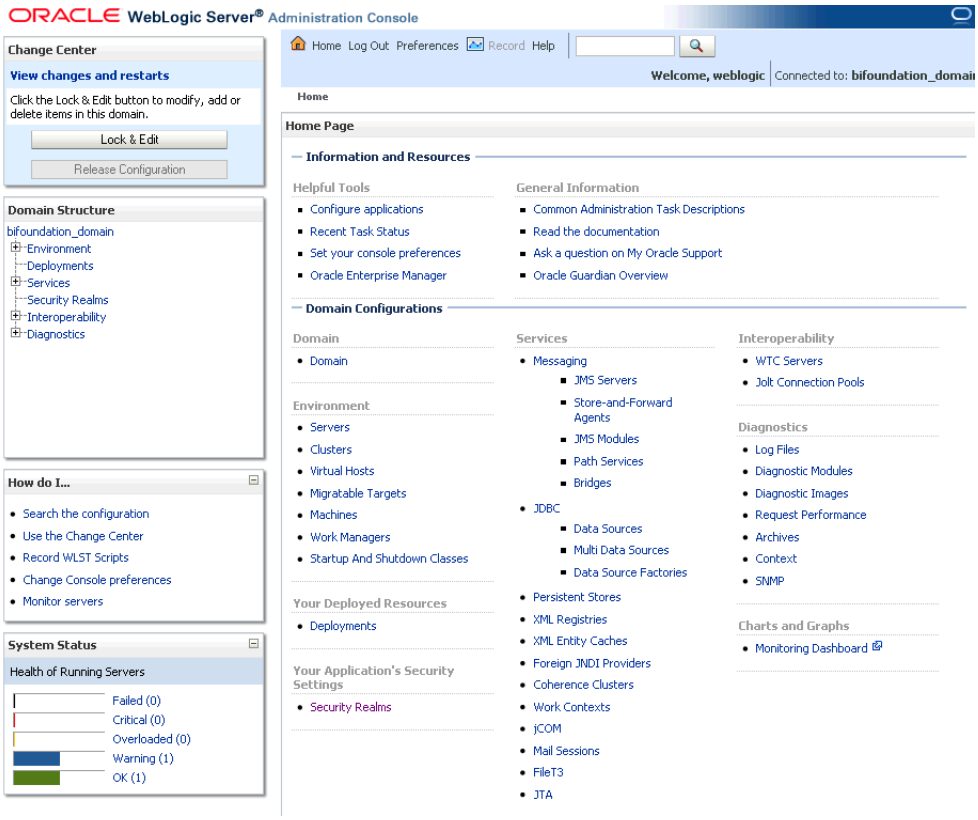
   For example, http://*hostname:7001*/console.



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

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

   The Administration Console displays.



3. Use the tabs and options in the Domain Structure as required to configure users, groups, and other options.

> **Note:** If you use an alternative authentication provider, such as Oracle Internet Directory instead of the default the WebLogic LDAP Server, then you must use the alternative authentication provider administration application (for example an administration console) to manage users and groups.

1.6.2 Using Oracle Fusion Middleware Control

Fusion Middleware Control is a Web browser-based graphical user interface that enables you to administer a collection of components called a farm. A farm contains 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. The domain is named **bifoundation_domain** (in Simple or Enterprise installations), and is found under the **WebLogic Domain** folder in the Fusion Middleware Control navigation pane.

You use Oracle Fusion Middleware Control to manage Oracle Business Intelligence security as follows:

- Manage the application roles and application policies that control access to Oracle Business Intelligence resources
- Manage Single Sign On (SSO), and Secure Sockets Layer (SSL)
- Configure multiple authentication providers for Oracle Business Intelligence

To log in to Fusion Middleware Control, open a Web browser and enter the Fusion Middleware Control URL, in the following format:

`http://hostname.domain:port/em`

The port number is the number of the Administration Server, and the default port number is 7001.

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

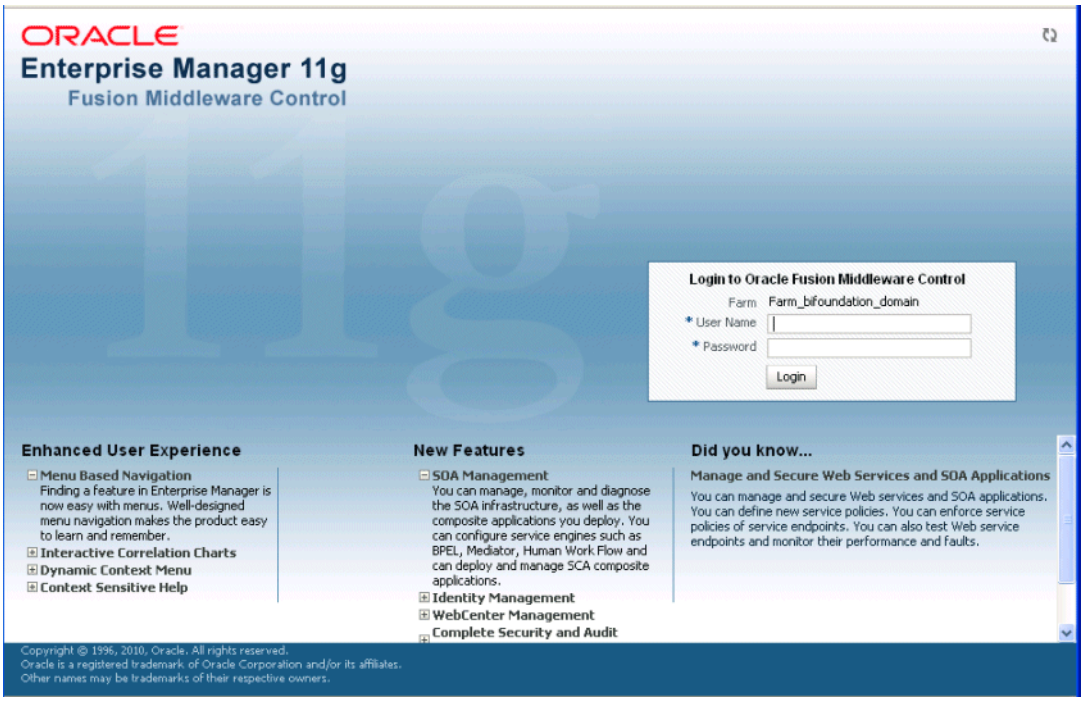
To use Fusion Middleware Control:

1. Enter the URL in a Web browser. For example:

   `http://host1.example.com:7001/em`

   The Fusion Middleware Control login page is displayed, as shown in Figure 1–3.

Figure 1–3 Login Page for Fusion Middleware Control



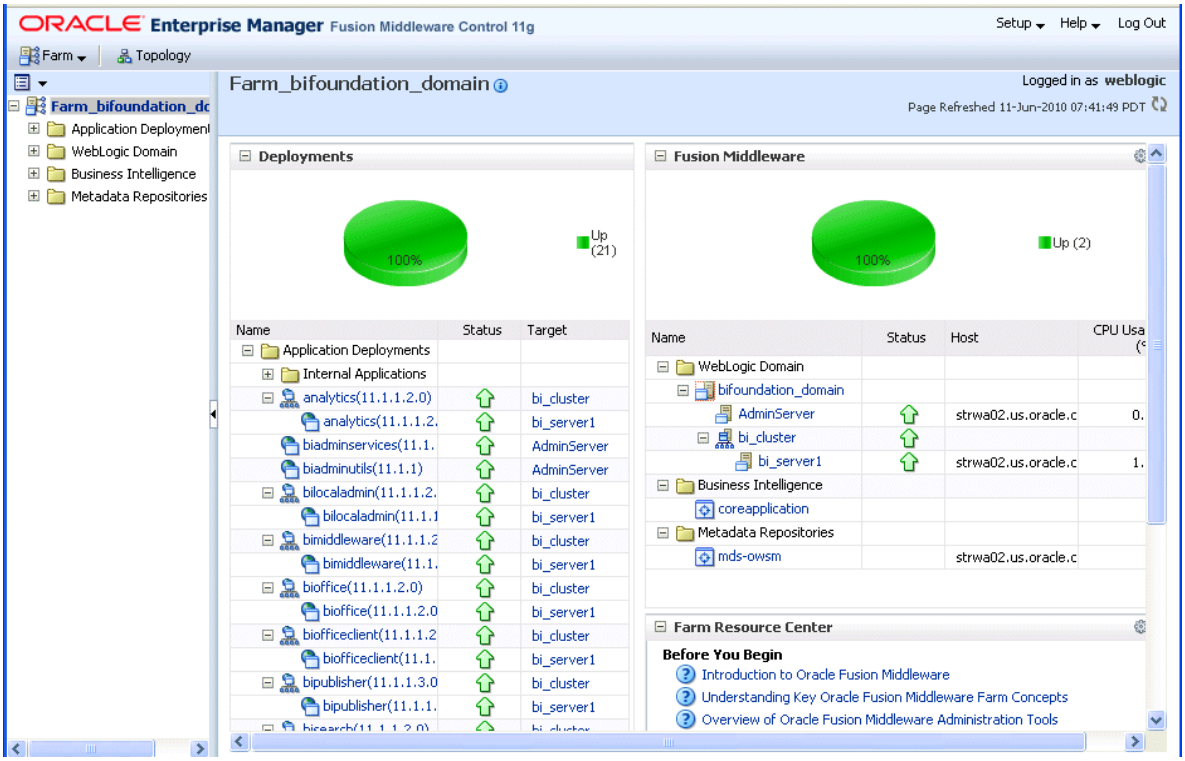
2. Enter the system administrator user name and password and click **Login**.

   This system-wide administration user name and password was specified during the installation process, and you can use it to log in to Oracle WebLogic Server Administration Console, Fusion Middleware Control, and Oracle Business Intelligence.

   Alternatively, enter any other user name and password that has been granted the Oracle BI Administrator application role.

   Fusion Middleware Control opens, as shown in Figure 1–4.

Figure 1–4 Main Page in Fusion Middleware Control



3. From the farm main page, expand the **Business Intelligence** folder.
4. Select **coreapplication** to display pages specific to Oracle Business Intelligence.
5. Manage Oracle Business Intelligence security using Fusion Middleware Control as follows:
   - Manage application roles and application policies

     For more information, see Section 2.4, "Managing Application Roles and Application Policies Using Fusion Middleware Control".
   - Configure Single Sign On (SSO)

     For more information, see Section 4.6, "Enabling SSO Authentication Using Fusion Middleware Control".
   - Configure Secure Sockets Level (SSL)

     For more information, see:
     - Section 5.3.1, "Configuring SSL Communication Between Components Using Fusion Middleware Control and Oracle WebLogic Server Administration Console"
     - Section 5.3.2, "Configuring SSL for the SMTP Server Using Fusion Middleware Control"

1.6.3 Using Oracle BI Administration Tool

You use the Oracle BI Administration Tool to configure permissions for users and application roles against objects in the metadata repository.

To use the Administration Tool:

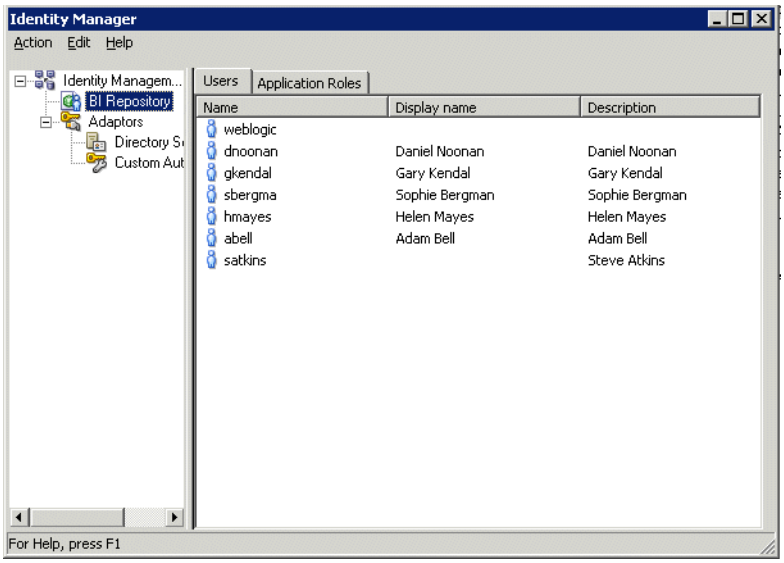
1. Log in to the Administration Tool.

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

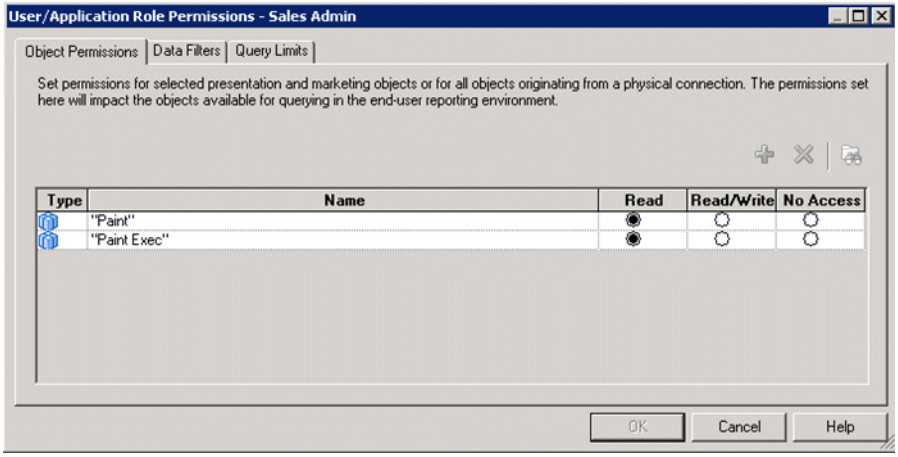
2. (Optional) Select **Manage**, then **Identity** to display the Identity Manager dialog.

   Figure 1–5 shows the Identity dialog and the installed application roles BIAdministrator, BIAuthor, and BIConsumer.

Figure 1–5 Identity Manager Dialog Showing Installed Application Roles



   If you double-click an application role to display the Application Role <*Name*> dialog, then click Permissions, you can use the Object Permissions tab to view or configure (in the repository) the Read and Write permissions for that application role, in relation to objects and folders in the Oracle BI Presentation Catalog.

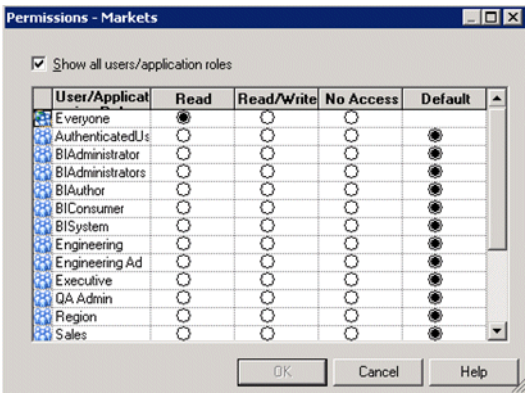


3. Close Identity Manager.

4. In the Presentation pane, expand a folder, then right-click an object to display the Presentation Table <*Table name*> dialog.
5. Click **Permissions** to display the Permissions <*Table name*> dialog.

   Figure 1–6 shows users and installed application roles BIAdministrator, BIAuthor, and BIConsumer, and the radio buttons Read, Read/Write, No Access, and Default that you use to set the permissions for the application roles.

Figure 1–6 Permissions Dialog Showing Installed Application Roles



1.6.4 Using Presentation Services Administration

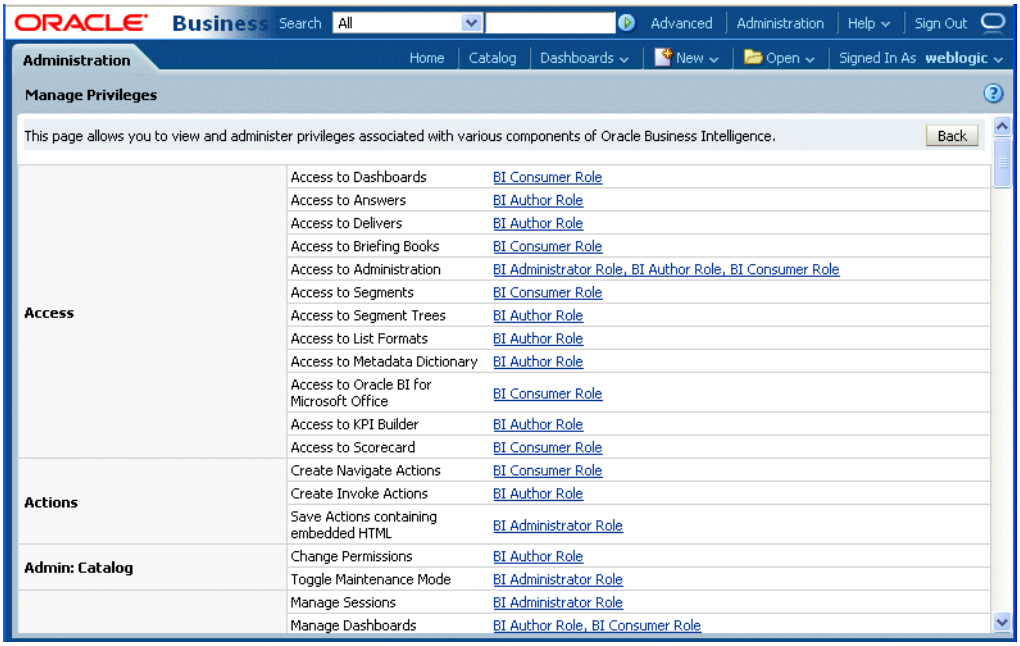
You use the Presentation Services Administration page to configure user privileges.

To use the Presentation Services Administration page:

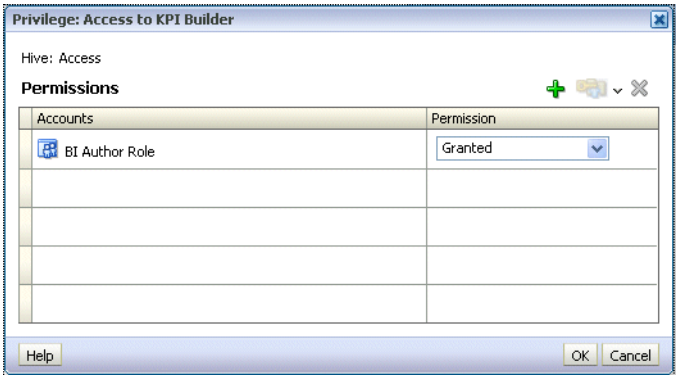
1. Log in to Oracle Business Intelligence with Administrator privileges.
2. Select the **Administration** link to display the Administration page.
3. Select the **Manage Privileges** link.

   Figure 1–7 shows application roles listed against the privileges to which they are assigned.

Figure 1–7 Manage Privileges Page in Presentation Services Administration Showing Application Roles

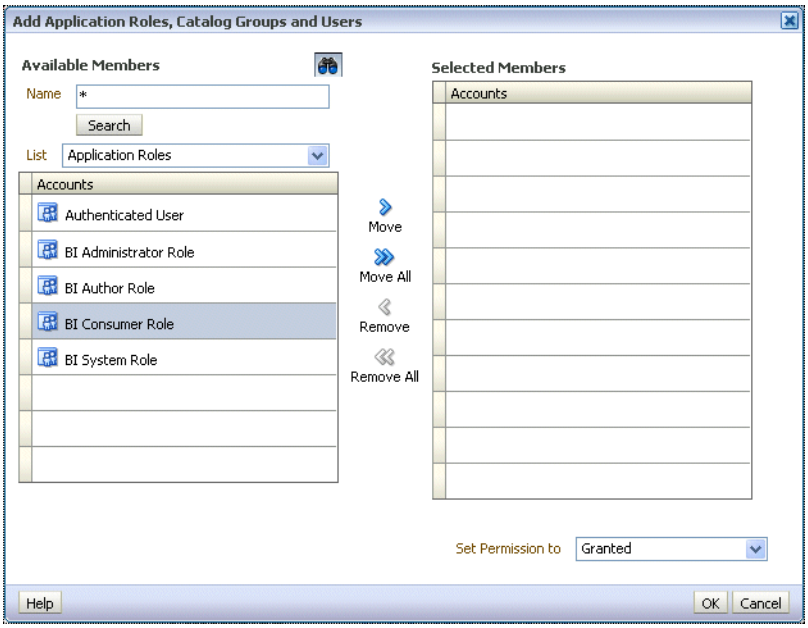


4. Select a link (for example BIAuthor) for a particular privilege (for example, Access to KPI Builder), to display the Privilege <*Privilege name*> dialog.



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

   Use this dialog to assign application roles (for example, BIAdministrator, BIAuthor, and BIConsumer) to this privilege.



1.7 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 default preconfigured users, groups, and application roles. 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 following order:

1. Read this chapter 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, "Using Tools to Configure Security in Oracle Business Intelligence"
2. Learn about the default set of users, groups, and application roles by reading the summary in Section 2.1, "Working with the Default Users, Groups, and Application Roles".
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 4.
   - If you want to reconfigure Oracle Business Intelligence to use an alternative authentication provider such as Oracle Internet Directory (OID), then follow the tasks listed in Step 5.

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 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:

   a. Set up the users that you want to deploy as described in Section 2.3.2, "Creating a New User in the Embedded WebLogic LDAP Server".

      For example, if you want to deploy Oracle Business Intelligence to 20 people who need to view analyses, you might create 20 users.

   b. If you want to assign users to default groups, (for example, BIAuthors, or BIAdministrators), then follow the steps in Section 2.3.1.1, "Assigning a User to a Default Group".

      For example, you might assign 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 2.3.3, "Creating 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 2.3.4, "Assigning a User to a Group in the Embedded WebLogic LDAP Server".

      For example, you might assign users to the preconfigured group named BIAuthors, or you might assign users to a new group that you have created.

      Tip: The simplest way to set up security is to create users and assign them to the default groups (for example, BIAuthors, and BIAdministrators). For detailed steps, see Section 2.3.1.1, "Assigning 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 assign your users to the new groups. For detailed steps, see Section 2.3.1.2, "Assigning a User to a New Group and a New Application Role".

Note: If you have upgraded the system from 10*g* and you want to authenticate users with the authentication provider that is configured for the Oracle WebLogic domain, then review the initialization blocks that set the User system session variable in the repository. Setting this variable is an alternative mechanism that authenticates users who attempt to access the BI Server if they fail authentication using the provider that is configured for the Oracle WebLogic domain. For more information on working with repositories, see Appendix A.1.1.2, "Defining a USER Session Variable for LDAP Authentication" and *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

---

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, "High-Level Steps for Configuring an Alternative Authentication Provider".

   b. (Optional) Configure OID as the Credential Store and Policy Store Provider as described in Section 3.9, "Configuring Oracle Internet Directory as the Policy Store and the Credential 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 2.4.2, "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 2.4.3, "Creating Application Policies Using Fusion Middleware Control".

   For example, you might use the default application policies that are used by the default application roles named BIConsumer, BIAuthor, and BIAdministrator, or you might create your own application policies.

8. Assign 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 assigned to an appropriate application role (that is, BIConsumer, BIAuthor, or BIAdministrator). No additional steps are required to assign the default groups to application roles.

     If you have created new groups, you must assign the new groups to appropriate application roles as described in Section 2.4.2.3, "Assigning a Group to an Application Role".

   - If you are using a commercial Authenticator Provider such as Oracle Internet Directory, then you must assign the groups to appropriate application roles as described in Section 2.4.2.3, "Assigning 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, use the Administration Tool to update the permissions as described in Section 2.5, "Managing Metadata Repository Privileges Using the Oracle BI Administration Tool".

For example, you might want to enable an application role called BISuperConsumer to create analyses, so you use the Administration Tool to change the Read access to a subject area to Read/Write access.

Note: If you are using the default SampleAppLite.rpd file in a production system, you should change the password from its installed value, using the Administration Tool. For more information about the SampleAppLite repository file, see *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

---

10. If you want to fine-tune the permissions that users and groups have in the Presentation Services Administration page to change the permissions as described in Section 2.6, "Managing Presentation Services Privileges Using Application Roles".

    For example, you might want to prevent an application role called BISuperConsumer from viewing scorecards, so you use Presentation Services Administration Page 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.

1.8 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 the 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 Oracle BI Presentation 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 assigned 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 11*g* 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 11*g* the entire repository is encrypted using a key derived from a user supplied password.

  Note: 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*:

- BI Server Initialization Blocks – The BI Server in Release 11*g* continues to support the use of initialization blocks for authentication and authorization. In Release 10*g* the 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.

  For more information, see "Working With Initialization Blocks" in *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*
- Catalog Groups – Oracle Business Intelligence Release 11*g* continues to support the definition of Catalog groups within the Oracle BI 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.

  For more information, see Section D.2.2, "Working with Catalog Groups".
- SA System Subject Area – Oracle Business Intelligence Release 11*g* supports the use of SA System Subject Area, in combination with the BI Server initialization blocks, to access user, group and profile information stored in database tables.

  For more information, see "Setting Up the SA System Subject Area" in *Oracle Fusion Middleware Scheduling Jobs Guide for Oracle Business Intelligence Enterprise Edition*.

For more information on differences between Releases 10*g* and 11*g*, see *Oracle Fusion Middleware Upgrade Guide for Oracle Business Intelligence Enterprise Edition*.

1.9 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 log in.

Authentication Provider
A security provider used to access user and group information and 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.

Catalog Permissions
These rights grant access to objects that are stored in the Oracle BI Presentation Catalog. The rights are stored in the catalog and managed by Presentation Services.

Catalog Privileges
These rights grant access to features of the Oracle BI Presentation Catalog. The rights are stored in the catalog and managed by Presentation Services. These privileges are either granted or denied.

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 plain text 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 re configured 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 assigning 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 assigned (users, groups, and application 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.

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:

- Oracle BI Presentation Catalog:

  Defines which Oracle BI Presentation 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 the 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 Oracle 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 1.7, "Detailed List of Steps for Setting Up Security In Oracle Business Intelligence".

By deploying the default embedded WebLogic LDAP Server, you can use the preconfigured users, groups, and application roles. You can also develop your own users, groups, and application roles.

This chapter contains the following sections:

You can migrate users (with their encrypted passwords), and groups from the default embedded WebLogic LDAP server into an alternative authentication provider (for example, OID, external tables, or another LDAP directory). For more information, see *Oracle Fusion Middleware Securing Oracle WebLogic Server*.

2.1 Working with the Default Users, Groups, and Application Roles

When you install Oracle Business Intelligence, there are a number of preconfigured users, groups, and application roles that you can use to deploy Oracle Business

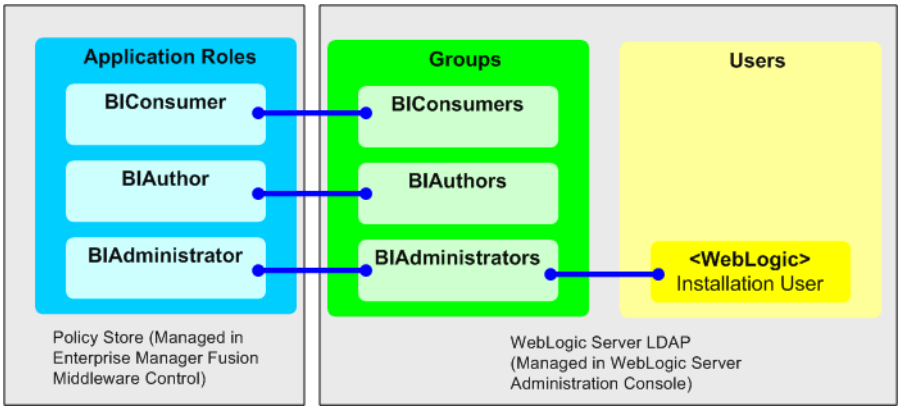
Intelligence. For example, there is a user that is assigned to a BIAdministrators group (with a name that is user-specified at installation time, for example WebLogic), a group named BIAdministrators, and an associated application role named BIAdministrator. The default installed users, groups, and application 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, 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, and Policy Store. For example, the application role named BIAuthor is preconfigured with permissions and privileges that are required to create dashboards, reports, actions, and so on.

Figure 2–1 shows application roles, groups and users that are preconfigured during installation.

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



The following groups are available:

- BIConsumers (preconfigured with the BIConsumer application role).
- BIAuthors (preconfigured with the BIAuthor application role).
- BIAdministrators (preconfigured with the BIAdministrator application role).

The user that is specified at installation time (for example, Weblogic), is automatically assigned to the WebLogic Administrators group named BIAdministrators and to the associated application role named BIAdministrator. The 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 BIAdministrators group automatically inherits privileges from the BIAuthors and BIConsumers groups. 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 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."

2.2 An Example Security Setup Using the Default 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. 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.

Figure 2–2 shows the users, groups, and application roles that you would deploy to implement this security model.

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

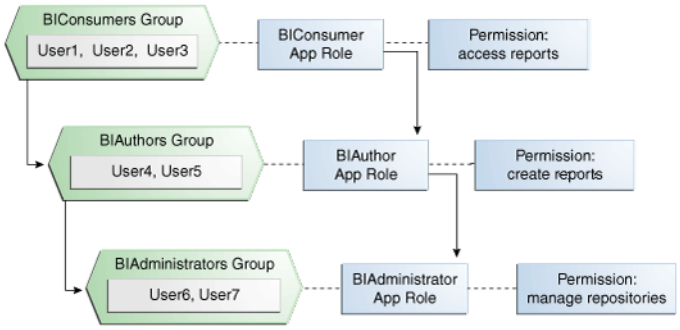


Figure 2–2 shows the following:

- The group named BIConsumers contains User1, User2, and User3. Users in the group BIConsumers are assigned to 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 to the application role named BIAuthor, which enables the users to create reports.

- The group named BIAdministrators contains User6 and User7. Users in the group BIAdministrators are assigned to the application role named BIAdministrator, which enables the users to manage repositories.

To implement this example security model:

1. Create seven users named User1 to User 7, as described in Section 2.3.2, "Creating a New 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 Section 2.3.4, "Assigning a User to a Group in the Embedded WebLogic LDAP Server".

2.3 Managing Users and Groups in the Embedded WebLogic LDAP Server

This section explains how to manage users and groups in the Embedded WebLogic LDAP Server, and contains the following topics:

- Section 2.3.1, "Setting Up Users, Groups, and Application Roles"
- Section 2.3.2, "Creating a New User in the Embedded WebLogic LDAP Server"
- Section 2.3.3, "Creating a Group in the Embedded WebLogic LDAP Server"
- Section 2.3.4, "Assigning a User to a Group in the Embedded WebLogic LDAP Server"
- Section 2.3.5, "(Optional) Changing a User Password in the Embedded WebLogic LDAP Server"

2.3.1 Setting Up Users, Groups, and Application Roles

This section summarizes recommended approaches for setting 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 BIAuthors group is preconfigured with the privileges it requires to access the other Oracle BI components, such as the Oracle BI repository and Oracle BI Presentation Catalog.

  For detailed steps, see Section 2.3.1.1, "Assigning 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 assigned to a new application role named BIMarketingRole.

For detailed steps, see Section 2.3.1.2, "Assigning a User to a New Group and a New Application Role".

2.3.1.1 Assigning a User to a Default Group

To create a new user and assign that user to a default group:

1. Launch WebLogic Administration Console as described in Section 1.6.1, "Using Oracle WebLogic Server Administration Console".
2. Create a new user as described in Section 2.3.2, "Creating a New 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 2.3.4, "Assigning a User to a Group in the Embedded WebLogic LDAP Server".

2.3.1.2 Assigning a User to a New Group and a New Application Role

To create a new user and assign the user to a new group and a new application role:

1. Launch WebLogic Administration Console as described in Section 1.6.1, "Using Oracle WebLogic Server Administration Console".
2. Create a new user as described in Section 2.3.2, "Creating a New User in the Embedded WebLogic LDAP Server".
3. Create a new group as described in Section 2.3.3, "Creating a Group in the Embedded WebLogic LDAP Server".
4. Assign the new user to the new group as described in Section 2.3.4, "Assigning a User to a Group in the Embedded WebLogic LDAP Server".
5. Create a new application role and assign it to the new group as described in Section 2.4.2.2, "Creating an Application Role".

   If you simply want to assign a group to an application role, follow the steps in Section 2.4.2.3, "Assigning a Group to an Application Role".
6. Edit the Oracle BI repository and set up the privileges for the new application role as described in Section 2.5.2, "Setting Repository Privileges for an Application Role".
7. Edit the Oracle BI Presentation Catalog and set up the privileges for the new user and group as described in Section 2.6.3, "Setting Presentation Services Privileges for Application Roles".

2.3.2 Creating a New 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, 3 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 2.2, "An Example Security Setup Using the Default Groups and Application Roles".

Repeat this task for each user that you want to deploy.

A new user who logs in without being assigned to any groups, is given a basic level of operational permissions conferred by the BIConsumer application role, through association with the Authenticated User application role. For more information, see Appendix B.4, "Default Security Configuration".

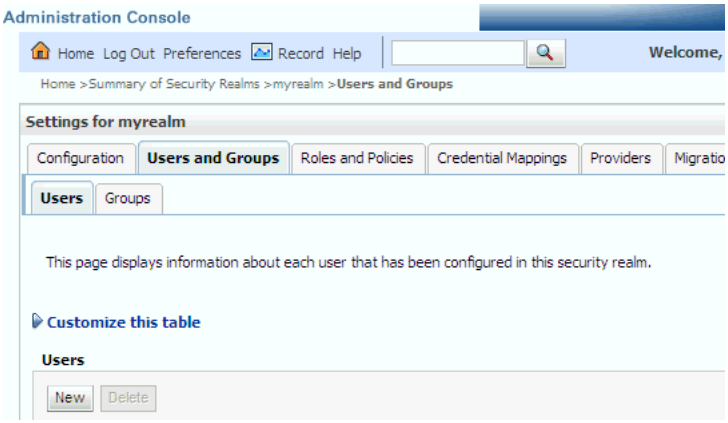
To create a new user in the embedded WebLogic LDAP server:

1. Log in to the Oracle WebLogic Server Administration Console.

   For more information, see Section 1.6.1, "Using Oracle WebLogic Server Administration Console".

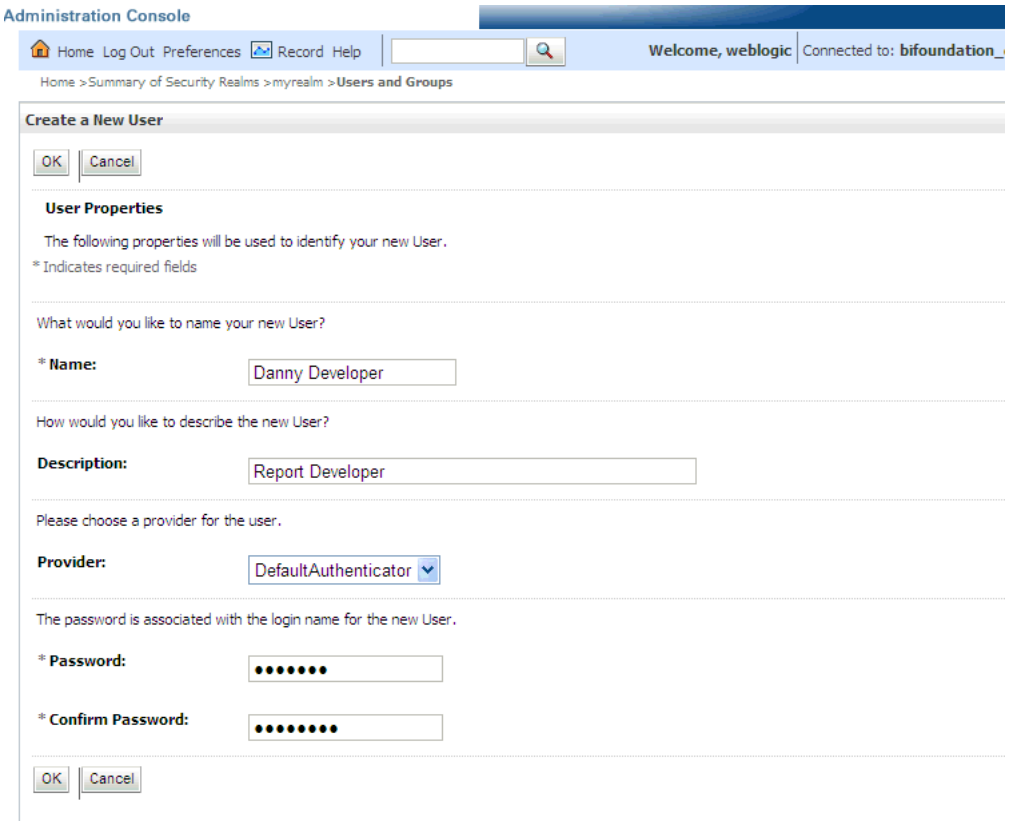
2. In Oracle WebLogic Server Administration Console, select **Security Realms** from the left pane and click the realm you are configuring. For example, **myrealm**.

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 the 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.

2.3.3 Creating 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 2.2, "An Example Security Setup Using the Default Groups and Application Roles".

Repeat this task for each group that you want to deploy

To create a group in the embedded WebLogic LDAP server:

1. Launch Oracle WebLogic Server Administration Console.

   For more information, see Section 1.6.1, "Using 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 the 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.

2.3.4 Assigning a User to a Group in the Embedded WebLogic LDAP Server

You typically assign 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 2.2, "An Example Security Setup Using the Default Groups and Application Roles".

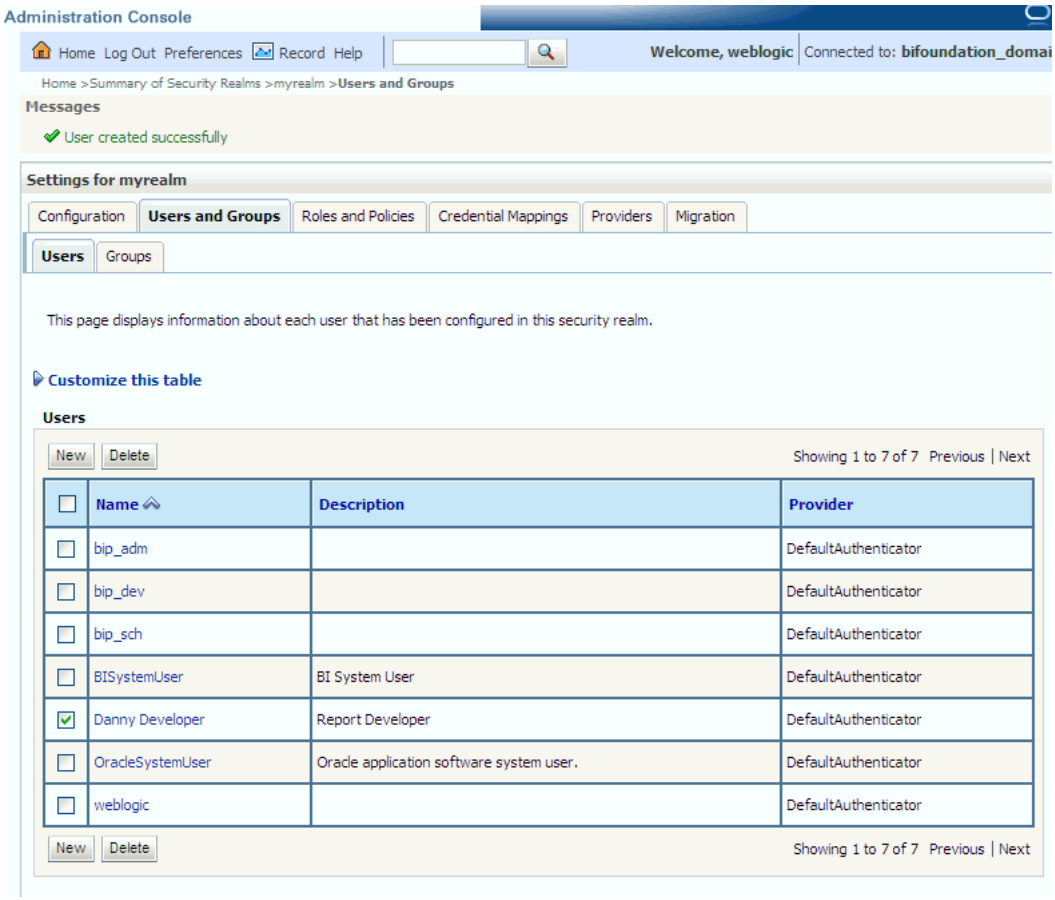
Repeat this task to assign each user to an appropriate group.

To add a user to a group in the embedded WebLogic LDAP server:

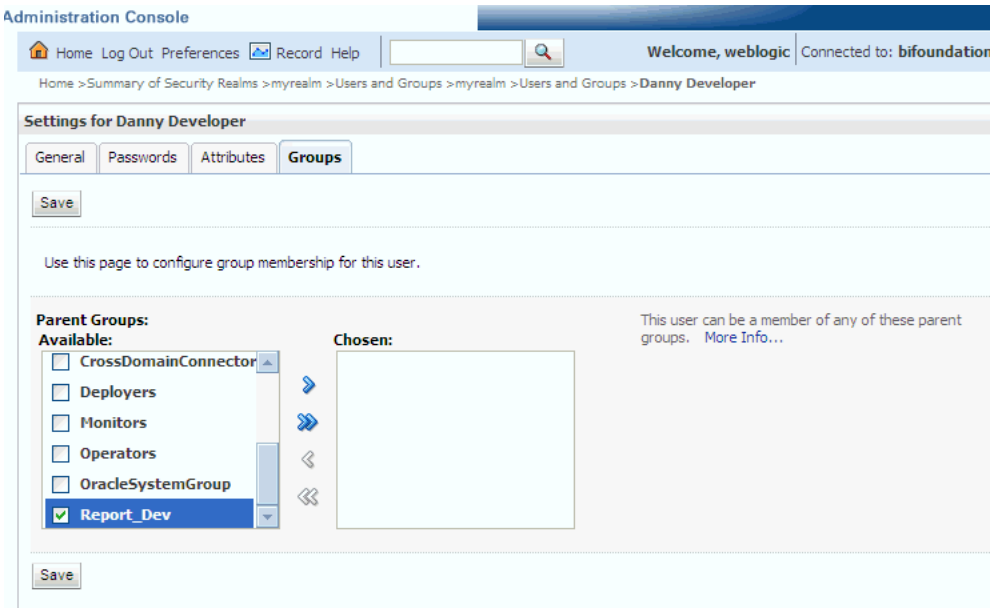
1. Launch Oracle WebLogic Server Administration Console.

   For more information, see Section 1.6.1, "Using 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.



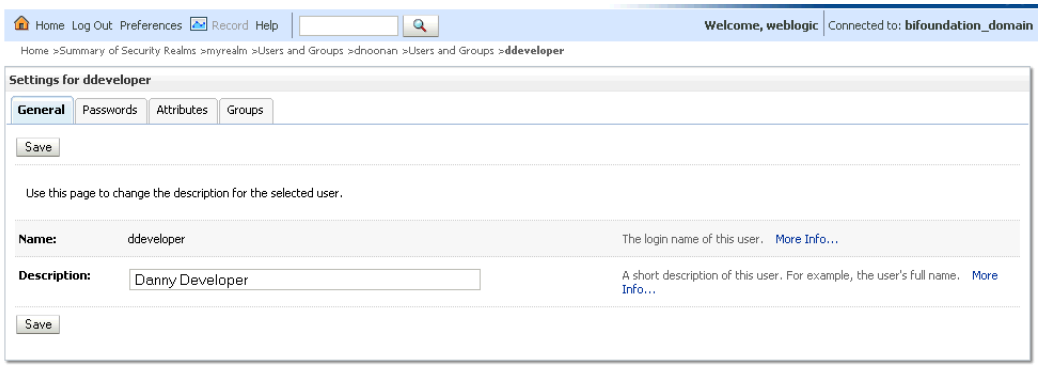
7. Click **Save**.

2.3.5 (Optional) Changing 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 embedded WebLogic LDAP 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**.

Note: If you change the password of the system user, you also need to change it in the credential store.

2.4 Managing Application Roles and Application Policies Using Fusion Middleware Control

In Oracle Business Intelligence, you use Fusion Middleware Control to manage 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 2.4.1, "Displaying Application Policies and Application Roles Using Fusion Middleware Control"
- Section 2.4.2, "Creating Application Roles Using Fusion Middleware Control"
- Section 2.4.3, "Creating Application Policies Using Fusion Middleware Control"
- Section 2.4.4, "Modifying Application Roles Using 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 assigned to an appropriate application role (that is, BIConsumer, BIAuthor, or BIAdministrator). No additional steps are required to assign the default groups to application roles.

The simplest way to set up security is to assign your groups to the default application roles, (that is, BIConsumer, BIAuthor, and BIAdministrator). Each default group is preconfigured to use the appropriate default application role. For example, the default group named BIAuthors is assigned 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 Oracle BI 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 2.4.2, "Creating Application Roles Using Fusion Middleware Control".

Note: You can create application roles based on default Application policies, or you can create your own Application policies. For more information about the default users, groups, and application roles, see Section 2.1, "Working with the Default Users, Groups, and Application Roles".

- If required, create new Application policies. For more information, see Section 2.4.3, "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 2.4.4, "Modifying Application Roles Using Fusion Middleware Control".

2.4.1 Displaying Application Policies and Application Roles Using Fusion Middleware Control

This section explains how to use Fusion Middleware Control to access the pages that manage application roles and application policies.

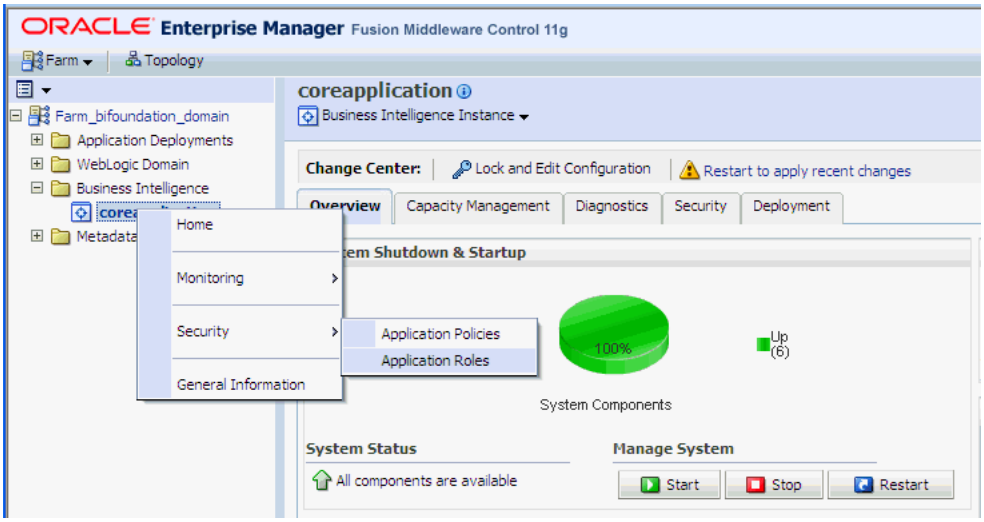
To display application policies and application roles using Fusion Middleware Control:

This method explains how to display **application policies** or **Application Roles** for Oracle Business Intelligence

1. Log in to Fusion Middleware Control.

For more information, see Section 1.6.2, "Using Oracle Fusion Middleware Control".

2. From the navigation pane expand the **Business Intelligence** folder and select **coreapplication**.
3. Choose one of the following options:
   - Right-click **coreapplication** and choose **Security** from the menu, then choose **Application Policies** or **Application Roles**.



   - Alternatively from the content pane, click **Business Intelligence Instance** to display a menu, then choose **Security**, and **Application Policies** or **Application Roles**.

     Other Fusion Middleware Control Security menu options are not available from these menus.

4. (Optional) An alternative option to Steps 2 and 3 is to expand the **WebLogic Domain** folder, select **bifoundation_domain** and right-click (or click the **WebLogic Domain** menu).

   A **Security** menu displays with appropriate menu options.

   Other Fusion Middleware Control menu options are available from this menu.

5. Choose **Application Policies** or **Application Roles** to display either the Application Policies page or the Application Roles page.
   - If the `obi` application stripe is displayed by default

     Oracle Business Intelligence policies or roles will be displayed.
   - If the `obi` application stripe is not displayed by default

     You must search using the `obi` application stripe to display Oracle Business Intelligence policies or roles.

   Figure 2–3 shows the **Application Policies** page and the default Oracle Business Intelligence application policies.

Figure 2–3 Application Policies Page

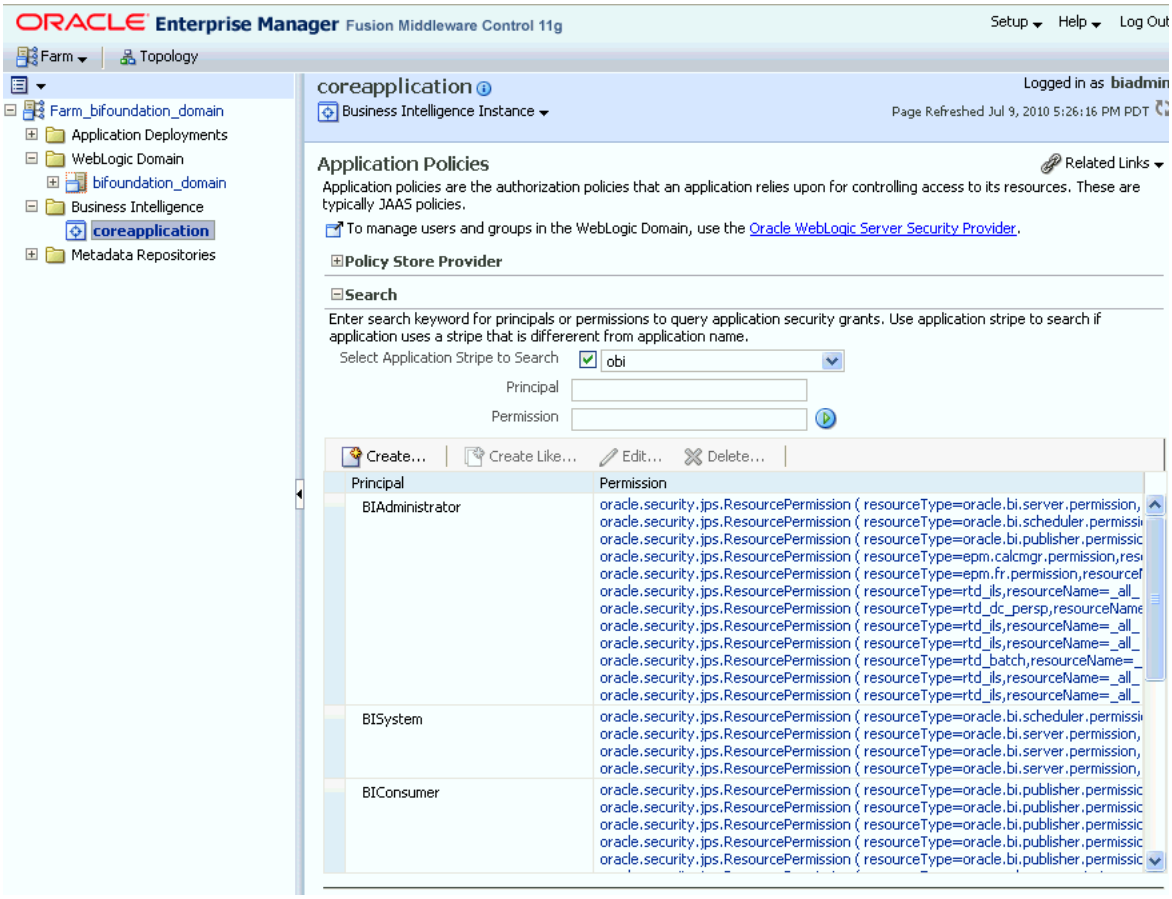
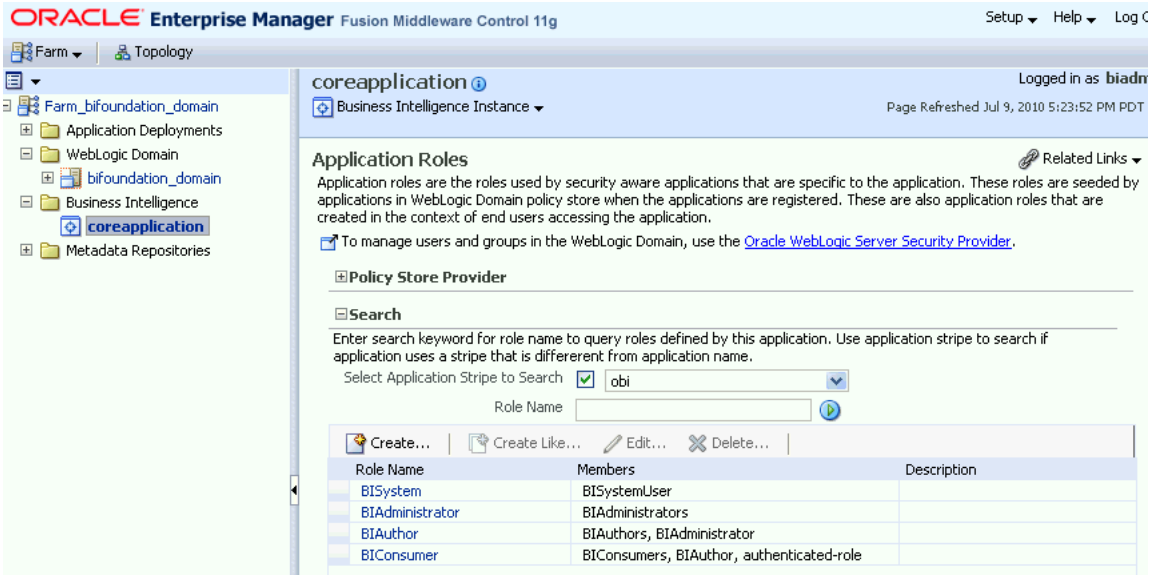


Figure 2–4 shows the **Application Roles** page with default Oracle Business Intelligence application roles.

Figure 2–4 Application Roles Page



2.4.2 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 2.4.2.1, "Overview"
- Section 2.4.2.2, "Creating an Application Role"
- Section 2.4.2.3, "Assigning a Group to an Application Role"

2.4.2.1 Overview

In a new Oracle Business Intelligence deployment, you typically create an application role for each type of business user activity in your Oracle Business Intelligence environment. For example, a typical deployment might require three application roles: BIConsumer, BIAuthor, 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 default application roles, see Section 2.1, "Working with the Default Users, Groups, and Application Roles".

