Oracle® E-Business Suite

Security Guide

Release 12.2

Part No. E22952-09

November 2013

ORACLE

Oracle E-Business Suite Security Guide, Release 12.2

Part No. E22952-09

Copyright © 1994, 2013, Oracle and/or its affiliates. All rights reserved.

Primary Author: Robert Farrington, Mildred Wang

Contributor: George Buzsaki, Anne Carlson, Steve Carter, Steven Chan, Siu Chang, Kenny Tak Chi Ching, Jennifer Collins, Elanchelvan Elango, Surya Narayana Nellepalli, Srikanth Sallaka, Mike Skees, Hope Skinner-Savallisch, Jan Smith, Susan Stratton, Keith M. Swartz, Carol Tilley, Sanjeev Topiwala, Roger Wigenstam

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

This software or hardware and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Contents

Send Us Your Comments

Preface

1 Introduction

Access Control in Oracle E-Business Suite 1-1
Oracle User Management 1-1
Oracle Application Object Library Security 1-2
User and Data Auditing 1-2

2 Access Control with Oracle User Management

Overview 2-1
Function Security 2-2
Data Security 2-3
Role Based Access Control (RBAC) 2-3
Delegated Administration 2-6
Delegating to Proxy Users 2-7
Provisioning Services 2-8
Self-Service and Approvals 2-16

3 Oracle User Management Setup and Administration

Setup Tasks 3-1
Defining Role Categories 3-1
Creating and Updating Roles 3-2
Security Wizard 3-3
Assigning Permissions to Roles 3-3

Searching For Assigned Roles....3-4
Diagnostics for User-Role Assignment....3-7
Creating Instance Sets and Permission Sets....3-7
Defining Delegated Administration Privileges for Roles....3-9
Defining Data Security Policies....3-14
Defining Role Inheritance Hierarchies....3-15
Creating and Updating Registration Processes....3-23
Configuring the User Name Policy....3-26
Delegated Administration Tasks....3-28
Maintaining People and Users....3-28
Creating, Inactivating, and Reactivating User Accounts....3-30
Resetting User Passwords....3-30
Unlocking Locked User Accounts....3-31
Assigning Roles to or Revoking Roles from Users....3-32
Fine Grained Access Control for Role Administration....3-33
Managing System Accounts....3-36
Managing Proxy Users....3-38
Registering External Organization Contacts....3-40
Self Service Features....3-40
Self-Service Registration....3-40
Requesting Additional Application Access....3-41
Login Assistance....3-41
Security Reports....3-42
Home Page....3-42
Listing Functions for a User....3-44
Listing Data Security and Business Objects for a User....3-44
Listing Roles and Responsibilities for a User....3-46
Listing Users With a Given Role....3-47
Listing Functions That Can Be Accessed From a Given Role....3-49
Listing Objects for a Given Role....3-49
Listing Users for a Given Function....3-50
Listing Roles and Responsibilities for a Given Object....3-51

4 Oracle Application Object Library Security

Overview of Oracle E-Business Suite Security....4-1
HRMS Security....4-2
Oracle E-Business Suite User Passwords....4-2
Guest User Account....4-3
User Session Limits....4-4
Defining a Responsibility....4-4

Additional Notes About Responsibilities....4-5
Defining Request Security....4-6
Oracle Applications Manager Security Tests....4-8
Overview of Security Groups in Oracle HRMS....4-8
Defining Security Groups....4-9
Overview of Function Security....4-9
Terms....4-9
Executable functions vs. Non-executable functions....4-11
Functions, Menus, and the Navigate Window....4-11
Menu Entries with a Submenu and Functions....4-12
How Function Security Works....4-12
Implementing Function Security....4-13
Defining a New Menu Structure....4-13
Notes About Defining Menus....4-14
Menu Compilation....4-14
Preserving Custom Menus Across Upgrades....4-15
Overview of Data Security....4-15
Concepts and Definitions....4-15
Implementation of Data Security....4-18
Responsibilities Window....4-19
Security Groups Window....4-22
Users Window....4-23
Form Functions Window....4-28
Menus Window....4-33
Menu Viewer....4-36
Objects....4-37
Find Objects....4-38
Update Object....4-39
Create Object....4-39
Object Detail....4-40
Delete Object....4-41
Object Instance Sets....4-41
Manage Object Instance Set....4-41
Create Object Instance Set....4-42
Update Object Instance Set....4-42
Delete Object Instance Set....4-43
Object Instance Set Details....4-43
Grants....4-44
Search Grants....4-44
Create Grant....4-44
Define Grant....4-45

Select Object Data Context 4-45
Define Object Parameters and Select Set 4-46
Review and Finish 4-46
Update Grant 4-46
View Grant 4-46
Functions 4-46
Search 4-47
Create Function 4-48
Update Function 4-48
Duplicate Function 4-49
View Function 4-49
Delete Function 4-49
Navigation Menus 4-50
Search for Menus 4-51
Create Navigation Menu 4-51
Update Menu 4-52
Duplicate Menu 4-53
View Menu 4-53
Delete Menu 4-53
Permissions 4-53
Create Permission 4-54
Update Permission 4-54
Duplicate Permission 4-54
View Permission 4-55
Delete Permission 4-55
Permission Sets 4-55
Create Permission Set 4-55
Update Permission Set 4-56
Duplicate Permission Set 4-56
View Permission Set 4-57
Delete Permission Set 4-57
Compile Security Concurrent Program 4-57
Parameter 4-57
Function Security Reports 4-57
Users of a Responsibility Report 4-58
Report Parameters 4-58
Report Heading 4-58
Column Headings 4-59
Active Responsibilities Report 4-59
Report Parameters 4-59
Report Heading 4-59

Column Headings 4-60
Active Users Report 4-60
Report Parameters 4-60
Report Heading 4-60
Column Headings 4-61
Reports and Sets by Responsibility Report 4-61
Report Parameters 4-61
Report Headings 4-62
Oracle Application Object Library REST Security Services 4-62
Cookie Domain Scoping 4-74
Allowed JSPs 4-77

5 Auditing and Monitoring

Overview of Auditing and Monitoring 5-1
Auditing User Activity 5-1
Auditing Database Row Changes 5-1
Auditing User Activity 5-2
Major Features 5-2
Setting Up Sign-On Audit 5-3
Using the Application Monitor 5-5
Sign-On Audit Reports 5-5
Reporting On AuditTrail Data 5-6
AuditTrail 5-6
Audit Trail Update Tables Report 5-6
Changing Your Audit Tables 5-6
Setting Up AuditTrail 5-7
AuditTrail Tables, Triggers and Views 5-8
Reporting on Audit Information 5-14
Disabling AuditTrail and Archiving Audit Data 5-15
Additional Audit Trail Reporting 5-16
Audit Industry Template 5-16
Audit Hierarchy Editor 5-17
Audit Query Navigator 5-19
Audit Report 5-20
Monitor Users Window 5-21
Audit Installations Window 5-23
Audit Groups Window 5-25
Audit Tables Window 5-27
Signon Audit Concurrent Requests Report 5-30
Report Parameters 5-30

Report Heading 5-31
Column Headings 5-31
Signon Audit Forms Report 5-32
Report Parameters 5-32
Report Heading 5-33
Column Headings 5-33
Signon Audit Responsibilities Report 5-34
Report Parameters 5-34
Report Heading 5-35
Column Headings 5-35
Signon Audit Unsuccessful Logins Report 5-36
Report Parameters 5-36
Report Heading 5-37
Column Headings 5-37
Signon Audit Users Report 5-38
Report Parameters 5-38
Report Heading 5-39
Column Headings 5-39
Purge Signon Audit Data Program 5-40
Parameters 5-40
Database Connection Tagging 5-40
Usage 5-40
Management 5-41

6 Single Sign-On Integration

Overview of Single Sign-On Integration 6-1
Introduction to Enterprise User Management 6-3
Integration Actions and Options 6-4
Enterprise User Management 6-9
Deployment Scenario 0: E-Business Suite + SSO and OID 6-12
User Management Options 6-14
End-User Experience 6-16
Session Timeout Behavior 6-17
User Management Options 6-18
Critical Implementation Decisions 6-24
Implementation Instructions 6-25
Deployment Scenario 1: Multiple Oracle E-Business Suite Instances + Central SSO and OID Instance 6-27
Deployment Scenario 2: New Oracle E-Business Suite Installation + Existing Third-Party Identity Management Solution 6-29
End-User Experience 6-30

User Management 6-31
Critical Implementation Decisions 6-33
Implementation Instructions 6-34
Deployment Scenario 3: Existing Oracle E-Business Suite Instance + Existing Third-Party Identity Management Solutions 6-35
Critical Implementation Decisions 6-40
Implementation Instructions 6-41
Deployment Scenario 4: Multiple Oracle E-Business Suite Instances with Unique User Populations 6-42
Advanced Features 6-43
Single Sign-On Profile Options 6-48
Configuring Directory Integration Platform Provisioning Templates 6-58
Administering the Provisioning Process 6-63
Changing E-Business Suite Database Account Password 6-67
Manual Subscription Management With Provsubtool 6-68
Migrating Data between Oracle E-Business Suite and Oracle Internet Directory 6-71
Enabling and Disabling Users 6-80
Synchronizing Oracle HRMS with Oracle Internet Directory 6-81
Supported Attributes 6-83
References and Resources 6-85
Glossary of Terms 6-85

Index

Send Us Your Comments

Oracle E-Business Suite Security Guide, Release 12.2

Part No. E22952-09

Oracle welcomes customers' comments and suggestions on the quality and usefulness of this document. Your feedback is important, and helps us to best meet your needs as a user of our products. For example:

- Are the implementation steps correct and complete?
- Did you understand the context of the procedures?
- Did you find any errors in the information?
- Does the structure of the information help you with your tasks?
- Do you need different information or graphics? If so, where, and in what format?
- Are the examples correct? Do you need more examples?

If you find any errors or have any other suggestions for improvement, then please tell us your name, the name of the company who has licensed our products, the title and part number of the documentation and the chapter, section, and page number (if available).

Note: Before sending us your comments, you might like to check that you have the latest version of the document and if any concerns are already addressed. To do this, access the new Oracle E-Business Suite Release Online Documentation CD available on My Oracle Support and www.oracle.com. It contains the most current Documentation Library plus all documents revised or released recently.

Send your comments to us using the electronic mail address: appsdoc_us@oracle.com

Please give your name, address, electronic mail address, and telephone number (optional).

If you need assistance with Oracle software, then please contact your support representative or Oracle Support Services.

If you require training or instruction in using Oracle software, then please contact your Oracle local office and inquire about our Oracle University offerings. A list of Oracle offices is available on our Web site at www.oracle.com.

Preface

Intended Audience

Welcome to Release 12.2 of the *Oracle E-Business Suite Security Guide.*

This guide assumes you have a working knowledge of the following:

- The principles and customary practices of your business area.
- Computer desktop application usage and terminology.

If you have never used Oracle E-Business Suite, we suggest you attend one or more of the Oracle E-Business Suite training classes available through Oracle University.

> **Note:** This book typically uses UNIX nomenclature in specifying files and directories. Windows users should substitute the appropriate Windows terms where applicable. For example, a UNIX .env (environment) file will be a .cmd (command) file on Windows.

See Related Information Sources on page xiv for more Oracle E-Business Suite product information.

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.

Structure

1 Introduction
2 Access Control with Oracle User Management
3 Oracle User Management Setup and Administration
4 Oracle Application Object Library Security
5 Auditing and Monitoring
6 Single Sign-On Integration

Related Information Sources

This book is included in the Oracle E-Business Suite Documentation Library, which is supplied in the Release 12.2 Media Pack. If this guide refers you to other Oracle E-Business Suite documentation, use only the latest Release 12.2 versions of those guides.

Online Documentation

All Oracle E-Business Suite documentation is available online (HTML or PDF).

- **Online Help** - Online help patches (HTML) are available on My Oracle Support.
- **PDF Documentation** - See the Oracle E-Business Suite Documentation Library for current PDF documentation for your product with each release.
- **Release Notes** - For information about changes in this release, including new features, known issues, and other details, see the release notes for the relevant product, available on My Oracle Support.
- **Oracle Electronic Technical Reference Manual -** The Oracle Electronic Technical Reference Manual (eTRM) contains database diagrams and a detailed description of database tables, forms, reports, and programs for each Oracle E-Business Suite product. This information helps you convert data from your existing applications and integrate Oracle E-Business Suite data with non-Oracle applications, and write custom reports for Oracle E-Business Suite products. The Oracle eTRM is available on My Oracle Support.

Related Guides

You should have the following related books on hand. Depending on the requirements of your particular installation, you may also need additional manuals or guides.

Oracle E-Business Suite Concepts

This book is intended for all those planning to deploy Oracle E-Business Suite Release 12.2, or contemplating significant changes to a configuration. After describing the Oracle E-Business Suite architecture and technology stack, it focuses on strategic topics, giving a broad outline of the actions needed to achieve a particular goal, plus the installation and configuration choices that may be available.

Oracle E-Business Suite Installation Guide: Using Rapid Insatll

This book is intended for use by anyone who is responsible for installing or upgrading Oracle E-Business Suite. It provides instructions for running Rapid Install either to carry out a fresh installation of Oracle E-Business Suite Release 12.2, or as part of an upgrade to Release 12.2.

Oracle E-Business Suite Maintenance Guide

This guide explains how to patch an Oracle E-Business Suite system, describing the adop patching utility and providing guidelines and tips for performing typical patching operations. It also describes maintenance strategies and tools that can help keep a system running smoothly.

Oracle E-Business Suite Setup Guide

This guide contains information on system configuration tasks that are carried out either after installation or whenever there is a significant change to the system. The activities described include defining concurrent programs and managers, enabling Oracle Applications Manager features, and setting up printers and online help.

Integration Repository

The Oracle Integration Repository is a compilation of information about the service endpoints exposed by the Oracle E-Business Suite of applications. It provides a complete catalog of Oracle E-Business Suite's business service interfaces. The tool lets users easily discover and deploy the appropriate business service interface for integration with any system, application, or business partner.

The Oracle Integration Repository is shipped as part of the E-Business Suite. As your instance is patched, the repository is automatically updated with content appropriate for the precise revisions of interfaces in your environment.

You can navigate to the Oracle Integration Repository through Oracle E-Business Suite Integrated SOA Gateway.

Do Not Use Database Tools to Modify Oracle E-Business Suite Data

Oracle STRONGLY RECOMMENDS that you never use SQL*Plus, Oracle Data Browser, database triggers, or any other tool to modify Oracle E-Business Suite data unless otherwise instructed.

Oracle provides powerful tools you can use to create, store, change, retrieve, and maintain information in an Oracle database. But if you use Oracle tools such as SQL*Plus to modify Oracle E-Business Suite data, you risk destroying the integrity of your data and you lose the ability to audit changes to your data.

Because Oracle E-Business Suite tables are interrelated, any change you make using an Oracle E-Business Suite form can update many tables at once. But when you modify Oracle E-Business Suite data using anything other than Oracle E-Business Suite, you may change a row in one table without making corresponding changes in related tables.

If your tables get out of synchronization with each other, you risk retrieving erroneous information and you risk unpredictable results throughout Oracle E-Business Suite.

When you use Oracle E-Business Suite to modify your data, Oracle E-Business Suite automatically checks that your changes are valid. Oracle E-Business Suite also keeps track of who changes information. If you enter information into database tables using database tools, you may store invalid information. You also lose the ability to track who has changed your information because SQL*Plus and other database tools do not keep a record of changes.

1

Introduction

Access Control in Oracle E-Business Suite

This release of Oracle E-Business Suite includes a sophisticated security system. Core Security includes a Role Based Access Control model that builds on the Function Security and Data Security models. A set of administrative features that build on Core Security are also provided.

> **Important:** As well as following the guidance given in this book, system administrators should refer to My Oracle Support Knowledge Document 403537.1, *Secure Configuration Guide for Oracle E-Business Suite Release 12*, for important additional configuration guidelines about securing Oracle E-Business Suite.

Oracle User Management

Oracle User Management is a secure and scalable system that enables organizations to define administrative functions and manage users based on specific requirements such as job role or geographic location. With Oracle User Management, instead of exclusively relying on a centralized administrator to manage all its users, an organization can create local administrators and grant them sufficient privileges to manage a specific subset of the organization's users. This provides the organization with a more granular level of security, and the ability to make the most effective use of its administrative capabilities.

Oracle's function and data security models constitute the base layers of this system, and contain the traditional system administrative capabilities. Organizations can optionally add more layers to the system, depending on the degree of flexibility they require.

Key features of Oracle User Management include:

- **Role Based Access Control (RBAC)** - Enables organizations to create roles based on specific job functions, and to assign these roles the appropriate permissions. With

RBAC, administrative privileges and user access are determined by assigning individuals the appropriate roles.

- **Delegated Administration** - Enables system administrators to delegate some of their administrative privileges to individuals that manage a subset of the organization's users. These individuals are assigned administrative privileges for a limited set of roles that they can assign to the users they manage.

- **Registration Processes** - Enable organizations to provide end-users with a method for requesting various levels of access to the system, based on their eligibility. Registration processes also simplify an administrator's job by providing streamlined flows for account maintenance and role assignment.

- **Self Service Requests and Approvals** - Enable end users to request initial access or additional access to the system.

Oracle User Management is used in both an administrative and a functional capacity. System administrators use Oracle User Management to define the available levels of access control as required, including RBAC, Delegated Administration, Registration Processes, and Self Service & Approvals. Part of this setup includes defining local administrators primarily by creating administrative roles and assigning them to individuals who serve as an organization's local administrators. Once this is accomplished, local administrators use Oracle User Management to manage a subset of an organization's users.

Oracle Application Object Library Security

Oracle Application Object Library security comprises two main components, Function Security and Data Security.

Function Security restricts user access to individual menus of functions, such as forms, HTML pages, or widgets within an application. Function Security by itself restricts access to various functions, but it does not restrict access to the data a user can see or what actions a user can perform on that data.

Data Security restricts the access to the individual data that is shown once a user has selected a menu or menu option. For example, with Data Security you can control the set of users that a particular local security administrator can access within Oracle User Management. In conjunction with Function Security, Data Security provides additional access control on data that a user can see or actions a user can perform on that data.

User and Data Auditing

Oracle E-Business Suite allows you to audit users and changes they make to application data.

The Sign-On Audit feature allows you to track your users' activities. You can choose

who to audit and what type of user information to track. Sign-On Audit reports give you historical, detailed information on your users' activities within an application. Also, the Monitor Users form allows you to view online, real-time information on user activity.

AuditTrail lets you keep a history of changes to important data: what changed, who changed it, and when. With AuditTrail, you can easily determine how any data row or element obtained its current value. You can track information on most types of fields, including character, number, and date fields.

2

Access Control with Oracle User Management

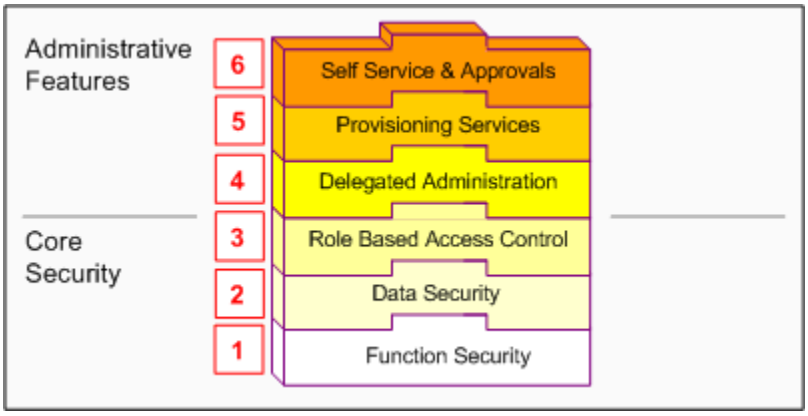
Overview

This chapter introduces the Core Security and Administrative Features of Oracle User Management. Core Security includes Oracle's Function and Data Security models, as well as Role Based Access Control. Administrative Features build upon Core Security and include Delegated Administration, Registration Processes, and Self Service and Approvals.

Core Security and Administrative Features are implemented in successive layers and each builds upon the one that precedes it. Organizations can optionally uptake the various layers, depending on the degree of automation and scalability that they wish to build upon the existing Function and Data Security models.

In general, Access Control with Oracle User Management begins with basic system administration tasks, progresses to more distributed, local modes of administration, and ultimately enables users to perform some basic, predefined registration tasks on their own. The following diagram illustrates how the layers build upon each other.

Oracle User Management Layers



Oracle User Management provides support for legacy and application-specific security mechanisms through workflow business events. Oracle User Management raises these events once a user's request is approved. Organizations can then intercept these events, determine the appropriate action, and assign any additional privileges that may be required.

Function Security

Function Security Layer

Function Security is the base layer of access control in Oracle E-Business Suite. It restricts user access to individual menus and menu options within the system, but does not restrict access to the data contained within those menus. For example, an organization could use Function Security to provide its sales representatives with the required menus and menu options for querying customers. It could also control access to specific components of those pages such as a button on a sales forecasting page. For a more comprehensive explanation of function security, see the Oracle Application Object Library Security chapter, page 4-1.

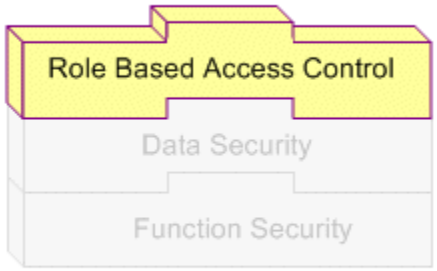
Data Security

Data Security Layer

Data Security is the next layer of access control. Building on Function Security, Data Security provides access control within Oracle E-Business Suite on the data a user can access, and the actions a user can perform on that data. Oracle E-Business Suite restricts access to individual data that is displayed on the screen once the user has selected a menu or menu option. For example, Data Security restricts the set of users that a local administrator can access within Oracle User Management. Data Security policies can only be defined for applications that have been written to utilize the Data Security Framework. For a more comprehensive explanation of data security, see the Oracle Application Object Library Security chapter, page 4-1.

Role Based Access Control (RBAC)

Role Based Access Control Layer



RBAC is the next layer and builds upon Data Security and Function Security. With RBAC, access control is defined through roles, and user access to Oracle E-Business Suite is determined by the roles granted to the user. Access control in Oracle E-Business Suite closely follows the RBAC ANSI standard (ANSI INCITS 359-2004) originally proposed by the US National Institute of Standards & Technology (NIST), which defines a role as "a job function within the context of an organization with some

associated semantics regarding the authority and responsibility conferred on the user assigned to the role."

A role can be configured to consolidate the responsibilities, permissions, function security and data security polices that users require to perform a specific function. This is accomplished with a one-time setup, in which permissions, responsibilities, and other roles are assigned to the role. Users are not required to be assigned the lower-level permissions directly, since permissions are implicitly inherited on the basis of the roles assigned to the user. This simplifies mass updates of user permissions, since an organization need only change the permissions or role inheritance hierarchy defined for a given role, and the users assigned that role will inherit the new set of permissions automatically.

Organizations can define roles that closely mirror their business situation. For example, an organization can create an "Employee" role and then assign that role to all of its employees. It can also create an "External" role and assign that role to customers and suppliers. Further examples may include specific roles such as "Support Agent", "Sales Rep", "Sales Managers". In these examples, each role contains a specific level of access privileges that restricts its assignees to the scope of their job functions. Some members of the organization will probably be assigned more than one role. A sales representative would be assigned the Employee and Sales Representative roles, and a Sales Manager would be assigned the Employee, Sales Representative, and Sales Manager roles. Roles and role assignments are stored in the workflow directory, which is interpreted by the security system at runtime.

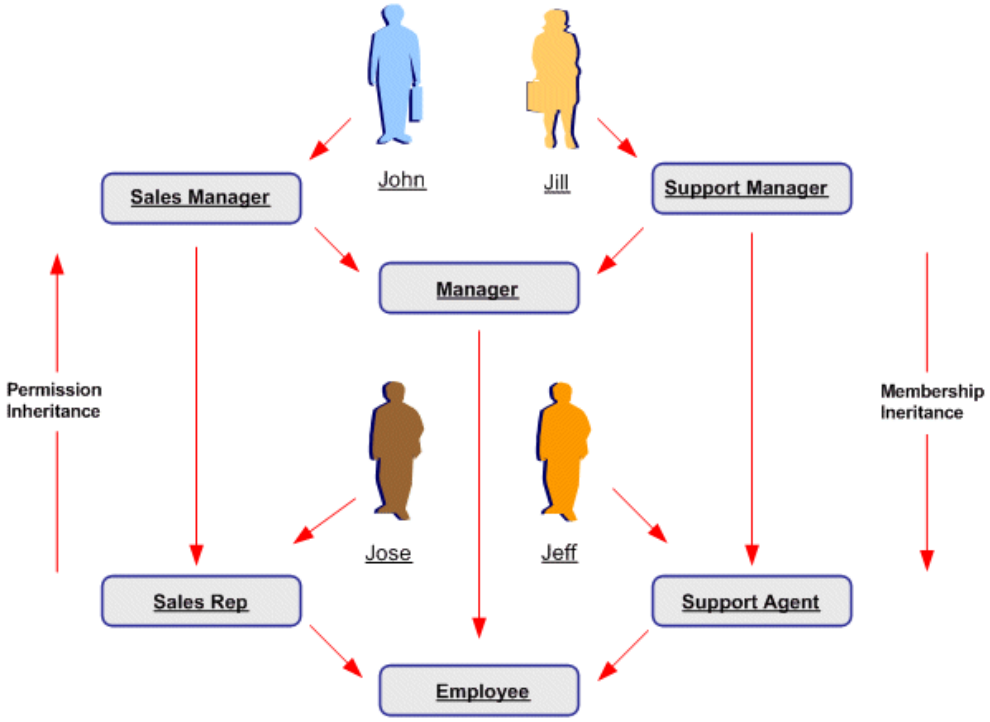
Role Categories

As part of the Oracle E-Business Suite RBAC model, Oracle User Management introduces Role Categories. Administrators can create role categories to bundle roles and responsibilities to make the process of searching for roles and responsibilities easier. For example, all sales and marketing related roles could be included in the Sales & Marketing category.

Role Inheritance Hierarchies

Roles can be included in role inheritance hierarchies that can contain multiple subordinate roles and superior roles. With role inheritance hierarchies, a superior role inherits all of the properties of its subordinate role, as well as any of that role's own subordinate roles. The following example demonstrates how role inheritance hierarchies can greatly simplify user access control and administration.

Role Inheritance Hierarchy



In the above figure, the arrows on each side of the diagram indicate membership inheritance and permission inheritance. Text in the rounded boxes indicates roles. An arrow pointing from an individual to a role indicates that this individual is assigned the role. An arrow pointing from one role to another indicates that the role from which the arrow points is the superior role, and the role to which it points is the subordinate role. Permissions associated with a role are inherited by all of its superior roles and the individuals to which any of these roles are assigned.

In this example, some roles such as "Employee" or "Manager" are assigned general permissions for a given function. For example, the Employee role may provide access to menus generally available to all employees, while the Manager role provides access to menus that should only be viewed by managers. Because the Employee role is a subordinate role of the Manager role, anyone assigned the Manager role automatically obtains the permissions associated with the Employee role. Other roles in this example pertain to more specific job functions, such as Sales Manager and Sales Representative, or Support Manager and Support Agent. These roles may provide access to job-specific menus and data such as the Sales Forecasting menu, or the Support application.

Delegated Administration

Delegated Administration Layer

Delegated Administration is a privilege model that builds on the RBAC system to provide organizations with the ability to assign the required access rights for managing roles and user accounts. With delegated administration, instead of relying on a central administrator to manage all its users, an organization can create local administrators and grant them sufficient privileges to manage a specific subset of the organization's users and roles. This provides organizations with a tighter, more granular level of security, and the ability to easily scale their administrative capabilities. For example, organizations could internally designate administrators at division or even department levels, and then delegate administration of external users to people within those (external) organizations. Delegation policies are defined as data security policies. The set of data policies that are defined as part of delegated administration are known as Administration Privileges.

A delegated administrator can be given the capability to perform one or more of the following role management actions: Create Role, Manage Role, Manage Role Hierarchy, Run Security Wizard, Assign Role, and Revoke Role. Older releases required delegated administrators to be given either all role management privileges, or none. Now the administration operations have been separated, so.the super administrator can specify which operations can be performed by which delegated administrator on which set of roles.

Administration Privileges

Administration Privileges determine the users, roles and organization information that delegated administrators (local administrators) can manage. Each privilege is granted separately, yet the three work in conjunction to provide the complete set of abilities for the delegated administrator.

- **User Administration Privileges** A local administrator must be granted User

Administration Privileges to determine the users and people the local administrator can manage. Local administrators can be granted different privileges for different subsets of users. For example, a local administrator can be granted privileges only to query one set of users, and granted full privileges (including update and reset password) for another set. Local administrators cannot query users for which they do not have administration privileges.

- **Role Administration Privileges** Role Administration Privileges define the roles that local administrators can directly assign to and revoke from the set of users they manage.

- **Organization Administration Privileges** Organization Administration Privileges define the external organizations a local administrator can view in Oracle User Management. This privilege enables an administrator to search for people based on their organization, if the local administrator has additionally been granted access to view the people in that organization (User Administration Privileges). Depending on the user administration privileges, an administrator may have the ability to register new people for that organization.

Oracle E-Business Suite continues to support the traditional "System Administrator" level of administration privileges, where a designated group of people manages all users and access privileges. Oracle User Management ships a predefined Security Administrator role, which gives the administrator the privileges to manage all users including system accounts and all roles in the system.

Delegated administration setup for User Administration requires the creation of instance sets and permission sets. Instance sets can be created from the main UMX screen. All possible combinations of seeded UMX permissions are seeded as permission sets and made available from this screen. A data security object, UMX_SYS_ACCT, represents system accounts. Administrators can create instance sets against this object to specify system accounts that can be managed.

Delegating to Proxy Users

There are a number of business scenarios in which users of Oracle E-Business Suite need to grant delegates the ability to act on their behalf (act as *proxy users* for them) when performing specific E-Business Suite functions. Traditionally, delegators have done this by giving passwords for specific applications to other users. A delegate who was given another user's passwords for certain applications could assume the identity and privileges of the delegator within those applications, and only those applications.

The integration of Oracle E-Business Suite with Oracle Single Sign-On (SSO) makes this traditional strategy insecure. If a delegator grants a delegate access to his SSO password, the delegate will be able to access every SSO-enabled application to which the delegator has access, not just to specific applications. The new mechanism was designed to enable limited, auditable delegation of privilege from delegators to their delegates.

Important: Employing the Proxy User mechanism gives all-or-nothing delegation capability. However, start and end dates can be defined to limit the duration of proxy access.

Examples of Delegation

There are a number of common scenarios where a user may need to allow another user or users to interact with Oracle E-Business Suite on their behalf:

- Executives allowing their assistants to access selected business applications on their behalf
- In a similar way to executives and their assistants, but for a more limited duration, managers may need to grant peers or subordinates limited authority to act on their behalf while they are out of the office
- Users may need to grant help-desk staff limited duration access to their E-Business Suite accounts, so that help desk staff can investigate problems and provide assistance
- The Proxy User mechanism allows such users to obtain limited, auditable access to accounts such as SYSADMIN that might otherwise have to be shared and therefore harder to audit
- Companies may be subject to audits that require granting a specific user (the auditor) access to employees' E-Business Suite accounts, normally on a read-only basis.

The ability for users to access the proxy feature is controlled by a *Security Administrator* role. Users with this role determine which set of users can create delegates who can act on their behalf.

Provisioning Services

Provisioning Services Layer

Provisioning services are modeled as *registration processes* that enable end users to perform some of their own registration tasks, such as requesting new accounts or additional access to the system. They also provide administrators with a faster and more efficient method of creating new user accounts, as well as assigning roles. Registration processes accomplish this by encapsulating core components of registration, including:

- The role(s) assigned after the user successfully completes the process.
- An optional registration user interface for collecting account or additional information.
- A workflow for approval, confirmation, rejection, and identity verification notifications.
- The Approval Management Transaction Type. A transaction type represents a set of approval routing rules that are interpreted at runtime.
- The set of users that are eligible to sign up for additional access (only applicable for Request for Additional Access registration processes).
- Whether identity verification is required. Identity verification confirms the identity of a requester before the registration request is processed, by sending an email notification to the requester's email address. If the recipient does not reply within a specified time, the request will be automatically rejected.
- The set of local administrators that should be able to register people and/or create users through the Account Creation by Administrators registration process.

When a user completes registration using a registration process, the system captures the required information from the user, and subsequently assigns that person a new user account, role, or both. Oracle User Management supports three types of registration processes: Self-service Account Requests, Requests for Additional Access, and Account Creation by Administrators.

Self-Service Account Requests

Commonly referred to as Self-Service Registration, self-service account requests provide a method for individuals to request a new user account. Consider a case where customers may need to register before they can purchase an item from an online store. Once the registration process has been completed, the customer obtains both a user account and the necessary role(s) for accessing some portion of the web site in which they registered.

This release of Oracle User Management provides sample Self-Service registration UIs for internal employees, and for new, external individuals. Organizations can copy these sample Self-Service registration and extend them based on their own requirements. In addition, organizations that wish to support other types of users, or capture additional information specific to their applications, are able to extend or create their own

registration UIs and business logic.

Oracle User Management provides support for displaying different registration links on the login page based on the application tier login page that provides access. The registration link can contain additional parameters that are not known at design time, such as the country code. These additional parameters can be used later during the registration process. Using country code as an example, a registration process could route the approval requests to the most appropriate approver. Therefore, all those who request an account from Norway could be routed to a Norwegian account approver.

> **Note:** "Accounts" and "User Accounts" refer to login accounts, stored in the FND_USER table.

Requests for Additional Access

Users can request additional access through the Oracle User Management Access Request Tool (ART), available in the Global Preferences menu. Requests for Additional Access uses the same Oracle User Management infrastructure and processing logic as Self Service Account Requests.

Additional Access and Self Service Eligibility

Eligibility defines the Roles for which a user can sign up using the Access Request Tool. It determines the groups of users defined in the workflow directory that are entitled to register for a given role. A registration process of type "Additional Access" can be made available to predefined sets of users across all roles or groups. Eligibility is defined as a data security policy, and interrogated at runtime by the Access Request Tool.

Because roles are stored in the workflow directory, they can be used both to grant access to applications and to define eligibility. This enables organizations to define an incremental registration process in which new users can sign up for roles if they are first approved for the ones that precede them. For example, once a new user is approved for the A Role, the user can then sign up for the B Role. If, however, the user is not first approved for the A Role, then the user cannot sign up for the B Role.

Oracle User Management can define eligibility policies for any groups and roles stored in the workflow directory.

Delegated Administration and Registration Processes

When an administrator assigns a role to a user, the administrator essentially fulfills a registration request on behalf of the user. When the administrator assigns a role to the user, Oracle User Management invokes the corresponding "Additional Access (Administrator)" registration process (if defined) and interprets the registration processes metadata. If a registration UI is defined, Oracle User Management launches it and the administrator completes the registration process. Notification workflows are only invoked when a registration process is defined for the role that is being assigned to the user.

Directly assigning a role to a user bypasses any pre-defined approval routing rules, as defined in Oracle Approval Management. Administrators can view all roles that are

assigned to a user, but cannot assign or revoke roles for which they do not have administrative privileges. An administrator assigning a role to a user is essentially fulfilling a registration request on behalf of the user.

Account Creation By Administrators

Administrators benefit from registration processes having been designed to streamline the process of creating and maintaining user access. Registration processes of this type are geared toward administrators, especially delegated administrators, to ensure consistent application of the organization's user security policies. Each account creation registration process can be made available to selected administrators.

Registration Process Infrastructure

This section describes components of the common infrastructure that handles all registration requests submitted through Oracle User Management.

User Name Policies

Oracle User Management enables organizations to define their own user name policies for new users. These can include such formats as email address, "firstname.lastname" (or an abbreviated version), employee number, social security number, or some other meaningful information. When the account request is submitted, Oracle User Management reserves the specified user name for the duration of the approval process.

Oracle User Management ships with a default user name policy that identifies users by their *email address*. This is implemented as a configurable infrastructure that organizations can easily customize to suit their specific needs.

Email Verification

Oracle User Management provides a mechanism for verifying the identity of the requester before the registration request is processed. Identity verification is based on the email address provided by the requester. Oracle User Management sends the requester an email notification when the requester has completes the registration flow. If the user does not reply to the email notification within a specified time, the request is automatically rejected. Email verification is only applicable to Self-Service account requests, and is enabled or disabled for each registration process.

> **Note:** Oracle recommends that when building self-service registration UIs with identity verification enabled, an organization should indicate in the UIs and confirmation messages that a response is required to process the user's request.

Temporary Storage of Registration Data

Oracle User Management provides a mechanism to store registration data in a pending state until a request is approved. This data is available to the workflow notifications used for sending approvals, to Approval Management routing rules, and to the business logic that writes the information in the final destination tables. Oracle User Management accomplishes this by using event objects that are part of the Workflow Business Events infrastructure.

Registration Engine

The Oracle User Management registration engine uses a workflow to define the business logic that drives the registration process once a request has been submitted. The name of the workflow is UMX Registration Workflow (UMXREGWF).

This process:

- Raises business events
- Provides temporary storage of registration data
- Provides identity verification
- Includes the integration point with Oracle Approval Management
- Activates user accounts
- Reserves and releases user names
- Assigns roles
- Maintains registration status in the Oracle User Management schema
- Launches notification workflows

Organizations can customize the components of the registration process (such as notifications, approval routing rules, and user name policies) without having to review and understand all Oracle User Management code.

Routing Approval Requests

Approvers can be configured based on rules that are specific to each type of request. Organizations can define these rules according to their requirements, and can specify types of requests that do not require approval. Oracle User Management is integrated with Oracle Approval Management, an application that provides a flexible and powerful rules engine that can be configured through declarative means to route approval requests. Oracle User Management also provides APIs that enable approval rules to be based on any information captured during the registration process, including any parameters passed from the "Register Here" link on the Login page, which may not have been known at design time.

Workflow Business Events

Oracle User Management raises the following Workflow business events:

Oracle User Management Workflow Business Events

Event	Description
oracle.apps.fnd.umx.rolerequested	An event that is raised when a role is requested.
oracle.apps.fnd.umx.accountrequested	An event that is raised when an account is requested.
oracle.apps.fnd.umx.requestapproved	An event that is raised when an account or role is approved.
oracle.apps.fnd.umx.requestrejected	An event that is raised when an account or role is rejected.
<custom event>	A custom business event is raised for the owner of the registration process to write the registration. The custom event is raised multiple times. For more information, see the *UMX Developer's Guide*, Knowledge Document 399400.1 on My Oracle Support.

> **Note:** Oracle recommends using the UMX events mentioned above only for centralized requirements such as auditing. For any registration-specific processing, use the custom event defined for the registration process.

Depending on the context, the event parameters listed in the following table are set automatically by the Oracle User Management registration engine when business events are raised. Any additional information captured in the registration UI, approval notifications, or programmatically through business logic is also available as event parameters.

Oracle User Management Workflow Business Event Parameters

Name	Description
REG_SERVICE_CODE	Represents the primary key of the registration process

Name	Description
REG_SERVICE_TYPE	The type of registration process
REQUESTED_BY_USER_ID	Identifies the user submitting the request
REQUESTED_FOR_USER_ID	Identifies the user for whom the request is submitted
REQUESTED_USERNAME	The requested user name
WF_ROLE_NAME*	Represents the primary key value of the requested role or the default role for any account requests
AME_TRANSACTION_TYPE_ID	Represents part of the primary key for the transaction type in Oracle Approval Management
AME_APPLICATION_ID	Represents part of the primary key for the transaction type in Oracle Approval Management

* WF_ROLE_NAME is not required for Self Service Account Creation or Account Creation for Administrators registration processes. In such cases, a null value is passed. Any additional information captured in the registration UI, from approvers, in approval notifications, or set by business logic is also available as parameters when an Oracle User Management business event is raised.

Sample Program

```
/************************************************************
This is a sample subscription to any of the above events.

Function custom_logic (p_subscription_guid in raw,
 p_event in out NOCOPY WF_EVENT_T)
Return varchar2 is
 l_first_name varchar2(30);
Begin
 l_first_name :=  p_event.getvalueforparameter ('FIRST_NAME');
// Manipulate the data
End custom_logic;
************************************************************/
```

Registration Status

Users can check registration status of requests through the Access Request Tool (ART) and administrators can do so using the Administration screens. For any pending

requests, the Show Info icon shows the current approver and confirmation number. The confirmation number represents the number (ITEM_KEY) of the Oracle User Management Registration Workflow (UMXREGWF) workflow process handling the request.

Notification Workflows

Notification workflows enable an organization to define its own email notifications that are specific to each Role or Registration Process. Notifications include:

Oracle User Management Notification Types

Notification	Recipient
Approver notifications	Each approver.
Approval confirmation notifications	Individual for whom the request was filed.
Rejection notifications	Individual for whom the request was filed.
Identity verification notifications	Individual for whom the request was filed.

For each request that requires approval as determined by the Oracle Approval Management Engine, Oracle User Management invokes the notification workflow to request approval. Notification workflows can be written to allow approvers to review the information submitted in the registration process, make changes, and provide additional information if required.

Any changes or additional information provided can be passed back to the Oracle User Management registration engine for further processing. For example, if Oracle User Management is used to provide self service registration capability for iSP (Internet Supplier Portal), then approvers can provide additional information about site and contact restrictions for the requester. Information entered by previous approvers, including comments, are available to subsequent approvers.

Oracle User Management provides the following sample notification workflows that organizations can use directly or can copy and modify based on their requirements:

Sample Notification Workflows

Name	Item Type	Description
Oracle User Management Additional Access Request notification workflow	UMXNTWF1	Sends notifications pertaining to all requests for additional access.

Name	Item Type	Description
Oracle User Management Notification Workflow (Account Request)	UMXNTWF2	Sends notifications pertaining to all account requests.

Self-Service and Approvals

Self-Service & Approvals Layer

Once registration processes have been configured as required, individuals can subsequently perform self-service registration tasks, such as obtaining new user accounts or requesting additional access to the system. In addition, organizations can use the Oracle Approvals Management engine to create customized approval routing for these requests. For example, an organization may enable users to request a particularly sensitive role: however, before the user is granted the role, the organization can require that two senior members of staff, such as a manager and a vice president, must approve the request.

Oracle User Management also provides self-service features for resetting forgotten passwords, and ships with the following sample self-service registration processes:

- Employee Self-Service Registration
- Customer Self-Service Registration (external individuals)

Organizations can either use these registration processes in their existing form, or as references for developing their own registration processes.

3

Oracle User Management Setup and Administration

Setup Tasks

This section discusses the setup tasks for Oracle User Management. The implementor or system administrator sets up access control and security policies in Oracle E-Business Suite by defining roles, role inheritance hierarchies, role categories, and registration processes. These components specify the different levels of access to various application menus and data that are available to administrators.

Defining Role Categories

As part of the Oracle E-Business Suite RBAC model, Oracle User Management introduces Role Categories. Administrators can create role categories to bundle roles and responsibilities to make the process of searching for roles and responsibilities easier. In the Oracle User Management Overview section, see Role Based Access Control (RBAC), page 2-3.

Steps

1. Log on as a user that is assigned the Security Administrator role (typically as sysadmin), select the User Management responsibility in the navigator and then click the **Role Categories** subtab.

2. Go to the editable table, click the **Update** button and then click the **Create Lookup Code** button.

3. Enter the required information in the Create Lookup Code fields and click the **Apply** button.

Creating and Updating Roles

In Oracle E-Business Suite, a role represents a job function that confers the privileges required to perform that job. Roles can be defined to determine what applications (responsibilities) as well as what data and functions within those applications users can access. In the Oracle User Management Overview section, see Role Based Access Control (RBAC), page 2-3.

Steps

1. Log on as a user that is assigned the Security Administrator role (typically as sysadmin), select the User Management responsibility in the navigator and then click the **Roles & Role Inheritance** subtab.

2. Click the **Create Role** button.

3. Enter the required information to configure your role and optionally continue to configure it by accessing the following:

   - **Permissions**, page 3-3. Use this tab to assign permissions to your role.

     Delegated Administration Setup Using the Security Wizard

     Information in this section only applies to delegated administration roles in the context of the Oracle User Management application.

     - **User Administration**, page 3-10. Enables you to determine the set of users that can be managed by administrators to whom your role is assigned. The administrator can assign or revoke user accounts and roles for the users you specify here.

     - **Organization Administration**, page 3-9. Enables you to determine the external organizations that can be viewed in Oracle User Management by administrators to whom your role is assigned.

     - **Role Administration**, page 3-12. Enables you to determine which roles the administrator can assign to or revoke from the set of users specified in the User Administration section.

4. Click **Save** or **Apply** to save your changes.

5. Optionally update the role by performing the following:

   1. Locate the role you want to modify by using the Search fields or by expanding the appropriate nodes in the Role Inheritance Hierarchy menu.

   2. Click the **Update** icon and modify the role as required.

Guidelines

The **Save** button saves your changes and continues to display them in the current page. The **Apply** button saves your changes and returns to the previous page. You can optionally organize your roles using role categories during the process of creating and updating roles, otherwise they will be stored under the "Miscellaneous" role category by default. For more information, see role categories, page 3-1. You can also define any required subordinate roles or superior roles through role inheritance hierarchies, page 3-15.

Security Wizard

The Security Wizard page lists the security wizards available to the currently logged-in user. After launching the wizard by clicking its name, the user can use it to set up the data security policies associated with the role. After completion of the wizard, the user will be returned to the Create/Update Role UI.

Assigning Permissions to Roles

You can assign permissions to a role by creating a grant that specifies the navigation menu, permission sets, and/or the data security policies that are available at runtime to the role's assignees. Menus and permission sets in turn include individual functions and permissions. In the Oracle User Management Overview section, see Role Based Access Control (RBAC), page 2-3.

Steps

1. Log on as a user that is assigned the Security Administrator role (typically as sysadmin), select the User Management responsibility in the navigator and then click the **Roles & Role Inheritance** subtab.

2. In the Role Inheritance Hierarchy, access the role to which you want to assign a permission and click the **Update** icon.

3. Click the **Permissions** subtab and the click **Create Grant** button.

4. Define the grant by entering the required information and clicking **Next**:

    1. Enter the required information to identify the grant, such as Name and Effective From date.

    2. **Security Context**. These optional parameters restrict the availability of the permissions being assigned. If you do not define the security context, then permissions are available to users in all contexts. Security contexts are also referred to as *Activation Contexts*.

        1. **Operating Unit**. In many cases, an organization consists of several different

operating units. You can limit your grant to only be active in the context of an individual operating unit.

2. **Responsibility**. Responsibilities determine the applications that can be accessed by users. You can optionally limit your grant to be available only in the context of an individual responsibility, or with all responsibilities.

3. **Data Security**. You must select a business object when you create Data Security policies. For more information, see the Oracle Application Object Library Security chapter, page 4-1.

5. If you have defined a specific object in the preceding step, then choose the object data context for the object, also referred to as the *data scope*. Specifying the object data context provides an additional level of access granularity for the object. Choose one of the following from the Data Context menu:

   - **All Rows**. This option provides access to *all rows* for the database object. For example, if the database object is a book, creating a data security policy for all rows of the object will provide access to all books catalogued in the database.

   - **Instance**. This option provides access to an *instance* of the object. A specific instance generally corresponds to a single row in the database, and is typically identified by the primary key value for the object. For example, a data security policy for the book object could contain a unique ISBN number, to return only one book from the database.

   - **Instance Set**. This option provides access to a *related set of instances* of the object. This set is specified as a predicate on the attributes of the object. The predicate is expressed as a SQL WHERE clause, and can optionally be implemented as a VPD policy. For example, a data security policy could include an instance set for all books published in the year 2013.

6. Select the required permission set or navigation menu containing the functions (permissions) that you wish to assign to the role, by choosing an option from the LOV.

7. Review your grant information and click **Finish**.

Searching For Assigned Roles

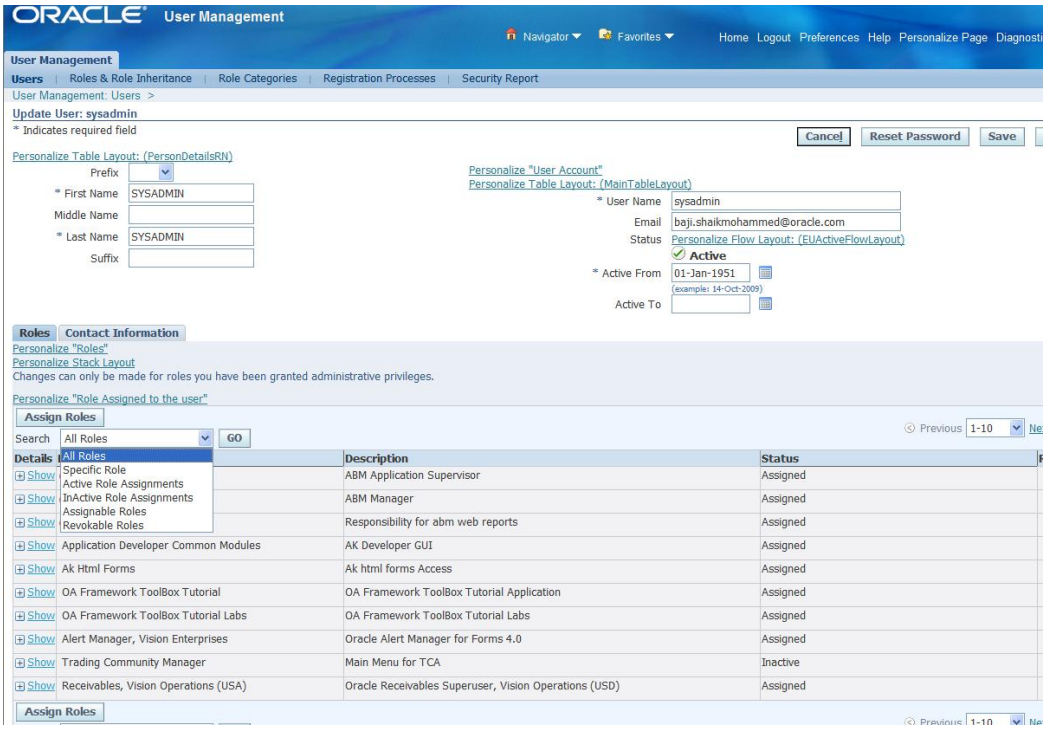
The number of roles and responsibilities in some installations can be in the tens of thousands, or even more. Since any given user can potentially have a very large number of roles and responsibilities assigned, it can be very time-consuming to determine which roles have been assigned to which users.

A search capability allows administrators to look for:

- **All Roles**: Find all roles assigned to the current user
- **Specific Role**: Find if a role has been assigned to an user, and quickly change the attributes associated with it.
- **Inactive Role Assignments**: Find all inactive User-Role assignments.
- **Active Role Assignments**: Find all active User-Role assignments.
- **Assignable Roles**: Find all roles for which the current logged in administrator has "Can Assign" privilege.
- **Revokable Roles**: Find all roles for which the current logged in administrator has "Can Revoke" Privilege.

Steps

1. Navigate to the User Management responsibility and then click the Users sub-tab.
2. Use the search fields to locate the required people or users.
3. Click on the "Update" icon.
4. Select any of the above specified criteria, such as "Specific Role" in the drop-down menu and "Sales Manager" in the text box.
5. Click on the "Go" button.



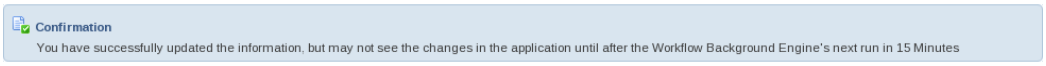
Examples

- **All Roles**: If a user selects "All Roles" in the drop down, all the roles assigned to the user will be displayed.
- **Specific Role**: If a user selects "Specific Role" from the drop down menu, another text box user appears to allow entry of a role (for example, User Management). A list of users with that role will then be displayed.
- **Inactive Role Assignments**: If a user selects "Inactive Role Assignments" from the drop down menu, all inactive User-Role assignments will be displayed.
- **Active Role Assignments**: If a user selects "Active Role Assignments" from the drop down menu, all active User-Role assignments will be displayed.
- **Assignable Roles**: If a user selects "Assignable Roles" from the drop down menu, all roles for which the current logged in administrator has "Can Assign" Privilege will be displayed.
- **Revokable Roles**: If a user selects "Revokable Roles" from the drop down menu, all roles for which the current logged in administrator has "Can Revoke" Privilege will be displayed.

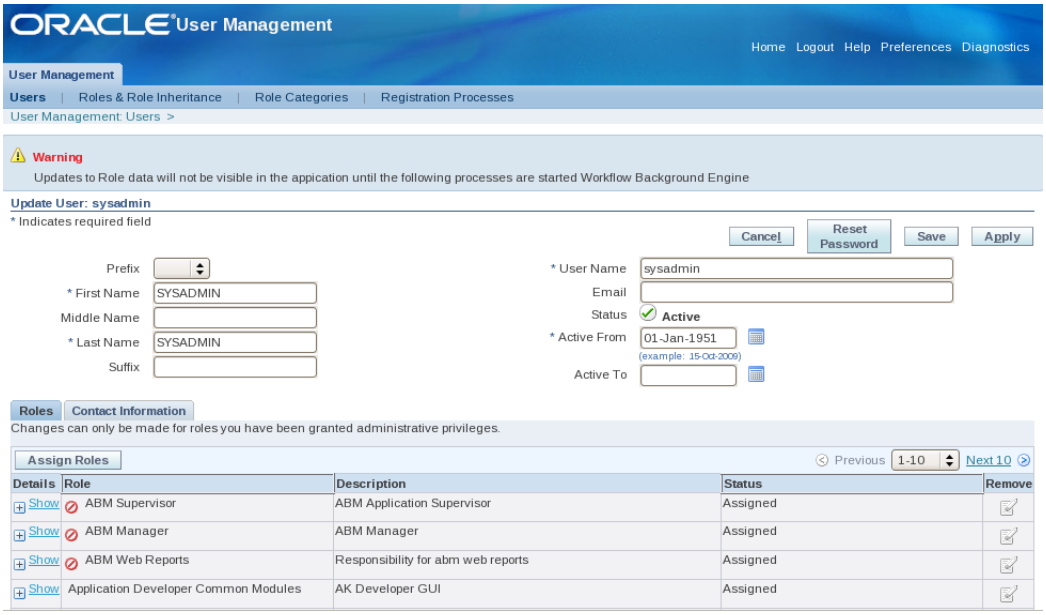
Diagnostics for User-Role Assignment

UMX is heavily dependent on Concurrent Manager, Deferred Agents and Background engines. If any of these are down, the assignments do not take place or may only take place after an excessively long time.

A diagnostic feature built in the User-Role Assignment page checks that the required processes are running when an update is submitted. If they are all running, it reports how much time may be needed for the changes to be effected.



If one or more are down, the diagnostic feature displays a warning and advises which processes will need to be started in order for the changes to be made successfully.



Creating Instance Sets and Permission Sets

Delegated administration setup for User Administration requires the creation of instance sets and permission sets. All possible combinations of permissions are seeded as permission sets that are available from this screen. A data security object, UMX_SYS_ACCT, represents system accounts. Administrators can create instance sets against this object to specify system accounts that can be managed.

Steps

1. Log on as a user who has been assigned the Security/LSA Administrator role (typically as sysadmin), select the User Management responsibility in the

navigator.then click the Roles & Role Inheritance subtab.

2. In the role hierarchy, access the role to which you want to assign user administration privileges, and click the Update icon.

3. Click on the Security Wizards button.

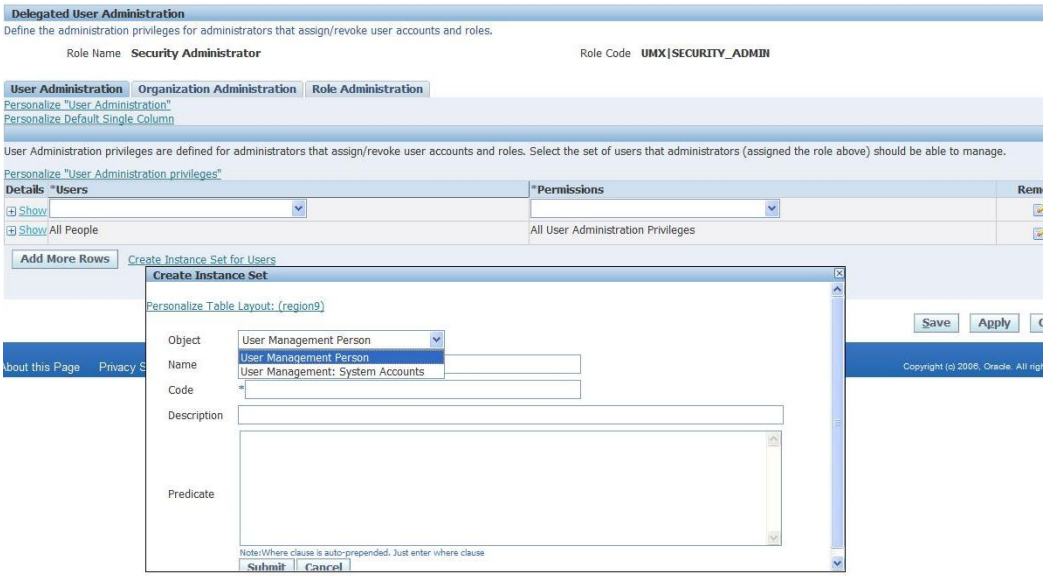
4. Click on the Run Wizard icon for "User Management: Security Administration Setup".

5. Click the User Administration sub-tab, then click the Add More Rows button.

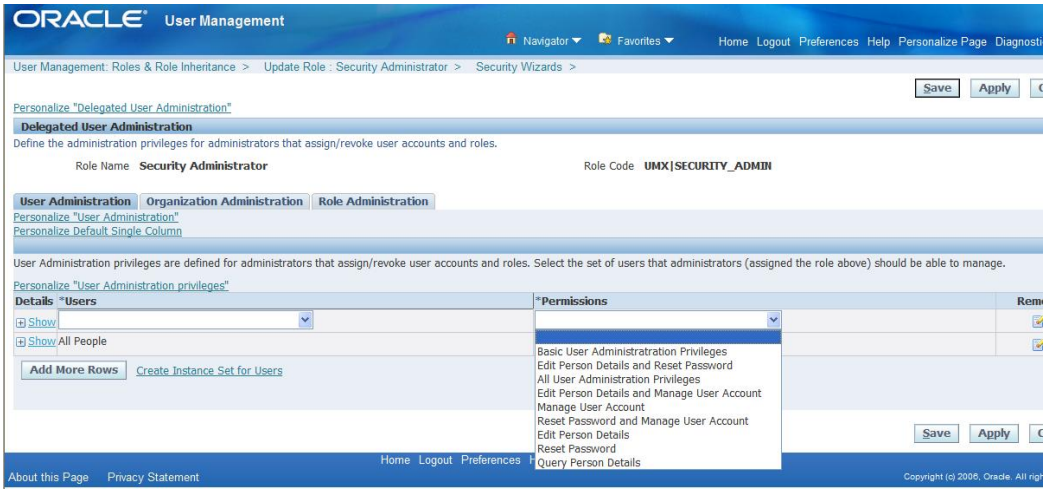
6. In the Users field, select the set of users that can be managed by Administrators to whom the role is assigned. The drop down list contains various data security policies that relate to the User Management Person Object (UMX_PERSON_OBJECT) and User Management: system accounts object (UMX_SYS_ACCT). The user can now create his own policies on both these objects by clicking on the link "Create Instance Set For Users".

7. In the Permissions field, select the permissions to be associated with the delegated administration role. The Permissions drop down list includes permission sets that contain permissions associated with the User Management Person object and User Management: system accounts object. All possible combinations of the existing permissions have been seeded here, enabling organizations to add permission sets based on their general business needs and the level of granularity they prefer for administering users.

Create Instance Set (Data Security Policy)



Selecting Required Permission Set (Data Security Policy)



This capability means that there is no longer any need to navigate to the Functional Administrator or Functional Developer responsibilities when creating permission sets and instance sets, so that the entire delegated administration set up should now take no more than a few minutes.

Defining Delegated Administration Privileges for Roles

Delegated Administration Privileges determine the users, roles and organization information that delegated administrators (local administrators) can manage. Each

privilege is granted separately, yet the three work in conjunction to provide the complete set of abilities for the delegated administrator. In the Oracle User Management Overview section, see Delegated Administration, page 2-6.

Defining User Administration Privileges for Roles

A local administrator must be granted User Administration Privileges to determine the users and people the local administrator can manage. Local administrators can be granted different privileges for different subsets of users. For example, a local administrator can be granted privileges only to query one set of users, and granted full privileges (including update and reset password) for another set. Local administrators cannot query users for which they do not have administration privileges.

Oracle User Management ships with the following seeded permissions for defining user administration privileges for roles:

Seeded User Administration Permissions

Function Code	Display name	Description
UMX_OBJ_ACTIVATE_ACCT	Create, Inactivate, Reactivate User Account, Update Username	Permission for creating, inactivating, and reactivating user accounts, and updating username. Must be granted with a data security policy on the User Management Person.
UMX_OBJ_EDIT_PERSON	Edit Person Details	Permission for editing person details. Must be granted with a data security policy on the User Management Person (UMX_PERSON_OBJECT) business object.
UMX_OBJ_PASSWD_MGMT	Reset Password	Permission to reset passwords. Must be granted with a data security policy on the User Management Person (UMX_PERSON_OBJECT) business object.

Function Code	Display name	Description
UMX_OBJ_VIEW_PERSON	Query Person Details	Permission to query person details Must be granted with a data security policy on the User Management Person (UMX_PERSON_OBJECT) business object. **Note:** This is the minimum permission required by any security administrator that wishes to manage people and users in Oracle User Management.
UMX_SYSTEM_ACCT_ADMINSTRATION	Maintain System Accounts (users not linked to a person)	Create, Inactivate, Reactivate, Reset Password for all System Accounts (defined as user accounts not associated with a person). **Note:** Only grant to System Administrators.

Steps

1. Log on as a user that is assigned the Security Administrator role (typically as sysadmin), select the User Management responsibility in the navigator and then click the **Roles & Role Inheritance** subtab.

2. In the role hierarchy, access the role to which you want to assign user administration privileges and click the **Update** icon.

3. Click on the Security Wizards button.

4. Click on the Run Wizard icon for "User Management: Security Administration Setup".

5. Click the **User Administration** subtab and then click the **Add More Rows** button.

6. In the Users field, select the set of users that can be managed by Administrators to whom the role is assigned. The drop down list contains various data security policies that pertain to the User Management Person Object

(UMX_PERSON_OBJECT). Oracle User Management ships with sample data security policies for users. Organizations can use these policies or create their own. For more information, see Defining Data Security Policies, page 3-14.

7. In the Permissions field, select the permissions that you wish to associate with the delegated administration role. Permissions determine the actions an administrator can perform when managing the set of users defined in the previous step. The Permissions drop down list includes permission sets that contain permissions associated with the User Management Person object. Different combinations of the existing permissions can be grouped into new permission sets, enabling organizations to add permission sets based on their business needs and the level of granularity they prefer for administering users. For more information, see Permission Sets, page 4-55.

8. Click **Save** or **Apply** to save your changes.

Guidelines

Delegated administration can provide different permissions on different subsets of users. Once you define users and permissions for a role, you can optionally view the permissions that belong to the permission set by clicking the **Show** node. You can also remove the user administration privileges for a set of users by clicking the **Remove** icon.

Defining Role Administration Privileges for Roles

Role Administration Privileges define the roles that local administrators can directly assign to and revoke from the set of users they manage.

Oracle User Management ships with the following seeded permission for defining role administration privileges for roles:

Seeded Role Administration Permission

Function Code	Display Name	Description
UMX_OBJ_ADMIN_ROLE	Assign/Revoke Role	Permission for assigning/revoking roles in the User Management application. Must be granted with a data security policy on the User Management Role (UMX_ACCESS_ROLE) business object.

Steps

1. Log on as a user that is assigned the Security Administrator role (typically as

sysadmin), select the User Management responsibility in the navigator and then click the **Roles & Role Inheritance** subtab.

2. In the navigation menu access the role for which you want to define role administration and click the **Update** icon.

3. Click on the Security Wizards button.

4. Click on the "Run Wizard" icon for "User Management: Security Administration Setup".

5. Click the **Role Administration** link and use the Available Roles fields to search for the role(s) that you want to associate with this role and which administrators can manage once they are assigned this role.

6. Select the desired role(s), move them to the Selected Roles column and click **Save** or **Apply**.

Guidelines

The **Save** button saves your changes and continues to display them in the current page. The **Apply** button saves your changes and returns to the previous page.

Defining Organization Administration Privileges for Roles

Organization Administration Privileges define the external organizations a local administrator can view in Oracle User Management. This privilege enables an administrator to search for people based on their organization, assuming the local administrator has also been granted access to view the people in that organization (User Administration Privileges). Depending on what administration account registration process has been granted, the administrator may have the ability to register new people for that organization.

Oracle User Management ships with the following seeded permission for defining organization administration privileges for roles:

Seeded Organization Administration Permission

Function Code	Display Name	Description
UMX_OBJ_VIEW_RLTNSHPS	Query/Register Organization Relationship	Permission to query/register organization relationship. Must be granted with a data security policy on the User Management Organization (UMX_ORGANIZATION_OBJECT) business object.

Steps

1. Log on as a user that is assigned the Security Administrator role (typically as sysadmin), select the User Management responsibility in the navigator and then click the **Roles & Role Inheritance** subtab.

2. In the navigation menu access the role to which you want to define organization administration and click the **Update** icon.

3. Click on the Security Wizards button.

4. Click on the "Run Wizard" icon for "User Management : Security Administration Setup".

5. Click the **Organization Administration** link and then click the **Assign Organization Privileges** button. The drop down list contains various data security policies that pertain to the User Management Person Object (UMX_PERSON_OBJECT). Oracle User Management ships with sample data security policies for organization administration privileges. Organizations can use these policies to create their own.

6. Search for and select the appropriate organization privileges.

7. Click **Save** or **Apply** to save your changes.

Guidelines

The **Save** button saves your changes and continues to display them in the current page. The **Apply** button saves your changes and returns to the previous page.

Defining Data Security Policies

With Oracle E-Business Suite, organizations can use Data Security to manage permission assignments that control access to objects. Data Security policies can only be defined for applications that have been written to utilize the Data Security Framework. For more information, see Data Security, page 4-15. Access to the specific object must be formed with a specified Data Security Policy (also referred to as the Data Scope or Access Policy). The Data Security Policy restricts operations so that they only can be performed on a subset of instances of the corresponding database object. For more information, see Object Instance Sets, page 4-41.

Steps

1. Log on as a user with the Functional Developer responsibility, click the **Functional Developer** responsibility in the navigator, navigate to the **Security** tab and then click the **Objects** subtab.

2. Search for and access the object for which you want to create data security policies. For example, to locate the User Management Person business object

(UMX_PERSON_OBJECT), enter "UMX%" in the Code field, click the **Go** button, and then click User Management Person object (UMX_PERSON_OBJECT) in the search results list. For any object for which you are creating a policy, ensure that the SQL statement returns the primary key value for that object. In this example, this is a list of person party IDs.

3. Click the Object Instance Sets subtab. Click the **Create Instance Set** button to create a new object instance set or click the **Update** icon to modify an existing one.

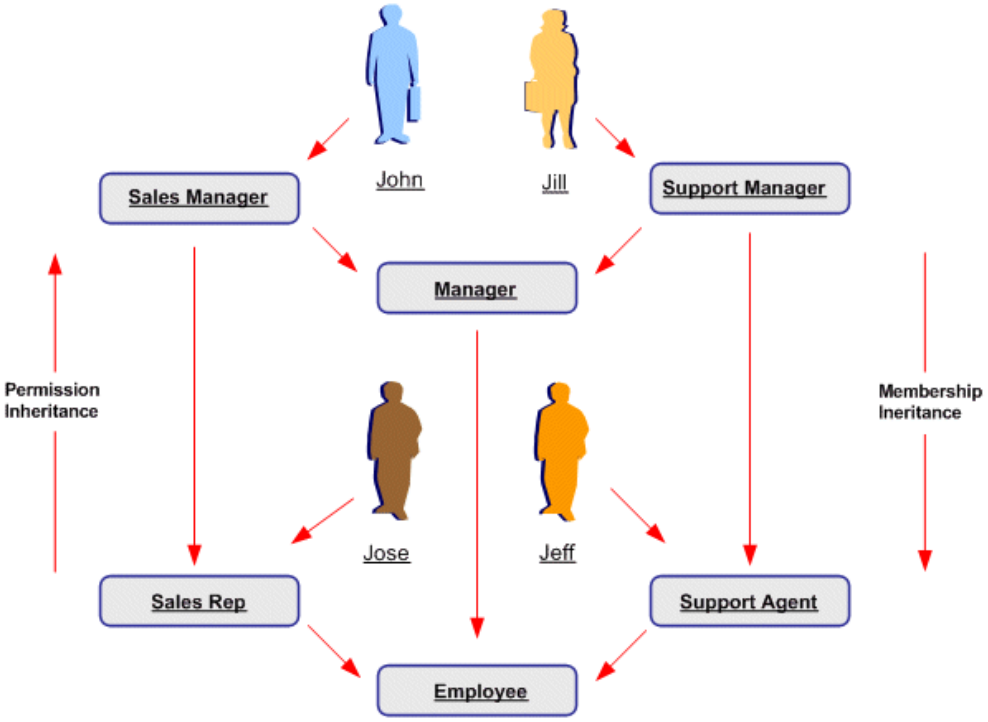
4. Enter the required information and then click the **Apply** button.

> **Caution:** For performance reasons, ensure that SQL predicates are tuned properly. For security reasons, ensure that they are tested and that they return the correct result. Oracle is not responsible for the performance or correctness of data security policies defined by organizations.

Defining Role Inheritance Hierarchies

With role inheritance hierarchies, a role can contain sub roles. When a user is assigned a role, the user inherits the privileges defined for that role and for all of its sub roles. For example, the Sales Manager role can contain the Manager and Sales Rep roles, both of which in turn contain the Employee role. Any individual who is granted the Sales Manager role automatically inherits the Manager, Sales Rep and Employee roles.

Role Inheritance Hierarchies



With Role Inheritance Hierarchies, roles inherit the permissions assigned to their sub roles.

Steps

1. Log on as a user that is assigned the Security Administrator role (typically as sysadmin), select the User Management responsibility in the navigator and then click the **Roles & Role Inheritance** subtab.

2. Locate the role for which you want to create a role inheritance hierarchy by using the Search fields or by expanding the appropriate nodes in the Role Inheritance Hierarchy menu. If you are building a role inheritance hierarchy that contains several roles, start with highest level role to which you want to add inherited subordinate roles.

3. Click the **Add Node** icon next to this role.

4. In the resulting menu, search for the role either by using the Search fields or by locating it in the Role Inheritance Hierarchy menu.

5. Select the role and then click the **Select** button or the **Quick Select** icon.

6. Repeat this process until you have added all of the required subordinate roles to their corresponding super roles. You can optionally verify the results by expanding the nodes for all super roles within your role inheritance hierarchy. You can also remove any subordinate roles by clicking the **Remove Node** icon.

Deployment Options

Organizations can use different deployment options for role inheritance hierarchies depending on their requirements.

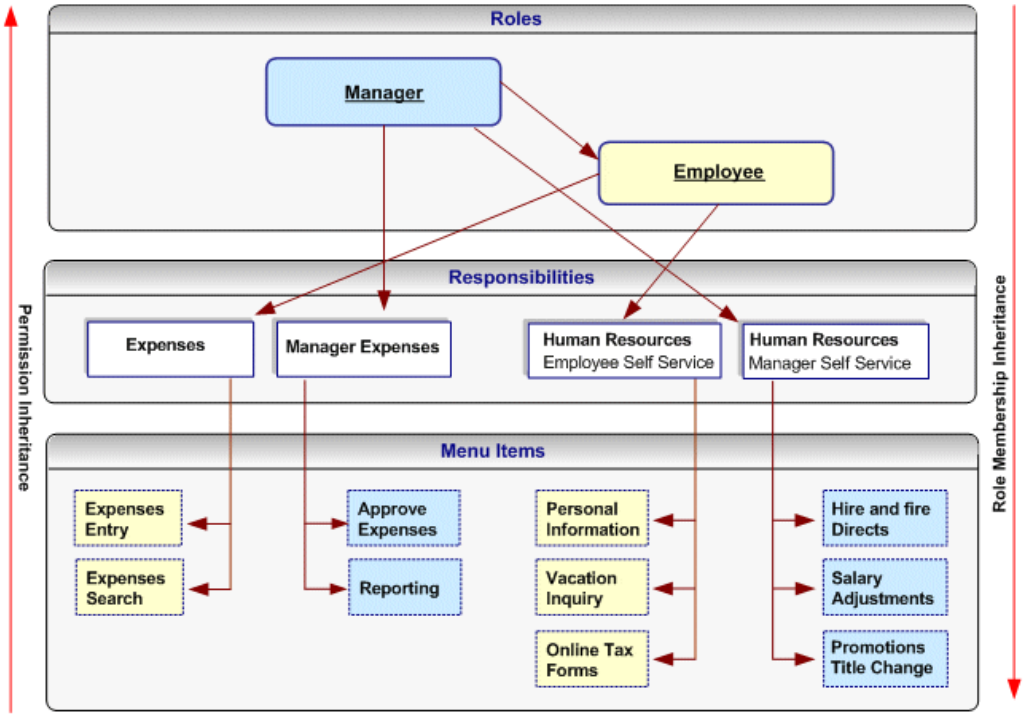
Assigning Existing Responsibilities to Roles Using Role Inheritance

Organizations that have already defined their responsibilities can utilize RBAC by creating roles and assigning their existing responsibilities to those roles. For example, an organization could create an Employee role and a Manager role, and add to these the Expenses and Human Resources responsibilities that it wishes to make available to employees and managers respectively. Then, instead of manually assigning or revoking each of these responsibilities to or from its employees, the organization can simply assign or revoke the Employee and Manager roles as required. Since the Manager role inherits the employee role, managers that are assigned the Manager role also inherit all the responsibilities and privileges associated with the Employee role.

In the following example, a Human Resource Manager inherits the Human Resources Manager Self Service responsibility through the Manager role as well as the Human Resources Employee Self Service responsibility, which the Manager role inherits from the Employee role.

> **Note:** In this section, references to the Expenses and Human Resources responsibilities are used as examples only. Some applications may require organizations to create multiple responsibilities to operate with their existing security models. For more information, please consult the application-specific documentation.

Assigning Existing Responsibilities to Roles Using Role Inheritance



Steps

1. Create roles representing the required job functions such as Manager and Employee.

2. Define a role inheritance hierarchy. For more information, see Defining Role Inheritance Hierarchies, page 3-15.

3. Ensure the responsibilities are inherited by their corresponding roles.

4. Assign the roles to users as required.

Fully Utilizing RBAC and Role Inheritance to Determine Access to an Application

In older releases of Oracle E-Business Suite, access to individual functions within an application could only be defined through responsibilities, menu hierarchies, and menu exclusions. Responsibilities had the dual role of defining application navigation menus and granting permissions to the application. New responsibilities with one of the following had to be defined for each set of users with different job functions that required access to a set of pages within an application:

- A completely new menu hierarchy for each responsibility, or

- A common menu covering the superset of all functions within the application, and menu exclusion rules defined for each responsibility.

The Human Resources application, for example, typically required a minimum of two responsibilities, one for employees and one for managers.

Separating Navigation Menus and Access Control

Oracle User Management provides new alternatives for defining access to an application with RBAC and Role Inheritance, allowing organizations to separate navigation menus from access control. Responsibilities can now be defined to represent an application itself and as a result, only one responsibility may be required for each application. A menu can be tailored for each application with specific consideration to usability and end user navigation experience. Access to parts of the application (responsibility) and its corresponding menu hierarchy are instead controlled by different roles, each representing a specific job function or set of people.

Benefits

Using this mechanism for determining access control provides several benefits.

- Administration and changes can be accomplished with minimal effort:
  - A new page only has to be added to a single menu.
  - The permission to access a new page, only has to be granted once to the lowest level (subordinate role) in the role inheritance hierarchy.
  - An entirely new application (responsibility) can automatically be assigned to a set of people by simply defining it as the subordinate role of an existing role.
  - Permissions to access the various pages and functions within a new application should only be assigned at the lowest level in the role inheritance hierarchy. The permissions are then automatically inherited by all superior roles in the hierarchy.
  - Revoking access to a page, or an entire application, can be accomplished as easily as adding access.
- Improved end user experience. In the applications navigator, end users will see a list of applications to which they have access. Access to the various functions within each application is determined by the roles assigned to the end user.

Steps

> **Note:** In this section, references to the Expenses and Human Resources responsibilities are used as examples only. Some applications may require organizations to create multiple responsibilities to operate with their existing security models. For more information, please consult the application-specific documentation.

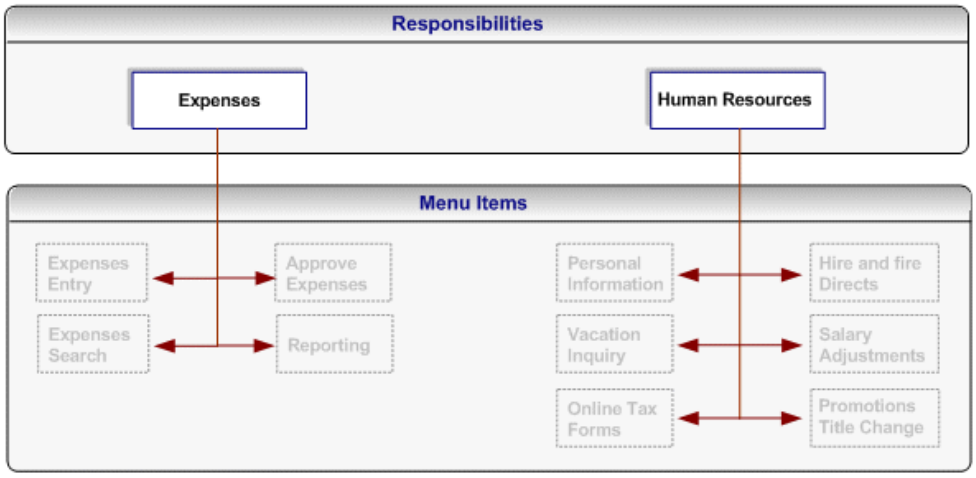
1. Define a new responsibility that will be used to represent a specific application such as Expenses or Human Resources. For more information, see Defining a Responsibility, page 4-4.

2. Design a complete menu that includes all the menu functions within an application as well as any required submenus, and attach this menu to the new responsibility. For example, both the Expenses and Human Resources responsibilities would include all employee and manager menus. For more information, see Defining a New Menu Structure, page 4-33.

3. Following the "principle of least privilege", all the menu options within the application (each menu item corresponds to a function/permission) should be disabled by default. To accomplish this, remove the selection from the "grant" checkbox for each menu item:

   The following figure illustrates application responsibilities (in this case, Expenses and Human Resources) with all their menus disabled:

Responsibilities Representing an Entire Application with Disabled Menus

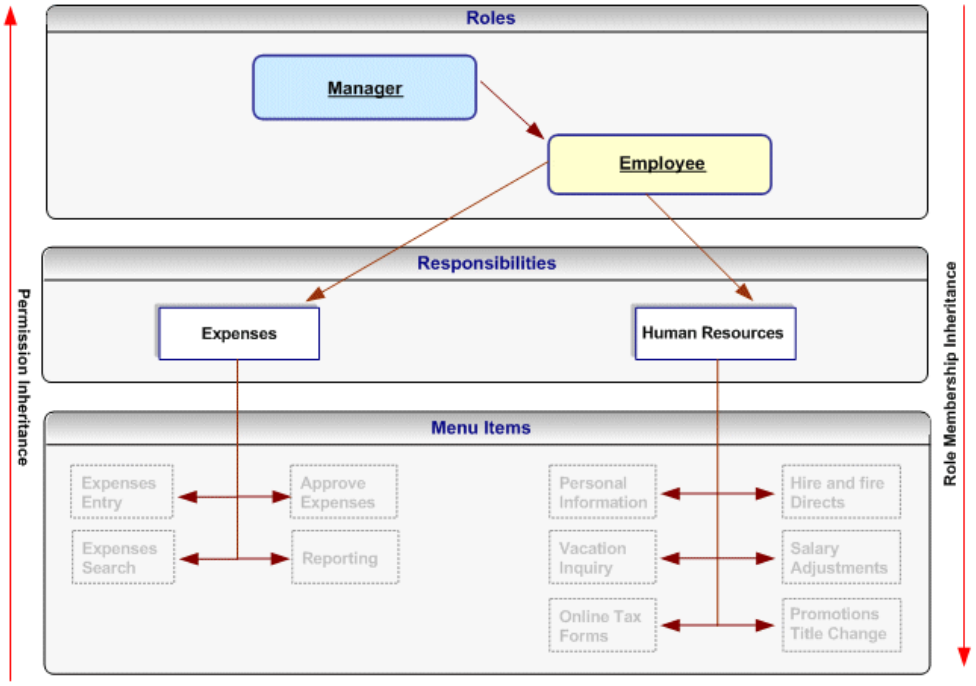


Note: A user cannot access any of the menu items (functions) within the application if you assign the responsibility to the user at this stage.

4. Create roles representing the people with various job functions that require access to the application, for example, a Manager role and an Employee role. For more information, see Creating and Updating Roles, page 3-2

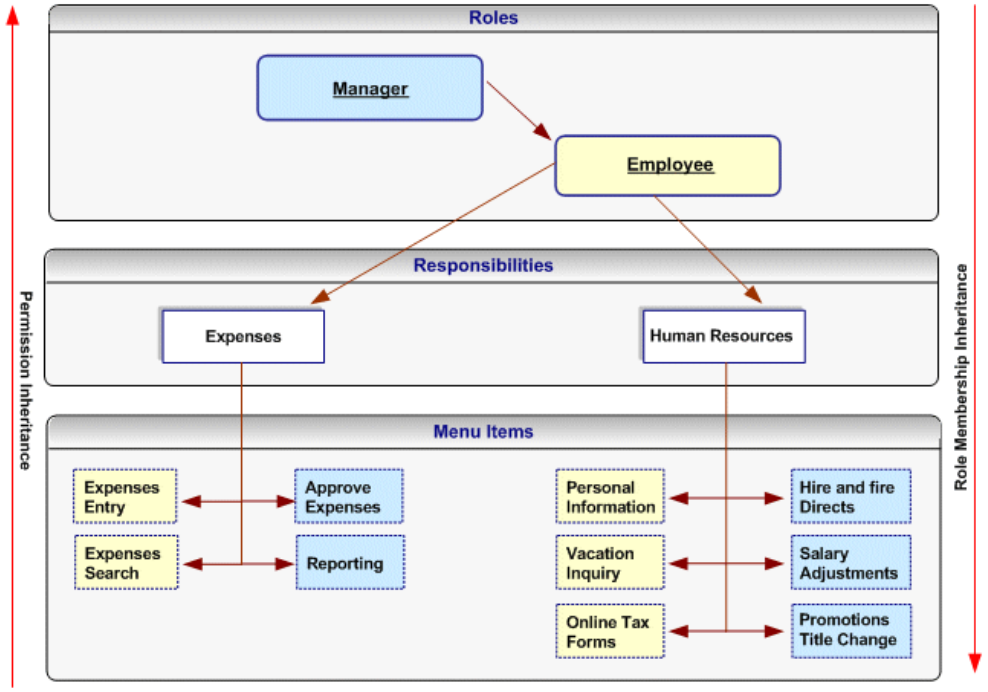
5. Define role inheritance relationships. For more information, see Defining Role Inheritance Hierarchies, page 3-15 For example, the Manager role should inherit the Employee role, and the Employee role should inherit the Expenses and Human Resources responsibilities. The following figure illustrates a role inheritance relationship in which a role inherits the responsibilities that are inherited by its subordinate role:

Role Inheritance Relationship in Which a Role Inherits the Responsibilities Inherited by its Subordinate Role



6. Assign permissions to each role. For more information, see Assign permissions to each role, page 3-3. Each permission maps to a menu item (function) within the application (responsibility) that should be available to the users to whom the role is assigned. For example, an organization will grant the employee-related permissions from the Expenses and Human Resources responsibilities to the Employee role, and will grant the manager-related permissions for these responsibilities to the Manager role. Consequently, the manager role will have access to all the menu items within these responsibilities, but the Employee role will only have access to the Employee-related functions.

Permissions, Roles and Inheritance



Permissions assigned to a subordinate role in the role inheritance hierarchy are automatically inherited by the superior roles. For example, if you grant the permission for accessing the Online Tax Forms page to the Employee role, anyone with the Manager role will automatically have access to this page through role inheritance. Because the Hire and Fire Directs page is only granted to the Manager role, it is not available to users that are only assigned the Employee role.

Permissions are always assigned through permission sets, which represent named sets of functions (permissions). When determining what permissions (functions/menu items) should be granted to each role, you may have to create new permission sets, page 4-55. Menus and permission sets are stored in the same tables in the database; which means that they are interchangeable (both can be used) to assign permissions.

7. Optionally assign any additional permissions and data security policies to roles as required by each application.

Guidelines

Oracle User Management ships with the following Customer Administrator and Security Administrator roles. These roles illustrate how to setup Roles and Role Inheritance to determine user access within an application (responsibility). Both roles inherit the User Management responsibility but each role is granted different permissions and data security policies. The User Management responsibility has the grant flag removed for all functions (permissions) in the menu hierarchy. Instead, these

permissions are granted to the role depending on each role's requirements:

Role Attributes and Roles

Role Attributes	Customer Administrator	Security Administrator
Permission Sets	• User Maintenance UIs	• User Maintenance UIs • Setup screens • Maintain system accounts
User Administration	• Data security policies to manage people and user accounts for the customer administrator's own organization • Typically, the Customer Administrator can only assign or revoke a subset of roles	• Data security policies to manage all people and user accounts • The Security Administrator can assign or revoke all roles

Creating and Updating Registration Processes

Registration processes are predefined registration components that enable end users to perform some of their own registration tasks, such as requesting new accounts or requesting additional access to the system. They also provide administrators with a faster and more efficient method of creating new user accounts.

Oracle User Management provides four types of registration process:

- Self Service Account Requests
- Self-Service Requests for Additional Access
- Account Creation by Administrators
- Administrator Assisted Request for Additional Access

In the Oracle User Management Overview section, see Registration Processes, page 2-8.

Steps

Registration processes all use the same infrastructure and processing logic. Steps for

defining a registration process will vary depending on the type of registration process you are creating.

1. Log on as a user that is assigned the Security Administrator role (typically as sysadmin), select the User Management responsibility in the navigator and then click the **Registration Processes** subtab.

2. Click the **Create Registration Process** button.

3. Enter the required information for the Registration Process Description and click the **Next** button. This information specifies:

   - **Role**. The role with which you optionally associate the registration process and that is assigned to the user at the end of the registration process once the request has been processed.

   - **Type**. The type of registration process you wish to create.

   - **Registration Process Code**. The unique identifier for the registration process.

   - **Display Name**. The display name for the registration process.

   - **Description**. A description of the registration process.

   - **Application**. The application with which the registration process is classified. This can be used to help query the registration process.

   - **Active From**. The date from which the registration process is first active.

   - **Active To**. The date you can optionally specify to terminate the registration process.

4. Enter the runtime execution information for the registration process and click the **Next** button. This information specifies:

   - **Registration Start Page**. The first page (which is represented as a function) in the registration process that captures any additional user registration information. This is optional unless you are creating a Self Service Account Request registration process.

   - **Notification Event**. The workflow business event that invokes a workflow. The notification workflow subscribes to the event and subsequently sends notifications to the approver or to the user.

   - **Approval Transaction Type**. The set of approval routing rules that is interpreted at runtime by the Oracle Approval Management rules engine. The rules determine whether approval is required and by what set of users based on user transaction types you have defined specifically for use with Oracle User

Management.

- **Business Event Name**. Custom business event that will be raised by Oracle User Management with context information for processing.

5. Enter the eligibility information for the registration process by selecting the appropriate roles or groups from the Available Groups column and clicking the **Submit** button. For Self-Service Requests for Additional Access, eligibility defines the users who are able to register for the role associated with the registration process. For Account Creation by Administrators, eligibility determines what administrators can register new users through the registration process. Oracle User Management ships with the following seeded permissions for defining eligibility policies:

Seeded Permissions for Self Service Additional Access and Account Creation by Administrators Eligibility

Function Code	Display Name	Description
UMX_OBJ_ADMIN_CRTN_FLOW	Administrator Assisted Account Creation	Permission representing "Administrator Assisted Account Creation" registration processes. This must be granted as a data security policy on the Registration Process (UMX_REG_SRVC) business object.
UMX_OBJ_ROLE_ELGBLTY	Self Service Eligibility	Permission representing registration processes for additional access. Determines the set of end users that should be eligible to register for a given role/registration process. This must be granted as a data security policy on the Registration Process (UMX_REG_SRVC) business object.

6. Register subscriptions to the appropriate business events raised by Oracle User Management, and ensure that your subscription logic writes the registration data

into the appropriate destination schemas.

7. Optionally update the registration process by searching for it and clicking the **Update** button in the search results page.

8. Optionally set the following profile options for registration processes of type Self Service Account Request:

   - **Registration Links**. Oracle User Management provides support for displaying different registration links on the login page based upon the mid-tier through which the login page is accessed. Organizations can set the server level profile option, "UMX: Register Here Link - Default Registration Process" (UMX_REGISTER_HERE_REG_SRV) to specify different destinations for the registration link.

   - **Registration Parameters**. The registration link can also contain additional parameters that are not known at design time. These parameters are available at all stages of the registration process; for example, for routing approval requests. You can set the server level profile option "UMX: Register Here Link - Default Registration Parameters" (UMX_REGISTER_HERE_REGPARAMS) for this purpose. The format for setting this profile option is: "ParamName1=ParamValue1&ParamName2=ParamValue2":

   - **UI-specific Parameters**. Organizations can additionally specify parameters used to control the rendering of the registration user interface, such as the menu displayed in the registration UI. The server level profile option, "UMX: Register Here Link - Default HTML Parameters" (UMX_REGISTER_HERE_HTMLPARAMS) can be set for this purpose. The format for setting this profile option is: "ParamName1=ParamValue1&ParamName2=ParamValue2":

     Note: The Apache server may need to be restarted for the changes to take effect.

Configuring the User Name Policy

The Oracle User Management registration infrastructure supports a *configurable user name policy*. This policy is used to generate a suggested user name in the sample user creation flows shipped with the application, as well as for validating the chosen user name format.

Note: Oracle User Management is supplied with a default policy that identifies users by their email address.

Seeded User Name Policies

The following table lists the seeded user name policies that are shipped with Oracle E-Business Suite.

Seeded User Name Policies

Code	Description
UMX_USERNAME_POLICY:EMAIL_ADDRESS	User name policy with email address format defined as the policy.
UMX_USERNAME_POLICY:NONE	User name policy with no restriction on user name format.

Administrators can configure either of these seeded policies. In addition to these, custom policies can also be implemented if desired.

> **Note:** For details of how to create a custom policy, see the *UMX Developer's Guide*, Knowledge Document 399400.1 on My Oracle Support.

Configuration of user name policy is a three-stage process.

Stage 1 - Suggested User Name Generation Subscription Setup

1. Log on as a user that is assigned the Workflow Administrator Web Applications responsibility (typically sysadmin).
2. Go to Workflow Administrator Web Applications > Business Events
3. From the Business Events page, search for the Business Event with the name *oracle.apps.fnd.umx.username.generate*.
4. Click on the Subscription icon to go to the Subscriptions page.
5. For the subscription corresponding to the policy, change the status to "Enabled".

Stage 2 - Validation Event Subscription Setup

1. Log on as a user that is assigned the Workflow Administrator Web Applications responsibility (typically sysadmin).
2. Go to Workflow Administrator Web Applications > Business Events
3. From the Business Events page, search for the Business Event with the name

oracle.apps.fnd.user.name.validate.

4. Click on the Subscription icon to go to the Subscriptions page.

5. For the subscription corresponding to the policy, change the status to "Enabled".

Stage 3 - Profile Option Setup

1. Log on as a user that is assigned the Functional Administrator responsibility (typically sysadmin).

2. Go to Functional Administrator > Core Services > Profiles

3. Search with the Profile Name of UMX: User Name Policy in the Maintain Profile Options page.

4. Click on the Update icon to go to the Update Profile Option page.

5. Choose a value corresponding to the policy and click on the Apply button.

Additional Requirements

- In all the three of the stages above, the values set must correspond to the same user name policy.

- The Listener and JVMs must be restarted after the user name policy is changed.

Delegated Administration Tasks

The Delegated Administration layer of Access Control in Oracle E-Business Suite enables local administrators to perform a variety of specifically defined administrative tasks. Once they are assigned the appropriate roles, local administrators manage the subset of users and people to which they have access by creating, updating, or disabling accounts, granting or revoking a limited subset of their organization's roles, and changing passwords.

Maintaining People and Users

Oracle User Management enables local administrators to manage people and users in the system. People are individuals in the system who may or may not possess a user account, whereas users are individuals in the system who possess user accounts. In addition, system administrators can also manage system accounts that are not linked to people.

Typically, people and users are managed by local administrators, who can perform the following tasks:

- Register new people (optional: requires access to have been granted to the "Account Creation by Administrators" registration process)
- Create, update, or disable user accounts
- Reset passwords
- Grant users access to different parts of the system by assigning or revoking roles

Common Prerequisites

The following are prerequisites for performing any delegated administration task listed in the preceding section. Each task may have additional prerequisites:

- A role that is granted the *User Maintenance UIs* (UMX_USER_ADMIN_UI_PERMS) permission set. The role must also inherit the User Management responsibility.
- Appropriate privileges for User Administration, Role Administration, and Organization Administration.
- The Query Person Details (UMX_PERSON_OBJECT) permission for the set of people and administrator can manage.
- Optionally, the Edit Person Details (UMX_OBJECT_EDIT_PERSON) permission for the set of people that the administrator can manage.
- For system administrators, the Maintain System Accounts (UMX_SYSTEM_ACCOUNT_ADMINISTRATION) permission.

Steps

1. Navigate to the **User Management** responsibility and then click the **Users** subtab.
2. Use the search fields to locate the required people or users.
3. Manage the generated list of people or users by clicking the required icon and performing the necessary steps in the resulting window. Options for managing people and users vary depending on the permissions assigned to the administrator. Oracle User Management ships with the following basic and advanced options for maintaining people and users:
    - Query users
    - Edit personal information
    - Reset password
    - Maintain account information (create, inactivate, reactivate accounts)

- Maintain system accounts
- Assign or revoke roles

Creating, Inactivating, and Reactivating User Accounts

Administrators can create a user account for any person in the system who does not already possess one.

Prerequisites

To create, inactivate, and reactivate user accounts, an administrator must be assigned the following:

- Common prerequisites, as detailed in the Maintain People and Users section, Common Prerequisites, page 3-29.
- The Create, Inactivate, Reactivate User Account (UMX_OBJ_ACTIVATE_ACCT) permission for the set of people that the administrator can manage.

By default, user names are derived from the person's email address.

Steps

1. Log in as a user with a role granting you access to the User Management responsibility, select the User Management responsibility in the navigator and click the **Users** subtab.
2. Search for the person for whom you wish to create an account and then click the **Create Account** icon next to the person's name if the account does not already exist. Your search will only generate results for the subset of users that you are eligible to manage.
3. Enter or modify the required information and click the **Submit** button.

Guidelines

Oracle recommends that you base user names on the person's email address.

Resetting User Passwords

Oracle User Management enables administrators to reset passwords for the set of users in the system that they manage. When the password is reset, an email message is sent to the user using the UMX Password (UMXUPWD) workflow.

Prerequisites

To reset user passwords, an administrator must be assigned the following:

- In the Maintain People and Users section, see the Common Prerequisites, page 3-29.
- The Reset Password (UMX_OBJ_PASSWD_MGMT) permission for the users that the administrator can manage

Steps

1. Log in as a user with a role granting you access to the User Management responsibility, select the User Management responsibility in the navigator and click the **Users** subtab.
2. Use the Search field to locate the user whose password you wish to change and then click the **Reset Password** icon next to the user.
3. Select one of the following options, provide any required information and click the **Submit** button.
   - **Generate Automatically**. No additional information is required and the system automatically generates the new password.
   - **Enter Manually**. The system prompts you to enter the password and a confirmation of the password.

The person for whom you reset the password receives and email notification stating that the password has expired and must be reset the next time the user logs in. This notification is sent by the UMX Password (UMXUPWD) workflow.

Unlocking Locked User Accounts

Oracle User Management enables administrators to unlock user accounts that have been locked due to unsuccessful attempts to log in using an incorrect password.

Prerequisites

To unlock an account, an administrator must be assigned the following:

- In the Maintain People and Users section, see Common Prerequisites, page 3-29.
- The Reset Password (UMX_OBJ_PASSWD_MGMT) permission for the users that the administrator can manage.

Steps

1. Log in as a user with a role granting access to the User Management responsibility.
2. Select the User Management responsibility in the navigator, and click the Users subtab.

3. Use the Search field to locate the user whose account you wish to unlock. The user account is locked if the Account Status column displays a padlock icon along with status "Locked".

4. Click the "Reset Password" icon next to that user and follow the steps mentioned in the section above to reset the user's password. As a result of resetting the password, the user account will be unlocked.

Assigning Roles to or Revoking Roles from Users

Oracle User Management enables administrators to assign roles to or revoke roles from the subset of users that they manage.

Prerequisites

To assign roles to or revoke roles from users, an administrator must be assigned the following:

- Common prerequisites from the Maintain People and Users section, Common Prerequisites, page 3-29.

- The appropriate administrative privileges for the role the administrator assigns or revokes. For more information, see Defining Role Administration Privileges for Roles, page 3-12.

Steps

1. Log in as a user with a role granting you access to the User Management responsibility, select the User Management responsibility in the navigator, and click on the **Users** subtab.

2. Search for the person to whom you wish to assign roles or from whom you wish to revoke roles.

3. From the search results table, navigate to the User Details page by clicking on the **Update** icon next to the person's name.

4. To assign a role to the user, click the **Assign Roles** button on the User Details page and select the desired role.

   To revoke a role from the user, you must end-date the role. If the role is an inherited role, you can only remove it by removing the role from which it originates in the role inheritance hierarchy. You can view a role's inheritance hierarchy by clicking on the **Show** hyperlink next to the role.

Additional Guidelines

The administrator can only grant or revoke roles for which he has the appropriate

privileges. If a registration process exists for the role, it will be invoked and the request will be handled by the Oracle User Management registration engine. If not, then the role is assigned directly. If the role is associated with a registration process for existing users and the registration process has a reference for capturing additional information, then the "Additional Information Required" link is rendered. The administrator must click on this link and provide any required additional information before the request is processed.

Fine Grained Access Control for Role Administration

Fine grained access (FGA) control for roles extends the delegated administration functionality by securing administrator operations for role administration. *Fine Grained Access for RBAC* (FGA for RBAC) , provides the functionality to support requirements.of the form "this administrator can run security wizards for some roles but not others". More specifically, FGA for RBAC allows a security administrator to set up a *limited administrator*, who can only perform restricted actions on a role.