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 assign 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 User Permissions Are Granted Using Application Roles".

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

Note: For advanced-level information about using a BI repository in offline mode, see Section 2.5.3.1, "About Managing Application Roles in the Metadata Repository".

2.4.2.2 Creating an Application Role

There are two methods for creating a new application role:

- **Create New** - Creates a new application role. You can add members at the same time or you can save the new role after naming it, and add members later.

- **Copy Existing** - Creates an application role 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 2.4.3, "Creating Application Policies Using Fusion Middleware Control".

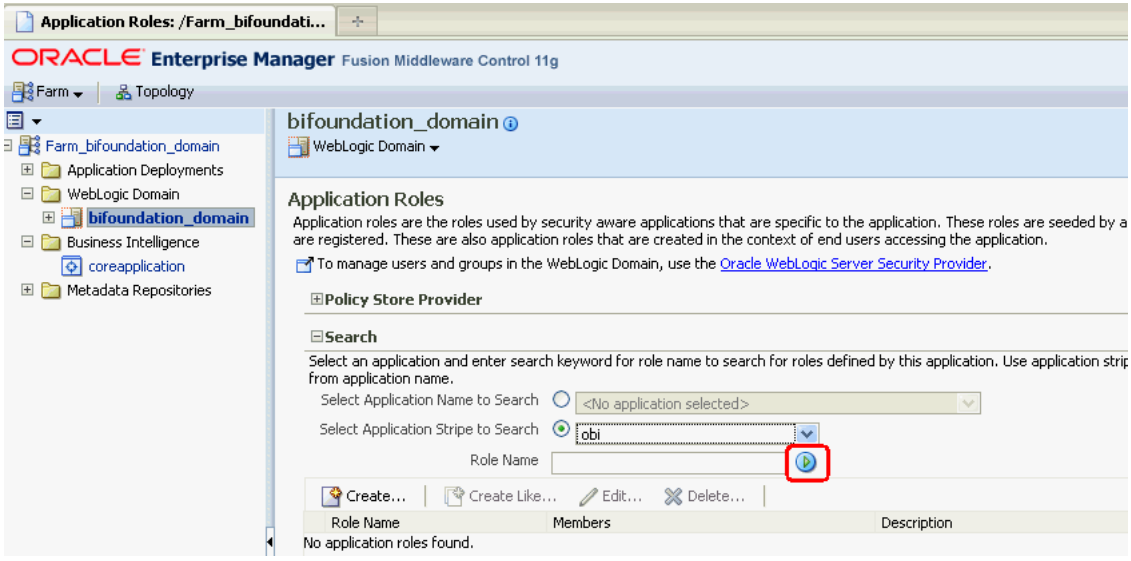
To create a new application role:

1. Log in to Fusion Middleware Control, and display the **Application Roles** page.

   For information, see Section 2.4.1, "Displaying Application Policies and Application Roles Using Fusion Middleware Control".

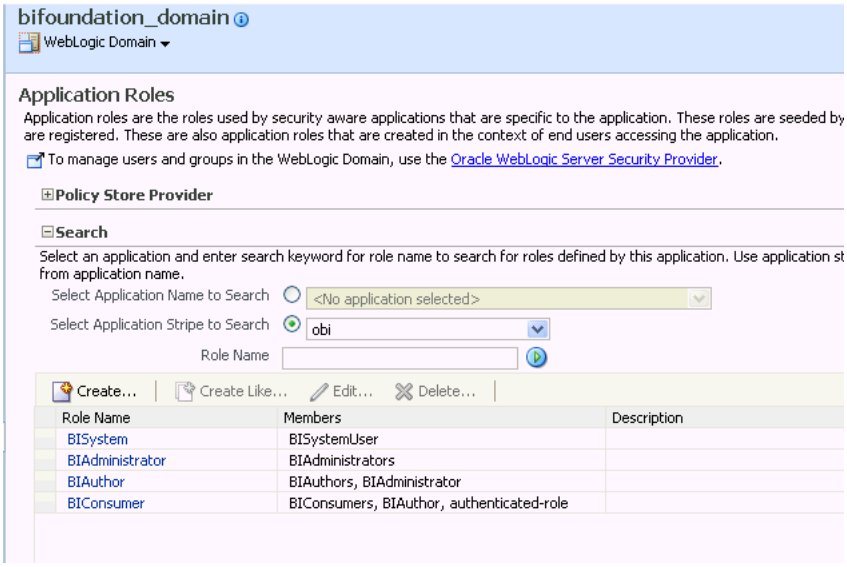
   Whether or not the obi application stripe is pre-selected and the application policies are displayed depends upon the method used to navigate to the **Application Roles** page.

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



The Oracle Business Intelligence application roles display. Figure 2–5 shows the default application roles.

Figure 2–5 Default Application Roles in Fusion Middleware Control



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 assigned 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, and display the **Application Roles** page.

   For information, see Section 2.4.1, "Displaying Application Policies and Application Roles Using Fusion Middleware Control".

   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**.

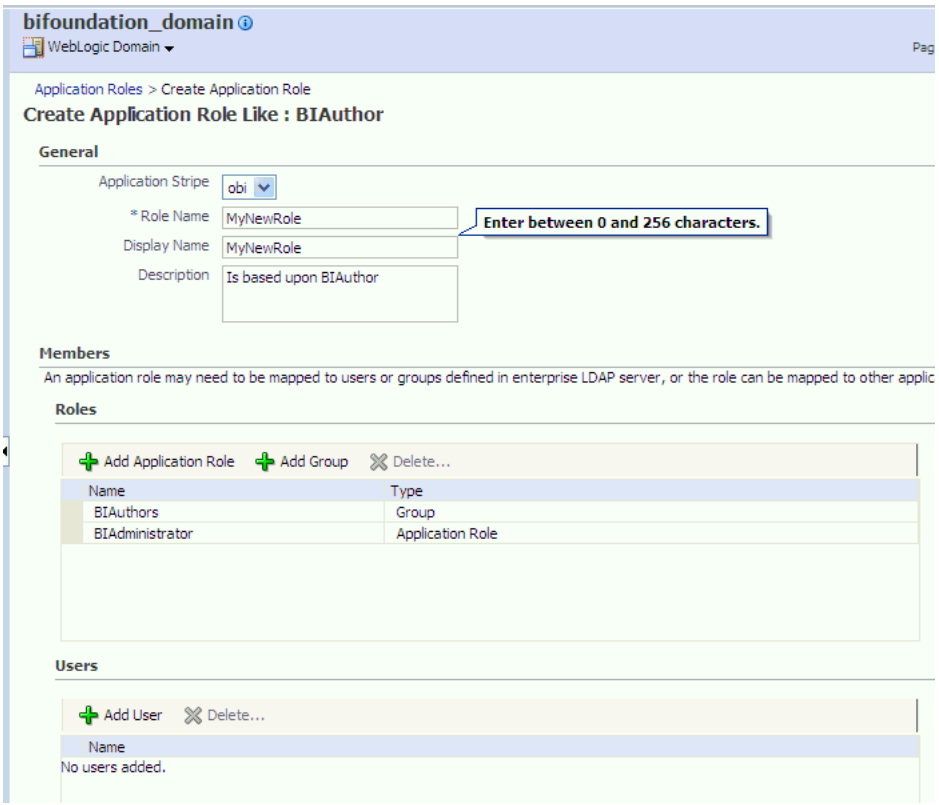
The Oracle Business Intelligence application roles display.

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 assigned to the original role.
5. Complete the **Role Name**, **Display Name**, and **Description** fields.

   Figure 2–6 shows creation of the new application role **MyNewRole**, based upon the default BIAuthor application role.

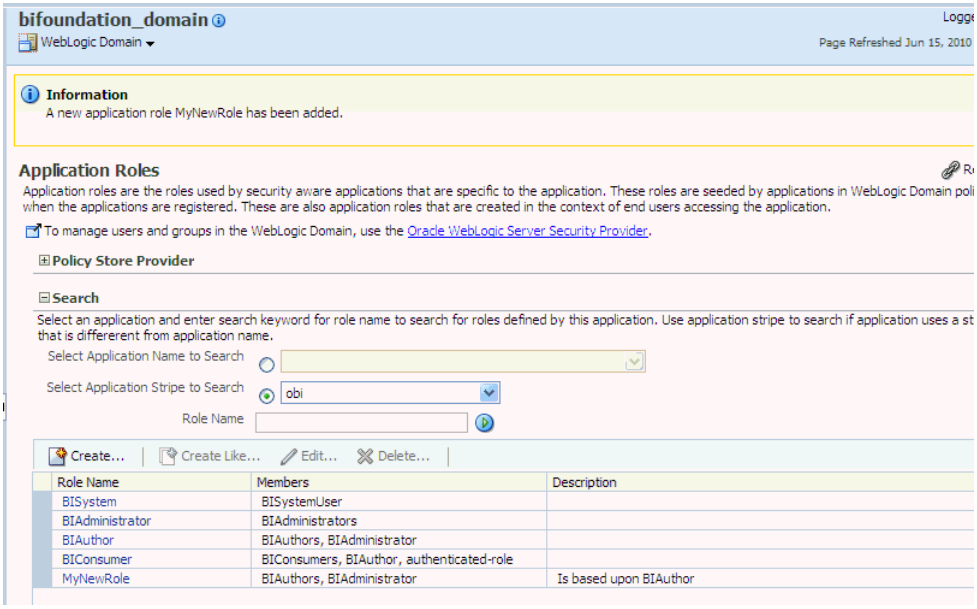
Figure 2–6 New Application Role Based on Default BIAuthor Role



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

   The newly-created application role displays in the table at the bottom of the page. Figure 2–7 shows the newly-created application role named **MyNewRole** that is based upon the default BIAuthor application role.

Figure 2–7 Newly Created Application Role



2.4.2.3 Assigning a Group to an Application Role

You assign 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 assign the group named BIMarketingGroup to the application role named BIConsumerMarketing.

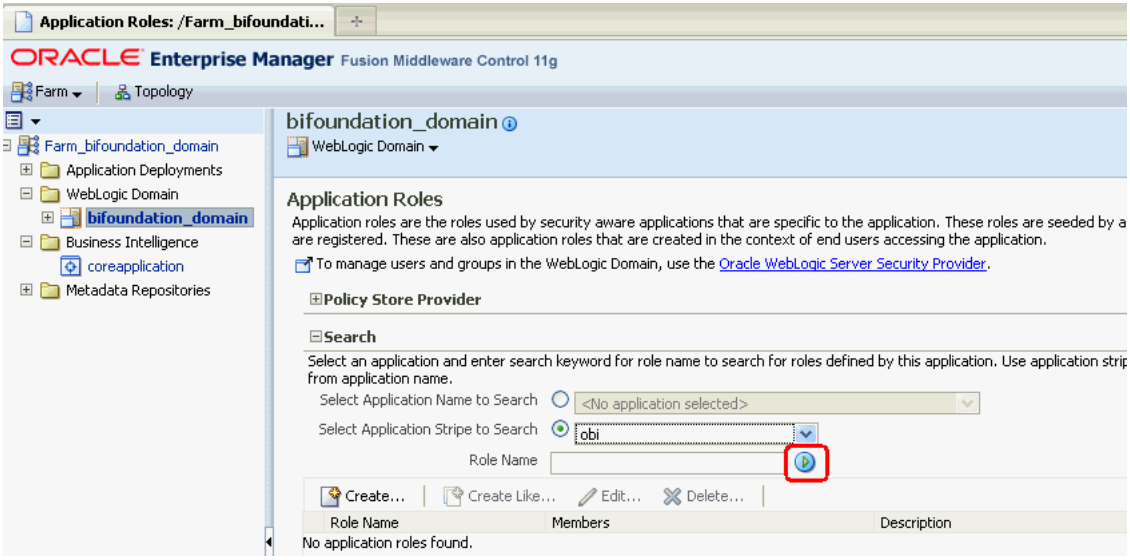
To assign a group to an application role:

1. Log in to Fusion Middleware Control, and display the **Application Roles** page.

   For information, see Section 2.4.1, "Displaying Application Policies and Application Roles Using Fusion Middleware Control".

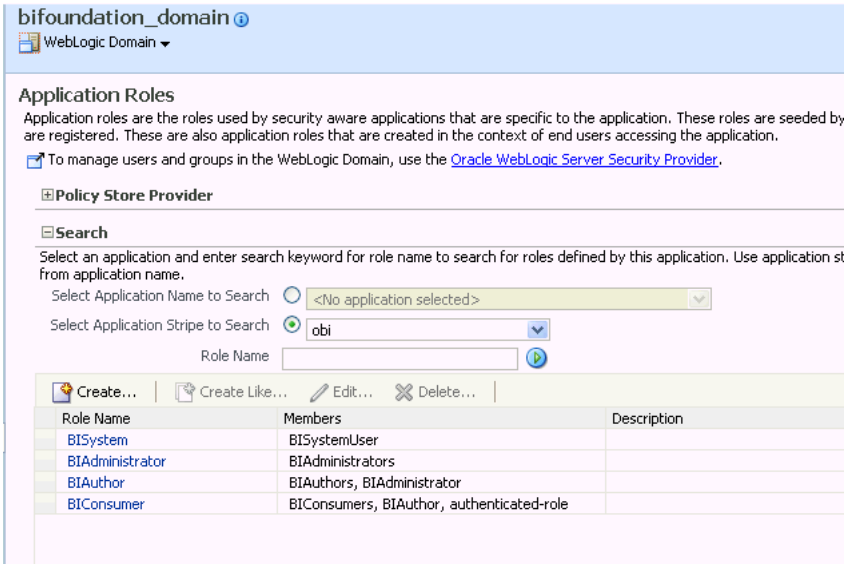
   Whether or not the obi application stripe is pre-selected and the application policies are displayed depends upon the method used to navigate to the **Application Roles** page.

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. Figure 2–8shows the default application roles.

Figure 2–8 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 assign 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.

2.4.3 Creating Application Policies Using Fusion Middleware Control

You can create application roles based on default preconfigured application policies, or you can create your own application policies.

Application policies do not apply privileges to the metadata repository or Oracle BI Presentation Catalog objects and functionality.

All Oracle Business Intelligence permissions are provided as part of the 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 assigned 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** - Create a new application policy and permissions are added to it.
- **Copy Existing** - Create new application policy 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 Application Security Guide*.

To create a new application policy:

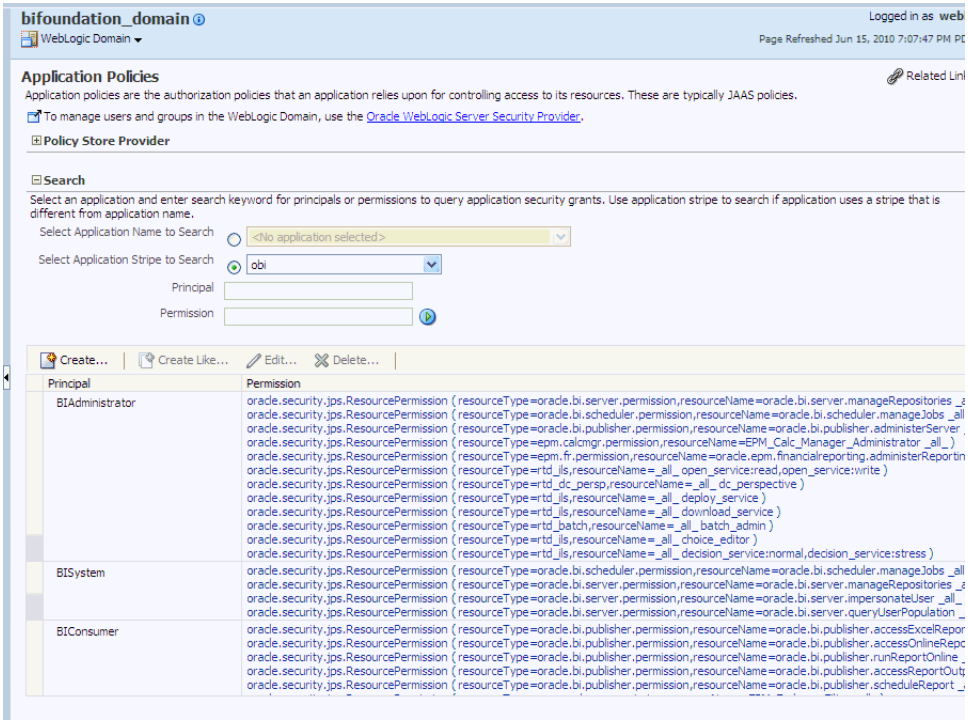
1. Log in to Fusion Middleware Control, and display the **Application Policies** page.

   For information, see Section 2.4.1, "Displaying Application Policies and Application Roles Using Fusion Middleware Control".

   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.

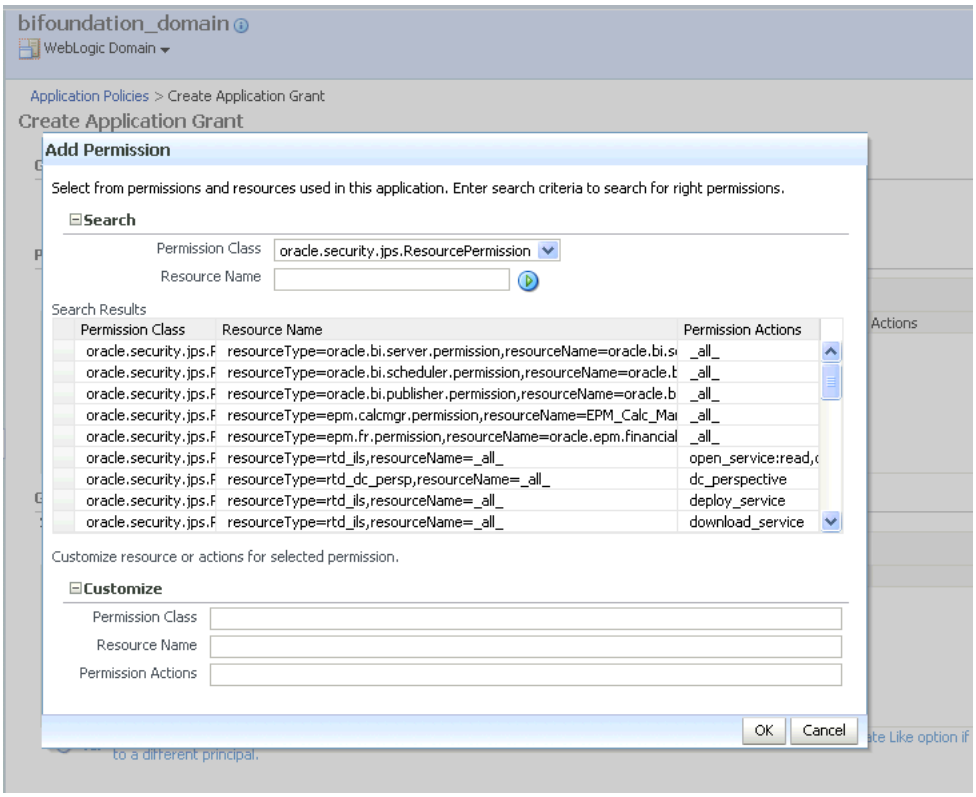
2. If necessary, select **Select Application Stripe to Search**, then select the **obi** from the list. Click the search icon next to **Role Name**.

   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.

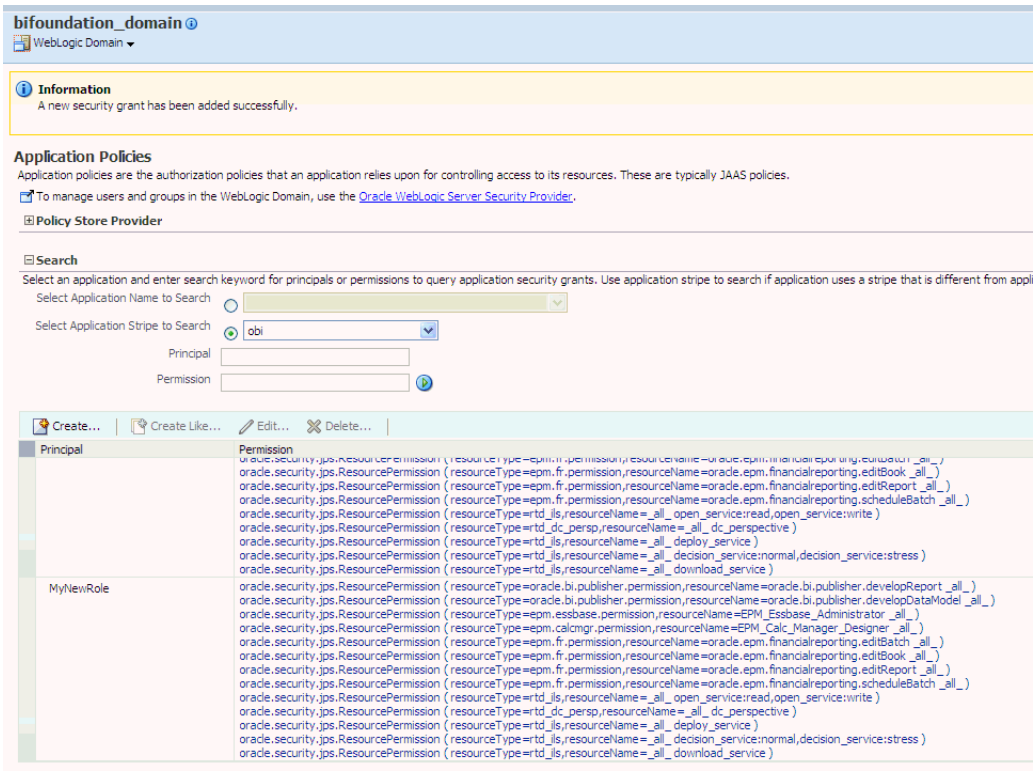


- Select the desired Oracle Business Intelligencer permission and click **OK**. Repeat until all desired permissions are selected. Selecting non-Oracle Business Intelligence permissions have no effect in the policy.
- To remove any items, select it and click **Delete**.

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, and display the **Application Policies** page.

   For information, see Section 2.4.1, "Displaying Application Policies and Application Roles Using Fusion Middleware Control".

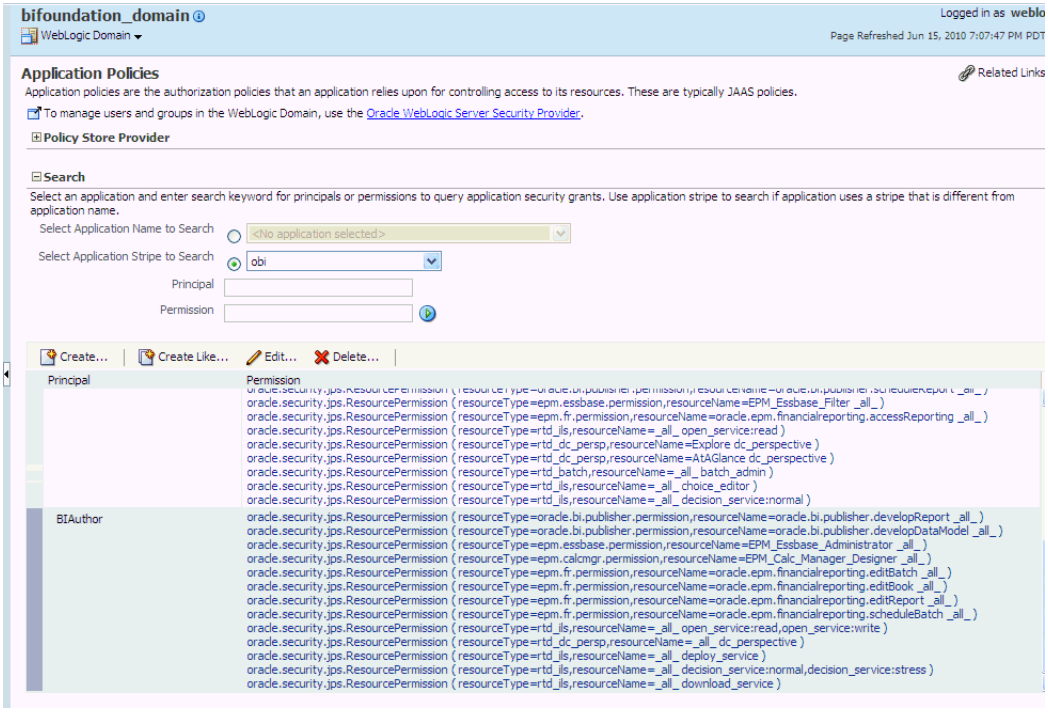
   Whether or not the obi application stripe is pre-selected and the application policies are displayed depends upon the method used to navigate to the **Application Policies** page.

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.

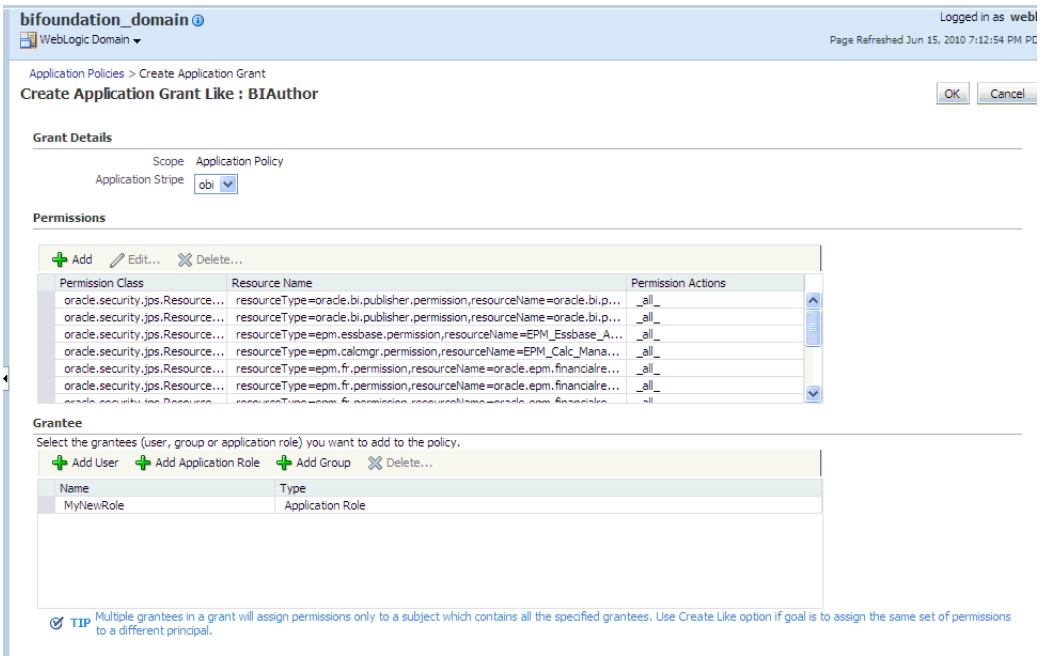
3. Select an existing policy from the table.

   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.

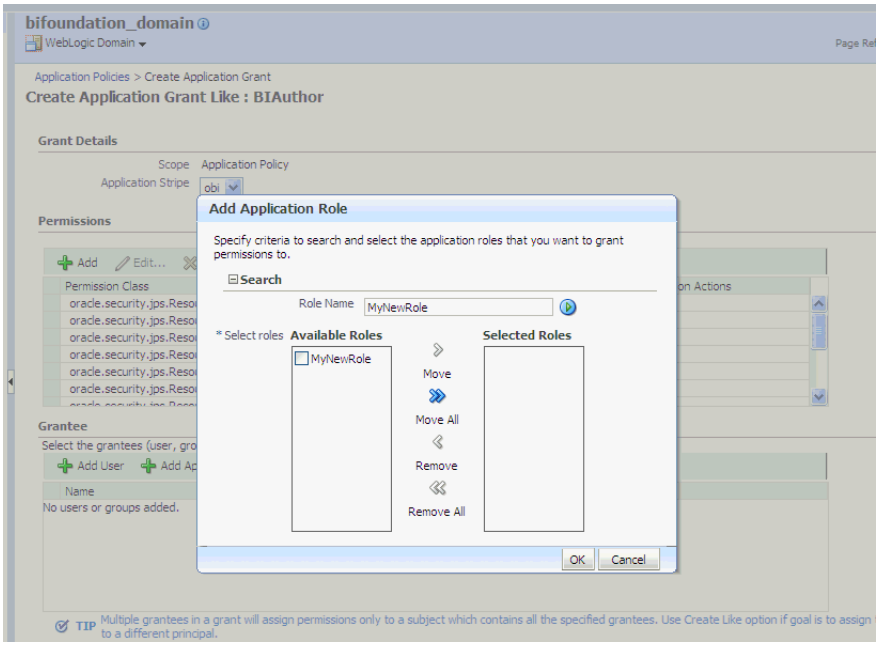
   The following figure shows the **Create Application Grant Like** dialog after the BIAuthor policy has been selected. Note that the **Permissions** section is completed with the permission grants for the BIAuthor policy.



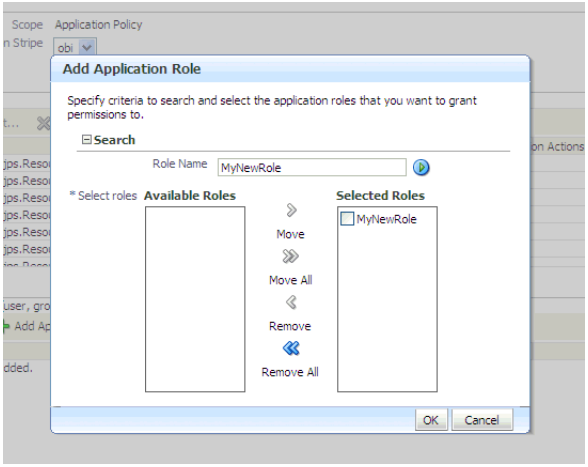
5. To remove any items, select it and click **Delete**.
6. To add application roles to the policy, click **Add Application Role** in the **Grantee** area to display the **Add Application Role** dialog.

The following figures use the **MyNewRole** application role as an example.

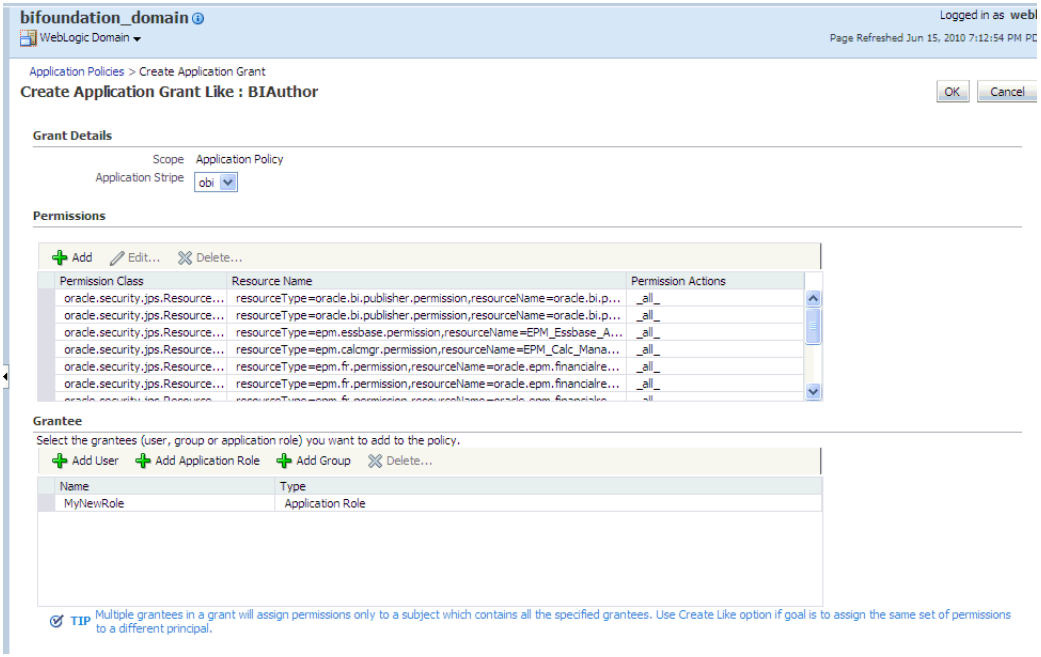
- Complete the **Search** area and click the blue search button next to the **Resource Name** field. The application roles matching the search are displayed.



- 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**.

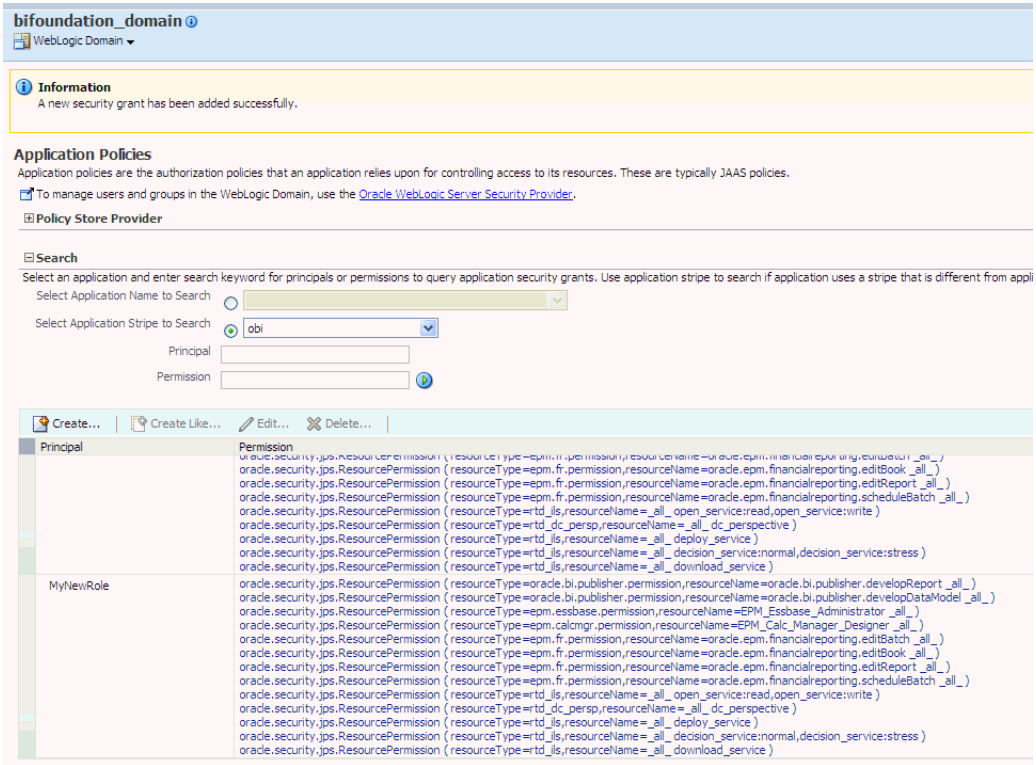


- Click **OK**. You are returned to the **Create Application Grant Like** dialog and the **Grantee** section is completed.



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

  The Principal and Permissions of the application policy just created are displayed in the table.



2.4.4 Modifying Application Roles Using Fusion Middleware Control

You can modify an application role by changing permission grants of the corresponding application policy (if the application role is a grantee of the application policy), or by changing its members, as follows:

- Section 2.4.4.1, "Adding or Removing Permission Grants from an Application Role"
- Section 2.4.4.2, "Adding or Removing Members from an Application Role"

> **Note:** Oracle recommends that you do not change the permission grants and membership for the default application roles named 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 Application Security Guide*.

2.4.4.1 Adding or Removing Permission Grants from an Application Role

Use this procedure if you want to change the permission grants for an application role. This is done by adding or removing the permission grants for the application policy which the application role is a grantee of.

To add or remove permission grants from an application policy:

1. Log in to Fusion Middleware Control, and display the **Application Policies** page.

   For more information, see Section 2.4.1, "Displaying Application Policies and Application Roles Using Fusion Middleware Control".

   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.

2.4.4.2 Adding or Removing Members from an Application Role

Members can be added to or deleted from an application role using Fusion Middleware Control. You must perform these tasks in the WebLogic Domain where Oracle Business Intelligence is installed (for example, in bifoundation_domain). Valid members of an application role are users, groups, or other application roles. Being assigned to an application role is to become a member of an application role. Best practice is to assign 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, and display the **Application Roles** page.

   For information, see Section 2.4.1, "Displaying Application Policies and Application Roles Using Fusion Middleware Control".

   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.

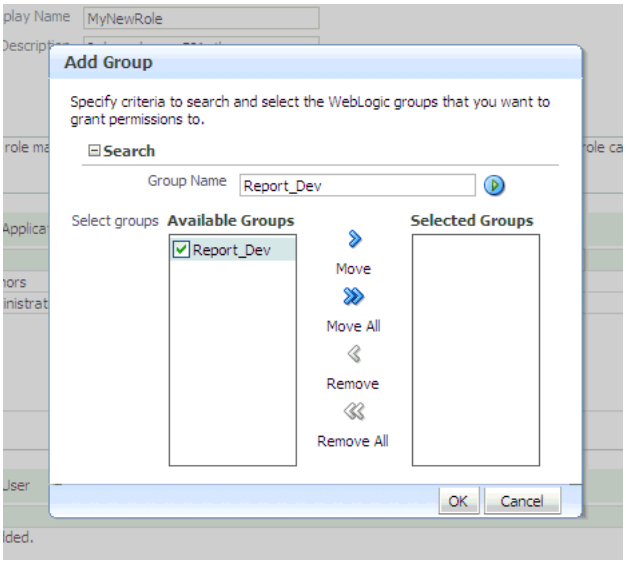
   You can add or delete members from the **Edit Application Role** page. Valid members are application roles, groups, and users.

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

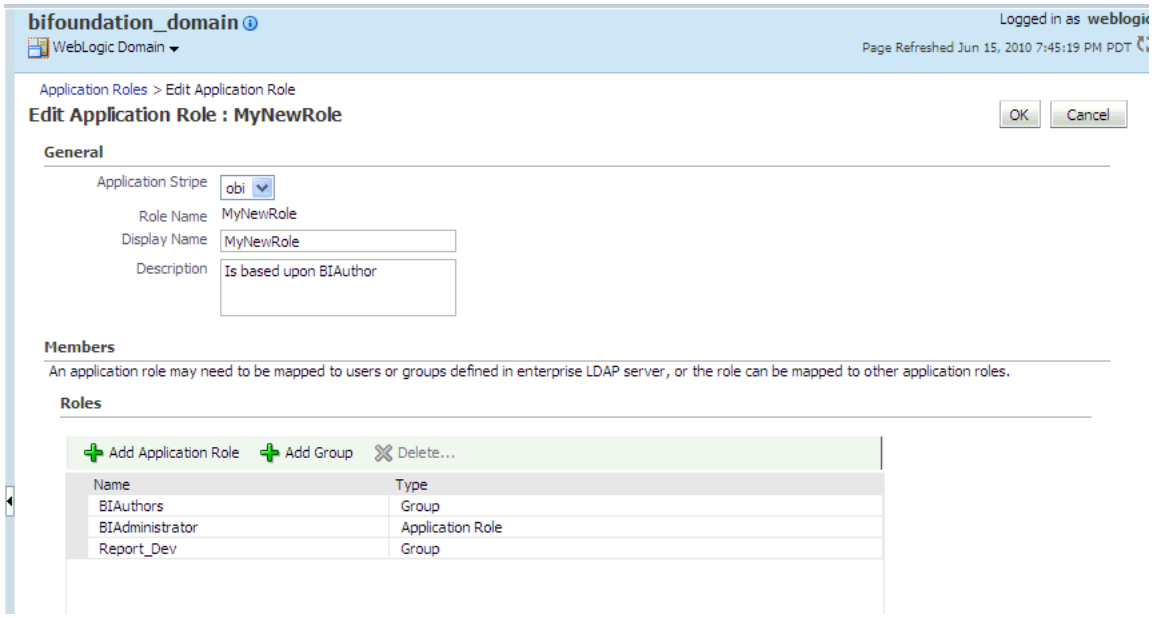
   - **To delete a member**: Select the **Name** of the member to activate the **Delete** button. Click **Delete**.
   - **To add a member**: Click the **Add** button that corresponds to the member type being added. Select from **Add Application Role**, **Add Group**, and **Add User**.

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.



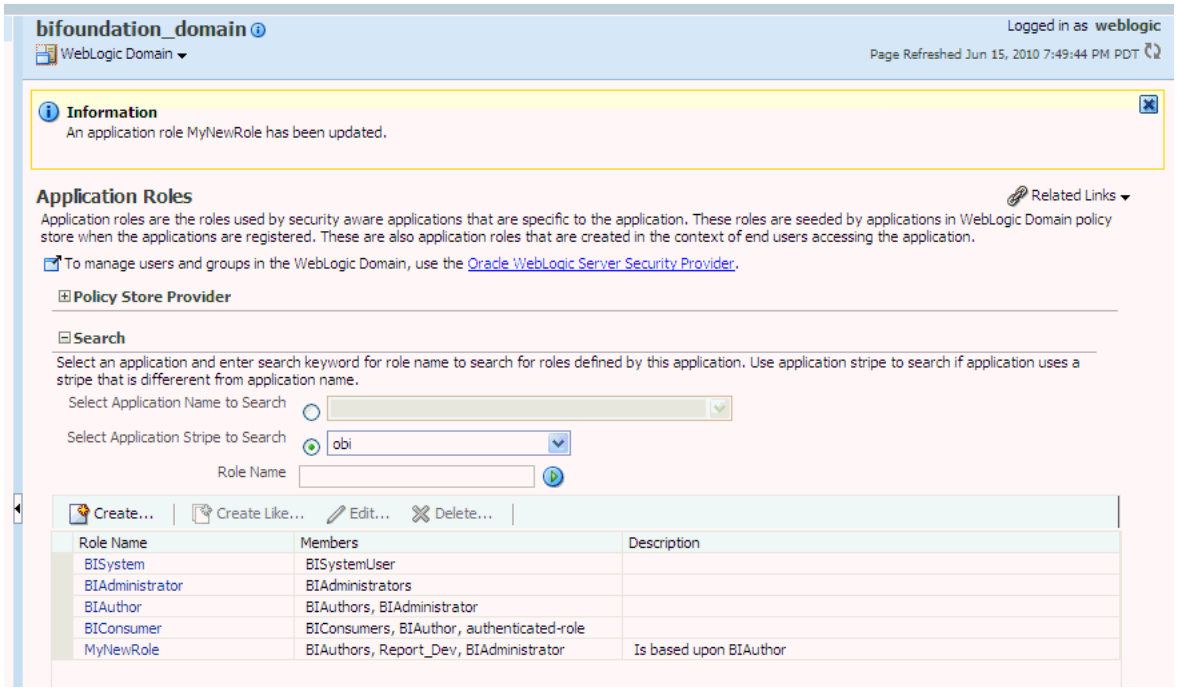
The added member displays in the **Members** column corresponding to the application role modified in the **Application Roles** page. For example, the following figure shows the **Edit Application Role** page for the **MyNewRole** application role after the **Report_Dev** group has been added.



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 Application Security Guide*.

2.5 Managing Metadata Repository Privileges Using the Oracle BI Administration Tool

This section explains how to use the Oracle BI Administration Tool to configure security in the Oracle BI repository, and contains the following topics:

- Section 2.5.1, "Overview"
- Section 2.5.2, "Setting Repository Privileges for an Application Role"
- Section 2.5.3, "Advanced Security Configuration Topics"

2.5.1 Overview

You use Identity Manager in the 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 about using the Oracle BI Administration Tool to configure security, see Section 1.6.3, "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 *Oracle Fusion Middleware Reference Guide for Oracle Business Intelligence Applications*.

2.5.2 Setting Repository Privileges for an Application Role

The default application roles (that is, BIConsumer, BIAuthor, and BIAdministrator) are preconfigured 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.

Note: In addition, you might assign Oracle BI Presentation Catalog privileges to a new application role. For more information, see Section 2.6.3, "Setting Presentation Services Privileges for Application Roles".

To set repository permissions for an application role:

1. Open the repository in the Oracle BI Administration Tool (in Online mode).

   For more information, see Section 1.6.3, "Using Oracle BI Administration Tool".

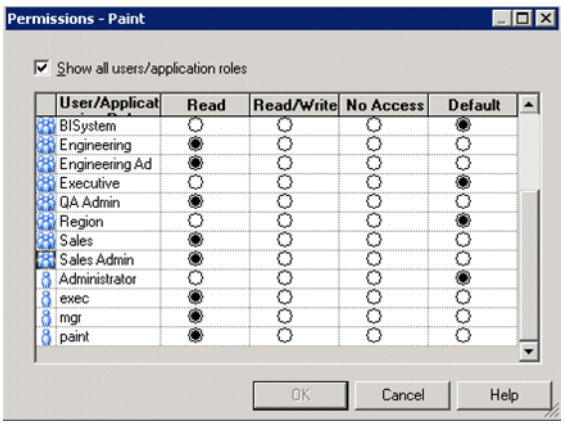
2. In the Presentation panel, navigate to the subject area or sub-folder for which you want to set permissions.

3. Right-click the subject area or sub-folder and select **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.

2.5.3 Advanced Security Configuration Topics

This section contains advanced topics.

2.5.3.1 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. The 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 the 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 the Administration Tool to facilitate offline repository development. But this is just a placeholder visible in the Administration Tool and is not an actual application role. You cannot created an actual application role in the 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 the 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*.

2.6 Managing Presentation Services Privileges Using Application Roles

This section explains how to manage Presentation Services privileges using application roles in Presentation Services Administration Manage Privileges page, and contains the following topics:

- Section 2.6.1, "Overview"
- Section 2.6.2, "About Presentation Services Privileges"
- Section 2.6.3, "Setting Presentation Services Privileges for Application Roles"
- Section 2.6.4, "Advanced Security Configuration Topics"

2.6.1 Overview

Privileges that are stored in Presentation Services control access to features such as the creation analyses and dashboards. 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 Presentation Services privileges, see Section A.2.1, "Changes Affecting Security in Presentation Services".

Note: Assigning an application role to be a member of a Catalog group creates complex group inheritance and maintenance situations and is not considered a best practice.

When groups are assigned to application roles, the group members are automatically granted associated privileges in Presentation Services. 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.

2.6.2 About Presentation Services Privileges

Presentation Services privileges are maintained in the Presentation Services Administration Manage Privileges page, and they grant or deny access to Presentation Services features, such as the creation of analyses and dashboards. Presentation Services privileges have no effect in other Oracle Business Intelligence components.

Being a member of a group assigned to a default application role grants Presentation Services 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 Presentation Services privileges granted by a default application role can be modified by adding or removing default privilege grants using the **Manage Privileges** page in Presentation Services Administration.

Whenever a new catalog is created, it is populated with the default application role to Presentation Services 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 Presentation Services 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.

2.6.3 Setting Presentation Services Privileges for Application Roles

If you create an application role, you must set appropriate Presentation Services privileges to enable users with the application 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.

Presentation Services privileges cannot be assigned using the administrative interfaces used to manage the policy store. If you create a new application role to grant Oracle Business Intelligence permissions, then you must the set Presentation Services privileges for the new role in addition to any Oracle Business Intelligence permissions.

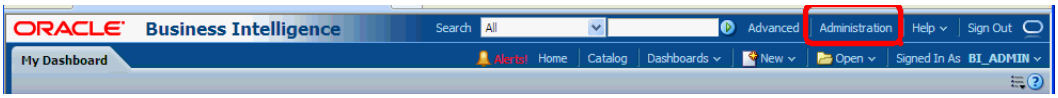
Note: Presentation Services privileges can be assigned to a new application role programmatically using SecurityService Service. For more information, see "SecurityService Service" in *Oracle Fusion Middleware Integrator's Guide for Oracle Business Intelligence Enterprise Edition*

To set Presentation Services privileges for an application role:

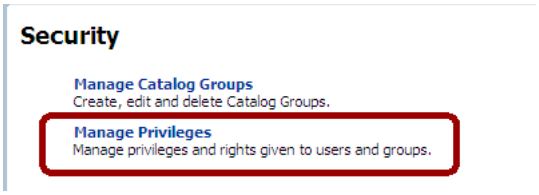
1. Log in to Oracle BI Presentation Services as a user with Administrator privileges.

   For more information, see Section 1.6.4, "Using Presentation Services Administration".

2. From the Home page in Presentation Services, select **Administration**.

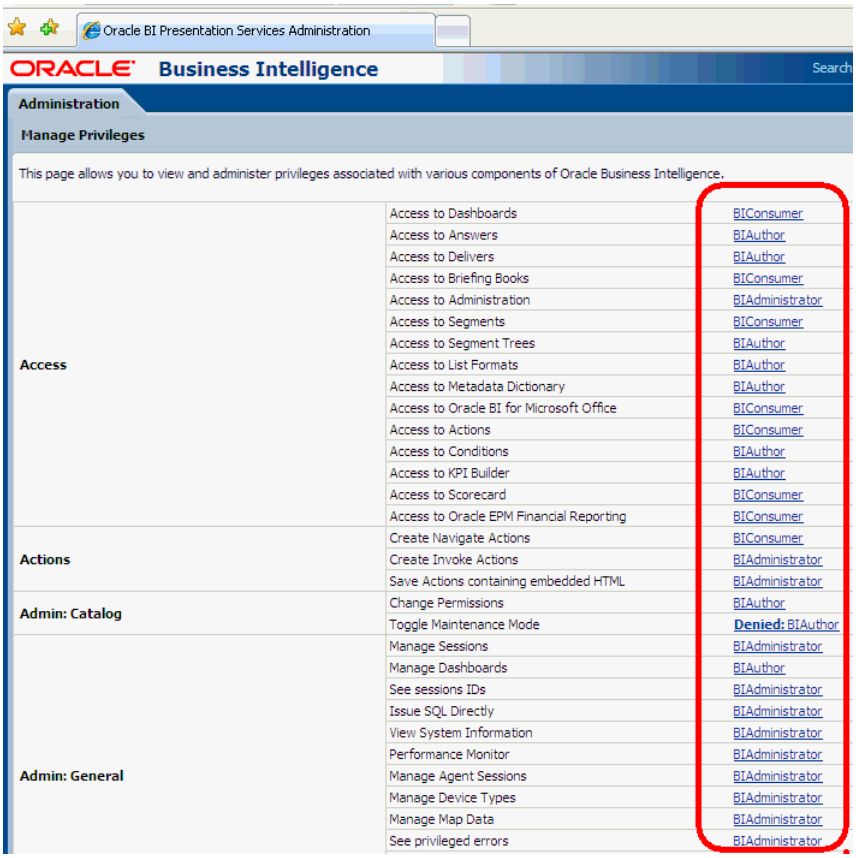


Note: If you log in as a user without Administrator privileges, the Administration option is not displayed.



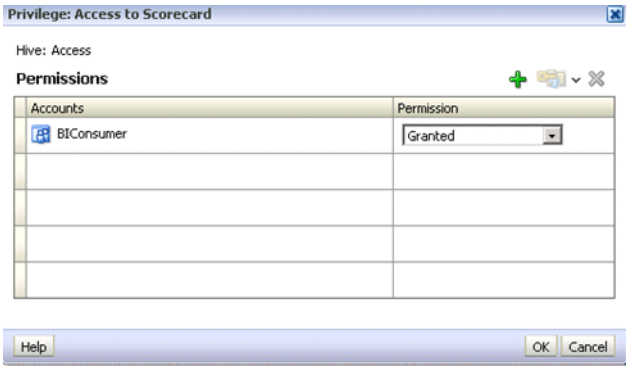
3. In the Security area, click **Manage Privileges** to display the Manage Privileges page.

   This page enables you to view application roles for Presentation Services privileges.



4. Click an application role next to the privilege that you want to administer.

   For example, to administer the privilege named Access to Scorecard for the application role named BIConsumer, you would click the **BIConsumer** link next to Access to Scorecard.



Use the Privilege <*privilege_name*> dialog to add application roles to the list of permissions, and grant and revoke permissions from application roles. For example, to grant the selected privilege to an application role, you must add the application role to the **Permissions** list.

5. Add an application role to the **Permissions** list, as follows:
   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** list.

> **Note:** Explicitly *denying* a Presentation Services permission 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 Presentation Services 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 Presentation Services Catalog privileges. You can then delete the original Catalog group from Presentation Services.

2.6.4 Advanced Security Configuration Topics

This section contains advanced topics.

2.6.4.1 About Encryption in BI Presentation Services

The BI Server and 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 BI Server uses the Hypertext 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 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.

2.7 Managing Data Source Access Permissions Using Oracle BI Publisher

This section discusses managing the data source access permissions that are stored in Oracle BI Publisher, using the Oracle BI Publisher Administration pages. Data source access permissions control application role access to data sources. A user must be assigned to an application role which is granted specific data source access permissions to enable the user to perform the following tasks:

- Create a data model against the data source.
- Edit a data model against a data source.
- View a report created with a data model built from the data source.

For more information regarding data source security in published reporting, see "Granting Access to Data Sources" in the *Oracle Fusion Middleware Administrator's and Developer's Guide for Oracle Business Intelligence Publisher*.

2.8 Enabling High Availability of the Default Embedded Oracle WebLogic Server LDAP Identity Store

To enable high availability of the default embedded Oracle WebLogic Server LDAP identity store in a clustered environment, you configure the virtualize attribute. When you set the virtualize attribute value to true, Managed servers are able to use a copy of the embedded default Oracle WebLogic Server LDAP identity store.