The following privileges are available for administering roles:

- **Assign Role** - Allows an administrator to assign only a certain set of roles.
- **Revoke Role** - Allows an administrator to revoke only a certain set of roles.
- **Update Role** - Allows an administrator to update only a certain set of roles.
- **Manage Grants** - Allows an administrator to create grants on a set of roles.
- **Alter Hierarchy** - Allows an administrator to change the role hierarchies of only those roles upon which this privilege is given.
- **Run Security Wizard** - Allows an administrator to run security wizards on a certain set of roles.

The security administrator can define privileges for roles via the Role Administration tab on the Delegated Administration Screen.

Steps

1. The Security Administrator creates a new role, such as one called Limited Security Administrator, then enables FGA on this role by running the Delegated Administration setup wizard.
2. In the setup's Role Administration tab, the Security Administrator creates a new role administration criterion, for example HRMS Role Administration. A criterion is simply a set of roles to which a set of privileges can be assigned. An administrator role can be associated with any number of criteria.
3. The Security Administrator assigns the Assign Role and Manage Grants privilege to

this new criterion.

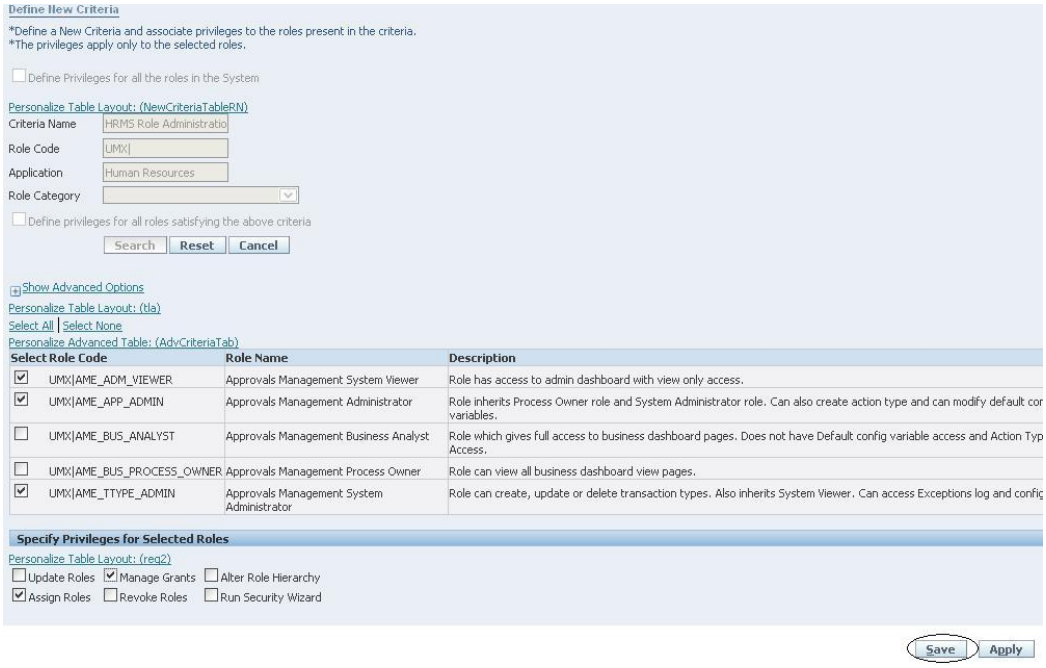
Any administrator to which the new Limited Security Administrator role is assigned will only be able to administer those roles present in the role administration criterion.

The following screenshots illustrate this process.

Create New Criteria - Starting Link

Create New Criteria - Define the Criteria

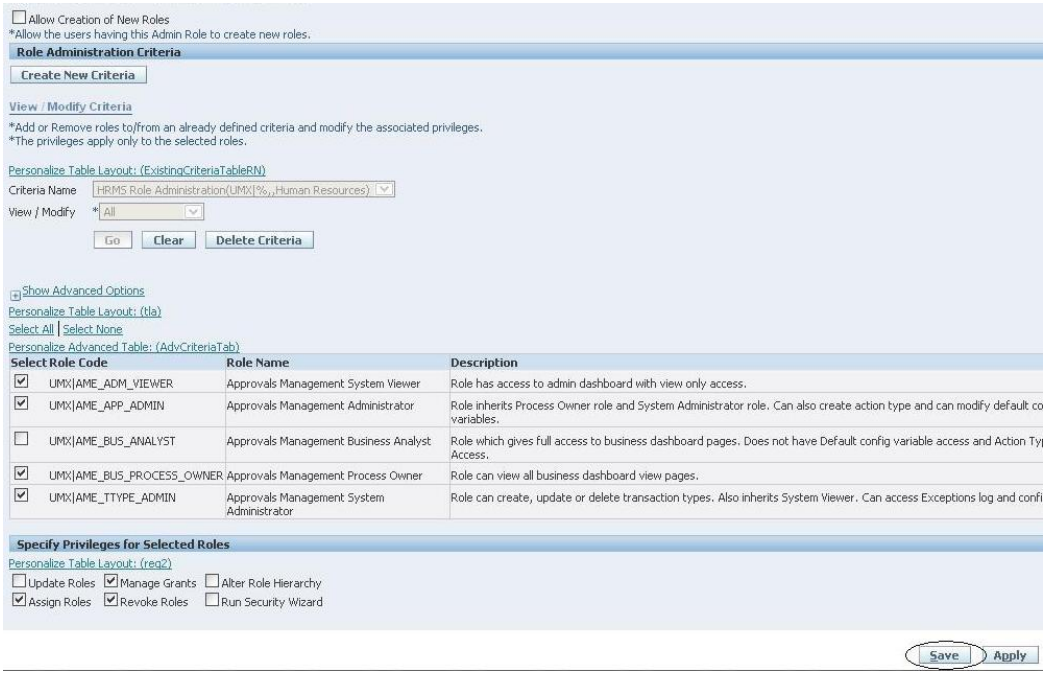
Create Criteria - Select Roles and Privileges and Save



Update Criteria - Select Criteria to be Updated

Update Criteria - Populate Roles and Privileges

Update Criteria - Modify and Save



Managing System Accounts

UMX formerly supported data security policies for users with a party_id in the TCA schema (HZ_PARTIES table). All such user operations were based on the "User

Management Person" Object (UMX_PERSON_OBJECT). As this object was based on the HZ_PARTIES table, it could only manage users linked to a person, or (to put it another way) who had a party_id in the TCA schema. Actions such as "Query Person Details", "Reset Password" , "Edit Person Details", and "Create, Inactivate, or Reactivate Account" on users were dependent on data security policies and permissions granted on the User Management Person Object (UMX_PERSON_OBJECT). Administrators therefore had to create data security policies on the User Management Person Object (UMX_PERSON_OBJECT).

This raised the question of how to administer system accounts, which lack a party_id. UMX had a permission called "Maintain System Account" Permission, which used to maintain all users who lacked party_id. Administrators with this permission could perform all operations on users who lacked a party_id. However, this did not address the issue of how to administer some sets of system accounts with restricted operations.

UMX_SYS_ACCT Object

A data security object called "User Management: system accounts" (UMX_SYS_ACCT) is now provided to support accounts that lack a party_id in the TCA Schema. This "User Management: system accounts" object is based on the database object (table) "FND_USER".

Steps to use UMX_SYS_ACCT:

1. Log on as a user who has been assigned the Security Administrator role (typically as sysadmin), select the User Management responsibility in the navigator, and click the Roles & Role Inheritance sub-tab.

2. In the role hierarchy, access the role to which you want to assign user administration privileges, and click the Update icon.

3. Click the Security Wizards button.

4. Click the Run Wizard icon for "User Management: Security Administration Setup".

5. Click the User Administration sub-tab, then click on link "Create Instance Set For Users".

6. Click the Add More Rows button.

7. In the Users field, select the set of users who can be managed by administrators to whom the role was assigned. The drop-down list contains various data security policies that pertain to the "User Management Person" and "User Management: system accounts" objects. Select the instance set that you created in Step 5.

8. In the Permissions field, select the permissions that you wish to associate with the delegated administration role.

   If you want to query a user and reset his password, select "Query and Reset

Password" permission set in the drop down

9. Click Apply or Save or to save your changes.

Managing Proxy Users

This section describes how to set up and use the Proxy User feature.

> **Note:** For more information on managing the roles of users, see the section Assigning Roles to or Revoking Roles from Users, page 3-32.

Setting Up Proxy Users

1. Log in as System Administrator and navigate to User Management > Users.
2. Query the user (delegator) that you wish to have the ability to grant proxy privileges to other users: click on the *Update* icon of the results table to navigate to the User Details page.
3. On the User Details page, click on the *Assign Role* button, and search for *Manage Proxies* role in the list of values.

   Pick this role, supply the justification, and click the *Apply* button.
4. By assigning the *Manage Proxies* role to the delegator, you make the delegator eligible to grant proxy privileges to other users to act on the delegator's behalf.

Delegating Proxy User Privileges

1. As a user with the Manage Proxies role (see previous section), log on to Oracle E-Business Suite and click on the global Preferences menu.
2. Under the Manage Proxies link, click on the *Add People* button (see Note below).
3. Select a user from the list of values, updating the start and end dates if required.
4. Click on *Apply* to save the changes.
5. Once the changes are saved, a notification will be sent to the user who has been granted the proxy privileges.

> **Note:** The permission that controls the list in the Add People LOV is UMX_OBJ_DESIGNATE_PROXY, and the object is UMX_USER_OBJECT. The out-of-the-box instance set contains all the people. The list can be modified by creating a new instance set and a

grant (and deleting the existing grant), to restrict the list of people.

Acting as a Proxy User

The proxy user mechanism is employed by users as follows:

1. If you are a user permitted to act on behalf of other users, you will see your name with the prefix *Logged in as* in the upper right-hand corner of the page. This reminds you who you are acting as.

2. To switch to another user (act as a delegate), choose the *Switch User* icon and link to access the Switch User page. These are only displayed for users who are permitted to use the Proxy User feature.

3. Click on the *Switch User* icon to switch to Proxy Mode, where you can act on behalf of the selected user.

4. The Switch User page shows an alphabetical list of people who have specified that you can act on their behalf, as a delegate.

5. After you have selected a delegator, the application will enter Proxy Mode. While in this mode, the icon and link will change from *Switch User* to *Return to Self*.

6. The user login information details reflect that you are now acting on behalf of the selected delegator.

7. While in Proxy Mode, you cannot switch directly to another proxy, but must first switch back to yourself.

8. To exit Proxy Mode, click on *Return to Self*.

Running the Proxy User Report

In Proxy Mode, *Page Access Tracking* (PAT) is automatically turned on, to audit the pages visited by the user when acting as a proxy for the delegator.

To run a report on proxy user activities, carry out the following steps:

1. Go to the Preferences > Manage Proxies function.

2. Click on *Run Proxy Report*.

3. Provide the appropriate parameters and run the report.

A concurrent program, *Page Access Tracking Data Migration*, needs to be run for the proxy to see the most recent updates in the report. Refer to Chapter 5 for details.

Registering External Organization Contacts

Oracle User Management provides a sample registration process that enables administrators to register new people for their organizations. Organizations can use the sample registration process directly or reference it as an example of how to define their own administration registration processes.

Prerequisites

To register new people, an administrator must be assigned the following:

- The common prerequisites detailed in the Maintain People and Users section, Common Prerequisites, page 3-29.
- The necessary privileges to invoke the specific administrative account creation registration processes; these are defined as part of the registration process definition.
- Organization Administration privileges for all organizations for which an administrator needs to be able to register new people.

Steps

1. Log in as a user with a role granting you access to the User Management responsibility, select the User Management responsibility in the navigator and click the **Users** subtab.
2. In the Register dropdown list, select administrative account registration process you wish to invoke, and click the **Go** button.
3. Enter the information required by the registration process as defined by the registration UI for the registration process, click the **Submit** button and then click the **OK** button in the resulting page.

Self Service Features

Implementors and administrators can verify the successful configuration of end user functions by performing the tasks described in this section.

Self-Service Registration

Oracle User Management enables users to register for access to applications without requiring assistance from administrators. To register for application access, users must provide information in the required fields and click the **Submit** button.

Oracle User Management ships with the following sample self-service registration processes:

- Employee Self-Service Registration
- Customer Self-Service Registration (external individuals)

Organizations can use these registration processes in their existing form, or can use them as references for developing their own registration processes.

Requesting Additional Application Access

Oracle User Management enables you to request additional access to the specific applications for which you are eligible. Application access is based on roles and to access an application you must be granted the appropriate role. Perform the following to view the roles you have been assigned and to request additional ones.

Steps

1. After logging into the system, click the **Preferences** link in the upper right corner, and click the **Access Requests** link in the sidebar menu. The Access Requests page displays the roles you have been assigned. Click the **Request Access** button to request one or more additional roles.
2. Most roles are organized according to role categories: roles that are not categorized appear under the Miscellaneous node. Select the role category that contains the role you want to request. If you do not see the required role, then either you are not eligible for the role or it has not been set up to for additional access requests.
3. Select the role or roles you require for additional access to the system, and click on the **Add to List** button. You can optionally remove roles from your list by clicking on the **Remove Roles** button.
4. When you have selected all your required roles, click on the **Next** button.
5. Enter a justification for your request and click on the **Next** button. You can remove any pending roles or check their status in the page that appears next.

Guidelines

Some roles may require you to provide additional information. In such cases, the system will prompt you for additional information before you can complete the process for requesting a role.

If the role being assigned would cause a separation of duties violation, the operation will flag this in the workflow attributes, and any approvers for the request will see the details.

Login Assistance

It is not uncommon for system administrators to have to reset a user's forgotten

password, or even advise a user of the account's user (login) name. This is unproductive for both the user, who cannot do any work in the meantime, and for the administrator. In addition, a user will occasionally request the password to be reset, when it is actually the user name that has been forgotten, or vice versa. This type of occurrence leads to even more time being lost.

A new feature reduces the time spent in such administrative activities by implementing a login help mechanism that is easily accessed from the E-Business Suite Login Page. A user simply clicks on the "Login Assistance" link located below the Login and Cancel buttons.

On the screen that appears, you can either:

- Go to the Forgot Password section, enter the correct user name and then click on the "Forgot Password" button. You will then be emailed details of how to reset your password.

- Go to the Forgot User Name section, enter the email address associated with the account, and click on the Forgot User Name button. The user name will then be emailed to the address specified.

For security, the relevant data is stored securely in workflow tables, and the URLs employed have both an expiration time and a single-use limitation.

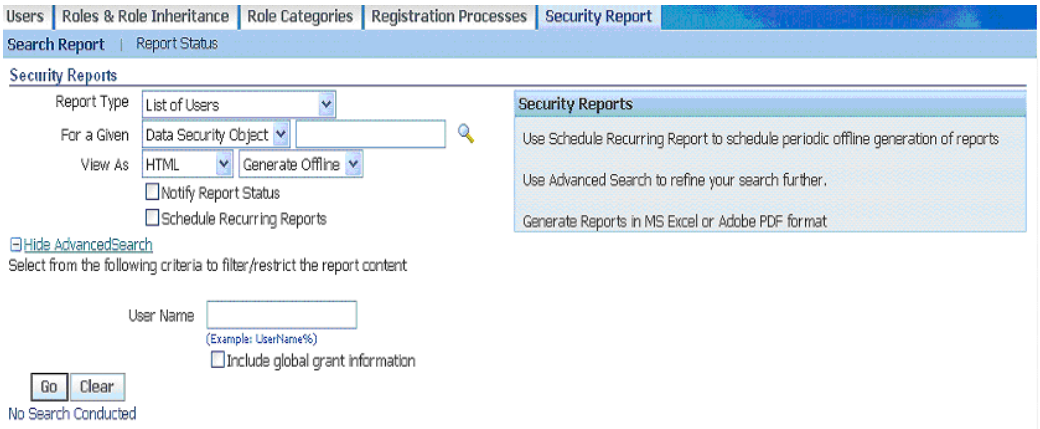
The identify verification process required in previous releases of Oracle E-Business Suite is no longer needed. Instead, a link to a secure page is sent to the email address of the user name defined in the system. From this secure page, the user can change password immediately.

Security Reports

The Security Reports feature of UMX enables a security administrator to query the security infrastructure, identifying users who have access to specified security entities and listing the type of access those security entities grant.

Home Page

From the main page of Security Reports, the security administrator can create reports on the basis of "User", "Role/Responsibility", "Function/Permission", or "Data Security Object". A different set of reports is created for each parameter.



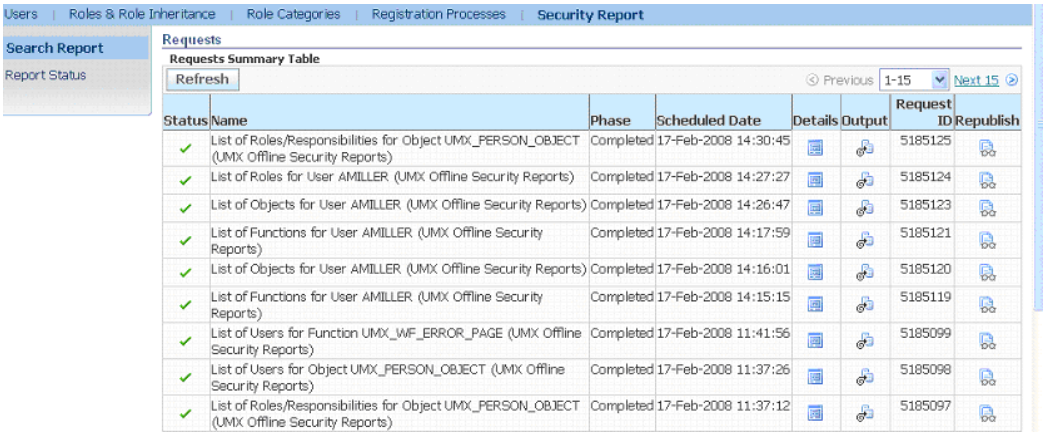
When creating reports, the security administrator can specify:

- The report required
- Whether the report is to be viewed online or offline
- The format of the report

In addition, the security administrator can:

- Schedule recurring reports (reports that are generated offline on a periodic basis)
- Filter conditions specific to the report, to help restrict the number of rows seen in the output

For example, a check of offline reports filed by a user might show:



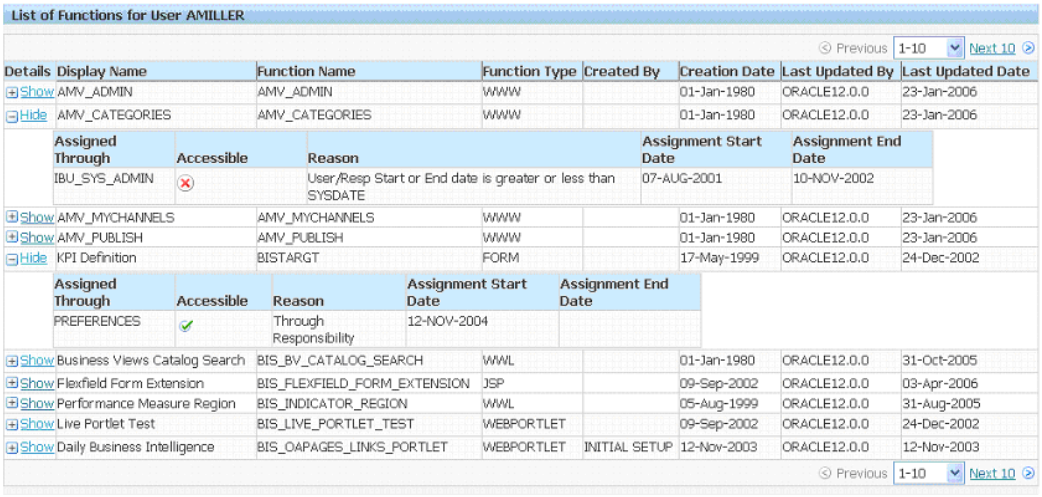
The following sections describe some the reports that can be produced.

Listing Functions for a User

This report will display assigned functions to a given user. The main record will show:

- Function display name
- Internal name
- Function type
- Who columns

For each main record there will be a detail row that will show all the paths from which this function is available to the end user, whether it is accessible from that path, and if not, the reason and the date of assignment.



List of Functions for User AMILLER

Previous 1-10 Next 10

Details	Display Name	Function Name	Function Type	Created By	Creation Date	Last Updated By	Last Updated Date
Show	AMV_ADMIN	AMV_ADMIN	WWW		01-Jan-1980	ORACLE12.0.0	23-Jan-2006
Hide	AMV_CATEGORIES	AMV_CATEGORIES	WWW		01-Jan-1980	ORACLE12.0.0	23-Jan-2006

Assigned Through	Accessible	Reason	Assignment Start Date	Assignment End Date
IBU_SYS_ADMIN	✗	User/Resp Start or End date is greater or less than SYSDATE	07-AUG-2001	10-NOV-2002

Show	AMV_MYCHANNELS	AMV_MYCHANNELS	WWW		01-Jan-1980	ORACLE12.0.0	23-Jan-2006
Show	AMV_PUBLISH	AMV_PUBLISH	WWW		01-Jan-1980	ORACLE12.0.0	23-Jan-2006
Hide	KPI Definition	BISTARGT	FORM		17-May-1999	ORACLE12.0.0	24-Dec-2002

Assigned Through	Accessible	Reason	Assignment Start Date	Assignment End Date
PREFERENCES	✓	Through Responsibility	12-NOV-2004	

Show	Business Views Catalog Search	BIS_BV_CATALOG_SEARCH	WWL		01-Jan-1980	ORACLE12.0.0	31-Oct-2005
Show	Flexfield Form Extension	BIS_FLEXFIELD_FORM_EXTENSION	JSP		09-Sep-2002	ORACLE12.0.0	03-Apr-2006
Show	Performance Measure Region	BIS_INDICATOR_REGION	WWL		05-Aug-1999	ORACLE12.0.0	31-Aug-2005
Show	Live Portlet Test	BIS_LIVE_PORTLET_TEST	WEBPORTLET		09-Sep-2002	ORACLE12.0.0	24-Dec-2002
Show	Daily Business Intelligence	BIS_OAPAGES_LINKS_PORTLET	WEBPORTLET	INITIAL SETUP	12-Nov-2003	ORACLE12.0.0	12-Nov-2003

Previous 1-10 Next 10

Filter Conditions

This report has the following filters:

- **Function Type**: Zero or one function types can be selected; only those records which have this function type will be shown.
- **Include Global Granted Functions**: This filter allows or prevents information on functions assigned from global grants being added to the report.
- **Function Name/Function Display Name**: This filter accepts a wildcard for function name, and can be used to check if a given user has this function.

Listing Data Security and Business Objects for a User

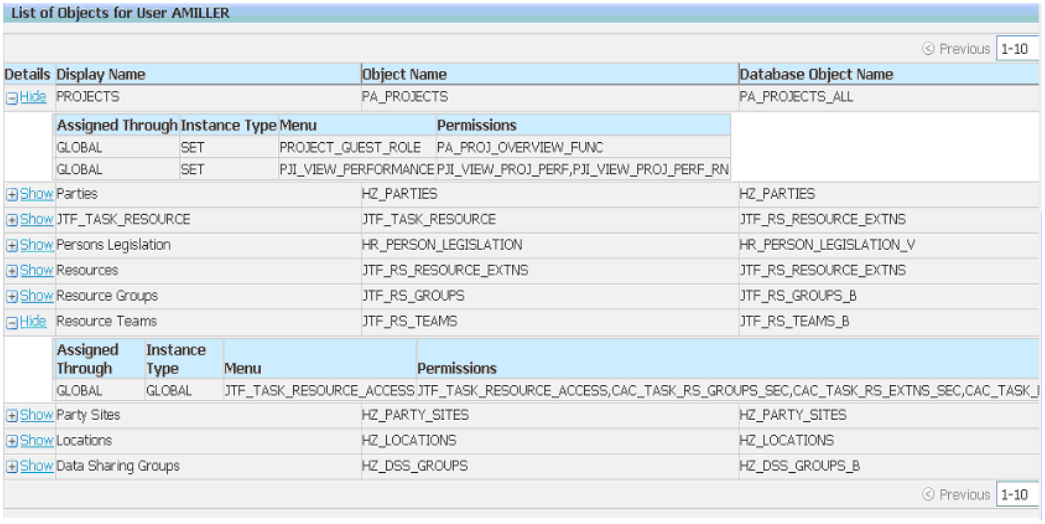
The fields listed in the main table for this report are:

- **Object Name**: The internal code for the object, and a sortable column for this table.
- **Object Display Name**: The 'user friendly' name for the object.
- **Database Object Name**: The database object with which the object is associated.

The detailed region (Show/Hide) contains the following information:

- **Instance Type**: The type of object instance to which the user has access. Valid values for this field are:
  - Set
  - Instance
  - All/Global
- **Assignment Type/Assigned Through**: This field indicates the source via which the user has an access on this object. Valid values for this field are:
  - Role Grant
  - User Grant
  - Global
  - Permission set: The permission set name through which the user has access on this object, the permissions are shown as a comma separated values.

As the same object could be assigned through multiple paths, all the paths are shown here.



List of Objects for User AMILLER

Previous 1-10

Details	Display Name	Object Name	Database Object Name
Hide	PROJECTS	PA_PROJECTS	PA_PROJECTS_ALL

Assigned Through	Instance Type	Menu	Permissions
GLOBAL	SET	PROJECT_GUEST_ROLE	PA_PROJ_OVERVIEW_FUNC
GLOBAL	SET	PJI_VIEW_PERFORMANCE	PJI_VIEW_PROJ_PERF,PJI_VIEW_PROJ_PERF_RN

Show	Parties	HZ_PARTIES	HZ_PARTIES
Show	JTF_TASK_RESOURCE	JTF_TASK_RESOURCE	JTF_RS_RESOURCE_EXTNS
Show	Persons Legislation	HR_PERSON_LEGISLATION	HR_PERSON_LEGISLATION_V
Show	Resources	JTF_RS_RESOURCE_EXTNS	JTF_RS_RESOURCE_EXTNS
Show	Resource Groups	JTF_RS_GROUPS	JTF_RS_GROUPS_B
Hide	Resource Teams	JTF_RS_TEAMS	JTF_RS_TEAMS_B

Assigned Through	Instance Type	Menu	Permissions
GLOBAL	GLOBAL	JTF_TASK_RESOURCE_ACCESS	JTF_TASK_RESOURCE_ACCESS,CAC_TASK_RS_GROUPS_SEC,CAC_TASK_RS_EXTNS_SEC,CAC_TASK_I

Show	Party Sites	HZ_PARTY_SITES	HZ_PARTY_SITES
Show	Locations	HZ_LOCATIONS	HZ_LOCATIONS
Show	Data Sharing Groups	HZ_DSS_GROUPS	HZ_DSS_GROUPS_B

Previous 1-10

Filter Conditions

This report has the following filters:

- **Database Object Name**: This filter is used to control which objects are shown in the report.

Listing Roles and Responsibilities for a User

The fields listed in the main table for this report are:

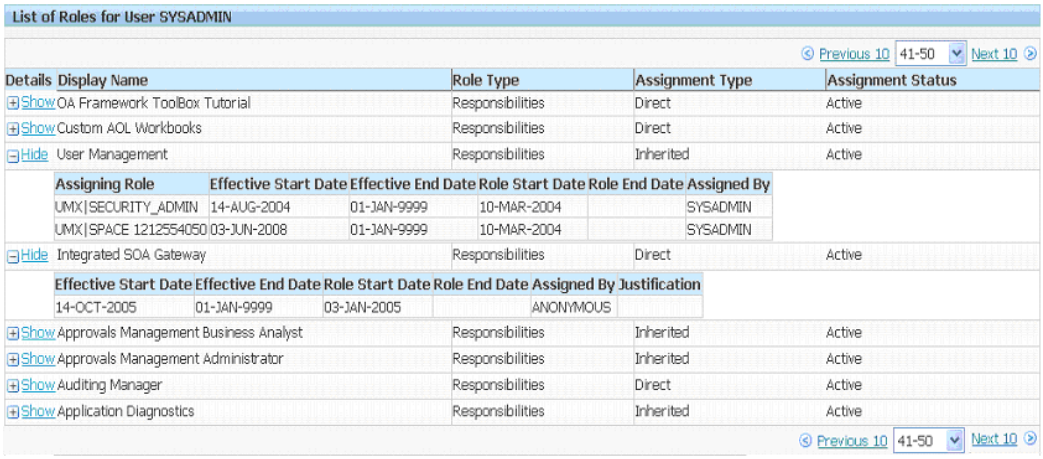
- **Role Display Name**: The 'user friendly' name for the role.
- **Role Type**: Can be a responsibility or role.
- **Assignment Status**: Indicates whether the User-Role/Responsibility Assignment is active or not.
- **Assignment Type:** This field indicates whether the role is directly assigned to the user, inherited by the user, or both. The valid values for this column are:
  - Direct
  - Indirect
  - Both

The detailed region (Show/Hide) contains the following information:

- **Dates information**: For all roles (both direct and indirect), this region contains information about:

- **Effective Start Date**: Date from which the user- role relationship is active.
- **Effective End Date**: Date on which the user-role relationship ends.
- **Role/Responsibility Start Date**
- **Role/Responsibility End Date**

- **Justification/Comments**: This field is shown only for roles whose assignment type is 'Direct/Both'. It lists any comments added by the administrator who has assigned the role or responsibility to the user.
- **Assigning Role**: In the case of indirect assignments, this column shows the parent role through which this role was assigned to the user.



List of Roles for User SYSADMIN

Previous 10 | 41-50 | Next 10

Details	Display Name	Role Type	Assignment Type	Assignment Status
Show	OA Framework ToolBox Tutorial	Responsibilities	Direct	Active
Show	Custom AOL Workbooks	Responsibilities	Direct	Active
Hide	User Management	Responsibilities	Inherited	Active

Assigning Role	Effective Start Date	Effective End Date	Role Start Date	Role End Date	Assigned By
UMX|SECURITY_ADMIN	14-AUG-2004	01-JAN-9999	10-MAR-2004		SYSADMIN
UMX|SPACE 1212554050	03-JUN-2008	01-JAN-9999	10-MAR-2004		SYSADMIN

Hide	Integrated SOA Gateway	Responsibilities	Direct	Active

Effective Start Date	Effective End Date	Role Start Date	Role End Date	Assigned By	Justification
14-OCT-2005	01-JAN-9999	03-JAN-2005		ANONYMOUS	

Show	Approvals Management Business Analyst	Responsibilities	Inherited	Active
Show	Approvals Management Administrator	Responsibilities	Inherited	Active
Show	Auditing Manager	Responsibilities	Direct	Active
Show	Application Diagnostics	Responsibilities	Inherited	Active

Previous 10 | 41-50 | Next 10

Filter Conditions

- **Role Name**: Used to control which roles and responsibilities are shown in the report. This filter accepts a wild card.
- **Assignment Status**: Controls whether the end user sees Active, Inactive, or All assignments.
- **Role Type**: Controls whether the end user wants to see Roles, Responsibilities, or All.
- **Assignment Type**: This filter controls whether the end user wants to see assignment types of Direct, Indirect, Both or, or All.

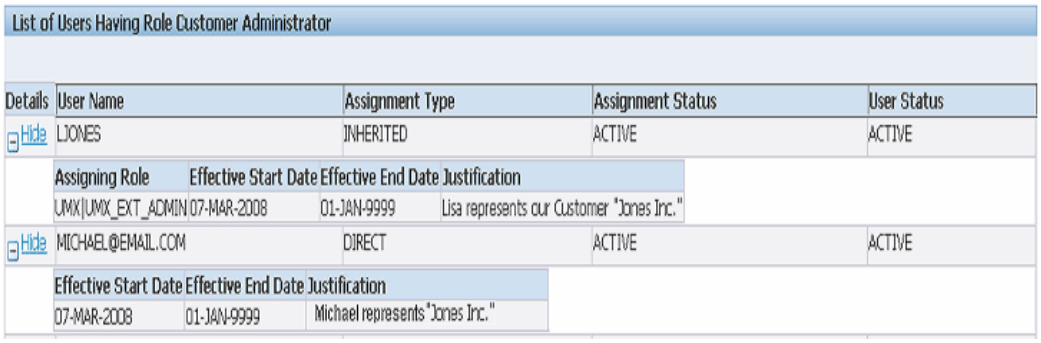
Listing Users With a Given Role

The fields listed in the main table for this report are:

- **User Name**
- **Assignment Status**: Whether the User to Role assignment is active or not.
- **User Status**: Whether the user is active or not.
- **Assignment Type**: Whether the role is inherited, directly assigned, or both. Valid values for this column are:
  - Direct
  - Indirect
  - Both

The detailed region (Show/Hide) contains the following information:

- **How**: This information is given only for the relationships that are indirectly inherited by the user.
  - **Parent Role Name**: Name of the 'Immediate Parent Role'' through which this role has been inherited by the user. If the role has been assigned to this user through different paths, all the parent roles from the various paths will be shown.
- **Justification**: Given only for the relationships that are directly assigned.
  - Justification is 'ASSIGNMENT_REASON' in WF_User_Role_Assignments.



Filter Conditions

- **Assignment Type**: Controls whether 'Direct', 'Indirect', 'Both' or 'All' types are shown.
- **User Status**: Displays report based on User Status, which can be specified as 'Active', 'Inactive', or 'All'.

- **Assignment Status**: Displays report based on User to Role Assignment Status, which can be specified as 'Active', 'Inactive', or 'All'.

- **User Name**: Displays report filtered by User Name.

Listing Functions That Can Be Accessed From a Given Role

This report displays assigned functions to a given user. All columns are sortable. The main record will show Function Display Name, Internal Name, Function Type, and Who columns.

List of Functions for Role Partner Administrator

Previous 1-10 Next 10

Function Name	Display Name	Function Type	Created By	Creation Date	Last Updated By	Last Updated Date
FND_GRANTS_SUMMARY	Grants Summary	JSP	INITIAL SETUP	28-Aug-2002	ORACLE12.0.0	19-Jul-2006
FND_MANAGE_PROXIES	Manage Proxies Page	WWW	INITIAL SETUP	02-Dec-2004	ORACLE12.1.0	20-Feb-2008
ICX_CLOSE	Close	WWW	INITIAL SETUP	02-Jul-2002	ORACLE12.1.0	20-Feb-2008
ICX_HELP	Self Service Help	WWW	SYSADMIN	01-Jan-1980	ORACLE12.1.0	20-Feb-2008
ICX_LOGOUT	Logout	WWW	SYSADMIN	01-Jan-1980	ORACLE12.1.0	20-Feb-2008
ICX_RETURN_TO_PORTAL	Return to Portal	WWW	SYSADMIN	01-Jan-1980	ORACLE12.1.0	25-Oct-2007
ICX_SSWA_USER_PREFERENCES	New General Preferences	JSP	INITIAL SETUP	22-Oct-2002	ORACLE12.1.0	20-Feb-2008
ICX_USER_PREFERENCES	General Preferences	WWW		01-Jan-1980	ORACLE12.1.0	20-Feb-2008
OAM_UALERT_PAGE	OAM User Alert Page	JSP	INITIAL SETUP	06-May-2004	ORACLE12.1.0	25-Oct-2007
UMX_B2B_REG	Registration UI - Customer / Supplier Organization Contacts	JSP	INITIAL SETUP	05-Mar-2004	ORACLE12.0.0	19-Jul-2006

Previous 1-10 Next 10

Filter Conditions

- **Function Name**: This filter accepts a wildcard for Function Name, and can be used to check if a given role has the functions in question.

- **Function Type**: Only those records with the specified function type will be shown.

Listing Objects for a Given Role

The fields listed in the main table for this report are:

- **Object Name**: The internal code for the object, and a sortable column for this table.

- **Object Display Name**: The 'user friendly' name for the object.

- **Database Object Name**: The database object with which this object is associated.

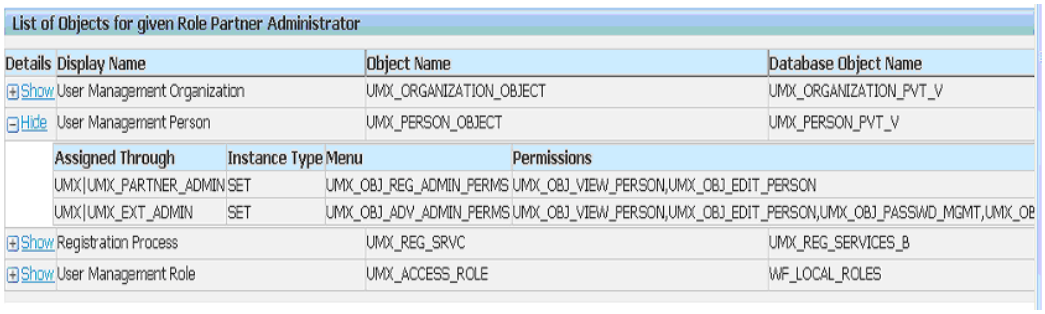
The detailed region (Show/Hide) contains the following information:

- **Instance Type**: The type of object instance to which this role gives access. Valid values for this field are:
  - Set

- Instance
- All/Global

- **Assignment Type/Assigned Through**: This field indicates the parent role through which this role grants access on this object.
- **Permission Set**: The name through which the user has access on this object's permissions, which are shown as comma-separated values.

As the same function or permission could be assigned through multiple paths, all the paths are shown here.



List of Objects for given Role Partner Administrator

Details	Display Name	Object Name	Database Object Name
Show	User Management Organization	UMX_ORGANIZATION_OBJECT	UMX_ORGANIZATION_PVT_V
Hide	User Management Person	UMX_PERSON_OBJECT	UMX_PERSON_PVT_V

Assigned Through	Instance Type	Menu	Permissions
UMX|UMX_PARTNER_ADMIN	SET	UMX_OBJ_REG_ADMIN_PERMS	UMX_OBJ_VIEW_PERSON,UMX_OBJ_EDIT_PERSON
UMX|UMX_EXT_ADMIN	SET	UMX_OBJ_ADV_ADMIN_PERMS	UMX_OBJ_VIEW_PERSON,UMX_OBJ_EDIT_PERSON,UMX_OBJ_PASSWD_MGMT,UMX_OB

Show	Registration Process	UMX_REG_SRVC	UMX_REG_SERVICES_B
Show	User Management Role	UMX_ACCESS_ROLE	WF_LOCAL_ROLES

Filter Conditions

There are no applicable filter conditions.

Listing Users for a Given Function

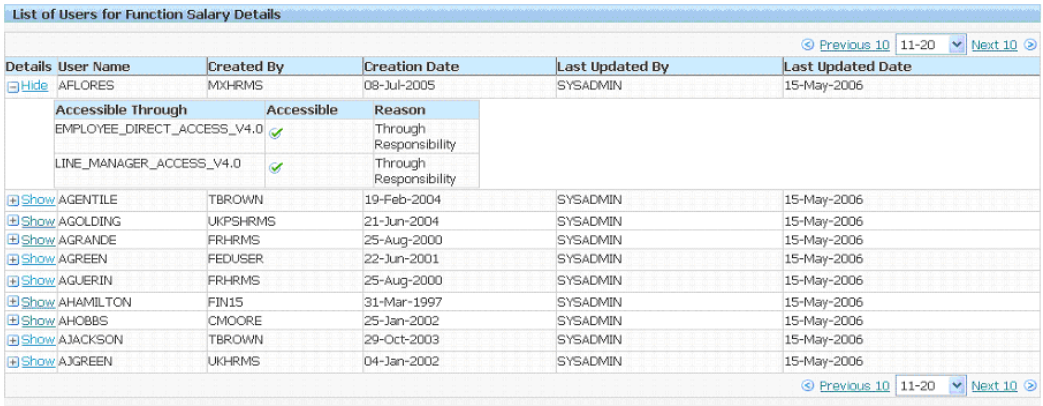
The fields listed in the main table for this report are:

- User Name
- Who Columns

The detailed region (Show/Hide) contains the following information:

- **Accessible Through**: The Child Role/Responsibility/Grant through which the function is accessible from this role.
- **Accessibility**: Whether the function is accessible through this path.
- **Reason**: The reason the function is not accessible.

As the same function or permission could be assigned through multiple paths, all the paths are shown here.


List of Users for Function Salary Details

Previous 10 11-20 Next 10

Details	User Name	Created By	Creation Date	Last Updated By	Last Updated Date
Hide	AFLORES	MXHRMS	08-Jul-2005	SYSADMIN	15-May-2006
	Accessible Through	Accessible	Reason		
	EMPLOYEE_DIRECT_ACCESS_V4.0	✓	Through Responsibility		
	LINE_MANAGER_ACCESS_V4.0	✓	Through Responsibility		
Show	AGENTILE	TBROWN	19-Feb-2004	SYSADMIN	15-May-2006
Show	AGOLDING	UKPSHRMS	21-Jun-2004	SYSADMIN	15-May-2006
Show	AGRANDE	FRHRMS	25-Aug-2000	SYSADMIN	15-May-2006
Show	AGREEN	FEDUSER	22-Jun-2001	SYSADMIN	15-May-2006
Show	AGUERIN	FRHRMS	25-Aug-2000	SYSADMIN	15-May-2006
Show	AHAMILTON	FIN15	31-Mar-1997	SYSADMIN	15-May-2006
Show	AHOBBS	CMOORE	25-Jan-2002	SYSADMIN	15-May-2006
Show	AJACKSON	TBROWN	29-Oct-2003	SYSADMIN	15-May-2006
Show	AJGREEN	UKHRMS	04-Jan-2002	SYSADMIN	15-May-2006

Previous 10 11-20 Next 10

Filter Conditions

- **User Name**: The report can be restricted on the basis of the User Name (for example, "Joe%").

Listing Roles and Responsibilities for a Given Object

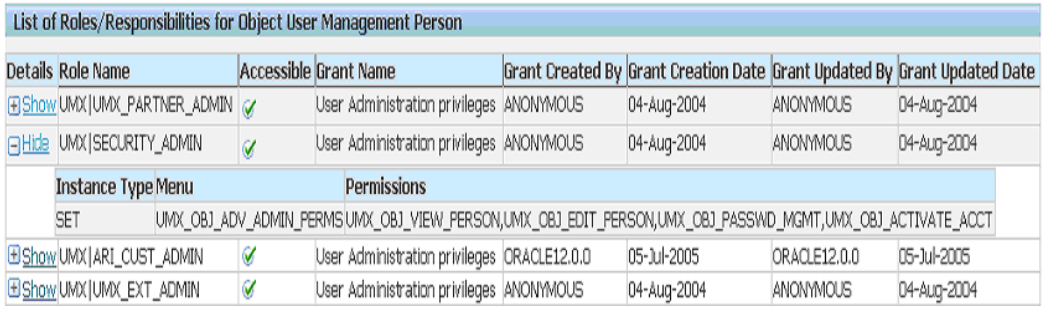
The fields listed in the main table for this report are:

- Role Display Name: User Friendly Name
- Role Type: Responsibility/Role
- Who Columns

The detailed region (Show/Hide) contains the following information:

- **Instance Type of Grant**: Can be Set, All, or Instance
- **Permission Set**: Permissions granted for this role on this object

As the same role or responsibility could be assigned through multiple paths, all the paths are shown here.


List of Roles/Responsibilities for Object User Management Person

Details	Role Name	Accessible	Grant Name	Grant Created By	Grant Creation Date	Grant Updated By	Grant Updated Date
Show	UMX|UMX_PARTNER_ADMIN	✓	User Administration privileges	ANONYMOUS	04-Aug-2004	ANONYMOUS	04-Aug-2004
Hide	UMX|SECURITY_ADMIN	✓	User Administration privileges	ANONYMOUS	04-Aug-2004	ANONYMOUS	04-Aug-2004
	Instance Type	Menu	Permissions				
	SET	UMX_OBJ_ADV_ADMIN_PERMS	UMX_OBJ_VIEW_PERSON,UMX_OBJ_EDIT_PERSON,UMX_OBJ_PASSWD_MGMT,UMX_OBJ_ACTIVATE_ACCT				
Show	UMX|ARI_CUST_ADMIN	✓	User Administration privileges	ORACLE12.0.0	05-Jul-2005	ORACLE12.0.0	05-Jul-2005
Show	UMX|UMX_EXT_ADMIN	✓	User Administration privileges	ANONYMOUS	04-Aug-2004	ANONYMOUS	04-Aug-2004

Filter Conditions

- **Role Name**: The report can be filtered on the basis of Role Name.

4

Oracle Application Object Library Security

Overview of Oracle E-Business Suite Security

As System Administrator, you define Oracle E-Business Suite users, and assign one or more responsibilities to each user.

Defining Application Users

You allow a new user to sign-on to Oracle E-Business Suite by defining an *application user*. An application user has a username and a password. You define an initial password, then the first time the application user signs on, they must enter a new (secret) password.

When you define an application user, you assign to the user one or more responsibilities.

Responsibilities Define a User's Context

A *responsibility* provides a context in which a user operates. This context can include profile option values, navigation menus, available concurrent programs, and so on.

For example, a responsibility can allow access to:

- A restricted list of windows that a user can navigate to; for example, a responsibility may allow certain Oracle Planning users to enter forecast items, but not enter master demand schedule items.
- A restricted list of functions a user can perform. For example, two responsibilities may have access to the same window, but one responsibility's window may have additional function buttons that the other responsibility's window does not have.
- Reports in a specific application; as system administrator, you can assign groups of reports to one or more responsibilities, so the responsibility a user choose determines the reports that can be submitted.

Each user has at least one or more responsibilities, and multiple users can share the same responsibility. A system administrator can assign users any of the standard responsibilities provided with Oracle E-Business Suite, or create new custom responsibilities if required.

HRMS Security

The Human Resources Management Systems (HRMS) products have an additional feature using Security Groups. For more information, see: *Customizing, Reporting, and System Administration in Oracle HRMS*.

Related Topics

Defining a Responsibility, page 4-4

Defining Request Security, page 4-6

Overview of Function Security, page 4-9

Form Functions, page 4-28

Responsibilities, page 4-19

Users Window, page 4-23

Oracle E-Business Suite User Passwords

The following are features related to passwords for end users of Oracle E-Business Suite.

Passwords can be defined in the Users Window; see: Users Window, page 4-23 for more information on setting user passwords.

Case Sensitivity in Oracle E-Business Suite User Passwords

Oracle E-Business Suite user passwords can optionally be treated as case sensitive, depending on the setting you choose for the site-level profile option *Signon Password Case*.

The two available settings are:

- **Sensitive** - Passwords are stored and compared as they are, with the password case preserved. During comparison, if the entered password does not match the decrypted version, then an error message is displayed. With Release 12, this option is the default behavior. All newly created or changed passwords are treated as case sensitive.

  Note: Users who have not changed their passwords since the installation of release 12 are not affected until they do change their

passwords.

A password expiration utility is available if the System Administrator requires that all users convert to case sensitive passwords upon the next login. This utility expires all passwords in FND_USER, including that of SYSADMIN and default Vision accounts, and can be run as a SQL Script ($FND_TOP/sql/AFCPEXPIRE.sql) or as a Concurrent Program (FNDCPEXPIRE_SQLPLUS).

- **Insensitive (or not set)** - Passwords are treated as case insensitive. In Insensitive mode, passwords are stored and compared in uppercase, similar to that in earlier releases. The entered password and the decrypted password are converted to uppercase prior to comparison.

If you want to preserve case insensitivity in passwords, i.e. retain the behavior from previous releases, ensure that Signon Password Case value is either set to 'Insensitive', or not set at all.

There are no upgrade or data migration issues with this new feature. The profile option affects only how new passwords are stored. Existing passwords are tested using the policy in effect when they were created.

Non-Reversible Hash Password Scheme

For enhanced security of passwords, you can use the FNDCPASS utility to migrate local Oracle E-Business Suite user passwords from their current encryption scheme to a non-reversible hash that makes them non-recoverable.

For specific information on FNDCPASS usage, see My Oracle Support Knowledge Document 457166.1, *FNDCPASS Utility New Feature: Enhance Security With Non-Reversible Hash Password*.

For additional information on FNDCPASS and the related AFPASSWD utility, see: Oracle E-Business Suite Schema Password Change Utilities, *Oracle E-Business Suite Setup Guide*.

Restriction on the GUEST User Password

The GUEST User password cannot include the special character "#".

Guest User Account

Credentials (username and password) for the Guest user are stored in a secure repository that was specifically designed to store sensitive data such as credentials, certificates and keys. Oracle E-Business Suite products can read Guest user information from this repository using standard APIs.

Note: Prior to Release 12.1, such items were stored in a FND profile option, GUEST_USER_PWD. This profile option did not offer the advanced security features now employed, and is no longer supported.

The only supported way to change the Guest user password is to update the context variable `s_guest_pass` and run AutoConfig, which runs the AdminAppServer utility. See My Oracle Support Knowledge Document 1380535.1, *Using AutoConfig to Manage System Configurations with Oracle E-Business Suite Release 12.2* and AdminAppServer Utility, *Oracle E-Business Suite Setup Guide*.

User Session Limits

Using the following profile options you can specify limits on user sessions.

ICX:Session Timeout

Use this profile option to enforce an inactivity time-out. If a user performs no Oracle E-Business Suite operation for a time period longer than the time-out value (specified in minutes), the user's session is disabled. The user is provided an opportunity to re-authenticate and re-enable a timed-out session. If re-authentication is successful, the session is re-enabled and no work is lost. Otherwise, Oracle E-Business Suite exits without saving pending work.

If this profile option to 0 or NULL, then user sessions will never time out due to inactivity.

ICX: Limit time

Use this profile option to specify the absolute maximum length of time (in hours) of any user session, active or inactive.

Defining a Responsibility

When you define a responsibility, you assign to it some or all of the components described below.

Menu (Required)

A menu is a hierarchical arrangement of application functions (forms). In the definition of a responsibility, the specified menu defines what is displayed in the navigator. The specified menu does not necessarily define the functions that can be accessed by the responsibility, which are granted. See: Overview of Function Security, page 4-9.

Data Group (Required)

A data group defines the mapping between Oracle E-Business Suite products and ORACLE database IDs. A data group determines which Oracle database accounts a responsibility's forms, concurrent programs, and reports connect to. See: Defining Data Groups, *Oracle E-Business Suite Setup Guide*.

Oracle Application Framework functionality does not support data groups.