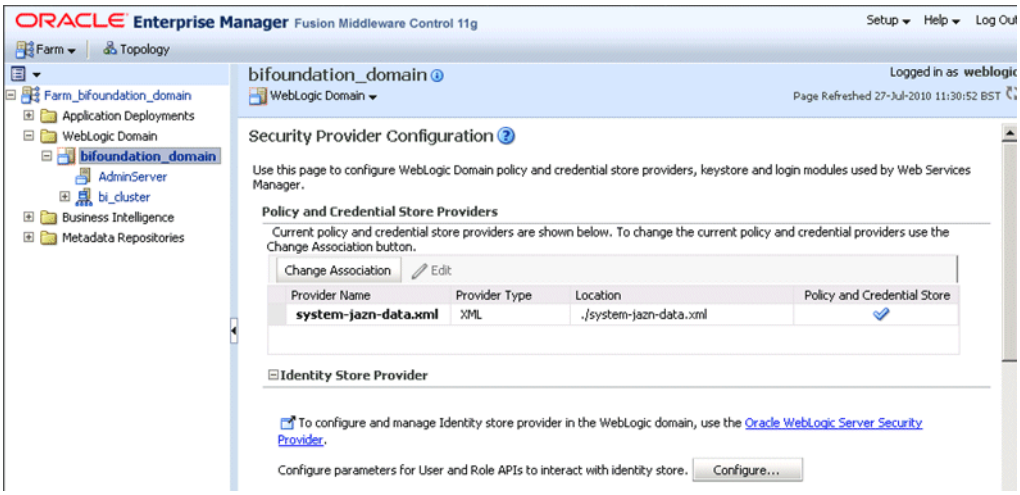
To configure the virtualize attribute for high availability of the default embedded Oracle WebLogic Server LDAP identity store:

1. Log in to Fusion Middleware Control.

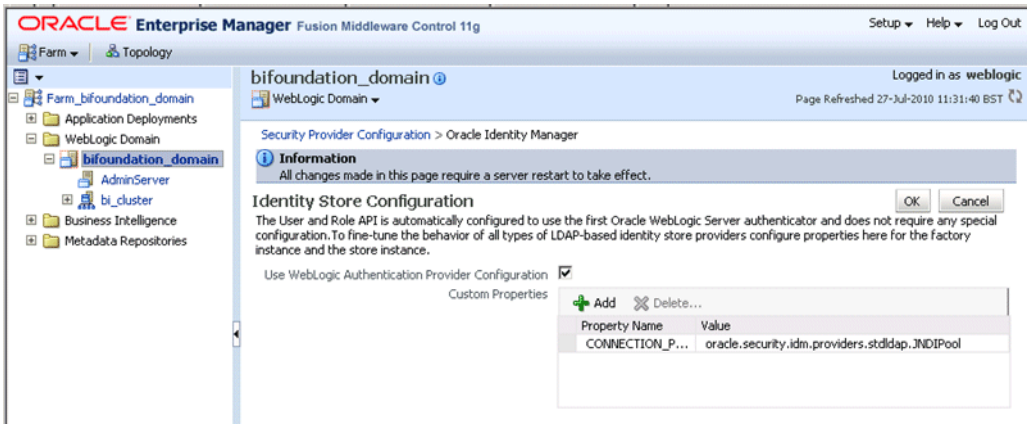
   For more information, see Section 1.6.2, "Using Oracle Fusion Middleware Control".

2. From the navigation pane expand the **WebLogic Domain** folder and select **bifoundation_domain**.

3. Right-click bifoundation_domain and select Security, then Security Provider Configuration to display the Security Provider Configuration page.



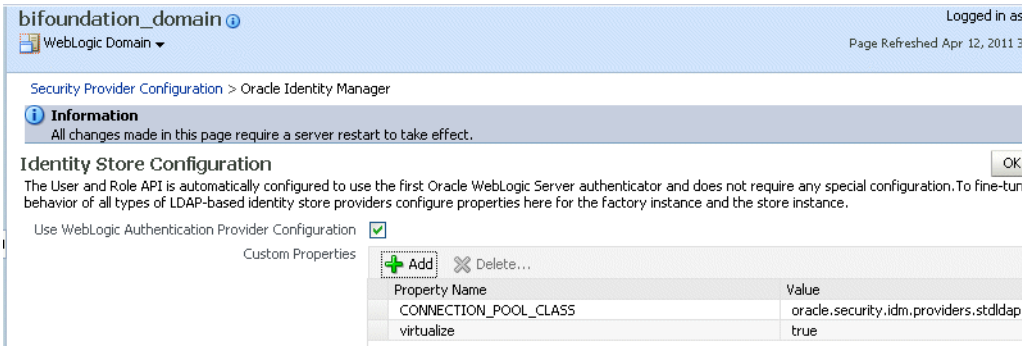
4. In the Identity Store Provider area, click **Configure** to display the Identity Store Configuration page.



5. In the Custom Properties area, use the Add option to add a Custom Property called virtualize.

   Figure 2–9 shows an example set of Custom Properties including a new property called virtualize with its value set to true.

Figure 2–9 Identity Store Configuration Page Showing New Custom Property



6. Click **OK** to save the changes.

7. Restart the Administration Server, any Managed Servers, and Oracle BI components.

3

Using Alternative Authentication Providers

This chapter explains how to configure Oracle Business Intelligence to use alternative directory servers for authentication instead of using the default Oracle WebLogic Server LDAP directory. This chapter explains how to set up Oracle Business Intelligence to use Oracle Internet Directory, Active Directory, and other authentication providers, and also explains how to use OID LDAP as a policy store, and credential store.

> **Note:** For a detailed list of security setup steps, see Section 1.7, "Detailed List of Steps for Setting Up Security In Oracle Business Intelligence".

This chapter contains the following sections:

3.1 Introduction

When you use an alternative authentication provider, you will typically use administrative tools provided by your provider vendor to set up your users and groups. You can then assign 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 assigning users and groups to application roles, see Section 2.4, "Managing Application Roles and Application Policies Using Fusion Middleware Control".

You continue to use the other Oracle Business Intelligence tools (such as, the Oracle BI Administration Tool, Fusion Middleware Control, and the Presentation Services Administration Page) 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, you select the authentication provider from the **Type** list in the **Create a New Authentication Provider** page. For more information, see System Requirements and Certification.

You can configure more than one supported authentication provider. For more information, see Section 3.4.5, "Configuring Multiple Authentication Providers Using Fusion Middleware Control".

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

3.2 High-Level Steps for Configuring an Alternative Authentication Provider

To configure an alternative authentication provider:

Prerequisite: Ensure that only the Administration Server is running.

1. Set up and configure groups and users to enable Oracle Business Intelligence to use an alternative authentication provider as described in Section 3.3, "Prerequisites for Using Alternative Authentication Providers".
2. Configure Oracle Business Intelligence to use authentication providers as described in Section 3.4, "Configuring Alternative Authentication Providers".
3. Configure the User Name Attribute in the identity store to match the User Name Attribute in the authentication provider as described in Section 3.5, "Configuring User and Group Name Attributes In the Identity Store".
4. Go to the **myrealm\Users and Groups** tab to verify that the users and groups from the alternative authentication provider are displayed correctly. If the users and groups are displayed correctly, then proceed to Step 5. Otherwise, reset your configuration settings and retry.
5. Configure a new trusted user account for a user in the alternative authentication provider to match the account for DefaultAuthenticator as described in Section 3.7, "Configuring a New Trusted User (BISystemUser)".
6. Update the user GUIDs to be the values in the alternative authentication provider as described in Section 3.8, "Refreshing User GUIDs".
7. Assign application roles to the correct groups (enterprise roles) for the new identity store, using Fusion Middleware Control.

   For more information, see Section 2.4.4.2, "Adding or Removing Members from an Application Role".

3.3 Prerequisites for Using Alternative Authentication Providers

Before you configure an Oracle Business Intelligence installation to use an alternative authentication provider, you must make sure that groups and users exist, and are

correctly configured in the alternative authentication provider. They can then be associated with corresponding Oracle Business Intelligence application roles that already exist in the Oracle Business Intelligence installation.

To set up users and groups in an alternative authentication provider:

1. Create groups in the alternative authentication provider that can be assigned to existing Oracle Business Intelligence application roles. For example:

   BIAdministrators, BISystemUsers, BIAuthors, BIConsumers

2. Create users in the alternative authentication provider, that correspond to the groups created in Step 1. For example:

   BIADMIN, BISYSTEM, BIAUTHOR, BICONSUMER.

3. Assign the users to their respective groups, in the alternative authentication provider.

   For example you would assign the BIADMIN user to the BIAdministrators group, and the BISYSTEM user to the BISystemUsers group.

4. Make the BIAuthors group part of the BIConsumers group in the alternative authentication provider.

   This grouping enables BIAuthors to inherit permissions and privileges of BIConsumers.

3.4 Configuring Alternative Authentication Providers

The following procedures describe how to configure one or more authentication providers instead of the default Oracle WebLogic Server LDAP directory.

- Section 3.4.1, "Configuring Oracle Internet Directory as the Authentication Provider"
- Section 3.4.2, "Configuring Active Directory as the Authentication Provider"
- Section 3.4.3, "Configuring a Database as the Authentication Provider"
- Section 3.4.4, "Configuring LDAP as the Authentication Provider and Storing Groups In a Database"
- Section 3.4.5, "Configuring Multiple Authentication Providers Using Fusion Middleware Control"
- Section 3.4.6, "Setting the JAAS Control Flag Option"
- Section 3.4.7, "Configuring a Single LDAP Authentication Provider as the Authenticator"

> **Note:** This section shows settings for specific authentication providers. However, the instructions can also be used as a general guide for other authentication providers.

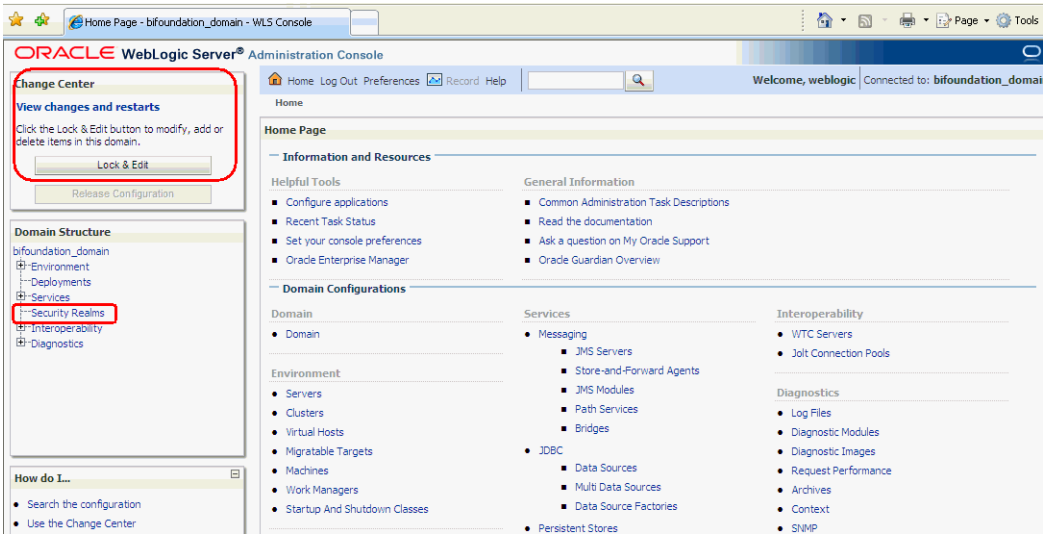
3.4.1 Configuring Oracle Internet Directory as the Authentication Provider

This procedure illustrates how to reconfigure your Oracle Business Intelligence installation to use Oracle Internet Directory(OID LDAP).

To configure OID LDAP as the authentication provider:

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

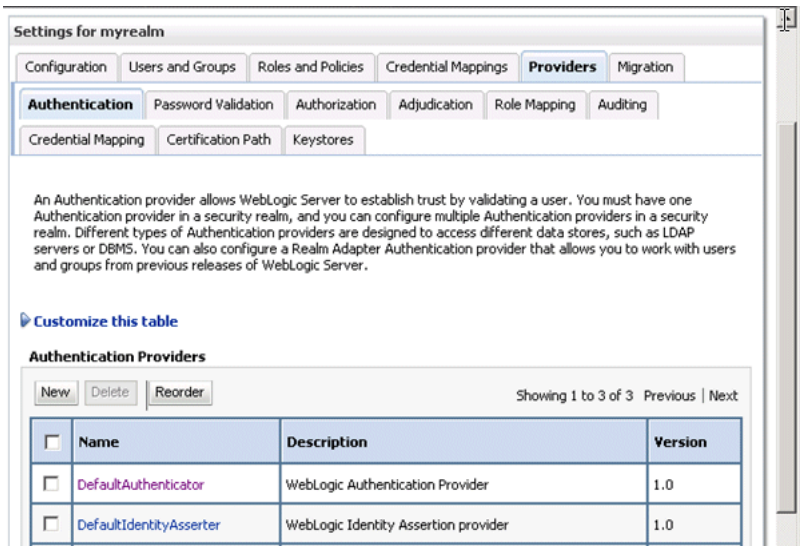
   For more information, see Section 1.6.1, "Using Oracle WebLogic Server Administration Console".



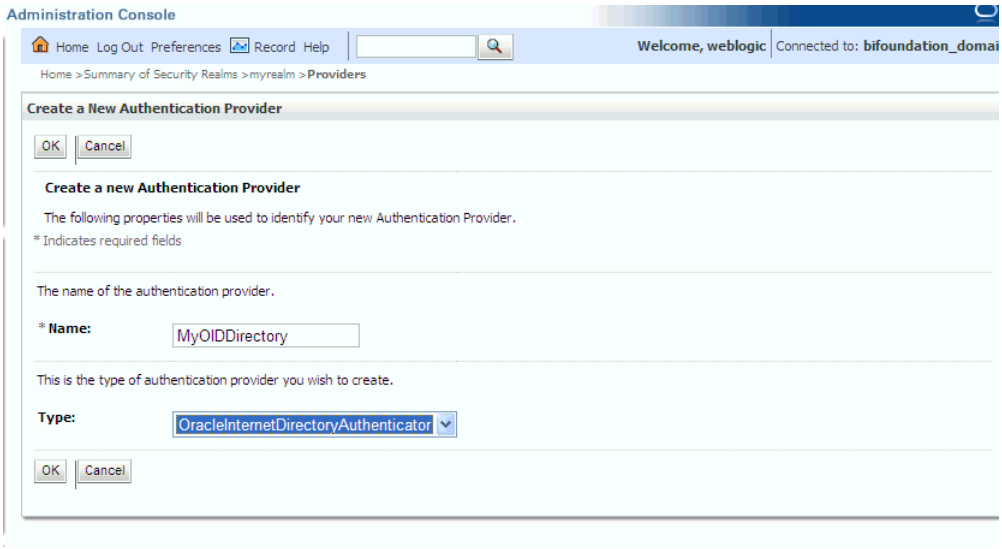
2. Select **Security Realms** from the left pane and click **myrealm**.

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

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

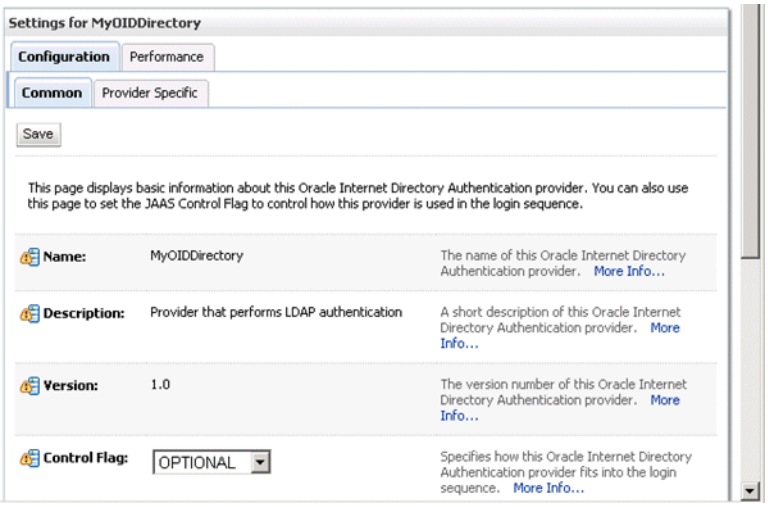


4. Click **New** to launch the **Create a New Authentication Provider** page.



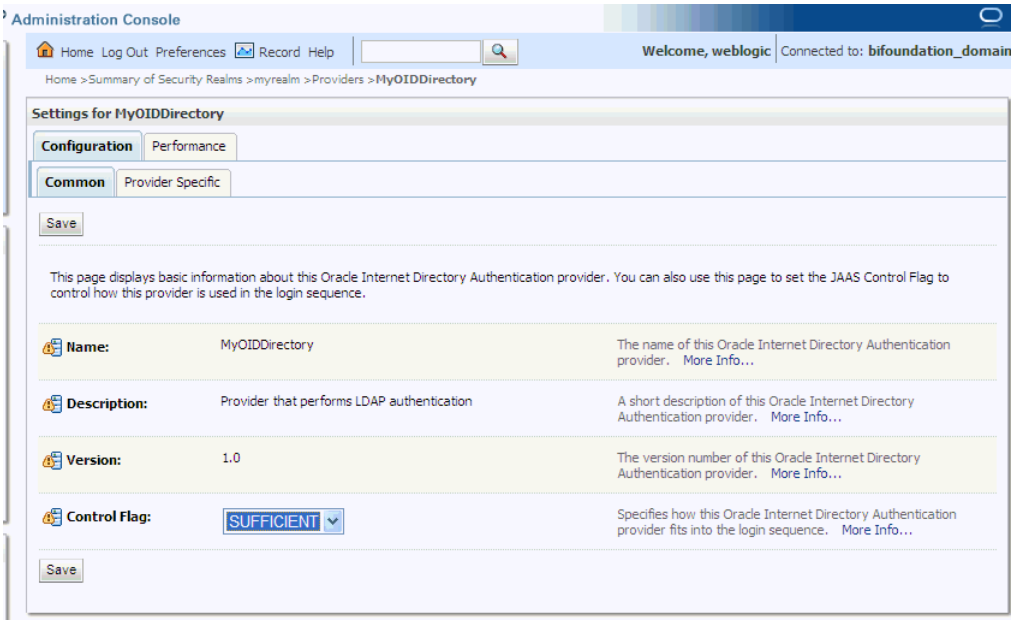
5. Enter values in the **Create a New Authentication Provider** page as follows:
   - **Name**: Enter a name for the authentication provider. For example, MyOIDDirectory.
   - **Type**: Select OracleInternetDirectoryAuthenticator from the list.
   - Click **OK** to save the changes and display the authentication providers list updated with the new authentication provider.

6. Click MyOIDDirectory in the **Name** column of the **Authentication Providers** table to display the **Settings** page.

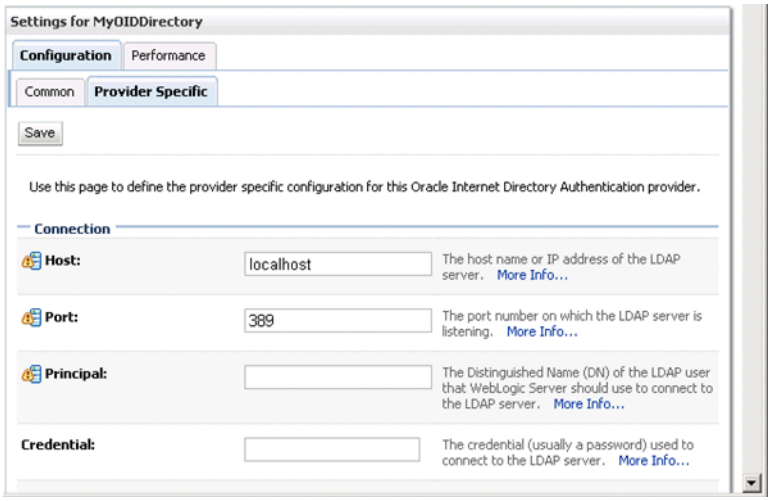


7. Display the **Configuration\Common** tab, and use the **Control Flag** list to select 'SUFFICIENT', then click **Save**.

   For more information, see Section 3.4.6, "Setting the JAAS Control Flag Option".



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



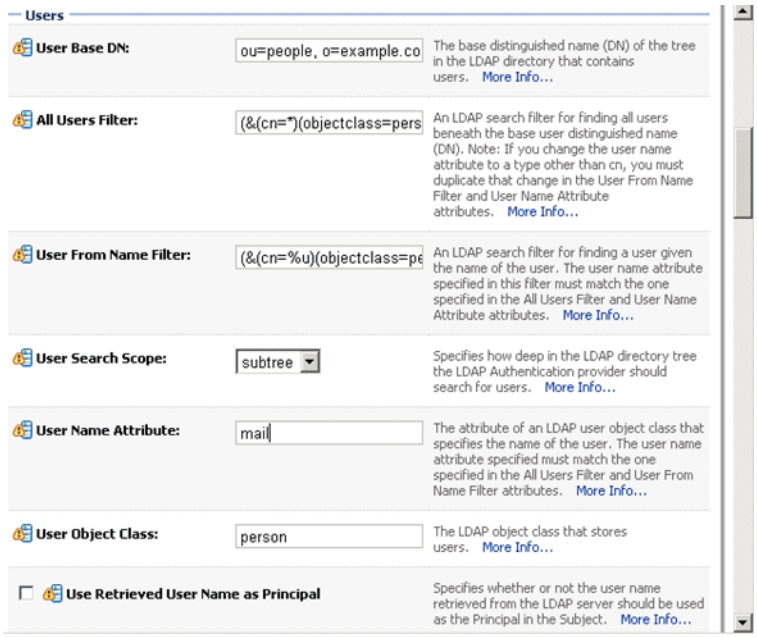
9. Use the Provider Specific tab to specify the following details:

Section Name	Field Name	Description
Connection	Host	The host name of the Oracle Internet Directory server.
Connection	Port	The port number on which the Oracle Internet Directory server is listening.

Section Name	Field Name	Description
Connection	Principal	The distinguished 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.
Connection	Credential	The 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.
Users	All Users Filter	The LDAP search filter. Click More Info... for details.
Users	User From Name Filter	The LDAP search filter. Click More Info... for details.
Users	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 3.5.1, "Configuring the User Name Attribute In the Identity Store".
General	GUID attribute	The attribute used to define object GUIDs in OID LDAP. orclguid **Note**: You should not normally change this default value, however, if you do, you must also specify the changed value in Fusion Middleware Control, as described in the task Section 3.6, "Configuring the GUID Attribute In the Identity Store".

Figure 3–1 shows the Users area of the Provider Specific tab.

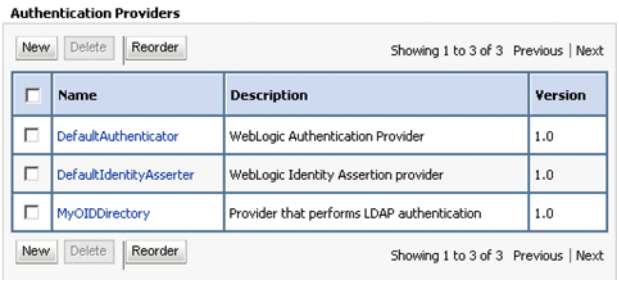
Figure 3–1 Provider Specific Tab - Users Area



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

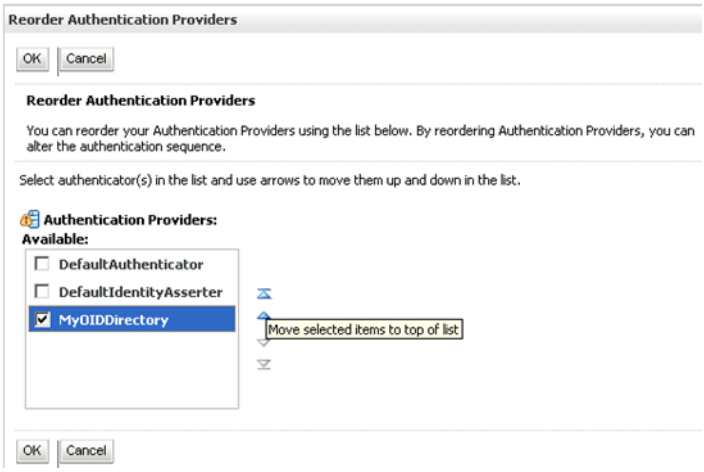
10. Click **Save**.
11. Perform the following steps to set up the DefaultAuthenticator **Control Flag** setting:
    a. At the main **Settings for myrealm** page, display the **Providers** tab, then display the **Authentication** sub-tab, then select **DefaultAuthenticator** to display its configuration page.
    b. Display the **Configuration\Common** tab and select 'SUFFICIENT' from the **Control Flag** list.

       For more information, see Section 3.4.6, "Setting the JAAS Control Flag Option".
    c. Click **Save**.
12. Perform the following steps to reorder Providers:
    a. At the main **Settings for myrealm** page, display the **Providers** tab, then display the **Authentication** sub-tab.



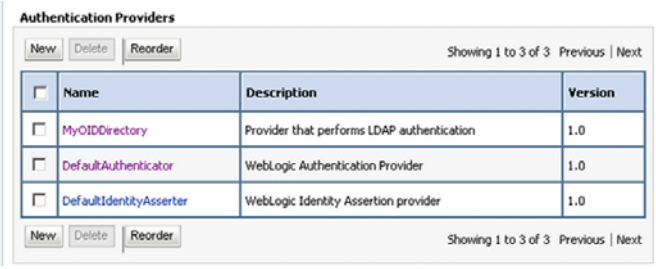
    b. Click **Reorder** to display the **Reorder Authentication Providers** page

c. Select MyOIDDirectory and use the arrow buttons to move it into the first position in the list, then click **OK**.



d. Click **OK** to save your changes.

The authentication providers are displayed in the re-ordered sequence.



13. Click **Save**.
14. In the Change Center, click **Activate Changes**.
15. Restart Oracle WebLogic Server.

3.4.2 Configuring Active Directory as the Authentication Provider

This procedure illustrates how to configure your Oracle Business Intelligence installation to use Active Directory.

The example data in this section uses a fictional company called XYZ Corporation that wants to set up WNA SSO for Oracle Business Intelligence for their internal users.

This example uses the following information:

- Active Directory domain

  The XYZ Corporation has an Active Directory domain, called xyzcorp.com, which authenticates all the internal users. When users log in to the corporate network from Windows computers, the log in to the Active Directory domain. The domain controller is addc.xyzcor.cop, which controls the Active Directory domain.

- Oracle BI EE WebLogic domain

  The XYZ Corporation has a WebLogic domain called bifoundation_domain (default name) installed on a network server domain called bieesvr1.xyz2.com.

- System Administrator and Test user

The following system administrator and domain user test the configuration:

- System Administrator user

  Jo Smith (login=jsmith, hostname=xyz1.xyzcorp.com)

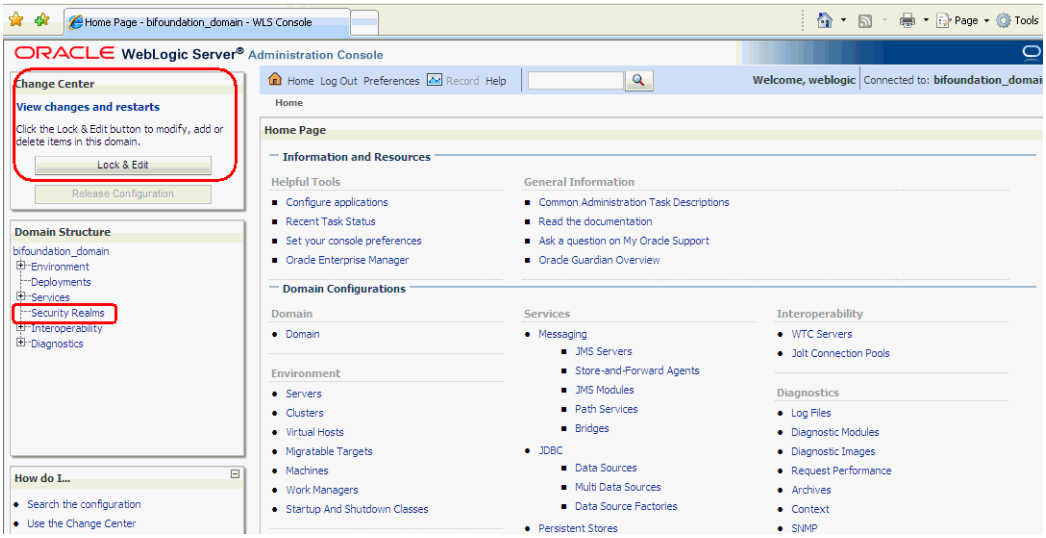
- Domain user

  Bob Jones (login=bjones hostname=xyz47.xyzcorp.com)

To configure Active Directory as the Authentication Provider:

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

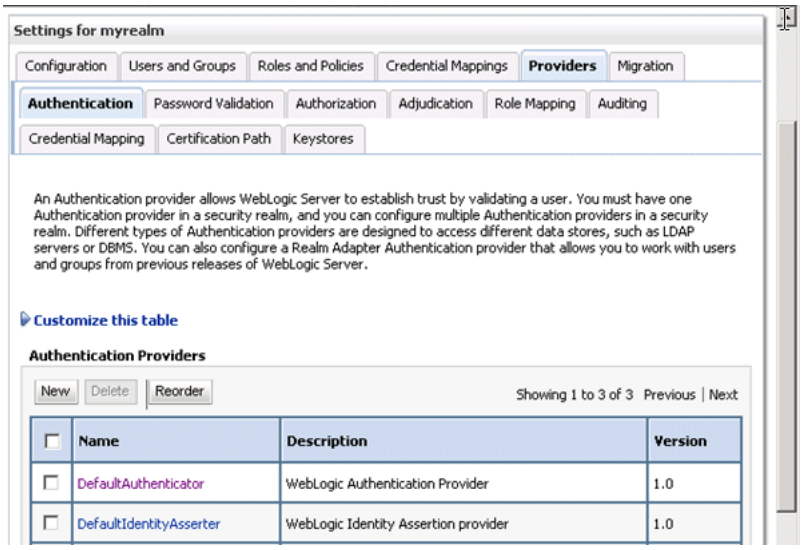
   For more information, see Section 1.6.1, "Using Oracle WebLogic Server Administration Console".



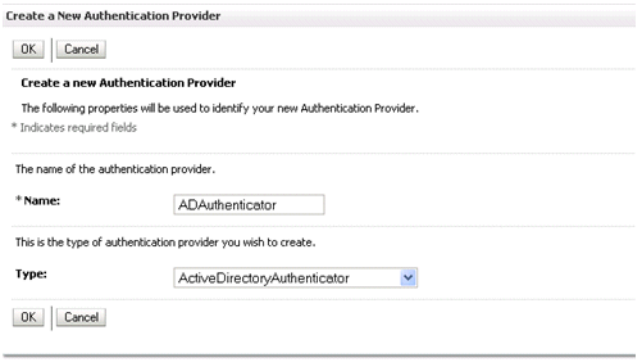
2. Select **Security Realms** from the left pane and click **myrealm**.

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

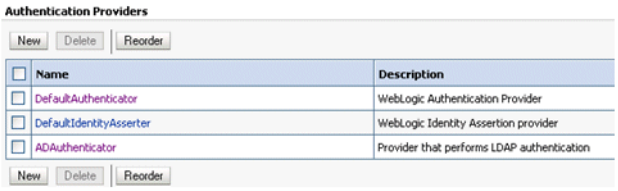
3. Display the **Providers** tab, then display the **Authentication** sub-tab.



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

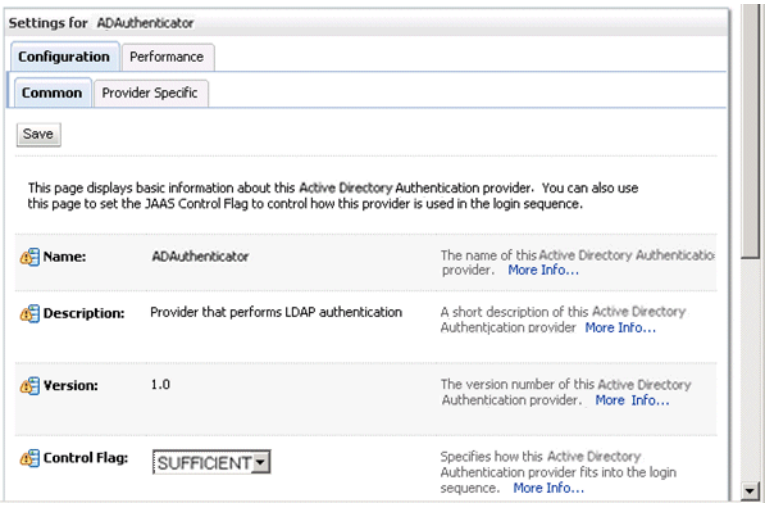


5. Enter values in the **Create a New Authentication Provider** page as follows:
   - **Name**: Enter a name for the authentication provider. For example, ADAuthenticator.
   - **Type**: Select ActiveDirectoryAuthenticator from the list.
   - Click **OK** to save the changes and display the authentication providers list updated with the new authentication provider.



6. Click DefaultAuthenticator in the **Name** column to display the Settings page.
7. In the Common Authentication Provider Settings page, change the **Control Flag** from REQUIRED to SUFFICIENT and click **Save**.

   For more information, see Section 3.4.6, "Setting the JAAS Control Flag Option".
8. In the authentication providers table, click ADDirectory in the **Name** column to display the Settings page.
9. Display the **Configuration\Common** tab, and use the **Control Flag** list to select 'SUFFICIENT', then click **Save**.



10. Display the **Provider Specific** tab to access the options which apply specifically to connecting to an Active Directory LDAP authentication store.
11. Use the Provider Specific tab to specify the following details:

Section Name	Field Name	Description
Connection	Host	The name of the Active Directory server addc.xyzcorp.com.
Connection	Port	The port number on which the Active Directory server is listening (389).
Connection	Principal	The LDAP DN for the user that connects to Active Directory when retrieving information about LDAP users. For example: cn=jsmith,cn=users,dc=us,dc=xyzcorp,dc=com.
Connection	Credential/Confirm Credential	Password for the specified Principal (for example welcome1).
Groups	Group Base DN	The LDAP query used to find groups in AD. **Note:** Only groups defined under this path will be visible to WebLogic. (CN=Builtin,DC=xyzcorp,DC=com).
Users	User Base DN	The LDAP query used to find users in AD. CN=Users,DC=xyzcorp,DC=com
Users	User Name Attribute	Attribute used to specify user name in AD. Default value is cn. Do not change this value unless you know your Active Directory is configured to use a different attribute for user name. If you do change it, see, Section 3.5.1, "Configuring the User Name Attribute In the Identity Store".
Users	All Users Filter	LDAP search filter. Click More Info... for details.
Users	User From Name Filter	LDAP search filter. Click More Info... for details.
Users	User Object class	The name of the user.
General	GUID attribute	The attribute used to define object GUIDs in AD. objectguid **Note**: You should not normally change this default value, however, if you do, you must also specify the changed value in Fusion Middleware Control, as described in the task Section 3.6, "Configuring the GUID Attribute In the Identity Store".

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

12. Click **Save**.
13. At the main **Settings for myrealm** page, display the **Providers** tab, then display the **Authentication** sub-tab.
14. Click **Reorder** to display the Reorder Authentication Providers page.
15. Select ADDirectory and use the arrow buttons to move it into the first position in the list, then click OK.
16. In the Change Center, click **Activate Changes**.
17. Restart Oracle WebLogic Server.

3.4.3 Configuring a Database as the Authentication Provider

This section describes how to configure Oracle Business Intelligence to use a database as the authentication provider by using a SQLAuthenticator and a virtualized identity store database adapter, and contains the following topics:

- Section 3.4.3.1, "Introduction and Prerequisites"
- Section 3.4.3.2, "Creating a Sample Schema for Users and Groups"
- Section 3.4.3.3, "Configuring a Data Source and SQL Authenticator Using the Oracle WebLogic Server Administration Console"
- Section 3.4.3.4, "Configuring the Virtualized Identity Store"
- Section 3.4.3.5, "Troubleshooting the SQL Authenticator"
- Section 3.4.3.6, "Correcting Database Adapter Errors Deleting and Recreating the Adapter"

3.4.3.1 Introduction and Prerequisites

You can configure more than one identity store to enable user role and profile information to be split across different identity stores (for example, LDAP and database identity stores) using virtualization.

User role and profile information can be stored in a database with the help of an adapter that enables the database to appear like an LDAP server. A virtualized identity store provider can retrieve user profile information from a database through a database adapter.

The method of database authentication described here is only possible in Release 11.1.1.5 (or higher), because earlier releases require the use of initialization blocks.

This topic explains how to configure Oracle Business Intelligence with a SQLAuthenticator and a virtualized identity store provider (including a database adapter), both running against a suitable database schema. The examples given are illustrative only, and your database schema need not be identical to the sample described here.

Use this procedure when you need to authenticate users against a database schema. The preferred identity store for authentication purposes is an LDAP directory service, such as Oracle Internet Directory (OID LDAP).

The approach to database authentication described here requires two database columns, one containing users and another containing passwords. This method is not based on database user accounts.

Oracle Business Intelligence Enterprise Edition Release 11.1.1.5.0 (or higher) must be installed and running.

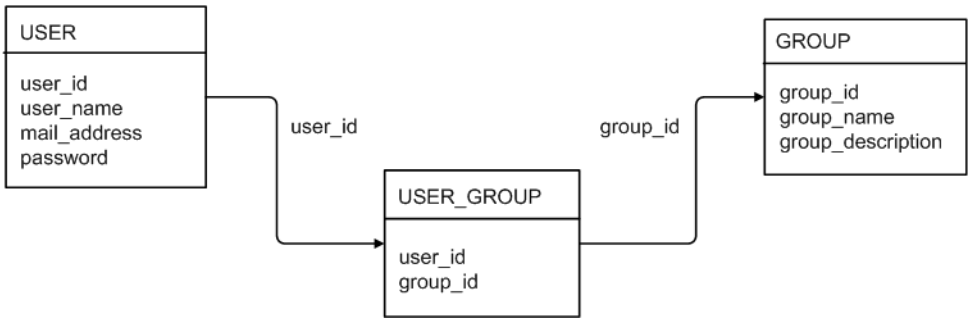
3.4.3.2 Creating a Sample Schema for Users and Groups

In practice, you will have your own schemas, which you are using in an earlier installation of Oracle BI EE. The sample schema described here is deliberately simplistic, and is intended only to illustrate how to configure the system to use the schema.

> **Note:** A suitable database schema containing the users, credentials and groups required for authentication, must be accessible from the WebLogic Server on which Oracle BI EE is running.

Figure 3–2 has tables, USER, GROUP and USER_GROUP, where USER_GROUP serves to join the other two tables.

Figure 3–2 Sample Schema for Users and Groups



If either USER, USER_GROUP or GROUP information exist in more than one table, you must create a view over the tables of each type of information. You can then present the views to the database adapter, configured in Section 3.4.3.3.2.

3.4.3.3 Configuring a Data Source and SQL Authenticator Using the Oracle WebLogic Server Administration Console

You configure a data source and SQL authenticator using the Oracle WebLogic Server Administration Console as follows:

- Section 3.4.3.3.1, "Configuring a Data Source Using the Oracle WebLogic Server Administration Console"
- Section 3.4.3.3.2, "Configuring a SQL Authenticator Using the Oracle WebLogic Server Administration Console"

3.4.3.3.1 Configuring a Data Source Using the Oracle WebLogic Server Administration Console

To configure a data source using the Oracle WebLogic Server Administration Console:

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

   For more information, see Section 1.6.1, "Using Oracle WebLogic Server Administration Console".

2. Click **Services** in the left pane and click **Data Sources**.
3. In the Summary of Data Sources page, click **New**, and select **Generic Data Source**.

4. In the JDBC Data Sources Properties page, enter or select values for the following properties:
   - **Name** - For example, enter: UserGroupDS

     The name used in the underlying configuration file (config.xml) and throughout the Administration Console whenever referring to this data source.
   - **JNDI Name** - For example, enter: `jdbc/UserGroupDS`

     The JNDI path to which this JDBC data source will be bound.
   - **Database Type** - For example, select: Oracle

     The DBMS of the database that you want to connect to.
5. Click **Next**.
6. Select a database driver from the **Database Driver** drop down list.

   For example, select: Oracle's Driver (Thin) for Service Connections; Versions:9.0.1 and later
7. Click **Next**.
8. Click **Next**.
9. On the Connection Properties page, enter values for the following properties:
   - **Database Name** - For example, enter: `ora11g`

     The name of the database that you want to connect to.
   - **Host Name** - For example, enter: `mymachine.mycompany.com`

     The DNS name or IP address of the server that hosts the database.
   - **Port** - For example, enter: `1521`

     The port on which the database server listens for connections requests.
   - **Database User Name**

     Typically the schema owner of the tables defined in Section 3.4.3.2
   - **Password/Confirm Password**

     The password for the **Database User Name**.
10. Click **Next**.
11. Check the details on the page are correct, and click **Test Configuration**.
12. Click **Next**.
13. In the Select Targets page select the servers or clusters for deploying the data source.

    You should select the Administration Server and Managed server as your targets, for example:
    - In the Servers pane

      Select the **AdminServer** check box.
    - In the Clusters pane

      Select the **bi_server1** option.
14. Click **Finish**.

15. In the Change Center, click **Activate Changes**.
16. Restart Oracle WebLogic Server.

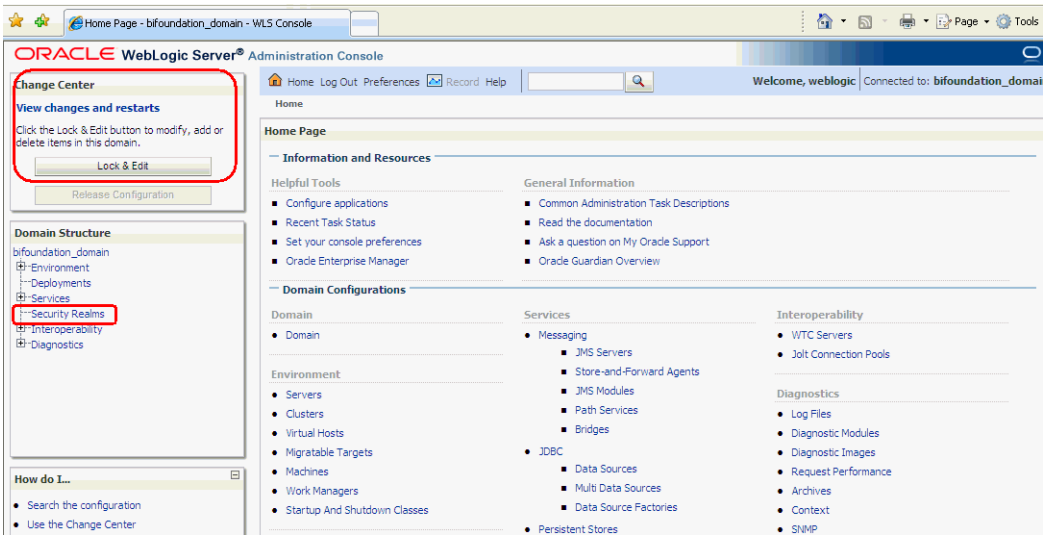
3.4.3.3.2 Configuring a SQL Authenticator Using the Oracle WebLogic Server Administration Console

This task enables a suitably privileged user to log in to the Oracle WebLogic Server Administration Console using the WebLogic database authenticator.

To configure a SQL authenticator using the Oracle WebLogic Server Administration Console:

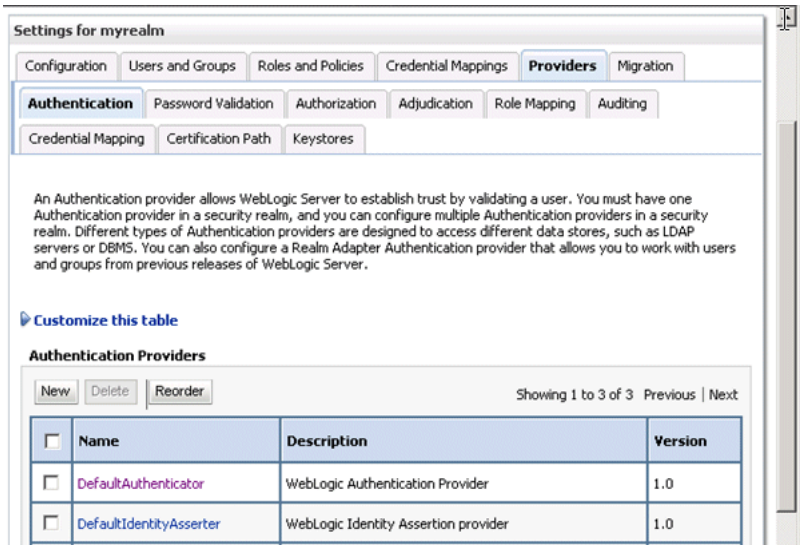
1. Log in to Oracle WebLogic Server Administration Console, and click **Lock & Edit** in the Change Center.

   For more information, see Section 1.6.1, "Using Oracle WebLogic Server Administration Console".

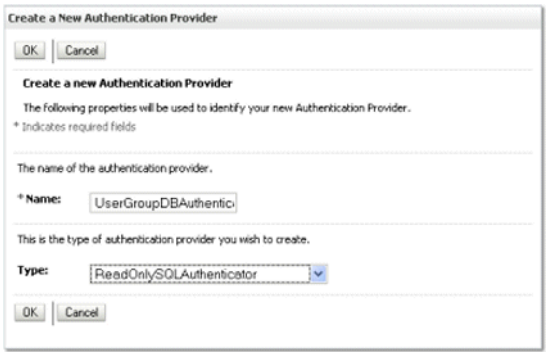


2. Select **Security Realms** from the left pane and click **myrealm**.

   The default Security Realm is named **myrealm**.
3. Display the **Providers** tab, then display the **Authentication** sub-tab.



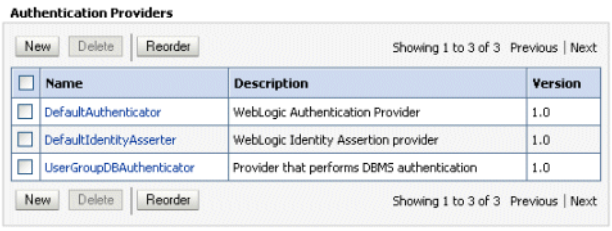
4. Click **New** to launch the **Create a New Authentication Provider** page.



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

   - **Name**: Enter a name for the authentication provider. For example, UserGroupDBAuthenticator.
   - **Type**: Select ReadOnlySQLAuthenticator from the list.

     This creates a read-only SQL Authenticator, and WebLogic does not write back to the database.
   - Click **OK** to save the changes and display the authentication providers list updated with the new authentication provider.



6. In the authentication providers table, click UserGroupDBAuthenticator in the **Name** column to display the Settings page.
7. Display the **Provider Specific** tab, and enter in the **Data Source Name** field, the name of the data source that you created in Section 3.4.3.3.1.

For example, `UserGroupDS`.

8. In the **Provider Specific** tab you specify the SQL statements used to query, and authenticate against, your database tables.

   Table 3–1 shows SQL statements for the sample schema outlined in Section 3.4.3.2

Table 3–1 SQL Statements for the Sample Schema

Query	SQL	Notes
SQL Get Users Password (used to authenticate)	SELECT PASSWORD FROM USER WHERE USER_ID = ?	The SQL statement used to look up a user's password. The SQL statement requires a single parameter for the username and must return a resultSet containing at most a single record containing the password.
SQL User Exists	SELECT USER_ID FROM USER WHERE USER_ID = ?	The SQL statement used to look up a user. The SQL statement requires a single parameter for the username and must return a resultSet containing at most a single record containing the user.
SQL List Users	SELECT USER_ID FROM USER WHERE USER_ID LIKE ?	The SQL statement used to retrieve users that match a particular wildcard search. The SQL statement requires a single parameter for the usernames and returns a resultSet containing matching usernames.
SQL List Groups	SELECT GROUP_ID FROM GROUP WHERE GROUP_ID LIKE ?	The SQL statement used to retrieve group names that match a wildcard. The SQL statement requires a single parameter for the group name and returns a resultSet containing matching groups.
SQL Group Exists	SELECT GROUP_ID FROM GROUP WHERE GROUP_ID = ?	The SQL statement used to look up a group. The SQL statement requires a single parameter for the group name and must return a resultSet containing at most a single record containing the group.
SQL Is Member	SELECT GROUP_ID FROM USER_GROUP WHERE GROUP_ID=? AND USER_ID LIKE ?	The SQL statement used to look up members of a group. The SQL statement requires two parameters: a group name and a member or group name. It must return a resultSet.
SQL List Member Groups	SELECT GROUP_ID FROM USER_GROUP WHERE USER_ID = ?	The SQL statement used to look up the groups a user or group is a member of. The SQL statement requires a single parameter for the username or group name and returns a resultSet containing the names of the groups that matched.
SQL Get User Description (if description supported enabled)	SELECT USER_NAME FROM USER WHERE USER_ID = ?	The SQL statement used to retrieve the description of a specific user. The SQL statement requires a single parameter for the username and must return a resultSet containing at most a single record containing the user description.

Table 3–1 (Cont.) SQL Statements for the Sample Schema

Query	SQL	Notes
SQL Get Group Description (if description supported enabled)	SELECT GROUP_DESCRIPTION FROM GROUP WHERE GROUP_ID = ?	The SQL statement used to retrieve the description of a group. It is valid only if Descriptions Supported is enabled. The SQL statement requires a single parameter for the group name and must return a resultSet containing at most a single record containing the group description.

Note: If you are using a different table structure, you might need to adapt these SQL statements (table or column names) to your own schema. Also, you should leave the question mark (?) as a runtime query placeholder (rather than hardcode a user or group name).

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

9. Enter all of the SQL statements appropriate to your Authenticator.
10. If your password column is in plain text (that is, if the result of the query supplied for the **SQL Get Users Password** column is not hashed or encrypted), select the **Plaintext Password Enabled** checkbox.

    If the **Plaintext Password Enabled** checkbox is cleared, the SQLAuthenticator expects passwords to have been hashed using SHA-1 (default encryption algorithm). For more information on the supported encryption algorithms, see the documentation for the base SQLAuthenticator Mbean PasswordAlgorithm attribute.
11. Click **Save**.
12. Perform the following steps to configure default authenticator **Control Flag** setting:
    a. At the main **Settings for myrealm** page, display the **Providers** tab, then display the **Authentication** sub-tab, then select **DefaultAuthenticator** to display its configuration page.
    b. Display the **Configuration\Common** tab and select 'SUFFICIENT' from the **Control Flag** list.

       For more information, see Section 3.4.6, "Setting the JAAS Control Flag Option".
    c. Click **Save**.
13. Perform the following steps to reorder the Authentication Providers:
    a. Display the **Providers** tab.
    b. Click **Reorder** to display the **Reorder Authentication Providers** page
    c. Select UserGroupDBAuthenticator and use the arrow buttons to move it into the first position in the list.
    d. Click **OK** to save your changes.

14. You must ensure there is a trusted system user in your database and that you replace the credentials in the Credential store to point to this user's credentials as described in Section 3.7, "Configuring a New Trusted User (BISystemUser)".
15. In the Change Center, click **Activate Changes**.
16. Restart the Oracle BI components (use Fusion Middleware Control once the Administration Server has been restarted), Oracle WebLogic Server, and Managed servers.

> **Note:** Check the **Users and Groups** tab to confirm that the database users and groups appear there.

3.4.3.4 Configuring the Virtualized Identity Store

Configure the virtualized identity store as follows:

- Section 3.4.3.4.1, "Enabling Virtualization by Configuring the Identity Store"
- Section 3.4.3.4.2, "Configuring a Database Adaptor"

3.4.3.4.1 Enabling Virtualization by Configuring the Identity Store You must configure the identity store to enable virtualization so that more than one Identity Store can be used with the identity store service, and therefore user profile information can be split across different authentication providers (identity stores).

For more information, see Section 3.4.5, "Configuring Multiple Authentication Providers Using Fusion Middleware Control".

3.4.3.4.2 Configuring a Database Adaptor You configure a database adaptor to make the database appear like an LDAP server, which enables the virtualized identity store provider to retrieve user profile information from a database using the database adapter.

To configure a database adaptor:

This task shows how to edit and apply adapter templates that specify how to use your database tables as an identity store.

1. Create a file named adapter_template_usergroup1.xml.

   This file describes the mapping of the user table to a virtual LDAP store.

2. Make sure that the file contains the following contents:

> **Note:** You must adapt the section shown in bold, to match the columns in your own table with attributes in the LDAP server. The example given here is for the sample schema that is used throughout Section 3.4.3.