For almost all cases, you should accept the default value in defining a responsibility.

Function and Menu Exclusions (Optional)

A responsibility may optionally have function and menu exclusion rules associated with it to restrict the application functionality enabled for that responsibility. See: Overview of Function Security, page 4-9.

Additional Notes About Responsibilities

Predefined Responsibilities

All Oracle E-Business Suite products are installed with predefined responsibilities. Consult the reference guide for your Oracle E-Business Suite product for the names of those predefined responsibilities.

Additionally, instances of the major components that help define a responsibility (data groups, request security groups, menus, and functions) are predefined for Oracle E-Business Suite.

Responsibilities and Request Security Groups

Note: The Request Security Groups feature is for backward compatibility only.

When a request group is assigned to a responsibility, it becomes a *request security group*.

From a standard submission form, such as the Submit Requests form, the choice of concurrent programs and request sets to run are those in the user's responsibility's request security group.

If you do not include the Submit Requests form on the menu for a responsibility, then you do not need to assign a request security group to the responsibility.

Responsibilities and Function Security

Oracle E-Business Suite architecture may aggregate several related business functions into a single form. Parts of an application's functionality may be identified as individual Oracle E-Business Suite functions, which can then be secured (i.e. included or excluded

from a responsibility).

See: Overview of Function Security, page 4-9

Defining Request Security

You can control user access to requests and request sets using request security groups or Role-Based Access Control (RBAC). Beyond this short introduction, request groups and request security groups are discussed in greater detail, as part of a broader range of topics not necessarily limited to application security, in *Oracle E-Business Suite Setup Guide*.

Using Request Security Groups

You can use request security groups to specify the reports, request sets, and concurrent programs that your users can run from a standard submission form, such as the Submit Requests form.

Define a request group using the Request Groups form. Using the Responsibilities form, you assign the request group to a responsibility. The request group is then referred to as a *request security group*. See: Request Security Groups, *Oracle E-Business Suite Setup Guide* .

You can define a request group to contain single requests, request sets, or all the requests and request sets in an application.

If you choose to include all the requests and requests sets in an application, the user has automatic access to any new requests and request sets (without owners) in the future.

A request security group can contain requests and request sets from different applications. If you want to define request security groups that own requests from different applications, refer to the discussion on Data Groups. See: Defining Data Groups, *Oracle E-Business Suite Setup Guide*.

> **Note:** A *request security group* or *request group* is not the same as a *security group*.

Individual Requests and Request Sets

Reports or concurrent programs which are not included in a request security group on an individual basis, but do belong to a request set included in a request security group, have the following privileges:

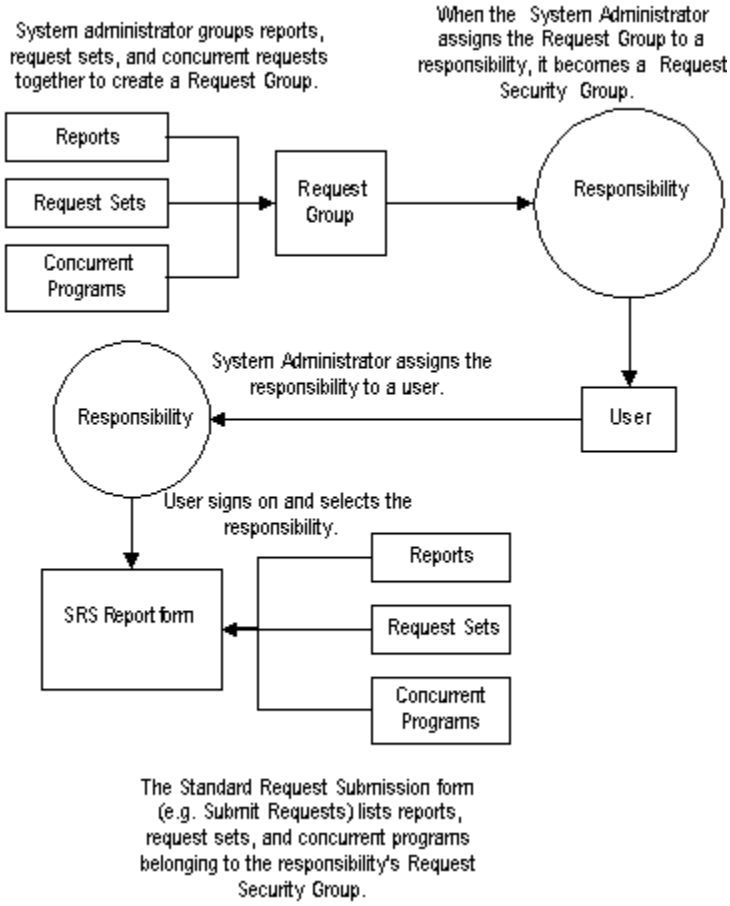
- Users can, however, run request sets that contain requests that are not in their request security group, if the request set is in their request security group.

If you assign a request set, but not the requests in the set, to a request security group, the user:

- Can edit the request set by deleting requests from it or adding other requests to it, only if the user is the assigned owner of the request set.
- Cannot edit request information in the request set definition.
- Cannot stop specific requests in the set from running.

The Request Security Groups figure below illustrates the relationship between a request security group, application user, and a responsibility.

Responsibilities, Request Groups, and Request Security Groups



Request Security using RBAC

By using RBAC, administrators have more granular control in granting submission privileges to users. In short, administrators can assign individual programs/sets, all programs/sets in a request group, programs/sets belonging to one or more applications, and so on, either to the user directly or to a role that can then be assigned to one or more users.

If applications are included in the request groups, all programs/requests sets that are created in these applications will also be automatically included. Please note that request submission applies to both programs and request sets.

See: Controlling Access to Concurrent Programs using Role-Based Access Control (RBAC), *Oracle E-Business Suite Setup Guide*.

Related Topics

Overview of Oracle E-Business Suite Security, page 4-1

Defining a Responsibility, page 4-4

Form Functions, page 4-28

Menus, page 4-33

Responsibilities, page 4-19

Users, page 4-23

Request Sets and Owners, *Oracle E-Business Suite Setup Guide*

Oracle Applications Manager Security Tests

You can manage Oracle E-Business Suite Diagnostics tests from the Oracle Applications Manager Dashboard.

The two key tests accessible from the OAM Security tab are:

- Best Practices: Database Security Tests
- Best Practices: Oracle E-Business Suite Security Tests

These tests are described further in My Oracle Support Document 403537.1, *Secure Configuration Guide for Oracle E-Business Suite Release 12*.

Overview of Security Groups in Oracle HRMS

Security groups, used exclusively by Oracle HRMS, allow data to be partitioned in a single installation. A single installation can use a particular set of configuration data, but store data for multiple clients, where the data is partitioned by security groups. A user with a responsibility assignment of one security group can only access data within that security group.

A security group represents a distinct client or business entity. Data that must be distinct for each client in an installation is partitioned by security group. All other data is shared across all security groups.

For Oracle Application Object Library, the data items that are "striped" by security groups are responsibility assignments, lookups, and concurrent programs.

Security is maintained at the level of responsibility/security group pairs. That is, users are assigned specific responsibilities within each security group. When signing on to Oracle E-Business Suite, a user, if assigned more than one responsibility, will be asked to choose a responsibility and security group pair. Partitioned data accessed through security group sensitive views will show only data assigned to the current security group.

Use the Enable Security Groups profile option to enable this feature.

Defining Security Groups

Every installation will have a single "Standard" security group seeded in. If no other security groups are created, this single group will be hidden from users when they sign on.

In the Users form, you assign a security group when you assign a responsibility.

For more information, see: *Configuring, Reporting and System Administration in Oracle HRMS.*

Overview of Function Security

Function security is the mechanism by which user access to applications functionality is controlled.

Function security can be considered as "global data security", in that it grants access to a function regardless of the current row of data.

Oracle E-Business Suite architecture aggregates several related business functions into a single form. Because all users should not have access to every business function in a form, Oracle E-Business Suite provides the ability to identify pieces of applications logic as *functions*. When part of an application's functionality is identified as a function, it can be secured (i.e., included or excluded from a responsibility).

Application developers register functions when they develop forms. A system administrator administers function security by creating responsibilities that include or exclude particular functions.

Terms

Function

A function is a part of an application's functionality that is registered under a unique name for the purpose of assigning it to, or excluding it from, a responsibility.

There are two types of function: *executable functions* (originally called form functions), and *non-executable functions* (originally called subfunctions).

Executable Function

Executable functions have the unique property that you may navigate to them using the Navigate window.

Non-Executable Function

A non-executable function is a securable subset of a form's functionality: in other words, a function executed from within a form.

A developer can write a form to test the availability of a particular non-executable function, and then take some action based on whether the non-executable function is available in the current responsibility.

Non-executable functions are frequently associated with buttons or other graphical elements on forms. For example, when a non-executable function is enabled, the corresponding button is enabled.

However, a non-executable function may be tested and executed at any time during a form's operation, and it need not have an explicit user interface impact. For example, if a non-executable function corresponds to a form procedure not associated with a graphical element, its availability is not obvious to the form's user.

Menu

A menu is a hierarchical arrangement of functions and menus of functions. Each responsibility has a menu assigned to it. Menus can map to permission sets.

Menu Entry

A menu entry is a menu component that identifies a function or a menu of functions. In some cases, both a function and a menu of functions correspond to the same menu entry. For example, both a form and its menu of subfunctions can occupy the same menu entry.

Responsibility

A responsibility defines an application user's current privileges while working with Oracle E-Business Suite. When an application user signs on, they select a responsibility that grants certain privileges, specifically:

- The functions that the user may access. Functions are determined by the menu assigned to the responsibility.
- The concurrent programs, such as reports, that the user may run.
- The application database accounts that forms, concurrent programs, and reports connect to.

Related Topics

How Function Security Works, page 4-12

Form Functions, page 4-28

Forms and Subfunctions , page 4-11

Functions, Menus, and the Navigate Window, page 4-11

Overview of Oracle E-Business Suite Security, page 4-1

Implementing Function Security, page 4-13

Executable functions vs. Non-executable functions

An executable function, as a whole, including all of its program logic, is always designated as a function. Subsets of a form's program logic can optionally be designated as subfunctions if there is a need to secure those subsets.

For example, suppose that an executable function such as a form contains three windows. The entire form is designated as a function that can be secured (included or excluded from a responsibility). Each of the form's three windows can be also be designated as non-executable functions, which means they can be individually secured. Thus, while different responsibilities may include this form, certain of the form's windows may not be accessible from each of those responsibilities, depending on how function security rules are applied.

Related Topics

Overview of Function Security, page 4-9

Functions, Menus, and the Navigate Window, page 4-11

How Function Security Works, page 4-12

Functions, Menus, and the Navigate Window

Executable functions are selected using the Navigate window. The arrangement of form names in the Navigate window is defined by the menu structure assigned to the current responsibility.

The following types of menu entries are not displayed by the Navigate window:

- Non-executable functions
- Menus without Entries
- Menu Entries without a Prompt

If none of the entries on a menu are displayed by the Navigate window, the menu itself is not displayed.

Menu Entries with a Submenu and Functions

If a menu entry has both a submenu and a function defined on the same line, then the behavior depends on whether or not the function is executable. If it is executable, then the submenu on the same line is treated as content to be rendered by the function. The submenu will not appear on a navigation tree, but will be available in function security tests (FND_FUNCTION.TEST calls). If the function is not executable, then it is treated as a "tag" for enforcing exclusion rules, and the submenu on the same line is displayed in the navigation tree.

A function is considered executable if it can be executed directly from the current running user interface. For example, an Oracle E-Business Suite form using Oracle Forms is an executable function from within Oracle Forms, but not within the Self Service applications.

How Function Security Works

Registering Functions

- Developers can require parts of their Oracle Forms code to look up a unique *function name*, and then take some action based on whether the function is available in the current responsibility. Function names are unique.

- Developers can register functions. They can also register parameters that pass values to a function. For example, a form may support data entry only when a function parameter is passed to it.

  Warning: In general, you should not modify names, parameters, or other material features of predefined functions for Oracle E-Business Suite products. The few exceptions are documented in the relevant manuals or product notes.

Excluding Functions

Each Oracle E-Business Suite product is delivered with one or more predefined menu hierarchies. System Administrators can assign a predefined menu hierarchy to a responsibility. To tailor a responsibility, System Administrators exclude functions or menus of functions from that responsibility using exclusion rules.

Note: The ability to exclude functions is to be used for backward compatibility only. Menu exclusions do not apply to grants.

Available Functions for a User

Functions are available to a user through responsibilities (as well as grants).

When a user first selects or changes their responsibility, a list of functions obtained from the responsibility's menu structure is cached in memory.

Functions a System Administrator has excluded from the current responsibility are marked as unavailable.

Executable functions in the function hierarchy (i.e. menu hierarchy) are displayed in the Navigate window. Available non-executable functions are accessed by working with the application's forms.

Related Topics

Overview of Function Security, page 4-9

Overview of Data Security, page 4-15

Forms and Subfunctions , page 4-11

Overview of Oracle E-Business Suite Security, page 4-1

Form Functions, page 4-28

Implementing Function Security

Securing Functions Using New Menus

Use the Menus form to define menus pointing to functions that you want to make available to a user.

- Use forms and their associated menus of non-executable functions to define new menus.

The new menu can be then granted to a user.

Defining a New Menu Structure

When defining a new menu structure:

- Create a logical, hierarchical listing of functions. This allows for easy exclusion of functions when customizing the menu structure for different responsibilities.
- Create a logical, hierarchical menu that guides users to their application forms.

Tasks for Defining a Custom Menu Structure

- Determine the application functionality required for different job responsibilities.

- Identify predefined menus, forms, and form subfunctions to use as entries when defining a new menu. Understand predefined menus by printing Menu Reports using the Submit Requests window.

  Tip: To simplify your work, use predefined menus for your menu entries. You can exclude individual functions after a menu structure is assigned to a responsibility.

- Plan your menu structure. Sketch out your menu designs.
- Define the lowest-level menus first. A menu must be defined before it can be selected as an entry on another menu.
- Assign menus and functions to higher-level menus.
- Assign menus and functions to a top-level menu (root menu).
- Document your menu structure by printing a Menu Report.

  Warning: Always start with a blank Menus form (blank screen). See Notes About Defining Menus, below.

Notes About Defining Menus

Define Menus for Fast and Easy Keyboard Use

- Design menu prompts with unique first letters, so typing the first letter automatically selects the form or menu.
- Design the sequence of menu prompts with the most frequently used functions first (i.e. lower sequence numbers).

Menu Compilation

The Compile Security (FNDSCMPI) concurrent program is used to compile menus so that the system can more quickly check if a particular function is available to a particular responsibility/menu.

You should compile your menus after you make changes to your menu data. A request for this concurrent program is automatically submitted after you make changes using the Menus form.

Related Topics

Menus Window, page 4-33

Compile Security Concurrent Program, page 4-57

Preserving Custom Menus Across Upgrades

Preserve custom menus during upgrades of Oracle E-Business Suite by using unique names for your custom menus. For example, you can start the menu's name with the application short name of a custom application. Define a custom application named *Custom General Ledger*, whose application short name is XXCGL. Define your custom menu names to start with XXCGL, for example, XXCGL_MY_MENU.

Remember that the Oracle E-Business Suite standard menus may be overwritten with upgrade versions. Therefore, if you attached your custom menu as a submenu to one of the preseeded Oracle E-Business Suite menus, recreate the attachment to it following an upgrade. An alternative is to attach a standard Oracle E-Business Suite menu as a submenu to your custom menu; the link from your custom menu to the standard menu should survive the upgrade.

Related Topics

Overview of Oracle E-Business Suite Security, page 4-1

Overview of Function Security, page 4-9

Implementing Function Security, page 4-13

Form Functions, page 4-28

Function Security Reports, page 4-57

Overview of Data Security

Data Security allows administrators to control user access to specific data, as well as what functions users can apply to that data.

Function security can be considered "global" data security, in that access to a function is granted regardless of the data.

Concepts and Definitions

Objects

Data Security uses the concept of an Object to define the data records that are secured.

Object

Data security permissions are managed on objects. Business entities such as Projects and Users are examples of objects. Only a securable business-level concept should be registered as an object.

An object definition includes the business name of the object and identifies the main

table and primary key columns used to access the object.

Object Instance

An object instance is a specific example of an object, such as Project Number 123 or User JDOE. An object instance generally corresponds to a row in the database. An instance is identified by a set of one or more primary key values as defined by the object.

In addition, "All Rows" for an object indicates all data rows of the object.

Object Instance Set

An object instance set is a group of related object instances within an object. A set is specified as a predicate on the keys or attributes of an object, expressed as a SQL "WHERE clause". All instances that satisfy the predicate are considered members of the object instance set. For example:

```
STATUS = 'ACTIVE'
```

could determine a set of object instances with the "Active" status.

The specific instances in the set can vary over time as object instance attributes change, or as new object instances are created.

An example is:

```
OWNER = FND_GLOBAL.USER_ID
```

The predicate can also be parameterized, so that the logic can define instance sets as a function of one or more input parameters. An example is:

```
COLOR = :PARAM1
```

Object instance sets are also called "data instance sets".

Users and Groups

Users and groups are Oracle Workflow roles. See the Oracle Workflow documentation for more information on roles.

Privileges given to users and groups determine their access to secured objects.

The data security system allows you to assign privileges to groups of users instead of assigning privileges to each user individually.

Users

Users are individuals who have access to software applications at a particular enterprise.

A user must have a unique name and should map one-to-one with an individual human or system. "Group" accounts are not correct uses of the user entity.

Groups

Users can belong to Groups. The grouping can come from position or organization relationships modeled in applications such as Oracle Human Resources. Alternatively, ad-hoc groups can be created explicitly for security purposes. A group is sometimes referred to as a role.

Functions and Permissions

A function or a permission is the smallest unit of securable product functionality. You can register function definitions with the security system to represent actions that can be performed on an object or on the system in general. Granting a function to a set of users gives them permission to perform that function, and so a function may also be referred to as a permission.

There are two broad categories of functions and permissions:

- An *executable function/permission* can be invoked from a generic navigator user interface. An executable function definition must contain all information necessary to launch the function; often this includes the form name or URL plus parameters.

- An *abstract function/permission* does not refer to a specific piece of code, but represents permission to perform a higher-level business action. The code that implements an abstract function calls the function security system to test whether the abstract function is granted. The system only allows the action if the abstract function is granted.

Examples of these are a particular JSP page (executable) and View Person (abstract).

Functions and permissions can either be at the system level or be sensitive to a data context.

Navigation Menus and Permission Sets

Functions and permissions are grouped into related sets so that administration of these functions can be performed in higher-level business terms.

Functions and permissions are bundled into named sets, which can be defined for two purposes: as navigation menus and/or permission sets. Each set can also contain other sets.

Menus are defined for navigation purposes and group UI pages into functional areas. Users access menus by selecting responsibilities. Each menu item maps to a permission which optionally may be granted to the user as part of the menu/responsibility assignment. Menu items that are not granted as part of the menu/responsibility assignment will not be rendered unless the user is granted the permission separately.

Permission sets are granted to users or roles independently of menus/responsibilities. Permission sets are granted to users in order to enable menu items and other operations (functions) that should not be available to all users assigned a given

menu/responsibility. Permission sets are granted to users or roles through permission assignments (grants).

Grants

A *grant* authorizes a particular role to perform a specified action or actions (set of functions) on a specified object instance (or object instance set).

Note that where you are creating a data security policy for an object by creating a grant, you need to include that object in your grant definition. Other than in this specific type of case, you do not need to specify an object in your definition.

Security Context

Security context refers to the context of the data in which the user is working. For example, data context could be the organization or responsibility with which the user is logged in.

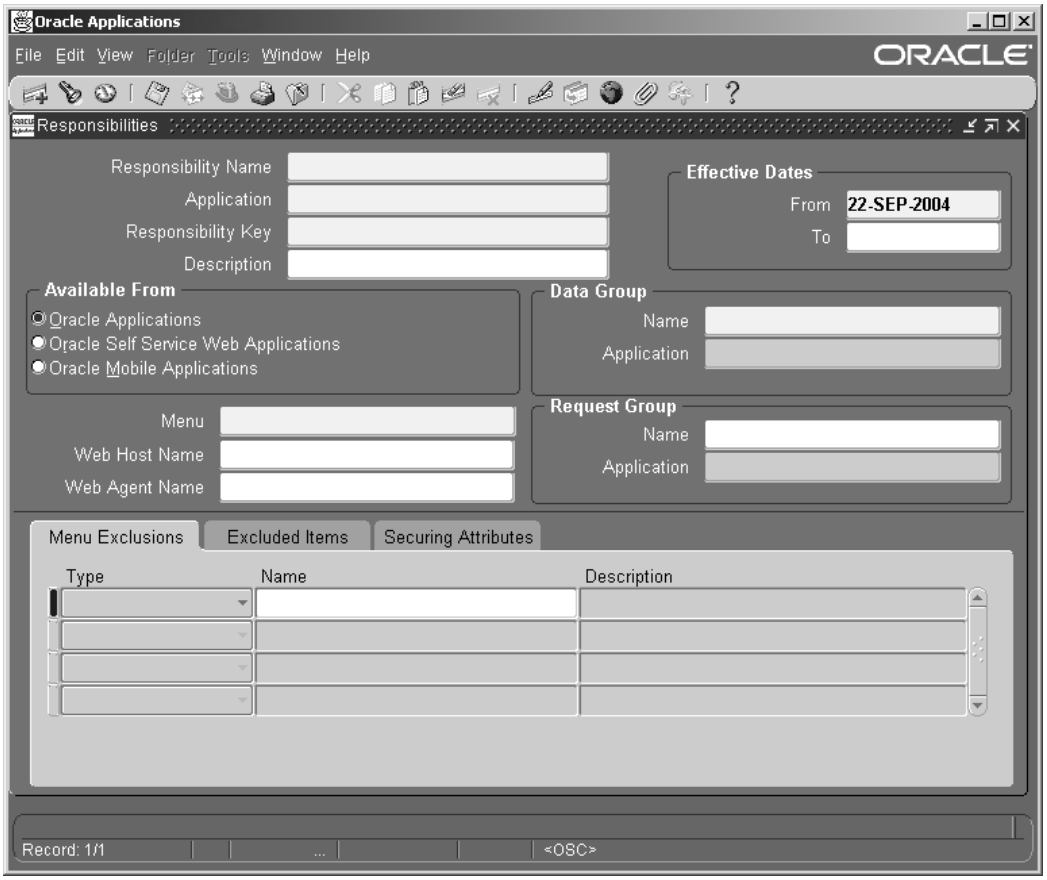
Implementation of Data Security

Implement data security by granting access to a set of functions (either a navigation menu or a permission set) to a user or group of users.

Data security policies can reflect access to:

- A specific instance (row) identified by a primary key value
- All instances (rows) of an object
- An instance set defined by a SQL predicate (WHERE clause)

Responsibilities Window



Use this window to define a responsibility. Each application user is assigned at least one responsibility.

A responsibility determines whether the user accesses Oracle E-Business Suite or Oracle Mobile Applications; which applications functions a user can use; which reports and concurrent programs the user can run; and which data those reports and concurrent programs can access.

Responsibilities cannot be deleted. To prevent a responsibility from being used, set the Effective Date's To field to a past date and restart Oracle E-Business Suite.

See: Overview of Function Security, page 4-9

Before defining your responsibility, do the following:

- Use the Data Groups window to list the ORACLE username your responsibility's concurrent programs reference on an application-by-application basis.
- Use the Request Groups window to define the Request Group you wish to make available with this responsibility.

- Use the Menus window to view the predefined Menu you can assign to this responsibility.

Responsibilities Block

An application name and a responsibility name uniquely identify a responsibility.

Responsibility Name

If you have multiple responsibilities, a pop-up window includes this name after you sign on.

Application

The owning application for the responsibility.

This application name does not prevent the user of this responsibility from accessing other applications' forms and functions if you define the menu to access other applications.

Responsibility Key

This is the internal key for the responsibility that is used by loader programs, (concurrent programs that load messages, user profiles, user profile values, and other information into Oracle E-Business Suite tables). The responsibility key is unique per application.

Avoid using the following characters in the responsibility keys: !, ", ;, [,], (,), {, }, %, |, <, >.

Effective Dates (From/To)

Enter the start/end dates on which the responsibility becomes active/inactive. The default value for the start date is the current date. If you do not enter an end date, the responsibility is valid indefinitely.

You cannot delete a responsibility, because its information helps to provide an audit trail. You can deactivate a responsibility at any time by setting the end date to the current date. If you wish to reactivate the responsibility later, either change the end date to a date after the current date, or clear the end date.

Available From

This is the navigator from which the responsibility will be available (Oracle E-Business Suite forms navigator, mobile navigator).

A responsibility may be associated with only one Oracle E-Business Suite system.

Data Group

Note: Data groups are used for backward compatibility only. Oracle Application Framework does not support the data groups feature.

Name/Application

The data group defines the pairing of application and ORACLE username.

Select the application whose ORACLE username forms connect to when you choose this responsibility. The ORACLE username determines the database tables and table privileges accessible by your responsibility. Transaction managers can only process requests from responsibilities assigned the same data group as the transaction manager.

Menu

The menu whose name you enter must already be defined with Oracle E-Business Suite.
See: Menus, page 4-33.

Request Group - Name/Application

Specify a request security group to associate the responsibility to a set of requests, request sets, or concurrent programs that users logged in with this responsibility can run from the Submit Requests window. Note that such users can also access requests from a Submit Requests window you customize with a request group code through menu parameters

Note: The Request Security Groups feature is provided for backward compatibility.

New responsibilities should be created in accordance with Role-Based Access Control and should not have a default request security group.

See:

Overview of Oracle E-Business Suite Security, page 4-1

Customizing the Submit Requests Window Using Codes, *Oracle E-Business Suite Setup Guide*

Request Groups, *Oracle E-Business Suite Setup Guide*

Menu Exclusions Block

Note: Menu exclusions should be used for backward compatibility

only.

Define function and menu exclusion rules to restrict the application functionality accessible to a responsibility.

Type

Select either Function or Menu as the type of exclusion rule to apply against this responsibility.

- When you exclude a function from a responsibility, all occurrences of that function throughout the responsibility's menu structure are excluded.
- When you exclude a menu, all of its menu entries, that is, all the functions and menus of functions that it selects, are excluded.

Name

Select the name of the function or menu you wish to exclude from this responsibility. The function or menu you specify must already be defined in Oracle E-Business Suite.

HTML-Based Applications Security

Oracle HTML-based applications use columns, rows and values in database tables to define what information users can access. Table columns represent attributes that can be assigned to a responsibility as Securing Attributes or Excluded Attributes. These attributes are defined in the Web Application Dictionary.

Excluded Items

Use the List of Values to select valid attributes. You can assign any number of Excluded Attributes to a responsibility.

Securing Attributes

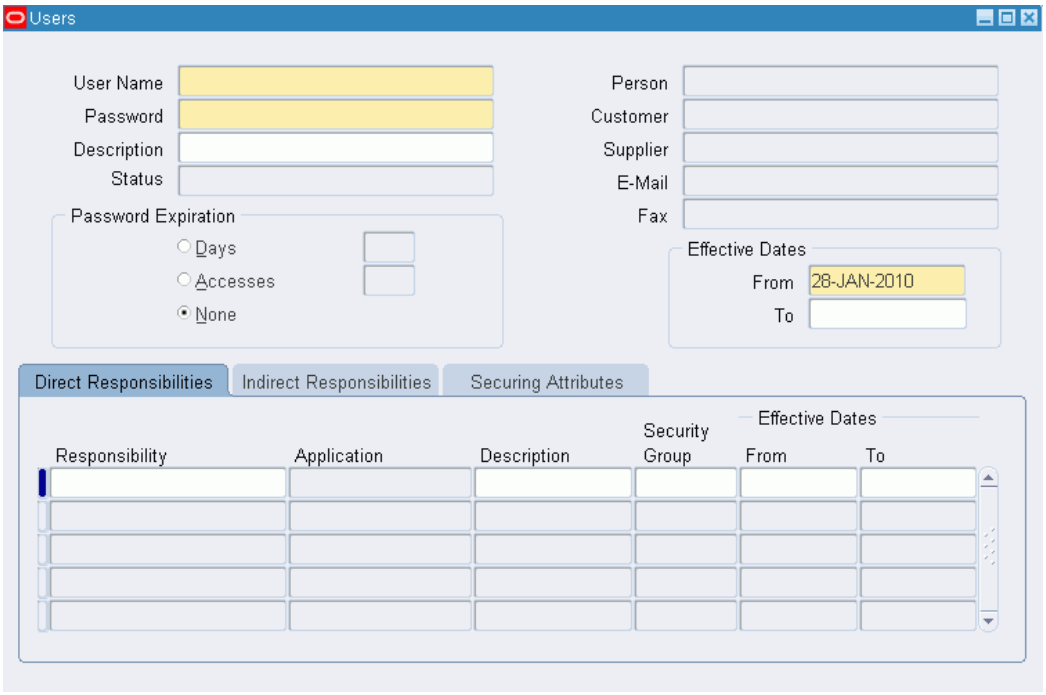
Use the List of Values to select valid attributes. You can assign any number of securing attributes to the responsibility.

Security Groups Window

This form is for HRMS security only.

For more information on setting up system administration for the HRMS products, see: *Customizing, Reporting, and System Administration in Oracle HRMS.*

Users Window



Use this window to define an Oracle E-Business Suite user. This user is an authorized user of Oracle E-Business Suite, and is uniquely identified by a username.

Once defined, a new Oracle E-Business Suite user can sign on to Oracle E-Business Suite and access data through Oracle E-Business Suite windows.

> **Note:** If you have upgraded from a previous release of Oracle E-Business Suite, ensure that you have run the Party Merge concurrent program to update your user data. If you have not run this program, you may receive errors in querying your user data.
>
> For more information, see the Oracle Trading Community Architecture documentation.

Users Block

Enter these fields for the user.

User Name

An application user enters this username to sign on to Oracle E-Business Suite.

The username should only contain characters allowed by Oracle Single Sign-On.

Tip: We recommend that you define meaningful usernames, such as the employee's first initial followed by their last name. Or, for a group account, you can define the application username so as to indicate the purpose or nature of the group account.

Password

Enter the initial password of an application user. An application user enters this password along with his username to sign on to Oracle E-Business Suite.

- A password must be at least five (5) characters and can be up to thirty (30) characters.
- All characters are allowed except control characters, which are non-printable. Oracle encourages the use of non-alphanumeric characters because they add complexity, making passwords harder to guess.

This window does not display the password you enter. After you enter a password, you must re-enter it to ensure you did not make a typing error.

If the application user already exists and the two entries do not match, the original password is not changed and an error message is displayed.

If you are defining a new application user and the two entries do not match, you are required to enter the password again. For a new user, you cannot navigate to the next field until the two entries match.

The first time an application user signs on, he must change his password. If a user forgets his password, you can reassign a new password in this field.

As System Administrator, you can set an initial password or change an existing password, but you cannot access the user's chosen password.

You can set the minimum length of Oracle E-Business Suite user passwords using the profile option Signon Password Length. If this profile option is left unset, the minimum length defaults to 5.

You can set the minimum number of days that a user must wait before being allowed to reuse a password with the Signon Password No Reuse profile option.

You can use the profile option Signon Password Hard to Guess to set rules for choosing passwords to ensure that they will be "hard to guess." A password is considered hard-to-guess if it follows these rules:

- The password contains at least one letter and at least one number.
- The password does not contain the username.
- The password does not contain repeating characters.

The Signon Password Failure Limit profile option determines the maximum number of

login attempts before the user's account is disabled.

For information on case sensitivity in passwords, see: Case Sensitivity in Oracle E-Business Suite User Passwords, page 4-2.

Status

The Status field indicates the status of the user account. This field is display-only and values are generated by the system. This field is similar to Status in Oracle User Management for managing user accounts.

Possible statuses of a user account are:

- Unassigned - This status is used for the moment of creating a new user in the form, before committing the transaction. Since a user ID hasn't been assigned yet at that moment, the record status is Unassigned.

- Pending - This user account exists but cannot be used yet. For example, a user account with "Effective Dates" that are in the future would have a Pending status.

- Locked - This user account is locked. For example, if a user has unsuccessfully tried to log in over the maximum number of tries allowed (per the profile option "Signon Password FailureLimit"), then the user account becomes locked.

- Active - The status for a user account is Active if both of the following conditions are true:
  - The start date is not NULL and is before or equal to the current date
  - The end date is NULL or is after the current date

- Inactive - This user has an inactive account. For example, a user account with "Effective Dates" that are in the past would have an Inactive status.

Person, Customer, and Supplier

Use these fields to enter the name of an employee (person), customer, or supplier contact. Enter the last name and first name, separated by a comma, of the employee, customer, or supplier who is using this application username and password. Use the List of Values to select a valid name.

For more information on using these fields, see the Oracle Trading Community Architecture documentation.

Email/Fax

Enter the email address and/or fax number for this user.

Password Expiration

- Days - Enter the maximum number of days between password changes. A pop-up window prompts an application user to change his password after the maximum number of days you specify has elapsed.

- Accesses - Enter the maximum allowed number of sign-ons to Oracle E-Business Suite allowed between password changes. A pop-up window prompts an application user to change his password after the maximum number of accesses you specify has elapsed.

  Tip: We recommend that you requireall application users to make regular password changes. This reduces the likelihood of unauthorized access to Oracle E-Business Suite.

Effective Dates (From/To)

The user cannot sign on to Oracle E-Business Suite before the start date or after the end date. The default for the start date is the current date. If you do not enter an end date, the username is valid indefinitely.

You cannot delete an application user from Oracle E-Business Suite because this information helps to provide an audit trail. You can deactivate an Oracle E-Business Suite user at any time by setting the End Date to the current date.

If you wish to reactivate a user, change the End Date to a date after the current date, or clear the End Date field.

Direct Responsibilities

Direct responsibilities are responsibilities assigned to the user directly.

Responsibility

Select the name of a responsibility you wish to assign to this application user. A responsibility is uniquely identified by application name and responsibility name.

Security Group

This field is for HRMS security only. See: *Customizing, Reporting, and System Administration in Oracle HRMS*.

This field is enabled only if the profile Enable Security Groups is enabled.

From/To

You cannot delete a responsibility because this information helps to provide an audit trail. You can deactivate a user's responsibility at any time by setting the End Date to

the current date.

If you wish to reactivate the responsibility for the user, change the End Date to a date after the current date, or clear the End Date.

Indirect Responsibilities

Indirect responsibilities are used with Oracle User Management only. A user may "inherit" an indirect responsibility through membership of a group to which the responsibility has been assigned.

This block is read-only.

Securing Attributes

Securing attributes are used by some Oracle HTML-based applications to allow rows (records) of data to be visible to specified users or responsibilities based on the specific data (attribute values) contained in the row.

You may assign one or more values for any of the securing attributes assigned to the user. If a securing attribute is assigned to both a responsibility and to a user, but the user does not have a value for that securing attribute, no information is returned for that attribute.

For example, to allow a user in the ADMIN responsibility to see rows containing a CUSTOMER_ID value of 1000, assign the securing attribute of CUSTOMER_ID to the ADMIN responsibility. Then give the user a security attribute CUSTOMER_ID value of 1000.

When the user logs into the Admin responsibility, the only customer data they have access to has a CUSTOMER_ID value of 1000.

Attribute

Select an attribute you want used to determine which records this user can access. You can select from any of the attributes assigned to the user's responsibility.

Value

Enter the value for the attribute you want used to determine which records this user can access.

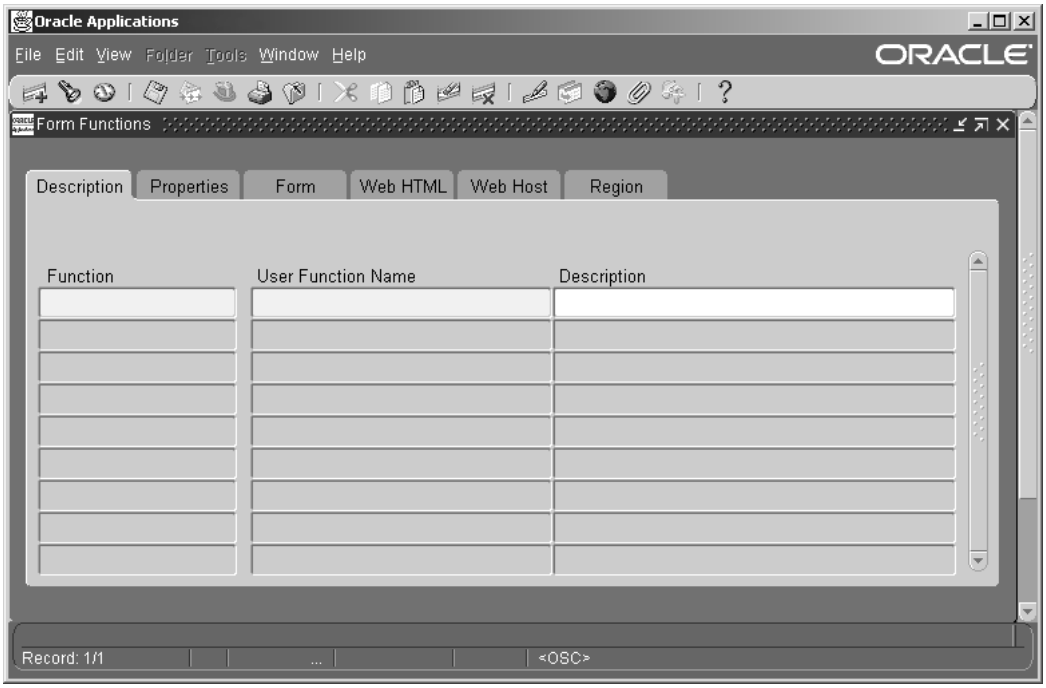
Related Topics

Defining a Responsibility, page 4-4

Overview of Function Security, page 4-9

Responsibilities, page 4-19

Form Functions Window



Used to define new functions. A function is a part of an application's functionality that is registered under a unique name for the purpose of assigning it to, or excluding it from, a responsibility.

Description

Fields include:

Function

Users do not see this unique function name. However, you may use this name when calling your function programmatically. You should follow the naming conventions for functions.

User Function Name

Enter a unique name that describes your function. You see this name when assigning functions to menus. This name appears in the Top Ten List of the Navigator window.

Properties

Fields include:

Type

A function's type describes its use. A function's type is passed back when a developer tests the availability of a function. The developer can write code that takes an action based on the function's type.

Standard function types include the following:

ADFX	External ADF Application. Used for linking an external Application Developer Framework (ADF) 11*g* application deployed on an Oracle Application Server 11*g* container from the Oracle E-Business Suite home page.
DBPORTLET	Database provider portlet.
FORM	Oracle E-Business Suite form functions are registered with a type of FORM.
JSP	Functions used for some products in the Oracle Self-Service Web Applications. These are typically JSP functions.
REST	REST service.
SERVLET	Servlet functions used for some products in the Oracle Self-Service Web Applications.
SUBFUNCTION	Subfunctions are added to menus (without prompts) to provide security functionality for forms or other functions.
WEBPORTLET	Web provider portlet.
WWK	Functions used for some products in the Oracle Self-Service Web Applications. These are typically PL/SQL functions that open a new window.
WWR or WWL	Functions used for some products in the Oracle Self-Service Web Applications.
WWJ	OA Framework JSP portlet.
WWW	Functions used for some products in the Oracle Self-Service Web Applications. These are typically PL/SQL functions.

For information on functions used by Oracle Application Framework, see the *Oracle Application Framework Developer's Guide*, available from My Oracle Support Knowledge Document 1315485.1.

Maintenance Mode Support

This field should not be used. Maintenance Mode is disabled in an online patching-enabled environment.

Context Dependence

In general, the context dependence determines the required context for the function to work properly. The context dependence controls whether the user must choose a specified context (if not already in that context) before executing the function.

For example, some functions are controlled by profile options that affect what the user can perform within the current context. Types of context dependence are:

- **Responsibility** - The function is controlled by the user's responsibility (RESP_ID/RESP_APPL_ID (includes ORG_ID)).
- **Organization** - The function is controlled by the user's organization (ORG_ID).
- **Security Group** - The function is controlled by the user's security group (service bureau mode).
- **None** - There is no dependence on the user's session context.

Form

Fields include the following:

Form/Application

If you are defining a form function, select the name and application of your form.

Parameters

Enter the parameters you wish to pass to your function. Separate parameters with a space.

For an executable (form) function:

- If you specify the parameter QUERY_ONLY=YES, the form opens in query-only mode. Oracle Application Object Library removes this parameter from the list of form parameters before opening the form in query-only mode.
- You can also specify a different form name to use when searching for help for a form in the appropriate help file. The syntax to use is:

  HELP_TARGET = "*alternative_form_name*"

  Your form name overrides the name of the form. See: Help Targets in Oracle E-Business Suite, *Oracle E-Business Suite Setup Guide*.

For a concurrent program submitted through the Standard Request Submission form, the following syntax may be used:

TITLE="*appl_short_name*:*message_name*"

where *appl_shortname*:*message_name* is the name of a Message Dictionary message. See: Customizing the Submit Requests Window using Codes, *Oracle E-Business Suite Setup Guide*.

> **Warning:** In general, system administrators should not modify parameters passed to predefined functions for Oracle E-Business Suite products. The few exceptions are documented in the relevant manuals or product notes.

Web HTML

The fields in the Web HTML and Web Host are only required if your function will be accessed from Oracle Application Framework. You do not need to enter any of these fields for functions based on Oracle Developer forms.

HTML Call

The last section of your function URL is the HTML Call. The HTML Call is used to activate your function. The function may be either a static web page or a procedure.

The syntax for this field depends on the function type.

For functions used with Mobile Application Server, enter the full name of your java class file, including <package name>.<class name>. The class name and package name are case sensitive. Mobile Application Server will try to load this class from the classpath as it is. For example, 'oracle.apps.mwa.demo.hello.HelloWorld'.

Web Host

The fields in the Web HTML and Web Host are optional and only enabled for some types of functions. These fields apply only to Oracle Application Framework functions.

Host Name

The URL (universal resource locator) or address required for your function consists of three sections: the Host Name, Agent Name, and the HTML Call. The Host name is the IP address or alias of the machine where the Web server is running.

Agent Name

The second section of your function URL is the Oracle Web Agent. The Oracle Web Agent determines which database is used when running your function. Defaults to the last agent used.

Icon

Enter the name of the icon used for this function.

Secured

Secured is only required when your function is accessed by Oracle Workflow. Checking Secured enables recipients of a workflow email notification to respond using email.

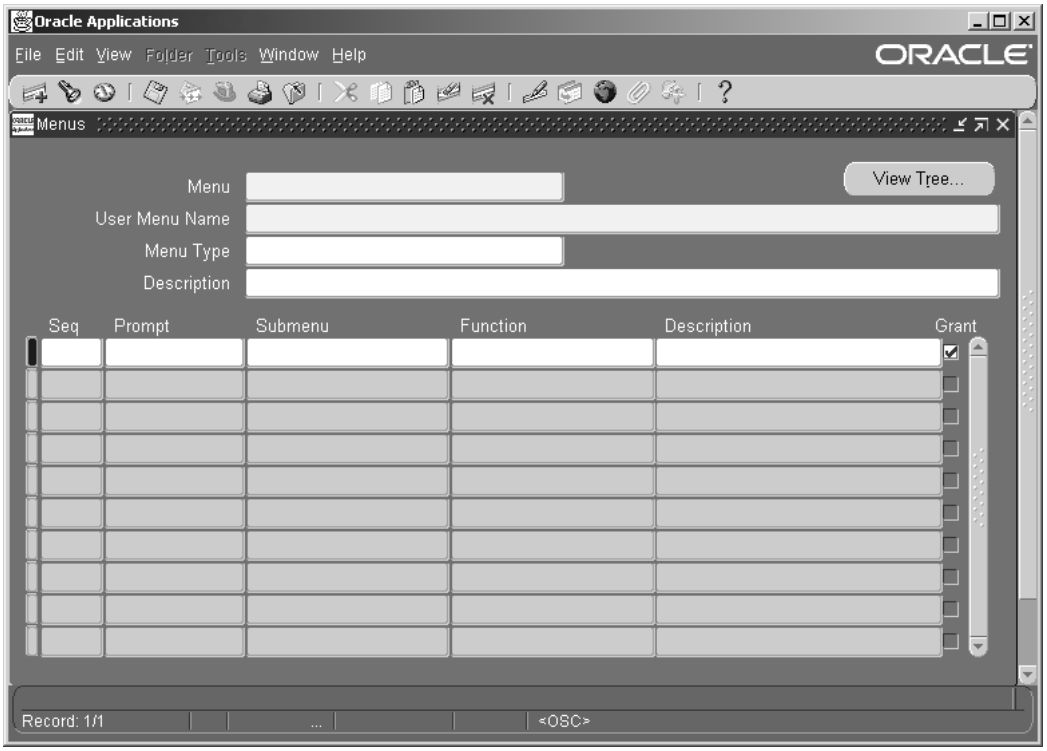
Encrypt Parameters

Checking Encrypt Parameters adds a layer of security to your function to ensure that a user cannot access your function by altering the URL in their browser window. You must define Encrypt Parameters when you define your function to take advantage of this feature.

Region

The fields on this page are for future use.

Menus Window



Used to define a new menu or modify an existing menu.

A menu is a hierarchical arrangement of functions and menus of functions. Each responsibility has a menu assigned to it.

You can build a custom menu for that responsibility using predefined forms. However, we recommend that you do not disassociate a form from its developer-defined menus.

After you save your changes in this form, a request is submitted to compile the menu data.

See:

Overview of Function Security, page 4-9

Implementing Function Security, page 4-13

Before you define your menu, perform the following:

- Register your application using adsplice. See: AD Splicer, *Oracle E-Business Suite Setup Guide* and My Oracle Support Knowledge Document 1577707.1, "Creating a Custom Application in Oracle E-Business Suite Release 12.2."

- Register any forms you wish to access from your menu with Oracle Application Object Library using the Forms window.

- Define any functions you intend to call from your menu.
- Define any menus that you intend to call from your menu. Define the lowest-level submenus first. A submenu must be defined before it can be called by another menu.

  Tip: By calling submenus from your menu, you can group related windows together under a single heading on your menu. You can reuse your menu on other menus.

Menus Block

Menu entries detail the options available from your menu.

Menu

Choose a name that describes the purpose of the menu. Users do not see this menu name.

Note: Once the menu is saved, this menu name cannot be updated.

View Tree...

Once you have defined a menu, you can see its hierarchical structure using the "View Tree..." button. See: Menu Viewer, page 4-36.

User Menu Name

You use the user menu name when a responsibility calls a menu or when one menu calls another.

Menu Type

Specify a menu type to describe the purpose of your menu. Options include:

- Standard - for menus that would be used in the Navigator form
- Tab - for menus used in self service applications tabs
- Security - for menus that are used to aggregate functions for data security or specific function security purposes, but would not be used in the Navigator form

In addition, see the section on Oracle Application Framework menu types, page 4-50.

Menu Entries Block

Fields include the following:

Sequence

Enter a sequence number to specify where a menu entry appears relative to other menu entries in a menu. The default value for this field is the next whole sequence number.

> **Important:** You can only use integers as sequence numbers.

A menu entry with a lower sequence number appears before a menu entry with a higher sequence number.

You cannot replace a menu entry sequence number with another sequence number that already exists. If you want to add menu entries to a menu entry sequence, carefully renumber your menu entries to a sequence range well outside the sequence range you want, ensuring that you do not use existing sequence numbers. If you want to renumber an entry, then delete the entire row and save your work; and then insert a new row with the desired sequence number and same prompt and submenu/function as the previous one.

Navigator Prompt

Enter a user-friendly, intuitive prompt your menu displays for this menu entry. You see this menu prompt in the hierarchy list of the Navigator window.

> **Tip:** Enter menu prompts that have unique first letters so that power users can type the first letter of the menu prompt to choose a menu entry.

Submenu

Call another menu and allow your user to select menu entries from that menu.

Function

Call a function you wish to include in the menu. A form function (form) appears in the Navigate window and allows access to that form. Other non-form functions (subfunctions) allow access to a particular subset of form functionality from this menu.

Description

Descriptions appear in a field at the top of the Navigate window when a menu entry is highlighted.

Grant

The Grant check box should usually be checked. Checking this box indicates that this menu entry is automatically enabled for the user. If this is not checked then the menu entry must be enabled using additional data security rules.

For more information on grants, see Overview of Data Security, page 4-15 and Grants, page 4-44.

Menu Viewer

The Menu Viewer is a read-only window that provides a hierarchical view of the submenus and functions of a menu, and also lists properties of the menus and functions.

You can launch the viewer from the Menus form by clicking on the "View Tree..." button. The viewer will appear for the menu specified in the Menus form.

> **Note:** When you are creating or editing a new menu, your changes must be committed to the database before you will be able to see them in the Menu Viewer.

Functionality

The Menu Viewer consists of two panes, one showing the menu tree and the other the node properties.

Menu Tree

To view the menu tree, click on the plus (+) sign next to the menu. You will see a hierarchical tree with a number of nodes. Each node represents a function or submenu of your main menu.

> **Note:** The menu tree displays the user menu name for the main menu, and displays the prompts from the Menus form for submenus and functions. If no prompt has been specified, then no label will appear for the node.

To print a menu tree, choose **Print** from the File menu.

Node Properties

To view properties of a particular menu or function, highlight the node in the menu tree. The node properties will appear in the Properties pane. You can create a separate Properties page for a node by clicking the "push pin" button at the top of the Properties pane.

The entry's sequence number, prompt, and description are shown.

View Options

The View menu provides options on how the viewer displays your menu.

You can specify whether the Node Properties pane, the toolbar, or the status bar are displayed. You can also choose the display style in which you view your menu tree.

Display Styles

There are three styles for viewing your menu tree. You can select one from the View menu or from the buttons on the toolbar.

Vertical	Menu entries are displayed vertically, similar to how they appear in the Navigator window when you log on to Oracle E-Business Suite.
Interleaved	Menu entries are displayed horizontally and vertically.
Org-Chart	Menu entries are displayed horizontally as in an organizational chart.

Edit Menu

From the Edit menu you can bring up a Properties window for the node you have highlighted in the menu tree.

> **Note:** You can view the properties for your menu or function here, but you cannot edit them.

You can view and edit your Preferences for the Menu Viewer. You can choose colors for your menu tree pane as well as the text font and size.

Objects

Use these pages to find, create, and edit data objects. You define objects to be secured in the Data Security system.

Objects can be tables or views. An object must be queryable in SQL, and the combination of primary key columns specified must be a unique key.

In these pages, objects are described with the following

- The **Name** is the name that appears in the Object Instance Set and Grants pages. This name should be user-friendly.
- The **Code** is the internal name of the object.

- The **Application Name** is the owning application.
- The **Database Object Name** is the name of the underlying database object.

Related Topics

Overview of Data Security, page 4-15

Find Objects

Use this page to find an existing object.

Simple Search

Name

The display name of the object.

Code

The object name.

Application Name

The object's owning application.

Database Object Name

The database object name.

Advanced Search

Use the Advanced Search screen to find data that meet a set of criteria. With the Advanced Search screen, you can enter in special conditions based on the given fields, and the search results will consist of all data that match the conditions.

For example, for a specified application, you can search for all objects whose name begins with a letter before "P". (Note: all uppercase letters precede all lowercase letters for this type of search).

Search Results

The search results are shown in a table with the following columns:

- Name - click on the object name to view details on the object.
- Code

- Application Name
- Database Object
- Description
- Last Update

To update an object, click on the icon under the Update column.

Update Object

Use this page to update the fields listed below for an object. You cannot change the internal Object Name of an existing object.

Display Name

Enter a user-friendly name for the object.

Application Name

The owning application for the object. This application owns the database table on which the object is based.

Database Object Name

Typically this is a table in the database.

Description

Enter a description for the object.

Create Object

Use this page to create a new object. Enter the following information:

Name

Enter a user-friendly name for the object.

Code

Enter a code that will be used as an internal name for the object. This name cannot include spaces and can include underscores and hyphens. You cannot update the object name after the object is created and saved.

Application Name

The owning application for the object. This application owns the database table on

which the object is based.

Database Object

Typically this is a table in the database.

Description

Enter a description for the object.

Object Column Details

Enter in information on the primary key for the object (*n* below indicates an integer between 1 and 5). The primary key is used to identify rows (object instances) for inclusion in object instance sets.

PK*n* Column Name

The primary key column name.

PK*n* Column Type

The datatype for the column.

Object Detail

This page provides the following information for an object:

- Object Name
- Display Name
- Application
- Database Object Name
- Description

Columns

You can also view details on columns that comprise the primary key (*n* below indicates an integer between 1 and 5):

- PK*n* Column Name
- PK*n* Column Type

Instances of an object can be grouped together into an object instance set. For example, you may want to create a group of projects or a group of items. To create and manage

objects instance sets, click on the "Manage Object Instance Sets" button.

Click on the "Return to Object Search" link to go back to the main Objects page.

Delete Object

Confirm the deletion of an object from this page. Review the information shown, and click the "Delete" button.

Related Topics

Object Details, page 4-40

Object Instance Sets

After you create an object you can create a set of instances of the object. For example, you could define the object "User" corresponding to the User table. Each row in the User table becomes an instance of the User object. Users in the sales organization could then be grouped into an Object Instance Set named "Sales Organization".

Object Instance Sets are described by the following:

- The **Object Instance Set Name** is its internal name. This name must not contain any spaces and can include underscores.
- The **Display Name** is a user-friendly name for the object that appears in the Grants pages.
- The **Predicate** is the WHERE clause used to define the object instances in the set. It must be a valid SQL predicate for the database object.

Manage Object Instance Set

Use this page to manage existing object instance sets or create new ones.

The following object information is displayed:

- Object Name
- Display Name
- Application
- Database Object Name
- Description

Existing Object Instance Sets

- Instance Set Name - click on the Instance Set Name to view details
- Display Name
- Description

To update an object, click on the icon under the Details column to open up the Update Object page.

To delete a row, click on the icon under the Delete icon, or select the object and click the Delete button.

To return to the main Objects page, click on the "Return to Object Search" link.

Related Topics

Objects, page 4-37

Create Object Instance Set

The containing object's Name, Display Name, Application ID, Database Object Name, and Description are shown.

Enter the following for the Object Instance Set:

Code

Enter a name that will be used internally for the object instance set. This name cannot include spaces and can include underscores and hyphens. The Object Instance Set Name cannot be updated once the object instance set has been created and saved.

Name

Enter a user-friendly, descriptive name to appear in the Grants pages.

Description

Enter a description for the object instance set.

Predicate

This predicate determines which object instances are included in the set. Do not include "WHERE" in your entry, but only the body of the WHERE clause.

Update Object Instance Set

The containing object's Name, Display Name, Application ID, Database Object Name,

and Description are shown.

> **Note:** The Object Instance Set Name cannot be updated after the object instance set has been created and saved.

Display Name

Enter a user-friendly, descriptive name to appear in the Grants pages.

Description

Enter a description for the object instance set.

Predicate

This predicate determines which object instances are included in the set. Do not include "WHERE" in your entry, but only the body of the WHERE clause.

Delete Object Instance Set

Confirm the deletion of an object from this page. Review the information shown, and click the "Delete" button.

Related Topics

Object Instance Set Details, page 4-43

Object Instance Set Details

Details of an object instance set are shown on this page.

The containing object's Name, Display Name, Application ID, Database Object Name, and Description are shown.

The following is shown for the object instance set:

- Code
- Name
- Description
- Predicate

Use the "Return to Manage Object Instance Sets" to return to the main page.

Related Topics

Object Instance Sets, page 4-41

Grants

The HTML-based pages for maintaining Grants can be accessed from the Functional Administrator responsibility. For more information on this responsibility, see:Overview of Functional Administrator and Functional Developer Responsibilities, *Oracle E-Business Suite Setup Guide*.

Search Grants

Use this page to search for grants.

You can search using the following criteria:

- Name
- Grantee Type - Select from one of the following:
  - All Users - The grant applies to all users.
  - Group of Users - The grant applies to a group of users.
  - Specific User - The grant applies to a single user.

    If you select Group of Users or Specific User, you will be prompted to specify the group or the user.
- Set - The Navigation Menu or Permission Set included in the grant.
- Object Type - A grant can apply to either all objects or only a specific object. Under Object Type, specify if your search should include only grants that apply to all objects ("All Objects"), only grants that apply to a specific object ("Specific Object"), or both ("Any").

  If you select Specific Object, you will be prompted to specify the object.
- Effective Dates.

Create Grant

Use these pages to create a grant. Grants are used to manage user access to product functionality. In these pages you give access to functions to specified users.

Related Topics

Overview of Data Security, page 4-15

Define Grant

In this page you specify basic information for the grant.

To define a grant:

1. Enter a name and description for your grant.
2. Enter effective dates for your grant.
3. Enter the security context information.

   The security context defines the circumstances in which the grant is active.

   For Grantee, you can select a single user, a role, or global (all users and roles).
4. For Operating Unit, specify an operating unit if you want your grant to apply to a specific one.
5. For Responsibility, specify a responsibility if you want your grant to apply to a specific one.
6. Enter the Data Security information if you are creating a data security policy for an object. The grant applies to the object you specify.

   If you are not creating a data security policy, you will skip the next step.

   > **Note:** You cannot change a data security policy once it has been saved. You can delete it or provide an end date to a data security policy.

Select Object Data Context

If you specified that your grant applies to a single object, you add context for that object in this page.

Choose one of the following:

- Global (All Rows) - Indicates that the set of functions is being granted for all rows of the object (for a function security grant).
- Instance - Indicates that the set of functions are being granted for a single row, specified by value(s) for the primary key.
- Instance Set - Indicates that the set of functions are being granted for a set of rows which is specified by an instance set predicate.

Define Object Parameters and Select Set

If you selected either an object instance or an instance set earlier, you can further customize the resulting set by additional information for the data context.

Additionally, you can select either a permission set or a navigation menu that can additionally specify how the grant will be applied in the security context.

For an instance set:

1. In the Predicate region, the predicate that defines the instance set is shown. In the Instance Set Details region, specify the values for the parameters to be used in the predicate above.
2. Select the permission set or navigation menu set that defines the grantee's access.

For an instance:

1. In the Instance Details region, specify information identifying the instance.
2. Select the permission set or navigation menu set that defines the grantee's access.

Review and Finish

Use this page to review the definition of your grant. Click **Finish** to save your work.

Update Grant

Use this page to update the definition of your grant.

View Grant

Use this page to view details for a grant, including:

- Security Context
- Object information, if applicable
- Set information

You can update or delete a grant from this page.

Functions

Use these pages to define new functions. A function is a part of an application's functionality that is registered under a unique name for the purpose of assigning it to, or excluding it from, a responsibility.

You can search for functions from the main page.

Function Types

When you define a function, you assign it one of the following types:

- External ADF Application - Used for linking an external Application Developer Framework (ADF) 11*g* application deployed on an Oracle Application Server 11*g* container from the Oracle E-Business Suite home page.
- Database Provider Portlet
- Form - an Oracle Forms form function.
- JSP Interoperable with OA
- SSWA JSP function
- Mobile Application - A function used in an Oracle mobile application.
- Process
- REST service - Used for REST services. For more information on REST services and other Oracle Application Framework functions, see the *Oracle Application Framework Developer's Guide*, available from My Oracle Support Knowledge Document 1315485.1.
- SSWA servlet function
- Web Provider portlet
- SSWA PL/SQL function that opens a new window (kiosk mode)
- Plug
- Generic Plug
- SSWA PL/SQL function

Related Topics

Form Functions Window, page 4-28

Search

Using Simple Search, You can search for functions using the following criteria:

- Name

- Code
- Type

Advanced Search

Using Advanced Search, you can be more flexible with your criteria, as well as search on the description field.

Create Function

Use these pages to create a function.

1. Specify a name for the function.
2. Specify a code for the function. The code is the internal name for the function. Once the function has been saved, the code cannot be updated.
3. Specify a type for the function.
4. For context dependence, specify 'None' or Responsibility.
5. If you are defining a form function, select the name and application of your form. If the function applies to a specific object, select the object name and specify parameters.

> **Note:** The Maintenance Mode Support field is not used. Maintenance Mode is disabled in an online patching-enabled environment.

Update Function

Use this page to update an existing function. Note that you cannot update the code for an existing function.

To update a function:

1. Specify a name for the function.
2. If this function applies to a specific object, specify the object.
3. Specify a type for the function.
4. For context dependence, specify 'None' or Responsibility.

> **Note:** The Maintenance Mode Support field is not used. Maintenance Mode is disabled in an online patching-enabled environment.

To update function details:

1. If this is a form function, select the name and application of your form.
2. If the function applies to a specific object, you can update the object name and specify parameters.

In updating menus,

- You can remove the function from menus containing it using the Menus subtab.
- You can also update menu prompts and descriptions for the function here.

Duplicate Function

Use this page to duplicate an existing function.

Note that you must enter a unique code for the new function you are creating.

To duplicate a function:

1. Specify a name for the function.
2. Specify a code for the function. The code is the internal name for the function. Once the function has been saved, the code cannot be updated.
3. Specify a type for the function.
4. For context dependence, specify 'None' or Responsibility.
5. If you are defining a form function, select the name and application of your form. If the function applies to a specific object, select the object name and specify parameters.

> **Note:** The Maintenance Mode Support field is not used. Maintenance Mode is disabled in an online patching-enabled environment.

View Function

Use this page to view details on an existing function.

You can update and duplicate a function from this page. If the function is not on a menu, you can also delete the function.

Delete Function

Use this page to delete a function.

Navigation Menus

Define a new menu or modify an existing menu.

A menu is a hierarchical arrangement of functions and menus of functions. Each responsibility has a menu assigned to it.

You can build a custom menu for that responsibility using predefined forms. However, we recommend that you do not disassociate a form from its developer-defined menus.

Before creating a menu, perform the following:

- Register your application using adsplice. See: AD Splicer, *Oracle E-Business Suite Setup Guide* and My Oracle Support Knowledge Document 1577707.1, "Creating a Custom Application in Oracle E-Business Suite Release 12.2."

- Define any menus that you intend to call from your menu. Define the lowest-level submenus first. A submenu must be defined before it can be called by another menu.

  Tip: By calling submenus from your menu, you can group related windows together under a single heading on your menu. You can reuse your menu on other menus.

Terms

Terms used in defining menus include:

- Name - The display name for the menu

- Code - The internal name for the menu

- Type - The purpose of the menu
  - Permission Set - For menus that are used to aggregate functions for data security or specific function security purposes, but would not be used in the Navigator form.
  - Standard - For menus used in the Navigator form
  - App Pref Menu Container - For preferences
  - Global Menu - For providing access to tasks and content that are applicable to the entire application
  - HTML Side Navigator Menu

- HTML SideBar
- HTML SideList
- HTML Sub Tab - A tab-like control for switching content or action views in the page's content area. Sub tabs can be used with a horizontal navigation element, with a tab and horizontal navigation elements, or with a side navigation
- HTML Tab
- Homepage

If you are creating a menu to be used with Oracle Application Framework, see the *Oracle Application Framework Developer's Guide*, available from My Oracle Support Knowledge Document 1315485.1.

Search for Menus

Enter any of the following criteria for the menu:

- Name
- Code
- Type

Create Navigation Menu

Use this page to create a navigation menu.

1. Choose a user-friendly name that describes the purpose of the menu.
2. Enter a code for the menu. Choose an internal name that indicates the purpose of the menu. Users do not see this menu code.
3. Optionally specify a menu type and description to describe the purpose of your menu.

Add your information for your menu entries using the Menu Builder.

1. Enter a prompt for your menu entry.

   Enter a user-friendly, intuitive prompt your menu displays for this menu entry. You see this menu prompt in the hierarchy list of the Forms Navigator window.

   Tip: Enter menu prompts that have unique first letters so that power users can type the first letter of the menu prompt to choose a

menu entry.

2. If this menu entry is a menu itself (a submenu), enter in the menu name.

   You can call another menu and allow your user to select menu entries from that menu.

3. If this menu entry is a function, enter in the function name.

   Call a function you wish to include in the menu.

4. Specify the function type.

5. Apply your changes.

If you want to reorder the menu entries, click the **Reorder** button.

Menu Manager

Once you have your menu defined, you can update its list of entries in the Menu Manager tab.

Hierarchy of Children

The Hierarchy of Children subtab provides information on the child nodes within the menu structure. Child nodes are either functions or menus (submenus). Child nodes are displayed in a hierarchy with the following information, as applicable: display name, internal menu name, function name, type, and description.

Direct Parents

The Direct Parents subtab allows the user to see the direct parent(s), if any, of the navigation menu. A direct parent is a menu that contains this menu directly as a submenu. This feature is useful in identifying the direct impact of any changes that may be made to this menu.

For each parent, the prompt and internal menu name is shown.

Grants

The Grants subtab displays the associated grants that secure the navigation menu.

For each associated grant the following is shown: name, grantee type, grantee, valid dates, data context type, object, and instance set.

Update Menu

Use this page to update an existing navigation menu.

All fields can be updated except for the menu code.

The direct parents of a menu can be deleted in the Direct Parents tab.

You cannot update a parent menu from this tab. You must navigate to the parent menu record itself to update it.

> **Note:** You cannot replace an existing parent menu with another menu, as the parent menu is used as the primary key of the hierarchy mapping. Instead, you have to delete this existing (child) menu and add a new menu. Also, the sequence number cannot be updated since it is the primary key. You can update the prompt and description.

Duplicate Menu

Use this page to duplicate a menu and copy its hierarchy of children. You must give the duplicate menu and new code (internal name).

View Menu

Use this page to view details of a menu.

Delete Menu

Use this page to delete a menu.

Note that you cannot delete a referenced menu. A menu can be referenced by any of the following:

- Children (menu or function)
- Menu parents
- Grants

Permissions

A permission is the smallest unit of securable action that can be performed on the system. A permission can either be abstract permissions or executable functions (menu). It can either be a system level permission or be sensitive to a data context. For example, a particular JSP page may be an executable permission and "View Person" may be an abstract permission.

The Permissions pages can be accessed from the Functional Administrator and Functional Developer responsibilities. For more information on these, see:Overview of Functional Administrator and Functional Developer Responsibilities, *Oracle E-Business Suite Setup Guide*.

You can search for permissions from the main page. You can update, duplicate, or remove a permission found in your search results. You can also create a new permission from this page.

Search for permissions using the following criteria:

- Name
- Code
- Object Name

Create Permission

Use these pages to create a permission.

1. Specify a name for the permission.
2. Specify a code for the permission. The code is the internal name for the permission. Once the permission has been saved, the code cannot be updated.
3. If this permission applies to a specific object, specify the object.
4. If you want to add this permission to a permission set now, select a permission set.

Update Permission

Use this page to update an existing permission.

Note that you cannot update the code (internal name) for the permission.

1. You can specify a new name for the permission.
2. You can specify a new object if the permission applies to a specific object.

You can update the permission set information as well:

1. To add this permission to a permission set, select a permission set from the list of values for "Add this to a Permission Set".
2. To delete this permission from a permission set, select the permission set in the table and click the **Remove** button.

Select the **Apply** button to save your changes.

Duplicate Permission

Use this page to duplicate an existing permission.

Note that you must enter a unique code for the new permission you are creating.

1. Specify a name for the permission.
2. Specify a code for the permission. The code is the internal name for the permission. Once the permission has been saved, the code cannot be updated.
3. If this permission applies to a specific object, specify the object.
4. If you want to add this permission to a permission set now, select a permission set.

View Permission

Use this page to view details on an existing permission.

You can update or duplicate a permission from this page. You can delete a permission from this page if it does not belong to a permission set.

Delete Permission

Use this page to delete a permission.

Permission Sets

Permission sets provide a way to group related permissions together. You can create a new permission set from this page.

The Permission Sets HTML-based pages can be accessed from the Functional Administrator and Functional Developer responsibilities. For more information on these, see:Overview of Functional Administrator and Functional Developer Responsibilities, *Oracle E-Business Suite Setup Guide*.

You can search for permission sets using the following criteria:

- Name
- Code

You can update, duplicate, or delete permission sets found in your search.

Create Permission Set

Use this page to create a permission set.

1. Specify a name for the permission set.
2. Specify a code for the permission set. The code is the internal name for the permission set. Once the permission set has been saved, the code cannot be updated.

Use the **Permission Set Builder** to add permissions to your new permission set. You

can also add existing permission sets to the new permission set.

Update Permission Set

Use this page to update an existing permission set.

You can specify a new name for the permission set. Note that you cannot update the code (internal name) for the permission set.

If you want to update which permissions and permission sets belong to this permission set, use the **Permission Set Builder** to do so.

Permission Set Manager

Once you have your permission set defined, you can update the contents of the permission set in the Permission Set Manager tab.

Hierarchy of Children

The Hierarchy of Children subtab provides information on the child nodes in the permission set structure. A child node is either a permission or permission set. Child nodes are displayed in a hierarchy with the following information: display name, permission set name (if applicable), permission name (if applicable), and description.

Direct Parents

The Direct Parents subtab allows you to see the permission sets, if any, that include the current permission set. This feature is useful in identifying the direct impact of any changes that may be made to this permission set.

Grants

The Grants subtab displays the associated grants that secure the navigation menu.

For each associated grant, the name, grantee type, grantee, valid dates, data context type, object name, and instance set name is displayed.

Duplicate Permission Set

Use this page to duplicate an existing permission set.

Note that you must enter a unique code for the new permission set you are creating.

1. Specify a name for the permission set.

2. Specify a code for the permission set. The code is the internal name for the permission set. Once the permission set has been saved, the code cannot be updated.

If you want to update which permissions and permission sets belong to this permission

set, use the **Permission Set Builder** to do so.

View Permission Set

Use this page to view details on an existing permission set.

Click **Update** to update the permission set.

Delete Permission Set

Use this page to delete a permission set. If a permission set is a child of another permission set, it cannot be deleted without first being removed from its parent permission set.

Compile Security Concurrent Program

Use this concurrent program to compile your menu data. Compiling your menu data allows for the system to determine more quickly whether a function is available to a particular responsibility/menu.

A request to run this program is automatically submitted when you make changes using the Menus form.

Parameter

Everything

This parameter takes the value Yes or No. "No" is used to recompile only those entities that are marked as needing recompilation. "Yes" is used to recompile all entities, and can take a long time. "No" is the default value.

Function Security Reports

Use the function security reports to document the structure of your menus. You can use these reports as hardcopy to document your customized menu structures before upgrading your Oracle E-Business Suite software.

The function security reports consist of the Function Security Functions Report, the Function Security Menu Report, and the Function Security Navigator Report.

These reports are available through the Function Security Menu Reports request set. For each report, specify the responsibility whose function security you want to review.

> **Note:** If a function and a menu are associated with the same menu entry and the function is excluded then the submenu and its children are also excluded.

If the submenu is also included on another branch of the menu (same level or higher) than the submenu and functions will be included and should be on the reports assuming all other function security conditions are met.

Function Security Function Report

Specify a responsibility when submitting the report. The report output lists the functions accessible by the specified responsibility.

The report does not include items excluded by function security rules.

Function Security Menu Report

Specify a responsibility when submitting the report. The report output lists the complete menu of the responsibility, including all submenus and functions.

The report indicates any excluded menu items with the rule that excluded it.

Function Security Navigator Report

Specify a responsibility when submitting the report. The report output lists the menu as it appears in the navigator for the responsibility specified.

This report does not include items excluded by function security rules, or non-form functions that do not appear in the navigator.

Users of a Responsibility Report

This report documents who is using a given responsibility. Use this report when defining or editing application users.

Report Parameters

Application Name

Choose the name of the application to which the responsibility you want in your report belongs.

Responsibility Name

Choose the name of the responsibility you want in your report.

Report Heading

The report heading indicates the application name and responsibility for which you

requested a report.

Column Headings

User Name

The name of the user who is assigned to the responsibility.

Start Date

The date the responsibility became active for the user.

End Date

The date the responsibility either becomes inactive or became inactive for the user. If no end date appears for a user, then this responsibility is always enabled for the user.

Description

The description of the user who is assigned to the responsibility.

Related Topics

Overview of Oracle E-Business Suite Security, page 4-1

Defining a Responsibility, page 4-4

Overview of Function Security, page 4-9

Responsibilities field help, page 4-19

Users field help, page 4-23

Active Responsibilities Report

This report shows all the responsibilities that are currently active, the users who can currently access each responsibility, and the start and end dates when they can access the responsibility.

Report Parameters

None.

Report Heading

This displays the name of the report, the date and time the report was run, and the page number.

Column Headings

Application Name

The name of the application associated with the responsibility.

Responsibility Name

The name of the currently active responsibility.

User Name

The name of the user who can currently access the responsibility.

Start Date

The date when the user can begin accessing the responsibility.

End Date

The date when the user can no longer access the responsibility. See: Overview of Oracle E-Business Suite Security, page 4-1.

Related Topics

Overview of Oracle E-Business Suite Security, page 4-1

Defining a Responsibility, page 4-4

Responsibilities field help, page 4-19

Users field help, page 4-23

Active Users Report

This report shows all the usernames that are both currently active and have at least one active responsibility. It also displays all the responsibilities that users can access, and the start and end dates when they can access each responsibility.

Report Parameters

None.

Report Heading

The report heading displays the name of the report, the date that the report was run, and the page number.

Column Headings

User Name

The Oracle E-Business Suite name of the currently active user. The start and end dates that you specify in the Users window determine whether a username is currently active.

Application Name

The name of the application associated with the responsibility.

Responsibility Name

The name of the currently active responsibility.

Start Date

The date when the user can begin accessing the responsibility. You can specify a start date when you assign the responsibility to the user in the Responsibilities block of the Users window.

End Date

The date when the user can no longer access the responsibility. You specify an end date when you assign the responsibility to the user in Responsibilities block of the Users window.

Reports and Sets by Responsibility Report

This report identifies which reports (and other concurrent programs) and report sets are included in the request security groups available to any given responsibility. Use this report when defining or editing responsibilities.

Report Parameters

If you enter no parameters, the report documents all reports and report sets accessible from each responsibility.

Application Short Name

Choose the application name associated with the responsibility whose available reports and report sets you wish to report on.

If you do not choose an application name, the report documents all reports and report sets accessible from each responsibility.

Responsibility Name

Choose the name of a responsibility whose available reports and report sets you wish to report on. You must enter a value for Application Short Name before entering a value for Responsibility Name.

Report Headings

The report headings list the report parameters you specify, and provide you with general information about the contents of the report.

Related Topics

Overview of Oracle E-Business Suite Security, page 4-1

Defining Request Security, page 4-6

Responsibilities field help, page 4-19

Oracle Application Object Library REST Security Services

Oracle E-Business Suite Release 12.2 introduces Oracle Application Object Library *REpresentational State Transfer* (REST) security services as a new integration option, providing more versatility than previously possible. In particular, the REST security services facilitate the development of customizable support for mobile applications.

How these services are used is detailed under the descriptions of the four main APIs that are associated with them:

- Login Service
- Session Management Service
- Authorization Service
- Logout Service

Login Service

Every web service request made to Oracle E-Business Suite must be *authenticated*: that is to say, have the caller's credentials validated. The authentication process is often more informally referred to as *logging in*.

The REST Login Service validates the Oracle E-Business suite user credentials, and returns an access token. This access token can then be used with every subsequent service request that requires authentication, without the need for the username and password to be sent every time.

The Login Service is based on the HTTP basic authentication scheme.

```
URL:
  http(s)://<EBSHost>:<EBSPort>/OA_HTML/RF.jsp?function_id=mLogin

HTTP Methods
  GET or POST

Content Type:
  JSON, XML

HTTP Headers:
  Authorization header as per HTTP BASIC authentication scheme
  Accept-Language header for client language in RFC 5646 format.
  Input Parameters:
    No input payload
 Output Parameters:
     accessToken - Token to be passed with every service request
requiring authentication
   accessTokenName - Name of the access token
   ebsVersion - Oracle E-Business Suite release version
   userName - Authenticated Oracle E-business Suite user name

Sample Request:

  GET /OA_HTML/RF.jsp?function_id=mLogin HTTP/1.1
  Authorization: Basic c3lzYWRtaW46c3lzYWRtaW4=
  Accept-Language: en-GB,en-US;q=0.8,en;q=0.6
  Content-Type: application/xml

Sample possible responses:

  200 (On Success):

  <response>
    <data>
      <accessToken>L6ZzqNQKSYSwniSQ2NZK9emtSz</accessToken>
      <accessTokenName>fnd122dv</accessTokenName>
      <ebsVersion>12.2.0</ebsVersion>
      <userName>SYSADMIN</userName>
    </data>
  </response>

  401 (On Failure)

  <response>
    <status>
      <code>401</code>
      <description>Invalid username/password</description>
    </status>
    <data>
      <accessToken>-1</accessToken>
      <accessTokenName></accessTokenName>
      <ebsVersion></ebsVersion>
      <userName></userName>
    </data>
  </response>
```

Session Management Service

Any operation or service processing Oracle E-Business Suite data (to read, insert, update, or delete) is sensitive to the Oracle E-Business Suite security context

(responsibility, application, security group, and operating unit). This means that the same operation will have different results if performed with a different security context. It is therefore critical to maintain a meaningful security context for the relevant requests, and to reset this security context when required.

The Session Management REST service allows the client to initialize and re-initialize the Oracle E-Business Suite session's security context at any time. This service upgrades the access token with the security context information, so that all the requests holding the access token implicitly carry the security context information to the service provider.

The Session Management REST service also retrieves the current session security context information when required.

```
URL:
  http(s)://<EBSHost>:<EBSPort>/OA_HTML/RF.jsp?function_id=mInit

Content Type:
  JSON, XML

HTTP Headers:
  Cookie header with accessTokenName and accessToken from mLogin Service

Operation 1:
  To retrieve current session context information

HTTP Method:
  GET

Input Parameters:
  No input payload

Output Parameters:
  • "resp" - responsibility information in the following structure
     id - responsibility ID
     applId - responsibility application id
     key  - responsibility internal name
     applKey - responsibility application short name
  • "securityGroup" - Security group information in the following
structure
     id - Security group ID
     key - security group internal name
  • "org" - Operating Unit information in the following structure
     id - Operating Unit id
     key - Operating unit internal name
  • "userId" - authenticated Oracle E-Business Suite user ID
  • "username" - authenticated Oracle E-Business Suite user name
  • "accessToken" - current access token
  • "accessTokenName" - current access token name
  • "language" - Current session language

Sample Request:

  GET /OA_HTML/RF.jsp?function_id=mInit HTTP/1.1
  Cookie: <accessTokenName>=<accessToken>
  Content-Type: application/xml

Sample Possible Responses:

  200 (On Success):

  <response>
    <data>
      <resp>
        <id>20872</id>
        <key>SYSTEM_ADMINISTRATION</key>
        <applId>178</applId>
        <applKey>ICX</applKey>
      </resp>
      <securityGroup>
        <id>0</id>
        <key>STANDARD</key>
      </securityGroup>
      <org>
        <id>1733</id>
```

```
<key>Vision Communications (USA)</key>
      </org>
      <userId>0</userId>
      <userName>SYSADMIN</userName>
      <accessToken>bXoQiG6Sxt9SAw7lUw49L5FjwL</accessToken>
      <accessTokenName>fnd122dv</accessTokenName>
    <language>US</language>
    </data>
  </response>

'On Failure' returns the following HTTP error status codes along with
error description based on different error
conditions. Besides this information in response body the service also
returns the corresponding HTTP error status
code:

400 – for any invalid input payload
500 – for any unexpected exceptional conditions, which should be a code
bug
401 – for unauthorized access

Sample failure response:

  <response>
    <status>
      <code>error status code</code>
      <description>error description</description>
    </status>
    <data></data>
  </response>


Operation 2:

To initialize or re-initialize the Oracle E-Business Suite session's
security context

HTTP Method:
  POST

Input Parameters:
  "resp" - responsibility information in the following structure:
    • id - responsibility ID
    • applId - responsibility application ID
    • key - responsibility internal name
    • applKey - responsibility application short name
        o Supports both IDs and keys (internal names).
        o Uses either id attributes or key (internal name) attributes
for passing responsibility information.
        o A combination of id attribute for one entity and key attribute
for another entity is not supported. For example,
          id (responsibility id), applKey is NOT supported. Similarly,
appId (responsibility application ID), key
          (responsibility internal name) is not supported.
    • "securityGroup"  - Security group information in the following
structure. Supports both IDs and keys (internal names). Use
       either id attribute or internal key attribute for passing
security group information.
       o id -  Security group ID
       o key - Security group internal name
    • "org" - Operating Unit information in the following structure. It
```

```
supports both the ID and key (internal name). Use either ID
      attributes or internal key attribute for passing operating unit
information.
       o id - Operating Unit ID
       o key - Operating unit internal name

The parameters "resp", "securityGroup", and "org" are all optional.
The  request must send at least one of the parameters (rest,
securityGroup, org)
All the input parameters are case sensitive

Output Parameters:

Status Response

Sample Request:

  POST /OA_HTML/RF.jsp?function_id=mInit HTTP/1.1
  Cookie: <accessTokenName>=<accessToken>
  Content-Type:application/xml

    <data>
      <resp>
        <key>SYSTEM_ADMINISTRATION</key>
        <applKey>ICX</applKey>
      </resp>
      <securityGroup>
        <key>STANDARD</key>
      </securityGroup>
      <org>
        <key>Vision Communications (USA)</key>
      </org>
    </data>

Sample Response:

200 (On Success):

  <response>
    <status>
      <code>200</code>
      <description>success</description>
    </status>
    <data></data>
  </response>

'On Failure' returns the following HTTP error status codes along with
error description based on different error
conditions. Besides this information in response body, the service also
returns the corresponding HTTP error status
code:

400 - for any invalid input payload
500 - for any unexpected exceptional conditions, which should be a code
bug
401 - for unauthorized access

Sample failure response:

<response>
  <status>
```

```
<code>error status code</code>
    <description>error description</description>
  </status>
  <data></data>
</response>
```

Authorization Service

This Oracle Applications Object Library REST security service allows client applications to retrieve the list of the assigned responsibilities, roles, and privileges for all logged-in users, filtered by specified criteria. The authorization security data returned by the service works with both with traditional function security and the RBAC model.

```
URL:
  http(s)://<EBSHost>:<EBSPort>/OA_HTML/RF.jsp?function_id=mACS

HTTP Method:
  POST

Content Type:
  JSON

HTTP Headers:

  Cookie header with accessTokenName and accessToken from mLogin Service

Operation 1

  Returns logged-in user's roles and responsibilities filtered by input
filter criteria.

  Input Parameters:

  "mode" - Honors values {"role", "resp", "roleresp", "parent"}. When
the value is:
    • "role":  The service returns all the logged-in user's roles
matching the filter criteria.
    • "resp": The service returns all the logged-in user's
responsibilities matching the filter criteria
    • "roleresp": The service returns the logged-in user's both the
roles and responsibilities matching the
       filter criteria
    • "parent": For all the roles/responsibilities matching the filter
criteria, this service returns the
       assigning role (wf_user_role_assignments.assigning_role). The
assigning role may be different than the
       immediate parent in the role hierarchy.
   "appName" - Application Short Name. To filter the authorization data
based on application short name.
   "roleCode" - Internal name of role/responsibility
(WF_LOCAL_ROLES.NAME). To filter the authorization data based
    on internal name of role/responsibility

Output Parameters:

  "data" - The array of roles/responsibilities in the following
structure.
    • "NAME" - Internal name of role/responsibility
    • "DISPLAY_NAME" - Display name of role/responsibility in session
language
    • "RESPONSIBILITY_ID" - WF Orig_System_Id for a role and
responsibility ID for a responsibility
    • "RESPONSIBILITY_APPLICATION_ID" - owning application ID of a
role/responsibility
    • "APPL_SHRT_NAME" - Owning application short name of a
role/responsibility
    • "SECURITY_GROUP_KEY" - Security group internal name for a
responsibility.

Sample Request:

  POST /OA_HTML/RF.jsp?function_id=mACS HTTP/1.1
  Cookie: <accessTokenName>=<accessToken>
  Content-type:application/json
```

```
   {
     mode:"role",
     appName:"FND",
     roleCode:"UMX|FND_SYSTEM%"
   }

Sample Response:


  "data":[{
  "NAME":"UMX|FND_SYSTEM_INTEGRATION_DEVELOPER",
  "DISPLAY_NAME":"System Integration Developer",
  "RESPONSIBILITY_ID":"0",
  "RESPONSIBILITY_APPLICATION_ID":"0",
  "APPL_SHRT_NAME":"FND",
  "SECURITY_GROUP_KEY":"NONE"
},{
  "NAME":"UMX|FND_SYSTEM_INTEGRATION_ANALYST",
  "DISPLAY_NAME":"System Integration Analyst",
  "RESPONSIBILITY_ID":"0",
  "RESPONSIBILITY_APPLICATION_ID":"0",
  "APPL_SHRT_NAME":"FND",
  "SECURITY_GROUP_KEY":"NONE"
}]

On Failure returns the following HTTP error status codes, along with
error description based on different error
conditions. Besides this information in response body, the service also
returns the corresponding HTTP error
status code:
400 – for any invalid input payload
500 – for any unexpected exceptional conditions, which should be a code
bug
401 – for unauthorized access

Sample failure response:

  {
    "status":{
    "code":"401",
    "description":"This is a bad request"
  }

Operation 2

Returns logged-in user's privileges (EBS executable functions and
non-executable permissions), filtered by input
filter criteria.

Input Parameters:

  "mode" - Honors only value {"function"}. When the value is:
    • "function":  the service returns all the logged-in user's
privileges matching the filter criteria.
  "resp" - array of responsibilities (with the below structure of
attributes) for which accessible privileges is
retrieved
  • "resp_id" – responsibility Id
  • "appl_id" – responsibility_application_id
  • "secgrp_id" – security_group_id
  • "filter" – An optional filter criteria at "resp" record level to
```

```
filter the list of accessible privileges from
     this responsibility.
      o "functionName" – Function internal code based on which list of
accessible privileges from this responsibility are
       filtered.
      o "webCall" – EBS Function FND_FORM_FUNCTIONS.WEB_HTML_CALL based
on which list of accessible privileges from this
       responsibility are filtered
  • "filter" - Global filter criteria applied on all the accessible
privileges retrieved from the list of all
     input responsibilities. Global filter criteria is being over-ridden
by the resp specific filter criteria (if
     provided) for it's accessible privileges.
      o "functionName" – Function internal code based on which list of
accessible privileges from the entire
          responsibility list are filtered
      o "webCall" – EBS Function FND_FORM_FUNCTIONS.WEB_HTML_CALL based
on which list of accessible privileges from
         the entire responsibility list are filtered.

Output Parameters:

  "data" - The array of responsibilities along with the corresponding
privileges list in the following structure.
    • "resp_id" : Responsibility Id
    • "appl_id" : Responsibility Application Id
    • "secgrp_id"  : Security Group Id
    • "responsibility_name" : Responsibility Display Name
    • "funcDetail" : Array of privileges accessible from this
responsibility. The privilege has the following structure:
       "RESPONSIBILITY_NAME" : Responsibility display name
       "FUNCTION_ID" : Function Id
       "FUNCTION_NAME" : Internal Function Name
       "USER_FUNCTION_NAME" : Function display name
       "WEB_HTML_CALL" : For executable functions, the function URL

Sample Request:

  POST /OA_HTML/RF.jsp?function_id=mACS HTTP/1.1
  Cookie: <accessTokenName>=<accessToken>
  Content-type:application/json

    {
      mode:"function",
      resp:[
            { resp_id:20420, appl_id:1, secgrp_id:0,
              filter: {functionName:"%HELP%"}
            },
           { resp_id:23175, appl_id:861, secgrp_id:0
            }
          ],
      filter: {
        functionName:"%HELP%",
        webCall:"%"
       }
    }

Sample Response:

  {
  data:
```

```
{
     resp_id: "20420"
     appl_id: "1"
     secgrp_id: "0"
     responsibility_name: "System Administrator"
     funcDetails:
      {
       function_id: "1002781"
       function_name: "FND_HELP_BUILDER"
       user_function_name: "Help Builder"
       web_html_call: "jsp/fnd/fndhelpbuilder.jsp?custom_level=10"
      }
      {
       function_id: "1035387"
       function_name: "FND_HELP_REPORTS_PAGE"
       user_function_name: "Fnd Help Reports Page"
       web_html_call:
"OA.jsp?page=/oracle/apps/fnd/gfm/webui/FndHelpReportsPG"
      }
    }
    {
     resp_id: "23175"
     appl_id: "861"
     secgrp_id: "0"
     responsibility_name: "iMeeting System Monitor responsibility"
     funcDetails:
      {
       function_id: "1005377"
       function_name: "ICX_HELP"
       user_function_name: "Self Service Help"
       web_html_call: "fndgfm/fnd_help.get/US/FND/@ICXPHP"
      }
      {
       function_id: "1032936"
       function_name: "UMX_LOGIN_HELP"
       user_function_name: "Login Help UI"
       web_html_call:
"OA.jsp?page=/oracle/apps/fnd/umx/password/webui/LoginHelpPG&akRegionApp
licationId=0"
      }
    }
  }

On Failure returns the following HTTP error status codes, along with
error description based on different error
conditions. Besides this information in response body, the service also
returns the corresponding HTTP error status
code:
400 – for any invalid input payload
500 – for any unexpected exceptional conditions, which should be a code
bug
401 – for unauthorized access

Sample failure response:

  {
  "status":{
  "code":"401",
  "description":"This is a bad request"
  }
```

```
Operation 3  - Oracle E-Business suite implementation for ADFmf ACS
interface)
Provides an Oracle E-Business Suite implementation for ADFmf pre-defined
interface for ACS REST service.

Input Parameters:
  "userId" : Logged-in EBS username. If this differs from logged-in
user, it reports an error.
  "filterMask" : Array honoring values {role, privilege}. When the value
is :
   • "role" : Filters the data based on roles passed through roleFilter
   • "privilege" : Filters the data based on privileges passed through
privilegeFilter
  "roleFilter" : List of role and responsibility internal codes
(WF_LOCAL_ROLES.NAME) on which data is filtered.
  "privilegeFilter"": List of functions and internal codes
(FND_FORM_FUNCTIONS.FUNCTION_NAME) on which data is filtered.

Output Parameters:
  "userId"   : logged-in Oracle E-Business Suite user name.
  "roles"   : List of logged-in user roles and responsibilities in
internal codes
  "privileges"    : List of logged-in user functions in internal codes

Sample Request:

  POST /OA_HTML/RF.jsp?function_id=mACS HTTP/1.1
  Cookie: <accessTokenName>=<accessToken>
  Content-type:application/json

    {
    "userId": "johnsmith",
    "filterMask": ["role", "privilege"],
    "roleFilter": [ "role1", "role2" ],
    "privilegeFilter": ["priv1", "priv2", "priv3"]
   }

Sample Response:

  {
    "userId": "johnsmith",
    "roles": [ "role1" ],
    "privileges": ["priv1", "priv3"]
  }
```

Logout Service

This service invalidates the access token as an authentication mechanism, and thereby also invalidates any associated authenticated sessions.

```
URL:
 /OA_HTML/RF.jsp?function_id=mLogout

HTTP Methods:
  GET

Content-Type:
  XML or JSON

HTTP Headers:
  Cookie header with accessTokenName and accessToken from mLogin Service

Input Parameters:
 No input parameters

Output parameters :
  "accessToken" : Invalidated access token

Sample Request:

  GET /OA_HTML/RF.jsp?function_id=mLogout HTTP/1.1
  Cookie: <accessTokenName>=<accessToken>
  Content-type:application/xml

Sample Response:

  <response>
    <data>
      <accessToken>-1</accessToken>
      <accessTokenName>fnd122dv</accessTokenName>
      <ebsVersion />
    </data>
  </response>

'On Failure' returns the following HTTP status codes along with error
description based on different error conditions:

500 - for any unexpected exceptional conditions

  <response>
    <status>
  <code>500</code>
  <description>Detailed error description</description>
    </status>
    <data></data>
  </response>
```

Cookie Domain Scoping

In the World Wide Web, a *cookie* is a mechanism of storing state across web requests When a site is accessed, a user's browser stores information in the cookie such as a session identifier When the site is accessed again, the information in the cookie can be reused

If a domain is not specified, the browser does not send the cookie beyond the originating host Explicitly setting the cookie domain scope tells the browser where the cookie can be sent.

Features of Cookie Domain Scoping include:

- Reduces the attack surface of Oracle E-Business Suite
- Provides additional protection for communication between the browser and the Oracle E-Business Suite web tier
- Provides the ability to define the scope for cookie sharing to avoid unnecessary exposure
- Allows for a custom scope to be defined

Cookie domain scoping configuration is set via the profile option 'Oracle Applications Session Cookie Domain Profile' (ICX_SESSION_COOKIE_DOMAIN).

This profile option can take the folowing values:

- **Host**: The domain attribute of the cookie will not be set. The cookie is scoped (restricted) to the originating server, and not sent to any other machines. This setting offers the minimum possible attack surface, and therefore the maximum level of protection. It should always be the setting used in a DMZ environment.

  Example:

  ```
  host=myebsserver.us.example.com
  ICX_SESSION_COOKIE_DOMAIN=HOST
  Set-Cookie: <no domain attribute>
  ```

- **Domain**: The domain attribute of the cookie will be set. The cookie is shared with all hosts in first level domain, with the domain value being derived from the APPS_WEB_AGENT profile option. This setting is the default, and is similar to pre-12.2 behavior.

  Example:

  ```
  host=myebsserver.us.example.com
  ICX_SESSION_COOKIE_DOMAIN=DOMAIN
  Set-Cookie: …; domain= .us.example.com
  ```

- **Custom**: The domain valuse is user-defined. A broader scope for the cookie may be specified. This setting is not generally recommended.

  Example:

  ```
  host=myebsserver.us.example.com
  ICX_SESSION_COOKIE_DOMAIN=.example.com (CUSTOM)
  Set-Cookie: …; domain=.example.com
  ```

Implementing Cookie Domain Scoping

When setting up this feature, you should take into account the following:

- Identify any integrations that use an Oracle E-Business Suite cookie and determine what domains these .cover

- Scope cookies to narrowest domain consistent with business or other operational rquirements.
- Always scope DMZ cookies to Host setting.
- Aim to use narrow cookie scoping when planning new configurations.

Troubleshooting Cookie Domain Scoping

Problem: Authenticated Oracle E-Business Suite users cannot navigate to external integrations such as Oracle E-Business Suite Extensions for Endeca.

Check the following:

- Value of the ICX_SESSION_COOKIE_DOMAIN profile at site and server levels.
- Domains of the Oracle E-Business Suite and External Integrations with which browser is communicating.
- Response headers containing `Set-Cookie`. The easiest method is to use a browser plug-in. Alternatives are intercepting proxy (more difficult with SSL enabled), and Apache logging.

Problem: With setting of Domain, browser is not sending cookie back to host.

Check the following:

- Ensure that the APPS_WEB_AGENT profile option matches the web entry point with which the browser is communicating.

Problem: With custom setting: browser is not sending cookie back to host.

Check the following:

- Custom value must be a strict subset of host name. For example, the browser will not send a cookie of domain `example.com` to `myapps.myexample.com`.

References

For more information, refer to:

- My Oracle Support Knowledge Document 403537.1, *Secure Configuration Guide For Oracle E-Business Suite Release 12*.
- My Oracle Support Knowledge Document 340490.1, *Oracle E-Business Suite R12 Configuration in a DMZ*.

Allowed JSPs

Introduction

The *Allowed JSPs* feature reduces the attack surface of Oracle E-Business Suite by enabling the creation of a whitelist of JavaServer Pages (JSPs) that are allowed access to Oracle E-Business Suite.

> **Note:** A *whitelist* is a list of items that are explicitly allowed access to a resource.

Configuration of actively allowed JSPs avoids unnecessary exposure, with JSPs that are not used being denied access. The implementation strategy also allows custom JSPs to be defined in the list of allowed JSPs.

> **Tip:** Conceptually, the principles are broadly similar to those employed in DMZs, which use a URL firewall as a whitelist mechanism. Refer to Document My Oracle Support Knowledge Document 380490.1, *Oracle E-Business Suite R12 Configuration in a DMZ*.

The Allowed JSPs feature offers two levels of protection. Using it with the shipped configuration provides some level of protection, for minimal effort. Alternatively, you can create and use customized whitelists to match your family and product usage. This option is recommended for the best reduction in attack surface.

Getting Started

The basic strategy for deploying the Allowed JSPs feature is as follows.

1. Evaluate product family usage.
2. Cross-check restricted JSPs against the access log.
3. Add custom JSPs as required.
4. Enable the Allowed JSPs feature.
5. Continue to refine the list (for example, comment out any JSPs not found to be used).

The main configuration files are:

- $OA_HTML/WEB-INF/web.xml
- $FND_SECURE/allowed_jsps.conf (master configuration file)

- $FND_SECURE/allowed_jsps_<Family>.conf
- $FND_SECURE/allowed_jsps_<Family>_<Product>.conf

These are all described in more detail in the next section. Custom configuration files may also be defined.

In addition to the configuration files, the feature uses a new profile option called 'Allow Unrestricted JSP Access'. This is described after the configuration files.

Configuration Files

Syntax

The following syntax is used with the Allowed JSPs configuration files:

- Full URL path must be specified for each JSP file.

  Example:

  ```
  /OA_HTML/example.jsp
  ```

- Comments are denoted by a leading '#'.

  Example:

  ```
  # /OA_HTML/example.jsp
  ```

- Include files may be defined.

  Example:

  ```
  include example_jsp.conf
  ```

$OA_HTML/WEB-INF/web.xml file

This file includes a filter to read the Allowed JSPs configuration file and set additional parameters for the configuration.

Example content:

```
<filter>
    <filter-name>AllowedJSPFilter</filter-name>
    <filter-class>oracle.apps.fnd.security.WLFilter</filter-class>
...
<filter-mapping>
    <filter-name>AllowedJSPFilter</filter-name>
    <url-pattern>*.jsp</url-pattern></filter-mapping>
...
```

The MASTER_FILE parameter defines the master configuration file name. The default name is $FND_SECURE/allowed_jsps.conf.

Example from $OA_HTML/WEB-INF/web.xml:

```
...
<init-param>
      <param-name>MASTER_FILE</param-name>
      <param-value>/[absolute path]/allowed_jsps.conf</param-value>
</init-param>
...
```

the UPDATE_CHECK_INTERVAL parameter defines the frequency (in seconds) of the refresh of the cached allowed JSP file. The default value is 60.

Example from $OA_HTML/WEB-INF/web.xml:

```
...
<init-param>
      <param-name>UPDATE_CHECK_INTERVAL</param-name>
      <param-value>60</param-value>
</init-param>
...
```

$FND_SECURE/allowed_jsps.conf file

This is the default MASTER_FILE listed in the web.xml file. File contents include common JSPs for Oracle E-Business Suite, and product family configuration files (as include files).

$FND_SECURE/allowed_jsps_<Family>.conf file

This is the default MASTER_FILE listed in the web.xml file. File contents include common JSPs for Oracle E-Business Suite product families, and JSPs that are accessible for a product family.

Examples

allowed_jsps_HR.conf:

```
# $Header: allowed_jsps_HR.conf ...
include allowed_jsps_HR_HXC.conf
include allowed_jsps_HR_PER.conf
include allowed_jsps_HR_PAY.conf
```

allowed_jsps_FIN.conf:

```
# $Header: allowed_jsps_FIN.conf ...
/OA_HTML/SelfRegistration.jsp
/OA_HTML/IrecVisitor.jsp
/OA_HTML/AnonymousLogin.jsp
/OA_HTML/jsp/gl/GLLaunchAHM.jsp
```

$FND_SECURE/allowed_jsps_<Family>_<Product>.conf file

File contents include JSPs that are accessible for a product.