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<adapters schvers="303" version="1"
xmlns="http://www.octetstring.com/schemas/Adapters"
xmlns:adapters="http://www.w3.org/2001/XMLSchema-instance">
   <dataBase id="directoryType" version="0">
      <root>%ROOT%</root>
      <active>true</active>
      <serverType>directoryType</serverType>
      <routing>
```

```
         <critical>true</critical>
         <priority>50</priority>
         <inclusionFilter/>
         <exclusionFilter/>
         <plugin/>
         <retrieve/>
         <store/>
         <visible>Yes</visible>
         <levels>-1</levels>
         <bind>true</bind>
         <bind-adapters/>
         <views/>
         <dnpattern/>
      </routing>
      <pluginChains
xmlns="http://xmlns.oracle.com/iam/management/ovd/config/plugins">
         <plugins>
            <plugin>
               <name>DBGUID</name>

<class>oracle.ods.virtualization.engine.chain.plugins.dbguid.DBGuidPlugin</clas
s>
               <initParams>
                  <param name="guidAttribute" value="orclguid"/>
               </initParams>
            </plugin>
         </plugins>
         <default>
            <plugin name="DBGUID"/>
         </default>
         <add/>
         <bind/>
         <delete/>
         <get/>
         <modify/>
         <rename/>
      </pluginChains>
      <driver>oracle.jdbc.driver.OracleDriver</driver>
      <url>%URL%</url>
      <user>%USER%</user>
      <password>%PASSWORD%</password>
      <ignoreObjectClassOnModify>false</ignoreObjectClassOnModify>
      <includeInheritedObjectClasses>true</includeInheritedObjectClasses>
      <maxConnections>10</maxConnections>
      <mapping>
         <joins/>
         <objectClass name="inetorgperson" rdn="cn">
            <attribute ldap="cn" table="USER" field="USER_NAME" type=""/>
            <attribute ldap="uid" table="USER" field="USER_ID" type=""/>
            <attribute ldap="usernameattr" table="USER" field="USER_NAME"
type=""/>
            <attribute ldap="loginid" table="USER" field="USER_ID" type=""/>
            <attribute ldap="description" table="USER" field="USER_NAME"
type=""/>
            <attribute ldap="orclguid" table="USER" field="USER_ID" type=""/>
         </objectClass>
      </mapping>
      <useCaseInsensitiveSearch>true</useCaseInsensitiveSearch>
      <connectionWaitTimeout>10</connectionWaitTimeout>
      <oracleNetConnectTimeout>0</oracleNetConnectTimeout>
```

```
        <validateConnection>false</validateConnection>
    </dataBase>
</adapters>
```

In this example the section highlighted in bold should be the only section that needs customizing, but the elements should be mapped by matching the attributes/classes used in a virtual LDAP schema with the columns in your database which correspond to them. The virtual schema is the same as that of Weblogic Embedded LDAP, so you can map database columns to any of the attributes shown in Table 3–2.

Table 3–2 Examples of Attributes to Map to Database Columns

Attribute	Example
description	John Doe
cn	john.doe
uid	john.doe
sn	Doe
userpassword	welcome1
displayName	John Doe
employeeNumber	12345
employeeType	Regular
givenName	John
homePhone	650-555-1212
mail	john.doe@example.com
title	Manager
manager	uid=mary.jones,ou=people,ou=myrealm,dc=wc_domain
preferredLanguage	en
departmentNumber	tools
facsimiletelephonenumber	650-555-1200
mobile	650-500-1200
pager	650-400-1200
telephoneNumber	650-506-1212
postaladdress	200 Oracle Parkway
l	Redwood Shores
homepostaladdress	123 Main St., Anytown 12345

3. Use the first, outer element (**<objectClass name="inetorgperson" rdn="cn">**) to declare mapping of the LDAP objectclass inetorgperson.

   The cn attribute is used as its RDN (Relative Distinguished Name). The sub-elements then declare which LDAP attributes map to which tables and columns in the database. For example, the line **<attribute ldap="uid" table="USER" field="USER_ID" type=""/>** maps the USER_ID field of the USER table to the standard LDAP attribute uid (that is, a unique user id for each user).

   Next, you map groups using the same method.

4. Create a file named adapter_template_usergroup2.xml.

   This file describes the mapping of the group table to a virtual LDAP store.

5. Add the following contents to the file:

   You must customize the section shown in bold to match the columns in your own table. The sample content shown here is to match the sample schema that is used throughout this example.

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<adapters schvers="303" version="1"
xmlns="http://www.octetstring.com/schemas/Adapters"
xmlns:adapters="http://www.w3.org/2001/XMLSchema-instance">
    <dataBase id="directoryType" version="0">
      <root>%ROOT%</root>
      <active>true</active>
      <serverType>directoryType</serverType>
      <routing>
         <critical>true</critical>
         <priority>50</priority>
         <inclusionFilter/>
         <exclusionFilter/>
         <plugin/>
         <retrieve/>
         <store/>
         <visible>Yes</visible>
         <levels>-1</levels>
         <bind>true</bind>
         <bind-adapters/>
         <views/>
         <dnpattern/>
      </routing>
      <pluginChains
xmlns="http://xmlns.oracle.com/iam/management/ovd/config/plugins">
         <plugins>
            <plugin>
               <name>VirtualAttribute</name>

<class>oracle.ods.virtualization.engine.chain.plugins.virtualattr.VirtualAttrib
utePlugin</class>
               <initParams>
                  <param name="ReplaceAttribute"
value="uniquemember={cn=%uniquemember%,cn=users,dc=oracle,dc=com}"/>
               </initParams>
            </plugin>
         </plugins>
         <default>
            <plugin name="VirtualAttribute"/>
         </default>
         <add/>
         <bind/>
         <delete/>
         <get/>
         <modify/>
         <rename/>
      </pluginChains>
      <driver>oracle.jdbc.driver.OracleDriver</driver>
      <url>%URL%</url>
      <user>%USER%</user>
      <password>%PASSWORD%</password>
      <ignoreObjectClassOnModify>false</ignoreObjectClassOnModify>
```

```
      <includeInheritedObjectClasses>true</includeInheritedObjectClasses>
      <maxConnections>10</maxConnections>
      <mapping>
         <joins/>
         <objectClass name="groupofuniquenames" rdn="cn">
            <attribute ldap="cn" table="USER_GROUP" field="GROUP_ID" type=""/>
            <attribute ldap="description" table="USER_GROUP" field="GROUP_ID"
type=""/>
            <attribute ldap="uniquemember" table="USER_GROUP" field="USER_ID"
type=""/>
            <attribute ldap="orclguid" table="USER_GROUP" field="USER_ID"
type=""/>
         </objectClass>
      </mapping>
      <useCaseInsensitiveSearch>true</useCaseInsensitiveSearch>
      <connectionWaitTimeout>10</connectionWaitTimeout>
      <oracleNetConnectTimeout>0</oracleNetConnectTimeout>
      <validateConnection>false</validateConnection>
   </dataBase>
</adapters>
```

6. Customize appropriate sections highlighted in bold, for the following elements:
   - ReplaceAttribute

     Specifies how to define the unique member for a group (the %uniquemember% is a placeholder for a value which will be passed in at runtime when looking up whether a user is a member of a group)

     The only aspect of this element you may want to change is the specification of the root for your users. While this is notional, by default it must match whatever you specify as the root of your user population when you run the libovdadapterconfig script in Step 10.
   - groupofuniquenames

     Specifies how group attributes are mapped to database fields and as with the user, the attributes correspond to the defaults in Weblogic Embedded LDAP.

     You must map the following attributes:
     - **cn** (map to a unique name for your group)
     - **uniquemember** (map to the unique name for your user in the user/group mapping table in your database schema)

     Mapping the following attributes is optional:
     - **description** is optional (although clearly helpful)
     - **orclguid** (maps to a UID, if available in your database schema)

     No other attributes are user-configurable.
7. Copy the two adapter files into the following folder:

   <MW_HOME>/oracle_common/modules/oracle.ovd_11.1.1/templates/
8. Open a command prompt/terminal at:

   <MW_HOME>/oracle_common/bin
9. Ensure the following environment variables are set:

- ORACLE_HOME=<*MW_HOME*>/Oracle_BI1
- WL_HOME=<*MW_HOME*>/wlserver_10.3/
- JAVA_HOME=<*MW_HOME*>/jdk160_24/

10. Run the libovdadapterconfig script to create each of the two adapters from the template files. The syntax is:

```
libovdadapterconfig -adapterName <name of adapter> -adapterTemplate <name (NOT
including path) of template file which defines adapter> -host localhost -port
<Admin Server port> -userName <user id of account which has administrative
privileges in the domain> -domainPath <path to the BI domain> -dataStore DB
-root <nominal specification of a pseudo-LDAP query to treat as the "root" of
this adapter - must match that specified in template for adapter 2 above>
-contextName default -dataSourceJNDIName <JNDI name for DataSource which points
at the database being mapped>
```

For example:

```
./libovdadapterconfig.sh -adapterName userGroupAdapter1 -adapterTemplate
adapter_template_usergroup1.xml -host localhost -port 7001 -userName weblogic
-domainPath /opt/oracle_bi/user_projects/domains/bifoundation_domain/
-dataStore DB -root cn=users,dc=oracle,dc=com -contextName default
-dataSourceJNDIName jdbcUserGroupsDS

./libovdadapterconfig.sh -adapterName userGroupAdapter2 -adapterTemplate
adapter_template_usergroup2.xml -host localhost -port 7001 -userName weblogic
-domainPath /opt/oracle_bi/user_projects/domains/bifoundation_domain/
-dataStore DB -root cn=users,dc=oracle,dc=com -contextName default
-dataSourceJNDIName jdbcUserGroupsDS
```

The scripts should exit without error.

11. Restart WebLogic Administration Server and Managed servers.

You should now be able to log in to WebLogic and Oracle Business Intelligence using credentials stored in the database

3.4.3.5 Troubleshooting the SQL Authenticator

This section provides troubleshooting information on the SQL authenticator, and contains the following topics:

- Section 3.4.3.5.1, "Adding a User to the Global Admin Role Using the Oracle WebLogic Server Administration Console"
- Section 3.4.3.5.2, "An Incorrect Data Source Name is Specified for the SQLAuthenticator"
- Section 3.4.3.5.3, "Incorrect SQL Queries"

3.4.3.5.1 Adding a User to the Global Admin Role Using the Oracle WebLogic Server Administration Console

If you cannot log in to Oracle Business Intelligence using a database user, a useful diagnostic test is to see whether your user can log in to WebLogic at all. If you do not have other applications on the WebLogic Server which take advantage of WebLogic container authentication, you can add your user (temporarily) to the WebLogic Global Admin role and see if the user can log in to the Oracle WebLogic Server Administration Console to test whether the SQLAuthenticator is working at all.

To add a user to the global admin role using the Oracle WebLogic Server Administration Console:

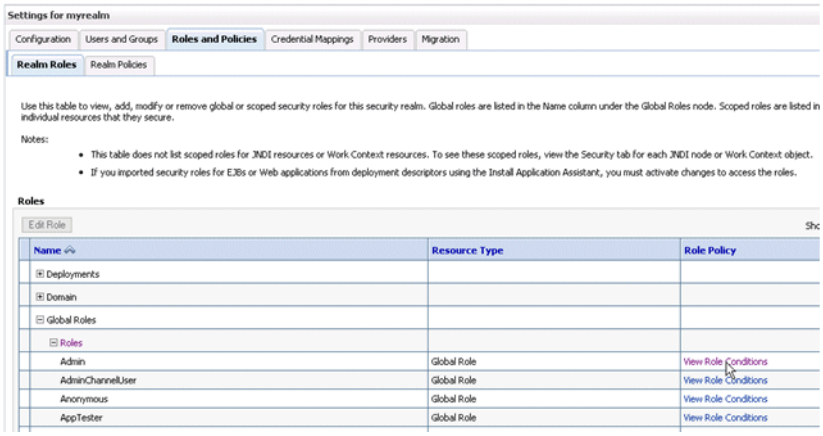
1. Log in to Oracle WebLogic Server Administration Console, and click **Lock & Edit** in the Change Center.

   For more information, see Section 1.6.1, "Using Oracle WebLogic Server Administration Console".

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

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

3. Display the **Roles and Policies** tab, then display the **Realm Roles** sub-tab.



4. In the list of roles, click on the plus sign to expand **Global Roles**, then **Roles**, then click the **View Role Conditions** link for the Admin role.

5. Ensure the conditions specified will match your user, either directly, or by belonging to a group.

   For example, a condition may be User=myadminaccount or Group=Administrators.

6. If you have made any changes, click **Save**.

   Changes are applied immediately.

7. You should now be able to check whether the user in question can log in to the Oracle WebLogic Server Administration Console at http://*<bi server address>*:*<AdminServer Port>*/console (for example, http://mybiserver:7001/console).

   If the user can log in to the console, but cannot log in to Oracle Business Intelligence, the SQLAuthenticator is working correctly, but there may be issues in the identity store service. Check that you have specified the virtualize=true property in Section 3.4.3.4, "Configuring the Virtualized Identity Store" and that your DBAdapter templates are correct.

3.4.3.5.2 An Incorrect Data Source Name is Specified for the SQLAuthenticator If you specify the wrong JNDI name for the data source field of the SQLAuthenticator, then errors such as the following are included in the log files for Administration Server and Managed Servers:

```
Caused by: javax.security.auth.login.FailedLoginException:
[Security:090761]Authentication failed for user jsmith java.sql.SQLException:
[Security:090788]"Problem with DataSource/ConnectionPool configuration, verify
DataSource name wrongdsname is correct and Pool configurations are correct"
        at weblogic.security.providers.authentication.shared.DBMSAtnLoginModuleI
mpl.login(DBMSAtnLoginModuleImpl.java:318)
```

Use the fully qualified JNDI name, not the **Name** field of the data source, so in the example shown in Section 3.4.3.3.1, "Configuring a Data Source Using the Oracle WebLogic Server Administration Console", the name of the data source was UserGroupDS but the JNDI name was jdbc/UserGroupDS. Use the fuller form.

3.4.3.5.3 Incorrect SQL Queries Ensure that the SQL queries that you specify when configuring the SQLAuthenticator are syntactically correct and refer to the correct tables. For example, the following error occurs in the Administration Server.log file when the wrong table name is specified for the password query:

```
####<Jul 7, 2011 4:03:27 PM BST> <Error> <Security> <gbr20020> <AdminServer>
<[ACTIVE] ExecuteThread: '8' for queue: 'weblogic.kernel.Default (self-tuning)'>
<<WLS Kernel>> <> <de7dd0dc53f3d0ed:e0ce69e:131007c1afe:-8000-00000000000007fa>
<1310051007798> <BEA-000000> <[Security:090759]A SQLException occurred while
retrieving password information
java.sql.SQLSyntaxErrorException: ORA-00942: table or view does not exist
     at oracle.jdbc.driver.T4CTTIoer.processError(T4CTTIoer.java:457)
     at oracle.jdbc.driver.T4CTTIoer.processError(T4CTTIoer.java:405)
     at oracle.jdbc.driver.T4C8Oall.processError(T4C8Oall.java:889)
     at oracle.jdbc.driver.T4CTTIfun.receive(T4CTTIfun.java:476)
```

3.4.3.6 Correcting Database Adapter Errors Deleting and Recreating the Adapter

You cannot modify an existing database adapter, so if you make an error in either the libovdadapter command, or the templates you use to create the adapters, you must delete then recreate the adapter using the following procedure.

To correct database adapter errors by deleting and recreating the adapter:

1. Log in to the WSLT console by running the WLST script.

   For example:

   MW_HOME/oracle_common/common/bin/wlst.sh (UNIX)

   MW_HOME\oracle_common\common\bin\wlst.cmd (Windows)

2. Connect to your Administration Server using the following syntax:

   connect ('<*WLS admin user name*>','<*WLS admin password*>','t3://<*admin server host*>:<*admin server port*>')

   For example:

   ```
   connect('weblogic','weblogic','t3://myserver:7001')
   ```

3. Delete the misconfigured adapter using the following syntax:

   deleteAdapter(adapterName='<*AdapterName*>')

   For example:

   ```
   deleteAdapter(adapterName='userGroupAdapter2')
   ```

4. Exit the WLST console using the command exit() and recreate the adapter with the correct settings by following the steps outlined in Section 3.4.3.4.2, "Configuring a Database Adaptor".

3.4.4 Configuring LDAP as the Authentication Provider and Storing Groups In a Database

This section describes how to configure Oracle Business Intelligence to authenticate against an LDAP Identity Store, and store group information in a database. The examples provided in this section use Oracle Internet Directory (OID LDAP), and a sample database schema. However, you do not have to use OID LDAP as your LDAP identity store and your database schema does not have to be identical to the sample provided.

Oracle Business Intelligence provides an authentication provider for WebLogic Server called BISQLGroupProvider that enables you to use this method. This authentication provider does not authenticate end user credentials but enables external group memberships held in a database table to contribute to an authenticated user's identity.

This section contains the following topics:

- Section 3.4.4.1, "Prerequisites"
- Section 3.4.4.2, "Creating a Sample Schema for Groups and Group Members"
- Section 3.4.4.3, "Configuring a Data Source and the BISQLGroupProvider Using Oracle WebLogic Server Administration Console"
- Section 3.4.4.4, "Configuring the Virtualized Identity Store"
- Section 3.4.4.5, "Testing the Configuration by Adding a Database Group to an Application Role"
- Section 3.4.4.6, "Correcting Errors In the Adaptors"

3.4.4.1 Prerequisites

The following prerequisites must be satisfied before you attempt to configure LDAP authentication as described in this section:

- Oracle Business Intelligence Enterprise Edition Release 11.1.1.5.0 (or higher) must be installed and running.
- You must apply all relevant patches to the Oracle BI 11.1.1.5.0 system.
- A suitable database schema containing at least one table with the required groups in it, and a mapping table which maps those groups to the names of users authenticated by LDAP must be running and accessible from the WebLogic Server on which Oracle BI EE is running.
- The configuration must include a supported LDAP server to use as the identity store that contains users.
- If you need Oracle Business Intelligence to deliver content to members of an application role the following restrictions apply:
  - You can only pair a single LDAP authenticator with a single BISQLGroupProvider.

    When you configure multiple LDAP authenticators and want to retrieve group membership from the BISQLGroupProvider, content cannot be delivered to all members of an application role. In this configuration Oracle BI Delivers cannot resolve application role membership based on users and group membership.
  - You cannot define the same group in more than one identity store.

For example, you cannot have a group called BIAdministrators in both LDAP and database groups table. If you do, the security code invoked by Oracle BI Delivers cannot resolve application role membership.

3.4.4.2 Creating a Sample Schema for Groups and Group Members

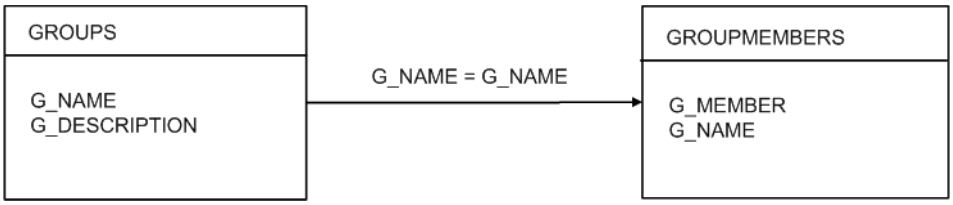
The sample schema described here is deliberately simplistic, and is intended only to illustrate how to configure Oracle Business Intelligence to use the schema.

The sample schema is called ACME_BI_GROUPS and contains two tables: GROUPS, which defines the list of external groups, and GROUPMEMBERS, which describes group membership for users that exist in your primary identity store.

The advantage of defining tables (or views) identical to Figure 3–3 is that the configuration of the BISQLGroupProvider can use the default SQL outlined in Table 3–3.

Figure 3–3 has the tables GROUPS and GROUPMEMBERS.

Figure 3–3 Sample Schema for Groups and Group Members



You must map the users in your LDAP store to Groups in your database table by login name. In Figure 3–3, the value of G_MEMBER in the GROUPMEMBERS table must match the value of the LDAP attribute used for login (for example, uid, cn or mail), as specified in the LDAP authenticator. For example, you should not map the database groups by uid if the login attribute is mail.

3.4.4.3 Configuring a Data Source and the BISQLGroupProvider Using Oracle WebLogic Server Administration Console

You configure a data source and the BISQLGroupProvider using Oracle WebLogic Server Administration Console as follows:

- Section 3.4.4.3.1, "Configuring Oracle Internet Directory as the Primary Identity Store for Authentication Using Oracle WebLogic Server"
- Section 3.4.4.3.2, "Installing the BISQLGroupProvider"
- Section 3.4.4.3.3, "Configuring the Data Source Using Oracle WebLogic Server Administration Console"
- Section 3.4.4.3.4, "Configuring the BISQLGroupProvider SQL Authenticator Using Oracle WebLogic Server Administration Console"

3.4.4.3.1 Configuring Oracle Internet Directory as the Primary Identity Store for Authentication Using Oracle WebLogic Server Follow the link to instructions that will enable you to configure WebLogic to authenticate your user population against OID LDAP.

For more information, see Section 3.4.1, "Configuring Oracle Internet Directory as the Authentication Provider".

Note: When following the steps of this task, make a note of the value of the **User Base DN** and **User Name Attribute** in the Provider Specific configuration page for your OID LDAP authenticator, which will be needed later. For more information, see Section 3.4.4.4.3, "Configuring a Database Adaptor to Retrieve Group Information".

3.4.4.3.2 Installing the BISQLGroupProvider

Before you can configure a BISQLGroupProvider authenticator, you must first install the JAR file BISecurityProviders.jar, which contains the authenticator. The file is available in the following location for both Oracle Business Intelligence Release 11.1.1.6.0 (or higher) without applying a patch, and Oracle Business Intelligence Release 11.1.1.5.0 after applying a patch (but the file must be copied to the correct location):

MW_HOME/*ORACLE_HOME*/bifoundation/security/providers

You must copy the file to the following location:

MW_HOME/wlserver_10.3/server/lib/mbeantypes

After copying the file into the specified location you must restart the Administration Server to enable the new provider to appear in the list of available authenticators.

Note: If you perform an Enterprise Install to create a clustered environment, then the installation cannot start the scaled-out Managed server because the BISecurityProviders.jar file is not available. When this situation occurs during installation, copy the Jar file to the correct location and click **Retry** in the installer.

3.4.4.3.3 Configuring the Data Source Using Oracle WebLogic Server Administration Console

To configure the data source using Oracle WebLogic Server Administration Console:

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

   For more information, see Section 1.6.1, "Using Oracle WebLogic Server Administration Console".

2. Click **Services** in the left pane and click **Data Sources**.
3. In the Summary of Data Sources page, click **New**, and select **Generic Data Source**.
4. In the JDBC Data Sources Properties page, enter or select values for the following properties:
   - **Name** - For example, enter: `BIDatabaseGroupDS`

     The name used in the underlying configuration file (config.xml) and throughout the Oracle WebLogic Server Administration Console whenever referring to this data source.
   - **JNDI Name** - For example, enter: `jdbc/BIDatabaseGroupDS`

     The JNDI path to which this JDBC data source will be bound.
   - **Database Type** - For example, select: Oracle

     The DBMS of the database that you want to connect to.
5. Click **Next**.

6. Select a database driver from the **Database Driver** drop down list.

   For example, select: Oracle's Driver (Thin) for Service Connections; Versions:9.0.1 and later.

   Note: If using an Oracle database, select 'Oracle's Driver (Thin) for Service Connections; Versions:9.0.1 and later'.

7. Click **Next**.
8. Click **Next**.
9. On the Connection Properties page, enter values for the following properties:
   - **Database Name** - For example, enter: `ora11g`

     The name of the database that you want to connect to.
   - **Host Name** - For example, enter: `mymachine.mycompany.com`

     The DNS name or IP address of the server that hosts the database.

     Note: Do not use localhost if you intend to use a cluster.

   - **Port** - For example, enter: `1521`

     The port on which the database server listens for connections requests.
   - **Database User Name**

     Typically the schema owner of the tables defined in Section 3.4.4.2.

     For example, enter `MYUSER`.
   - **Password/Confirm Password**

     The password for the **Database User Name**.

     For example, enter `mypassword`.
10. Click **Next**.
11. Check the details on the page are correct, and click **Test Configuration**.
12. Click **Next**.
13. In the Select Targets page select the servers or clusters for your data source to be deployed to.

    You should select the Administration Server and Managed Servers as your targets, for example:
    - In the Servers pane

      Select the **AdminServer** option.
    - In the Clusters pane

      Select the **bi_server1** check box to deploy to the cluster (this does not apply to a Simple Install).
14. Click **Finish**.
15. In the Change Center, click **Activate Changes**.

Note: In this example, the data source is called BIDatabaseGroupDS.

3.4.4.3.4 Configuring the BISQLGroupProvider SQL Authenticator Using Oracle WebLogic Server Administration Console

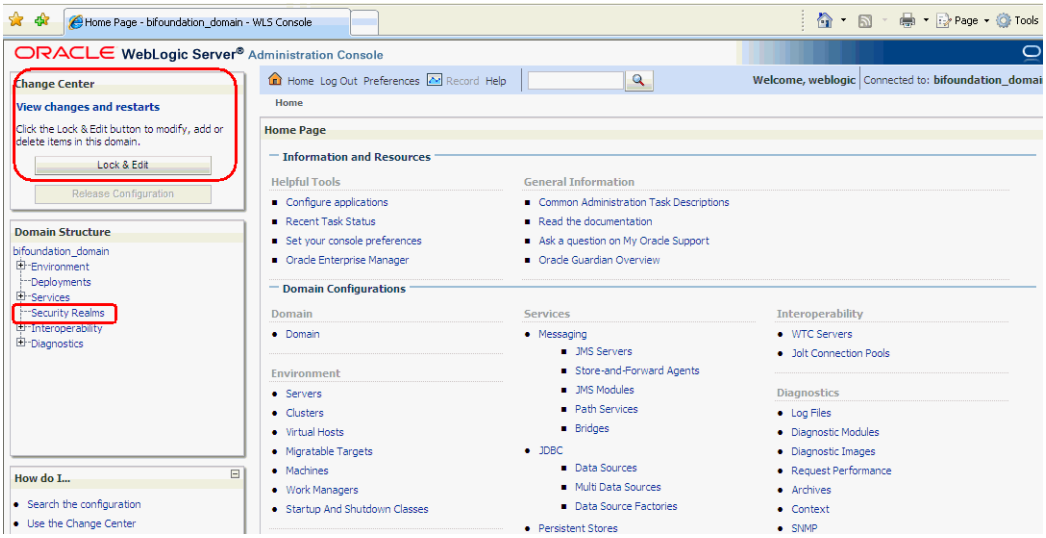
This task explains how to create a BISQLGroupProvider against the BIDatabaseGroupDS data source using the example table structure outlined in Section 3.4.4.2, "Creating a Sample Schema for Groups and Group Members". You may need to modify the SQL statements used (table or column names) if your structure differs from the example.

Note: There is no authentication against the database, as it just stores the groups to be associated with users. Authentication occurs against LDAP and the database is exposed when the BISQLGroupProvider assigns groups to application roles in Oracle WebLogic Server Administration Console.

To configure the BISQLGroupProvider SQL authenticator using Oracle WebLogic Server Administration Console:

1. Log in to Oracle WebLogic Server Administration Console as a WebLogic administrator, and click **Lock & Edit** in the Change Center.

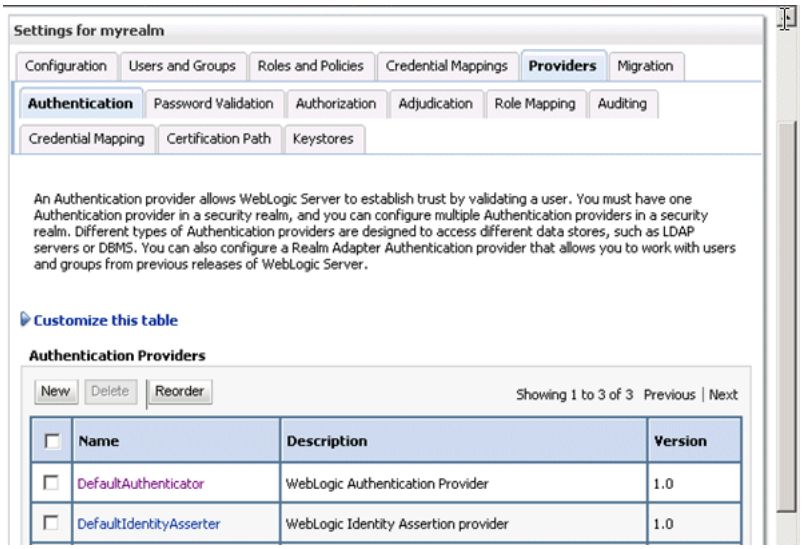
   For more information, see Section 1.6.1, "Using Oracle WebLogic Server Administration Console".



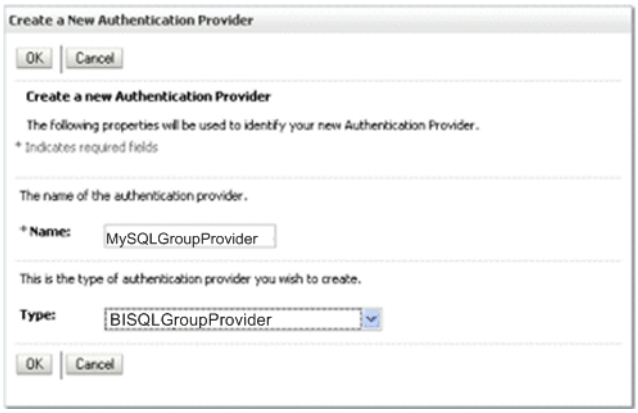
2. Select **Security Realms** from the left pane and click **myrealm**.

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

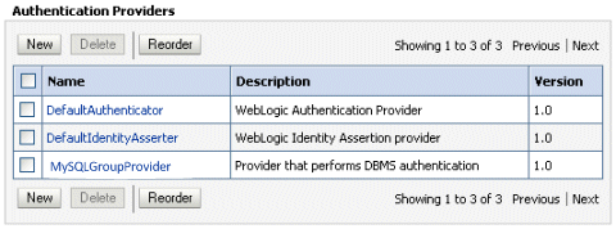
3. Display the **Providers** tab, then display the **Authentication** sub-tab.



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, MySQLGroupProvider.
   - **Type**: Select BISQLGroupProvider from the list.
   - Click **OK** to save the changes and display the authentication providers list updated with the new authentication provider.



6. In the authentication providers table, click MySQLGroupProvider in the **Name** column to display the Settings page.
7. Display the **Provider Specific** tab to specify the SQL statements used to query and authenticate against your database tables.

8. Specify the **DataSource Name**. This should be the JNDI name rather than the data source name. For example: `jdbc/BIDatabaseGroupDS`.

   Table 3–3 shows SQL statements for the sample schema outlined in Section 3.4.4.2

Table 3–3 SQL Statements for the Sample Schema

Query	SQL	Notes
SQL List Groups	SELECT G_NAME FROM GROUPS WHERE G_NAME LIKE ?	The SQL statement used to retrieve group names that match a wildcard. The SQL statement requires a single parameter for the group name and must return a resultSet containing matching groups.
SQL Group Exists	SELECT G_NAME FROM GROUPS WHERE G_NAME = ?	The SQL statement used to look up a group. The SQL statement requires a single parameter for the group name and must return a resultSet containing at most a single record containing the group.
SQL Is Member	SELECT G_MEMBER FROM GROUPMEMBERS WHERE G_NAME = ? AND G_MEMBER = ?	The SQL statement used to look up members of a group. The SQL statement requires two parameters: a group name and a member or group name. It must return a resultSet containing the group names that matched.
SQL List Member Groups	SELECT G_NAME FROM GROUPMEMBERS WHERE G_MEMBER = ?	The SQL statement used to look up the groups a user or group is a member of. The SQL statement requires a single parameter for the username or group name and returns a resultSet containing the names of the groups that matched.
SQL Get Group Description (if description supported enabled)	SELECT G_DESCRIPTION FROM GROUPS WHERE G_NAME = ?	The SQL statement used to retrieve the description of a group. Only valid if Descriptions Supported is enabled. The SQL statement requires a single parameter for the group name and must return a resultSet containing at most a single record containing the group description.

> **Note:** If you are using a different table structure, you might need to adapt these SQL statements (table or column names) to your own schema. Also, you should leave the question mark (?) as a runtime query placeholder (rather than hardcode a user or group name).

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

9. Enter all of the SQL statements appropriate to your authenticator.
10. Click **Save**.
11. Perform the following steps to reorder the authentication providers:
    a. Display the **Providers** tab.
    b. Click **Reorder** to display the **Reorder Authentication Providers** page

c. Select **BISQLGroupProvider** and use the arrow buttons to move it into the first position in the list.

d. Click **OK** to save your changes.

12. Perform the following steps to configure the **Control Flag** setting of **BISQLGroupProvider**:

a. At the main **Settings for myrealm** page, display the **Providers** tab, then display the **Authentication** sub-tab, then select **BISQLGroupProvider** to display its configuration page.

b. Display the **Configuration\Common** tab and select 'OPTIONAL' from the **Control Flag** list.

For more information, see Section 3.4.6, "Setting the JAAS Control Flag Option".

c. Click **Save**.

13. In the Change Center, click **Activate Changes**.

14. Restart the Oracle Business Intelligence components (use Fusion Middleware Control once the Administration Server has been restarted), Oracle WebLogic Server, and Managed servers.

Note: Check the **Users and Groups** tab to confirm that the database users and groups appear there.

3.4.4.4 Configuring the Virtualized Identity Store

You configure the virtualized identity store as follows:

- Section 3.4.4.4.1, "Enabling Virtualization by Configuring the Identity Store"
- Section 3.4.4.4.2, "Configuring SSL Against LDAP"
- Section 3.4.4.4.3, "Configuring a Database Adaptor to Retrieve Group Information"

3.4.4.4.1 Enabling Virtualization by Configuring the Identity Store You configure the identity store to enable virtualization so that more than one identity store can be used with the identity store service, and therefore user profile information can be split across different authentication providers (identity stores).

For more information, see Section 3.4.5, "Configuring Multiple Authentication Providers Using Fusion Middleware Control".

3.4.4.4.2 Configuring SSL Against LDAP If you have configured an LDAP Authenticator to communicate over SSL (one-way SSL only), you must put the corresponding LDAP server's route certificate in an additional keystore used by the virtualization (libOVD) functionality.

For more information, see Section 5.4.6, "Configuring SSL when Using Multiple Authenticators".

3.4.4.4.3 Configuring a Database Adaptor to Retrieve Group Information You configure a database adaptor to make it appear like an LDAP server, which enables the virtualized identity store provider to retrieve group information from a database using the database adapter.

To configure a database adaptor to retrieve group information:

This task shows how to edit and apply adapter templates that specify how to use your database tables as an identity store to map groups.

1. Create a file named bi_sql_groups_adapter_template.xml.

   This file describes the mapping of the GROUPMEMBERS table to a virtual LDAP store.

2. Make sure that the file contains the following contents:

   Note: You must adapt the sections of **bold** text below to match your table and column attributes against LDAP server attributes. The example shown here is of the sample schema that is used throughout Section 3.4.4.

   Note: For the element:<param name="ReplaceAttribute" value="uniquemember={cn=%uniquemember%,cn=users,dc=oracle,dc=com}"/>

   This must match the user attribute and root User DN of the main authenticator. For example, for the default authenticator:

   uid=%uniquemember%,ou=people,ou=myrealm,dc=bifoundation_domain

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<adapters schvers="303" version="1"
xmlns="http://www.octetstring.com/schemas/Adapters"
xmlns:adapters="http://www.w3.org/2001/XMLSchema-instance">
   <dataBase id="directoryType" version="0">
      <root>%ROOT%</root>
      <active>true</active>
      <serverType>directoryType</serverType>
      <routing>
         <critical>true</critical>
         <priority>50</priority>
         <inclusionFilter/>
         <exclusionFilter/>
         <plugin/>
         <retrieve/>
         <store/>
         <visible>Yes</visible>
         <levels>-1</levels>
         <bind>true</bind>
         <bind-adapters/>
         <views/>
         <dnpattern/>
      </routing>
      <pluginChains
xmlns="http://xmlns.oracle.com/iam/management/ovd/config/plugins">
         <plugins>
            <plugin>
               <name>VirtualAttribute</name>

<class>oracle.ods.virtualization.engine.chain.plugins.virtualattr.VirtualAttributePlugin</class>
               <initParams>
                  <param name="ReplaceAttribute"
```

```
value="uniquemember={cn=%uniquemember%,cn=users,dc=oracle,dc=com}"/>
               </initParams>
            </plugin>
         </plugins>
         <default>
            <plugin name="VirtualAttribute"/>
         </default>
         <add/>
         <bind/>
         <delete/>
         <get/>
         <modify/>
         <rename/>
      </pluginChains>
      <driver>oracle.jdbc.driver.OracleDriver</driver>
      <url>%URL%</url>
      <user>%USER%</user>
      <password>%PASSWORD%</password>
      <ignoreObjectClassOnModify>false</ignoreObjectClassOnModify>
      <includeInheritedObjectClasses>true</includeInheritedObjectClasses>
      <maxConnections>10</maxConnections>
      <mapping>
         <joins/>
         <objectClass name="groupofuniquenames" rdn="cn">
            <attribute ldap="cn" table="GROUPMEMBERS" field="G_NAME" type=""/>
            <attribute ldap="description" table="GROUPMEMBERS" field="G_NAME"
type=""/>
            <attribute ldap="uniquemember" table="GROUPMEMBERS" field="G_
MEMBER" type=""/>
         </objectClass>
      </mapping>
      <useCaseInsensitiveSearch>true</useCaseInsensitiveSearch>
      <connectionWaitTimeout>10</connectionWaitTimeout>
      <oracleNetConnectTimeout>0</oracleNetConnectTimeout>
      <validateConnection>false</validateConnection>
   </dataBase>
</adapters>
```

3. Customize appropriate sections highlighted in bold, for the following elements:

   - ReplaceAttribute

     Specifies how to define the unique member for a group (the %uniquemember% is a placeholder for a value which will be passed in at runtime when looking up whether a user is a member of a group)

     The only aspect of this element you may want to change is the specification of the root for your users. While this is notional, by default it must match whatever you specify as the root of your user population when you run the libovdadapterconfig script in Step 10.

   - groupofuniquenenames

     Specifies how group attributes are mapped to database fields and as with the user, the attributes correspond to the defaults in Weblogic Embedded LDAP.

     You must map the following attributes:

     – **cn** (map to a unique name for your group)

     – **uniquemember** (map to the unique name for your user in the user/group mapping table in your database schema)

Mapping the following attributes is optional:

- **description** is optional (although clearly helpful)
- **orclguid** (maps to a UID, if available in your database schema)

No other attributes are user-configurable.

4. Copy the adapter file into the following folder:

   <*MW_HOME*>/oracle_common/modules/oracle.ovd_11.1.1/templates/

5. Open a command prompt/terminal at:

   <*MW_HOME*>/oracle_common/bin

6. Ensure the following environment variables are set:
   - ORACLE_HOME=<*MW_HOME*>/Oracle_BI1
   - WL_HOME=<*MW_HOME*>/wlserver_10.3/
   - JAVA_HOME=<*MW_HOME*>/jdk160_24/
7. Run the libovdadapterconfig script to create a database adapter from the template file. The syntax is:

```
libovdadapterconfig -adapterName <name of adapter> -adapterTemplate <name (NOT
including path) of template file which defines adapater> -host localhost -port
<Admin Server port> -userName <user id of account which has administrative
privileges in the domain> -domainPath <path to the BI domain> -dataStore DB
-root <nominal specification of a pseudo-LDAP query to treat as the "root" of
this adapter - must match that specified in template for adapter 2 above>
-contextName default -dataSourceJNDIName <JNDI name for DataSource which points
at the database being mapped>
```

For example:

```
./libovdadapterconfig.sh -adapterName biSQLGroupAdapter -adapterTemplate bi_
sql_groups_adapter_template.xml -host localhost -port 7001 -userName weblogic
-domainPath /opt/oracle_bi/user_projects/domains/bifoundation_domain/
-dataStore DB -root cn=users,dc=oracle,dc=com -contextName default
-dataSourceJNDIName jdbc/BIDatabaseGroupDS
```

Note: The dataSourceJNDIName must be the JNDI name and not just the DS name.

Note: The root parameter value should match the root dn specified in the <param name="replaceattribute" element in the adaptor template. For example, if user is specified in the default authenticator, the root would normally be set to ou=people, ou=myrealm, dc=bifoundation_domain.

   The script should exit without error.

8. Restart WebLogic Administration Server and Managed servers.

Note: When you start WebLogic you will see the following warning which you can ignore:

Warning: BISQLGroupsProvider: Connection pool not usable.

You should now be able to log in to WebLogic and Oracle Business Intelligence using credentials stored in the database.

3.4.4.5 Testing the Configuration by Adding a Database Group to an Application Role

To test the configuration by adding a database group to an application role:

1. Log in to Fusion Middleware Control, and open WebLogic domain and bifoundation_domain in the navigation menu on the left of the page.

   For more information, see Section 1.6.2, "Using Oracle Fusion Middleware Control".

2. Right-click bifoundation_domain and select **Security**, then **Application Roles** to display the Application Role Configuration page.

3. Add a database group which contains an LDAP user to one of the application roles (for example, BIAdministrator) which that user does not currently have access to.

4. Log in to Oracle Business Intelligence as a user that is a member of the group that was newly added to the application role.

   In the top right of the page, you will see the text "Logged in as <*user id*>".

5. Click the user id to display a drop down menu.

6. Select **My Account** from the menu.

7. Display the **Roles and Catalog Groups** tab and verify the user now has the new application role.

3.4.4.6 Correcting Errors In the Adaptors

You cannot modify an existing database adapter, so if you make an error in either the libovdadapter command, or the templates you use to create the adapters, you must delete then recreate the adapter.

For more information, see Section 3.4.3.6, "Correcting Database Adapter Errors Deleting and Recreating the Adapter".

3.4.5 Configuring Multiple Authentication Providers Using Fusion Middleware Control

This section describes how to configure Oracle Business Intelligence to use multiple authentication providers using Fusion Middleware Control.

To configure multiple authentication providers using Fusion Middleware Control:

If you are communicating with LDAP over SSL (one-way SSL only), see Section 5.4.6, "Configuring SSL when Using Multiple Authenticators".

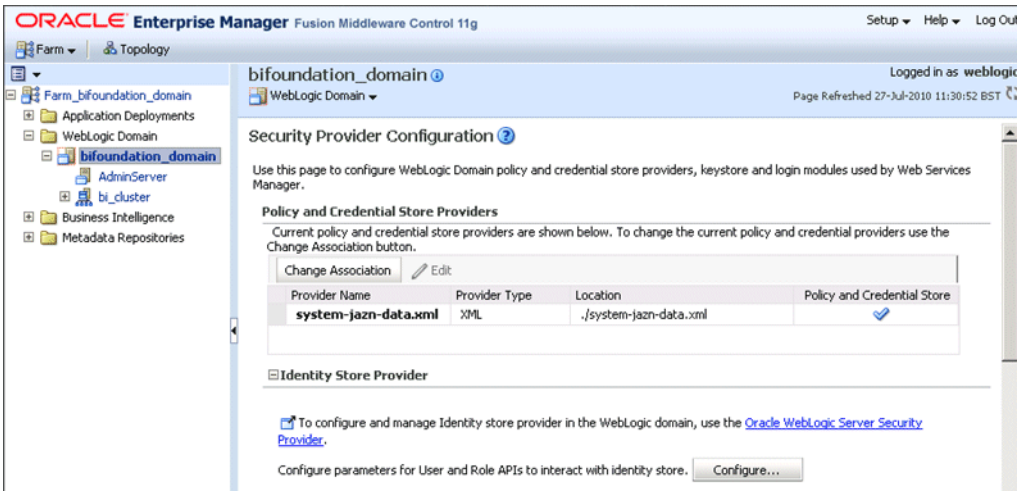
1. (Optional) If not already done, configure supported authentication providers as described in Section 3.4.

2. Log in to Fusion Middleware Control.

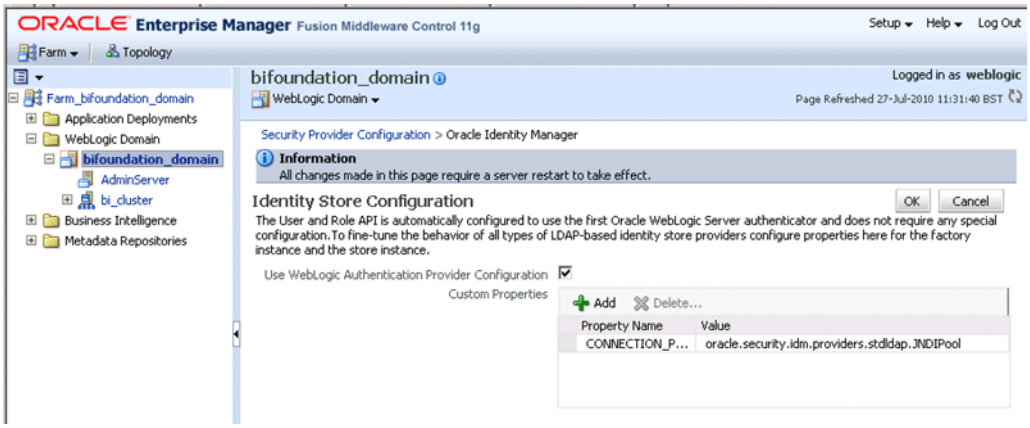
For more information, see Section 1.6.2, "Using Oracle Fusion Middleware Control".

3. From the navigation pane expand the **WebLogic Domain** folder and select **bifoundation_domain**.

4. Right-click bifoundation_domain and select Security, then Security Provider Configuration to display the Security Provider Configuration page.



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



6. In the Custom Properties area, use the **Add** option to add a new custom property as follows:

   Property Name=`virtualize`

   Value=`true`

Note: If you are using multiple authentication providers, go to Section 3.4, "Configuring Alternative Authentication Providers" and configure the **Control Flag** setting as follows:

- If each user appears in only one authentication provider

  set the value of **Control Flag** for all authentication providers to SUFFICIENT

- If users appear in more than one authentication provider (for example, if a user's group membership is spread across more than one authentication provider)

  set the value of **Control Flag** for all authentication providers to OPTIONAL

7. Click **OK** to save the changes.
8. Restart the Administration Server and Managed Servers.

3.4.6 Setting the JAAS Control Flag Option

When you configure multiple Authentication providers, use the JAAS Control Flag for each provider to control how the Authentication providers are used in the login sequence. You can set the JAAS Control Flag in the Oracle WebLogic Server Administration Console. For more information, See "Set the JAAS control flag" in the *Oracle Fusion Middleware Oracle WebLogic Server Administration Console Online Help*. You can also use the WebLogic Scripting Tool or Java Management Extensions (JMX) APIs to set the JAAS Control Flag for an Authentication provider.

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, control is returned to the application.
- SUFFICIENT - This LoginModule need 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 - This LoginModule can succeed or fail. 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.

When additional Authentication providers are added to an existing security realm, by default the Control Flag is set to OPTIONAL. If necessary, change the setting of the Control Flag and the order of Authentication providers so that each Authentication provider works properly in the authentication sequence.

3.4.7 Configuring a Single LDAP Authentication Provider as the Authenticator

This topic explains how to reconfigure Oracle Business Intelligence to use a single LDAP authentication provider, by disabling the default WebLogic Server LDAP authenticator.

When you install Oracle Business Intelligence, the system is automatically configured to use WebLogic Server LDAP as the default authenticator. The install process will automatically generate the required users and groups in WebLogic Server LDAP. However, you may have your own LDAP directory (for example Oracle Internet Directory) that you may want to use as the default authenticator, and disable the WebLogic Server default authenticator. Having a single source authentication provider prevents user names and passwords being derived from multiple authentication sources, which could lead to multiple points of attack, or entry from unauthorizeed users.

This topic contains the following sections:

3.4.7.1 Configuring Oracle Internet Directory LDAP Authentication as the Only Authenticator

The examples shown in this section are for configuring Oracle Internet Directory (OID LDAP) but could easily apply to other LDAP authentication providers by using minor changes.

To configure Oracle Internet Directory LDAP authentication as the only authenticator:

Task 1 Enable Backup and Recovery

Before you begin the process of disabling the WebLogic Server LDAP default method of authentication it is strongly recommended that you back up the system first.

Otherwise, if you make an error during configuration you may find that you become locked out of the system or cannot restart it.

To enable backup and recovery, during the re-configuration phase, take a copy of the config.xml file in <*BIEE_HOME*>\user_projects\domains\bifoundation_domain\config directory.

As you make changes it is advised that you keep copies of this file.

Task 2 Configure the System to use WebLogic Server and an Alternative Authentication Provider

To remove the default WebLogic Server authenticators and use an alternative LDAP source (for example, OID LDAP), you must configure the system to use both WebLogic Server and the alternative method. For more information, see Section 3.4, "Configuring Alternative Authentication Providers". Your starting point should be that the WebLogic Server LDAP users (default authenticator) and the new alternative LDAP users are both configured to allow access to Oracle Business Intelligence.

When you have configured the system to enable you to log on as either a WebLogic Server LDAP user or an OID LDAP user, you can then proceed to follow the steps to remove the WebLogic Server default authenticator, as described in these tasks.

Task 3 Identify or Create Essential Users Required in OID LDAP

You must ensure that the essential users shown in Table 3–4 are migrated from WebLogic Server LDAP to OID LDAP.

Table 3–4 Essential Users Required in OID LDAP

Standard WebLogic Server Users	New Users Required in OID LDAP
BISystemUser	OID_BISystemUser (this can be any existing OID LDAP user)
WebLogic	OID_Weblogic (this can be any existing OID LDAP user)
OracleSystemUser	OracleSystemUser (this user must exist with this name in OID LDAP, which is a fixed requirement of OWSM)

Three users are created during install:

- BISystemUser

  This user is created in WebLogic Server, and is used to perform the communication between Presentation Services and Oracle Business Intelligence components. You must create or identify an equivalent user in OID LDAP (for example, OID_BISystemUser). Ensure that the passwords used here confirm to your security password standards (for example, never use welcome1).

- Weblogic (specified during install or upgrade, so can be different).

  This administrator user is created during the install (sometimes called Weblogic, but can have any name). You need to identify or create an equivalent user in OID LDAP but this user can have any name.

- OracleSystemUser

  This user is specifically required (by Oracle Web Services Manager - OWSM) for the Global Roles mapping, and you must create this user in OID LDAP using this exact name.

Task 4 Identify or Create Essential Groups in OID LDAP

The essential groups shown in Table 3–5 are required in the OID LDAP directory.

Table 3–5 Essential Groups Required

WebLogic Server Groups Automatically Created	New OID LDAP Groups Required
Administrators	OID_Administrators
AdminChannelUsers	OID_AdminChannelUsers
AppTesters	OID_AppTesters
CrossDomainConnectors	OID_CrossDomainConnectors
Deployers	OID_Deployers
Monitors	OID_Monitors
Operators	OID_Operators
OracleSystemGroup	OracleSystemGroup (fixed requirement)
BIAdministrators	OID_BIAdministrators
BIAuthors	OID_BIAuthors
BIConsumers	OID_BIConsumers

The groups in Table 3–5 are automatically created in WebLogic Server during the default Oracle Business Intelligence installation process.

Before you can remove the default WebLogic Server authentication you need to identify OID LDAP groups that will replace the WebLogic Server groups. You can choose to have an individual OID LDAP group for each WebLogic Server group (in Table 3–5) or use a single OID LDAP group to replace one or many WebLogic Server groups.

Currently the only specific requirement is that you must have a group defined in OID LDAP as OracleSystemGroup using this exact name (an OWSM requirement).

Task 5 Associate OID LDAP Groups with Global Roles in the WebLogic Console

The global role mappings shown in Table 3–6 must be configured in OID LDAP.

Table 3–6 Global Role Mapping in WebLogic Admin Console

Global Roles	Current WebLogic Server Groups	New OID LDAP Groups Required
Admin	Administrators	OID_Administrators
AdminChannelUsers	AdminChannelUsers	OID_AdminChannelUsers
AppTester	AppTesters	OID_AppTesters
CrossDomainConnector	CrossDomainConnectors	OID_CrossDomainConnectors
Deployer	Deployers	OID_Deployers
Monitor	Monitors	OID_Monitors
Operator	Operators	OID_Operators
OracleSystemRole	OracleSystemGroup	OracleSystemGroup (fixed requirement)

You must associate the global roles from Table 3–6 (displayed in the WebLogic Server Admin Console) with your replacement OID LDAP groups (defined in Task 4), before you can disable the default WebLogic Server authenticator.

To associate OID LDAP groups with global roles in Oracle WebLogic Server Administration Console:

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

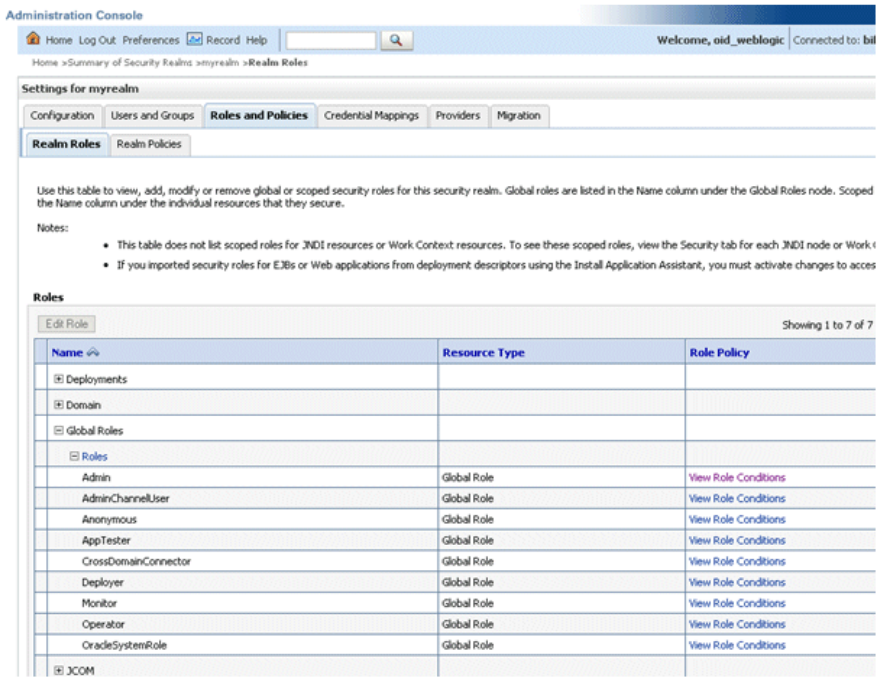
   For more information, see Section 1.6.1, "Using Oracle WebLogic Server Administration Console".

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

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

3. Click **Realm Roles**.

4. Click **Global Roles** and expand **Roles**.



5. Add a new condition for each Role as follows:

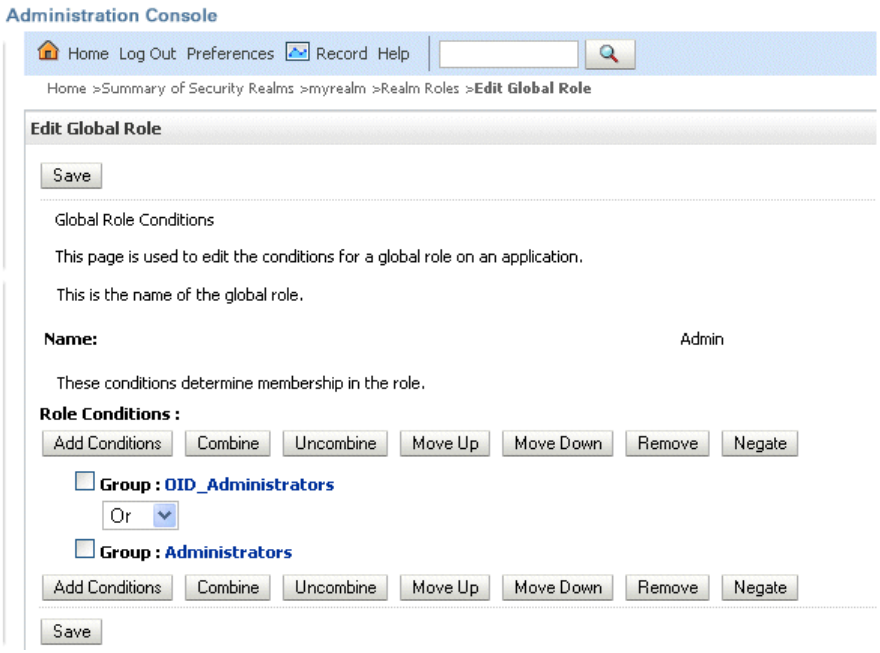
   Note: Do not do add a new condition for the Anonymous and Oracle System roles, which can both remain unchanged.

   a. Click View Role Conditions.

   b. Select group from the **Predicate List** drop down.

   c. Enter your newly-associated OID LDAP group from Table 3–5.

      For example, assign the Admin role to the OID_Administrators role.



Note: Once you have successfully disable the Default WebLogic Server Authentication you can return here and remove the old WebLogic Server groups (for example, here you would remove Group: Administrators). For more information, see Task 12, "Setup Tasks After Single OID LDAP Authentication".

d. Save your changes.

Task 6 Set User to Group Membership in OID LDAP

Now that you have created new users and groups in OID LDAP to replicate the users and groups automatically created in WebLogic Server LDAP you must ensure that these users and groups also have the correct group membership in OID LDAP as shown in Table 3–7.

Table 3–7 User to Group Membership Required in OID LDAP

New OID LDAP User	Is A Member Of These New OID LDAP Groups
OID_Weblogic	OID_Administrators OID_BIAdministrators
OracleSystemUser **Note:** A user with this exact name must exist in OID LDAP.	OracleSystemGroup **Note:** A group with this exact name must exist in OID LDAP

Note: In order to achieve the user and group membership shown in Table 3–7 you must have suitable access to update your OID LDAP server, or someone else must be able to update group membership on your behalf.

Task 7 Set OID LDAP Users and Groups Application Roles Membership in Fusion Middleware Control

You must add the recently created OID LDAP users and groups (in Table 3–8), as members of existing application roles using Fusion Middleware Control.

Table 3–8 OID LDAP User and Group Application Role Membership Required

Make a member of the existing WebLogic Server application roles	New OID LDAP User/Groups
BISystem	OID_BISystemUser (OID user)
BIAdministrator	OID_BIAdministrators (OID group
BIAuthor	OID_BIAuthors (OID group)
BIConsumer	OID_BIConsumers (OID group)

To set required OID LDAP users and group application roles membership using Fusion Middleware Control:

1. Log in to Fusion Middleware Control.

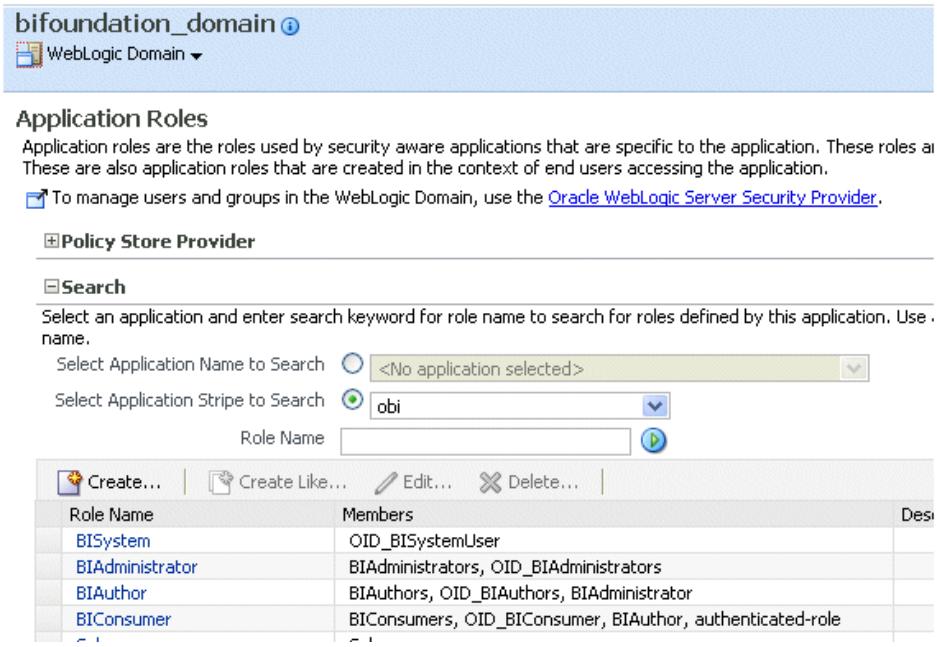
   For more information, see Section 1.6.2, "Using Oracle Fusion Middleware Control".

2. From the navigation pane expand the **Business Intelligence** folder and select **coreapplication**.

3. Display the application roles for Oracle Business Intelligence.

   For more information, see Section 2.4.1, "Displaying Application Policies and Application Roles Using Fusion Middleware Control"

4. Assign members to application roles as follows:



Caution: Although you can assign groups to the BISystem application role you should only ever assign users to this role to protect security.

Note: Oracle recommends that you remove the BISystemUser as a member of the BISystem application role at this point.

Task 8 Update the Credential Store Password for the New Trusted System User

The user name and password you created for the BISystemUser in OID LDAP must be exactly the same as created in Task 3, "Identify or Create Essential Users Required in OID LDAP" (for example, for the OID_BISystemUser).

To update the Credential Store password for the new OID_BISystemUser:

1. Log in to Fusion Middleware Control.

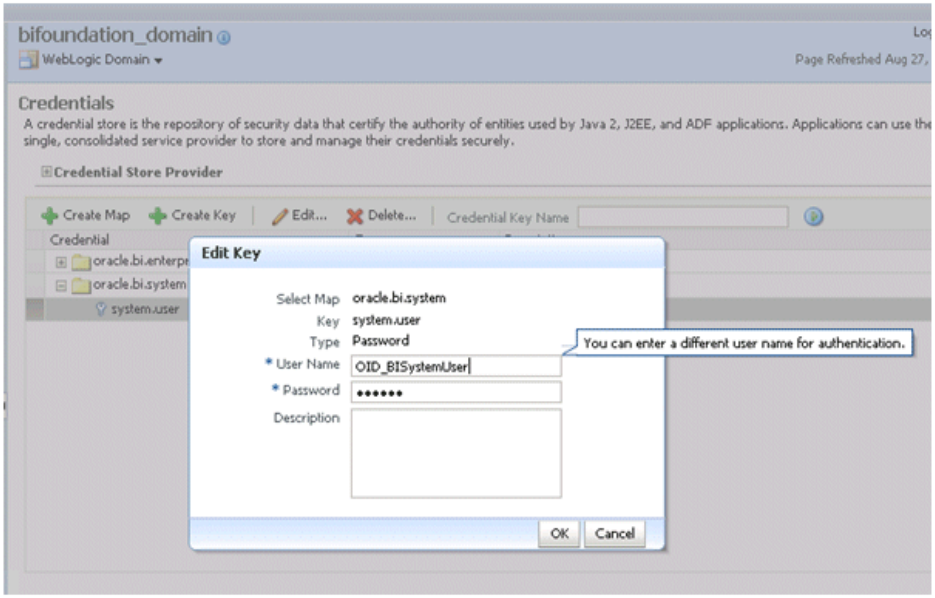
   For more information, see Section 1.6.2, "Using Oracle Fusion Middleware Control".

2. From the navigation pane expand the **WebLogic Domain** folder and select **bifoundation_domain**.

3. Click WebLogic Domain in main pane to display a menu, then select **Security**, and **Credentials** to display the Credentials page.

4. Expand oracle.bi.system and select system.user.

5. Click the **Edit** to display the Edit Key dialog.



6. Input the new user name and password.

7. Click **OK**.

Task 9 Remove the Default Authenticator

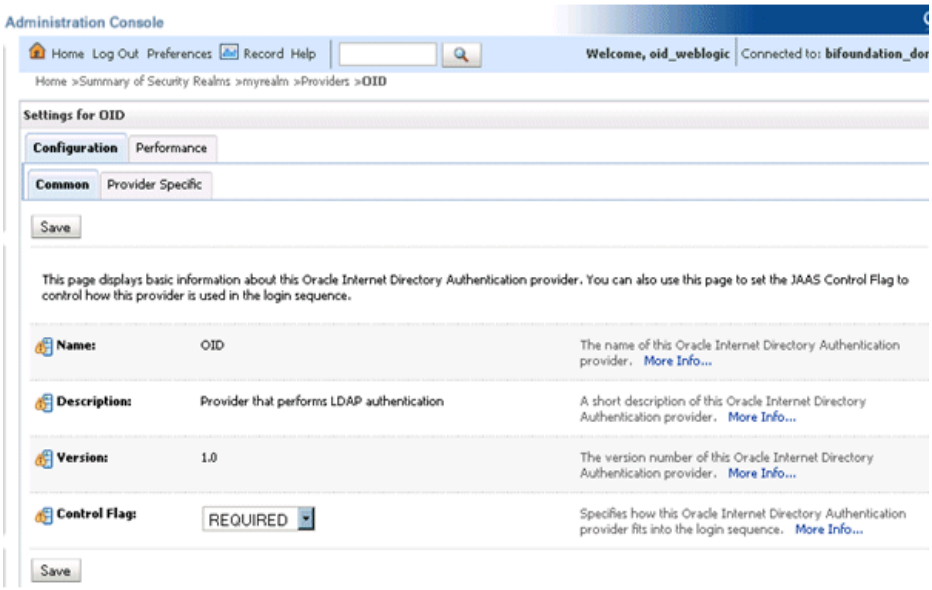
You are now ready to remove the Default Authenticators.

To remove the default authenticators:

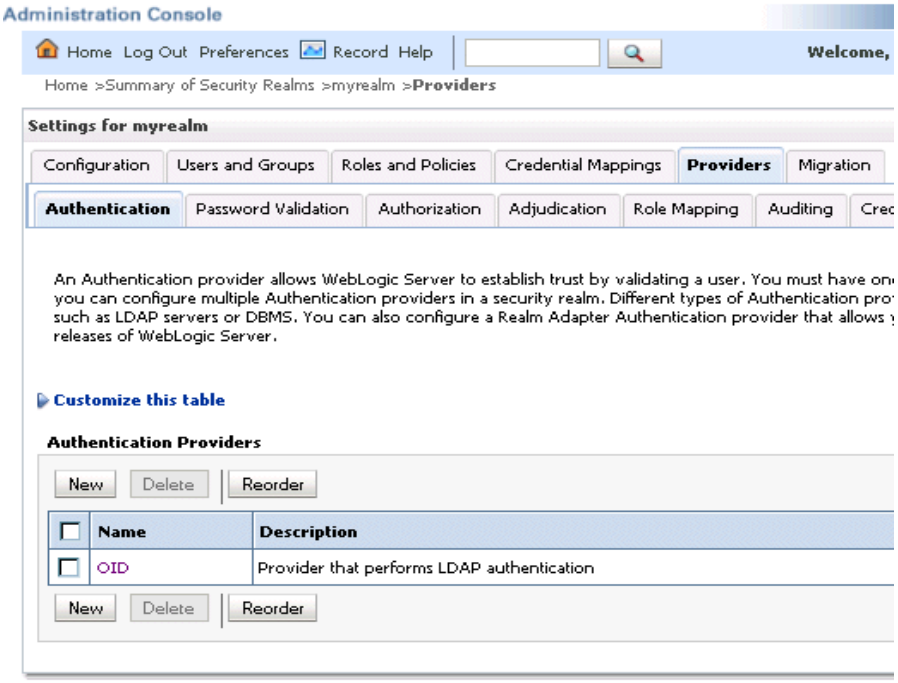
You must have first created an LDAP authenticator that maps to your LDAP source (for more information, see Task 2, "Configure the System to use WebLogic Server and an Alternative Authentication Provider").

1. Change the **Control Flag** from SUFFICIENT to REQUIRED in the Oracle WebLogic Server Administration Console.

   For more information, see Section 3.4.6, "Setting the JAAS Control Flag Option".



2. Save the changes.
3. Delete any other authenticators so that your OID LDAP authenticator is the single source.



Task 10 (Optional) Remove Old GUID References

Complete this task if you are using OID LDAP for the first time, that is, if moving from a 10*g* LDAP authentication (upgraded to 11*g*) to OID LDAP authentication. This will

resynchronize the system user GUIDs (Global Unique Identifiers). Otherwise you may find you cannot login and will get the following error message:

```
The GUID of user {username} does not match user reference GUID
of the repository. Please ask the administrator to delete the
old user reference at the repository and login again.
```

To remove old GUID references:

1. Stop all Oracle Business Intelligence Services.

   In Windows use the menu option **Stop BI Services** providing the original administrator user name, and password specified during install (for example, weblogic/welcome1).

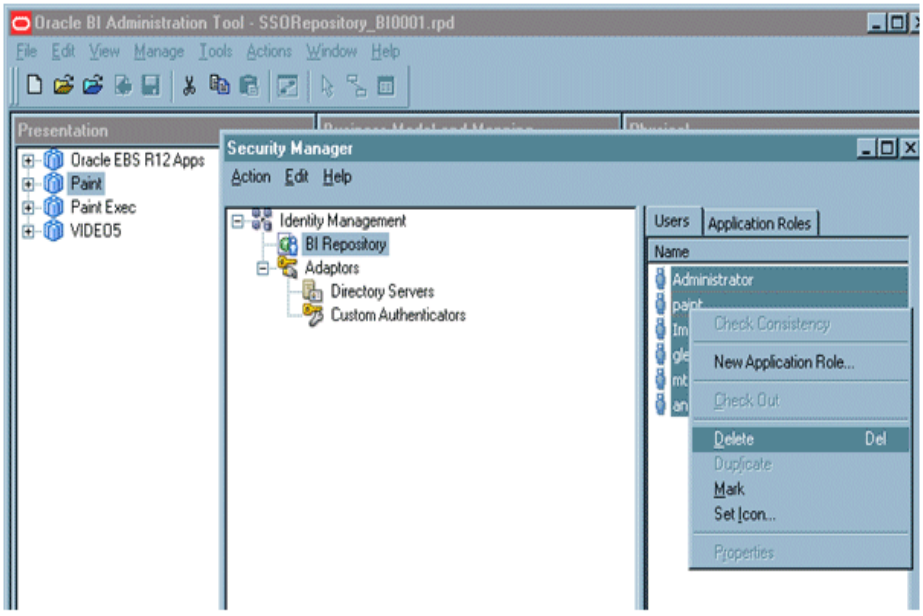
2. In the Administration Tool, open the repository file that you are using in 11*g*, in offline mode.

3. Select Manage and Identity from the menu.

4. Click **BI Repository** and display the Users tab.

5. Select all users and delete them.

Note: If you have specific permissions defined in the repository file for a particular user these will be lost. In this case, when you start up your Business Intelligence system you will need to re-associate any user level permissions with these users in your LDAP (OID) source. This will ensure that a user with the same name, (but who is not the same person), will be identified correctly by the system, as a different user.



Task 11 Restart the BI Services

Now you are ready to restart the BI services. You must use the new OID administrator user (for example, OID_Weblogic), because the WebLogic Server administration user created during installation was removed, and users now exist in the single OID source.

The OID administration user must have sufficient privileges (granted by the Global Admin role) to start WebLogic.

> **Note:** When you log in to the Administration Tool online you must now provide the OID LDAP user and password (for example, OID_Weblogic) along with the repository password.

Task 12 Setup Tasks After Single OID LDAP Authentication

Complete this task if everything is working correctly.

> **Note:** Back up your config.xml, now, before performing this step (see Task 1, "Enable Backup and Recovery")

To remove all automatically created WebLogic server roles from the OR clause:

1. Edit global roles.

   For more information, section: Task 5, "Associate OID LDAP Groups with Global Roles in the WebLogic Console".

2. Remove all WebLogic Server roles that were automatically created, from the OR clause.

   For example:

   - Admin
   - AdminChannelUsers
   - AppTester
   - CrossDomainConnector
   - Deployer
   - Monitor
   - Operator

3. Save your changes.

Task 13 Stop Alternative Methods of Authentication

Oracle Business Intelligence allows various forms of authentication methods to be applied at once. While some can see this as a desirable feature it also comes with security risks. To implement a single source of authentication, you must remove the authentication methods that use initialization blocks from the Metadata Repository.

To stop all initialization block authentication access:

You stop access through initialization blocks using the Oracle BI Administration Tool. Successful authentication requires a user name, and initialization blocks populate user names using the special system session variable called USER.

1. Remove the USER System Variable from the Metadata Repository.
2. Ensure that initialization blocks in the Metadata Repository have the **Required for authentication** check box cleared.
3. Check that initialization blocks in the Metadata Repository that set the system session variables PROXY and PROXYLEVEL do not allow users to bypass security.

The system variables PROXY and PROXYLEVEL allow connected users to impersonate other users with their security profile. This method is acceptable when the impersonated user account has less privileges, but if the account has more privileges it can be a security issue.

Caution: If you disable an initialization block, then any dependent initialization blocks will also be disabled.

You can now be sure that any attempted access using initialization block authentication cannot be successful. However, you must check all of your initialization blocks.

3.4.7.2 Troubleshooting

You might receive the following error after you have configured Oracle Internet Directory LDAP authentication as the single source:

```
<Critical> <WebLogicServer> <BEA-000386> <Server subsystem
failed.

Reason: weblogic.security.SecurityInitializationException: User
<oidweblogic> is not permitted to boot the server. The server
policy may have changed in such a way that the user is no longer
able to boot the server. Reboot the server with the
administrative user account or contact the system administrator
to update the server policy definitions.
```

Solution

If when you restart the system as the new WebLogic OID LDAP administrator (oidweblogic), you are locked out, and the message is displayed, it is because the oidweblogic user has insufficient privileges. The oidweblogic user requires the Admin global role to enable it to belong to an OID LDAP Administrator group. You resolve this issue by adding the BIAdministrators group (or an OID LDAP equivalent) to the Admin global role.

Note: To restore a previously working configuration, you must replace the latest updated version of the config.xml file with a backup version that you have made before changing the configuration (for more information, see Task 1, "Enable Backup and Recovery").

To complete the restoration of the backup config.xml file, restart Oracle Business Intelligence as the original WebLogic administrator user, instead of as the OID LDAP user.

3.5 Configuring User and Group Name Attributes In the Identity Store

This topic contains the following sections:

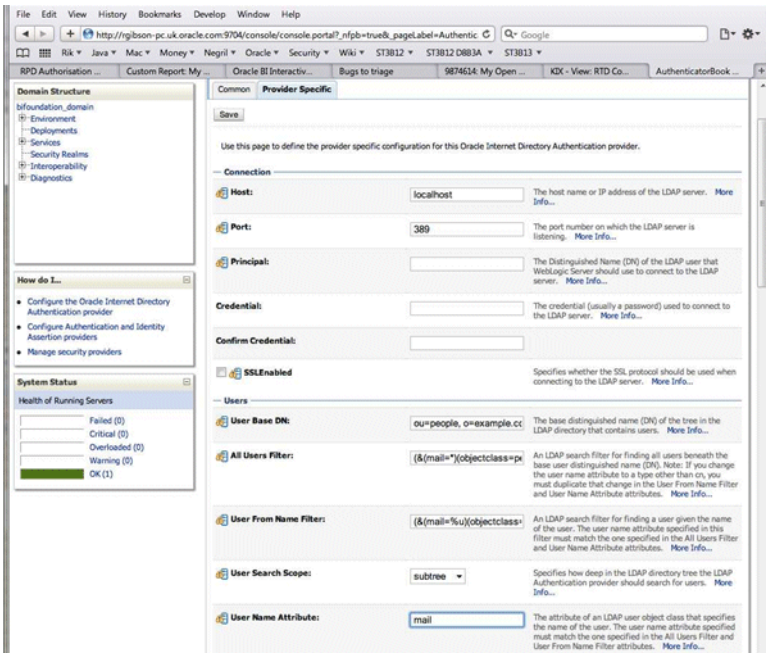
3.5.1 Configuring the User Name Attribute In the Identity Store

If you configure an alternative authentication provider such as Oracle Internet Directory (OID LDAP) or Active Directory (AD), 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 alternative 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.

Figure 3–4 shows the **User Name Attribute** in OID LDAP Authenticator set to `mail`.

Figure 3–4 Example - Provider Specific Tab



The UserNameAttribute in the alternative authentication provider is usually set to the value 'cn'; if it is not, you must make sure the settings for AllUsersFilter and UserFromNameFilter are configured correctly as shown in Table 3–9. Table 3–9 illustrates the default setting (using the value cn), and a required new setting (using a new value in the attribute AnOtherUserAttribute).

Table 3–9 Changing User Name Attributes

Attribute Name	Default Setting	Required New Setting
UserNameAttribute	cn	**AnOtherUserAttribute**
AllUsersFilter	(&(cn=*)(objectclass=person))	(&(**AnOtherUserAttribute** =*)(objectclass=person))
UserFromNameFilter	(&(cn=%u)(objectclass=person))	(&(**AnOtherUserAttribute** =%u)(objectclass=person))

Make the changes in the Provider Specific tab, using Table 3–9 (substitute the AnOtherGroupAttribute setting with your own value). For more information about how to display the Provider Specific tab, see Section 3.4, "Configuring Alternative Authentication Providers".

Note: For the UserName Attribute only; you must use the following task to add two properties to the identity store configuration (user.login.attr and username.attr). This specification tells the identity store about the attribute from which you expect to retrieve the user name. The default is "uid" if no attribute is specified.

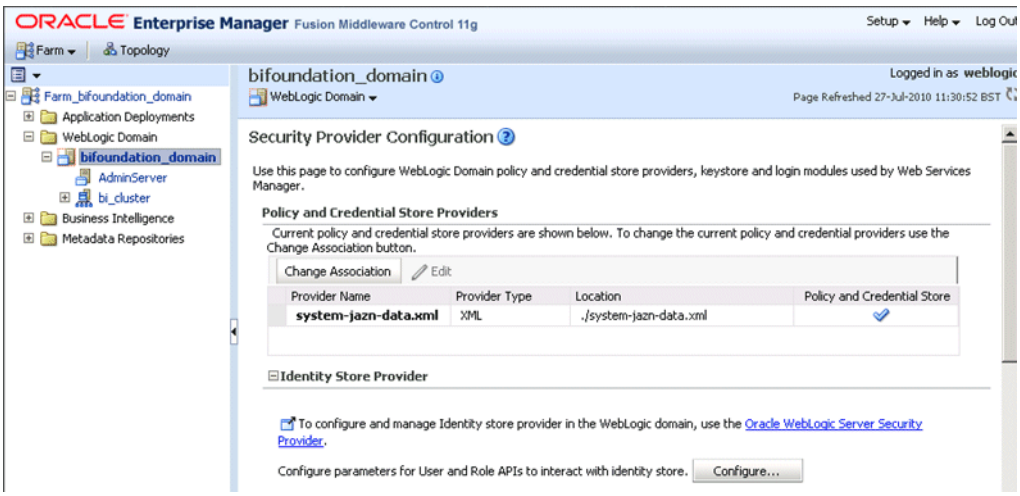
To configure the User Name attribute in the identity store:

1. Log in to Fusion Middleware Control.

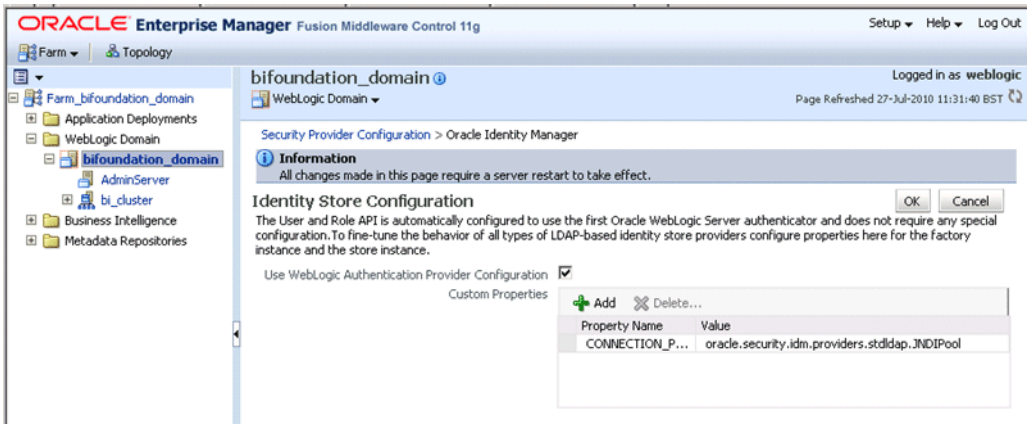
   For more information, see Section 1.6.2, "Using Oracle Fusion Middleware Control".

2. From the navigation pane expand the **WebLogic Domain** folder and select **bifoundation_domain**.

3. Right-click bifoundation_domain and select Security, then Security Provider Configuration to display the Security Provider Configuration page.



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



5. In the Custom Properties area, click **Add** to add the custom properties that are described in Table 3–10.

Table 3–10 Custom Properties

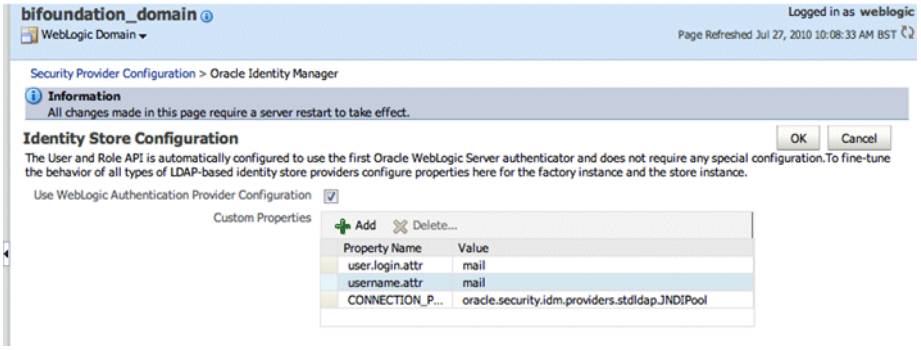
Property Name	Value
user.login.attr	Specify the User Name Attribute that is set in the authentication provider. For example, if the User Name Attribute is set to `mail` in the authentication provider, then set this value to `mail`.

Table 3–10 (Cont.) Custom Properties

Property Name	Value
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`.

Figure 3–5 shows an example set of Custom Properties with the User Name Attribute set to `mail`.

Figure 3–5 Custom Properties - User Name Attribute



6. Click **OK** to save the changes.
7. Restart the Administration Server.

> **Note:** Ensure that the users and groups from your authentication provider (for example, OID LDAP, AD), are displayed in WebLogic Console, as described in Step 4 in Section 3.2, "High-Level Steps for Configuring an Alternative Authentication Provider".

3.5.2 (Optional for Active Directory) Changing Group Name Attributes

If your Active Directory server uses a Group Name attribute other than the default value 'cn', you must to change it. If you do change this attribute, you must also change the settings for AllGroupsFilter and GroupFromNameFilter as shown in Table 3–11 (the example shows a group name stored in an attribute called AnOtherGroupAttribute).

Table 3–11 Changing Group Name Attribute

Attribute Name	Default Setting	Required New Setting
StaticGroupNameAttribute/DynamicGroupNameAttribute	cn	**AnOtherGroupAttribute**
AllGroupsFilter	(&(cn=*)(objectclass=person))	(&(**AnOtherGroupAttribute**=*)(objectclass=person))
GroupFromNameFilter	(&(cn=%u)(objectclass=person))	(&(**AnOtherGroupAttribute**=%u)(objectclass=person))

Make the changes in the Provider Specific tab, using Table 3–11 (substitute the AnOtherGroupAttribute setting with your own value). For more information about

how to display the Provider Specific tab, see Section 3.4.2, "Configuring Active Directory as the Authentication Provider".

3.6 Configuring the GUID Attribute In the Identity Store

If you configure an alternative authentication provider such as Oracle Internet Directory (OID LDAP) or Active Directory (AD), and you change the GUID attribute from its default value, then you must ensure that the value that you use in the identity store matches the changed value that you are using in the alternative authentication provider.

For example, if you are using OID LDAP and have changed the default value of the GUID attribute from `orclguid` to `newvalue`, you must set the value to `newvalue` in both the identity store and the authentication provider.

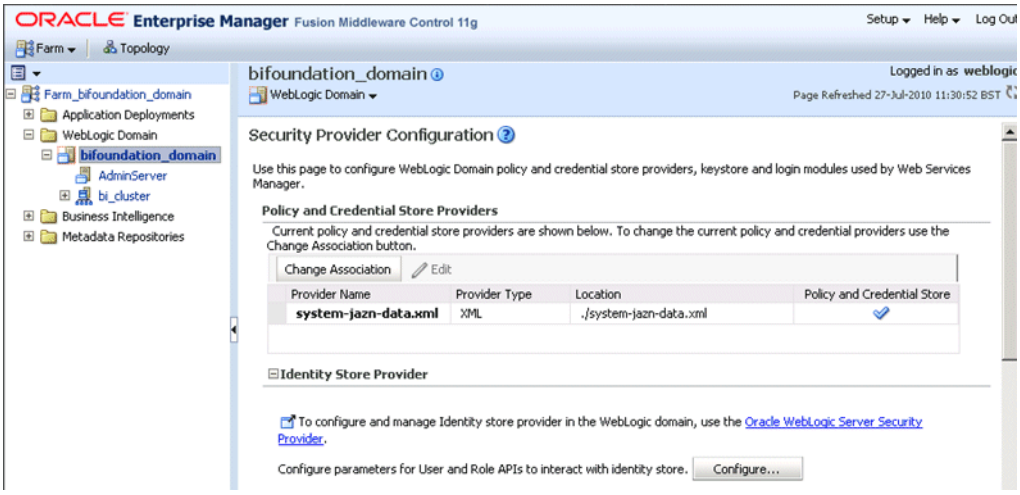
To configure the GUID attribute in the identity store:

1. Log in to Fusion Middleware Control.

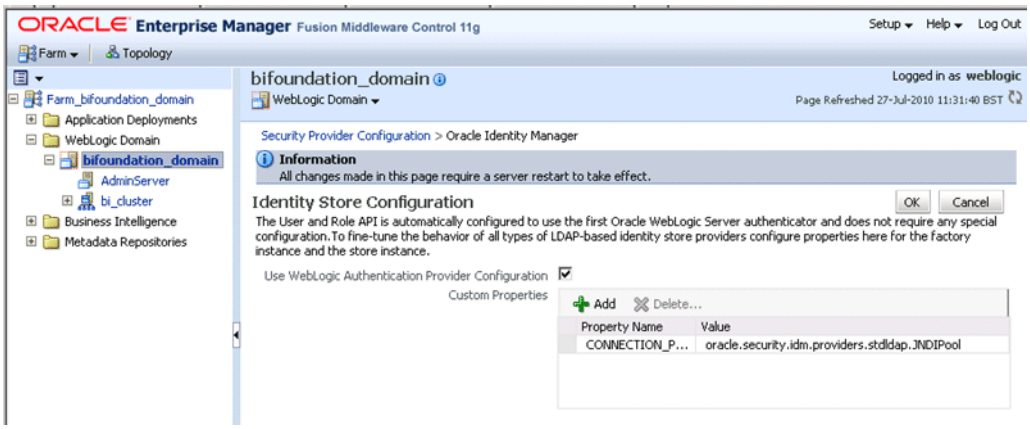
   For more information, see Section 1.6.2, "Using Oracle Fusion Middleware Control".

2. From the navigation pane expand the **WebLogic Domain** folder and select **bifoundation_domain**.

3. Right-click bifoundation_domain and select Security, then Security Provider Configuration to display the Security Provider Configuration page.



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



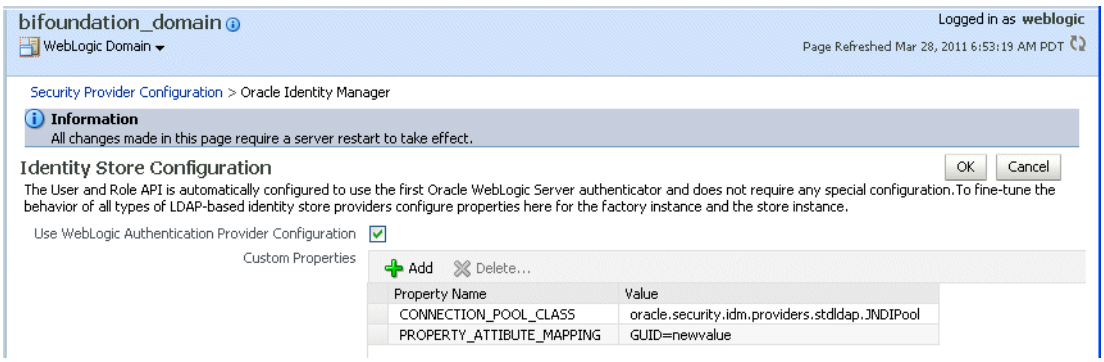
5. In the Custom Properties area, click **Add** to add the custom property described in Table 3–12.

Table 3–12 Custom Properties

Property Name	Value
PROPERTY_ ATTRIBUTE_ MAPPING	Specify the GUID attribute value that is set in the authentication provider. For example, if the GUID attribute is set to `newvalue` in the authentication provider, then set this value to `GUID=newvalue`.

Figure 3–6 shows an example set of Custom Properties including a new property called PROPERTY_ATTRIBUTE_MAPPING having a GUID attribute value set to `GUID=newvalue`.

Figure 3–6 Custom Properties - GUID Attribute



6. Click **OK** to save the changes.
7. Restart the Administration Server, any Managed servers, and Oracle BI components.

3.7 Configuring a New Trusted User (BISystemUser)

Oracle Business Intelligence uses a specific user for the configured authenticator for internal communication. If for example, you configure Oracle BI to use an alternative authentication provider (for example, Oracle Internet Directory, Active Directory), then you must create a new user (or select an existing user), in the alternative authentication provider to use for this purpose and give that user the required

permissions. You grant the chosen user the permission they need by making them a member of the preexisting BISystem application role. When configuring multiple authenticators (for more information, see Section 3.4.5), this user only needs to exist in one of the identity stores.

To create a new trusted user account with a user from the alternative authentication provider:

The credentials of the trusted user account are stored in the Credential Store under the system.user key. You must point the system.user key to a set of credentials available in your authentication provider (for example, Oracle Internet Directory, Active Directory).

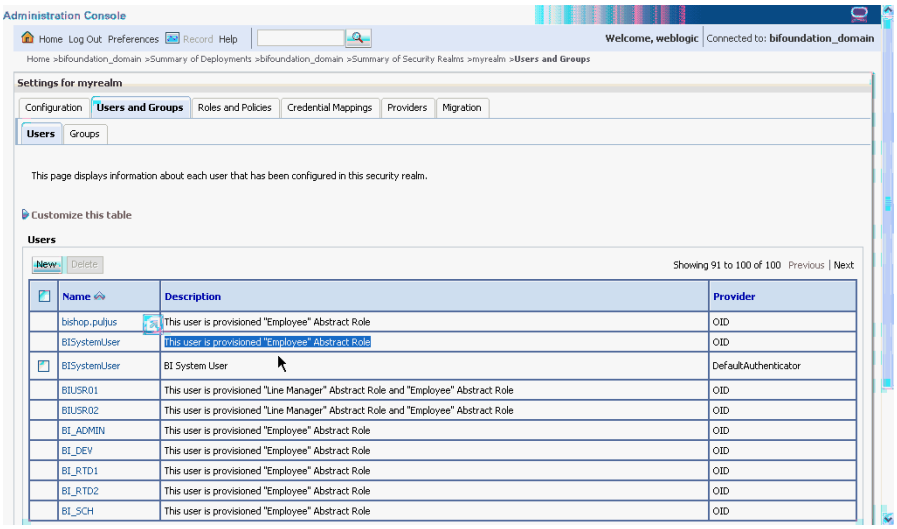
Whether you decide to use an existing user or create a new one, the process for configuring credentials in the system.user key is the same.

1. In the alternative authentication provider create, or identify 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.

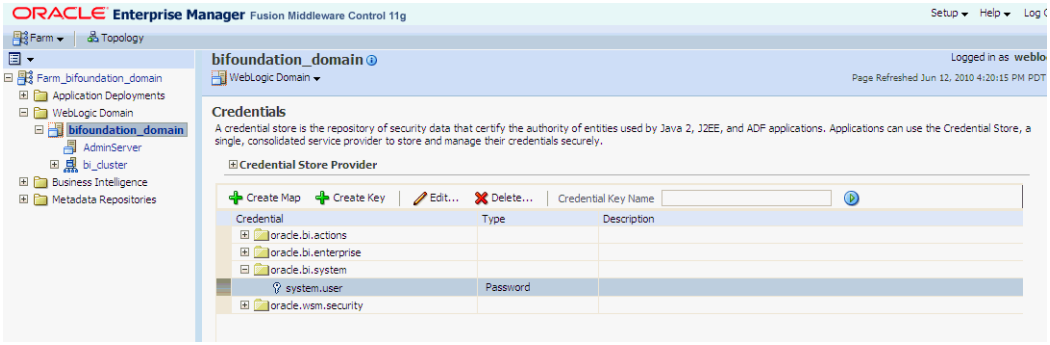
   When you are finished, the **Users** table in Oracle WebLogic Server Administration Console should resemble Figure 3–7 (the example given is for Oracle Internet Directory).

Figure 3–7 Users Table in Oracle WebLogic Server Administration Console

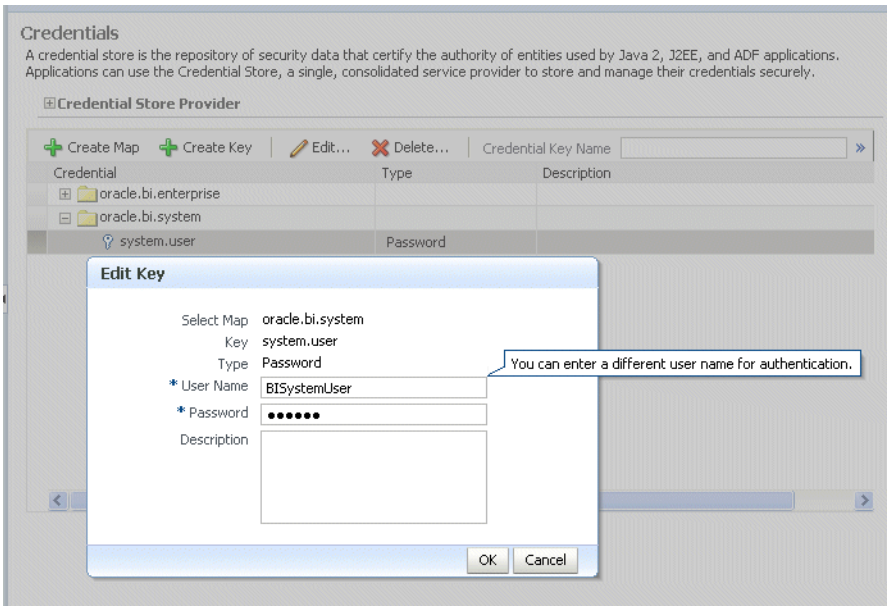


   Next add the trusted user's credentials to the **oracle.bi.system** credential map.

2. From the 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 the name you selected) in the **User Name** field. In the **Password** field, enter the trusted user's password that is contained in the authentication provider (for example, Oracle Internet Directory, Active Directory).



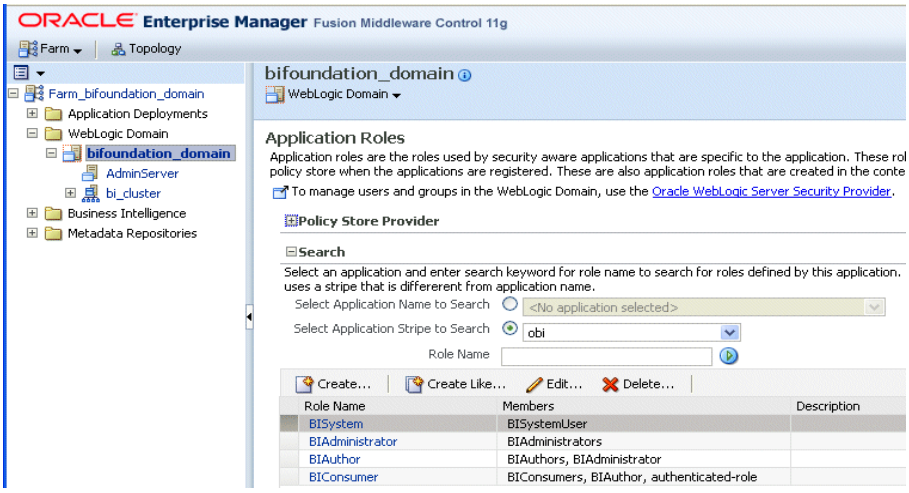
- Click **OK**.

  Next you must make the new trusted user a member of the BISystem application role.

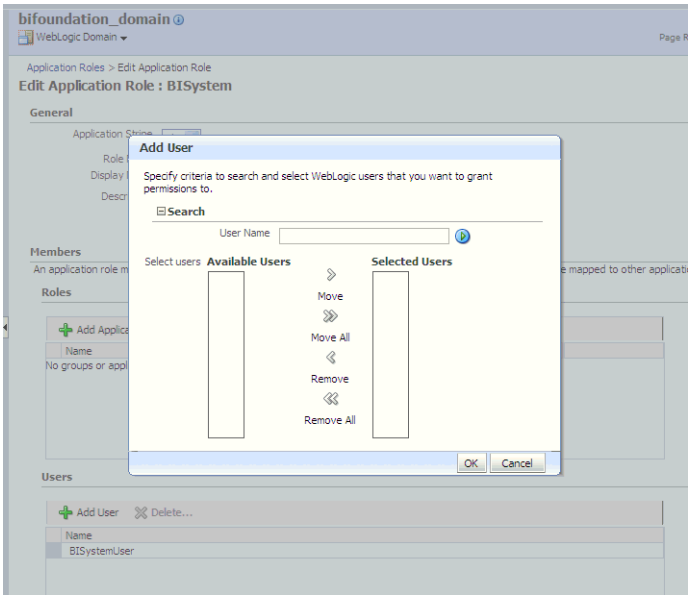
3. In the Fusion Middleware Control target navigation pane, go to the Oracle WebLogic Server domain in which Oracle Business Intelligence is installed. For example, bifoundation_domain.
4. Select Security and application roles from the WebLogic Domain menu, to display the **Application Roles** page.
5. Click the **Select Application Stripe to Search** radio button, and 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 Figure 3–8.

Figure 3–8 Application Roles Page in Fusion Middleware Control



6. Select the **BISystem** application role and click **Edit**.
7. In the **Edit Application Role** page, scroll down to the Users section and click **Add User**.
8. In the **Add User** dialog, click the arrow next to the **User Name** field to search for the trusted user created in the alternative authentication provider (for example, Oracle Internet Directory). Use the shuttle controls to move the trusted user name (**BISystemUser**) from the **Available Users** list to the **Selected Users** list.



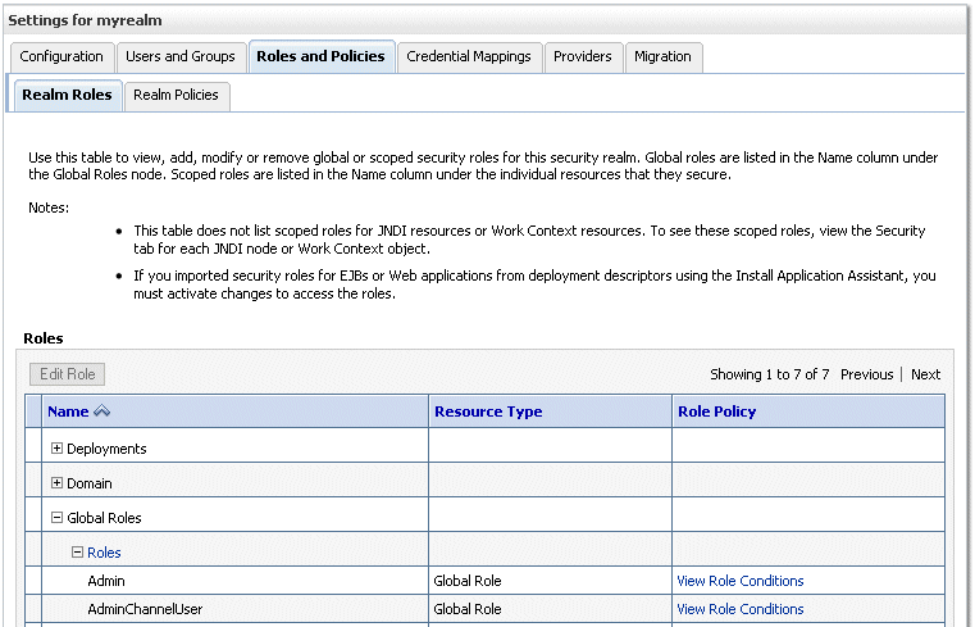
9. Click **OK**.

   The trusted user (**BISystemUser**) contained in the alternative authentication provider (for example, Oracle Internet Directory, or Active Directory), is now a member of the BISystem application role.

   The next stage of configuring the new system user is to ensure it is part of the WebLogic Server Global Admin role.

10. In the Oracle WebLogic Server Administration Console, click myrealm to display the Settings for <*Realm*> page, display the Roles and Policies tab.

11. In the list of roles, click on the plus sign to expand Global Roles, then Roles, then click **View Role Conditions** link for the Admin Role.



12. Add the new trusted user to the Global Admin Role.

    Ensure the conditions specified will match your user, either directly, or by virtue of a group it belongs to (for example, condition may be User = BISystemUser or Group=Administrators).

13. Click **Save**.

14. If you change the trusted user name to a value other than BISystemUser, you must also change the equivalent user name for Oracle Business Intelligence Publisher JMS Modules.

    Oracle Business Intelligence Publisher JMS modules use BISystemUser by default, therefore if you have changed your trusted user account name to a value other than BISystemUser, you must also change the user name for JMS Modules to the value of the new trusted user as follows:

    a. Log in to Oracle WebLogic Server Administration Console, and click **Lock & Edit** in the Change Center.

       For more information, see Section 1.6.1, "Using Oracle WebLogic Server Administration Console".

    b. Select **Services, Messaging**, and **JMS Modules** from the left pane.

    c. Select **BipJmsResource**.

    d. Go to the Security tab, and display the Policies sub-tab.

    e. Replace BISystemUser with the name of the new trusted user.

    f. Click **Activate Changes**.

15. Start the Managed Servers.

    When you have changed the system user credentials in this way, restart the BI Server and Presentation Services as follows:

    a. Log in to Fusion Middleware Control.

For information, see Section 1.6.2, "Using Oracle Fusion Middleware Control".

b. Go to the **Availability** page and display the **Processes** tab.

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

c. Click **Restart All**.

The new trusted user from the authentication provider (for example, Oracle Internet Directory, Active Directory), is configured for Oracle Business Intelligence.

3.8 Refreshing User GUIDs

In Oracle Business Intelligence 11*g* Release 1 (11.1.1), users are recognized by their global unique identifiers (GUIDs), not by their names. GUIDs are identifiers that are unique for a given user. Using GUIDs to identify users provides a higher level of security because it ensures that data and metadata are uniquely secured for a specific user, independent of the user name.

GUID refresh (also called GUID synchronization or GUID regeneration) updates any metadata references to user GUIDs in the Oracle BI repository and Oracle BI Presentation Catalog. During the GUID refresh process, each user name is looked up in the identity store. Then, all metadata references to the GUID associated with that user name are replaced with the GUID in the identity store.

GUID refresh might be required when Oracle Business Intelligence is reassociated with an identity store that has different GUIDs for the same users. This situation might occur when reassociating Oracle Business Intelligence with a different type of identity store, or when moving from test to production if a different identity store is used in production, and should be a rare event.

Note that if Oracle best practices are not observed and Oracle Business Intelligence repository data is migrated between systems that have different GUIDs for the same users, GUID refresh is required for the system to function. This is not a recommended practice, because it raises the risk that data and metadata secured to one user (for example, John Smith, who left the company two weeks ago) becomes accessible to another user (for example, John Smith, who joined last week). Using application roles wherever possible and using GUIDs consistently across the full development production lifecycle prevents this problem from occurring.

To refresh user GUIDs:

This task requires that you manually edit the configuration files to instruct the Oracle BI Server and Presentation Services 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*.

To refresh user GUIDs, perform the following steps on APPHOST1 and APPHOST2. Note that GUID refresh must occur with only one node operating at a time.

1. Stop Oracle BI Server and Presentation Services on all nodes except where you are refreshing the user GUIDs. For example:

```
cd ORACLE_HOME/admin/instancen/bin
./opmnctl stopproc ias-component=coreapplication_obips1
./opmnctl stopproc ias-component=coreapplicaiton_obis1
```

2. 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_obis*n*

   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.

3. Update the Catalog element in instanceconfig.xml:

   a. Open instanceconfig.xml for editing at:

      ORACLE_INSTANCE/config/OracleBIPresentationServicesComponent/coreapplication_obips*n*

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

      ```
      <Catalog>
      <UpgradeAndExit>false</UpgradeAndExit>
      <UpdateAccountGUIDs>UpdateAndExit</UpdateAccountGUIDs>
      </Catalog>
      ```

   c. Save and close the file.

4. Restart the Oracle BI Server and Presentation Services using opmnctl:

   ```
   cd ORACLE_HOME/admin/instancen/bin
   ./opmnctl stopproc ias-component=coreapplication_obips1
   ./opmnctl stopproc ias-component=coreapplicaiton_obis1
   ./opmnctl startproc ias-component=coreapplicaiton_obis1
   ```

   After you confirm that the Oracle BI Server is running, then start Presentation Services:

   ```
   ./opmnctl startproc ias-component=coreapplicaiton_obips1
   ```

5. 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.

6. Update the Catalog element in instanceconfig.xml to remove the UpdateAccount GUIDs entry.

7. Restart the Oracle Business Intelligence system components again using opmnctl:

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

3.9 Configuring Oracle Internet Directory as the Policy Store and the Credential Store

To re-configure Oracle Business Intelligence to use Oracle Internet Directory (OID LDAP) as a Credential Store and a Policy Store Provider, follow the steps in

"Reassociating the OPSS Security Store" in *Oracle Fusion Middleware Application Security Guide*.

Notes

- The only LDAP server supported for this purpose in this release is Oracle Internet Directory. For more information, see "System Requirements and Certification".
- The prerequisites for using an LDAP-based credential store are the same as for using an LDAP-based policy store. For more information, see "Using an LDAP-Based OPSS Security Store" in *Oracle Fusion Middleware Application Security Guide*
- Oracle Entitlements Server Basic (OES Basic) replaces the embedded authorization engine within Oracle Platform Security Services (OPSS) and is used to define, enforce and audit basic Role Based Access Control and Java2/JAAS permission based authorization policies. A license of Oracle Entitlements Server Basic is included and available for use only with Oracle products that lists this component in their respective licensing documentation.

  Nothing changes visibly, and references to OPSS are still applicable but will be replaced over time by references to OES Basic.

  If you are able to re-associate your Policy Store to use OID you can optionally use the OES user interface to manage the Policy Store (subject to the terms of the OES Basic license), rather than Fusion Middleware Control.

  For more information about Oracle Entitlements Server Basic, see *Oracle Fusion Middleware Licensing Information*.

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 1.7, "Detailed List of Steps for Setting Up Security In Oracle Business Intelligence".

This chapter contains the following topics:

- Section 4.1, "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"
- Section 4.5, "Configuring Custom SSO Environments"
- Section 4.6, "Enabling SSO Authentication Using Fusion Middleware Control"
- Section 4.7, "Enabling the Online Catalog Manager to Connect"

> **Note:** Oracle recommends using Oracle Access Manager as an enterprise-level SSO authentication provider with Oracle Fusion Middleware 11*g*. Section 4.2, Section 4.3, and Section 4.4 assume that Oracle Access Manager is the SSO authentication provider. Section 4.5 references alternative authentication providers in custom SSO environment solutions.
>
> For more information about configuring and managing Oracle Access Manager with Oracle Fusion Middleware, see "Introduction to Single Sign-On in Oracle Fusion Middleware" in *Oracle Fusion Middleware Application Security Guide*.
>
> For more information about supported SSO providers, see "System Requirements and Certification".

4.1 SSO Configuration Tasks for Oracle Business Intelligence

Table 4–1 contains SSO 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 Oracle Access Manager as 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 Application Security Guide*
Configure the HTTP proxy.	Configure the Web proxy to forward requests from Presentation Services to the SSO provider.	"Configuring Single Sign-On in Oracle Fusion Middleware" in *Oracle Fusion Middleware Application 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 3.4, "Configuring Alternative Authentication Providers" *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 Oracle Access Manager as a New Identity Asserter for Oracle WebLogic Server" Section 3.4, "Configuring Alternative Authentication Providers" *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 3.7, "Configuring a New Trusted User (BISystemUser)"
Refresh the user and group GUIDs.	Refresh the GUIDs of users and groups which migrated from the original identity store to the new identity store (authentication source).	Section 3.8, "Refreshing User GUIDs"
Configure custom SSO solutions.	Configure alternative custom SSO solutions to protect the Oracle Business Intelligence URL entry points.	Section 4.5, "Configuring Custom SSO Environments"
Enable Oracle Business Intelligence to accept SSO authentication.	Enable the SSO provider configured to work with Oracle Business Intelligence using Fusion Middleware Control.	Section 4.6, "Enabling SSO Authentication Using Fusion Middleware Control"

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. 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 Application 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 Presentation Services where this name is extracted. Next a session with the BI Server is established using the impersonation feature (a connection string between the Oracle BI Presentation Server and the BI Server using credentials that act on behalf of a user being impersonated).

After successfully logging in using SSO, users are still required to have the oracle.bi.server.manageRepositories permission to log in to the 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 are 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

This section describes how Oracle Access Manager authentication provider works with Oracle WebLogic Server using Identity Asserter for single sign-on, providing 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 a suitable token and creates a WebLogic-authenticated session. It also provides single sign-on between WebGate 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.

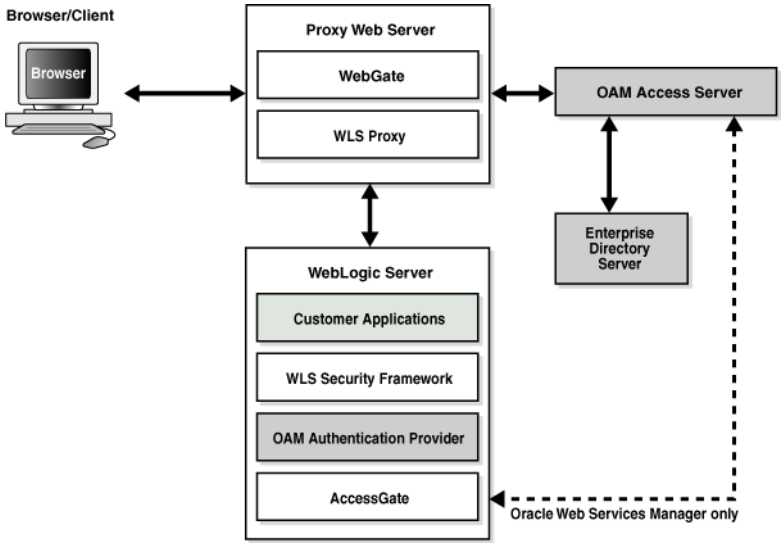
After the authentication provider for Oracle Access Manager is configured as the Identity Asserter for single sign-on, the Web resources are protected. Perimeter authentication is performed by WebGate on the Web tier and by the appropriate token to assert the identity of users who attempt access to the 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 a token 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 token 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, Presentation Services operates as if the incoming Web request is from a user authenticated by the SSO solution. Presentation Services next creates a connection to the BI Server using the impersonation feature and establishes the connection to the 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 Presentation Services. 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 WebLogic Server, see "Configuring Single Sign-On in Oracle Fusion Middleware" in *Oracle Fusion Middleware Application Security Guide*. For more information about managing Oracle Access Manager, see *Oracle Fusion Middleware Administrator's Guide for Oracle Access Manager*.

For information about how to configure Oracle BI Publisher to use Oracle Access Manager as the SSO authentication provider, see "Configuring BI Publisher to Use Oracle Access Manager (OAM) Single Sign-On" in *Oracle Fusion Middleware Administrator's and Developer's Guide for Oracle Business Intelligence Publisher*

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 Presentation Services 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 Oracle Access Manager as 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 4.6, "Enabling SSO Authentication Using Fusion Middleware Control".

4.4.1 Configuring a New Authenticator for Oracle WebLogic Server

After installing Oracle Business Intelligence, the Oracle WebLogic Server embedded LDAP server is the default authentication source (identity store). To use a new identity store (for example, OID), as the main authentication source, you must configure the Oracle WebLogic Server domain (where Oracle Business Intelligence is installed).

For more information about configuring authentication providers in Oracle WebLogic Server, see *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 and click **Lock & Edit** in the Change Center.

   For more information, see Section 1.6.1, "Using Oracle WebLogic Server Administration Console".

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.

   Complete the fields as follows:

   - **Name**: *OID Provider*, or a name of your choosing.
   - **Type**: OracleInternetDirectoryAuthenticator
   - Click **OK** to save the changes and display the authentication providers list updated with the new authentication provider.

5. Click the newly added authenticator in the **authentication providers** table.

6. Navigate to **Settings**, then select the **Configuration\Common** tab:

   - Select **SUFFICIENT** from the **Control Flag** list.
   - Click **Save**.

7. Display the **Provider Specific** tab and specify the following settings using appropriate values for your environment:

Section Name	Field Name	Description
Connection	Host	The LDAP host name. For example, *<localhost>*.
Connection	Port	The LDAP host listening port number. For example, 6050.
Connection	Principal	The distinguished name (DN) of the user that connects to the LDAP server. For example, cn=orcladmin.
Connection	Credential	The password for the LDAP administrative user entered as the Principal.
Users	User Base DN	The base distinguished name (DN) of the LDAP server tree that contains users. For example, use the same value as in Oracle Access Manager.
Users	All Users Filter	The LDAP search filter. For example, (&(uid=*) (objectclass=person)). The asterix (*) filters for all users. Click More Info... for details.
Users	User From Name Filter	The LDAP search filter. Click More Info... for details.

Section Name	Field Name	Description
Users	User Name Attribute	The attribute that you want to use to authenticate (for example, cn, uid, or mail). Set as the default attribute for user name in the directory server. For example, *uid* . **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 3.5.1, "Configuring the User Name Attribute In the Identity Store".
Groups	Group Base DN	The base distinguished name (DN) of the LDAP server tree that contains groups (same as User Base DN).
General	GUID attribute	The attribute used to define object GUIDs in LDAP. orclguid **Note**: You should not normally change this default value, however, if you do, you must also specify the changed value in Fusion Middleware Control, as described in the task Section 3.6, "Configuring the GUID Attribute In the Identity Store".

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

8. Click **Save**.
9. Perform the following steps to set up the default authenticator for use with the Identity Asserter:
   a. At the main **Settings for myrealm** page, display the **Providers** tab, then display the **Authentication** sub-tab, then select **DefaultAuthenticator** to display its configuration page.
   b. Display the **Configuration\Common** tab and select 'SUFFICIENT' from the **Control Flag** list.

      For more information, see Section 3.4.6, "Setting the JAAS Control Flag Option".
   c. Click **Save**.
10. Perform the following steps to reorder Providers:
    a. Display the **Providers** tab.
    b. Click **Reorder** to display the **Reorder Authentication Providers** page
    c. Select a provider name and use the arrow buttons to order the list of providers as follows:
       - OID Authenticator (SUFFICIENT)
       - OAM Identity Asserter (REQUIRED)
       - Default Authenticator (SUFFICIENT)

d. Click **OK** to save your changes.

11. In the Change Center, click **Activate Changes**.
12. Restart Oracle WebLogic Server.

4.4.2 Configuring Oracle Access Manager as 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 Oracle Access Manager as the new asserter for Oracle WebLogic Server:

1. Log in to Oracle WebLogic Server Administration Console.

   For more information, see Section 1.6.1, "Using 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.
4. Click **OK**.
5. Click **Save**.
6. In the **Providers** tab, perform the following steps to reorder **Providers**:
   a. Click **Reorder**
   b. In 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.
7. In the Change Center, click **Activate Changes**.
8. 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.

9. Use Fusion Middleware Control to enable SSO authentication.

   For more information, see Section 4.6, "Enabling SSO Authentication Using Fusion Middleware Control".

4.5 Configuring Custom SSO Environments

For information about configuring Oracle Business Intelligence to participate in custom SSO environments (for example, setting up SSO using Active Directory or SiteMinder), see articles 1287479.1 and 1274953.1 on My Oracle Support at:

https://support.oracle.com

4.6 Enabling SSO Authentication Using Fusion Middleware Control

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

To enable Oracle Business Intelligence to use SSO authentication:

1. Log in to Fusion Middleware Control.

   For information, see Section 1.6.2, "Using Oracle Fusion Middleware Control".

2. Go to the **Security** page and display the **SSO** tab.

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

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

4. Select **Enable SSO**.

   When selected, this checkbox enables SSO to be the method of authentication into Oracle Business Intelligence. The appropriate form of SSO is determined by the configuration settings made for the chosen SSO provider.

5. Select the configured SSO provider from the list.

   The SSO provider list becomes active when you select the **Enable SSO** checkbox.

6. If required, enter logon and logoff URLs for the configured SSO provider.

   The logoff URL (specified by the SSO provider) must be outside the domain and port that the SSO provider protects, because the system does not log users out.

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

8. 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*.

4.7 Enabling the Online Catalog Manager to Connect

The online Catalog Manager might fail to connect to Oracle BI Presentation Services when the HTTP Web server for Oracle BI is enabled for SSO. When you enable SSO in Section 4.6, "Enabling SSO Authentication Using Fusion Middleware Control", the Oracle Business Intelligence URL http://*hostname:port_number*/analytics becomes protected, and you must point the online Catalog Manager to the URL http://*hostname:port_number*/analytics-ws instead. The URL should remain unprotected. It is configured only to accept SOAP access as used by Oracle BI Publisher, Oracle BI Add-in for Microsoft Office and the online Catalog Manager.

To log in to the online Catalog Manager when SSO is enabled you must change the URL Suffix to point to analytics-ws/saw.dll.

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 1.7, "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, "What is 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 5.3.3, "Manually Configuring WebLogic to Use the 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 5.3, "Configuring SSL Communication Between Components"

5.2 What is 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 he or she claims to be. This section contains the following topics:

- Section 5.2.1, "Using SSL in Oracle Business Intelligence"
- Section 5.2.2, "Creating Certificates and Keys in Oracle Business Intelligence"
- Section 5.2.3, "What is the Credential Store?"

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

5.2.1 Using SSL in Oracle Business Intelligence

Oracle Business Intelligence components communicate with each other using TCP/IP by default. 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.

SSL is configured 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 SSL:

- 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 5.5, "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 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 What is the Credential Store?

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 use are .der and .pem.

5.3 Configuring SSL Communication Between Components

This section explains how to configure SSL communication between components using Fusion Middleware Control (Oracle recommends this method), or manually, and contains the following topics:

- Section 5.3.1, "Configuring SSL Communication Between Components Using Fusion Middleware Control and Oracle WebLogic Server Administration Console"
- Section 5.3.2, "Configuring SSL for the SMTP Server Using Fusion Middleware Control"
- Section 5.3.3, "Manually Configuring WebLogic to Use the HTTPS Protocol"

- Section 5.3.4, "Manually Configuring SSL Communication Between Components Using the MBean Browser"

5.3.1 Configuring SSL Communication Between Components Using Fusion Middleware Control and Oracle WebLogic Server Administration Console

This section explains how you can set up SSL communication between components using Fusion Middleware Control and Oracle WebLogic Server Administration Console.

Note: This method does not verify peers (that is, two-way SSL between BI System components) and always assumes that the certificate used by the Weblogic servers is signed using the demo CA certificate.

To configure SSL communication between components using Fusion Middleware Control and Oracle WebLogic Server Administration Console:

You must configure SSL for the Web server, before you enable SSL communication between components. For more information, see Section 5.3.3, "Manually Configuring WebLogic to Use the HTTPS Protocol".

1. Log in to Fusion Middleware Control.

   For more information, see Section 1.6.2, "Using Oracle Fusion Middleware Control".

2. Go to the **Security** page and display the **SSL** tab.

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

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

4. If the **Use SSL for Middle-Tier Communications** checkbox is dimmed:

   If this checkbox is dimmed with a warning (WebLogic SSL Listen Port is not enabled. SSL for Oracle BI cannot be enabled until it is), one of the following statements might be true:

   - If the **SSL Listen Port Enabled** checkbox has been enabled in Oracle WebLogic Server Administration Console but the Administration Server, and any Managed Servers (depending on install type) might need to be restarted:

     Restart the Administration Server and any Managed Servers using "Starting and Stopping the Oracle Business Intelligence Components" in *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*.

   - If the **SSL Listen Port Enabled** checkbox has not been enabled in Oracle WebLogic Server Administration Console:

     Complete the following steps:

     a. Click **Go to the Oracle WebLogic Server Administrator Console to configure SSL** to configure the SSL listen port in Oracle WebLogic Server.

        The Oracle WebLogic Server Administration Console login page displays.

     b. Log in to Oracle WebLogic Server Administration Console.

        For more information, see Section 1.6.1, "Using Oracle WebLogic Server Administration Console".

c. Go to the Settings for AdminServer page, General tab.

d. In the Change Center click **Lock & Edit**.

e. Clear the **Listen Port Enabled** checkbox.

f. Select the **SSL Listen Port Enabled** checkbox.

g. In the Change Center, click **Activate Changes**.

h. Restart the Administration Server and any Managed Servers.

Note: Installation of Managed servers depends on the installation type. A Simple installation installs an Administration Server, an Enterprise installation installs an Administration Server and a Managed Server, and a Clustered installation installs an Administration Server and one or more Managed Servers.

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*.

i. Display the Fusion Middleware Control **SSL** tab.

5. If the **Use SSL for Middle-Tier Communications** checkbox is available, select it.

   When selected, this checkbox enables SSL to be the method of communication between Oracle Business Intelligence components.

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

7. 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*.

8. Click **View SSL report to verify internal SSL communications status** link to view SSL status.

   Alternatively, see Section 5.3.4.6, "Confirming SSL Status Using the MBean Browser".

   This link is only visible if SSL is enabled.

5.3.2 Configuring SSL for the SMTP Server Using Fusion Middleware Control

The server certificate from the SMTP server must be obtained.

To configure SSL for the SMTP server using Fusion Middleware Control:

1. Login to Fusion Middleware Control.

   For more information, see Section 1.6.2, "Using Oracle Fusion Middleware Control".

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

3. Display the **Mail** tab of the **Deployment** page.

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

4. Lock the configuring by clicking **Lock and Edit Configuration**.

5. Complete the fields under **Secure Socket Layer (SSL)** as follows:
   - Select an option from the **Connection Security** drop down list. Other fields may 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 verification 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.
6. Click **Apply**, then **Activate Changes**.

5.3.3 Manually Configuring WebLogic to Use the HTTPS Protocol

The WebLogic listening port must be configured to use HTTPS (for all Managed servers in a cluster) before enabling SSL communication for Oracle Business Intelligence. For more 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 the following files:

For Linux:

MW_HOME/user/projects/domains/bifoundation_domain/bin/setDomainEnv.sh

For Windows:

MW_HOME\user\projects\domains\bifoundation_domain\bin\setDomainEnv.cmd

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:

For Linux (all on one line):

```
JAVA_OPTIONS="${JAVA_OPTIONS} -Djavax.net.ssl.trustStore=MW_Home/wlsserver_
10.3/server/lib/DemoTrust.jks -Djavax.net.ssl.trustStorePassword="
```

For Windows (all on one line):

```
set JAVA_OPTIONS=%JAVA_OPTIONS% -Djavax.net.ssl.trustStore="MW_Home\wlserver_
10.3\server\lib\DemoTrust.jks" -Djavax.net.ssl.trustStorePassword=""
```

If this step is omitted then login will 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 any attempts to log in 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
```

5.3.4 Manually Configuring SSL Communication Between Components Using the MBean Browser

This section explains how to manually configure SSL communication between components using the MBean Browser. Oracle recommends that you use Section 5.3.1, "Configuring SSL Communication Between Components Using Fusion Middleware Control and Oracle WebLogic Server Administration Console".

Table 5–2 contains the tasks for manually configuring SSL communication between components anmd includes using the MBean Browser 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 5.3.3, "Manually Configuring WebLogic to Use the HTTPS Protocol".

Table 5–2 Task Map: Manually Configuring SSL Communication Between Components Using the MBean Browser

Task	Description	For Information
Lock the configuration.	Use the BIDomain MBean to lock the domain configuration before making changes.	Section 5.3.4.1, "Locking the Configuration"
Generate the SSL certificate.	Use the BIDomain.BIInstance.SecurityConfiguration MBean to generate the SSL certificate.	Section 5.3.4.2, "Generating the SSL Certificates"
Commit the SSL configuration changes.	Use the BIDomain MBean to commit the SSL configuration changes.	Section 5.3.4.3, "Committing the SSL Configuration Changes"
Verify SSL certificates in credential store.	Verify that the SSL certificates are saved in the credential store.	Section 5.3.4.4, "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 5.3.4.5, "Enabling the SSL Configuration"
Confirm that SSL communication is enabled between components.	Run the SSL report to confirm status.	Section 5.3.4.6, "Confirming SSL Status Using the MBean Browser"

Table 5–2 (Cont.) Task Map: Manually Configuring SSL Communication Between Components Using the MBean Browser

Task	Description	For Information
Configure SSL communication for the mail server.	Configure SSL communication for the mail server.	Section 5.3.2, "Configuring SSL for the SMTP Server Using Fusion Middleware Control"
Update expired SSL certificates.	Update expired SSL certificates and replace with new ones.	Section 5.3.4.7, "Updating Expired SSL Certificates"

You can manually configure internal SSL communication between components using Oracle Business Intelligence managed beans (MBeans). An 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 manually configure SSL communication between Oracle Business Intelligence components. The System MBean Browser is accessed from the Oracle WebLogic Server domain where Oracle Business Intelligence is installed in Fusion Middleware Control. For example, bifoundation_domain.

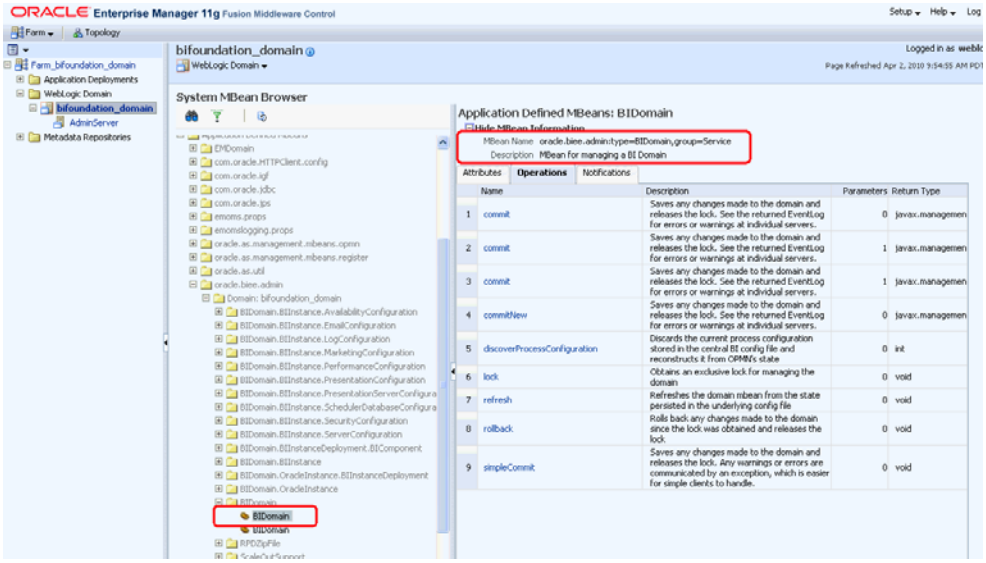
For more information about using and navigating within Fusion Middleware Control, see "Navigating Within Fusion Middleware" Control in *Oracle Fusion Middleware Administrator's Guide*.

5.3.4.1 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 is to generate the SSL certificates. For more information, see Section 5.3.4.2, "Generating the SSL Certificates".

5.3.4.2 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 (certificate authority) 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.

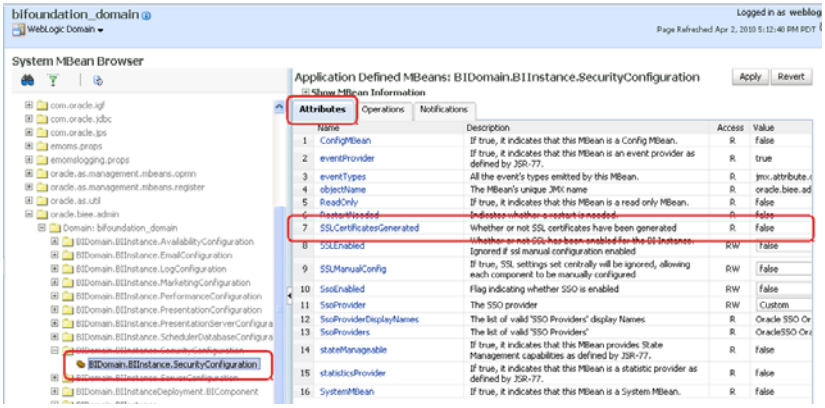
   For information, see Section 5.3.4.1, "Locking the Configuration".
2. In Fusion Middleware Control target navigation pane, expand the farm, then expand **WebLogic Domain**, and select **bifoundation_domain**.
3. Display the WebLogic Domain menu, and select **System MBean Browser**.

   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.

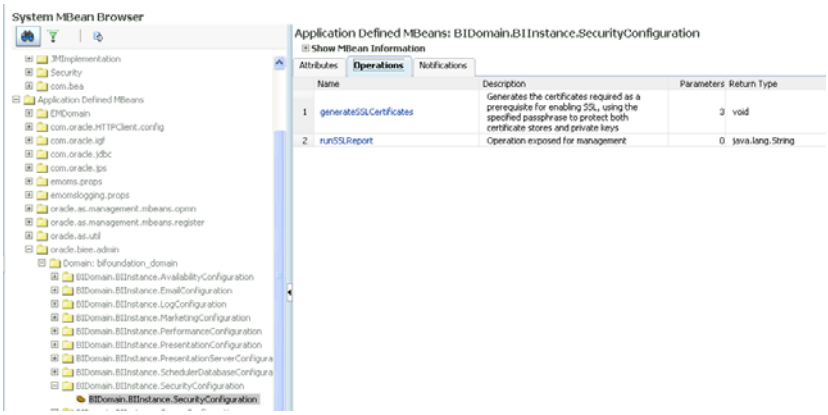
   The BIDomain.BIInstance.SecurityConfiguration MBean is displayed.
6. Select the BIDomain.BIInstance.SecurityConfiguration MBean.

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.

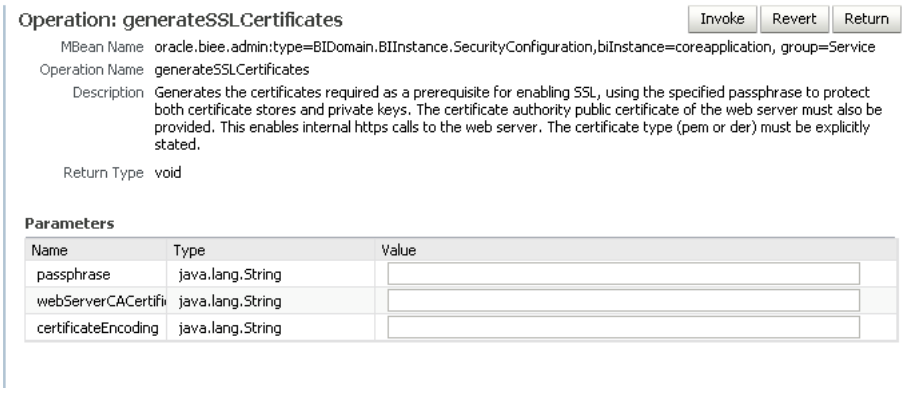


8. Select the **Operations** tab



9. Select **generateSSLCertificates** operation.

   The parameters for the generateSSLCertificates attribute for the BIDomain.BIInstance.SecurityConfiguration MBean appears.



10. 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 Certificate Authority (CA) root certificate for the CA used to sign the web server's certificate. Do not enter the individual web server certificate. Supported types are .der. and .pem. For Oracle WebLogic Server default demonstration certificate authority, enter *<MW_HOME>*/wlserver_10.3/server/lib/CertGenCA.der.

  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 CA 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`

11. 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 5.3.4.3, "Committing the SSL Configuration Changes".

5.3.4.3 Committing the SSL Configuration Changes

You commit the SSL configuration changes using the BIDomain MBean.

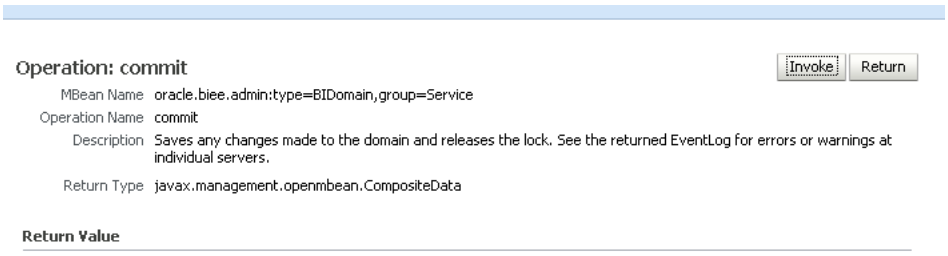
Note: You must configure SSL for the Web server before enabling SSL for Oracle Business Intelligence. For more information, see Section 5.3.3, "Manually Configuring WebLogic to Use the HTTPS Protocol".

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 5.3.4.1, "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 5.3.4.4, "Verifying the SSL Credentials in the Credential Store".

5.3.4.3.1 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
```

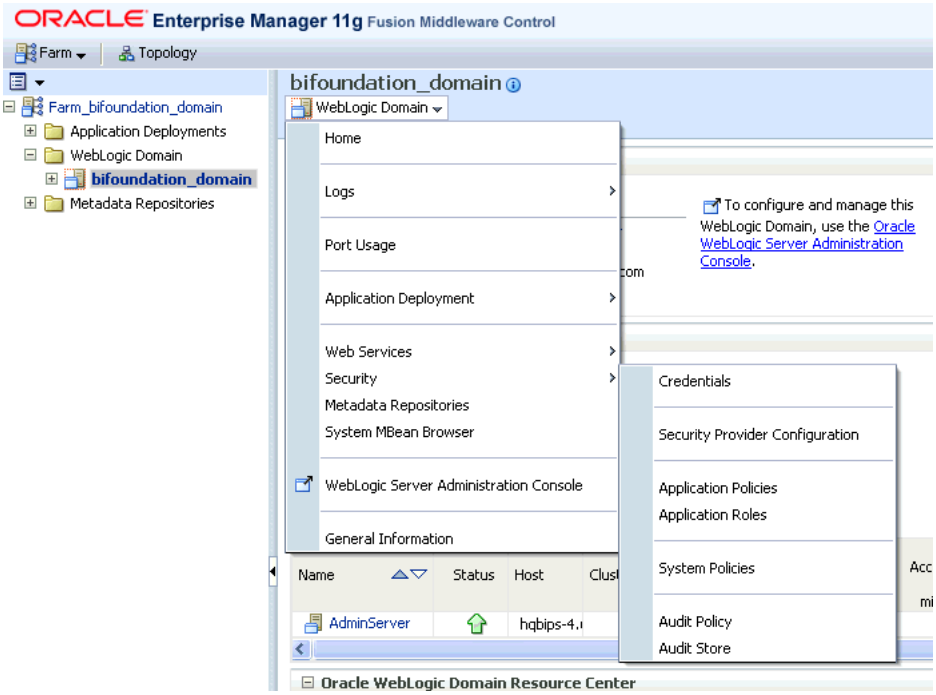
This message indicates that SSL has not been enabled on the Oracle WebLogic Server Managed Servers, which is a prerequisite step. For more information, see Section 5.3.3, "Manually Configuring WebLogic to Use the HTTPS Protocol". After this prerequisite is completed you can repeat the commit operation.

5.3.4.4 Verifying the SSL Credentials in the Credential Store

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

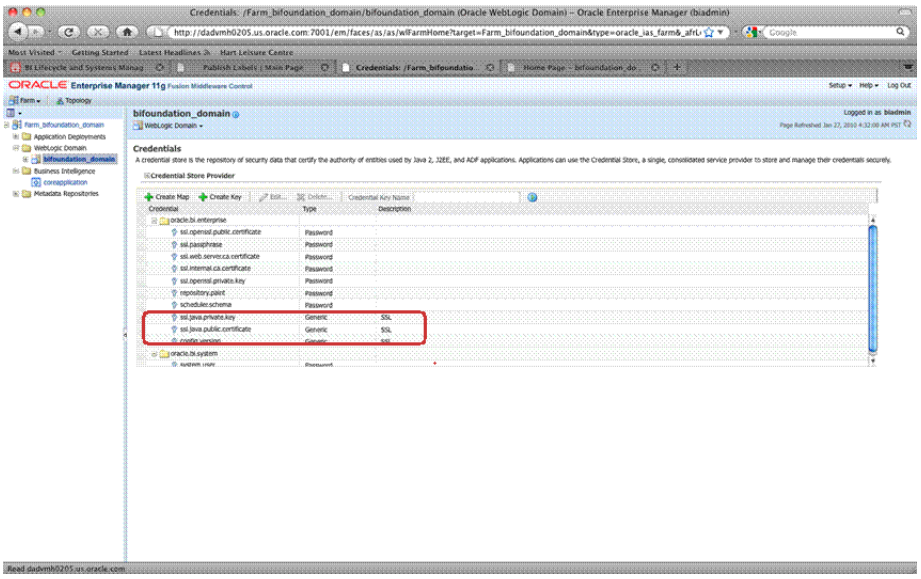
To verify the SSL credentials in the credential store:

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 point 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 5.3.4.5, "Enabling the SSL Configuration".

5.3.4.5 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 5.3.3, "Manually Configuring WebLogic to Use the HTTPS Protocol".

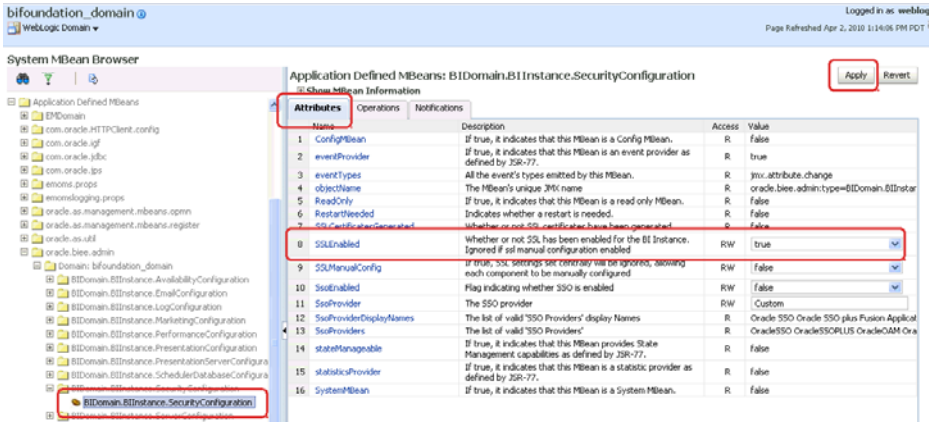
2. Lock the configuration.

   For information, see Section 5.3.4.1, "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 5.3.4.2, "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 5.3.3, "Manually Configuring WebLogic to Use the HTTPS Protocol".



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

   For information, see Section 5.3.4.3, "Committing 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 more information, see "Starting and Stopping Oracle Business Intelligence System Components" in *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*.

5.3.4.6 Confirming SSL Status Using the MBean Browser

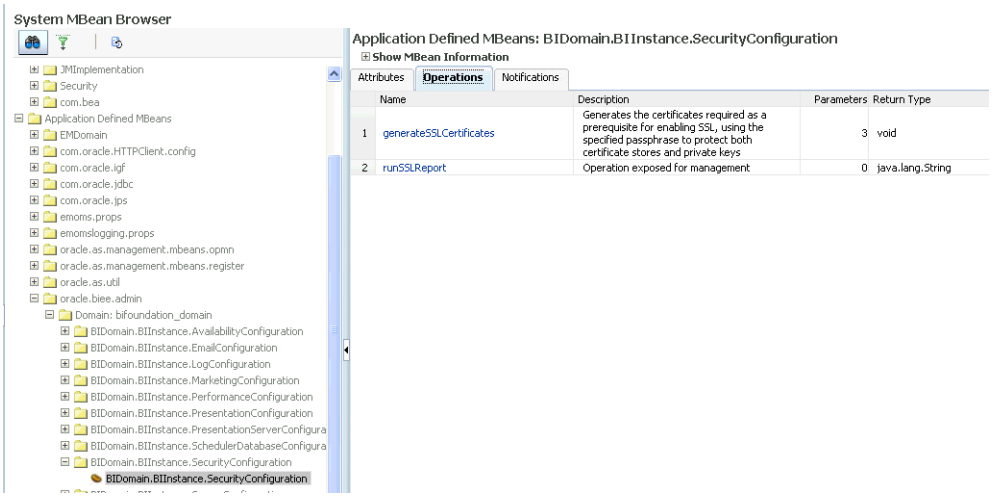
You can run a SSL report using the BIDomain.BIInstance.SecurityConfiguration MBean to verify that SSL communication is operating between components.

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 5.3.4.2, "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.
```

5.3.4.7 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 5.3.4.6, "Confirming SSL Status Using the MBean Browser". 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 5.3.4.2, "Generating the SSL Certificates" and restart the Oracle Business Intelligence components.

5.4 Additional SSL Configuration Options

Additional configuration options are required for Oracle Business Intelligence components and tools as follows:

- Section 5.4.1, "Using SASchInvoke when BI Scheduler is SSL-Enabled"
- Section 5.4.2, "Configuring Oracle BI Job Manager"
- Section 5.4.3, "Enabling the Online Catalog Manager to Connect"
- Section 5.4.4, "Configuring the Oracle BI Administration Tool to Communicate Over SSL"
- Section 5.4.5, "Configuring an ODBC DSN for Remote Client Access"
- Section 5.4.6, "Configuring SSL when Using Multiple Authenticators"

5.4.1 Using SASchInvoke when BI Scheduler is SSL-Enabled

When the BI Scheduler is enabled for communication over SSL, you can invoke the BI Scheduler using the SASchInvoke command line utility.

Use the following syntax to run the SASchInvoke command:

```
SASchInvoke -u <Admin Name>  (-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>] ] ]
```

The command will prompt you to enter the administrator password.

5.4.2 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.
3. Click **OK** to exit.
4. If BI Scheduler has been set to "Require Client Certificate", then you must set Key Store and Key Store Password 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 5.3.4.2, "Generating the SSL Certificates"

5. 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.
6. 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.
7. In the **Trust Store Password** text box, enter the password of the keystore entered in Step 5.
8. Copy the keystore and trust store files to the locations specified in the parameters above.

5.4.3 Enabling the Online Catalog Manager to Connect

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 (for example, JRocket) that is specified by the system JAVA_HOME variable.

To enable the online Catalog Manager to connect:

1. Navigate to Java's default trust store located at *MW_HOME/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.

5.4.4 Configuring the Oracle BI Administration Tool to Communicate Over SSL

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

To configure the Administration Tool to communicate over SSL:

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

4. Enter the repository user name and password.

   The Administration Tool is now connected to the BI Server using SSL.

5.4.5 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*.

5.4.6 Configuring SSL when Using Multiple Authenticators

If you are configuring multiple authenticators, and have configured an additional LDAP Authenticator to communicate over SSL (one-way SSL only), you need to put the corresponding LDAP server's root certificate in an additional keystore used by the virtualization (libOVD) functionality.

To configure SSL when using multiple authenticators:

> **Note:** Before completing this task, you must configure the custom property called `virtualize`, and set its value to `true` (for more information, see Section 3.4.5, "Configuring Multiple Authentication Providers Using Fusion Middleware Control").

1. Create the keystore:

   a. Set environment variables ORACLE_HOME, WL_HOME and JAVA_HOME.

      For example (on Windows):

      ```
      set ORACLE_HOME=<MW_HOME>\Oracle_BI1
      set WL_HOME=<MW_HOME>\wlserver_10.3
      set JAVA_HOME=<MW_HOME>\jdk160_24
      ```

   b. Set up the keystore by running libovdconfig.sh (on UNIX), or libovdconfig.bat (on Windows), using -createKeystore option.

      For example, on UNIX, open a shell prompt and change the directory to <*MW_HOME*>/oracle_common/bin. Then, run the following command

(which prompts for the Oracle Business Intelligence administrator user name and password), for example:

./libovdconfig.sh -host <*hostname*> -port <*Admin_Server_Port*> -username <*BI Admin User*> -domainPath <*MW_HOME*>/user_projects/domains/bifoundation_domain -createKeystore

Windows location:

<*MW_HOME*>\oracle_common\bin\libovdconfig.bat

c. When prompted, enter the Oracle Business Intelligence administrator password, and the OVD Keystore password (a new password that will be used to secure a Keystore file), created by the libovdconfig.sh -createKeystore command.

Once this command runs, you should see two new credentials in the Credential Store and a new Keystore file called adapters.jks under <*MW_HOME*>\user_projects\domains\bifoundation_domain\config\fmwconfig\ovd\default\keystores.

2. Export the root certificate from the LDAP directory (refer to your LDAP documentation on how to do this).

3. Import the root certificate to the libOVD keystore using the keytool command:

<*MW_HOME*>/jdk160_24/bin/keytool -import -keystore <*MW_HOME*>\user_projects\domains\bifoundation_domain\config\fmwconfig\ovd\default\keystores/adapters.jks -storepass <*KeyStore password*> -alias <*alias of your choice*> -file <*Certificate filename*>

4. Restart WebLogic and BI System processes.

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

5.5 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.

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 5.3.4, "Manually Configuring SSL Communication Between Components Using the MBean Browser".

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://download.oracle.com/javase/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 located at ORACLE_INSTANCE\config\OracleBIJavaHostComponent\coreapplication_obijh*n*\config.xml 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 Presentation Services configuration file located at ORACLE_INSTANCE/config/OracleBIPresentationServicesComponent/coreapplication_obips*n*/instanceconfig.xml and add the attribute cipherSuites="RSA+RC4+SHA" to the Listener and the JavaHostProxy elements within the ServerInstance element.

7. Edit the BI Scheduler configuration file located at ORACLE_INSTANCE/config/OracleBISchedulerComponent/coreapplication_obisch*n*/instanceconfig.xml and 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 more 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. From the System MBean Browser, select the BIDomain.BIInstanceSecurityConfiguration MBean.

    Make sure that the SSLManualConfig attribute is set to `false` before running the SSL status report.

For information about how to navigate to the MBean, see Section 5.3.4.2, "Generating the SSL Certificates". You do not need to lock the configuration to run the SSL report.

11. Run a SSL status report to confirm SSL is enabled by following the steps in Section 5.3.4.6, "Confirming SSL Status Using the MBean Browser".

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 login is verified to ensure the user has the necessary credentials to log in to the system. The BI Server authenticates each connection request it receives. The following legacy authentication methods are supported by the 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 Using Initialization Blocks"
- 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 Using Initialization Blocks

You can set up the 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 the 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 using initialization blocks:

1. Create an LDAP Server as follows:
   a. Select **Manage** then **Identity** in the Administration Tool to launch the Identity Manager.
   b. Select **Directory Servers** from the left pane in Identity Manager.
   c. Right-click in the right pane in Identity 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 assign the USER variable to an LDAP attribute (for example, uid, sAMAccountName, cn).

   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 the 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 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 Active Directory instance:

- In **Log On To**, select **All Computers**, or if you list some computers, include the Active Directory server as a Logon workstation.

- Ensure that **User must change password at next logon** is not selected.

In the 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:** The 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 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:
   - **Name**. The name to identify this connection (for example, My LDAP).
   - **Host name**. The name of your LDAP server.
   - **Port number**. The default LDAP port is 3060.
   - **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 enter the required information. The 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 the 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 using initialization blocks, 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 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. The legacy LDAP mechanism is only attempted if authentication fails against Oracle Platform Security Services.

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.

To define the USER session variable for LDAP authentication:

1. Open a repository in the Administration Tool in either offline or online mode.
2. Select **Manage**, then **Variables** from the Administration Tool menu.
3. Select the **Session -> Initialization Blocks** leaf of the tree in the left pane.
4. Right-click in the right pane and select **New Initialization Block**.
5. In the Session Variable - Initialization dialog box, enter `Authentication` in the **Name** field.
6. Click **Edit Data Source**.
7. Select LDAP Server from the **Data Source Type** drop down list.
8. Browse to select the appropriate LDAP server from the list.

9. Click **OK**.
10. Click **Edit Data Target**.
11. Click **New**.
12. Enter `USER` in the **Name** field.
13. Click **OK**.
14. Click **Yes** to the warning message about the USER session variable having a special purpose.
15. Enter in the **Mapped Variable** field, the LDAP attribute that holds the user ID.
16. Click **OK**.
17. Select the **Required for Authentication** checkbox.
18. Click **OK**.

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 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 are 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 Using Initialization Blocks".

To set up external table authentication:

1. Import information about the external table into the Physical layer.
2. Select **Manage**, then **Variables** in the 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, enter a name for the initialization block.
6. Select **Database** from the **Data Source Connection** list.
7. Click **Browse** to search for the name of the connection pool this block uses.
8. In the **Initialization String** area, enter the SQL statement that is issued at authentication time.

   The values returned by the database in the columns in the SQL statement 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 entered.

   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 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 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 the 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.

Sample custom authenticator code can be found in the BI EE Sample Application downloadable from Oracle Technology Network (OTN).

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 the Administration Tool, select **Manage**, then **Identity**. Select **Custom Authenticators** from the navigation tree. Select from the following options:
   - To create 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 plug-in DLL for this custom authenticator.
   - **Configuration parameters**: The parameters that have been explicitly exposed for configuration for this custom authenticator.
   - **Encrypted parameter**: The parameters that have been encrypted, such as passwords for this custom authenticator.
   - **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 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 the 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 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 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 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 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

For backward capability, this release supports the ability to manage catalog object privileges using Catalog groups, and the ability to set application role membership for users using initialization blocks, when authentication is also being performed by initialization blocks.

> **Note:** It is not possible to set application role membership using initialization blocks, when authentication is performed by Oracle Platform Security Services.

This section contains the following topics:

- Section A.2.1, "Changes Affecting Security in Presentation Services"
- Section A.2.2, "Managing Catalog Privileges Using Catalog Groups"
- Section A.2.3, "Setting Up Authorization Using Initialization Blocks"

A.2.1 Changes Affecting Security in Presentation Services

If you have upgraded from a previous release, the best practice is to begin managing 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 privileges in the Oracle BI Presentation Catalog:

- 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 the BI Server. Members of BI Server groups are no longer automatically made members of Catalog groups having the same name, as was the practice in earlier releases.
- Privileges continue to be stored in the Oracle BI Presentation Catalog and cannot be accessed from the administrative interfaces used to manage the policy store.
- The Everyone 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.
- Catalog groups can no longer be password protected. All Catalog groups migrated during upgrade no longer have a password.

A.2.2 Managing 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 Section D.2.2, "Working with Catalog Groups".

You can grant these privileges by assigning other Catalog groups, users, or application roles to a Catalog group.

Note: Assigning 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 display the Manage Privileges page.
3. Click the link for the privilege from the Manage Privileges page.
4. To assign 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 7 until the privileges have been granted or denied as needed.

A.2.3 Setting Up Authorization Using Initialization Blocks

To set application role membership for users using initialization blocks, the following conditions apply:

- Initialization blocks to set ROLES or GROUP session variables will only function when the user fails to authenticate through an authenticator configured in the WebLogic security realm, and the user instead authenticates through an initialization block.
- You must set up an initialization block to set the values of either ROLES or GROUP, and the BI Server will make the values of both variables the same.
- When using an initialization block to set ROLES or GROUP session variables, the values of the variables should be set to match by name against one or more application roles configured using Fusion Middleware Control, for example, BIConsumer. A user will be assigned these application roles and associated permissions during authentication.
- For information about application roles, and how to add a new application role, see Section 2.4, "Managing Application Roles and Application Policies Using Fusion Middleware Control".
- When using initialization blocks to set ROLES or GROUP session variables, the association of groups to application roles is performed using the logic previously described. Assignment of groups to application roles in the policy store is not used in this case.
- Any value of the ROLES or GROUP variable that does not match an application role will be matched by name against the available Catalog groups in the Oracle BI Presentation Catalog. The user will be assigned these Catalog groups and associated privileges.
- Any value of ROLES or GROUP that does not match an application role or a Catalog group will be ignored.

To define the ROLES session variable for database authorization:

1. Open a repository in the Administration Tool in either offline or online mode.
2. Select **Manage**, then **Variables** from the Administration Tool menu.
3. Select the **Session -> Initialization Blocks** leaf of the tree in the left pane.
4. Right-click in the right pane and select **New Initialization Block**.
5. In the Session Variable - Initialization dialog box, enter `Authorization` in the **Name** field.
6. Click **Edit Data Source**.
7. Select Database from the **Data Source Type** drop down list.
8. Enter the SQL.

   The SQL can be anything that returns either a list of groups, or a single group if row-wise initialization is not used.

   For more information, see "Using Variables in the Oracle BI Repository" in the *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.
9. Click **Browse** to select a connection pool.
10. Click **Select**.

11. Click **OK**.
12. Click **OK**.
13. Click **Edit Data Target**.
14. Click **New**.
15. Enter `ROLES` in the **Name** field.
16. Click **OK**.
17. Click **Yes** to the warning message about the ROLES session variable having a special purpose.
18. Click **OK**.
19. Clear the **Required for Authentication** checkbox.
20. Click **OK**.

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 privileges discussed in this chapter are those maintained in the policy store provider, such as the Oracle Business Intelligence Presentation Services privileges. Catalog permissions are distinct because they are maintained in the Oracle BI Presentation Catalog. For more information about Presentation Services privileges, see Section D.2.3, "Managing Presentation Services Privileges".

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 Application 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 (OPSS) 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 Application Security Guide*.

Note: In future versions of the documentation, references to Oracle Platform Security Services (OPSS) will be replaced by references to Oracle Entitlements Server Basic (OES Basic), with no change to customer-visible behavior. For more information, see Section 3.9, "Configuring Oracle Internet Directory as the Policy Store and the Credential Store".

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 Application Security Guide*.

Oracle Business Intelligence uses these security platform elements as follows:

Application Policy

For more information about application policies, see Section 1.9, "Terminology".

An **application stripe** defines a subset of policies in the policy store. The Oracle Business Intelligence application stripe is named **obi**.

Application Role

For more information about application roles, see Section 1.4.1, "About Application Roles". 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 assigned 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 assigned to a policy, the permissions become associated with the application role through the relationship between policy and role. If groups of users have been assigned 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

For more information about authentication providers, see Section 1.3, "About Authentication".

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 assigning 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 assigned to application roles.	■ system.jazn-data.xml file. ■ Managed with Fusion Middleware Control.	Oracle Business Intelligence can be configured to use Oracle Internet Directory.

Table B–1 (Cont.) Default Security Providers

Security Provider Type	Purpose	Default Provider	Options
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.

Catalog privileges and permissions are not maintained in the policy store provider.

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 3.9, "Configuring Oracle Internet Directory as the Policy Store and the Credential 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.

Table B–2 (Cont.) Default Permissions

Permission Name	Description
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.
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	Used by internal components that need to act on behalf of end users.
oracle.bi.server.manageRepositories	Grants permission to open, view, and edit repository files using the Administration Tool or the Oracle BI Metadata Web Service.
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.

Table B–2 (Cont.) Default Permissions

Permission Name	Description
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 the following types of resource:

- Inline Service
- Decision Center Perspective
- Batch Job

  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 lists the Oracle RTD resource types.

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.
Inline Service	rtd_ils	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.
Inline Service	rtd_ils	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.
Inline Service	rtd_ils	open_service:read	Authorizes the use of Decision Center to open the named Inline Service for viewing. 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.
Inline Service	rtd_ils	open_ service:write	Authorizes the use of Decision Center to open the named Inline Service for editing.

Table B–3 (Cont.) Oracle RTD Resource Types and Actions

Type of Resource	Resource Type Name Stored in Application Grants	Action[:Qualifier]	Comments
Inline Service	rtd_ils	deploy_service	Authorizes the deployment of the named Inline Service from Decision Studio.
Inline Service	rtd_ils	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	Opens 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.

> **Note:** The BIAdministrator role must exist (with the BISystem role), for Oracle Business Intelligence to function correctly.

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.

> **Note:** The BISystem Role must exist (with the BIAdministrator role), for Oracle Business Intelligence to function correctly.

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, which is a replacement Everyone Catalog group in release 10*g* . The Authenticated role is not part of 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 assign 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 Application 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

Table B–4 (Cont.) Default Application Role, Permission Grants, and Members

Role Name	Role Permissions	Members
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
BISystem	■ oracle.bi.scheduler.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 open_service:write deploy_service download_service choice_editor decision_service:normal decision_service:stress dc_perspective batch_admin
BIAuthors	rtd_ils	_all_	open_service:read open_service:write deploy_service download_service decision_service:normal decision_service:stress
BIAuthors	rtd_dc_persp	_all_	dc_perspective
BIConsumer	rtd_ils	_all_	open_service:read choice_editor decision_service:normal
BIConsumer	rtd_dc_persp	Explore	dc_perspective
BIConsumer	rtd_dc_persp	At a Glance	dc_perspective
BIConsumer	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 currently 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. Groups are then assigned to application roles to grant rights. Oracle recommends that you organize your users into groups for easier maintenance.

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 assigned 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.

Table B–6 (Cont.) Default Groups and Members

Purpose	Group Name and Members	Description
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 keep using 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
Repository 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 User Permissions Are Granted Using Application Roles

The default Oracle Business Intelligence security configuration provides preconfigured permissions granted to application roles. Application roles have groups as members, and permissions are inherited by users through their membership of groups. A group assigned to an application role conveys the role's permissions to all members of the group.

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 of one or more groups according to the level of access required.
- An application role defines the role a user typically performs when using Oracle Business Intelligence. The default security configuration provides the following roles: administrator (BIAdministrator), author (BIAuthor), and consumer (BIConsumer).
- A group is assigned to one or more application roles that match the type of access required by each group.
- An application policy defines Oracle Business Intelligence permissions that grant a set of access rights corresponding to each role type.
- An application role is assigned to an application policy that grants the set of permissions required by the role type (administrator, author, consumer). Once configured, the application role is the grantee of the application policy.
- Group membership can be inherited by nature of the group hierarchy. Application roles assigned to inherited groups are also inherited, and their permissions are likewise conveyed.

How the system determines a user's permissions:

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 the user's groups.

3. A list of the user's groups is checked against the application roles. A list is created of the application roles that are assigned 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 the 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 association 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.

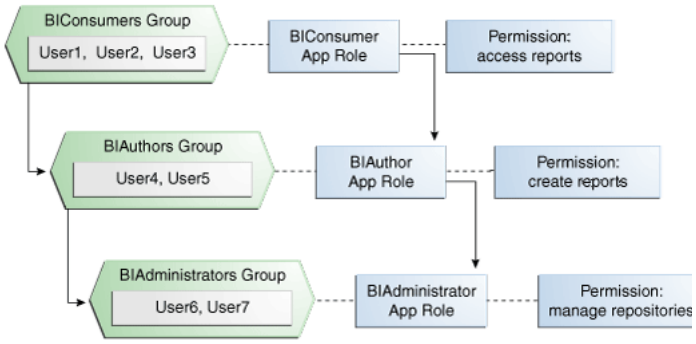
Users can also obtain permissions by inheriting group membership and application roles. For more information and an example of how this is accomplished, see Section B.4.4.1, "Permission Inheritance and Role Hierarchy".

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.

Figure B–2 shows the relationship between default application roles and how permissions are granted to members.

Figure B–2 Default Application Role Hierarchy Example



In Figure B–2 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 that members of the BIAdministrators group are

granted *all* the permissions of the BIAdministrator role, the BIAuthor role, and the BIConsumer role. So a 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 Catalog Groups and Precedence

If *Catalog groups* and application roles are used in combination to manage 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 Catalog group takes precedence over an application role. For more information about Presentation Services permissions and privileges, see Section D.2.3, "Managing Presentation Services Privileges".

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 Application 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 2.3.2, "Creating a New 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 2.4.4, "Modifying Application Roles Using Fusion Middleware Control" Section B.4.1.3, "Default Application Roles, Permission Grants, and Group Mappings" *Oracle Fusion Middleware Application 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 2.4.2, "Creating Application Roles Using Fusion Middleware Control" *Oracle Fusion Middleware Application 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 3.4, "Configuring Alternative Authentication Providers" Appendix A, "Alternative Security Administration Options"

Table B–11 (Cont.) Task Map: Common Security Tasks to Implement Oracle Business Intelligence

Task	Description	For Information
Create a new application role.	Create a new application role and make the role a grantee of an application policy.	Section 2.4.2, "Creating Application Roles Using Fusion Middleware Control"
Assign a group to a newly created application role.	Assign a group to a newly created application role to convey the permission grants to group members.	Section 2.4.4, "Modifying Application Roles Using 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.
- References to old Catalog groups and users in the Oracle BI Presentation Catalog 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 Oracle BI 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 Oracle BI 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 Oracle BI Presentation Catalog:

- The Oracle BI 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. The internal representation of each user is updated 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 Oracle BI Presentation Catalog is reviewed.
- Leaves the Release 10*g* Catalog groups in the upgraded Oracle BI 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 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 using and configuring Oracle Business Intelligence security, and explains how to solve them. It contains the following sections

- Section C.1, "Resolving User Login Authentication Failure Issues"
- Section C.2, "Resolving Inconsistencies with the Identity Store"
- Section C.3, "Resolving Inconsistencies with the Policy Store"
- Section C.4, "Resolving SSL Communication Problems"
- Section C.5, "Resolving Issues with BI System User Credentials"
- Section C.6, "Resolving Custom SSO Environment Issues"
- Section C.7, "Resolving RSS Feed Authentication When Using SSO"

C.1 Resolving User Login Authentication Failure Issues

This section helps you resolve some of the most common user login authentication failure issues encountered while using Oracle Business Intelligence Enterprise Edition 11*g*. It is not intended to be a comprehensive list of every possible scenario, and contains the following topics:

- Section C.1.1, "Authentication Concepts"

  This section describes the basic concepts of authentication in Oracle Business Intelligence Enterprise Edition 11*g*. You must understand concepts described throughout this guide as a prerequisite for using this section.

- Section C.1.2, "Using the Oracle BI Security Diagnostics Helper to Automatically Identify Security Issues"

  This section describes how to deploy and use an Oracle BI security diagnostics helper application, enabling you to automatically identify some common security issues.

- Section C.1.3, "Identifying Causes of User Login Authentication Failure"

  This section provides a cause-and-effect diagram to use as a checklist for identifying authentication failure causes.

- Section C.1.4, "Resolving User Login Authentication Failures"

  This section provides reasons and solutions for login authentication failure.

C.1.1 Authentication Concepts

This section describes authentication concepts helps you to resolve user login authentication failure issues, and contains the following topics:

- Section C.1.1.1, "Authentication Defaults on Install"
- Section C.1.1.2, "Using Oracle WebLogic Server Administration Console and Fusion Middleware Control to Configure Oracle Business Intelligence"
- Section C.1.1.3, "WebLogic Domain and Log Locations"
- Section C.1.1.4, "Oracle Business Intelligence Key Login User Accounts"
- Section C.1.1.5, "Oracle Business Intelligence Login Overview"

C.1.1.1 Authentication Defaults on Install

Immediately after install, Oracle Business Intelligence is configured to authenticate users against the WebLogic embedded LDAP server through the DefaultAuthenticator. Default user accounts will have been set up, including the WebLogic Admin user which uses the account credentials entered as the WebLogic administrator user during the install.

C.1.1.2 Using Oracle WebLogic Server Administration Console and Fusion Middleware Control to Configure Oracle Business Intelligence

You configure Oracle Business Intelligence using Oracle WebLogic Server Administration Console and Fusion Middleware Control. For more information about using these applications, see Section 1.6, "Using Tools to Configure Security in Oracle Business Intelligence".

You must log in to Oracle WebLogic Server Administration Console and Fusion Middleware Control with the username and password that you specified for the Administrative user during the install process, unless you have altered or removed that account or configured another account with the appropriate access (see Section C.1.1.4, "Oracle Business Intelligence Key Login User Accounts").

C.1.1.3 WebLogic Domain and Log Locations

To diagnose and resolve user login authentication issues, you must know the locations of the WebLogic domain, and log files, as follows:

> **Note:** This section assumes that the install used the default locations. If you specified different install locations, you must modify the paths accordingly.