$FND_SECURE/<allowed_jsps_CUSTOM>.conf file

This is a custom configuration file. Contents may typically include custom JSPs (from outside Oracle E-Business Suite).

Creating a Custom Configuration File

The procedure to do this is as follows.

1. Create a new custom configuration file. For example, `allowed_jsps_CUSTOM.conf`.

2. Add your custom JSP to the custom configuration file. For example, `/OA_HTML/MyCustomPage.jsp`.

3. Add an corresponding entry in the MASTER_FILE, `allowed_jsps.conf`. For example, `include allowed_jsps_CUSTOM.conf`.

The `allowed_jsps.conf` file would then look something like this:

```
# $Header: allowed_jsps.conf ...
/OA_HTML/AppsLocalLogin.jsp
/OA_HTML/OALogout.jsp
/OA_HTML/OARegion.jsp
/OA_HTML/RF.jsp
...
include allowed_jsps_FIN.conf
include allowed_jsps_HR.conf
...
include allowed_jsps_CUSTOM.conf
...
```

Profile Option For Allowed JSPs

The profile option 'Allow Unrestricted JSP Access' (FND_SEC_ALLOW_JSP_UNRESTRICTED_ACCESS) allows unrestricted access or restricted access to be set.

Key characteristics are:

- Can be set at either Site or Server level.
- Default value is 'Yes', at Server level. This allows unrestricted access to JSPs.
- A value of 'No' allows restricted access to allowed JSPs as per the JSP whitelist filter.
- A value of 'NULL' enables restricted access if the JSP servlet filter is configured.
- The profile is refeshed at the UPDATE_CHECK_INTERVAL rate.

Reducing Access to JSPs

Three levels of granularity exist here:

- If you are not using any products in a particular product family, comment out that family in the MASTER_FILE.
- To restrict access at the product level, comment out the appropriate product-level JSPs.

- To restrict access at the individual JSP level, comment out the JSPs in question.

> **Note:** In the event of problems, you shoukd be prepared to revert modifications to product level JSPs or individual JSPs.

Commenting Out Product Families

Example configuration file: allowed_jsps.conf (MASTER_FILE):

```
# $Header: allowed_jsps.conf ...

/OA_HTML/AppsLocalLogin.jsp
/OA_HTML/OALogout.jsp
/OA_HTML/OARegion.jsp
/OA_HTML/RF.jsp
...
include allowed_jsps_FIN.conf
#include allowed_jsps_HR.conf
#include allowed_jsps_CRM.conf
...
```

Practical Aspects of Using the Allowed JSPs Feature

Using the Allowed JSPs feature should be undertaken pragmatically, and you should be ready to tune access to meet your business needs. To help do this, you can refer to your Apache access log (`access_log`). This log stores a history of JSP usage for your instance, and can be used to validate your configuration.

> **Tip:** JSPs mentioned in the access log should be in your Allowed JSP lists.

You can create a JSP inventory using the following script.

```
mkdir ~/jsp

# Get URI path of all JSP files called, along with the HTTP response
code

cd $LOG_HOME/ora/10.1.3/Apache/
cat access_log* | tr '?' ' ' |
awk '$7 ~ /\.jsp$/  {printf "%s\t%s\n", $(NF-1), $7}' |
  sort | uniq -c > ~/jsp/called-jsps

# Separate JSPs based on response codes (good and bad).

cat ~/jsp/called-jsps | awk '$2 < 400 {print $3}' > ~/jsp/good-jsp-calls
cat ~/jsp/called-jsps | awk '400 <= $2 {print $3}' > ~/jsp/bad-jsp-calls

# See if all the good JSPs from your system are in the Allowed JSPs.

cd $FND_SECURE/  # Where the Allowed JSP files are
cat ~/jsp/good-jsp-calls | while read jsp
do
  if ! grep "$jsp" allowed_*.conf > /dev/null
    then
      echo "Missing: $jsp"
    fi
done
```

Logging and Troubleshooting

Given the need to tune and tweak access to JSPs using this feature, you may find unexpected access errors - either access being permitted when you did not expect it would be, or (more commonly) access being denied when you expected it to be permitted.

For example, you may see an error with the text "Requested resource or page is not allowed in this site":

Seeing this message is the expected behavior (i.e. not an error) if it is seen on an attempt to access a JSP that is intentionally restricted. On the other hand, it is an error (i.e. should not be displayed) if it is seen on an attempt to access a JSP that is unrestricted.

Logging can be used to investigate issues such as this. By default, it is disabled (turned off). When enabled, logging will write messages to the designated log file as follows:

- Whether or not the filter is enabled.
- Whether or not access is allowed to a JSP file on which an access attempt is being

made.

Logging Parameters

Key parameters include LOG_ENABLED and LOG_FILENAME. Each of these is described briefly below.

- The LOG_ENABLED parameter (in $OA_HTML/WEB-INF/web.xml) is used to enable and disable logging. The default value is FALSE. When LOG_ENABLED set to TRUE, LOG_FILENAME must be valid.

  Example from $OA_HTML/WEB-INF/web.xml:

  ```
  <init-param>
        <param-name>LOG_ENABLED</param-name>
        <param-value>false</param-value>
  </init-param>
  ```

- The LOG_FILENAME parameter is used as the directory location and filename to write specific log messages for this filter. The default value is `/tmp/AllowedJSPs.log`.

  Example from $OA_HTML/WEB-INF/web.xml:

  ```
  <init-param>
        <param-name>LOG_FILENAME</param-name>
        <param-value>/tmp/AllowedJSPs.log</param-value>
  </init-param>
  ```

Enabling Logging

To start logging:

1. Verify LOG_FILENAME location.

2. Set LOG_ENABLED to true in the $OA_HTML/WEB-INF/web.xml file:

   ```
   <init-param>
         <param-name>LOG_ENABLED</param-name>
         <param-value>true</param-value>
   </init-param>
   ```

3. Restart application tier web services.

Example Log File

You may see content that looks something like the following:

```
AllowedJSPFilter:LOG_ENABLED
AllowedJSPFilter:MASTER_FILE='/d2/r122/j122/fs1/inst/apps/j122_testsys/a
ppl/fnd/12.0.0/secure/allowed_jsps.conf'
AllowedJSPFilter:Update check interval '60' =60 seconds
AllowedJSPFilter:loading='/d2/r122/j122/fs1/inst/apps/j122_testsys/appl/
fnd/12.0.0/secure/allowed_jsps.conf'
AllowedJSPFilter:Creating master list
AllowedJSPFilter:Including from
/d2/r122/j122/fs1/inst/apps/j122_testsys/appl/fnd/12.0.0/secure/allowed_
jsps.conf
AllowedJSPFilter:start loading
/d2/r122/j122/fs1/inst/apps/j122_testsys/appl/fnd/12.0.0/secure/allowed_
jsps.conf
AllowedJSPFilter:Pending load:
/d2/r122/j122/fs1/inst/apps/j122_testsys/appl/fnd/12.0.0/secure/ATGAppro
vedJSPList.conf
AllowedJSPFilter:Pending load: /tmp/CustomApprovedJSPList.conf

...

AllowedJSPFilter:loaded 0 records pending inclusion 2 files
AllowedJSPFilter:start loading
/d2/r122/j122/fs1/inst/apps/j122_testsys/appl/fnd/12.0.0/secure/ATGAppro
vedJSPList.conf
AllowedJSPFilter:loaded 452 records pending inclusion 1 files
AllowedJSPFilter:start loading /tmp/CustomApprovedJSPList.conf
AllowedJSPFilter:loaded 1 records pending inclusion 0 files
AllowedJSPFilter:Profile Allow Unrestricted JSP Access is set to: N
AllowedJSPFilter:Profile Allow Unrestricted JSP Access is DISABLED -
RESTRICT ACCESS
AllowedJSPFilter:Start checking update job
AllowedJSPFilter:initiated
AllowedJSPFilter:CONFIG END
AllowedJSPFilter:Profile Allow Unrestricted JSP Access is set to: N
AllowedJSPFilter:Profile Allow Unrestricted JSP Access is DISABLED -
RESTRICT ACCESS
AllowedJSPFilter:Profile Allow Unrestricted JSP Access is set to: N
AllowedJSPFilter:Profile Allow Unrestricted JSP Access is DISABLED -
RESTRICT ACCESS

...
AllowedJSPFilter:In doFilter whiteList is enabled
AllowedJSPFilter:PATH=/OA_HTML/AppsLocalLogin.jsp
AllowedJSPFilter:/OA_HTML/AppsLocalLogin.jsp is in
AllowedJSPFilter:Authorized /OA_HTML/AppsLocalLogin.jsp
AllowedJSPFilter:In doFilter whiteList is enabled
AllowedJSPFilter:PATH=/OA_HTML/RF.jsp
AllowedJSPFilter:/OA_HTML/RF.jsp is in
AllowedJSPFilter:Authorized /OA_HTML/RF.jsp
AllowedJSPFilter:In doFilter whiteList is enabled
AllowedJSPFilter:PATH=/OA_HTML/jsp/fnd/aolj_check_cookie.jsp
AllowedJSPFilter:/OA_HTML/jsp/fnd/aolj_check_cookie.jsp is not in
AllowedJSPFilter:Rejected /OA_HTML/jsp/fnd/aolj_check_cookie.jsp
AllowedJSPFilter:Profile Allow Unrestricted JSP Access is set to: N
AllowedJSPFilter:Profile Allow Unrestricted JSP Access is DISABLED -
RESTRICT ACCESS

...

AllowedJSPFilter:File
/d2/r122/j122/fs1/inst/apps/j122_testsys/appl/fnd/12.0.0/secure/allowed_
```

```
jsps.conf changed
AllowedJSPFilter:File/d2/r122/j122/fs1/inst/apps/j122_testsys/appl/fnd/1
2.0.0/secure/ATGApprovedJSPList.conf has disappeared
AllowedJSPFilter:Modification detected, refreshing list
AllowedJSPFilter:Including from
/d2/r122/j122/fs1/inst/apps/j122_testsys/appl/fnd/12.0.0/secure/allowed_
jsps.conf
AllowedJSPFilter:start loading
/d2/r122/j122/fs1/inst/apps/j122_testsys/appl/fnd/12.0.0/secure/allowed_
jsps.conf
AllowedJSPFilter:Pending load:
/d2/r122/j122/fs1/inst/apps/j122_testsys/appl/fnd/12.0.0/secure/ATGAppro
vedJSPList.conf

...

AllowedJSPFilter:Pending load: /tmp/CustomApprovedJSPList.conf
AllowedJSPFilter:loaded 3 records pending inclusion 2 files
AllowedJSPFilter:start loading
/d2/r122/j122/fs1/inst/apps/j122_testsys/appl/fnd/12.0.0/secure/ATGAppro
vedJSPList.conf
AllowedJSPFilter:loaded 452 records pending inclusion 1 files
AllowedJSPFilter:start loading /tmp/CustomApprovedJSPList.conf
AllowedJSPFilter:failed to load /tmp/CustomApprovedJSPList.conf (No such
file or directory) /tmp/CustomApprovedJSPList.conf
AllowedJSPFilter:Master list was refreshed
AllowedJSPFilter:Profile Allow Unrestricted JSP Access is set to: N
AllowedJSPFilter:Profile Allow Unrestricted JSP Access is DISABLED -
RESTRICT ACCESS
AllowedJSPFilter:In doFilter whiteList is enabled
AllowedJSPFilter:PATH=/OA_HTML/OA.jsp
...
```

Common Issues

- *Symptom*: Only the RF.jsp, OA.jsp, and fnderror.jsp JSPs are accessible.

  Probable Cause: Configuration is invalid or incomplete, so master JSP whitelist configuration file is not found.

  Resolution: Review and confirm the value of the MASTER_FILE parameter in the $OA_HTML/WEB-INF/web.xml file.

- *Symptom*: Error message denying access is displayed upon trying to access JSP that should not be restricted.

  Probable Causes: (1) JSP is commented out in configuration file or is not defined in the allowed JSPs list. (2) Browser caching issue.

  Resolution: (1) Uncomment JSP in configuration file or add JSP to configuration file. (2) Clear browser cache and restart browser.

- *Symptom*: Error message denying access is displayed upon trying to access JSP that is in list of allowed JSPs.

  Probable Causes: (1) URL does not match exactly. May be caused by double slashes in URLs, or different location. (2) Browser caching issue.

Resolution: (1) Ensure JSP and location match exactly. (2) Clear browser cache and restart browser

- *Symptom*: Logging fails when enabled.

  Probable Causes: (1) LOG_FILENAME location in $OA_HTML/WEB-INF/web.xml is invalid, or file system is full. (2) LOG_ENABLED is not set to 'true'.

  Resolution: (1) Update LOG_FILENAME parameter to point to valid location, or resolve file system space issues. (2) Set LOG_ENABLED parameter to 'true' and restart application web tier services.

5

Auditing and Monitoring

Overview of Auditing and Monitoring

Oracle E-Business Suite supports auditing two categories of actions that have been performed: *user activity* and *database row changes*.

As well as this capability to audit past activities, support is also provided for identifying the use to which a database connection is currently being put: this is accomplished via the *Database Connection Tagging* feature.

Auditing User Activity

Auditing users is supported by the following settings and features:

- Sign-On:Audit Level profile option setting
- Audit Reports

Based on the audit level chosen, Sign-On audit records usernames, dates, and times of system access, as well as users' terminals, forms, and responsibilities.

Auditing Database Row Changes

Auditing database row changes is supported by:

- From the **Help** menu, **About This Record** ...
- AuditTrail:Activate profile option setting
- Audit forms

Related Topics

Auditing User Activity, page 5-2

Setting Up Sign-On Audit, page 5-3

Sign-On Audit Reports, page 5-5

Monitor Users, page 5-21

Reporting on AuditTrail Data, page 5-6

Setting Up AuditTrail, page 5-7

AuditTrail Tables, Triggers and Views, page 5-8

Reporting on Audit Information, page 5-14

Disabling AuditTrail and Archiving Audit Data, page 5-15

Audit Installations, page 5-23

Audit Groups, page 5-25

Audit Tables, page 5-27

Auditing User Activity

Oracle E-Business Suite provides a Sign-On Audit feature that allows you to:

- Track what your users are doing and when they do it.
- Choose who to audit and what type of information to audit.
- View quickly online what your users are doing.
- Check the security of your application.

With Sign-On Audit, you can record usernames, terminals, and the dates and times your users access Oracle E-Business Suite. Sign-On Audit can also track the responsibilities and forms your users use, as well as the concurrent processes they run.

Major Features

Selective Auditing

Sign-On Audit lets you choose who to audit and what type of user information to track. You can selectively determine what audit information you need, to match your organization's needs.

Monitor Application Users

The Monitor Users form gives you online, real-time information about who is using Oracle E-Business Suite and what they are doing.

You can see what users are signed on (application username and operating system login name), what responsibilities, forms, and terminals they are using, how long they have

been working on forms, and what Oracle database processes they are using.

Sign-On Audit Reports

Sign-On Audit Reports give you historical, detailed information on what your users do in your application.

You can give search criteria to narrow your search for information. You can also sort your Sign-On Audit information to create easy-to-read reports.

Setting Up Sign-On Audit

You use the Sign-On:Audit Level user profile option to control who Sign-On Audit tracks and the level at which they are audited.

Use the *Monitor Users* form to view online what your users are doing.

Use the *Submit Reports* form to submit Sign-On Audit Reports that give you detailed audit information.

Enabling Sign-On Audit

Use the System Profile Values form to enable Sign-On Audit. Choose the scope of your audit and who to audit by setting the user profile level at the user, responsibility, application, or site profile levels.

> **Note:** Users cannot see or change this profile option.

After you set or change audit levels, the new audit levels for a user take effect the next time the user signs onto Oracle E-Business Suite from the operating system.

Selecting Audit Levels

The Sign-On:Audit Level profile option allows you to select a level at which to audit users who sign on to Oracle E-Business Suite.

Four audit levels provide increasing levels of monitoring: None, User, Responsibility, and Form.

Auditing level None is the default, and tracks:

- No activities by any users who sign on to Oracle E-Business Suite

Auditing at the User level tracks:

- Who signs on to your system
- The times users log on and off
- The terminals in use

Auditing at the Responsibility level performs the User level audit functions and also tracks:

- The responsibilities users choose
- How much time users spend using each responsibility

Auditing at the Form level performs the Responsibility and User level audit functions, and also tracks:

- The forms users choose
- How long users spend using each form

Auditing Levels and System Overhead

In planning your organization's Sign-On Audit implementation, you should consider the additional system overhead required to monitor and audit your users as they access Oracle E-Business Suite. The more users you audit, and the higher the level of auditing, the greater the system overhead such as processing costs and disk space. You should balance your organization's auditing needs with the resources available, obtaining additional resources if the existing ones are insufficient to support the required auditing activities as well as the actual workload.

Example - Audit Users, Responsibilities, and Forms

An example implementation of Sign-On Audit would be to audit all of your users' sign-ons, the responsibilities they select, and the forms they access.

To accomplish this, you would set Sign-On:Audit Level to:

- Form audit
- At the Site profile level

Example - Audit a specific responsibility, except for one user

Another example of using Sign-On Audit is for an organization to audit all users of the Personnel Manager responsibility, except for MJONES.

In this example, you do not need to audit the forms the user accesses, or the responsibilities they select.

To set up this implementation, set Sign-On:Audit Level to:

- User audit
- At the responsibility profile level for the Personnel Manager responsibility

You also set Sign-On:Audit Level to:

- None
- At the user profile level for the application user MJONES

Using the Application Monitor

Use the Monitor Users form to monitor who is using Oracle E-Business Suite and what they are doing. You can monitor users at any time.

The Application Monitor lets you see what users are signed on, what responsibilities, forms, and terminals they are using, how long they have been working on forms, and what Oracle database processes they are using.

> **Important:** You can only monitor those users that are being audited by Sign-On Audit. The Application Monitor also reflects the level of auditing you define for your users.

Sign-On Audit Reports

Use the Submit Requests form to print standard audit reports.

You can generate reports detailing which users are signing on; the responsibilities they are accessing; the forms they are using; concurrent requests they are submitting; and details of any attempts to log on to other users' accounts.

Oracle E-Business Suite provides the following Sign-On Audit reports:

Signon Audit Concurrent Requests, page 5-30 (shows who submitted what requests)

Signon Audit Forms, page 5-32 (shows who accessed what forms)

Signon Audit Responsibilities, page 5-34 (shows who accessed what responsibilities)

Signon Audit Unsuccessful Logins, page 5-36 (shows who unsuccessfully attempted to sign on as another user)

Signon Audit Users, page 5-38 (shows who signed on to Oracle E-Business Suite)

For each report, you can also specify search criteria that makes your report as brief as you need.

Related Topics

Overview of User and Data Auditing, page 5-1

Auditing User Activity, page 5-2

Setting Up Sign-On Audit, page 5-3

Sign-On Audit Reports, page 5-5

Monitor Users, page 5-21

Reporting On AuditTrail Data

AuditTrail lets you keep a history of changes to your important data: what changed, who changed it, and when. With AuditTrail, you can easily determine how any data row or element obtained its current value. You can track information on most types of fields, including character, number and date fields.

When you enter or update data in your forms, you change the database tables underlying those forms. AuditTrail tracks which rows in the database were updated at what time, and which user was logged in using the associated form(s).

AuditTrail

Oracle E-Business Suite provides a auditing mechanism based on Oracle database triggers. AuditTrail stores change information in a "shadow table" of the audited table. This mechanism saves audit data in an uncompressed but "sparse" format, and you enable auditing for particular tables and groups of tables ("audit groups").

Audit Trail Update Tables Report

This program creates database triggers on the tables in your audit groups for your installations. It also creates shadow tables, one for each audited table, to contain the audit information. If you have changed your audit definitions or disabled auditing for an audit group, the program drops or modifies the auditing triggers and shadow tables appropriately.

The program also builds special views you can use to retrieve your audit data for reporting.

Changing Your Audit Tables

You may add additional columns to audit after auditing has begun on a table. However, the shadow table does not track the column changes that occurred before the column(s) were added. If you add columns you must rerun the AuditTrail Update Tables Report to:

- Add the necessary column(s) to the shadow table
- Regenerate the audit triggers and procedures for the table so that they now audit the additional column(s)

Related Topics

Overview of User and Data Auditing, page 5-1

Reporting on AuditTrail Data, page 5-6

Setting Up AuditTrail, page 5-7

Reporting on Audit Information, page 5-14

Disabling AuditTrail and Archiving Audit Data, page 5-15

Audit Installations, page 5-23

Audit Groups, page 5-25

Audit Tables, page 5-27

Setting Up AuditTrail

You can choose to store and retrieve a history of all changes users make on a given table. Auditing is accomplished using *audit groups*, which functionally group tables to be audited. For a table to be audited, it must be included in an enabled audit group.

The steps for setting up AuditTrail are as follows.

Verify Select Privileges on SYS.DBA_TABLES

Have your database administrator grant SELECT privileges on SYS.DBA_TABLES to the APPLSYS account. Normally, this step will already have been done as part of the installation or upgrade.

Define Audit Groups

These are groups of tables and columns; you do not necessarily need to include all the columns in a given table. You enable auditing for audit groups rather than for individual tables. You would typically group together those tables that belong to the same business process (for example, purchase order tables).

A given table can belong to more than one audit group. If so, the table is audited according to the highest level of enabling for any of its groups, where Enabled is the highest, followed by Disable Dump Data, Disable No Growth, and Disable Purge Table, in that order.

You can enable auditing for a maximum of 240 columns for a given table, and you can enable auditing for all types of table columns except LONG, RAW, or LONG RAW. Your audit group must include all columns that make up the primary key for a table; these columns are added to your audit group automatically. Once you have added a column to an audit group, you cannot remove it. See: Audit Groups, page 5-25.

Define Audit Installations

You choose the registered Oracle IDs at your site that you want to audit. This allows you to audit across multiple application installations. When a table is added to an audit group, auditing will automatically be enabled for all installations of the table for which audit is enabled. See: Audit Installations, page 5-23.

Run the Audit Trail Update Tables Report to Enable Auditing

Your AuditTrail definitions (and auditing) do not take effect until you run the Audit Trail Update Tables Report. If you change any of your definitions later, you must rerun this program. You run the Audit Trail Update Tables Report from the standard submission (Submit Reports) form.

> **Important:** AuditTrail requires two database connections. If your operating system does not automatically support two database connections (e.g. VMS or MPE/XL), then add to your environment file the environment variable FDATDB=<database connect string>.

AuditTrail Tables, Triggers and Views

When auditing is enabled for the first time, a shadow table to the audited table is automatically created in the same Oracle ID as the audited table. The shadow table contains only the columns to be audited, and all columns in the shadow table are unconstrained, regardless of their status in the table to be audited.

For example, NULLs are always permitted in the shadow table. All columns in the shadow table have the same data types and sizes as their counterparts in the audited table.

The name of the shadow table is the first 24 characters of the original table name plus the suffix "_A" (Audit).

Shadow Table Columns

All AuditTrail shadow tables contain certain special auditing columns. These columns include:

- AUDIT_USER_NAME (the Application User ID, except when changes are applied using SQL*Plus, in which case it is the Oracle ID).
- AUDIT_TIMESTAMP (the date/time when the insertion occurred).
- AUDIT_TRANSACTION_TYPE (I for Insert, U for Update, D for Delete, L for Last, and C for Current).
- AUDIT_TRUE_NULLS (VARCHAR2(250) column containing a delimited list of column names that have changed from NULL).
- The primary key for the table. This is not a special column, but rather all the columns comprising the primary key of the audited table. Note that, by convention, all audited columns are stored when a row is deleted. Likewise, an insert results in a row of NULL values in the shadow table. Changes to the primary key are marked as deletes, but new primary key values are inserted also.

For example, suppose you have the following table:

```
SQL> DESCRIBE AUDIT_DEMO

 NAME            NULL?    TYPE
 --------------- -------- ----
 PRIMARY_KEY              NUMBER(5)
 VALUE_ONE                VARCHAR2(5)
 VALUE_TWO                VARCHAR2(5)
 VALUE_THRE               VARCHAR2(5)
```

Its shadow table is as the following (assuming you audit all your table columns):

```
SQL> DESCRIBE AUDIT_DEMO_A

NAME                     NULL?     TYPE
-----------------------  --------  ----
AUDIT_TIMESTAMP          NOT NULL  DATE
AUDIT_TRANSACTION_TYPE   NOT NULL  VARCHAR2(1)
AUDIT_USER_NAME          NOT NULL  VARCHAR2(100)
AUDIT_TRUE_NULLS                    VARCHAR2(250)
AUDIT_SESSION_ID         NOT NULL    NUMBER
AUDIT_SEQUENCE_ID    NOT NULL    NUMBER
AUDIT_COMMIT_ID        NOT NULL   NUMBER
PRIMARY_KEY                         NUMBER
VALUE_ONE                           VARCHAR2(5)
VALUE_TWO                          VARCHAR2(5)
VALUE_THREE                        VARCHAR2(5)
```

Auditing Triggers and Procedures

When auditing is enabled, the automatically-generated database trigger in the "After" event on the audited table performs the auditing.

This trigger calls a stored procedure to compare each column being audited to see if its value is changing. If so, the procedure saves the previous (old) value to the shadow table.

Auditing creates one row in the shadow table for each audited transaction against the table; thus, a single row in the shadow table represents all old values for all changed columns on that transaction.

The data is not compressed, since a table uses only one byte for a NULL, and AuditTrail represents all unchanged values as NULLs in the shadow table ("sparse" format).

The audit trigger names contain the first 24 characters of the audited table name plus "_AI", "_AU" or "_AD", where one of I, U or D indicates Insert, Update or Delete, respectively. Likewise, the audit procedure names use the first 24 characters of the table name plus "_AIP", "_AUP" or "_ADP". Your table names must be unique within the first 24 characters.

Views

After a shadow table is created, views onto the shadow table are created to allow easier access to the data in the "sparse" rows. These views simplify tasks such as querying a row/column's value on a given date and tracking changes to a row/column over time.

The view name contains the first 24 characters of the audited table name plus "_AC#" or "_AV#" where C or V indicates the type of view and # indicates a number. Due to limitations in creation size, the shadow table columns may need to be broken into multiple views, which are numbered sequentially.

Each view allows slightly different access to the data. One allows the user to reconstruct the value for a row at a given time (_AC), while the other provides simple access to when a value was changed (_AV).

For our example table, the _AV1 and _AC1 views are created as follows:

```
SQL> DESCRIBE AUDIT_DEMO_AV1

NAME                      NULL? TYPE
------------------------- ----- ----
PRIMARY_KEY                     NUMBER
AUDIT_TIMESTAMP                 DATE
AUDIT_SEQUENCE_ID               NUMBER
AUDIT_SESSION_ID                NUMBER
AUDIT_TRANSACTION_TYPE          VARCHAR2(1)
AUDIT_USER_NAME                 VARCHAR2(100)
VALUE_ONE                       VARCHAR2(5)
VALUE_TWO                       VARCHAR2(5)
VALUE_THREE                     VARCHAR2(5)

SQL> DESCRIBE AUDIT_DEMO_AC1

NAME                      NULL? TYPE
------------------------  ----- ----
PRIMARY_KEY                     NUMBER
AUDIT_TIMESTAMP                 DATE
AUDIT_SEQUENCE_ID               NUMBER
AUDIT_SESSION_ID                NUMBER
AUDIT_TRANSACTION_TYPE          VARCHAR2(1)
AUDIT_USER_NAME                 VARCHAR2(100)
AUDIT_COMMIT_ID                 NUMBER
VALUE_ONE                       VARCHAR2(5)
VALUE_TWO                       VARCHAR2(5)
VALUE_THREE                     VARCHAR2(5)
```

How Data Appears in Tables and Views

Here is an example of how data appears in your original table, your shadow table, and your audit views after a series of changes (starting with an empty AUDIT_DEMO table).

```
SQL> INSERT INTO AUDIT_DEMO VALUES (1,'A','A','A');
SQL> INSERT INTO AUDIT_DEMO VALUES (2,'X','X','X');
SQL> SELECT PRIMARY_KEY KEY, VALUE_ONE VAL_1,
     VALUE_TWO VAL_2, VALUE_THREE VAL_3 FROM AUDIT_DEMO;

 KEY VAL_1 VAL_2 VAL_3
---- ----- ----- -----
   1 A     A     A
   2 X     X     X
```

```
SQL> UPDATE AUDIT_DEMO SET VALUE_ONE ='B'
     WHERE PRIMARY_KEY = 1;

 KEY VAL_1 VAL_2 VAL_3
---- ----- ----- -----
   1 B     A     A
   2 X     X     X


SQL> UPDATE AUDIT_DEMO SET VALUE_TWO ='B'
          WHERE PRIMARY_KEY = 1;

 KEY VAL_1 VAL_2 VAL_3
---- ----- ----- -----
   1 B     B     A
   2 X     X     X

SQL> UPDATE AUDIT_DEMO SET VALUE_THREE ='B'
     WHERE PRIMARY_KEY = 1;
SQL> UPDATE AUDIT_DEMO SET VALUE_ONE ='Y'
     WHERE PRIMARY_KEY = 2;
SQL> UPDATE AUDIT_DEMO SET VALUE_ONE = NULL
     WHERE PRIMARY_KEY = 1;
SQL> UPDATE AUDIT_DEMO SET VALUE_ONE ='C'
     WHERE PRIMARY_KEY = 1;
```

After our two inserts and six updates, the final values in the audited table are:

```
KEY VAL_1 VAL_2 VAL_3
---- ----- ----- -----
   1 C     B     B
   2 Y     X     X
```

The final values in the corresponding shadow table are as follows. A row in the shadow table represents the state of the audited row *before* the audited row was changed. Note that if a value in a row doesn't change during the transaction, the shadow table records a null for that value in that transaction.

In our example, the first two rows in the shadow table represent the state where there was no data for our two audited rows before they were inserted. The "prior values" are null values for the two insert transaction (type I) rows. Similarly, when we update the first value of row 1 to be the value B instead of A, the shadow table records the value A in its third row:

```
SQL> SELECT TO_CHAR(AUDIT_TIMESTAMP, 'HH24:MI:SS') TIME,
     AUDIT_TRANSACTION_TYPE TYPE, AUDIT_USER_NAME NAME,
     PRIMARY_KEY KEY, VALUE_ONE VAL_1, VALUE_TWO VAL_2,
     VALUE_THREE VAL_3, AUDIT_TRUE_NULLS FROM AUDIT_DEMO_A;

TIME     TYPE NAME    KEY VAL_1 VAL_2 VAL_3 AUDIT_TRUE_NULLS
-------- ---- ------ ---- ----- ----- ----- ----------------
11:08:16 I    FND60     1
11:08:40 I    FND60     2
11:18:40 U    FND60     1 A
11:20:12 U    FND60     1       A
11:21:54 U    FND60     1             A
11:22:15 U    FND60     2 X
14:20:50 U    FND60     1 B
14:21:15 U    FND60     1                   NYNN

8 rows selected.
```

Given the current values of the row in the audited table, you can trace the changes made to the row by backing up through the corresponding rows in the shadow table.

In our example table, we made two insert and six update transactions, so we see those eight transactions in our shadow table. In the last row, the NYNN indicates that the value in the second table column (VALUE_ONE) has changed from an actual null value (the Y) rather than being an unchanged value (represented by null in the shadow table).

The following two views provide further ways of examining your audited data.

The rows with a transaction type of C in the view indicate the current value of the row when the data was selected (the view is a join between the shadow table and the audited table, so the current value row reflects the current state of the audited table).

The _AC view provides a "filled-in" version of the data, where unchanged values appear instead of being represented by null values. You can order this view by the primary key (rather than by timestamp), so all rows in the shadow table that correspond to a single audited row appear together, with a secondary ordering by timestamp.

```
SQL> SELECT TO_CHAR(AUDIT_TIMESTAMP, 'HH24:MI:SS') TIME,
     AUDIT_TRANSACTION_TYPE TYPE, AUDIT_USER_NAME NAME,
     PRIMARY_KEY KEY, VALUE_ONE VAL_1, VALUE_TWO VAL_2,
     VALUE_THREE VAL_3 FROM AUDIT_DEMO_AC1
     ORDER BY PRIMARY_KEY, AUDIT_TIMESTAMP;

TIME     TYPE NAME        KEY VAL_1 VAL_2 VAL_3
-------- ---- ---------- ---- ----- ----- -----
11:08:16 I    FND60         1 A     A     A
11:18:40 U    FND60         1 B     A     A
11:20:12 U    FND60         1 B     B     A
11:21:54 U    FND60         1 B     B     B
14:20:50 U    FND60         1       B     B
14:21:15 U    FND60         1 C     B     B
17:53:34 C                  1 C     B     B
11:08:40 I    FND60         2 X     X     X
11:22:15 U    FND60         2 Y     X     X
17:53:34 C                  2 Y     X     X

10 rows selected.
```

> **Important:** If the changes to your audited table occur faster than one change per second (that is, more frequently than the one-second granularity provided by SYSDATE), you may see "blurring" of records (i.e. more than one record per transaction) in the _AC view, because of joins used in this view. However, the shadow table itself remains correct, and you can resolve the relevant transactions by referring to the shadow table directly.

The _AV1 view provides a more sparse view of the audit data, ordered by timestamp:

```
SQL> SELECT TO_CHAR(AUDIT_TIMESTAMP, 'HH24:MI:SS') TIME,
     AUDIT_TRANSACTION_TYPE TYPE, AUDIT_USER_NAME NAME,
     PRIMARY_KEY KEY, VALUE_ONE VAL_1, VALUE_TWO VAL_2,
     VALUE_THREE VAL_3, AUDIT_TRUE_NULLS
     FROM AUDIT_DEMO_AV1;

TIME     TYPE NAME    KEY VAL_1 VAL_2 VAL_3 AUDIT_TRUE_NULLS
-------- ---- ------ ---- ----- ----- ----- ----------------
11:08:16 I    FND60     1
11:08:40 I    FND60     2
11:18:40 U    FND60     1 A
11:20:12 U    FND60     1       A
11:21:54 U    FND60     1             A
11:22:15 U    FND60     2 X
14:20:50 U    FND60     1 B
14:21:15 U    FND60     1                   NYNN
17:58:31 C              1 C     B     B
17:58:31 C              2 Y     X     X

10 rows selected.
```

Here is an example of how you might use a view to determine who changed a particular value and when:

```
SQL> SELECT TO_CHAR(AUDIT_TIMESTAMP, 'HH24:MI:SS') TIME,
     AUDIT_TRANSACTION_TYPE TYPE, AUDIT_USER_NAME NAME
     FROM AUDIT_DEMO_AV1
     WHERE PRIMARY_KEY = 1
     AND VALUE_ONE = 'B';

TIME     TYPE NAME
-------- ---- ------
14:20:50 U    FND60
```

Similarly, you might want to determine who changed a value to null and when:

```
SQL> SELECT TO_CHAR(AUDIT_TIMESTAMP, 'HH24:MI:SS') TIME,
     AUDIT_TRANSACTION_TYPE TYPE, AUDIT_USER_NAME NAME
     FROM AUDIT_DEMO_AV1
     WHERE PRIMARY_KEY = 1
     AND VALUE_ONE  IS NULL
     AND SUBSTR(AUDIT_TRUE_NULLS,2,1) = 'Y';

TIME     TYPE NAME
-------- ---- ------
14:21:15 U    FND60
```

Reporting on Audit Information

Report on Your Audit Data

You should write audit reports as needed. AuditTrail provides the views of your shadow tables to make audit reporting easier; you can write your reports to use these views.

You may want to create one or more indexes to your shadow table to speed up your reporting. However, such indexes decrease performance during actual auditing of transactions, so you should drop your indexes from the shadow table when you have finished reporting.

> **Important:** Because the structure of the audited table may change between product versions, AuditTrail does not support upgrading existing shadow tables or audited data. Before an upgrade, you should archive the shadow tables and perform all necessary reporting on the audited data.

Related Topics

Overview of User and Data Auditing, page 5-1

Reporting on AuditTrail Data, page 5-6

Setting Up Release AuditTrail, page 5-7

AuditTrail Tables, Triggers and Views, page 5-8

Disabling AuditTrail and Archiving Audit Data, page 5-15

Audit Installations, page 5-23

Audit Groups, page 5-25

Audit Tables, page 5-27

Disabling AuditTrail and Archiving Audit Data

You may report on your audits or disable auditing at any time. When you disable auditing, you should do the following procedure:

Stop Auditing New Transactions

Disable auditing using *either* "Disable - Prepare for Archive" or "Disable - Interrupt Audit" and running the Audit Trail Update Tables report.

Disable - Prepare for Archive Copies the current values of all rows in the audited table into the shadow table, and then disables the auditing triggers. There is no longer any recording of any changes. You should archive the shadow table before you purge it.

Disable - Interrupt Audit Modifies the triggers to store one "final" row in the shadow table for each row that is modified in the audit table (remember that a given row in the shadow table represents the data in the audited row *before* an update). If a row in the table being audited is changed again (a second time), that change is not recorded. The shadow table grows slowly, until it contains one row for each row in the table being audited. Then there is no longer any recording of any changes.

Archive Your Audit Data

You should archive the information in the shadow tables according to your business needs.

Clean Out the Shadow Table

Before you restart auditing, you should clean out the shadow table. If there were transactions during the time auditing was disabled, and you did not clean out the shadow table, the data in the shadow table would be invalid because it would have a gap where transactions were not recorded. You purge the shadow table(s) by setting the audit group to Disable - Purge Table and running the Audit Trail Update Tables report.

Disable - Purge Table Drops the auditing triggers and views and deletes all data from the shadow table.

Restart Auditing (If Desired)

You restart auditing by setting the audit group to Enable Requested and running the Audit Trail Update Tables report again.

> **Important:** If you disable using Disable Purge Table and then re-enable auditing for a table, AuditTrail flushes the contents of the shadow table when auditing is re-enabled. You should archive any shadow table data that you want to keep before you re-enable auditing.

Related Topics

Overview of User and Data Auditing, page 5-1

Reporting on AuditTrail Data, page 5-6

Setting Up AuditTrail, page 5-7

AuditTrail Tables, Triggers and Views, page 5-8

Reporting on Audit Information, page 5-14

Audit Installations, page 5-23

Audit Groups, page 5-25

Audit Tables, page 5-27

Additional Audit Trail Reporting

This section describes how to set up and manage Audit Trail Reporting functions that are used within OPM.

The following topics are covered:

- Audit Industry Template
- Audit Hierarchy Navigator
- Audit Query Navigator
- Running the Audit Report

Audit Industry Template

This window defines the Industry Audit templates. These templates facilitate binding of the required Audit groups together for easy querying and inquiries.

Before using this window, perform the following:

- Define Audit Tables and Audit columns using Oracle Application Audit under the System Administrator responsibility
- Define Audit Groups using Oracle Application Audit under the System Administrator responsibility

Audit Industry Template Procedure

Use this procedure in completing the Industry Template.

1. Navigate to the **Industry Template** window.
2. Complete the fields as described.
3. Save your changes.

Audit Industry Template Fields

These are the fields in the Audit Industry templates.

Template Name

Enter the name of the desired Audit Template.

Functional Areas

- Functional Group - Enter the functional group associated with this template. This is the same as the Audit Group field on the Audit Group window in System Administration.

Audit Hierarchy Editor

Auditing Navigation

In addition to the standard menu and toolbar, a navigator tree provides a hierarchical display of the objects in a treelike framework.

Nodes and Leaves

The higher level nodes in the navigator tree include windows and database objects. All other nodes, and the objects they contain, are indented to indicate that they belong to these higher level nodes. The terminal node is a leaf.

On the Hierarchy Navigator, the highest level is the Audit Template. The next level is the Audit Group (Functional Group), then the audit table, and finally the columns being audited.

On the Query Navigator, the highest level is the Audit Group (Functional Group). The

next level is the audit table, and below the audit table are the actual data being audited.

Using the Audit Hierarchy Editor

You can navigate to find what has been set up for auditing. This functionality is accomplished by a tree navigator that starts with the Industry template and drill down to groups, tables, and columns. The navigator lets you see a drill-down view of what columns are being audited. A search facility on the tree is provided to search a table or column.

The navigator fetches the data from the audit table to construct the tree, and relies on the Oracle E-Business Suite Object Library table, column registration and uses USER_TABLE_NAME and USER_COLUMN_NAME fields from the FND_TABLES and FND_COLUMNS, respectively.

Before using this window, perform the following:

- Define Audit Tables and Audit columns using the Oracle Application Audit under the System Administrator responsibility
- Define Audit Groups using Oracle Application Audit under the System Administrator responsibility
- Define Industry Audit Templates under the OPM System Administrator responsibility
- Enable Audit Trail, a concurrent process under the System Administrator responsibility

Audit Hierarchy Navigation Procedures

Navigate to the Audit Hierarchy window.

To view table information:

1. Use the tree navigator to view the table names.
2. Select the table name and right-click to display the pop-up menu.
3. Select Display Columns. The Define Query Navigator Display for the Table window displays.

To use the Find Audit Hierarchy function:

1. Use the tree navigator to view the column names.
2. Select the column name and right-click to display the pop-up menu.
3. Select Find. The Find Audit Hierarchy window displays.

4. Select criteria and click Find. A list of templates displays. You can save these as a new audit.

Audit Query Navigator

This interactive query window lets you investigate the changes to any functional group interactively, using a visual approach that is similar to Windows Explorer. When a Particular Node in the left frame is selected, audit trail details are displayed in the right frame. The right frame shows all columns set for auditing. This information is retrieved from the FND_AUDIT_COLUMNS table. The left tree is linked to the right frame with the primary key combination of the table.

Auditing Navigation

In addition to the standard menu and toolbar, a navigator tree provides a hierarchical display of the objects in a treelike framework.

Nodes and Leaves

The higher level nodes in the navigator tree include windows and database objects. All other nodes, and the objects they contain, are indented to indicate that they belong to these higher level nodes. The terminal node is a leaf.

On the Hierarchy Navigator, the highest level is the Audit Template. The next level is the Audit Group (Functional Group), then the audit table, and finally the columns being audited.

On the Query Navigator, the highest level is the Audit Group (Functional Group). The next level is the audit table, and below the audit table are the actual data being audited.

Before using this window, perform the following:

- Define Audit Tables and Audit columns using the Oracle Application Audit under the System Administrator responsibility.
- Define Audit Groups using Oracle Application Audit under the System Administrator responsibility.
- Define Industry Audit Templates under the OPM System Administrator responsibility.
- Define the display look up using the Audit Hierarchy Navigator (Admin Mode). This setup step is not mandatory.
- Enable Audit Trail, a concurrent process under the System Administrator responsibility.

Audit Query Navigation Procedures

Navigate to the Audit Query window.

To use the Find Functional Groups function:

1. Use the tree navigator to view the table names.
2. Select the table name and right-click to display the pop-up menu.
3. Select Find. The Find Functional Groups window displays.
4. Select criteria and click Find. A list of templates displays. You can save these as a new audit.

To view the Audit Results window:

1. Use the tree navigator to view the column names.
2. Select a column name. The Audit Results window automatically displays.
3. Use the Horizontal View and Vertical View buttons to toggle between the two views.

   In the horizontal view, you see the first ten auditing columns. In the vertical view, the column number is unlimited, and can be viewed using the scroll bar.

Audit Report

In situations where comprehensive documentation is needed, (e.g. to support legal or regulatory requirements), a single report request resulting in a single comprehensive report is desirable. This report can then be printed, emailed, or archived.

Since this report could involve a considerable amount of data, a detailed parameter screen is available, allowing you to select only the items of interest.

Submitting the Report

1. Navigate to the Audit Report window. The Enter Report Parameters window is displayed.
2. Select the functional group, or a functional group and audit table name.
3. Complete the optional fields as necessary.
4. Click Select Columns. The Select Reporting Columns window is displayed.
5. Enter at least one column to run the report. The columns displayed are based on the functional group, or a functional group and audit table name criteria selected on the

Enter Report Parameters window.

6. Select Print Options. The Select Printing Options window is displayed.
7. Enter the necessary print information.
8. Select OK.
9. Run the report by selecting Run Report.

Enter Report Parameters Field Reference

Functional Group

Specify the name of the functional group for the report. This is the same as the Audit Group field on the Audit Group window in System Administration.

Audit Table Name (Optional)

Specify the table name from the functional group for the report.

Transacted By (Optional)

Specify the user who is requesting the report.

Transaction Type (Optional)

Specify the type of transaction.

From Date (Optional)

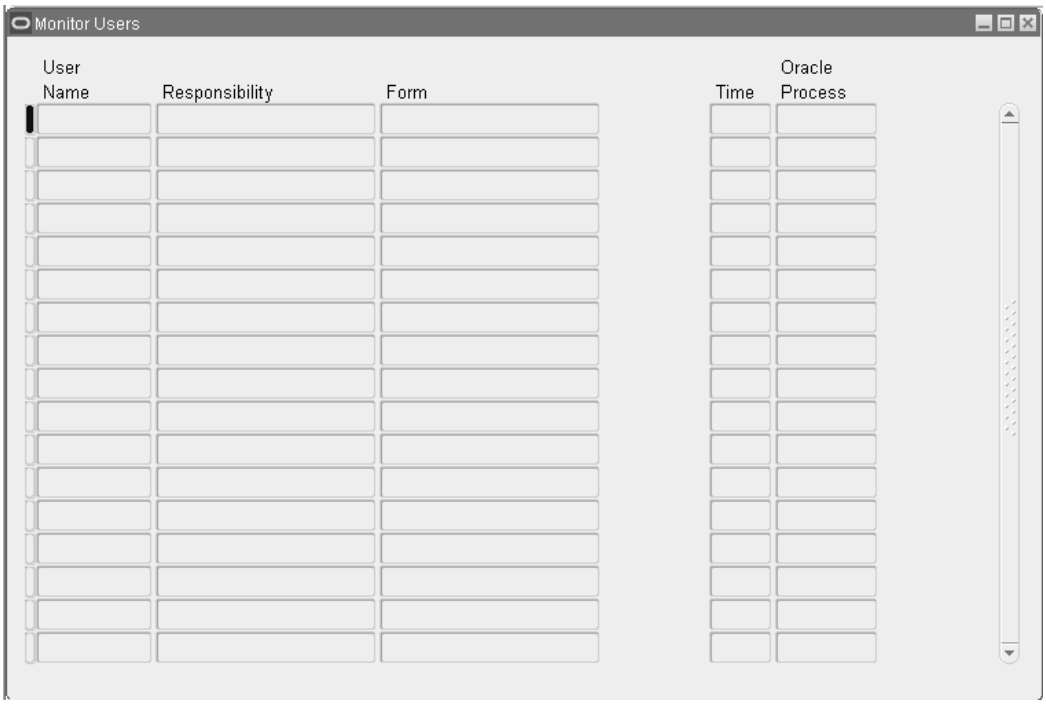
Specify the beginning date for the date range the report will run.

To Date (Optional)

Specify the end date for the date range the report will run.

Monitor Users Window

Use this window to monitor what your application users are currently doing.



As well as seeing which users are signed on, you can see:

- Which responsibilities and forms (windows) they are using
- How long they have been logged in
- What Oracle database processes they are using

In addition, you can monitor all users at a site, all users accessing a specific application or a specific responsibility, or individual users.

> **Note:** You can only monitor those users for whom you have activated Sign-On Audit. See: Overview of User and Data Auditing, page 5-1

Before using this form, select a value for the Sign-On:Audit Level profile option, using the Update System Profile Options window.

Responsibility

The user's responsibility only appears if you have enabled Sign-On Audit at either the Responsibility or Form audit level.

Form

The user's form only appears if you have enabled Sign-On Audit at the Form audit level.

Login

The user's login name.

Time

The length of time the user has been logged on to this application.

Oracle Process

The ORACLE process of the user.

Related Topics

Overview of User and Data Auditing, page 5-1

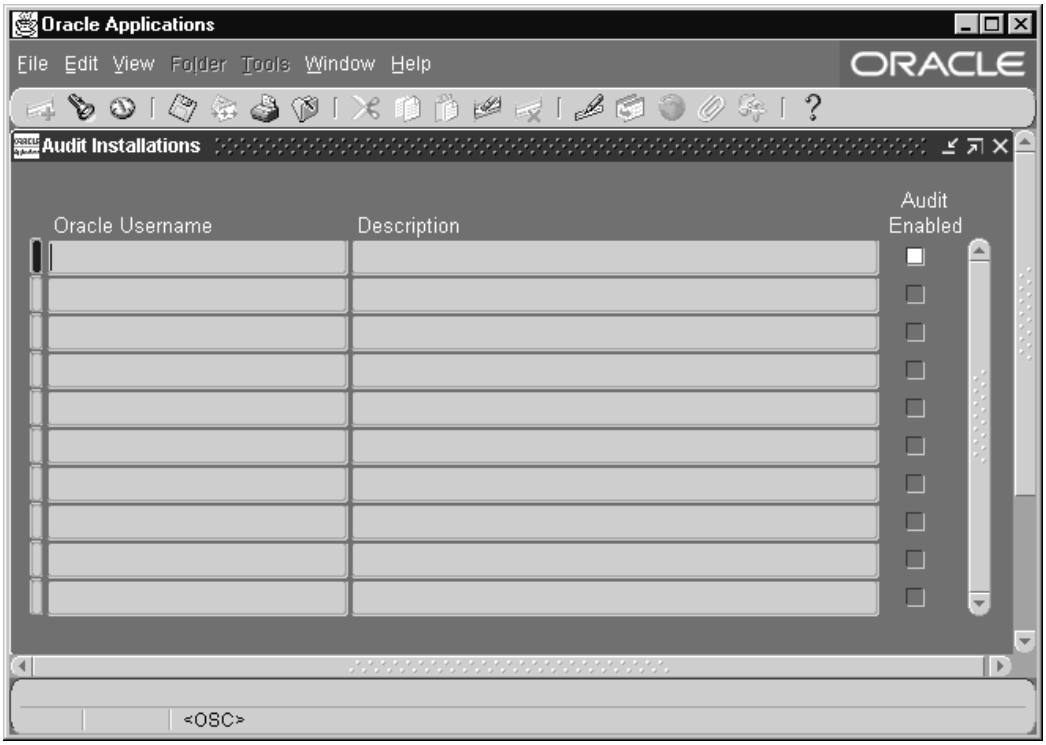
Auditing User Activity, page 5-2

Setting Up Sign-On Audit, page 5-3

Sign-On Audit Reports, page 5-5

Audit Installations Window

Use this window to enable AuditTrail for an Oracle database username at your installation. Such a username grants access privileges to an application's tables and database objects.



For auditing to take effect, you must also define one or more audit groups and run the Audit Trail Update Tables report. See: Reporting on AuditTrail Data, page 5-6.

Before using this form, ensure that the desired Oracle username is registered. The installation process automatically registers Oracle E-Business Suite Oracle usernames, but if you create a custom application, you should follow instructions in My Oracle Support Knowledge Document 1577707.1, "Creating a Custom Application in Oracle E-Business Suite Release 12.2," for Oracle usernames for custom applications.

Oracle Username

Select the Oracle username that owns the tables you wish to audit.

Audit Enabled

Check the Audit Enabled check box to enable AuditTrail for an Oracle username. Before auditing takes effect you must define one or more audit groups and run the Audit Trail Update Tables report.

Related Topics

Overview of User and Data Auditing, page 5-1

Reporting on AuditTrail Data, page 5-6

Setting Up AuditTrail, page 5-7

AuditTrail Tables, Triggers and Views, page 5-8

Reporting on Audit Information, page 5-14

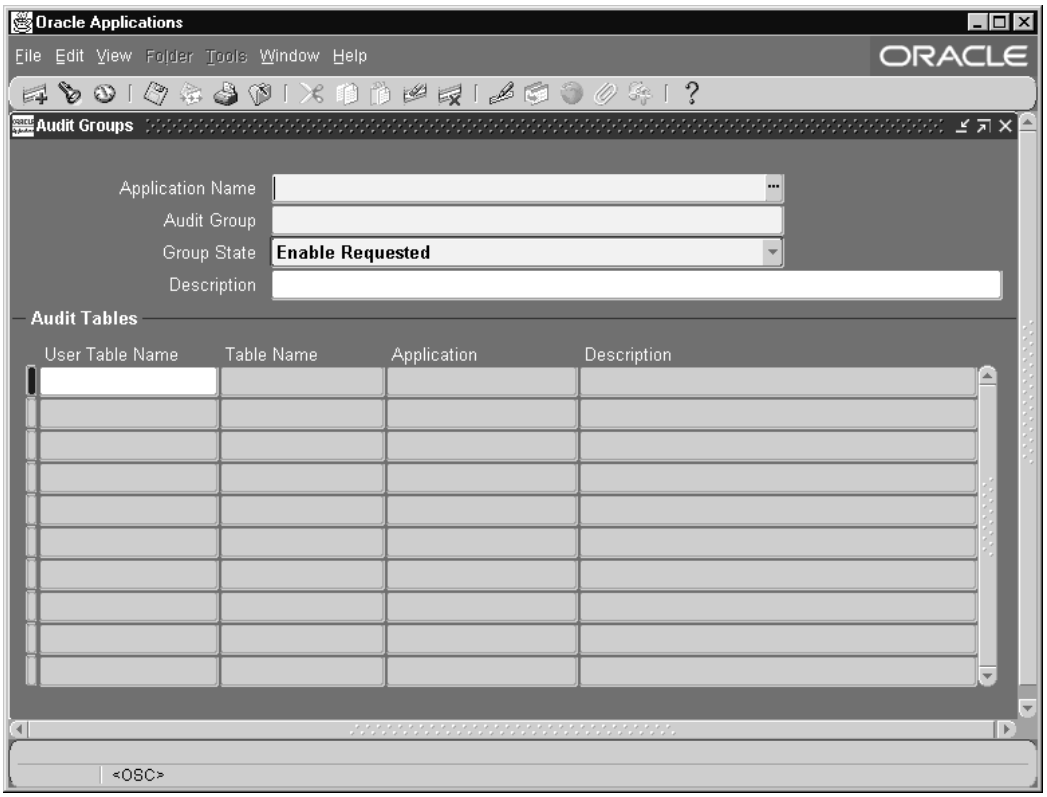
Disabling AuditTrail and Archiving Audit Data, page 5-15

Audit Groups, page 5-25

Audit Tables, page 5-27

Audit Groups Window

Use this window to select the tables that you wish to audit. You audit a table by defining an audit group, which can consist of one or more tables.



First, identify the tables you want to audit, then, using the Audit Tables window, select which columns in each table you wish to audit. Or, select which columns in a particular table you wish to audit (using the Audit Tables window), then define your audit group (using this window).

To enable or disable auditing for the tables in your audit group, run the Audit Trail Update Tables program using the Submit Requests window. If you change the definition or audit state of your group later, you must rerun this program.

Ensure you have done the following before defining your audit groups:

- Define an audit installation using the Audit Installations window.

  Important: Your tables and their primary key information must already be registered and defined for successful auditing. If the table you want to audit is a custom table (not shipped as part of Oracle E-Business Suite), you should also perform the following two steps:

- Register your table *and* its primary key columns using Oracle Application Object Library's Tables window (Application Developer Responsibility).
- Run the Register Tables concurrent program from the Submit Requests window.

Audit Groups Block

Identify your audit group and enable or disable auditing for this group.

Application Name

Select the name of an application to associate with your audit group. The combination of application name and group name uniquely identifies your audit group. An audit group may be used to audit tables in additional applications.

Audit Group

Enter the name of the audit group.

Group State

Choose Enable Requested if you are defining a new audit group. When you run the Audit Trail Update Tables report, the concurrent program creates database triggers for the tables in your audit group. Once you have run the program, this field displays Enabled for audit groups where AuditTrail is active.

Important: All primary key columns in each table in an audit group are automatically selected for auditing, whether or not you use the Audit Tables window to select which columns you wish to audit.

To disable auditing for a group, choose one of the following options and then run the Audit Trail Update Tables report to have your changes take effect.

Disable - Prepare for Archive Copies the current values of all rows in the audited table into the shadow table, and then disables the auditing triggers. This option requires the most space, since there is at least one row in the shadow table for every row in the audited table (and another row in the shadow table for

each transaction on the original row in the audited table). You should then archive the table before you empty the shadow table.

Disable - Interrupt Audit Modifies the triggers to store one final row in the shadow table as the audited row is modified in the audit table (remember that a given row in the shadow table represents the data in the audited row *before* an update). Inserts or further changes are no longer audited. The shadow table then grows slowly, and the data may be accessed by the existing audit views.

Disable - Purge Table Drops the auditing triggers and views and deletes all data from the shadow table.

Audit Tables Block

Identify the application tables you want to audit in your audit group**.**

User Table

Select the end user table name (frequently the same name as the table name) for your database table. Once you choose a table, you see its table name and associated application.

Table Name

This field displays the actual name for the table you have selected to include in your audit group.

Application

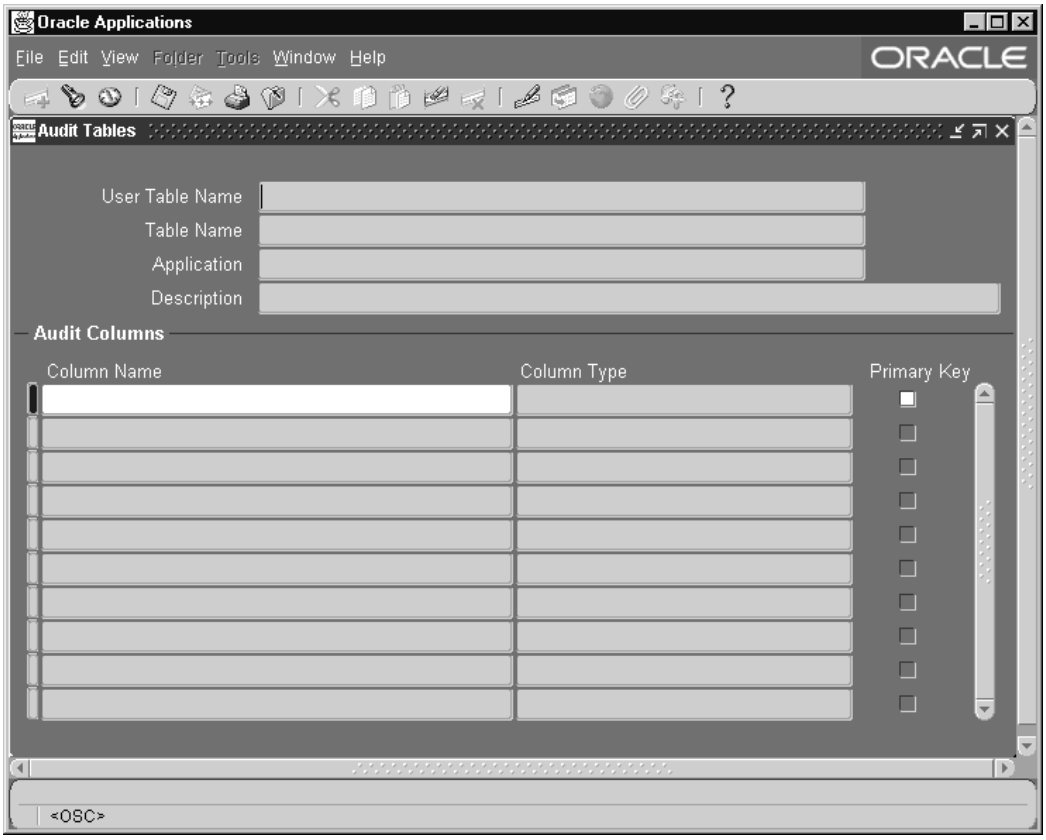
This field displays the application name for the table you have selected to include in your audit group.

Description

This field displays the description for the table you have selected to include in your audit group.

Audit Tables Window

Use this window to select which columns in a table you wish to audit.



First, identify the columns in a table you want to audit. Then, using the Audit Groups window, include the table as part of an audit group. Or, you may define your audit group first (using the Audit Groups window), and then select which columns in the table you want to audit (using this window).

To enable or disable auditing for the tables in your audit group (i.e., the columns you have selected here), you must run the Audit Trail Update Tables program using the Submit Requests window. If you select additional columns to audit, or change the definition or audit state of your group later, you must rerun this program.

Ensure the following is done before defining your audit tables:

- Define an audit installation using the Audit Installations window.

  Important: Your tables and their primary key information must already be registered and defined for successful auditing. If the table you want to audit is a custom table (not shipped as part of Oracle E-Business Suite), you should also perform the following two steps:

- Register your table *and* its primary key columns using Oracle Application Object Library's Tables window (Application Developer Responsibility).

- Run the Register Tables concurrent program from the Submit Requests window.

Define AuditTables Block

Identify the application table you want to audit. Successively selecting *Go - Next Record* from the menu or toolbar displays, in alphabetical order, the name of each application table registered at your installation site.

User Table Name

Select the end user table name (frequently the same name as the table name) for your database table. Once you choose a table, you see its table name and associated application.

Table Name

This field displays the actual name for the table you have selected to include in your audit group.

Application

This field displays the application name for the table you have selected to include in your audit group.

Audit Columns Block

Select the columns you want to audit. Successively selecting *Go - Next Record* from the menu or toolbar displays, in alphabetical order, the name of each application table registered at your installation site.

- You cannot delete a column from auditing once it has been selected.
- You may add additional columns to be audited.
- Each time you select a column to be audited, that change affects every audit group that includes the table which owns the column.

Column Name

Enter the name of the database column you want to audit. You should not explicitly enter the names of your table's primary key columns, since they are entered automatically, and you will get an error message if you try to save a duplicate column name. You can query to see which columns appear automatically.

Note that once you have chosen a column, you cannot delete it from the audit set, though you may add other columns to the set later.

Once you choose a column, you see its column type and whether it is part of the

primary key for this table.

Column Type

This field describes the type of data the column stores, for example, varchar2.

Primary Key

This field displays Yes or No indicating whether the column you are auditing is a primary key column.

Any primary key columns you do not select to audit are automatically included when you save your column selections. For example, if the table you are auditing has two primary key columns, and you choose to audit one of them, the second primary key column is automatically selected when you save your column selections.

Related Topics

Overview of User and Data Auditing, page 5-1

Reporting on AuditTrail Data, page 5-6

Setting Up AuditTrail, page 5-7

AuditTrail Tables, Triggers, and Views, page 5-8

Reporting on Audit Information, page 5-14

Disabling AuditTrail and Archiving Audit Data, page 5-15

Audit Installations, page 5-23

Audit Groups, page 5-25

Signon Audit Concurrent Requests Report

Use this report to view information about who is requesting what concurrent requests and from which responsibilities and forms.

> **Important:** You can only generate Signon Audit Concurrent Requests Reports for those users you are auditing.

Report Parameters

Sort By

Sort the information in your report by operating system login name, the requested start date, and/or application username.

Login Name

Search for a specific login name that meets your other search criteria. If you leave this parameter blank, your report contains all login names that meet your other search criteria.

User Name

Search for a specific application username that meets your other search criteria. If you leave this parameter blank, your report contains all application usernames that meet your other search criteria.

From Request Start Time/To Request Start Time

Search for concurrent requests that meet your other search criteria and have requested start times in a specific time period. Use these parameters to specify the start and end of your time period. If you leave these parameters blank, your report contains concurrent requests from any date that also meet your other search criteria to the current date for this parameter.

Report Heading

The report heading displays the search criteria you entered as parameter values.

Column Headings

Login Name

The operating system login name of the user who submitted the concurrent request.

Request ID

The concurrent request ID of the submitted concurrent request. Use the Concurrent Requests form to view completion information for a concurrent request ID.

Concurrent Program Name

The name of the concurrent program the user submitted. Use the Concurrent Programs form to view detail information about a concurrent program.

User Name

The Oracle E-Business Suite username of the user who submitted the concurrent request. Use the Users form to view detail information about an application user. See: Users, page 4-23.

Responsibility Name

The name of the responsibility from which the user submitted the concurrent request. The responsibility displays only if you audited the user at the responsibility or form Sign-on Audit level. Use the Responsibilities form to view detailed information about a responsibility. See: Responsibilities, page 4-19.

Form Name

The name of the form from which the user submitted the concurrent request. The form name displays only if you audited the user at the form Sign-On Audit level.

Requested Start Time

The date and time the concurrent request started running.

Related Topics

Overview of User and Data Auditing, page 5-1

Auditing User Activity, page 5-2

Setting Up Sign-On Audit, page 5-3

Sign-On Audit Reports, page 5-5

Monitor Users field help, page 5-21

Signon Audit Forms Report

Use this report to view who is navigating to what form and when they do it.

> **Important:** You can only generate a Signon Audit Forms Report for those users you are auditing.

Report Parameters

Sort By

Sort the information in your report by the time users entered or left a form, the name of the form that users access, the operating system login name of the user, the responsibility users access, the terminal that users are on, and/or the application username.

Login Name

Search for information about a specific login name that meets your other search criteria. If you leave this parameter blank, your report contains all login names that meet your

other search criteria.

User Name

Search for information about a specific application username that meets your other search criteria. If you leave this parameter blank, your report contains all application usernames that meet your other search criteria.

Terminal Name

Search for information about a specific terminal that meets your other search criteria. If you leave this parameter blank, your report contains all terminal names that meet your other search criteria.

Responsibility Name

Search for information about a specific responsibility that meets your other search criteria. If you leave this parameter blank, your report contains all responsibilities that meet your other search criteria.

Form Name

Search for information about a specific form that meets your other search criteria. If you leave this parameter blank, your report contains all forms that also meet your other search criteria.

From Active Date/To Active Date

Search for information about forms accessed by users within a specific time period and that meet your other search criteria. Use these parameters to specify the start and end of your time period. If you leave these parameters blank, your report contains forms accessed from any date that also meet your other search criteria to the current date for this parameter.

Report Heading

The report heading displays the search criteria you entered as parameter values.

Column Headings

Username

The Oracle E-Business Suite username of the user who accessed the form. Use the Users form to view detailed information about an application user. See: Users, page 4-23.

Login Name

The operating system login name of the user who accessed the form.

Terminal Name

The operating system ID of the terminal from which the user accessed the form.

Responsibility Name

The name of the responsibility from which the user accessed the form. The responsibility displays only if you audited the user at the responsibility or form Sign-on Audit level. Use the Responsibilities form to view detailed information about a responsibility. See: Responsibilities, page 4-19.

Start Active Time/End Active Time

The dates and times when the user accessed/exited the form. The start active time and end active time display only if you audited the user at the form Sign-on Audit level.

Form Name

The name of the form that the user accessed. The form name displays only if you audited the user at the form Sign-on Audit level.

Related Topics

Overview of User and Data Auditing, page 5-1

Auditing User Activity, page 5-2

Setting Up Sign-On Audit, page 5-3

Sign-On Audit Reports, page 5-5

Monitor Users field help, page 5-21

Signon Audit Responsibilities Report

Use this report to view who is selecting what responsibility and when they do it.

> **Important:** You can only generate Signon Audit Responsibilities Reports for those users you are auditing.

Report Parameters

Sort By

Sort the information in your report by the time users entered or left a responsibility, the operating system login name of the user, the responsibility name, the terminal that users are on, and/or the application username.

Login Name

Search for information about a specific login name that meets your other search criteria. If you leave this parameter blank, your report contains all login names that meet your other search criteria.

User Name

Search for information about a specific application username that meets your other search criteria. If you leave this parameter blank, your report contains all application usernames that meet your other search criteria.

Terminal Name

Search for information about a specific terminal that meets your other search criteria. If you leave this parameter blank, your report contains all terminal names that meet your other search criteria.

Responsibility Name

Search for information about a specific responsibility that meets your other search criteria. If you leave this parameter blank, your report contains all responsibilities that meet your other search criteria.

From Active Date/To Active Date

Search for information about responsibilities accessed by users within a specific time period and that meet your other search criteria. Use these parameters to specify the start and end of your time period. If you leave these parameters blank, your report contains responsibilities accessed from any date that also meet your other search criteria to the current date for this parameter.

Report Heading

The report heading displays the search criteria you entered as parameter values.

Column Headings

Username

The Oracle E-Business Suite username of the user who selected the form. Use the Users form to view detail information about an application user. See: Users, page 4-23.

Login Name

The operating system login name of the user who selected the responsibility.

Terminal Name

The operating system ID of the terminal from which the user selected the responsibility.

Responsibility Name

The name of the responsibility the user used. The responsibility displays only if you audited the user at the responsibility or form Sign-on Audit level. Use the Responsibilities form to view detailed information about a responsibility. See: Responsibilities, page 4-19.

Start Active Time/End Active Time

The dates and times when the user selected/exited the responsibility. The start active time and end active time display only if you audited the user at the responsibility or form Sign-On Audit level.

Related Topics

Overview of User and Data Auditing, page 5-1

Auditing User Activity, page 5-2

Setting Up Sign-On Audit, page 5-3

Sign-On Audit Reports, page 5-5

Monitor Users field help, page 5-21

Signon Audit Unsuccessful Logins Report

Use this report to view who unsuccessfully attempted to sign on to Oracle E-Business Suite as another user. An unsuccessful login occurs when a user enters a correct username but an incorrect password.

You can generate Signon Audit Unsuccessful Logins Reports for any users, regardless of whom you are auditing.

Report Parameters

Sort By

Sort the information in your report by the time users attempt to login, operating system login name of the user, the terminal that users are on, and/or the application username.

Login Name

Search for information about a specific login name that meets your other search criteria. If you leave this parameter blank, your report contains all login names that meet your

other search criteria.

User Name

Search for information about a specific application username that meets your other search criteria. If you leave this parameter blank, your report contains all application usernames that meet your other search criteria.

Terminal Name

Search for information about a specific terminal that meets your other search criteria to make your report as brief as you need. If you leave this parameter blank, your report contains all terminal names that meet your other search criteria.

From Attempt Date/To Attempt Date

Search for information about unsuccessful logins within a specific time period and that meet your other search criteria. Use these parameters to specify the start and end of your time period. If you leave these parameters blank, your report contains unsuccessful logins from any date that also meet your other search criteria to the current date for this parameter.

Report Heading

The report heading displays the search criteria you entered as parameter values.

Column Headings

Username

The Oracle E-Business Suite username of the user who unsuccessfully tried to sign on. Use the Users form to view detail information about an application user. See: Users, page 4-23.

Login Name

The operating system login name of the user who unsuccessfully tried to sign on.

Terminal

The operating system ID of the terminal from which the user unsuccessfully tried to sign on.

Attempt Time

The date and time when the user unsuccessfully tried to sign on. See: Monitor Users, page 5-21.

Related Topics

Overview of User and Data Auditing, page 5-1

Auditing User Activity, page 5-2

Setting Up Sign-On Audit, page 5-3

Sign-On Audit Reports, page 5-5

Signon Audit Users Report

Use this report to view who signs on and for how long.

> **Important:** You can only generate Signon Audit Users Reports for those users you are auditing.

Report Parameters

Sort By

Sort the information in your report by the time users start or finish using an application username, the operating system login name of the user, the terminal that users are on, and/or the application username.

Login Name

Search for information about a specific login name that meets your other search criteria to make your report as brief as you need. If you leave this parameter blank, your report contains all login names that meet your other search criteria.

User Name

Search for information about a specific application username that meets your other search criteria to make your report as brief as you need. If you leave this parameter blank, your report contains all application usernames that meet your other search criteria.

Terminal Name

Search for information about a specific terminal that meets your other search criteria to make your report as brief as you need. If you leave this parameter blank, your report contains all terminal names that meet your other search criteria.

From Active Date/To Active Date

You can search for information about users logged into Oracle E-Business Suite within a

specific time period and that meet your other search criteria. Use these parameters to specify the start and end of your time period. If you leave these parameters blank, your report contains user information from the first date that also meets your other search criteria to the current date.

Report Heading

The report heading displays the search criteria you entered as parameter values.

Column Headings

Session Number

The Oracle E-Business Suite session number that uniquely identifies each application user sign-on.

User Name

The Oracle E-Business Suite username of the user who signed on. Use the Users form to view detailed information about an application user. See: Users, page 4-23.

Login Name

The operating system login name of the user who signed on.

Terminal Name

The operating system ID of the terminal from which the user signed on.

Start Active Time/End Active Time

The dates and times when the user signed on and off from Oracle E-Business Suite. The start active time and end active time display only if you audited the user at the user Sign-On Audit level.

Oracle Process

The Oracle database process ID used during the user's sign-on. Consult your database administrator for more information concerning Oracle processes.

System Process

The operating system process ID used during the user's sign-on. Consult your operating system administrator for more information concerning your operating system process ID.

Related Topics

Overview of User and Data Auditing, page 5-1

Auditing User Activity, page 5-2

Setting Up Sign-On Audit, page 5-3

Sign-On Audit Reports, page 5-5

Monitor Users field help, page 5-21

Purge Signon Audit Data Program

Use this program to purge Sign-On Audit information created before a specified date.

The following data is deleted:

- Data for who signs on and for how long
- Data for who is selecting what responsibility and when they do it
- Data for who uses which forms in an application and when

Parameters

Audit Date

The Sign-On Audit information creation date. This program will delete all Sign-On Audit information created before this date.

Database Connection Tagging

The Database Connection Tagging feature utilizes several Oracle Database session attributes that allow applications to record the current use to which a database connection is being put.

Usage

The CLIENT_IDENTIFIER, MODULE, and ACTION columns of the V$SESSION database table are used to track user and application context. These columns are populated as follows:

- **CLIENT_IDENTIFIER** - The client for a particular database session. The value allows the end user of that database connection to be identified. For context-insensitive standalone modules such as FNDLOAD or FNDCPASS, the value of CLIENT_IDENTIFIER is set to 'SYSADMIN'.

- **MODULE** - Name of the currently executing module. The value indicates the application code where the database workload originates, and consequently allows identification of the specific application code (such as a user interface, program, or web service) that is currently using the connection.

- **ACTION** - Name and context of the currently executing business action; for example, a payroll task being undertaken by a particular responsibility.

Management

The Database Connection Tagging feature is controlled via the profile option FND_CONNECTION_TAGGING. Possible settings are 'Enabled' and 'Disabled'.

By default, the profile option value is set to 'Enabled', so Oracle E-Business Suite database connections are tagged with the information described in the previous section. If the feature is disabled, database connections will not be tagged and no information will be collected.

6

Single Sign-On Integration

Overview of Single Sign-On Integration

This chapter is intended to provide guidance for those planning to deploy or integrate Oracle E-Business Suite Release 12.2 in an enterprise single sign-on environment. Aimed primarily at project managers, DBAs, and system administrators, it describes how to integrate Oracle E-Business Suite Release 12.2 with the appropriate supporting products to provide an enterprise-wide single sign-on solution..

> **Important:** Integration is a complex subject, with different components and sequences of steps being needed to accommodate different requirements. The integration options described require an understanding of the relevant products and deployment options.

In the context of this chapter, the term *integration* is used to refer to two closely related (but distinct) aspects of optionally deploying Oracle E-Business Suite in an enterprise-level single sign-on environment.

- Integration with a single sign-on product such as Oracle Access Manager (OAM)
- Integration with Oracle Internet Directory (an LDAP directory) (OID)

> **Note:** All occurrences of the acronym "OAM" in this chapter are references to the Oracle Access Manager product, and not to the completely unrelated Oracle Applications Manager product.

There is a mutual dependency: authenticating against LDAP (OID) requires use of a single sign-on product (OAM), and deploying OAM requires the user population to be synchronized with OID.

Oracle Access Manager

Oracle Access Manager 11g is the preferred Oracle single sign-on product for use with

Oracle E-Business Suite Release 12.2. OAM also requires the use of *Oracle E-Business Suite AccessGate*, a Java Enterprise Edition application that maps a single sign-on user to an Oracle E-Business Suite user, and creates the Oracle E-Business Suite session for that user.

As the single sign-solution for Oracle Fusion Middleware, Oracle Access Manager deals with user *authentication* (validating the user's identity). In contrast, user *authorization* (controlling what the user can access) is handled by Oracle E-Business Suite itself.

Oracle Internet Directory

Oracle Internet Directory is needed to link the namespaces (user information repositories) employed by Oracle Access Manager and Oracle E-Business Suite respectively. Linking the namespaces ensures that a particular user logging in via Oracle Access Manager is the same user that is represented within Oracle E-Business Suite's own FND_USER repository. The linking is done by associating externally-managed Oracle Access Manager users with internally-managed Oracle E-Business Suite users via Global Unique Identifiers (GUIDs). These GUIDs are generated by Oracle Internet Directory, and the associated mapping functions are specific to it.

OID is also required for another reason. While most usage of Oracle E-Business Suite is within an organization (such as a manufacturing company), certain application modules such as iRecruitment need to be available to outside users without accounts having to be created manually and responsibilities assigned. This means application modules that support self-registration must create user accounts *synchronously* (in Oracle E-Business Suite and the external directory at the same time) and on demand. Oracle E-Business Suite uses specific Oracle Internet Directory function calls to handle these synchronous account creation tasks.

Combining Oracle Access Manager and Oracle Internet Directory

Together, OAM and OID enable an Oracle E-Business Suite Release 12.2 environment to provide the following enterprise single sign-on features:

- Users can access multiple Oracle E-Business Suite Release 12.2 instances (or a mixture of Oracle E-Business Suite Release 12.2 and other single sign-on enabled applications) by logging in only once (single sign-on)
- A user who logs out of one SSO-enabled application is logged out of all others as well. This is sometimes called single sign-out, and improves security.
- Administrators and users can perform user management activities, such as account creation, deletion, at enterprise level.

Oracle Identity Manager

A further optional integration option is provided by *Oracle Identity Manager* (OIM), which provisions users from a central repository to other repositories such as OID, third-party LDAPs such as Microsoft Active Directory, or non-LDAP repositories such as Oracle E-Business Suite's FND_USER. OIM can be used independently of OAM and

OID, or in conjunction with both.

Oracle Identity Manager includes *Connectors* for numerous *target systems* (IT resources) in an organization. For example, OIM provides Connectors for Oracle E-Business Suite that enable provisioning of users to the FND_USER table and the HRMS tables used by Trading Communities Architecture (TCA).

Introduction to Enterprise User Management

In large organizations, users often have a large number of userids for a variety of network-based resources such as corporate websites and custom applications. As the number of available resources grow, users and security administrators are faced with the increasingly difficult challenge of managing a proliferation of userids and passwords across different systems.

Enterprise identity management solutions allow security administrators to define a user in a single location such as an LDAP (Lightweight Directory Access Protocol) directory, and share that common user definition throughout multiple parts of their enterprise.

Oracle Identity Management may be integrated with Oracle E-Business Suite to support centralized user management via Oracle Internet Directory, and to support single sign-on functionality via Oracle Access Manager.

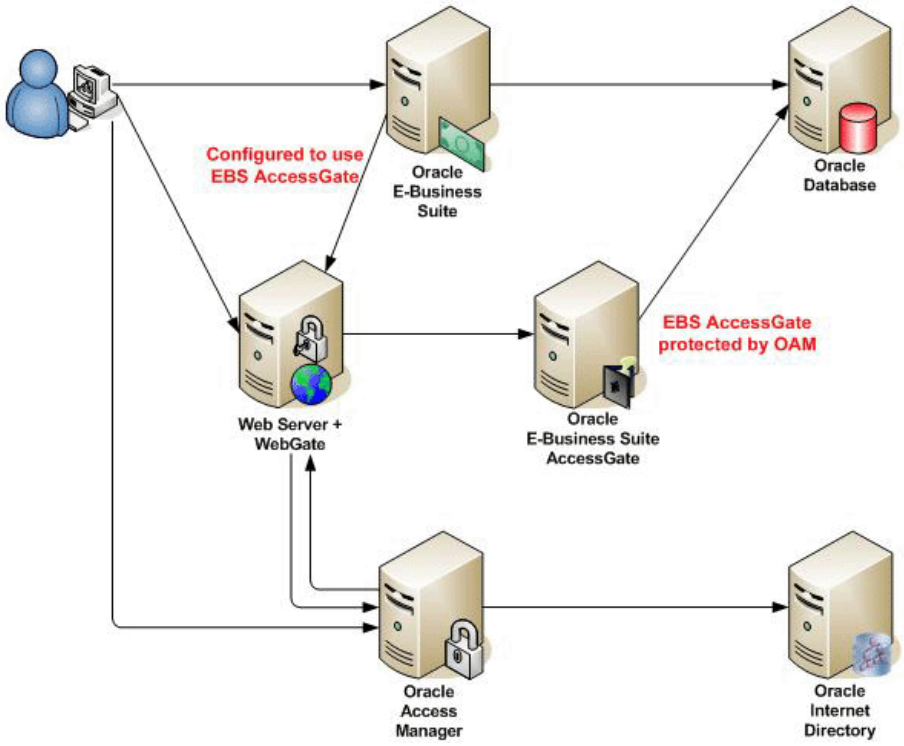
In its default configuration, Oracle E-Business Suite Release 12.2 allows registered users to log in using credentials stored directly in Oracle E-Business Suite. In this default configuration, Oracle E-Business Suite system administrators are responsible for maintaining the local repository of registered Oracle E-Business Suite users.

When optionally integrated with the Oracle Identity Management Suite (which includes OAM and OID), Oracle E-Business Suite system administrators can reconfigure their environments to delegate both user administration and user authentication to Oracle Access Manager. This integration requires significant changes to how Oracle E-Business Suite Release 12.2 handles authentication. Instead of performing authentication natively, via the local Oracle E-Business Suite FND_USER table, Oracle E-Business Suite Release 12.2 now delegates this functionality to Oracle Access Manager, via Oracle E-Business Suite AccessGate. In this configuration, Oracle E-Business Suite Release 12.2 can direct unauthenticated users to Oracle Access Manager for identity verification, and securely accept identities vouched for by the single sign-on mechanism.

Oracle Access Manager may, in turn, be integrated with existing third-party authentication systems such as Microsoft Windows (Kerberos), and Oracle Internet Directory may be integrated with existing third-party LDAP directories such as Microsoft Active Directory.

> **Note:** Oracle Access Manager always performs authentication against information stored in Oracle Internet Directory, even if a third-party authentication mechanism is in use.

The following diagram illustrates the high-level structure of a typical integration.



Note: Where a third-party authentication mechanism is in use, Oracle Access Manager and Oracle Internet Directory are still required: they provide bridge functionality between Oracle E-Business Suite and the third-party single sign-on solution.

Integration Actions and Options

Integration of Oracle E-Business Suite with Oracle Access Manager is achieved via the *OAM WebGate agent*, which is used in conjunction with Oracle E-Business Suite AccessGate.

Note: Each Oracle E-Business Suite instance requires its own deployment of the Oracle E-Business Suite AccessGate application.

Note: For a detailed description of the role of agents in Oracle Access Manager 11g, refer to Chapter 5 in *Oracle Fusion Middleware Administrator's Guide for Oracle Access Manager 11g Release 1*.

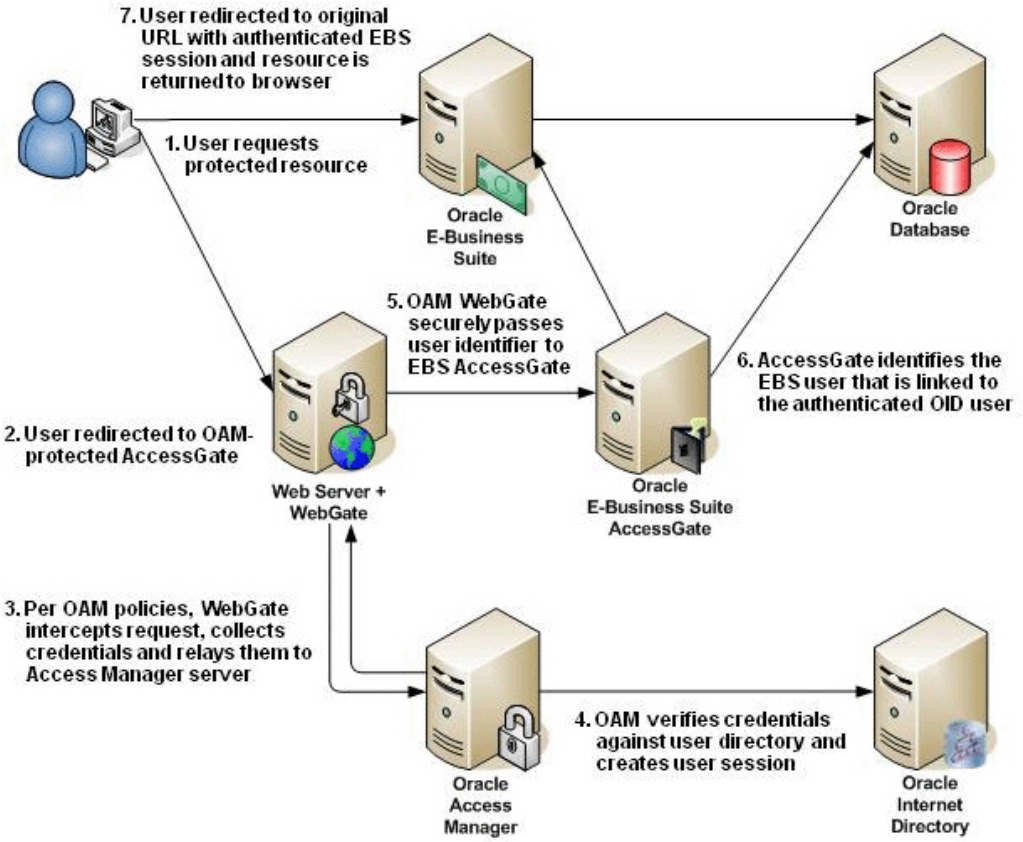
The rest of this section focuses on the details of integration using WebGate and Oracle E-Business Suite AccessGate.

When an unauthenticated user attempts to access a protected Oracle E-Business Suite resource, the user is directed to the Oracle E-Business Suite AccessGate application. This application is protected by the Oracle Access Manager server, so the authentication request is rerouted to a separate HTTP Server on which *Oracle Access Manager WebGate* is installed. This is a component of Oracle Access Manager that intercepts HTTP requests and redirects them to the Oracle Access Manager server to determine if and how the resources are allowed to be accessed, and to authenticate the current user if authentication is required. If Oracle Access Manager is already deployed in the environment, an existing WebGate can be configured for this purpose.

> **Warning:** Oracle E-Business Suite AccessGate must be deployed to a separate WebLogic Server instance and domain from Oracle E-Business Suite Release 12.2. It will not work if deployed on the same server, or another server in the same domain.

After a user is initially authenticated by Oracle Access Manager, the request for a resource and credentials returned by Oracle Access Manager server are picked up by Oracle E-Business Suite AccessGate. If the credentials are valid, Oracle Access Manager connects to the Oracle E-Business Suite database to link the Oracle Internet Directory (OID) user to an Oracle E-Business Suite user. If Oracle E-Business Suite fails to identify a linked user for the OID user, the user is redirected to the linking page so that he can map his unlinked OID user account to his Oracle E-Business Suite username. After this mapping is done, the originally requested resource is returned with a valid authenticated Oracle E-Business Suite user session. All subsequent requests for Oracle E-Business Suite resources are then returned directly to the user, for as long as the user session remains valid.

The sequence of actions is illustrated in the following diagram.



Oracle E-Business Suite AccessGate must be installed in the same Internet domain (for example, company.com) as the Oracle E-Business Suite application tier servers. This is because several Oracle E-Business Suite domain cookies are shared between the application tier servers and the Oracle E-Business Suite AccessGate server.

Advanced Options and Configurations

Various options exist for extending the basic integration of Oracle E-Business Suite with Oracle Access Manager. These include using multiple WebGates for load balancing, enabling SSL communication between the nodes, and configuring one or more nodes in a DMZ or with a reverse proxy. This subsection will briefly mention any special Oracle E-Business Suite integration steps that may need to be taken for different Oracle Access Manager configurations.

Deploying Oracle E-Business Suite AccessGate in an SSL-enabled Environment

In production environments, it is advisable to use SSL on both the Oracle E-Business Suite application tier and on the WebLogic Server instance where the Oracle E-Business Suite AccessGate is deployed. Also, the WebGate plugin should be deployed on an HTTP server that is secured using SSL.

An important related point is that Oracle E-Business Suite application tiers and

WebLogic Server instances must all be configured to use the same protocol (either http or https). If the relevant nodes are configured to use SSL (https), the HTTP server on which WebGate is installed must also be configured to use SSL. If, however, the HTTP server running WebGate is configured to use SSL, it is not necessary to configure SSL on the Oracle E-Business Suite application tiers or WebLogic Server instances.

There are other considerations that apply to SSL environments. For further information, refer to the following resources:

- Steps to enable SSL communication for the Oracle Access Manager components: *Oracle Fusion Middleware Administrator's Guide for Oracle Access Manager 11g Release 1*
- Oracle WebLogic Server configuration steps: *Configuring SSL* chapter in *Oracle Fusion Middleware: Securing Oracle WebLogic Server 11g Release 1*
- Oracle HTTP Server: *Configuring SSL* chapter in *Oracle Fusion Middleware* section of *Oracle Fusion Middleware Administrator's Guide 11g Release 1*

Deploying Oracle E-Business Suite AccessGate with an Oracle RAC Database

A database instance configured to use Oracle RAC load balancing requires creation of either a JDBC multidata source or Active GridLink for Oracle RAC. This can be done using the Oracle WebLogic Server Administration Console.

Once the multidata source has been created, an automated deployment script (*txkEBSAuth.xml*) can be run with the appropriate options.

This script performs two major functions, which can either be performed independently or together in a single execution. These functions are:

- Create a connection pool and data source for Oracle E-Business Suite AccessGate
- Deploy the Oracle E-Business Suite AccessGate war file with a customized application deployment plan

For more information, refer to:

- *Configuring JDBC Multi Data Sources* chapter in *Configuring and Managing JDBC for Oracle WebLogic Server*.

Deploying Oracle E-Business Suite AccessGate in a DMZ

When configuring single sign-on using Oracle Access Manager and Oracle E-Business Suite AccessGate in a DMZ, specific firewall ports must be opened to allow additional connections from the end user to Oracle E-Business Suite AccessGate, and to allow communication between WebGate (located in the DMZ) and Oracle Access Server (located on the internal network). In addition, some of the values needed to configure Oracle E-Business Suite AccessGate will need to be modified to point back to a reverse proxy. There is no need to open the ports for LDAP or LDAPS connections from the DMZ to the internal network.

For more information, refer to:

- My Oracle Support Knowledge Document 1375670.1, *Oracle E-Business Suite Release 12.2 Configuration in a DMZ*.

Centralized Logout

When logging out of an application protected by Oracle Access Manager, the host and domain cookies created by Oracle Access Manager are removed, which forces a user to re-authenticate the next time he accesses a protected resource. However, this process does not clean up sessions, or remove cookies specific to partner applications such as Oracle E-Business Suite.

The process of logging out must invalidate all sessions for Oracle Access Manager, WebGate, and Oracle E-Business Suite. To ensure that these cookies are cleared when a single sign-on session is terminated, Oracle Access Manager and Oracle E-Business Suite AccessGate must be configured to initiate a "callback" to a script that will clean up these sessions. Without this, a user who terminates his single sign-on session may still be able to access Oracle E-Business Suite, or even create a new Oracle E-Business Suite session.

To help implement this functionality, a sample script called `samplecleanup.html` is provided. Using this script requires making some site-specific customizations to it, then configuring Oracle E-Business Suite AccessGate to load it. This file will subsequently be invoked whenever a user logs out of SSO, terminating any current sessions the user has for Oracle E-Business Suite instances protected by that WebGate.

The script works with Oracle E-Business Suite AccessGate to perform several actions:

1. Registers logout callbacks for the current Oracle E-Business Suite environment, and any others that are protected by this WebGate, which must be provided. This logout callback is a servlet within Oracle E-Business Suite AccessGate, which:
    1. Destroys the Oracle E-Business Suite session for the instance it is protecting
    2. Removes the ICX cookie
    3. Frees allocated E-Business resources
    4. Returns an image indicating success or failure
2. Executes the logout callbacks one by one, indicating success or failure.
3. Removes any cookies that are listed, such as WebGate 10g cookies, or session-related cookies for any custom applications that are deployed.
4. Redirects the user to a target URL (if one is provided) to control where a user is sent after the logout completes. The value of the target URL which will depend on where the user initiated the logout.

The `samplecleanup.html` script can be used regardless of where a user initiates the

logout request. When logging out of Oracle E-Business Suite, Oracle E-Business Suite AccessGate will handle the logout process for the current instance, and then load this cleanup script on other WebGates (as configured in the script). As the scripts are loaded in parallel, the script also invokes a logout on the Oracle Access Manager server, which expires the Oracle Access Manager session and associated cookies. For external partner applications such as Oracle WebCenter, the logout link must be modified to invoke the cleanup script, which will redirect to the original logout URL or landing page upon completion.

The `samplecleanup.html` script has to be customized on a site-specific basis, as it must explicitly list the locations of every deployed Oracle E-Business Suite AccessGate application that is protected by the WebGate in question. The customized script must then be deployed to the server where WebGate is installed, so the Oracle Access Manager Server can load it.

Enterprise User Management

Oracle Internet Directory is the integration point that allows Oracle E-Business Suite to participate in enterprise-level user management. Each Oracle E-Business Suite instance must still maintain a record of registered users, in the form of the traditional application accounts. However, the level of abstraction needed for an enterprise level user requires a mechanism that can uniquely identify a user across the enterprise. This is accomplished via a globally unique identifier (GUID). Oracle Internet Directory and Oracle E-Business Suite store GUID information for each enterprise level user; the GUID can be considered as an identity badge that is recognized by both Oracle Internet Directory and Oracle E-Business Suite.

Another requirement in such an environment is for user enrollment to be done only once, at well-defined places, with the user subsequently being known to the rest of the enterprise. Two additional features enable support for automatic propagation of user information across an enterprise:

- A *synchronization* process between Oracle Internet Directory and a third-party LDAP server
- A *provisioning* process between Oracle Internet Directory and Oracle E-Business Suite

Much of the complexity involved with integrating Oracle E-Business Suite into a single sign-on environment arises because of the need to consolidate fragmented or duplicated user data in the single sign-on environment, as a legacy of integrating previously isolated systems. The solution described in this document provides mechanisms to link the existing data together using the GUID. In addition, bulk migration tools are provided to move a large number of users between Oracle Internet Directory and Oracle E-Business Suite during the transition to a single sign-on environment.

Additional Single-Sign on Features, Limitations, and Known Issues

Advanced features include automatically keeping a set of user profile information synchronized across an enterprise for an entity, and the ability to link an account in Oracle Internet Directory to multiple application accounts in Oracle E-Business Suite.

In this release, provisioning from Oracle E-Business Suite to Oracle Internet Directory is synchronous: that is, all user management operations carried out in Oracle E-Business Suite are also carried out in Oracle Internet Directory. However, provisioning from Oracle Internet Directory to Oracle E-Business Suite is done asynchronously.

The solution described here does not address the issue of *authorization*. After a user has been authenticated, Oracle E-Business Suite retrieves from the relevant FND tables the authorization information associated with the application account the user is logged into. Authorization information for application accounts is managed through application responsibilities. Oracle E-Business Suite applies authorization checks as and when required during the user's session.

Key Identity Management Configuration Options

Configuration Option:	Possible Settings:	Configured Via:
Initial Source of User Information	• Oracle E-Business Suite. • Oracle Internet Directory. • Third-Party LDAP Directory. • A combination of the above.	Execution of manual initial provisioning steps (described later).
Master Source of Truth for Updates to User Information	• Oracle E-Business Suite. • Oracle Internet Directory. • Third-Party LDAP Directory. • A combination of the above.	Provisioning profile selected for Directory Integration and Provisioning Platform (described later).

Configuration Option:	Possible Settings:	Configured Via:
New Userids Created in Oracle Internet Directory	• Are automatically created in Oracle E-Business Suite with subscriptions for user attribute updates. • Have manually-created equivalent userids in Oracle E-Business Suite, and are manually linked by the end-user at the time of first logon. • Have manually-created equivalent userids in Oracle E-Business Suite, and are automatically linked at the time of first logon. • Are automatically created in a third-party LDAP directory, combined with either of the two above options.	Related Oracle E-Business Suite Profile Options: • APPS_SSO_OID_IDENTITY • APPS_SSO_AUTO_LINK_USER • APPS_SSO_LINK_SAME_NAMES

Configuration Option:	Possible Settings:	Configured Via:
New Userids Created in Oracle E-Business Suite	• Are automatically created in Oracle Internet Directory with subscriptions for user attribute updates. • Have manually-created equivalent userids in Oracle Internet Directory, and are manually linked by the end-user at the time of first logon. • Have manually-created equivalent userids in Oracle Internet Directory, and are automatically linked at the time of first logon.	Related Oracle E-Business Suite Profile Options: • APPS_SSO_LDAP_SYNC • APPS_SSO_AUTO_LINK _USER
Specific Oracle E-Business Suite Userids	• Log in to Oracle E-Business Suite via Oracle Access Manager. • Log in to Oracle E-Business Suite directly. • Both of the above.	APPS_SSO_LOCAL_LOGIN profile option
All Oracle Internet Directory Userids	• Are linked to a single Oracle E-Business Suite userid. • Are linked to multiple Oracle E-Business Suite accounts.	APPS_SSO_ALLOW_MULTI PLE_ ACCOUNTS profile option

The above list of identity management configuration options is not exhaustive.

Deployment Scenario 0: E-Business Suite + SSO and OID

This section explains the technical details and deployment steps using a simplified deployment scenario, where an existing Oracle E-Business Suite instance is integrated

with a fresh Oracle Access Manager/Oracle Internet Directory infrastructure. Although many real world deployments are likely to be more complex, this scenario serves to illustrate the core concepts and procedures of the integration effort. In later sections, we build on this basic scenario to describe more sophisticated situations such as the existence of a third-party single sign-on solution, or the presence of multiple user repositories. The goal is not to describe every conceivable deployment variation, but rather to provide a number of representative cases from which implementers can intelligently derive the exact steps needed for their particular requirements.

Starting Point

This scenario presumes that:

- Oracle E-Business Suite Release 12.2 has been installed and has an existing user population
- Oracle Access Manager, Oracle E-Business Suite AccessGate, and Oracle Internet Directory have all been installed (on a separate machine) in accordance with the appropriate instructions
- Oracle Internet Directory has no currently existing users, apart from pre-seeded users

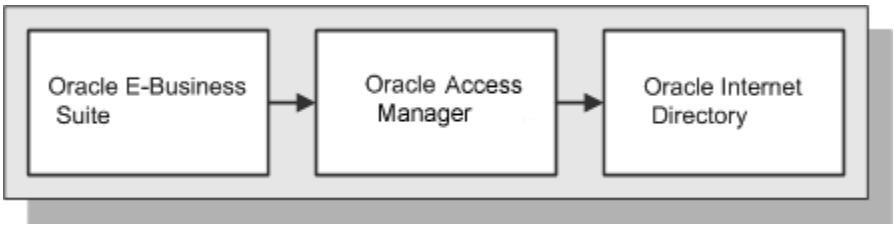
The requirement is to integrate Oracle E-Business Suite Release 12.2 with Oracle Access Manager, Oracle E-Business Suite AccessGate, and Oracle Internet Directory.

Solution Outline

The results of implementing this solution will be that:

- Oracle E-Business Suite will delegate user sign-on and authentication to Oracle Access Manager
- Oracle Access Manager will authenticate user credentials against user entries in Oracle Internet Directory
- Oracle Internet Directory will contain the account ID and password for every user that is configured for single sign-on.

> **Warning:** For security reasons, local users and standard administrative accounts such as SYSADMIN should never be configured for single sign-on.