- WebLogic domain where Oracle Business Intelligence is installed

  MW_HOME/user_projects/domains/bifoundation_domain/

- WebLogic Administration Server logs

  MW_HOME/user_projects/domains/bifoundation_domain/servers/AdminServer/logs/

- WebLogic Managed Server logs:

  MW_HOME/user_projects/domains/bifoundation_domain/servers/bi_server1/logs/

- BI Server logs:

MW_HOME/*INSTANCE*/diagnostics/logs/OracleBIServerComponent/coreapplication_obis*n*/

C.1.1.4 Oracle Business Intelligence Key Login User Accounts

This section describes the key login user accounts, and contains the following sections:

- Section C.1.1.4.1, "WebLogic Admin User Account"
- Section C.1.1.4.2, "BI System User Account"
- Section C.1.1.4.3, "Oracle System User Account"

C.1.1.4.1 WebLogic Admin User Account The WebLogic admin user account enables you to start the WebLogic Server, and to administer WebLogic Server using the Oracle WebLogic Server Administration Console and Fusion Middleware Control. The WebLogic admin account must have the WebLogic global role called Admin (this is not an Oracle Business Intelligence application role), which also enables them to add new WebLogic administrator accounts.

To add or remove users to or from the global admin role using the Oracle WebLogic Server Administration Console:

1. Log in to Oracle WebLogic Server Administration Console as a WebLogic administrator, and click **Lock & Edit** in the Change Center.

   For more information, see Section 1.6.1, "Using Oracle WebLogic Server Administration Console".

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

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

3. Select **Roles and Policies** from the tabs along the top.

4. In the list of roles, click on the plus sign to expand Global Roles, then Roles, then click the **View Role Conditions** link for the Admin global role.

5. Check to ensure that the specified conditions match your user, either directly, or through a group they belong to.

   For example, condition may be User=myadminaccount or Group=Administrators.

6. If you have made any changes, click **Save**.

7. In the Change Center, click **Activate Changes**.

C.1.1.4.2 BI System User Account The BI System User account enables internal authentication between different Oracle BI components, and must not be used as a normal user account (for example, to log in with or to use the application). Oracle recommends that if you change from using the default user created on install, you must assign a special account for this purpose, and not use an account which is used by an actual user of the system.

An account named BISystemUser is created by default in the WebLogic embedded LDAP store during install, to be used as the BI System User account. The credentials of this account (the password is created at random) are stored in the credential store under the system.user key in the oracle.bi.system map. If you change the account used for the BI System user, or remove the Default Authenticator account, you cannot authenticate against the WebLogic embedded LDAP, and you must create a new BI System user account. For more information, see Section 3.7, "Configuring a New Trusted User (BISystemUser)".

> **Note:** The BI System User account (BISystemUser by default) must be associated with the BISystem application role and the WebLogic global role named Admin. If you make any changes to the BI System User account, you must restart the Administration Server, Managed Servers, and Oracle Business Intelligence system components.

User authentication commonly fails between the BI Server and the BI Security Services, when either of the following are true:

- User credentials (in the credential store) are not synchronized with the user population.
- Oracle Business Intelligence has not been restarted.

  In this situation, the BI Server sends out of date system.user credentials to the BI Security Service.

C.1.1.4.3 Oracle System User Account The OracleSystemUser account is created by default in the WebLogic embedded LDAP store during Oracle BI EE install and is required for Oracle Web Services Manager (OWSM). OWSM is used by WebLogic Server or OPSS to secure calls to the Oracle Business Intelligence Security Service. If the Oracle System User account is incorrect, then the Oracle Business Intelligence login process fails (for more information, see Section C.1.1.5). The Oracle System User account must be named OracleSystemUser, and must be in a group named OracleSystemGroup. The group OracleSystemGroup must have the global role named OracleSystemRole assigned to it.

C.1.1.5 Oracle Business Intelligence Login Overview

When a user logs in to Oracle Business Intelligence without Single Sign-On, authentication and user profile look-up occurs. In a Single Sign-On (SSO) environment, authentication is performed outside the Oracle Business Intelligence system, and identity is asserted instead, but user profile look-up still occurs.

Authentication and identity assertion is performed by authentication providers and asserters respectively, and is configured using Oracle WebLogic Server Administration Console. The user profile lookup uses a User Role API which relies on having a single identity store configured. The first configured authentication provider is used by default as the identity store, for example to search for user GUIDs, or email. Successful login to Oracle Business Intelligence requires that the first configured authentication provider contains your user population. For more information, see Section 3.4, "Configuring Alternative Authentication Providers".

The login process flow begins by entering user credentials in the login screen these are sent to Presentation Services, and then to the BI Server. The BI Server attempts to authenticate the user credentials by calling the BI Security Web Service (deployed in the WebLogic Managed Server, and protected by a Web Service Security policy). The call requires the BI Server to authenticate itself to Oracle Web Services Manager, before it can be received by the BI Security Service. The BI Server uses credentials stored in the system.user key under the oracle.bi.system map (for more information, see Section C.1.1.4.3, "Oracle System User Account"), to authenticate itself to the BI Security Service when attempting to authenticate the credentials specified by the user. If the BI System User credentials are incorrect, or the Oracle Web Services Manager does not authenticate them, then the BI Server cannot call the BI Security Service and the login process fails.

C.1.2 Using the Oracle BI Security Diagnostics Helper to Automatically Identify Security Issues

You use the Oracle BI Security Diagnostics Helper to help identify security issues. Oracle recommends that you use the Oracle BI Security Diagnostics Helper when speaking to a representative from Oracle Support. For more information, see:

`https://support.oracle.com`

This topic includes information on how to setup, deploy, and run the Oracle BI Security Diagnostics Helper to identify problems with your Oracle BI system, and contains the following sections:

- Section C.1.2.1, "What Is the Oracle BI Security Diagnostics Helper?"
- Section C.1.2.2, "Setting Up the Oracle BI Security Diagnostics Helper Using a Script - First-Time Use Only"
- Section C.1.2.3, "Deploying the Oracle BI Security Diagnostics Helper"
- Section C.1.2.4, "Running the Oracle BI Security Diagnostics Helper"
- Section C.1.2.5, "Using the Oracle BI Diagnostics Helper"
- Section C.1.2.6, "Restarting the WebLogic Servers"

C.1.2.1 What Is the Oracle BI Security Diagnostics Helper?

The Oracle BI Security Diagnostics Helper is a JEE application that helps diagnose possible configuration issues which may prevent your users from being able to log in to your Oracle BI system.

In order to use Oracle BI Security Diagnostics Helper for the first time, you have to run a setup script before you deploy and run the application.

Once deployed you can start diagnosing possible security issues by accessing the Oracle BI Security Diagnostics Helper URL in a Web browser.

All of the instructions for setting up and using Oracle BI Security Diagnostics Helper are detailed in the following sections.

C.1.2.2 Setting Up the Oracle BI Security Diagnostics Helper Using a Script - First-Time Use Only

You must run a script before using the Oracle BI Security Diagnostics Helper for the first time use to ensure the Oracle BI Security Diagnostics Helper is setup correctly. The script grants the correct permissions to run the diagnostic checks. If you have already run the setup script you do not need to run it again.

To setup the Oracle BI Security Diagnostics Helper - for first-time use only:

You must run the script named addDiagnosticsCodeGrant.py for first time use only, to ensure that the Oracle BI Security Diagnostics Helper is setup correctly.

1. Open a command prompt and change to the scripts directory.

   In UNIX, the scripts directory is located in:

   MW_HOME/*ORACLE_HOME*/bifoundation/admin/scripts

   In Windows, the scripts directory is located in:

   *MW_HOME**ORACLE_HOME*\bifoundation\admin\scripts

   For example, in UNIX:

mw_home/Oracle_BI1/bifoundation/admin/scripts

2. Run the setup script.

   In UNIX use the following command syntax:

   ```
   MW_HOME/ORACLE_HOME/common/bin/wlst.sh addDiagnosticsCodeGrant.py
   t3://<WebLogic_host_name>:<WebLogic_port_number>
   ```

   For example, in UNIX enter the following command:

   ```
   mw_home/Oracle_BI1/common/bin/wlst.sh addDiagnosticsCodeGrant.py
   t3://localhost:7001
   ```

   In Windows use the following syntax:

   ```
   mw_home/Oracle_BI1/common/bin/wlst.cmd addDiagnosticsCodeGrant.py
   t3://<WebLogic_host_name>:<WebLogic_port_number>
   ```

   For example, in Windows enter the following command:

   ```
   mw_home\Oracle_BI1\common\bin\wlst.cmd addDiagnosticsCodeGrant.py
   t3://localhost:7001
   ```

   The script reports 'Added code grants to bidiagnostics' when it runs successfully. It may also display some warnings and error messages which you can ignore if the script ran successfully.

3. When prompted, enter the WebLogic administrator user name and password.
4. Restart the WebLogic Servers.

   For more information, see Section C.1.2.6, "Restarting the WebLogic Servers".

C.1.2.3 Deploying the Oracle BI Security Diagnostics Helper

You must deploy the Oracle BI Security Diagnostics Helper as it is not automatically deployed into the WebLogic Server during install.

To deploy the Oracle BI Security Diagnostics Helper:

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

   For more information, see Section 1.6.1, "Using Oracle WebLogic Server Administration Console".

2. Click **Deployments**.
3. Click **Install**.
4. Select the bidiagnostics.ear file

   In UNIX, the bidiagnostics.ear file is located in:

   MW_HOME/*ORACLE_HOME*/bifoundation/jee

   For example:

   mw_home/OracleBI1/bifoundation/jee/bidiagnostics.ear

5. Click Next to display the Install Application Assistant page.
6. Select **Install this deployment as an application** (already selected)
7. Click **Next** to display the Select deployment targets page.
8. In the Servers area select **AdminServer**.

9. Click **Next**, then **Finish**
10. Click, **Activate Changes**.

    The Oracle BI Diagnostic Helper changes into a 'prepared' State, and can be started.

    The following message should appear: "All changes have been activated. No restarts are necessary".

C.1.2.4 Running the Oracle BI Security Diagnostics Helper

After deploying the Oracle BI Diagnostics Helper it does not automatically start, you have to run it.

To run the Oracle BI Security diagnostics helper:

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

   For more information, see Section 1.6.1, "Using Oracle WebLogic Server Administration Console".
2. Click **Deployments**.
3. Select the **bidiagnostics** checkbox.
4. Click **Start** and choose one of the following options:

   - Servicing all requests.

   - Servicing only administration requests.

   The Oracle BI Security Diagnostics Helper is now setup and running.

   When started - bidiagnostics State is set to 'Active'

C.1.2.5 Using the Oracle BI Diagnostics Helper

The Oracle BI Security Diagnostics Helper performs a series of security tests outlined in the following topics:

- Section C.1.2.5.1, "OWSM Tests"
- Section C.1.2.5.2, "BI System User Tests"
- Section C.1.2.5.3, "User Credential Authentication Tests"

Note: To start the security tests enter a URL similar to the following, into a Web browser, for example:

```
http://mycomputer:7001/bidiagnostics/security/diagnostics.jsp
Use this URL for this release.

http://mycomputer:7001/bidiagnostics/security
Use this URL for later releases.
```

C.1.2.5.1 OWSM Tests The OWSM tests perform the following:

- Checks whether MDS-OWSM data source is present and reports where it is, if found.

  For example, the check might return the following message if the MDS-OWSM data source is found:

Found data source under JNDI path jdbc/mds/owsm.

- Checks the connection to MDS-OWSM.

  This check notes that it has successfully connected to the schema, or returns an error message if the check fails.

- Checks that the OracleSystemUser exists.

  This check notes that the user exists, or returns an error message if the check fails.

- Checks that the OracleSystemUser is in the OracleSystemGroup.

  This check notes whether the check succeeds, or returns an error message if the check fails.

C.1.2.5.2 BI System User Tests The BI System User tests perform the following:

- Checks that the system.user key exists in credential store.

  This check notes that the system.user key exists, or returns an error message if the check fails.

- Authenticates the system.user account.

- Checks the system.user permissions.

  This check that the system.user permissions exist, or returns an error message if the check fails

C.1.2.5.3 User Credential Authentication Tests The user credential authentication tests perform the following (when you complete all fields and click **Test Authentication**):

- Checks the identity store using the authentication providers.

- Checks Oracle BI Security Service authentication.

- Checks Oracle BI Web Service authentication.

C.1.2.6 Restarting the WebLogic Servers

To restart the WebLogic Servers:

1. Log in to Oracle WebLogic Server Administration Console.

   For more information, see Section 1.6.1, "Using Oracle WebLogic Server Administration Console".

2. In the Domain Structure area, click **Environment**, **Servers**.

3. In the Summary of Servers page, display the **Control** tab.

4. Select `bi_server1`, and click **Shutdown** (select 'When work completes', or 'Force shutdown now', as required.

   If you are using a cluster, you will select multiple BI Servers, for example, bi_server1, bi_server2.

   Wait until all BI Servers are shut down before proceeding to the next step.

5. Select `AdminServer`, and click **Shutdown** (select 'When work completes', or 'Force shutdown now', as required.

6. Open a command prompt and change directories to where the startWebLogic script is located.

   For example, in UNIX:

user_projects/domains/bifoundation_domain/bin

7. Start WebLogic Server.

   In UNIX, enter the following command:

   ```
   ./startWebLogic.sh
   ```

   Enter the username and password when prompted.

   Wait for WebLogic Server to start before proceeding to the next step.

8. Log in to Oracle WebLogic Server Administration Console.

   For more information, see Section 1.6.1, "Using Oracle WebLogic Server Administration Console".

9. In the Domain Structure area, click **Environment**, **Servers**.

10. In the Summary of Servers page, display the **Control** tab.

11. Select `bi_server1`, and click **Start**.

    If you are using a cluster, you will select multiple BI Servers, for example, bi_server1, bi_server2.

C.1.3 Identifying Causes of User Login Authentication Failure

This section helps you to identify causes of authentication failure when logging in to Oracle Business Intelligence.

Figure C–1 and Figure C–2 are cause and effect diagrams that you can use to identify possible causes of user login authentication failure. Once you have identified the likely cause of user login identification failure, refer to Section C.1.4, "Resolving User Login Authentication Failures" for information about how to resolve the issues.

Figure C–1 Causes of User Login Failure - Part 1

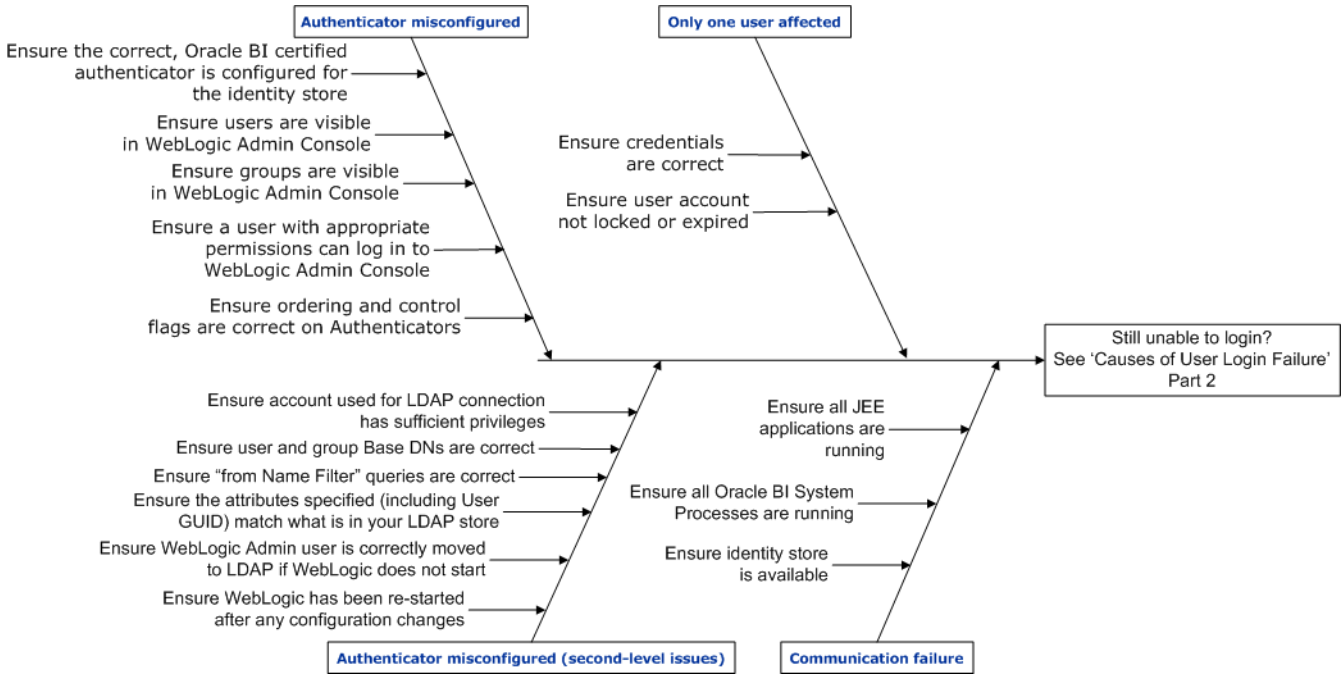


Figure C–1 helps you identify causes of log in failure. If you are unable to identify the cause of log in failure using Figure C–1, then use Figure C–2 instead.

The description for Figure C–1 is as follows:

- **Authenticator misconfigured**.
  - Ensure that the correct Oracle Business Intelligence certified authenticator is configured for the identity store.
  - Ensure that users are visible in the Oracle WebLogic Server Administration Console.
  - Ensure that groups are visible in the Oracle WebLogic Server Administration Console.
  - Ensure that a user with appropriate permissions can log in to Oracle WebLogic Server Administration Console.
  - Ensure that the ordering and control flags on authenticators are correct.
- **Authenticator misconfigured (second-level issues)**.
  - Ensure that WebLogic Server has been re-started after any configuration changes.
  - Ensure that the WebLogic Admin user is correctly moved to LDAP, if WebLogic Server does not start.
  - Ensure that the attributes specified (including user GUID) match what is in your LDAP store.
  - Ensure that 'from Name Filter' queries are correct.
  - Ensure that user and group Base DN settings are correct.
  - Ensure that the account used for LDAP connection has sufficient privileges.
- **Only one user affected**.
  - Ensure that correct credentials are used.
  - Ensure that the user account is not locked or expired.
- **Communication failure**.
  - Ensure that the identity store is available.
  - Ensure that all BI System processes are running.
  - Ensure that all JEE applications are running.

Figure C–2 Causes of User Login Failure - Part 2

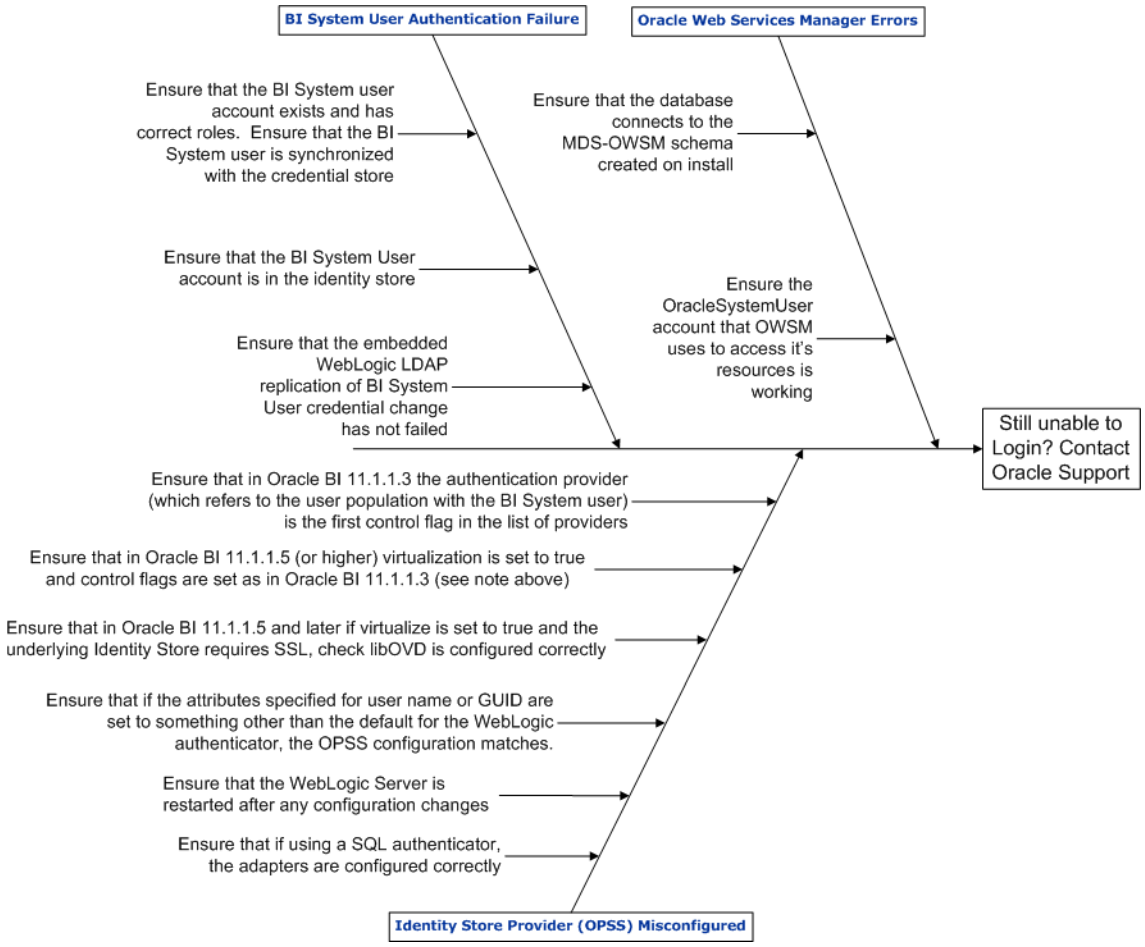


Figure C–2 helps you identify alternative causes of log in failure if you cannot identify them using Figure C–1. However, if you still cannot identify the causes of login failure after using Figure C–2, contact Oracle Support at:

`https://support.oracle.com`

The description for Figure C–2 is as follows:

- **BI System User authentication failure**.
  - Ensure that the BISystem user account exists and has correct roles.
  - Ensure that the BISystem user is synchronized with credential store.
  - Ensure that the BISystem user account is in the identity store.
  - Ensure that WebLogic embeddedLDAP replication of BI System User credential change has not failed.
- **Identity store provider (OPSS) misconfigured**.
  - Ensure that if using a SQL authenticator, the adapters are configured correctly.
  - Ensure that if the attributes specified for username or GUID are set to something other than the default value for the WebLogic authenticator, the OPSS configuration matches.
  - Ensure that in Oracle Business Intelligence Release 11.1.1.5 (or higher):

    * Virtualization is set to true.
    * Control flags are set as in Oracle Business Intelligence Release 11.1.1.3 (see following bullet).
  - Ensure that in Oracle Business Intelligence Release 11.1.1.3 the authentication provider (which refers to the user population with the BI System User), is the first control flag in the list of providers.
  - Ensure that the WebLogic Server is re-started after any configuration changes.
  - Ensure that in Oracle Business Intelligence Release 11.1.1.5 (or higher), if virtualization is set to true and the identity store requires SSL, virtualization must be configured correctly. For more information, see Section 5.4.6, "Configuring SSL when Using Multiple Authenticators".
- **Oracle Web Services Manager errors**.
  - Ensure the database connects to the MDS-OWSM schema created on install.
  - Ensure the OracleSystemUser account that OWSM uses to access its resources is working.

C.1.4 Resolving User Login Authentication Failures

This section explains user login authentication failures, describes how to resolve them, and contains the following topics:

- Section C.1.4.1, "Single User Cannot Log In to Oracle Business Intelligence"
- Section C.1.4.2, "Users Cannot Log In to Oracle Business Intelligence Due to Misconfigured Authenticators"
- Section C.1.4.3, "Users Cannot Log In to Oracle Business Intelligence When Oracle Web Services Manager is not Working"
- Section C.1.4.4, "Users Cannot Log In to Oracle Business Intelligence - Is BI System User Authentication Working?"
- Section C.1.4.5, "Users Cannot Log In to Oracle Business Intelligence - Is the External Identity Store Configured Correctly?"
- Section C.1.4.6, "Users Can Log In With Any or No Password"
- Section C.1.4.7, "Have Removed Default Authenticator and Cannot Start WebLogic Server"

C.1.4.1 Single User Cannot Log In to Oracle Business Intelligence

This section contains the following topics:

- Section C.1.4.1.1, "Is Login Failure the Result of User Error?"
- Section C.1.4.1.2, "Is Your Account Locked?"

C.1.4.1.1 Is Login Failure the Result of User Error? The first check is whether the user cannot log in to Oracle Business Intelligence due to a simple error for example, did the user enter the wrong password? If other users can log in to Oracle Business Intelligence, but one user cannot, check their credentials. Alternatively, see Section C.1.4.1.2.

C.1.4.1.2 Is Your Account Locked? Many LDAP authenticators lock user account when login attempts exceeds a specified threshold. For example, an account may be locked after 3 failed login attempts to defeat automated attacks.

If a user tries repeatedly to log in, their credentials might have been incorrectly entered enough times to trigger this mechanism. Refer to the documentation for your chosen identity store to discover how to unlock user accounts. For example, to unlock a locked user account when using WebLogic embdedded LDAP, see "Unlock user accounts" in *Oracle Fusion Middleware Oracle WebLogic Server Administration Console Online Help*.

C.1.4.2 Users Cannot Log In to Oracle Business Intelligence Due to Misconfigured Authenticators

The most common cause of authentication failure is misconfiguration of authenticators in WebLogic Server as follows:

> **Note:** Make sure you have read, and are familiar with the steps and concepts identified in Chapter 3, "Using Alternative Authentication Providers".

- Section C.1.4.2.1, "Have you Specificed the Correct Authenticator for Your Identity Store/LDAP Server?"
- Section C.1.4.2.2, "Is the Authenticator for Your LDAP Server Configured Correctly?"
- Section C.1.4.2.3, "Are the Control Flags for Your Authenticators Set Correctly and Ordered Correctly?"

C.1.4.2.1 Have you Specificed the Correct Authenticator for Your Identity Store/LDAP Server?

WebLogic Server uses a variety of server-specific authenticators in addition to the embedded LDAP authenticator. This is because some LDAP server products have differences which mean they do not appear to be generic LDAP servers and therefore the embedded LDAP authenticator may not work when configured to query against them. For example, the generic LDAP server does not work with Active Directory (AD), even though AD does apparently fully implement LDAP and successfully presents itself as an LDAP server to many LDAP query tools. Find out which LDAP server you are using and ensure you configure the appropriate authenticator.

C.1.4.2.2 Is the Authenticator for Your LDAP Server Configured Correctly?

If the configuration settings for the LDAP server used as the primary identity store are not correctly configured, then users cannot be correctly authenticated. Some common things to check include:

- Account used for LDAP connection.

  In the LDAP Authenticator provider-specific configuration, you must specify the DN of a principal that is used to connect to the LDAP server. This account must exist and have sufficient privileges to be able to run queries to retrieve the user or group population from the trees specified in the User or Group Base DNs. In a restricted LDAP environment, this may require elevated privileges beyond those granted to ordinary user accounts.

- Ensure user and group Base DNs are correct.

  Both groups and users are searched for within the tree specified by the user or group Base DN, make sure that the 'tree' specified actually contains your user or group population.

- Ensure 'from Name Filter' queries are correct.

Groups and users are found within the trees specified in the base DN by using the query specified in 'User from name filter' and 'Group from Name filter', with %u used as a placeholder for the user id passed in during queries on a specific user (including during authentication) and %g used similarly as the placeholder for the group name passed in when looking up a specific group. Check that the queries specified are syntactically and logically correct for your directory and test that you can run them (and get the expected results) from an LDAP browser, using the credentials specified in your authenticator configuration.

- Ensure the attributes specified match what is in your LDAP store.

  The attributes/objectclasses for users, groups and GUIDs are all specified in the Authenticator configuration. The authenticators are pre-configured with "sensible" defaults but on many sites these are not necessarily the ones in use. You should make sure (for example) that the value specified in User Name Attribute exists and is actually being used for the user's names in the LDAP server on your site.

- WebLogic Admin user moved to LDAP and cannot boot WebLogic.

  If you move the WebLogic Admin user from embedded LDAP and/or removed the DefaultAuthenticator (that is, you rely on LDAP authentication of the WebLogic admin user), and have misconfigured your authenticator configuration, then WebLogic Server does not start.

- Check if users are able to log in to Oracle WebLogic Server Administration Console.

  Assuming you can still log in to the Oracle WebLogic Server Administration Console (and if you can start WebLogic, you should be able to log in to the Oracle WebLogic Server Administration Console using the credentials you used to start the server) you can then assign one of the LDAP users the WebLogic Global Admin role and see if they are able to log in to the Oracle WebLogic Server Administration Console (http://<*biserver*>:7001/console). If they are able to log in to the Oracle WebLogic Server Administration Console, but not Oracle Business Intelligence, then the LDAP authenticator configuration is correct, but it may not be accessible to Oracle Business Intelligence. There are two things to check in this instance:

  - The identity store that contains your users may not be exposed as an identity store to OPSS - check the authenticator ordering and control flags section (see Section C.1.4.2.3, "Are the Control Flags for Your Authenticators Set Correctly and Ordered Correctly?").
  - Your users are authenticating correctly to LDAP, but there is a problem with the BI System User which prevents the BI Security Service from authenticating users (see Section C.1.4.4, "Users Cannot Log In to Oracle Business Intelligence - Is BI System User Authentication Working?").

  Note: if you temporarily assign the WebLogic Global Admin role to a user to test this scenario, please remember to remove this as soon as you have completed testing or else you may find you've given one (or many, if you specify the role condition by group name match) of your users a lot more power than you intended them to have!

C.1.4.2.3 Are the Control Flags for Your Authenticators Set Correctly and Ordered Correctly?

The primary identity store must be set as the first one in the list of authenticators (note that this restriction is lifted from Oracle Business Intelligence Release 11.1.1.5 (or higher) when virtualization is set to true). Oracle Business Intelligence uses the user

role API from OPSS which only picks up the first identity store from the list of authenticators for example, when looking up user GUIDs, profile information, roles. This is a prime cause of the scenario whereby a user can log in to Oracle WebLogic Server Administration Console (hence demonstrating that the authentication part of logging in is succeeding), but cannot log in to Oracle Business Intelligence (because the identity store containing the user is not first in the list).

Where more than one authenticator is configured, in the general case the control flags should all be set to SUFFICIENT. This enables each one to be tried in turn until authentication succeeds. If authentication is successful, no further authenticators are tried. If none of the authenticators can authenticate the supplied credentials, the overall authentication process fails.

Note: During install, the DefaultAuthenticator is set to REQUIRED; if another authenticator is configured, the DefaultAuthenticator must be set to SUFFICIENT or OPTIONAL instead, if it is to be retained. SUFFICIENT is the recommended setting, for the reasons outlined.

C.1.4.3 Users Cannot Log In to Oracle Business Intelligence When Oracle Web Services Manager is not Working

Oracle Web Services Manager (OWSM) secures the BI Security Service, so if OWSM is not working, then nothing can call the BI Security Service, and authentication cannot succeed until this issue is resolved.

Common causes of OWSM failure are:

- Section C.1.4.3.1, "Database Issues - OWSM Cannot Retrieve Policies"

  Issues connecting to the MDS-OWSM schema created on install.

- Section C.1.4.3.2, "OracleSystemUser Issues - OWSM Cannot Retrieve Policies"

  Issues with the OracleSystemUser account that OWSM uses to access it's resources.

For information about BI System User authentication failure, see Section C.1.4.4, "Users Cannot Log In to Oracle Business Intelligence - Is BI System User Authentication Working?".

C.1.4.3.1 Database Issues - OWSM Cannot Retrieve Policies

OWSM stores its metadata, including its policy definitions, in an OWSM subsection of the MDS schema. It accesses this metadata using a connection pool created on install, named mds-owsm. If there is a problem accessing the schema (for example, if the database is not available, there are incorrect credentials, or the database account is locked), then Oracle Business Intelligence authentication fails.

You see an error message like the following one in the Managed Server diagnostic log:

```
[2011-06-28T14:59:27.903+01:00] [bi_server1] [ERROR] []
[oracle.wsm.policymanager.bean.util.PolicySetBuilder] [tid: RTD_
Worker_2] [userId: <anonymous>] [ecid:
de7dd0dc53f3d0ed:11d7f503:130d6771345:-8000-0000000000000003,0]
[APP: OracleRTD#11.1.1] The policy referenced by URI
"oracle/wss_username_token_client_policy" could not be retrieved
as connection to Policy Manager cannot be established at
"t3://biserver:7001,biserver:9704" due to invalid configuration
or inactive state.
```

In addition, you see multiple errors related to a failure to establish or create the connection pool for the data source in the Administration Server logs.

To correct this issue, you must check the following:

- Is the database schema you specified for the MDS-OWSM data source available?
- Did you specify the correct credentials?
- Can you access the schema using standard database tools (for example, SQL Plus, Jdeveloper DB tools) using those credentials?
- Is the mds-owsm data source configured correctly?

To test the MDS-OWSM data source:

1. Log in to Oracle WebLogic Server Administration Console.
2. Click **Services** in the left hand pane and click **Data Sources**.
3. Display the Configuration page and click **mds-owsm**.
4. Select the Monitoring tab and display the Testing page.
5. Select a server and click **Test Data Source**.

To configure the MDS-OWSM data source:

1. Log in to Oracle WebLogic Server Administration Console, and click **Lock & Edit** in the Change Center.
2. Click **Services** in the left hand pane and click **Data Sources**.
3. Display the Configuration page and click **mds-owsm**.
4. Select the Configuration tab and display the Connection Pool page.
5. Configure appropriate changes.
6. Click **Save** to save your changes.
7. In the Change Center, click **Activate Changes**.
8. Restart WebLogic Server and Oracle Business Intelligence components.

C.1.4.3.2 OracleSystemUser Issues - OWSM Cannot Retrieve Policies

By default, OWSM uses the OracleSystemUser account to retrieve policies. If the account is missing, and cannot be authenticated or does not have the correct WebLogic global role assignments, this causes failures.

You see a log message like the following one in the Managed server diagnostic logs:

```
 [2011-06-28T14:59:27.903+01:00] [bi_server1] [ERROR] []
[oracle.wsm.policymanager.bean.util.PolicySetBuilder] [tid: RTD_
Worker_2] [userId: <anonymous>] [ecid:
de7dd0dc53f3d0ed:11d7f503:130d6771345:-8000-0000000000000003,0]
[APP: OracleRTD#11.1.1] The policy referenced by URI
"oracle/wss_username_token_client_policy" could not be retrieved
as connection to Policy Manager cannot be established at
"t3://biserver:7001,biserver:9704" due to invalid configuration
or inactive state.[[
```

After this entry, if the problem is that OWSM is not in the OracleSystemRole WebLogic global role, you see the following log entry:

```
java.rmi.AccessException: [EJB:010160]Security Violation: User:
'OracleSystemUser' has insufficient permission to access EJB:
```

```
type=<ejb>, application=wsm-pm, module=wsm-pmserver-wls.jar,
ejb=DocumentManager, method=retrieveDocuments,
methodInterface=Remote,
signature={java.lang.String,java.util.Map}.
```

You must ensure that the OracleSystemUser is a member of the OracleSystemGroup group in your identity store and that the group has the WebLogic global role OracleSystemRole assigned to it. For more information, see Steps 3-6 in Section 3.4.7.1, "Configuring Oracle Internet Directory LDAP Authentication as the Only Authenticator" (these steps still apply for other LDAP servers):

Alternately, if the problem is that the OracleSystemUser account cannot not be authenticated or does not exist (for example, because you migrated to an LDAP identity store and removed DefaultAuthenticator without creating a new OracleSystemUser account in your new identity store), you see a log entry like this:

```
Caused by: javax.security.auth.login.FailedLoginException:
[Security:090304]Authentication Failed: User OracleSystemUser
javax.security.auth.login.FailedLoginException:
[Security:090302]Authentication Failed: User OracleSystemUser
denied

at
weblogic.security.providers.authentication.LDAPAtnLoginModuleImp
l.login(LDAPAtnLoginModuleImpl.java:261)
```

This error message can be caused by several different issues:

- You have removed the DefaultAuthenticator and not created an account named OracleSystemUser in the new identity store you are using instead.
- You have misconfigured the authenticator for your new identity store such that the OracleSystemUser account cannot be found.
- The OracleSystemUser account has been locked or disabled in some way on your LDAP server.

Check the system for each of the possible causes, reconfigure and restart the system if needed, before retrying.

C.1.4.4 Users Cannot Log In to Oracle Business Intelligence - Is BI System User Authentication Working?

The BI System User account (named BISystemUser by default) is critical to the functioning of the BI Security Service and Oracle Business Intelligence authentication as a whole. The BI System User account authenticates the calls that the BI Server makes to the BI Security Service when it is trying to check the credentials the user has supplied when logging in. If this call fails, then Oracle Business Intelligence cannot authenticate user logins against Fusion Middleware security (that is, users in an identity store configured through WebLogic), the preferred mechanism. BI System User authentication can fail with the following error message in the BI Server nqserver.log:

```
[2011-06-28T11:30:36.000+00:00] [OracleBIServerComponent]
[ERROR:1] [] [] [ecid:
c594c519d241c3b9:-2173cea0:130d2098159:-8000-00000000000019ba]
[tid: 4734ba0] Error Message From BI Security Service:
FailedAuthentication : The security token cannot be
authenticated.
```

```
[2011-06-28T11:30:36.000+00:00] [OracleBIServerComponent]
[ERROR:1] [] [] [ecid:
c594c519d241c3b9:-2173cea0:130d2098159:-8000-00000000000019ba]
[tid: 4734ba0] [nQSError: 43126] Authentication failed: invalid
user/password.
```

You also see a corresponding entry in the Managed Server diagnostic log like this:

```
[2011-06-27T11:06:46.698-07:00] [bi_server1] [NOTIFICATION] []
[oracle.bi.security] [tid: [ACTIVE].ExecuteThread: '0' for
queue: 'weblogic.kernel.Default (self-tuning)'] [userId:
BISystemUser] [ecid:
004dfIJ^08LATOB[2011-06-28T04:27:48.011-07:00] [bi_server1]
[ERROR] [WSM-00008] [oracle.wsm.resources.security] [tid:
[ACTIVE].ExecuteThread: '2' for queue: 'weblogic.kernel.Default
(self-tuning)'] [userId: <anonymous>] [ecid:
c594c519d241c3b9:-2173cea0:130d2098159:-8000-00000000000019a1,0:
1:1:8:1] [WSM_POLICY_NAME: oracle/wss_username_token_service_
policy] [APP: bimiddleware#11.1.1] Web service authentication
failed.[[

javax.security.auth.login.LoginException:
[Security:090303]Authentication Failed: User BISystemUser
weblogic.security.providers.authentication.LDAPAtnDelegateExcept
ion: [Security:090295]caught unexpected exception

at
oracle.security.jps.internal.jaas.module.authentication.JpsUserA
uthenticationLoginModule.login(JpsUserAuthenticationLoginModule.
java:71)
```

This message indicates that OWSM does not allow the call from the BI Server to the BI Security Service to succeed because it cannot authenticate the credentials supplied by the BI Server (not the end user on login) as being valid. The BI Server retrieves the credentials it uses for this call from the credential store by looking in the oracle.bi.system map for the system.user key. These are the credentials that are being authenticated by OWSM.

The following list shows the possible reasons behind failures with the BISystemUser account:

- Section C.1.4.4.1, "The BI System User Account Does Not Exist, Does Not Have Correct Roles, or is Not Synchronized with the Credential Store"
  - You removed the DefaultAuthenticator because you are using an external LDAP store and did not create the BI System User account in the new identity store.
  - You changed the account specified in the system.user key in the credential store but the new account does not have the correct roles.
  - You changed the password of the account specified but have not updated the credential store with the new credentials (or not restarted the system afterwards).
- Section C.1.4.4.2, "Problem With BI System User Account in Underlying Identity Store"

  The account specified as the BI System User account has been locked or the password expired (there is a problem with the account in the underlying identity store).

- Section C.1.4.4.3, "Embedded WebLogic LDAP Replication of BI System User Credential Change Failed"

  You have changed the password of the account specified and replication to the Managed Server has failed (this only applies when you use the DefaultAuthenticator with WebLogic embedded LDAP).

C.1.4.4.1 The BI System User Account Does Not Exist, Does Not Have Correct Roles, or is Not Synchronized with the Credential Store You can resolve these BISystemUser account issues by following the instructions in Section 3.7, "Configuring a New Trusted User (BISystemUser)".

The account specified must have the BISystem application role and the WebLogic Global Admin role, and the system.user key in the credential store must be updated with the new account name and password.

Once you complete these steps restart all Oracle Business Intelligence components and the WebLogic Managed Servers and Administration Server to synchronize the Oracle Business Intelligence components with the BI Security Service otherwise the problems may persist.

C.1.4.4.2 Problem With BI System User Account in Underlying Identity Store It is not uncommon for some LDAP servers to be configured to lock a user account after multiple failed authentication attempts. The BI Server automatically presents the BI System User credentials when attempting to communicate with the BI Security Service. If you change your password without re synchronizing the credential store or restarting the services, the BI Server may make multiple failed authentication attempts, and lock the account by accident.

Equally, some servers are configured to require that credentials expire and be reset after a given period of time, which again lead to the BI System User failing authentication.

Check the policies on your LDAP server and make sure that the account has not become locked by mistake or expired.

C.1.4.4.3 Embedded WebLogic LDAP Replication of BI System User Credential Change Failed This scenario occurs rarely and only when the system uses the WebLogic embedded LDAP server through the DefaultAuthenticator. To understand this scenario you need to understand a BI System User password change (using Oracle WebLogic Server Administration Console) is made initially in the Administration Server and then replicated in the Managed Servers. The BI Security Service authenticates against the replicated copy in the Managed Servers. However, if replication in the Managed Servers failure goes unnoticed, and you change the credentials in the credential store to synchronize with the new password, the two will not match. When this situation occurs, a log entry similar to the following is created in the Administration Server log:

```
####<2011/06/09 17:18:17 GMT> <Error> <EmbeddedLDAP> <bisrv01>
<AdminServer> <VDE Replication Thread> <<anonymous>> <>
<3425d20f6361741a:-2e8537d2:130736e27a9:-8000-000000000000000f>
<1307607517792> <BEA-000000> <Agreement 'bi_server1': Error
Transmitting Change#1698- Invalid name:
cn=",ou=groups,ou=myrealm,dc=bifoundation_domain>
```

If you appear to have this problem but you do not see such an entry, the most likely explanation is that there is a genuine mismatch between the credentials stored in the credential store and those in the identity store. Double check that the change was applied correctly in both places and that all services were restarted after the change was made.

C.1.4.5 Users Cannot Log In to Oracle Business Intelligence - Is the External Identity Store Configured Correctly?

If you have configured an external identity store as your primary user population, check the following aspects of the provider configuration:

- The authentication provider which refers to the primary user population must be set first in the order of providers (unless Release 11.1.1.5 (or higher) and virtualization is set to true).
- If the DefaultAuthenticator is still enabled, ensure that both it and the authentication provider referring to the primary user population are set to 'SUFFICIENT'.
- If you set the username attribute to something other than the default, you need to follow the instructions in Section 3.5, "Configuring User and Group Name Attributes In the Identity Store". For example, the OID authentication provider defaults to expecting the UserName attribute to be "cn", but many organizations actually use the attribute "uid" instead. In this instance, follow the instructions to set both username.attr and user.login.attr to uid in the identity store configuration in Fusion Middleware Control.
- If you have reset the GUID attribute in the authentication provider configuration and see the following error message in the Managed Server console (it may also be found in the bi_server1.out file in the Managed Server logs directory), then you need to follow the instructions in Section 3.6, "Configuring the GUID Attribute In the Identity Store":

```
java.security.PrivilegedActionException:
oracle.bi.security.service.SecurityServiceException:
SecurityService::authenticateUserWithLanguage - 'ldapuser'
was authenticated but has an invalid GUID. Caused by:
oracle.bi.security.service.SecurityServiceException:
SecurityService::authenticateUserWithLanguage - 'ldapuser'
was authenticated but has an invalid GUID. Caused by:
oracle.bi.security.service.InvalidUserGUIDException: User
'ldapuser' has an invalid GUID value: 'null'
```

C.1.4.6 Users Can Log In With Any or No Password

In Oracle Business Intelligence Release 10*g*, authentication is managed through the Metadata Repository, and users wanting to authenticate against external database tables can do so using initialization block settings. The facility still exists in Oracle Business Intelligence 11*g*, and unfortunately it is possible to configure these blocks such that the query issued does not check the password of the user. For example, the query:

SELECT USER_ID FROM USERS WHERE USER_ID = ':USER'

only checks the user id and not whether the password is correct. In a scenario where such an initialization block is configured, it can lead to users being able to log in with any (or no) password.

This scenarios also leads to some apparently inconsistent behavior. For example, if user A and B exist in the primary identity store (Oracle Internet Directory), but user B also exists in a database which is referenced by the initialization block described in this section. When user A and user B try to log in using the wrong password they both fail authentication against OID. However, the BI Server will also attempt to run the initialization block for each user. User A fails, but user B logs in successfully because its user name is in the USER_ID column of the USERS table, and the initialization

block query succeeds, despite not checking the user's password. This kind of scenario must be avoided, so if you find an authentication initialization block that behaves in this way you must remove, or alter it.

C.1.4.7 Have Removed Default Authenticator and Cannot Start WebLogic Server

WebLogic Server must be started using administrative user credentials which are associated with the WebLogic (not Oracle Business Intelligence) Global Admin role. When you install Oracle Business Intelligence the installer prompts for administrative user name and password, which are created in the embedded LDAP, and accessed through the DefaultAuthenticator. When you want to move from using the embedded LDAP to using an external LDAP identity store, you create a new WebLogic administrative user in the external store, ensure it has the WebLogic Global Admin role, and remove the DefaultAuthenticator.

However, if you have performed these steps and have not correctly configured the authenticator configuration for the identity store that now contains the user that you want to use to start the WebLogic Server with, then you cannot start the server. The work around is to revert to the configuration settings that existed before you removed the DefaultAuthenticator.

The domain home for your WebLogic BI Domain (unless you specifically requested otherwise on install), is located in:

<MW_HOME>/user_projects/domains/bifoundation_domain/

This directory contains a configuration directory with the configuration file for the overall domain, including any authenticators. When you update the configuration settings, a backup of the main configuration file, config.xml, is created, starting with backup_config.xml and then numbered versions (for example, backup_config7.xml) for each subsequent revision.

Make sure you copy the current config.xml and the most recent backup_config xml file in case you run into problems. To restore your configuration, replace the current config.xml file with the most recent backup_config xml file, and restart WebLogic Server and all Oracle Business Intelligence components. When WebLogic Server restarts, the DefaultAuthenticator will be restored.

C.2 Resolving Inconsistencies with the Identity Store

A number of inconsistencies can develop between a repository, the Oracle BI Presentation Catalog, and an identity store. The following sections describe the usual ways this can occur and how to resolve the inconsistencies.

C.2.1 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 to identify inconsistencies.

C.2.2 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 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 Oracle BI Presentation Services updates the Oracle BI Presentation Catalog to refer to the new user name the next time this user logs in.

C.2.3 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 store, 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 the 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 the new user in the Oracle BI Presentation Catalog.

C.3 Resolving Inconsistencies with the Policy Store

A number of inconsistencies can develop between the Oracle BI Presentation Catalog and the policy store. The following sections describe the usual ways this can occur and how to resolve the inconsistencies.

C.3.1 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.3.2 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 the 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.3.3 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 cannot 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.3.4 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 BI Server fills in any blank GUIDs for application role references with the actual GUID.

C.4 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 5.3.3, "Manually Configuring WebLogic to Use the HTTPS Protocol" and Section 5.3.4.3, "Committing the SSL Configuration Changes".

C.5 Resolving Issues with BI System User Credentials

You might not be able to log in when using a valid user name and password. For example, an error message like the one in Example C–1 is displayed when BISystemUser credentials are not synchronized.

Example C–1 Example bifoundation_domain.log Output When BISystemUser Credentials Are Not Synchronized

```
####<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
```

C.6 Resolving Custom SSO Environment Issues

You might encounter issues when setting up custom SSO environments. For example, when setting up SSO with Windows Native Authentication and Active Directory, or with SiteMinder.

For more information, see article IDs 1287479.1 and 1274953.1 on My Oracle Support at:

https://support.oracle.com

C.7 Resolving RSS Feed Authentication When Using SSO

When attempting to read an Oracle BI RSS feed, trouble authenticating an RSS reader using SSO may stem from the way Oracle SSO is intercepting requests from that particular RSS reader. In this case Oracle cannot control the feed reader application. There are two scenarios, however, where SSO may be supportable:

- Using a browser-based RSS reader like Wizz RSS for Firefox, and using Firefox to log in to SSO before accessing the feed.
- Using Windows integrated authentication with an RSS reader that uses Internet Explorer.

  Firefox can support Windows authentication so you can use it in this case.

You must validate deployment strategies for your environment.

D

Managing Security for Dashboards and Analyses

This appendix explains how to manage security for dashboards and analyses such that users have only:

- Access to objects in the Oracle BI Presentation Catalog that are appropriate to them.
- Access to features and tasks that are appropriate to them.
- Access to saved customizations that are appropriate to them.

This appendix contains the following sections:

D.1 Managing Security for Users of Oracle BI Presentation Services

System administrators must configure a business intelligence system to ensure that all functionality (including administrative functionality) is secured so that only authorized users can access the system to perform appropriate operations. Administrators also must be able to configure the system to secure all middle-tier communications.

This overview section contains the following topics:

D.1.1 Where Are Oracle BI Presentation Services Security Settings Made?

Security settings that affect users of Presentation Services are made in the following Oracle Business Intelligence components:

- **Oracle BI Administration Tool** — Enables you to perform the following tasks:

  - Set permissions for business models, tables, columns, and subject areas.
  - Specify database access for each user.
  - Specify filters to limit the data accessible by users.
  - Set authentication options.

  For information, see *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

- **Oracle BI Presentation Services Administration** — Enables you to set privileges for users to access features and functions such as editing views and creating agents and prompts.
- **Oracle BI Presentation Services** — Enables you to assign permissions for objects in the Oracle BI Presentation Catalog.

  In previous releases, you could assign permissions to objects from the Presentation Services Administration pages. In this release, you set permissions either in the Catalog Manager or the Catalog page of Presentation Services. See *Oracle Fusion Middleware User's Guide for Oracle Business Intelligence Enterprise Edition* for information on assigning permissions in Presentation Services.
- **Catalog Manager** — Enables you to set permissions for Oracle BI Presentation Catalog objects. For information on the Catalog Manager, see "Configuring and Managing the Oracle BI Presentation Catalog" in *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*.

D.1.2 What Are the Security Goals in Oracle BI Presentation Services?

When maintaining security in Presentation Services, you must ensure the following:

- Only the appropriate users can sign in and access Presentation Services. You must assign sign-in rights and authenticate users through the BI Server.

  Authentication is the process of using a user name and password to identify someone who is logging on. Authenticated users are then given appropriate authorization to access a system, in this case Presentation Services. Presentation Services does not have its own authentication system; it relies on the authentication system that it inherits from the BI Server.

  All users who sign in to Presentation Services are granted the AuthenticatedUser Role and any other roles that they were assigned in Fusion Middleware Control.

  For information about authentication, see Section 1.3, "About Authentication".
- Users can access only the objects that are appropriate to them. You apply access control in the form of permissions, as described in *Oracle Fusion Middleware User's Guide for Oracle Business Intelligence Enterprise Edition*.
- Users have the ability to access features and functions that are appropriate to them. You apply user rights in the form of privileges. Example privileges are "Edit systemwide column formats" and "Create agents."

  Users are either granted or denied a specific privilege. These associations are created in a privilege assignment table, as described in Section D.2.3, "Managing Presentation Services Privileges."

You can configure Oracle Business Intelligence to use the single sign-on feature from the Web server. Presentation Services can use this feature when obtaining information for end users. For complete information on single sign-on, see Chapter 4, "Enabling SSO Authentication".

D.1.3 How Are Permissions and Privileges Assigned to Users?

When you assign permissions and privileges in Presentation Services, you can assign them in one of the following ways:

- To application roles — This is the recommended way of assigning permissions and privileges. Application roles provide much easier maintenance of users and their assignments. An application role defines a set of permissions granted to a user or group that has that role in the system's identity store. An application role is assigned in accordance with specific conditions. As such, application roles are granted dynamically based on the conditions present at the time authentication occurs.

  See Section 1.4.1, "About Application Roles" for information on application roles.

- To individual users — You can assign permissions and privileges to specific users, but such assignments can be more difficult to maintain and so this approach is not recommended.

- To Catalog groups — This approach is maintained for backward compatibility with previous releases only.

  See Section D.2.2, "Working with Catalog Groups" for information on Catalog groups.

D.2 Using Oracle BI Presentation Services Administration Pages

You can use the Administration pages in Oracle BI Presentation Services to perform the tasks that are described in the following sections:

- Section D.2.1, "Understanding the Administration Pages"
- Section D.2.2, "Working with Catalog Groups"
- Section D.2.3, "Managing Presentation Services Privileges"
- Section D.2.4, "Managing Sessions in Presentation Services"

D.2.1 Understanding the Administration Pages

The main Administration page contains links that allow you to display other administration pages for performing various functions, including those related to users in Presentation Services. You can obtain information about all these pages by clicking the Help button in the upper-right corner.

Note: Use care if multiple users have access to the Administration pages, because they can overwrite each other's changes. Suppose UserA and UserB are both accessing and modifying the Manage Privileges page in Presentation Services Administration. If UserA saves updates to privileges while UserB is also editing them, then UserB's changes are overwritten by those that UserA saved.

D.2.2 Working with Catalog Groups

In previous releases, Catalog groups were used for organizing users. Catalog group membership was used to determine the permissions and privileges that are associated with a user, either by explicit assignment or inheritance. In this release, Catalog groups have the following characteristics:

- Are referred to as Catalog groups.
- Can contain users, application roles, or other Catalog groups.
- Exist only for the purposes of compatibility with previous releases and only with Presentation Services.
- No longer have their own passwords.

While you can continue to use Catalog groups, it is recommended that you move to the use of application roles rather than Catalog groups for organizing users.

Presentation Services administrators must ensure that the names of Catalog groups are different from any user IDs that are used to log in to Oracle BI Presentation Services. If a user and a Catalog group share the same name, then the user receives an Invalid Account message when attempting to log in to Oracle BI Presentation Services.

On the Administration page in Presentation Services, you can perform the tasks that are described in the following sections:

- Section D.2.2.1, "Creating Catalog Groups"
- Section D.2.2.2, "Deleting Catalog Groups"
- Section D.2.2.3, "Editing Catalog Groups"

D.2.2.1 Creating Catalog Groups

To create Catalog groups:

1. From the Home page in Presentation Services, select **Administration**.
2. Click the **Manage Catalog Groups** link.
3. Click **Create a New Catalog Group**.
4. In the Add Group dialog, enter a name for the group.
5. Use the shuttle control to select the Catalog groups, users, and application roles to include in this group.

   Tip: It is best practice to not include application roles in Catalog groups, to avoid complex group inheritance and maintenance situations. In particular do not add the AuthenticatedUser Role to any other Catalog groups that you create. This ensures that only the desired Catalog groups (and users) have the specified permissions and privileges, by preventing users or authenticated users from unintentionally inheriting permissions and privileges from another Catalog group.

6. Click **OK**.

D.2.2.2 Deleting Catalog Groups

To delete Catalog groups:

1. From the Home page in Presentation Services, select **Administration**.
2. Click the **Manage Catalog Groups** link.
3. On the Manage Catalog Groups page, select the one or more groups to delete.

   To help you locate the group that you want, enter text in the **Name** field and click **Search**.

4. Click **Delete Selected Groups**.

5. Click **OK** to confirm the deletion.

D.2.2.3 Editing Catalog Groups

To edit Catalog groups:

1. From the Home page in Presentation Services, select **Administration**.
2. Click the **Manage Catalog Groups** link.
3. On the Manage Catalog Groups page, select the group to edit.

   To help you locate the group that you want, enter text in the **Name** field and click **Search**.

   You can click the **More Groups** button to display the next 25 groups in the list.
4. In the Edit Group dialog, change the name or add or remove application roles, Catalog groups, and users.
5. Click **OK**.

D.2.3 Managing Presentation Services Privileges

This section contains the following topics about Presentation Services privileges:

- Section D.2.3.1, "What Are Presentation Services Privileges?"
- Section D.2.3.2, "Setting Presentation Services Privileges for Application Roles"
- Section D.2.3.3, "Default Presentation Services Privilege Assignments"

D.2.3.1 What Are Presentation Services Privileges?

Presentation Services privileges control the rights that users have to access the features and functionality of Presentation Services. Privileges are granted or denied to specific application roles, individual users, and Catalog groups using a privilege assignment table.

Like permissions, privileges are either explicitly set or are inherited through role or group membership. Explicitly denying a privilege takes precedence over any granted, inherited privilege. For example, if a user is explicitly denied access to the privilege to edit column formulas, but is a member of an application role that has inherited the privilege, then the user cannot edit column formulas.

Privileges are most commonly granted to the BIAuthor or BIConsumer roles. This allows users access to common features and functions of Presentation Services. While you can continue to grant privileges to Catalog groups, it is recommended that you switch the grants to application roles.

D.2.3.2 Setting Presentation Services Privileges for Application Roles

You can set Presentation Services privileges for application roles, individual users, and Catalog groups from the Presentation Services Administration Manage Privileges page.

For more information, see Section 2.6.3, "Setting Presentation Services Privileges for Application Roles".

D.2.3.3 Default Presentation Services Privilege Assignments

Table D–1 lists the privileges that you can manage, along with the application role that is granted access to that privilege by default.

These privileges apply to the Oracle Business Intelligence infrastructure. If your organization uses prebuilt applications, then some privileges might be preconfigured. For more information, see the documentation for the application.

Table D–1 Privileges and Default Settings for the Oracle Business Intelligence Infrastructure

Component	Privilege	Description	Default Role Granted
Access	Access to Dashboards	Allows users to view dashboards.	BIConsumer
Access	Access to Answers	Allows users to access the Analysis editor.	BIAuthor
Access	Access to BI Composer	Allows users to access the BI Composer wizard.	BIAuthor
Access	Access to Delivers	Allows users to create and edit agents.	BIAuthor
Access	Access to Briefing Books	Allows users to view and download briefing books.	BIConsumer
Access	Access to Mobile	Allows users to access Presentation Services from the Oracle Business Intelligence Mobile application.	BIConsumer
Access	Access to Administration	Allows users to access the Administration pages in Presentation Services.	BIAdministrator
Access	Access to Segments	Allows users to access segments in Oracle's Siebel Marketing.	BIConsumer
Access	Access to Segment Trees	Allows users to access segment trees in Oracle's Siebel Marketing.	BIAuthor
Access	Access to List Formats	Allows users to access list formats in Oracle's Siebel Marketing.	BIAuthor
Access	Access to Metadata Dictionary	Allows users to access the metadata dictionary information for subject areas, folders, columns, and levels. For more information, see "Providing Access to Metadata Dictionary Information" in *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*.	BIAdministrator
Access	Access to Oracle BI for Microsoft Office	See Section D.2.3.3.2, "Access to Oracle BI for Microsoft Office Privilege."	BIConsumer
Access	Access to Oracle BI Client Installer	Allows users to download the Oracle BI Client Tools installer, which installs the Business Intelligence Administration Tool and the Oracle Business Intelligence Job Manager.	BIConsumer
Access	Catalog Preview Pane UI	Allows users access to the catalog preview pane, which shows a preview of each catalog object's appearance.	BIConsumer
Access	Access to KPI Builder	Allows users to create KPIs.	BIAuthor
Access	Access to Scorecard	Allows users access to Oracle BI Scorecard.	BIConsumer
Actions	Create Navigate Actions	See Section D.2.3.3.1, "Access to Oracle BI Enterprise Edition Actions."	BIAuthor
Actions	Create Invoke Actions	See Section D.2.3.3.1, "Access to Oracle BI Enterprise Edition Actions."	BIAuthor

Table D–1 (Cont.) Privileges and Default Settings for the Oracle Business Intelligence Infrastructure

Component	Privilege	Description	Default Role Granted
Actions	Save Actions Containing Embedded HTML	See Section D.2.3.3.1, "Access to Oracle BI Enterprise Edition Actions."	BIAdministrator
Admin: Catalog	Change Permissions	Allows users to modify permissions for catalog objects.	BIAuthor
Admin: Catalog	Toggle Maintenance Mode	Shows the Toggle Maintenance Mode link on the Presentation Services Administration page, which allows users to turn maintenance mode on and off. In maintenance mode, the catalog is read-only; no one can write to it.	BIAdministrator
Admin: General	Manage Sessions	Shows the Manage Sessions link on the Presentation Services Administration page, which displays the Manage Sessions page in which users manage sessions.	BIAdministrator
Admin: General	Manage Dashboards	Allows users to create and edit dashboards, including editing their properties.	BIAdministrator
Admin: General	See Session IDs	Allows users to see session IDs on the Manage Sessions page.	BIAdministrator
Admin: General	Issue SQL Directly	Shows the Issue SQL link on the Presentation Services Administration page, which displays the Issue SQL page in which users enter SQL statements.	BIAdministrator
Admin: General	View System Information	Allows users to view information about the system at the top of the Administration page in Presentation Services.	BIAdministrator
Admin: General	Performance Monitor	Allows users to monitor performance.	BIAdministrator
Admin: General	Manage Agent Sessions	Shows the Manage Agent Sessions link on the Presentation Services Administration page, which displays the Manage Agent Sessions page in which users manage agent sessions.	BIAdministrator
Admin: General	Manage Device Types	Shows the Manage Device Types link on the Presentation Services Administration page, which displays the Manage Device Types page in which users manage device types for agents.	BIAdministrator
Admin: General	Manage Map Data	Shows the Manage Map Data link on the Presentation Services Administration page, which displays the Manage Map Data page in which users edit layers, background maps, and images for map views.	BIAdministrator
Admin: General	See Privileged Errors	Allows users to see privileged error messages. Users can see detailed error messages about database connections or other details when lower level components fail.	BIAdministrator
Admin: General	See SQL Issued in Errors	Allows users to see SQL statements that are returned by the BI Server in error messages.	BIConsumer

Table D–1 (Cont.) Privileges and Default Settings for the Oracle Business Intelligence Infrastructure

Component	Privilege	Description	Default Role Granted
Admin: General	Manage Marketing Jobs	Shows the Manage Marketing Jobs link on the Presentation Services Administration page, which displays the Marketing Job Management page in which users manage marketing jobs.	BIAuthor
Admin: General	Manage Marketing Defaults	Shows the Manage Marketing Defaults link on the Presentation Services Administration page, which displays the Manage Marketing Defaults page in which users manage defaults for Oracle's Siebel Marketing application.	BIAdministrator
Admin: Security	Manage Catalog Groups	Shows the Manage Catalog Groups link on the Presentation Services Administration page, which displays the Manage Catalog Groups page in which users edit Catalog groups.	BIAdministrator
Admin: Security	Manage Privileges	Shows the Manage Privileges link on the Presentation Services Administration page, which displays the Manage Privileges page in which users manage the privileges that are described in this table.	BIAdministrator
Admin: Security	Set Ownership of Catalog Objects	Allows users to take ownership of catalog items that they did not create and do not own. Shows the "Set ownership of this item" link for individual objects and the "Set ownership of this item and all subitems" link for folders on the Properties page.	BIAdministrator
Admin: Security	User Population - Can List Users	Allows users to see the list of users for which they can perform tasks such as assigning privileges and permissions.	BIConsumer, BISystem
Admin: Security	User Population - Can List Groups	Allows users to see the list of groups for which they can perform tasks such as assigning privileges and permissions.	BIConsumer, BISystem
Briefing Book	Add To or Edit a Briefing Book	Allows users to see the **Add to Briefing Book** link on dashboard pages and analyses and the **Edit** link in briefing books.	BIAuthor
Briefing Book	Download Briefing Book	Allows users to download briefing books.	BIConsumer
Catalog	Personal Storage	Allows users to have write access to their own My Folders folders and create content there. If users do not have this privilege, then they can receive email alerts but cannot receive dashboard alerts.	BIConsumer
Catalog	Reload Metadata	Allows users to click the **Reload Server Metadata** link from the Refresh menu in the toolbar of the Subject Areas pane.	BIAdministrator
Catalog	See Hidden Items	Allows users to see hidden items in catalog folders. Users can also select the **Show Hidden Items** box on the Catalog page.	BIAuthor
Catalog	Create Folders	Allows users to create folders in the catalog.	BIAuthor
Catalog	Archive Catalog	Allows users to archive the folders and objects in the catalog.	BIAdministrator

Table D–1 (Cont.) Privileges and Default Settings for the Oracle Business Intelligence Infrastructure

Component	Privilege	Description	Default Role Granted
Catalog	Unarchive Catalog	Allows users to unarchive catalog objects that have been archived previously.	BIAdministrator
Catalog	Upload Files	Allows users to upload files into an existing catalog.	BIAdministrator
Catalog	Perform Global Search	Allows user to search the catalog using the basic catalog search, which is included by default with the Oracle BI Enterprise Edition installation.	BIAuthor
Catalog	Perform Extended Search	Allows users to search the catalog using the full-text search. To provide full-text search, the administrator must have integrated Oracle BI Enterprise Edition with Oracle Secure Enterprise Search.	BIAuthor
Conditions	Create Conditions	Allows users to create or edit named conditions.	BIAuthor
Dashboards	Save Customizations	See Section D.5, "Controlling Access to Saved Customization Options in Dashboards."	BIConsumer
Dashboards	Assign Default Customizations	See Section D.5, "Controlling Access to Saved Customization Options in Dashboards."	BIAuthor
Formatting	Save SystemWide Column Formats	Allows users to save systemwide defaults when specifying formats for columns.	BIAdministrator
Home and Header	Access Home Page	Allows users to access the home page from the global header.	BIConsumer
Home and Header	Access Catalog UI	Allows users to access the catalog from the global header.	BIConsumer
Home and Header	Access Catalog Search UI	Allows users to access the search fields from the global header.	BIConsumer
Home and Header	Simple Search Field	Allows users to access the **Search** field in the global header.	BIConsumer
Home and Header	Advanced Search Link	Allows users to access the Advanced link in the global header.	BIConsumer
Home and Header	Open Menu	Allows users to access the **Open** menu from the global header.	BIConsumer
Home and Header	New Menu	Allows users to access the **New** menu from the global header.	BIConsumer
Home and Header	Help Menu	Allows users to access the **Help** menu from the global header.	BIConsumer
Home and Header	Dashboards Menu	Allows users to access the **Dashboards** menu from the global header.	BIConsumer
Home and Header	Favorites Menu	Allows users to access the **Favorites** menu from the global header.	BIConsumer
Home and Header	My Account Link	Allows users to access the **My Account** link when they click on their **Signed In As** name in the global header.	BIConsumer

Table D–1 (Cont.) Privileges and Default Settings for the Oracle Business Intelligence Infrastructure

Component	Privilege	Description	Default Role Granted
Home and Header	Custom Links	Allows users to access the custom links that the administrator added to the global header.	BIConsumer
My Account	Access to My Account	Allows users to access the My Account dialog.	BIConsumer
My Account	Change Preferences	Allows users to access the Preferences tab of the My Account dialog.	BIConsumer
My Account	Change Delivery Options	Allows users to access the Delivery Options tab of the My Account dialog.	BIConsumer
Answers	Create Views	Allows users to create views.	BIAuthor
Answers	Create Prompts	Allows users to create prompts.	BIAuthor
Answers	Access Advanced Tab	Allows users to access the Advanced tab in the Analysis editor.	BIAuthor
Answers	Edit Column Formulas	Allows users to edit column formulas.	BIAuthor
Answers	Save Content with HTML Markup	See Section D.2.3.3.3, "Save Content with HTML Markup Privilege."	BIAdministrator
Answers	Enter XML and Logical SQL	Allows users to use the Advanced SQL tab.	BIAuthor
Answers	Edit Direct Database Analysis	Allows users to create and edit requests that are sent directly to the back-end data source.	BIAdministrator
Answers	Create Analysis from Simple SQL	Allows users to select the **Create Analysis from Simple SQL** option in the Select Subject Area list.	BIAdministrator
Answers	Create Advanced Filters and Set Operations	Allows users to click the **Combine results based on union, intersection, and difference operations** button from the Criteria tab in the Analysis editor.	BIAuthor
Answers	Save Filters	Allows users to save filters.	BIAuthor
Answers	Execute Direct Database Analysis	Allows users to issue requests directly to the back-end data source.	BIAdministrator
Delivers	Create Agents	Allows users to create agents.	BIAuthor
Delivers	Publish Agents for Subscription	Allows users to publish agents for subscription.	BIAuthor
Delivers	Deliver Agents to Specific or Dynamically Determined Users	Allows users to deliver agents to other users.	BIAdministrator
Delivers	Chain Agents	Allows users to chain agents.	BIAuthor
Delivers	Modify Current Subscriptions for Agents	Allows users to modify the current subscriptions for agents, including unsubscribing users.	BIAdministrator
Proxy	Act As Proxy	Allows users to act as proxy users for other users, as described in Section D.6, "Enabling Users to Act for Others."	Denied: BIConsumer

Table D–1 (Cont.) Privileges and Default Settings for the Oracle Business Intelligence Infrastructure

Component	Privilege	Description	Default Role Granted
RSS Feeds	Access to RSS Feeds	Allows users to subscribe to and receive RSS feeds with alerts and contents of folders. If Presentation Services uses the HTTPS protocol, then the RSS Reader that you use must also support the HTTPS protocol.	BIAuthor
Scorecard	Create/Edit Scorecards	Allows users to create and edit scorecards.	BIAuthor
Scorecard	View Scorecards	Allows users to view scorecards.	BIConsumer
Scorecard	Create/Edit Objectives	Allows users to create and edit objectives.	BIAuthor
Scorecard	Create/Edit Initiatives	Allows users to create and edit initiatives.	BIAuthor
Scorecard	Create Views	Allows users to create and edit scorecard objects that present and analyze corporate strategy, such as vision and mission statements, strategy maps, cause & effect maps, and so on.	BIAuthor
Scorecard	Create/Edit Causes and Effects Linkages	Allows users to create and edit cause and effect relationships.	BIAuthor
Scorecard	Create/Edit Perspectives	Allows users to create and edit perspectives.	BIAdministrator
Scorecard	Add Annotations	Allows users to add comments to KPIs and scorecard components.	BIConsumer
Scorecard	Override Status	Allows users to override statuses of KPIs and scorecard components.	BIConsumer
Scorecard	Create/Edit KPIs	Allows users to create and edit KPIs.	BIAuthor
Scorecard	Write Back to Database for KPI	Allows users to enter and submit a KPI's actual and target settings values to the repository.	BIConsumer
Scorecard	Add Scorecard Views to Dashboards	Allows users to add scorecard views (such as strategy trees) to dashboards.	BIConsumer
List Formats	Create List Formats	Allows users to create list formats in Oracle's Siebel Marketing.	BIAuthor
List Formats	Create Headers and Footers	Allows users to create headers and footers for list formats in Oracle's Siebel Marketing.	BIAuthor
List Formats	Access Options Tab	Allows users to access the Options tab for list formats in Oracle's Siebel Marketing.	BIAuthor
List Formats	Add/Remove List Format Columns	Allows users to add and remove columns for list formats in Oracle's Siebel Marketing.	BIAdministrator
Segmentation	Create Segments	Allows users to create segments in Oracle's Siebel Marketing.	BIAuthor
Segmentation	Create Segment Trees	Allows users to create segment trees in Oracle's Siebel Marketing.	BIAuthor
Segmentation	Create/Purge Saved Result Sets	Allows users to create and purge saved result sets in Oracle's Siebel Marketing.	BIAdministrator
Segmentation	Access Segment Advanced Options Tab	Allows users to access the Segment Advanced Options tab in Oracle's Siebel Marketing.	BIAdministrator

Table D–1 (Cont.) Privileges and Default Settings for the Oracle Business Intelligence Infrastructure

Component	Privilege	Description	Default Role Granted
Segmentation	Access Segment Tree Advanced Options Tab	Allows users to access the Segment Tree Advanced Options tab in Oracle's Siebel Marketing.	BIAdministrator
Segmentation	Change Target Levels within Segment Designer	Allows users to change target levels within the Segment Designer in Oracle's Siebel Marketing.	BIAdministrator
Mobile	Enable Local Content	Allows users of Oracle Business Intelligence Mobile to save local copies of BI content to their mobile devices.	BIConsumer
Mobile	Enable Search	Allows users of Oracle Business Intelligence Mobile to search the catalog.	BIConsumer
SOAP	Access SOAP	Allows users to access various Web services.	BIConsumer, BISystem
SOAP	Impersonate as System User	Allows users to impersonate a system user using a Web service.	BISystem
SOAP	Access MetadataService Service	Allows users to access the MetadataService Web service.	BIConsumer, BISystem
SOAP	Access AnalysisExportViewsService Service	Allows users to access the ReportingEditingService Web service.	BIConsumer
SOAP	Access ReportingEditingService Service	Allows users to access the ReportingEditingService Web service.	BIConsumer, BISystem
SOAP	Access ConditionEvaluationService Service	Allows users to access the ConditionEvaluationService Web service.	BIConsumer, BISystem
SOAP	Access ReplicationService Service	Allows users to access the ReplicationService Web service to replicate the Oracle BI Presentation Catalog.	BISystem
SOAP	Access CatalogIndexingService Service	Allows users to access the CatalogIndexingService Web service to index the Oracle BI Presentation Catalog for use with full-text search.	BISystem
SOAP	Access DashboardService Service	Allows users to access the DashboardService Web service.	BIConsumer, BISystem
SOAP	Access SecurityService Service	Allows users to access the SecurityService Web service.	BIConsumer, BISystem
SOAP	Access Tenant Information	Internal only.	BISystem
SOAP	Access ScorecardMetadataService Service	Allows users to access the ScorecardMetadataService Web service.	BIConsumer, BISystem
SOAP	Access ScorecardAssessmentService Service	Allows users to access the ScorecardAssessmentService Web service.	BIConsumer, BISystem
SOAP	Access HtmlViewService Service	Allows users to access the HtmlViewServiceService Web service.	BIConsumer, BISystem
SOAP	Access CatalogService Service	Allows users to access the CatalogService Web service.	BIConsumer, BISystem

Table D–1 (Cont.) Privileges and Default Settings for the Oracle Business Intelligence Infrastructure

Component	Privilege	Description	Default Role Granted
SOAP	Access IBotService Service	Allows users to access the IBotService Web service.	BIConsumer, BISystem
SOAP	Access XmlGenerationService Service	Allows users to access the XmlGenerationService Web service.	BIConsumer, BISystem
SOAP	Access JobManagementService Service	Allows users to access the JobManagementService Web service.	BIConsumer, BISystem
SOAP	Access KPIAssessmentService Service	Allows users to access the JKPIAssessmentService Web service.	BIConsumer, BISystem
Subject Area (*by its name*)	Access within Oracle BI Answers	Allows users to access the specified subject area within the Answers editor.	BIAuthor
Views	Add/Edit AnalyzerView	Allows users to access the Analyzer view.	BIAdministrator
Views	Add/Edit Column SelectorView	Allows users to create and edit column selector views.	BIAuthor
Views	Add/Edit CompoundView	Allows users to create and edit compound layout views.	BIAuthor
Views	Add/Edit GraphView	Allows users to create and edit graph views.	BIAdministrator
Views	Add/Edit FunnelView	Allows users to create and edit funnel graph views.	BIAuthor
Views	Add/Edit GaugeView	Allows users to create and edit gauge views.	BIAuthor
Views	Add/Edit FiltersView	Allows users to create and edit filter views.	BIAuthor
Views	Add/Edit Dashboard PromptView	Allows users to create and edit dashboard prompt views.	BIAuthor
Views	Add/Edit Static TextView	Allows users to create and edit static text views.	BIAuthor
Views	Add/Edit Legend View	Allows users to create and edit legend views.	BIAuthor
Views	Add/Edit MapView	Allows users to create and edit map views.	BIAuthor
Views	Add/Edit NarrativeView	Allows users to create and edit narrative views.	BIAuthor
Views	Add/Edit No ResultsView	Allows users to create and edit no result views.	BIAuthor
Views	Add/Edit Pivot TableView	Allows users to create and edit pivot table views.	BIAuthor
Views	Add/Edit Report PromptView	Allows users to create and edit prompt views.	BIAuthor
Views	Add/Edit Create SegmentView	Allows users to create and edit segment views.	BIAuthor
Views	Add/Edit SelectionStepsView	Allows users to create and edit selection steps views.	BIAuthor
Views	Add/Edit Logical SQLView	Allows users to create and edit logical SQL views.	BIAuthor

Table D–1 (Cont.) Privileges and Default Settings for the Oracle Business Intelligence Infrastructure

Component	Privilege	Description	Default Role Granted
Views	Add/Edit TableView	Allows users to create and edit table views.	BIAuthor
Views	Add/Edit Create Target ListView	Allows users to create and edit target list views.	BIAuthor
Views	Add/Edit TickerView	Allows users to create and edit ticker views.	BIAuthor
Views	Add/Edit TitleView	Allows users to create and edit title views.	BIAuthor
Views	Add/Edit View SelectorView	Allows users to create and edit view selector views.	BIAuthor
Write Back	Write Back to Database	Grants the right to write data into the data source.	Denied: BIConsumer
Write Back	Manage Write Back	Grants the right to manage write back requests.	BIAdministrator

D.2.3.3.1 Access to Oracle BI Enterprise Edition Actions

You must set the Action privileges, which determine whether the Actions functionality is available to users and specify which user types can create Actions. The following list describes these privileges:

- **Create Navigate Actions** — Determines which users can create a Navigate action type. The sessions of users who are denied this privilege do not contain any of the user interface components that allow them to create Navigate Actions. For example, if a user is denied this privilege and chooses to create an action from the Oracle BI Enterprise Edition global header, the dialog where the user selects an action type does not include the Navigate Actions options (Go to BI Content, Go to a Web Page, and so on). However, users who are denied this privilege can add saved actions to analyses and dashboards. And, users who are denied this privilege can execute an action from an analysis or dashboard that contains an action.
- **Create Invoke Actions** — Determines which users can create an Invoke action type. The sessions of user who are denied this privilege do not contain any of the user interface components that allow them to create Invoke Actions. For example, if a user is denied this privilege and chooses to access the agent editor's Actions tab and clicks the **Add New Action** button, the dialog where the user selects the action type does not include the Invoke Actions options (Invoke a Web Service, Invoke an HTTP Request, and so on). However, users who are denied this privilege can add saved actions to analyses and dashboards. And, users who are denied this privilege can execute an action from an analysis or dashboard that contains an action.
- **Save Actions Containing Embedded HTML** — Determines which users can embed HTML code in the customization of Web service action results. Use care in assigning this privilege, because it poses a security risk to allow users to run HTML code.

D.2.3.3.2 Access to Oracle BI for Microsoft Office Privilege

The Access to Oracle BI for Microsoft Office privilege shows the following option for the **Download BI Desktop Tools** link in the Get Started area of the Oracle BI EE Home page:

- **Oracle BI for MS Office**: Downloads the installation file for the Oracle BI Add-in for Microsoft Office.

The Access to Oracle BI for Microsoft Office privilege does not affect the display of the **Copy** link for analyses. The link is always available there.

The location of the installation file to download for Oracle BI for Microsoft Office is specified by default in the BIforOfficeURL element in the instanceconfig.xml file. For more information on using Oracle BI for Microsoft Office and the **Copy** option, see *Oracle Fusion Middleware User's Guide for Oracle Business Intelligence Enterprise Edition*.

D.2.3.3.3 Save Content with HTML Markup Privilege By default, Presentation Services is secured against cross-site scripting (XSS). Securing against XSS escapes input in fields in Presentation Services and renders it as plain text. For example, an unscrupulous user can use an HTML field to enter a script that steals data from a page.

By default, end users cannot save content that is flagged as HTML; instead only administrators who have the Save Content with HTML Markup privilege can save content that contains HTML code. Users that have the Save Content with HTML Markup privilege can save an image with the "fmap" prefix. If users try to save an image with the "fmap" prefix when they do not have this privilege assigned, then they see an error message.

D.2.4 Managing Sessions in Presentation Services

Using the Session Management page in Presentation Services Administration, you can view information about active users and running analyses, cancel requests, and clear the cache.

To manage sessions in Presentation Services:

1. From the Home page in Presentation Services, select **Administration**.
2. Click the **Manage Sessions** link.

   The Session Management screen is displayed with the following tables:

   – The Sessions table, which gives information about sessions that have been created for users who have logged in:

   – The Cursor Cache table, which shows the status of analyses:

To cancel all running requests:

1. Click **Cancel Running Requests**.
2. Click **Finished**.

To cancel one running analysis:

- In the Cursor Cache table, identify the analysis and click the **Cancel** link in the **Action** column.

  The user receives a message indicating that the analysis was canceled by an administrator.

To clear the Web cache:

1. In the Cursor Cache table, identify the analysis and click **Close All Cursors**.
2. Click **Finished**.

To clear the cache entry associated with an analysis:

- In the Cursor Cache table, identify the analysis and click the **Close** link in the **Action** column.

To view the query file for information about an analysis:

- In the Cursor Cache table, identify the analysis and click the **View Log** link.

  Note: Query logging must be turned on for data to be saved in this log file.

D.3 Inheritance of Permissions and Privileges for Oracle BI Presentation Services

Permissions and privileges can be assigned to users directly or through membership in application roles or Catalog groups. From another perspective, permissions and privileges can be assigned explicitly or effectively. Effective permissions and privileges are assigned indirectly through inheritance from application roles or Catalog groups, which is the recommended approach for assignments.

This section contains the following topics:

D.3.1 Rules for Inheritance for Permissions and Privileges

The following list describes the rules of inheritance for permissions and privileges:

- Any permissions or privileges granted explicitly to a user override any permissions or privileges inherited from the application roles or Catalog groups to which the user belongs.
- If a user belongs to two application roles or Catalog groups and both are granted permissions, then the least restrictive permissions are given to the user.

  For example, if one application role allows Open access and another allows Modify access, then the least restrictive access would be granted; in this example, Open access.

  Note: The exception to this is if one of the two application roles or Catalog groups is explicitly denied the permissions, in which case the user is denied.

- If a user belongs to Application Role X, and Application Role X is a member of Application Role Y, then any permissions assigned to Application Role X override any permissions assigned to Application Role Y. The same holds true if X and Y are Catalog groups.

  For example, if Marketing has Open permissions, Marketing Administrators, which is a member of Marketing, can have Full Control permission.

- If a Catalog group is specified along with an application role in the Permissions dialog in Presentation Services, then the Catalog group takes precedence.

  For example, suppose that for a certain object, the BIAdministrator role has Read-Only permission and the Admin Catalog Group has Full Control permission. If a user signs in who is a member of both the BIAdministrator role and the Admin Catalog Group, then he is granted full access to the object.

- Explicitly denying access takes precedence over any other permissions or privileges.

D.3.2 Example of Inherited Privileges for Application Roles

Figure D–1 shows an example of how privileges are inherited through application roles. At the top of the diagram is a rectangle labeled User1, which specifies that User1 is a member of Role1 and Role2. Attached beneath the User1 rectangle are two more rectangles — one on the left that represents Role1 and one on the right that represents Role2.

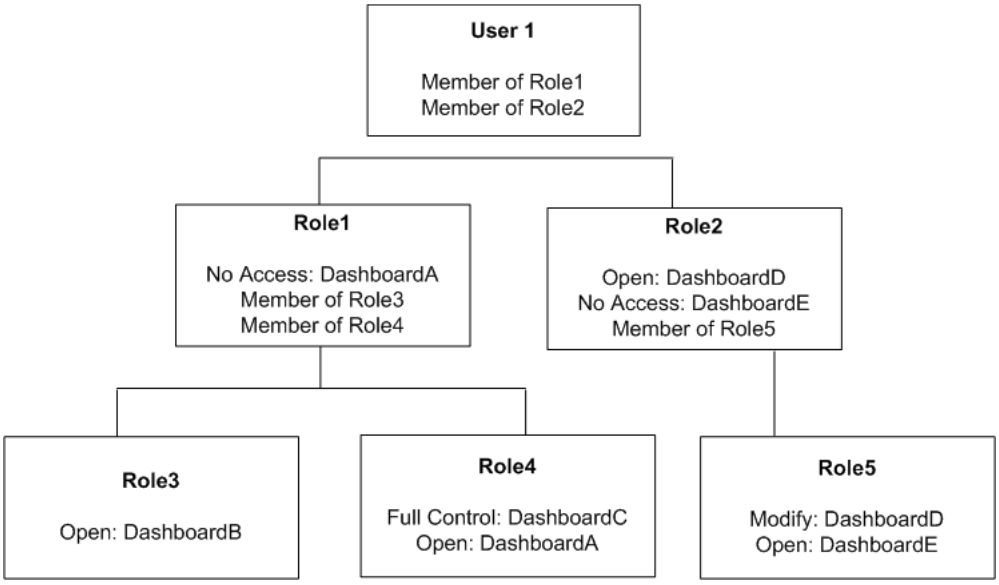
- The Role1 rectangle specifies that Role1 has no access to DashboardA and is a member of Role3 and Role4.
- The Role2 rectangle specifies that Role2 has Open access to DashboardD, is a member of Role5, and has no access to DashboardE.

Attached beneath the Role1 rectangle are two more rectangles — one on the left that represents Role3 and one on the right that represents Role4:

- The Role3 rectangle specifies that Role3 has Open access to DashboardB.
- The Role4 rectangle specifies that Role4 has Full Access to DashboardC and Open access to DashboardA.

And finally, attached beneath the Role2 rectangle is a rectangle that represents Role5. The Role5 rectangle specifies that Role5 has Modify access to DashboardD and Open access to DashboardE.

Figure D–1 Example of Inheritance of Permissions Using Roles



In this example:

- User1 is a direct member of Role1 and Role2, and is an indirect member of Role3, Role4, and Role5.
- The permissions and privileges from Role1 are no access to DashboardA, Open access to DashboardB, and Full Control for DashboardC.
- If one application role is a member of a second application role, then any permissions assigned to the first application role override any permissions

assigned to the first role. Therefore, the inherited permissions and privileges from Role2 include Modify access to DashboardD from Role5.

- Specifically denying access always takes precedence over any other settings. Therefore, Role1's denial of access to DashboardA overrides Role4's Open access. The result is that Role1 has no access to DashboardA. Likewise, Role5 has no access to DashboardE, because access to that dashboard is explicitly denied for Role2.

The total permissions and privileges granted to User1 are as follows:

- No access to DashboardA and DashboardE, because access is specifically denied.
- Open access to DashboardB.
- Full Control for DashboardC.
- Modify access to DashboardD.

D.3.3 Example of Inherited Privileges for Catalog Groups

Any permissions or privileges granted explicitly to a Catalog group take precedence over permissions or privileges granted to an application role. For example, suppose that you have an application role called Marketing_US that has Full Access to the Marketing Dashboard. You want to restrict a small set of the users in the Marketing_US role to not have access to that dashboard. To do so, you create a Catalog group called Marketing_SanJose and add the appropriate users as members of that group. You then deny the Marketing_SanJose Catalog group access to the Marketing Dashboard. Even though those users belong to the Marketing_US role, they are denied access to the Marketing Dashboard.

D.4 Providing Shared Dashboards for Users

This section contains the following topics on providing shared dashboards for users:

- Section D.4.1, "Understanding the Catalog Structure for Shared Dashboards"
- Section D.4.2, "Creating Shared Dashboards"
- Section D.4.3, "Testing the Dashboards"
- Section D.4.4, "Releasing Dashboards to the User Community"

D.4.1 Understanding the Catalog Structure for Shared Dashboards

The Oracle BI Presentation Catalog has two main folders:

- **My Folders** — Contains the personal storage for individual users. Includes a Subject Area Contents folder where you save objects such as calculated items and groups.
- **Shared Folders** — Contains objects and folders that are shared across users. Dashboards that are shared across users are saved in a Dashboards subfolder under a common subfolder under the /Shared Folders folder

---

Note: If a user is given permission to an analysis in the Oracle BI Presentation Catalog that references a subject area to which the user does not have permission, then the BI Server still prevents the user from executing the analysis.

D.4.2 Creating Shared Dashboards

After setting up the Oracle BI Presentation Catalog structure and setting permissions, you can create shared dashboards and content for use by others.

One advantage to creating shared dashboards is that pages that you create in the shared dashboard are available for reuse. Users can create their own dashboards using the pages from your shared dashboards and any new pages that they create. You can add pages and content as described in *Oracle Fusion Middleware User's Guide for Oracle Business Intelligence Enterprise Edition*.

If you plan to allow multiple users to modify a shared default dashboard, then consider putting these users into an application role. For example, suppose that you create an application role called Sales and create a default dashboard called SalesHome. Of the 40 users that have been assigned the Sales application role, suppose that there are three who must have the ability to create and modify content for the SalesHome dashboard. Create a SalesAdmin application role, with the same permissions as the primary Sales application role. Add the three users who are allowed to make changes to the SalesHome dashboard and content to this new SalesAdmin application role, and give this role the appropriate permissions in the Oracle BI Presentation Catalog. This allows those three users to create and modify content for the SalesHome dashboard. If a user no longer requires the ability to modify dashboard content, then you can change the user's role assignment to Sales. If an existing Sales role user must have the ability to create dashboard content, then the user's role assignment can be changed to SalesAdmin.

For more information about creating shared dashboards, see 'Managing Dashboards' in *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*.

D.4.3 Testing the Dashboards

Before releasing dashboards and content to the user community, perform some tests.

To test the dashboard:

1. Verify that users with appropriate permissions can correctly access it and view the intended content.
2. Verify that users without appropriate permissions cannot access the dashboard.
3. Verify that styles and skins are displayed as expected, and that other visual elements are as expected.
4. Correct any problems you find and test again, repeating this process until you are satisfied with the results.

D.4.4 Releasing Dashboards to the User Community

After testing is complete, notify the user community that the dashboard is available, ensuring that you provide the relevant network address.

D.5 Controlling Access to Saved Customization Options in Dashboards

This section provides an overview of saved customizations and information about administering saved customizations. It contains the following topics:

- Section D.5.1, "Overview of Saved Customizations in Dashboards"
- Section D.5.2, "Administering Saved Customizations"

D.5.1 Overview of Saved Customizations in Dashboards

Saved customizations allow users to save and view later dashboard pages in their current state with their most frequently used or favorite choices for items such as filters, prompts, column sorts, drills in analyses, and section expansion and collapse. By saving customizations, users need not make these choices manually each time that they access the dashboard page.

Users and groups with the appropriate permissions and dashboard access rights can perform the following activities:

- Save various combinations of choices as saved customizations, for their personal use or use by others.
- Specify a saved customization as the default customization for a dashboard page, for their personal use or use by others.
- Switch between their saved customizations.

You can restrict this behavior in the following ways:

- Users can view only the saved customizations that are assigned to them.
- Users can save customizations for personal use only.
- Users can save customizations for personal use and for use by others.

For information about end users and saved customizations with dashboards, see *Oracle Fusion Middleware User's Guide for Oracle Business Intelligence Enterprise Edition*.

D.5.2 Administering Saved Customizations

This section describes the privileges and permissions that are required to administer saved customizations. It also describes the relevant portions of the Oracle BI Presentation Catalog that relate to storing and administering saved customizations.

D.5.2.1 Privileges for Saved Customizations

In Oracle BI Presentation Services Administration, the following privileges in the Dashboards area, along with permission settings for key dashboard elements, control whether users or groups can save or assign customizations:

- Save Customizations
- Assign Default Customizations

You can set neither privilege, one privilege, or both privileges for a user or group, depending on the level of access desired. For example, a user who has neither privilege can view only the saved customization that is assigned as his or her default customization.

D.5.2.2 Permissions for Saved Customizations

This section describes the permissions that are required for users to administer saved customizations of dashboard pages, and the relevant portions of the Oracle BI Presentation Catalog structure for setting permissions on shared and personal saved customizations.

D.5.2.2.1 Assigning Permissions to Dashboards

You set permissions for dashboards and pages, such as Full Control or No Access, in the Permission dialog in Oracle BI EE. You assign these permissions in the same manner as for other objects in the catalog.

D.5.2.2.2 Assigning Permissions for Customizations on a Dashboard Page

You set permissions for working with saved customizations on a particular dashboard page in the Dashboard Properties dialog, which is available in the Dashboard Builder. After selecting a page in the list in the dialog, click one of the following buttons:

- **Specify Who Can Save Shared Customizations** displays the Permission dialog in which you specify who can save shared customizations for that dashboard page.
- **Specify Who Can Assign Default Customizations** displays the Permission dialog in which you specify who can assign default customizations for that dashboard page.

Catalog objects and permissions scenarios are described in the following sections.

D.5.2.2.3 Catalog Folder Structure for Saved Customizations

In addition to the privileges that you set in Oracle BI Presentation Services Administration, the level of control that users and groups have over saved customizations depends on their access rights to key elements. For example, users and groups that can create and edit underlying dashboards, save dashboard view preferences as customizations, and assign customizations to other users as default customizations require Full Control permission to the key elements in shared storage, while users and groups that can view only their assigned default saved customizations need only View access to the key elements in shared storage.

Key elements in the catalog include the following folders:

- Shared Storage Folders.

  Shared storage folders for dashboards are typically located within the Dashboards sub-folder of a parent shared folder. Dashboards are identified by their assigned names. You can save a dashboard anywhere in the Oracle BI Presentation Catalog. If you save a dashboard within a subfolder called "Dashboards", then that dashboard's name is displayed in the list of dashboards that is displayed from the Dashboards link in the global header.

  Permission settings control access to a specific dashboard for editing. Typically, if permissions are inherited down to the _selections and Dashboards sub-folders, then users who can edit dashboards can also save customizations and set defaults. Access to a specific dashboard folder controls whether a user or group can edit the dashboard.

  The _selections folder contains a page identifier folder for each dashboard page. Shared saved customizations are located within this folder. Access to the page identifier folder controls whether a user or group can display, save, or edit customizations for that page.

  The _defaults folder within a _selections folder contains assigned default customizations. Each group that has an assigned default is displayed here. Access to this folder controls whether a user or group can assign defaults.

- Personal Storage Folders.

  Within a user's personal folder, the _selections folder contains an individual user's saved customizations. Like the shared _selections folder, a personal _selections folder contains a page identifier folder for each dashboard page. The page identifier folder contains personal saved customizations and a _defaultlink file that specifies a user's preference for the personal defaulted customization.

A personal saved customization default overrides an assigned shared customization default.

Note: If a dashboard page with saved customizations is deleted, then the saved customizations are also deleted from the catalog. If the underlying dashboard structure changes such that a saved customization is no longer valid when a user accesses it, then the default content is displayed on the dashboard.

D.5.3 Permission and Privilege Settings for Creating Saved Customizations

Table D–2 describes typical user roles and specific permission settings that can be granted to users for creating saved customizations. The folder names listed in the Permission and Privilege Settings column are described in the preceding section.

Table D–2 User Roles and Permission Settings for Saved Customizations

User Role	Permission and Privilege Settings
Power users such as IT users who must perform the following tasks: ■ Create and edit underlying dashboards. ■ Save dashboard view preferences as customizations. ■ Assign customizations to other users as default customizations.	In the Shared section of the catalog, requires Full Control permission to the following folders: ■ dashboard_name ■ _selection ■ _defaults Typically, no additional privileges must be assigned.
Technical users such as managers who must perform the following tasks: ■ Save customizations as customizations for personal use. ■ Save customizations for use by others. Users cannot create or edit underlying dashboards, or assign view customizations to others as default customizations.	In the Shared section of the catalog, requires View permission to the following folders: ■ dashboard_name In the Shared section of the catalog, requires Modify permission to the following folders: ■ _selections ■ _defaults Typically, no additional privileges must be assigned.
Everyday users who must save customizations for personal use only.	In Oracle BI Presentation Services Administration, requires the following privilege to be set: ■ Save Customizations In the dashboard page, requires that the following option is set: ■ Allow Saving Personal Customizations In the catalog, no additional permission settings are typically required.

Table D–2 (Cont.) User Roles and Permission Settings for Saved Customizations

User Role	Permission and Privilege Settings
Casual users who must view only their assigned default customization.	In the Shared section of the catalog, the user needs View permission to the following folders: ■ dashboard_name ■ _selections ■ _defaults In the catalog, no additional permission settings are typically required.

D.5.4 Example Usage Scenario for Saved Customization Administration

Depending on the privileges set and the permissions granted, you can achieve various combinations of user and group rights for creating, assigning, and using saved customizations.

For example, suppose a group of power users cannot change dashboards in a production environment, but they are allowed to create saved customizations and assign them to other users as default customizations. The following permission settings for the group are required:

- Open access to the dashboard, using the Catalog page.
- Modify access to the _selections and _defaults subfolders within the dashboard folder in the Oracle BI Presentation Catalog, which you assign using the Dashboard Properties dialog in the Dashboard Builder. After selecting a page in the list in the dialog, click **Specify Who Can Save Shared Customizations** and **Specify Who Can Assign Default Customizations**.

D.6 Enabling Users to Act for Others

This section contains the following topics on enabling users to act for others:

- Section D.6.1, "Why Enable Users to Act for Others?"
- Section D.6.2, "What Are the Proxy Levels?"
- Section D.6.3, "Process of Enabling Users to Act for Others"

D.6.1 Why Enable Users to Act for Others?

You can enable one user to act for another user in Oracle BI Presentation Services. When a user (called the proxy user) acts as another (called the target user), the proxy user can access the objects in the catalog for which the target user has permission.

Enabling a user to act for another is useful, for example, when a manager wants to delegate some of his work to one of his direct reports or when IT support staff wants to troubleshoot problems with another user's objects.

See *Oracle Fusion Middleware User's Guide for Oracle Business Intelligence Enterprise Edition* for information on how users enable others to act for them.

D.6.2 What Are the Proxy Levels?

When you enable a user to be a proxy user, you also assign an authority level (called the proxy level). The proxy level determines the privileges and permissions granted to

the proxy user when accessing the catalog objects of the target user. The following list describes the proxy levels:

- **Restricted** — Permissions are read-only to the objects to which the target user has access. Privileges are determined by the proxy user's account (not the target user's account).

  For example, suppose a proxy user *has not* been assigned the Access to Answers privilege, and the target user *has*. When the proxy user is acting as the target user, the target user *cannot* access Answers.

- **Full** — Permissions and privileges are inherited from the target user's account.

  For example, suppose a proxy user *has not* been assigned the Access to Answers privilege, and the target user *has*. When the proxy user is acting as the target user, the target user *can* access Answers.

When you have enabled a user to act as a proxy user, that user can display the **Act As** option in the global header of Presentation Services to select the target user to act as, provided the Act As Proxy privilege has been set.

> **Tip:** Before a proxy user can act as a target user, the target user must have signed into Presentation Services at least once and accessed a dashboard.

> **Note:** If you are a user who can be impersonated by a proxy user, you can see the users with the permission to proxy (Act As) you. To see these users, log in to Analytics, go to the My Account dialog box and display the extra tab called Delegate Users. This tab displays the users who can connect as you, and the permission they have when they connect as you (Restricted or Full).

D.6.3 Process of Enabling Users to Act for Others

To enable users to act for others, perform the following tasks:

- Section D.6.3.1, "Defining the Association Between Proxy Users and Target Users"
- Section D.6.3.2, "Creating Session Variables for Proxy Functionality"
- Section D.6.3.3, "Modifying the Configuration File Settings for Proxy Functionality"
- Section D.6.3.4, "Creating a Custom Message Template for Proxy Functionality"
- Section D.6.3.5, "Assigning the Proxy Privilege"
- Section D.6.3.6, "Assigning the manageRepositories Permission"

D.6.3.1 Defining the Association Between Proxy Users and Target Users

You define the association between proxy users and target users in the database by identifying, for each proxy user/target user association, the following:

- ID of the proxy user
- ID of the target user
- Proxy level (either full or restricted)

For example, you might create a table called Proxies in the database that looks like this:

proxyId	targetId	proxyLevel
Ronald	Eduardo	full
Timothy	Tracy	restricted
Pavel	Natalie	full
William	Sonal	restricted
Maria	Imran	restricted

After you define the association between proxy users and target users, you must import the schema to the physical layer of the BI Server. For information on importing a schema, see *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

D.6.3.2 Creating Session Variables for Proxy Functionality

To authenticate proxy users, you must create the following two session variables along with their associated initialization blocks. For both variables, you must modify the sample SQL statement according to the schema of the database.

- **PROXY** — Use this variable to store the name of the proxy user.

  Use the initialization block named ProxyBlock and include code such as the following:

  ```
  select targetId
  from Proxies
  where 'VALUEOF(NQ_SESSION.RUNAS)'=targetId and ':USER'=proxyId
  ```

- **PROXYLEVEL** — Use this optional variable to store the proxy level, either Restricted or Full. If you do not create the PROXYLEVEL variable, then the Restricted level is assumed.

  Use the initialization block named ProxyLevel and include code such as the following:

  ```
  select proxyLevel
  from Proxies
  where 'VALUEOF(NQ_SESSION.RUNAS)'=targetId and ':USER'=proxyId
  ```

For more information on creating session variables, see *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

D.6.3.3 Modifying the Configuration File Settings for Proxy Functionality

Use various elements in the instanceconfig.xml file to configure the proxy functionality.

Before you begin this procedure, ensure that you are familiar with the information in 'Using a Text Editor to Update Oracle Business Intelligence Configuration Settings' in *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*.

To manually configure for proxy functionality:

1. Open the instanceconfig.xml file for editing, as described in 'Where are Configuration Files Located' in *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*.
2. Locate the section in which you must add the elements that are described in the following list:
   - LogonParam: Serves as the parent element for the TemplateMessageName and MaxValues elements.
   - TemplateMessageName: Specifies the name of the custom message template in the Custom Messages folder that contains the SQL statement to perform tasks related to displaying proxy and target users. The default name is LogonParamSQLTemplate.

     The name that you specify in the TemplateMessageName element must match the name that you specify in the WebMessage element in the custom message file. For more information, see Section D.6.3.4, "Creating a Custom Message Template for Proxy Functionality."
   - MaxValues: Specifies the maximum number of target users to be listed in the **User** box in the Act As dialog box. If the number of target users for a proxy user exceeds this value, then an edit box, where the proxy user can enter the ID of a target user, is shown rather than a list of target users. The default is 200.
3. Include the elements and their ancestor elements as appropriate, as shown in the following example:

```
<LogonParam>
     <TemplateMessageName>LogonParamSQLTemplate</TemplateMessageName>
     <MaxValues>100</MaxValues>
</LogonParam>
```

4. Save your changes and close the file.
5. Restart Oracle Business Intelligence.

D.6.3.4 Creating a Custom Message Template for Proxy Functionality

You must create a custom message template for the proxy functionality that contains the SQL statement to perform the following tasks:

- Obtain the list of target users that a proxy user can act as. This list is displayed in the User box in the Act As dialog box.
- Verify whether the proxy user can act as the target user.
- Obtain the list of proxy users that can act as the target user. This list is displayed on the target user's My Account screen.

In the custom message template, you place the SQL statement to retrieve this information in the following XML elements:

Element	Description
getValues	Specifies the SQL statement to return the list of target users and corresponding proxy levels. The SQL statement must return either one or two columns, where the: ■ First column returns the IDs of the target users ■ (Optional) Second column returns the names of the target users
verifyValue	Specifies the SQL statement to verify if the current user can act as the specified target user. The SQL statement must return at least one row if the target user is valid or an empty table if the target user is invalid.
getDelegateUsers	Specifies the SQL statement to obtain the list of proxy users that can act as the current user and their corresponding proxy levels. The SQL statement must return either one or two columns, where the: ■ First column returns the names of the proxy users ■ (Optional) Second column returns the corresponding proxy levels

You can create the custom message template in one of the following files:

- The original custom message file in the directory
- A separate XML file in the directory

To create the custom message template:

1. To create the custom message template in the original custom message file:
   a. Make a backup of the original custom message file in a separate directory.
   b. Make a development copy in a different directory and open it in a text or XML editor.
2. To create the custom message template in a separate XML file, create and open the file in the *ORACLE_INSTANCE*\bifoundation\OracleBIPresentationServicesComponent\coreapplication_obips*n*\analyticsRes\customMessages directory.
3. Start the custom message template by adding the WebMessage element's begin and end tags. For example:

```
<WebMessage name="LogonParamSQLTemplate">
</WebMessage>
```

> **Note:** The name that you specify in the WebMessage element must match the name that you specify in the TemplateMessageName element in the instanceconfig.xml file. For information, see Section D.6.3.3, "Modifying the Configuration File Settings for Proxy Functionality."

4. After the </WebMessage> tag:
   a. Add the <XML> and </XML> tags

b. Between the <XML> and </XML> tags, add the <logonParam name="RUNAS"> and </logonParam> tags.

c. Between the <logonParam name="RUNAS"> and </logonParam> tags, add each of the following tags along with its corresponding SQL statements:

   * <getValues> and </getValues>
   * <verifyValue> and </verifyValue>
   * <getDelegateUsers> and </getDelegateUsers>

The following entry is an example:

```
<XML>
   <logonParam name="RUNAS">
     <getValues> EXECUTE PHYSICAL CONNECTION POOL Proxy.Proxy
        select TARGET from Proxy where PROXYER='@{USERID}'
     </getValues>
     <verifyValue> EXECUTE PHYSICAL CONNECTION POOL Proxy.Proxy
        select TARGET from Proxy where PROXYER ='@{USERID}'
             and TARGET='@{VALUE}'
     </verifyValue>
     <getDelegateUsers>EXECUTE PHYSICAL CONNECTION POOL Proxy.Proxy
        select PROXYER, PROXY_LEVEL from Proxy where TARGET='@{USERID}'
     </getDelegateUsers>
   </logonParam>
</XML>
```

Note that you must modify the example SQL statement according to the schema of the database. In the example, the database and connection pool are both named Proxy, the proxyId is PROXYER, and the targetId is TARGET.

5. If you created the custom message template in the development copy of the original file, then replace the original file in the customMessages directory with the newly edited file.
6. Test the new file.
7. (Optional) If you created the custom message template in the development copy of the original file, then delete the backup and development copies.
8. Load the custom message template by either restarting the server or by clicking the **Reload Files and Metadata** link on the Presentation Services Administration screen. For information on the Administration page, see Section D.2.1, "Understanding the Administration Pages."

D.6.3.5 Assigning the Proxy Privilege

For each user whom you want to enable as a proxy user or for each application role or Catalog group whose members you want to enable as proxy users, you must grant the Act As Proxy privilege. For information on how to assign privileges, see Section D.2.3.2, "Setting Presentation Services Privileges for Application Roles."

D.6.3.6 Assigning the manageRepositories Permission

You must assign the manageRepositories permission to each user you want to allow to act as a proxy user. To assign the permission, create a group called Proxy and associate it with an application role called Proxy, and with an application policy called Proxy (granted the manageRepositories permission), and then add each user (who you want to be a proxy user), to the Proxy group. To achieve this aim, the following must be true:

- A group must exist, or must be created (for example, named Proxy) .

  For more information, see Section 2.3.3, "Creating a Group in the Embedded WebLogic LDAP Server."

- An application role must exist, or must be created (for example, named Proxy), and be mapped to the group called Proxy.

  For more information, see Section 2.4.2.2, "Creating an Application Role."

- An application policy must exist, or must be created (for example, named Proxy), and the Proxy application role must be made a **grantee** of the manageRepositories permission, where:

  – Permission Class

    ```
    oracle.security.jps.ResourcePermission
    ```

  – Resource Name

    ```
    resourceType=oracle.bi.server.permission,resourceName=orac
    le.bi.server.manageRepositories
    ```

  – Permission Actions

    ```
    _all_
    ```

  For more information, see Section 2.4.3, "Creating Application Policies Using Fusion Middleware Control."

- For each user you want to enable as a proxy user, you must add that user to the Proxy group.

  For more information, see Section 2.3.4, "Assigning a User to a Group in the Embedded WebLogic LDAP Server."

Index

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

R

S

T

U

V

W