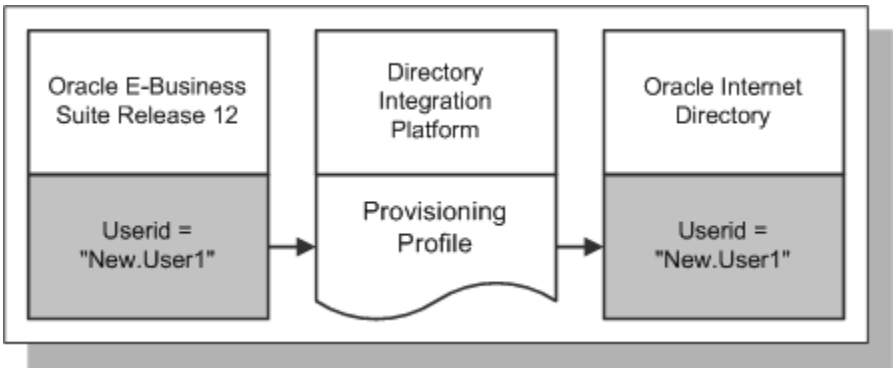
User Management Options

Oracle E-Business Suite Release 12.2 maintains a local cache of user information in its existing user directory (FND_USER). The Oracle E-Business Suite User Bulk Migration Tool can be used to migrate existing Oracle E-Business Suite application accounts to single sign-on accounts in Oracle Internet Directory. After the migration, a system administrator has a number of user management options about the locations where the user information is created, and where it is provisioned (sent).

Option 1: Provision Oracle E-Business Suite Users to Oracle Internet Directory

All user information is created in Oracle E-Business Suite, then provisioned into Oracle Internet Directory: Oracle E-Business Suite is configured as a *provisioning integrated application* with Oracle Internet Directory. System administrators configure the provisioning integration via Directory Integration and Provisioning (DIP) *provisioning profiles*. A DIP server synchronizes policy changes in the directory with connected databases, using a separate DIP provisioning profile for each database.

The creation of a new application account in Oracle E-Business Suite will automatically trigger the creation of a new single sign-on account in Oracle Internet Directory. Some of the user attributes from the application account may be provisioned in the single sign-on account in Oracle Internet Directory during account creation.

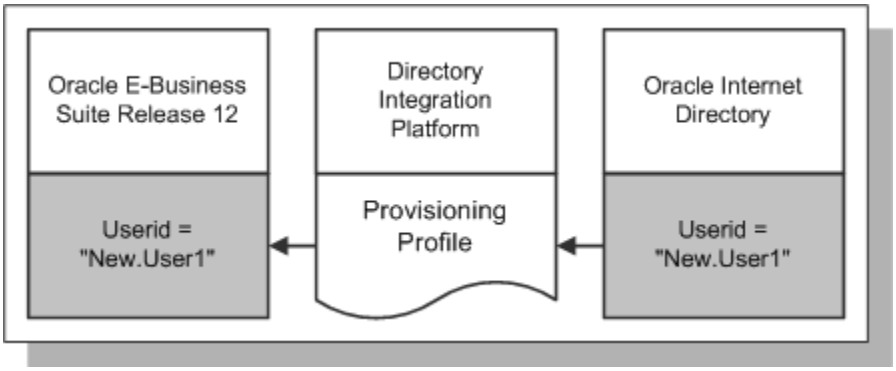


Option 2: Provision Oracle Internet Directory Users to Oracle E-Business Suite

All user information is created in Oracle Internet Directory, then provisioned into

Oracle E-Business Suite. Oracle E-Business Suite is configured as a provisioning integrated application with Oracle Internet Directory.

System administrators configure the provisioning integration via provisioning profiles: the creation of a new single sign-on account in Oracle Internet Directory will automatically trigger the creation of a new application account in Oracle E-Business Suite. Some of the user attributes from the single sign-on account may be provisioned in the application account in Oracle Internet Directory during account creation.

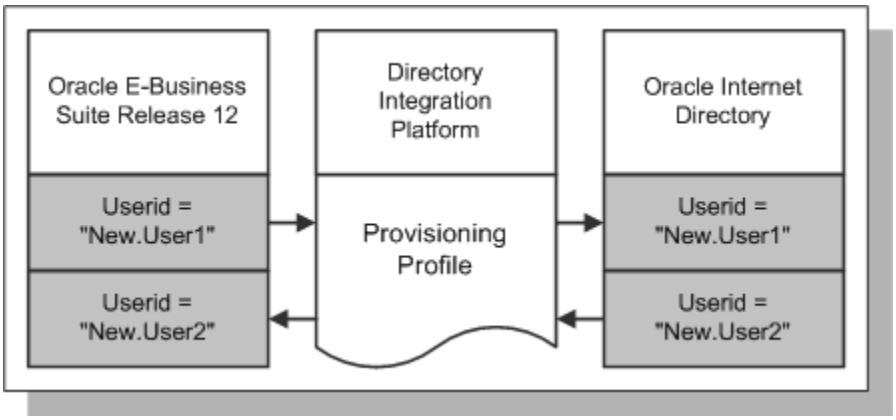


Option 3: Bidirectional Provisioning Between E-Business Suite and Oracle Internet Directory

All user information is created in either Oracle Internet Directory or Oracle E-Business Suite, then provisioned into the other system. Oracle E-Business Suite is configured as a provisioning integrated application with Oracle Internet Directory. System administrators configure the provisioning integration via provisioning profiles.

The creation of a new application account in Oracle E-Business Suite will automatically trigger the creation of a new single sign-on account in Oracle Internet Directory. The creation of a new single sign-on account in Oracle Internet Directory will automatically trigger the creation of a new application account in Oracle E-Business Suite.

During account creation, some of the user attributes from the application account may be provisioned in the single sign-on account in Oracle Internet Directory, and some of the user attributes from the single sign-on account may be provisioned in the application account in Oracle Internet Directory.



Synchronizing User Attributes

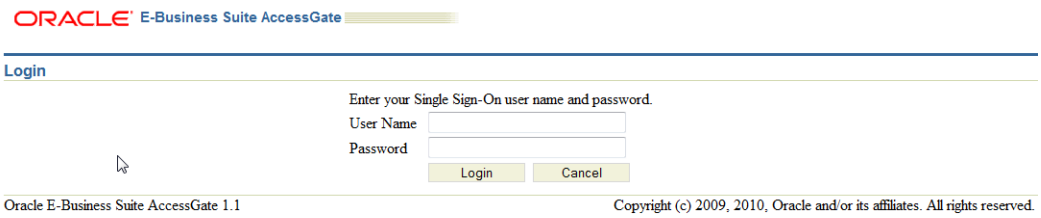
For all three options above, a predefined set of user attributes is synchronized between Oracle E-Business Suite and Oracle Internet Directory. Currently-supported attributes are listed later in the "Supported Attributes" section.

End-User Experience

This section describes the user's perception of the single sign-on environment.

Single Sign-On User Experience

On attempting to access an Oracle E-Business Suite environment, a user who has not yet been authenticated with Oracle Access Manager is directed to a single sign-on login page:



After authentication via Oracle Access Manager (or if authentication has previously been carried out) the user is redirected to the requested page or the user's home page in Oracle E-Business Suite Release 12.2.

Sign-Out User Experience

When a user logs out of an Oracle E-Business Suite instance, he is also logged out of Oracle Access Manager, as well as any other applications (partner applications) that have been integrated with Oracle Access Manager and have been accessed via Oracle Access Manager.

Single Sign-On Authentication Flow

The user attempts to access the Oracle E-Business Suite Release 12.2 instance, and Oracle E-Business Suite looks for a session cookie. If the cookie is found and validated, the user is directed to the requested application page, and the rest of the steps shown here are skipped.

If the session cookie is not found, Oracle E-Business Suite redirects the user to an Oracle E-Business Suite AccessGate URL in Oracle Access Manager. This URL is intercepted by OAM WebGate, which obtains policy information from OAM and then checks for a valid OAM session cookie in the user's browser. If one does not exist, it redirects the user to the credential collection page, which is an unprotected page in AccessGate. The credentials are submitted to OAM, which looks for an Oracle Single Sign-On security cookie in the user's browser. If the Oracle Single Sign-On security cookie is not found, the user must log into a valid account via Oracle Access Manager before authentication can proceed further.

Next, Oracle Access Manager contacts Oracle Internet Directory and authenticates the user's credentials against the list of registered users in Oracle Internet Directory. After successful authentication, Oracle Access Manager sets an OAM session cookie in the user's browser, and retrieves user attributes for the single sign-on account from Oracle Internet Directory.

Once the credentials are verified, OAM returns the user to the URL in Oracle E-Business Suite AccessGate. It passes a request that includes HTTP response headers containing a user identifier and the GUID.Oracle E-Business Suite verifies the URL token, locates the application user and creates an application session and corresponding cookie, based upon the user's assigned application responsibilities and roles. This process entrusts the process of user authentication to Oracle Access Manager, and user authorization to Oracle E-Business Suite. Oracle E-Business Suite then redirects the user to the requested application page, or the user's home page.

Session Timeout Behavior

When both the application session and the single sign-on session timeout, the user will be directed to the single sign-on login page to re-authenticate. After a successful re-authentication, the user will be redirected back to Oracle E-Business Suite. The application page the user sees depends on the application technology stack in use; see table below.

Currently, when the application session has expired, but not the single sign-on session, the user will be directed to Oracle E-Business Suite AccessGate, and then back to Oracle E-Business Suite Release 12.2, without being prompted to re-authenticate. Depending on the technology stack in use at the time when the session timeout occurred, the user will then see one of the following pages listed in the table below.

Technology Stack	Session Timeout Behavior
Oracle Application Framework	Application home page
CRM	If the current request on detection of application session expiration was a 'GET', the user sees the requested page. If the current request was a 'POST', the user sees the posting page without the post having been performed.
Forms	A series of pop-up windows will appear, leading the user to the Oracle Access Manager login page for re-authentication.

When an application session is terminated because the maximum valid period has been reached, or because of a period of user inactivity, Oracle E-Business Suite redirects the user to Oracle Access Manager for re-authentication. Oracle Access Manager checks the single sign-on cookie; if it is still valid, the user is redirected back to Oracle E-Business Suite Release 12.2. If the single sign-on cookie has expired as well, Oracle Access Manager requires the user to authenticate again before redirecting him back to Oracle E-Business Suite Release 12.2.

The application session timeout value takes precedence over the Oracle Access Manager timeout settings. For example, until an application session times out (or the user explicitly logs out), a user may continue to access the partner application even if his Oracle Access Manager security cookie has expired. We therefore recommend setting Oracle E-Business Suite's application session timeout value to be equal to, or less than, that of the Oracle Access Manager server.

User Management Options

This section describes the various options for management of users in a single sign-on environment.

Local Access to Oracle E-Business Suite

Selected users can be permitted to log in to the application directly, i.e. without going through the single sign-on process. This allows users such as the system administrator to troubleshoot a configuration when Oracle Access Manager is not functioning correctly, or is unavailable. Such local users can now log into the application directly via the applications login page, AppsLocalLogin.jsp. The supplied SYSADMIN account is configured to have local access. In addition, the SYSADMIN account can control which additional users (if any) are permitted to have local access to the Oracle E-Business Suite; this is accomplished via the Applications SSO Login Types (`APPS_SSO_LOCAL_LOGIN`) profile option.

Important: Generic accounts, especially those with administrative rights (such as SYSADMIN) should always be local only.

Identifying a User Across the Enterprise

After Oracle Access Manager integration is complete, user information exists in two places: Oracle Internet Directory and Oracle E-Business Suite Release 12.2.

This shared information has the following characteristics:

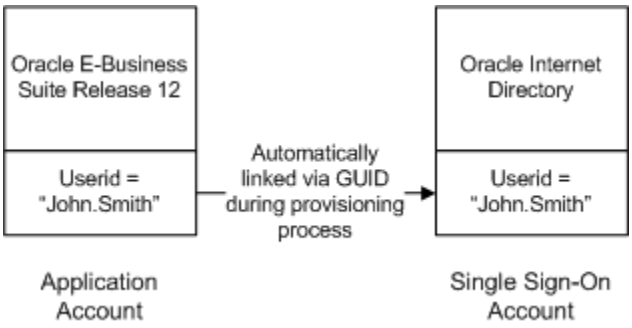
- A GUID uniquely identifies a user across multiple systems.
- Both Oracle Internet Directory and Oracle E-Business Suite store GUID information for each single sign-on user.
- During the authentication handshake between Oracle Internet Directory and Oracle E-Business Suite, Oracle Access Manager passes the authenticated user information in the form of GUID to Oracle E-Business Suite AccessGate, which then uses the GUID to locate the corresponding application account.
- Once a GUID is generated and stored in both a single sign-on account in Oracle Internet Directory and an application account in Oracle E-Business Suite, the two accounts are said to be linked.
- A number of processes are used to establish this link. The most commonly used ones are explained below, and the rest in the more advanced deployment scenarios later in this section.

Bulk Migration of Users

Tools are provided to migrate existing users in bulk between Oracle Internet Directory and Oracle E-Business Suite. Both Oracle Internet Directory and Oracle E-Business Suite provide command line utilities to export and import users via flat text files in LDIF format.

User Provisioning Between Oracle E-Business Suite and Oracle Internet Directory

New users created on either system can be provisioned into the other via the provisioning process. The provisioning system consists of components of both Oracle Internet Directory and Oracle E-Business Suite that queue user events on each system, plus an Oracle Internet Directory process that periodically pushes or pulls these events to or from Oracle E-Business Suite. The provisioning process establishes the GUID link for provisioned accounts. During this process, single sign-on accounts are automatically linked to Oracle E-Business Suite application accounts.



Provisioning has the following characteristics:

- Once linked, user changes from either system can be provisioned into the other.
- The provisioning process between Oracle Internet Directory and each Oracle E-Business Suite instance is determined by a provisioning profile.
- The provisioning profile controls which user events are provisioned, the direction of provisioning, and the user attributes included in each event.
- Oracle E-Business Suite is said to be a provisioning integrated application with Oracle Internet Directory when a provisioning profile is created for it.

Refer to the "Supported Attributes" section for information on which attributes can be provisioned between the systems, and "Configuring Directory Integration Platform Provisioning Templates" for more details on the provisioning process.

Strategies for User Management

At the start of the deployment, Oracle E-Business Suite Release 12 is the sole repository of user information. Users who will need to access Oracle E-Business Suite via Oracle Access Manager must already exist or be created in Oracle Internet Directory.

For pending users that are enabled in Oracle E-Business Suite after user creation, the IDENTITY_MODIFY event from Oracle E-Business Suite to Oracle Internet Directory must be enabled.

> **Note:** Refer to "Configuring Directory Integration Platform Provisioning Templates" for more details.

Populating Oracle Internet Directory with Existing Oracle E-Business Suite Users

Existing Oracle E-Business Suite users can be migrated into Oracle Internet Directory by means of the bulk migration tool (see "Migrating Data between Oracle E-Business Suite Release 12 and Oracle Internet Directory" for details).

Creating New Users

After the initial migration, you may choose to allow new users to be created either from Oracle Internet Directory or from Oracle E-Business Suite, and then provision them into the other system. This is achieved by enabling either the SUBSCRIPTION_ADD event from Oracle Internet Directory to Oracle E-Business Suite, or the IDENTITY_ADD event from Oracle E-Business Suite to Oracle Internet Directory. Refer to "Configuring Directory Integration Platform Provisioning Templates" for more details.

Bidirectional Provisioning

Alternatively, you may choose to create new users from either Oracle Internet Directory or Oracle E-Business Suite, and then provision them into the other system. This is achieved by enabling both the SUBSCRIPTION_ADD event from Oracle Internet Directory to Oracle E-Business Suite, and the IDENTITY_ADD event from Oracle E-Business Suite to Oracle Internet Directory. Refer to "Configuring Directory Integration Platform Provisioning Templates" for more details.

Bidirectional provisioning requires careful planning, and the following restrictions must be considered:

- The provisioning process from Oracle Internet Directory to Oracle E-Business Suite is *asynchronous*. In contrast, the provisioning process from Oracle E-Business Suite to Oracle Internet Directory is *synchronous*.

- Whether new users are created in either Oracle Internet Directory or Oracle E-Business Suite, they must be granted the appropriate roles or responsibilities via Oracle E-Business Suite User Management in order to access application functionality.

- The provisioning events will fail if, for example, a user with the same username has been created concurrently on the other system, or some aspect of the user's profile does not meet the policy set on the other system.As there is no mechanism to roll back the original change on the system that triggered the event, the failure can put the entire system into an unstable state. It is therefore essential to coordinate the account policy on all the systems involved, and place appropriate safeguards on the user creation process. For example, usernames created directly on one system need to be chosen in the context of names used across the single sign-on environment.

Updating User Information

User information stored in Oracle Internet Directory single sign-on accounts is generally managed independently of user information stored in Oracle E-Business Suite Release 12.2 application accounts.

System administrators must decide:

- Which user attributes are to be provisioned between an Oracle E-Business Suite Release 12.2 instance and Oracle Internet Directory.

- Which system is to be the master "source of truth" for a given attribute. This determines the provisioning direction for that attribute.

System administrators then enable the IDENTITY_MODIFY events in the appropriate direction with the appropriate attribute list. Please refer to "Configuring Directory Integration Platform Provisioning Templates" for more details.

Note the following current restrictions:

- Updates to email ID in Oracle Internet Directory are not correctly reflected in the E-Business Suite (HZ_CONTACT_POINTS in TCA) unless the PERSON_PARTY_ID foreign key in the FND_USER table has been defined. Furthermore, if PERSON_PARTY_ID is changed, because a user is linked to another person in TCA, information stored in OID can overwrite this other person's information during provisioning.

- Provisioning from Trading Community Architecture (TCA) to Oracle Internet Directory is not supported.

- Provisioning of data from Oracle Human Resources to Oracle Internet Directory is supported via the Oracle Human Resources Agent, which is released as part of the Oracle Internet Directory suite of utilities. Note that the Oracle Human Resources Agent supplied with Oracle Internet Directory is unidirectional. That is, it ensures that Oracle Internet Directory is synchronized with HR, so that changes to user data in HR cause the corresponding data to be updated in Oracle Internet Directory. However, if changes are made to user data in Oracle Internet Directory, the HR connector does not synchronize these changes back to HR. A bidirectional connector is planned for a future build.

Terminating and End-Dating Users

Dates are not synchronized between Oracle Internet Directory and Oracle E-Business Suite. However, the provisioning process may be set up so that when a single sign-on account in Oracle Internet Directory is deleted, the associated Oracle E-Business Suite application accounts is end-dated. This is accomplished in the provisioning profile, by enabling the IDENTITY_DELETE event from Oracle Internet Directory to Oracle E-Business Suite.

> **Note:** Refer to"Configuring Directory Integration Platform Provisioning Templates" for details.

Subject to organizational security and audit policies, it may be preferable to disable single sign-on accounts in Oracle Internet Directory rather than delete them, since this allows an applications account to be re-enabled at a later date as required. This can be particularly useful in the case of contractors who may leave and rejoin.

> **Note:** See "Enabling/Disabling Users" for more information on enabling/disabling users.

Password Management

One of the major objectives of single sign-on integration is centralized user password management using Oracle Internet Directory, which provides the following features:

- Accessing Oracle E-Business Suite via Oracle Access Manager does not require passwords in the Oracle E-Business Suite; the password stored in Oracle Internet Directory is sufficient for authentication.

- The password for an application account in Oracle E-Business Suite Release 12.2 is replaced with the reserved keyword 'EXTERNAL', if (as will usually be the case) the only permitted method to access that application account is via Oracle Access Manager

- Password management for such users is carried out entirely in Oracle Internet Directory.

End-User Password Changes

The majority of end users will be able to change their single sign-on passwords using the standard methods provided by Oracle Internet Directory. For example, users may employ Oracle Identity Manager.

System Administrator Password Changes and Resets

To reset single sign-on passwords, an administrator should follow the methods provided by Oracle Internet Directory as detailed in the chapters 'Directory Entries Administration' and 'The Delegated Administration Service', in the Oracle Internet Directory Administrator's Guide.

Password Policies

Oracle Internet Directory is designated as the master user directory for passwords. The user's password creation, modification and Oracle Access Manager login activities are subject to the Oracle Internet Directory rules that govern how passwords are created and used. For example, Oracle Internet Directory system administrators may establish policies for password expiration, minimum length, and alphanumeric mixes. Refer to the 'Password Policies in Oracle Internet Directory' chapter of the Oracle Internet Directory Administrator's Guide, for an explanation of supported password policies.

If the provisioning profile specifies that passwords in application accounts are to be provisioned from Oracle E-Business Suite Release 12.2 to Oracle Internet Directory, Oracle E-Business Suite Release 12.2 password policies must be at least as restrictive as the ones in Oracle Internet Directory. This ensures that passwords can be successfully propagated from Oracle E-Business Suite Release 12.2 to the single sign-on accounts in Oracle Internet Directory.

Note: Passwords stored in Oracle Internet Directory are case sensitive. Mixed case passwords in Oracle E-Business Suite are migrated with the case preserved.

Password Management and Applications SSO Login Types

For users who have been granted local access to Oracle E-Business Suite via the Applications SSO Login Types (APPS_SSO_LOCAL_LOGIN) profile, Oracle E-Business Suite retains the relevant applications account password. This is true even if Oracle Internet Directory or the third-party LDAP directory has been designated as the master user directory for passwords. All existing password-related features in the Oracle E-Business Suite remain the same for local accounts. For example, the user must use the Self-Service change password screen ('Preferences' page) to maintain passwords.

For users who have both single sign-on and local access to Oracle E-Business Suite, local password change in Oracle E-Business Suite can be synchronized to Oracle Internet Directory, if the provisioning profiles are set up accordingly. The reverse direction is not possible, because Oracle Internet Directory only stores the hash of the passwords, not encrypted passwords as Oracle E-Business Suite does.

Because of the potential difficulty of educating users about the special password management considerations that apply to application accounts configured with the Applications SSO Login Types (APPS_SSO_LOCAL_LOGIN) profile, this profile option should, as noted earlier, only be employed for a limited number of system administration or other advanced accounts. The system administrator is required to set the local password using the AFPASSWD utility or FNDCPASS utility, in case user passwords stored only in LDAP (APPS password is set to EXTERNAL) also need to be stored locally in Oracle E-Business Suite.

For more information about the AFPASSWD and FNDCPASS utilities, refer to the *Basic DBA Tasks* chapter of *Oracle E-Business Suite Maintenance Guide*.

Critical Implementation Decisions

1. Oracle Internet Directory has a powerful and flexible set of configuration options. Most Oracle E-Business Suite system and security administrators will be able to use the default Oracle Internet Directory configuration. Security administrators with advanced security requirements may choose to use alternative Oracle Internet Directory configurations.

   Items of particular importance to Oracle E-Business Suite integration include:

   - Identity management realm
   - DIT structure
   - What attribute is chosen as the nickname attribute

- Whether new users are to be created:
  - Only from Oracle Internet Directory
  - Only from Oracle E-Business Suite Release 12.2
  - From both Oracle E-Business Suite and Oracle Internet Directory

2. Whether updates to user information are to be provisioned. If so, what user attributes are to be provisioned, and the direction of provisioning.
3. Which users should only use local access to Oracle E-Business Suite Release 12.2, which users only need access via Oracle Access Manager, and which users need both types of access.
4. Oracle Access Manager settings:
   - Session timeout values for both Oracle E-Business Suite and Oracle Access Manager.
   - Password policy for both Oracle E-Business Suite and Oracle Identity Management.

Implementation Instructions

1. Identify the user population that only need local login access to Oracle E-Business Suite, and set the Applications SSO Login Types (APPS_SSO_LOCAL_LOGIN) profile accordingly for those users (see "Oracle E-Business Suite Release 12 Single Sign-On Profile Options").
2. Configure session time out values in both Oracle E-Business Suite Release 12.2 and Oracle Single Sign-On.
3. Configure password policies, as appropriate, in Oracle Internet Directory and Oracle E-Business Suite.
4. Migrate existing Oracle E-Business Suite accounts to Oracle Internet Directory using the Oracle E-Business Suite User Bulk Migration Tool (see "Migrating Data between Oracle E-Business Suite Release 12 and Oracle Internet Directory").
5. Set Oracle E-Business Suite profile options (see "Oracle E-Business Suite Release 12 Single Sign-On Profile Options").

Profile Name (Internal Profile Code)	Recommended Value
Applications Authentication Agent (APPS_AUTH_AGENT).	Set to the location of the Oracle E-Business Suite AccessGate login page.
Applications SSO Type (APPS_SSO)	Set to 'SSWA w/SSO' to switch to Single Sign-On mode.
Self-Service Personal Home Page mode (APPLICATIONS_HOME_PAGE)	Set to the desired choice of home page.
Applications SSO Login Types (APPS_SSO_LOCAL_LOGIN)	At the site level, set the value to be the usage mode the majority of users will be in. Override at the user level for users who have special needs.
Applications Local Login URL (APPS_LOCAL_LOGIN_URL)	If using a customized local login page, set the value to be the name of the page, otherwise leave unchanged.
Applications SSO Auto Link User (APPS_SSO_AUTO_LINK_USER)	Set as needed, see "Oracle E-Business Suite Release 12 Single Sign-On Profile Options".
Applications SSO Allow Multiple Accounts (APPS_SSO_ALLOW_MULTIPLE_ACCOUNTS)	Leave unchanged.
Applications SSO LDAP Synchronization (APPS_SSO_LDAP_SYNC)	Leave unchanged at the site level, override at user level for users with special needs.
Applications Local Change Password URL (APPS_LOCAL_CHANGE_PWD_URL)	Leave unchanged unless using a customized self-service change password page to change passwords in Oracle E-Business Suite Release 12.
Applications SSO Change Password URL (APPS_SSO_CHANGE_PWD_URL)	Set to the absolute URL for self-service password change page in Oracle Internet Directory.
Applications SSO Enable OID Identity Add Event (APPS_SSO_OID_IDENTITY)	Set as needed. See "Oracle E-Business Suite Release 12 Single Sign-On Profile Options".

Profile Name (Internal Profile Code)	Recommended Value
Applications SSO Link Same Names (APPS_SSO_LINK_SAME_NAMES)	Indicates whether the Oracle E-Business Suite Release 12.2 instance should link a newly-created Oracle E-Business Suite user to an existing Oracle Internet Directory account with the same name.

Deployment Scenario 1: Multiple Oracle E-Business Suite Instances + Central SSO and OID Instance

This section and the following three present more sophisticated deployment scenarios. The solutions given should be interpreted as guidelines or building blocks rather than definitive instructions, as all real world deployments will be unique. In the cases presented, the solutions are built upon the basic scenario discussed above, and only highlight those actions that are different from or additional to, the basic one.

Starting Point

- Multiple new Oracle E-Business Suite environments (Release 12.0.0 and higher) have been installed using Rapid Install. Other than the default seeded administrative accounts, no user accounts have been registered yet.

- No single sign-on infrastructure in place.

Architectural Requirements

This scenario applies when a customer wants to integrate multiple new Oracle E-Business Suite Release 12.2 environments with a single Oracle Access Manager instance.

Solution Outline

- Oracle Access Manager, Oracle E-Business Suite AccessGate, and Oracle Internet Directory are needed for the integration required. All the installations of Oracle E-Business Suite Release 12.2 delegate user sign-on and authentication to Oracle Access Manager.

- Oracle Access Manager authenticates user credentials against user entries in Oracle Internet Directory. Oracle Internet Directory contains every user's single sign-on account id and password (except those such as SYSADMIN that are configured for local access only,).

- Either Oracle Internet Directory or one Oracle E-Business Suite Release 12.2 instance can be designated as the source of user enrollment. If Oracle Internet Directory is the source, details of user accounts can be propagated to each Oracle E-Business Suite instance via the provisioning process. If an Oracle E-Business Suite instance is the source, the provisioning process will propagate user accounts from that instance to Oracle Internet Directory, and then to the other Oracle E-Business Suite instances.

- *Optional*: User profile information in an Oracle E-Business Suite Release 12.2 instance can be kept synchronized with the information in Oracle Internet Directory.

Solution Details

User Management Options

In this solution, the system administrator must decide which component will be the point of user enrollment and the source of truth for user information. Either Oracle Internet Directory or an Oracle E-Business Suite instance can be chosen for this role.

1. Oracle Internet Directory is the point of user enrollment and source of truth.

    - After a user is created in Oracle Internet Directory, the user identity can be propagated to each Oracle E-Business Suite instance via the provisioning process. To accomplish this, the provisioning profile for each Oracle E-Business Suite Release 12.2 instance needs to enable the SUBSCRIPTION_ADD event from Oracle Internet Directory to Oracle E-Business Suite Release 12.2.

    - *Optional*: The provisioning profile can also be configured such that user profile information change in Oracle Internet Directory can be propagated to each Oracle E-Business Suite Release 12.2 instance. To accomplish this, the provisioning profile for each Oracle E-Business Suite Release 12.2 instance needs to enable the IDENTITY_MODIFY event from Oracle Internet Directory to Oracle E-Business Suite Release 12.2.

2. An Oracle E-Business Suite Release 12.2 instance is designated as the point of user enrollment and source of truth (the master instance).

    - After a user is created from the master Oracle E-Business Suite Release 12.2 instance, the provisioning process can be used to propagate the user identity first to Oracle Internet Directory, then to other Oracle E-Business Suite Release 12 instances. To accomplish this, the provisioning profile for the master Oracle E-Business Suite Release 12.2 instance needs to enable the IDENTITY_ADD event from Oracle E-Business Suite Release 12.2 to Oracle Internet Directory. The provisioning profile for the rest of the Oracle E-Business Suite Release 12.2 instances needs to enable the SUBSCRIPTION_ADD event from Oracle Internet Directory to Oracle E-Business Suite Release 12.2.

Deployment Scenario 2: New Oracle E-Business Suite Installation + Existing Third-Party Identity Management Solution

This section presents a slightly more sophisticated deployment scenario.

Starting Point

- Oracle E-Business Suite Release 12.2 has been newly installed using th Rapid Install. Other than the default seeded Release 12.2 administrative accounts, no user accounts have been registered yet.
- A third-party authentication mechanism is in use as a corporate single sign-on solution.
- A third-party LDAP directory is in use as a corporate user directory.

Architectural Requirements

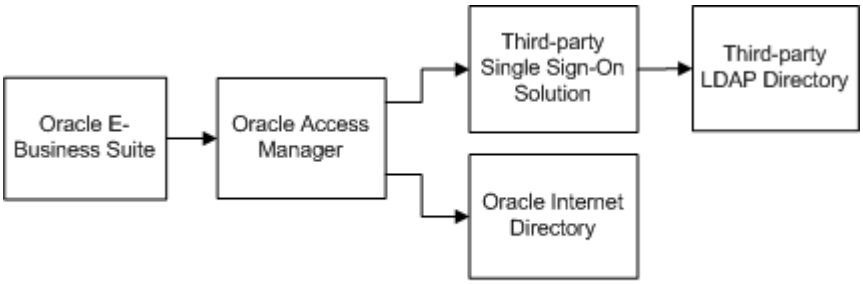
Need to integrate new installation of Oracle E-Business Suite Release 12.2 with existing third-party single authentication mechanisms and third-party LDAP directory infrastructure.

Solution Outline

- Oracle Access Manager, Oracle E-Business Suite AccessGate, and Oracle Internet Directory) are used for integration with third-party authentication mechanisms or third-party LDAP directories.

  Note: Integrating Oracle E-Business Suite directly with third-party authentication mechanisms or third-party LDAP directories is not supported.

- Oracle E-Business Suite and Oracle Access Manager need to be set up to enable Oracle E-Business Suite delegation of authentication to Oracle Access Manager, which in turn delegates the functionality to the third-party single sign-on authentication mechanism.



Single Sign-On Chain of Trust with Third-Party Single Sign-On Solution

- Oracle Internet Directory needs to be set up to synchronize a minimal set of user attributes when integrating with a third-party LDAP directory. Refer to the *Oracle Directory Integration and Provisioning Platform* in Oracle Internet Directory Administrator's Guide for more information about performing this integration.
- User information from the third-party LDAP directory for all users who will access Oracle E-Business Suite via single sign-on. Oracle Internet Directory also needs to be set up to provision users in Oracle Internet Directory to Oracle E-Business Suite.
- Existing users in the third-party LDAP can be bulk migrated into Oracle Internet Directory, and then bulk migrated into Oracle E-Business Suite.
- *Optional*: A set of user profile information in Oracle E-Business Suite can be kept synchronized with the information in the third-party LDAP directory.

End-User Experience

Single Sign-On User Experience

- **Sign on process**: the sign on user experience is the same as that in the base scenario, except that the login page is served by the third-party authentication mechanism.
- **Sign out process**: when a user logs out from Oracle E-Business Suite Release 12.2, Oracle Access Manager logs the user out of all registered Oracle partner applications. The user is also logged out of the third-party single sign-on solution, if the administrator has set this up in the `samplecleanup` script.
- **Session timeout**: the session timeout user experience is the same as that in the base scenario, except that the user will be asked to re-authenticate only when the application session, the Oracle single sign-on session and the third-party session have all become invalid.

Single Sign-On Technical Architecture

When an unauthenticated user attempts to access Oracle E-Business Suite Release 12.2, Oracle E-Business Suite Release 12.2 delegates user authentication to Oracle Access Manager, which in turn delegates to the third-party authentication mechanisms.

User Management

Oracle Internet Directory and Third-Party LDAP Directories

- Oracle Internet Directory can synchronize user information with a third-party LDAP server via the synchronization process.
- Oracle Internet Directory includes tools to bulk migrate user between Oracle Internet Directory and third-party LDAP server.

> **Note:** Refer to the Oracle Internet Directory 11g Administrator's Guide for more information.

Strategies for User Management

At the starting point of the deployment, the third-party LDAP server is the sole user repository. For users registered there who will need to access Oracle E-Business Suite, the single sign-on solution requires them to exist in Oracle Internet Directory as well as in Oracle E-Business Suite Release 12.2.

Oracle recommends retaining the third-party LDAP directory as the master source of truth for user information. Use the Oracle Internet Directory synchronization solution to migrate users from the third-party LDAP directory into Oracle Internet Directory, and then use the Oracle Internet Directory provisioning solution to move users into Oracle E-Business Suite.

> **Important:** For pending users that are enabled in Oracle E-Business Suite after user creation, the IDENTITY_MODIFY event from E-Business Suite to Oracle Internet Directory must be enabled.

Populating E-Business Suite with Third-Party LDAP Users

Existing users can be migrated from the third-party LDAP directory into Oracle Internet Directory, and then into Oracle E-Business Suite via the bulk migration tool.

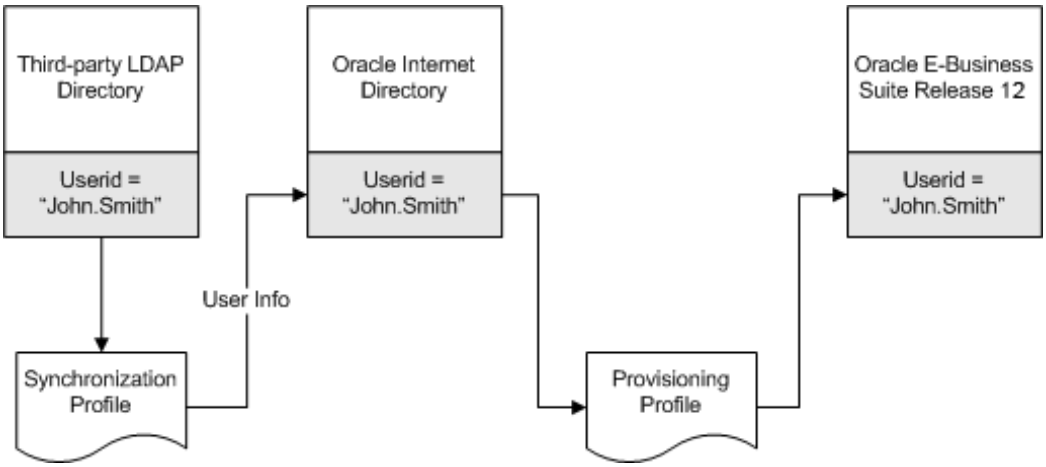
Creating New Users

System administrators can create synchronization profiles to integrate Oracle Internet Directory with the third-party LDAP directory, which results in:

- Creation of a new single sign-on account in the third-party LDAP directory automatically triggering the creation of a new single sign-on account in Oracle Internet Directory.
- Ability to specify users to be synchronized, and which attributes of the users are to be created in Oracle Internet Directory.
- Creation of a GUID attribute for each user created in Oracle Internet Directory.

System administrators also create provisioning profiles to integrate Oracle E-Business Suite Release 12 with Oracle Internet Directory, which results in:

- Creation of a new account in Oracle Internet Directory automatically triggering the creation of a new application account in Oracle E-Business Suite Release 12.
- Ability to specify user attributes created in Oracle E-Business Suite.



Updating User Information (optional)

System administrators can configure synchronization profiles to synchronize some or all of the user attributes from the single sign-on account in the third-party LDAP directory into the single sign-on account in Oracle Internet Directory when those attributes are modified.

System administrators can configure provisioning profiles to provision some or all of the user attributes from Oracle Internet Directory into Oracle E-Business Suite when those attributes are modified.

Terminating and End-Dating Users

Synchronization and provisioning profiles can also be used to configure the system such that terminating a user in the third-party LDAP directory also end-dates the user in Oracle E-Business Suite.

Password Management

Password management can, if desired, remain as it was before the integration. That is, user passwords can remain in the third-party LDAP; it is not necessary to duplicate them in Oracle Internet Directory. Note that Oracle E-Business Suite will not store passwords for users provisioned from Oracle Internet Directory.

- **End user tasks**: Most end users should use the methods provided by the third-party LDAP directory for password maintenance functions.
- **System administrator tasks**: To reset single sign-on passwords, an administrator should follow the methods provided by the third-party LDAP directory.
- **Password management policies**: User's password creation, modification and single sign-on login activities are subject to the third-party LDAP rules that govern how passwords are created and used.

Critical Implementation Decisions

Oracle Internet Directory has a powerful and flexible set of configuration options. Most Oracle E-Business Suite system and security administrators will be able to use the default Oracle Internet Directory configuration. Security administrators with advanced security requirements may choose to use alternate Oracle Internet Directory configurations. For more information, refer to the 'Directory Deployment' chapter in the Oracle Internet Directory Administrator's Guide.

1. Oracle E-Business Suite integration:
    - Identity management realm
    - DIT structure
    - The attribute chosen as the nickname attribute
2. Synchronization between Oracle Internet Directory and third-party LDAP directory:
    - Identifying users who need to access Oracle E-Business Suite Release 12.2, and must therefore be synchronized from the third-party LDAP directory to Oracle Internet Directory.

- Which user attributes to synchronize from the third-party LDAP directory to Oracle Internet Directory.

3. Provisioning between Oracle Internet Directory and Oracle E-Business Suite:
   - Which attributes to provision during account creation.
   - Whether to provision user changes from Oracle Internet Directory to Oracle E-Business Suite Release 12.2. If yes, which attributes to provision.
4. Single sign-on settings
5. Session timeouts for Oracle Access Manager, third-party single sign-on, and Oracle E-Business Suite Release 12.
6. Current third-party LDAP/single sign-on deployment information, including host, port, and administration account information.
7. Documentation from Oracle and third-party LDAP and single sign-on product vendors describing integration with Oracle Application Server 10g.

Implementation Instructions

1. Configure Oracle Access Manager to work with third-party authentication mechanism.
2. Replicate existing accounts that need to access Oracle E-Business Suite from third-party LDAP into Oracle Internet Directory. Configure Oracle Internet Directory and third-party LDAP synchronization process.
3. Migrate existing Oracle Internet Directory users into Oracle E-Business Suite.
4. Set Oracle E-Business Suite profile options. The profile settings should be similar to that of the base scenario. Refer to "Oracle E-Business Suite Release 12 Single Sign-On Profile Options" for details of all relevant profile options.

Variations On This Scenario

Variation of this scenario may have some of the following characteristics:

- Oracle E-Business Suite fresh install.
- Existing Oracle Access Manager and Oracle Internet Directory infrastructure.
- No third-party authentication mechanism or third-party LDAP directory involved.

The major difference here is that the steps relating to third-party (non-Oracle) software can be ignored.

Deployment Scenario 3: Existing Oracle E-Business Suite Instance + Existing Third-Party Identity Management Solutions

This scenario describes a more complex deployment possibility, which may be required in some larger organizations.

Starting Point

- Oracle E-Business Suite Release 12.2 is in use, and has existing users populated in an up-to-date FND_USER repository.
- A third-party authentication mechanism is in use as a corporate single sign-on solution.
- A third-party LDAP directory in use as a corporate user directory.
- At the start of the implementation, a user may exist in both Oracle E-Business Suite Release 12.2 and the third-party LDAP directory, with either the same user name in both, or a different user name in each.

Architectural Requirements

Need to integrate existing Oracle E-Business Suite Release 12.2 with existing third-party single sign-on and user directory infrastructure.

Solution Outline

- Oracle Access Manager, Oracle E-Business Suite AccessGate, and Oracle Internet Directory are used for the integration. Oracle E-Business Suite and Oracle Access Manager need to be set up so that Oracle E-Business Suite delegates authentication to Oracle Access Manager, which in turn delegates the functionality to the third-party authentication mechanism in use.
- Oracle Internet Directory must be configured to synchronize a minimal set of information from the third-party LDAP directory for users who will access Oracle E-Business suite via single sign-on.
- Existing users in the third-party LDAP directory can be bulk migrated into Oracle Internet Directory.
- Existing accounts in both Oracle E-Business Suite and third-party LDAP can be linked. With proper planning, new users can be synchronized from the third-party

LDAP directory into Oracle Internet Directory, and then into Oracle E-Business Suite.

- *Optional*: User profile information in Oracle E-Business Suite can be kept synchronized with the information in the third-party LDAP directory.

Solution Details

The single sign-on, sign-off and session timeout processes in this deployment scenario are similar to that in Scenario 2, with one significant difference during sign-on. In the case where a user already has an account in the third-party LDAP directory and an account in Oracle E-Business Suite (with the same account name or a different account name), Oracle recommends the following approach:

- Migrate the third-party LDAP account into Oracle Internet Directory through either the bulk migration tool (for existing accounts) or the synchronization process (for new accounts).

- Use the Link-on-the-Fly feature to link the single sign-on account in Oracle Internet Directory with the applications account in Oracle E-Business Suite Release 12.2, by proceeding as follows:

  1. In the single sign-on handshake (described in the base scenario), Oracle Access Manager returns the GUID of the authenticated user to Oracle E-Business Suite.

  2. Oracle E-Business Suite then uses the GUID to try to locate the user's Oracle E-Business Suite application account.

  3. If it is the first time the user is accessing an Oracle E-Business Suite instance, no associated application account will be found, since the user's Oracle E-Business Suite account did not have the GUID information before the Oracle Access Manager integration took place.

  4. The user is directed to a 'Link Account' screen for entry of the Oracle E-Business Suite username and password.

- Once the application account information has been successfully verified, the user is redirected to the requested Oracle E-Business Suite page or the user's home page, as applicable. Additional logic is as follows:

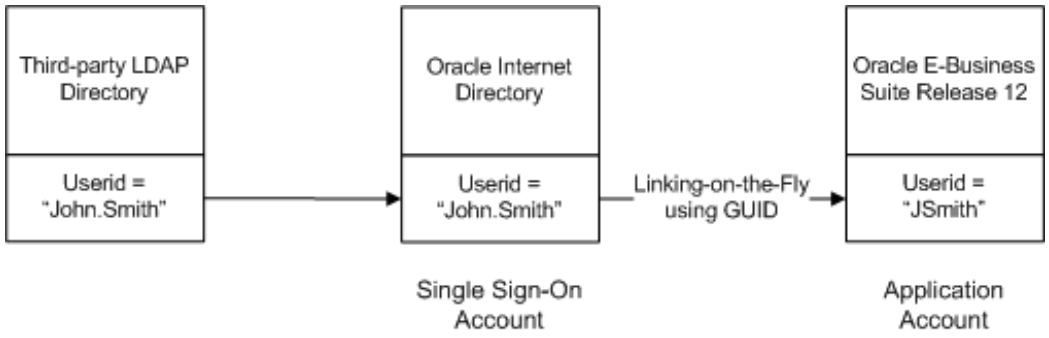
  1. The association between the single sign-on account and the application account (represented by the GUID) is retained.

  2. Oracle E-Business Suite will not redirect the user to the 'Link Account' page on subsequent accesses.

  3. If the application account information is not verified, the user is directed back

to the 'Link Account' page.

- This overall process is illustrated by the following diagram:



Advanced Option: In cases where users have accounts in both a third-party LDAP directory and Oracle E-Business Suite, it may sometimes be the case that all the LDAP account names are known to be identical to the Oracle E-Business Suite account names. In such cases, the value of the profile 'Applications SSO Auto Link User' can be set to 'Y'. Subsequently, when Oracle E-Business Suite fails to locate an application account by GUID, it will try to locate one by the account name, and if successful it will then link the two accounts by GUID. The linking operation will be performed behind the scenes, and the user will not see the 'link account' page. See "Oracle E-Business Suite Release 12 Single Sign-On Profile Options" for more details.

User Management Options

The complexity of user management in this scenario lies mostly in the process of reconciling existing user data in the third-party LDAP and Oracle E-Business Suite. It is always necessary to synchronize the third-party LDAP data into Oracle Internet Directory for any users who need to access Oracle E-Business Suite via single sign-on. The single sign-on accounts in Oracle Internet Directory should be identical to the accounts in the third-party LDAP directory. No action is required for users whose details reside in the third-party LDAP and who do not need to access Oracle E-Business Suite.

For the rest of this discussion, it is assumed that all existing third-party LDAP users will need to access Oracle E-Business Suite, and that such users will therefore need to exist in Oracle Internet Directory. Depending on the characteristics of the existing data and desired functionality, there are various possibilities.

Option 1: Require users always to have created an account in the third-party LDAP directory and an account in the Oracle E-Business Suite, via the user enrollment method provided by each system.

In this case, the LDAP accounts are migrated into Oracle Internet Directory. The Oracle Internet Directory accounts and the Oracle E-Business Suite accounts are linked via the Link-on-the-Fly process described above (neither SUBSCRIPTION_ADD nor

IDENTITY_ADD event are enabled in any provisioning profiles used).

Optionally, administrators can configure the synchronization and provisioning process so that changes in user attributes can be propagated:

- From the third-party LDAP directory into Oracle E-Business Suite via Oracle Internet Directory
- From Oracle E-Business Suite into the third-party LDAP directory via Oracle Internet Directory
- In both directions

The list of user attributes supported is currently limited, and listed later in "Supported Attributes".

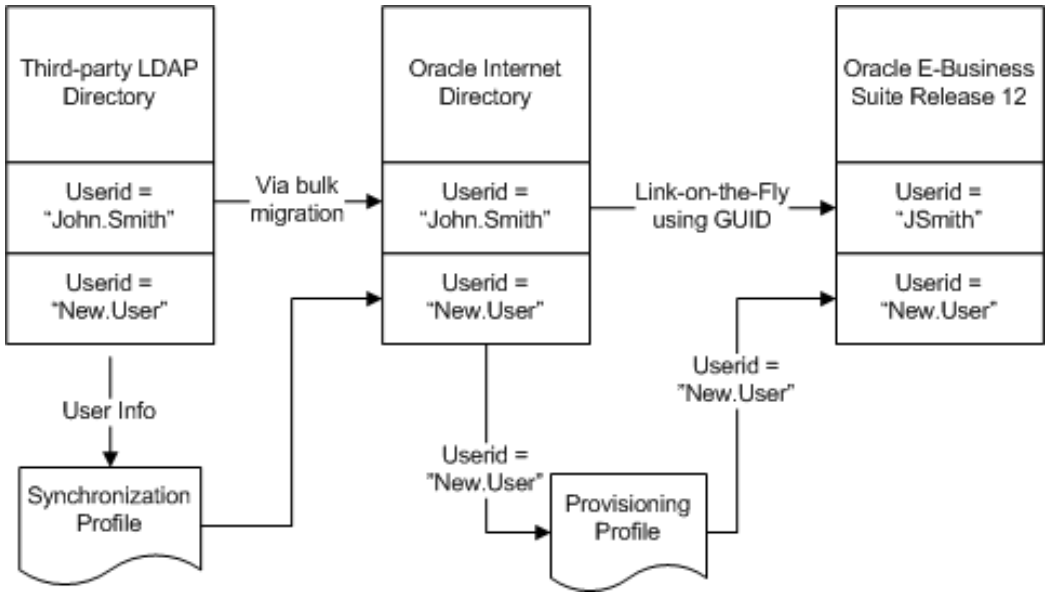
Option 2: Propagate new accounts from the third-party LDAP directory to Oracle E-Business Suite via Oracle Internet Directory (as described in Scenario 2).

Existing accounts in LDAP and/or Oracle E-Business Suite will need to be reconciled. If a user has an existing account in the LDAP directory, and an existing account in Oracle E-Business Suite, the Link-on-the-Fly feature can be used to link the two accounts; no other action is required If a user has an existing account in Oracle E-Business Suite, but not in the third-party LDAP directory, an account must be created in the LDAP directory, and Link-on-the-Fly used to link the two accounts (this step needs to be performed before provisioning is configured).

If a user has an existing account in the third-party LDAP directory, but not in the Oracle E-Business Suite, an account must be created in Oracle E-Business Suite, and Link-on-the-Fly used to link the two accounts.

To eliminate the need to use the "Link Account" functionality for new users, new accounts can be propagated from the third-party LDAP directory to Oracle E-Business Suite via the Oracle Internet Directory synchronization and provisioning process. This strategy also eliminates the need for new users to enroll multiple times. However, before enabling this process, system administrators must set up procedures to ensure that new account names created in the third-party LDAP directory will not conflict with any existing account names in Oracle E-Business Suite.

Optionally, administrators can configure the synchronization and provisioning process so that changes in user attributes can be propagated from the third-party LDAP directory into Oracle E-Business Suite via Oracle Internet Directory.



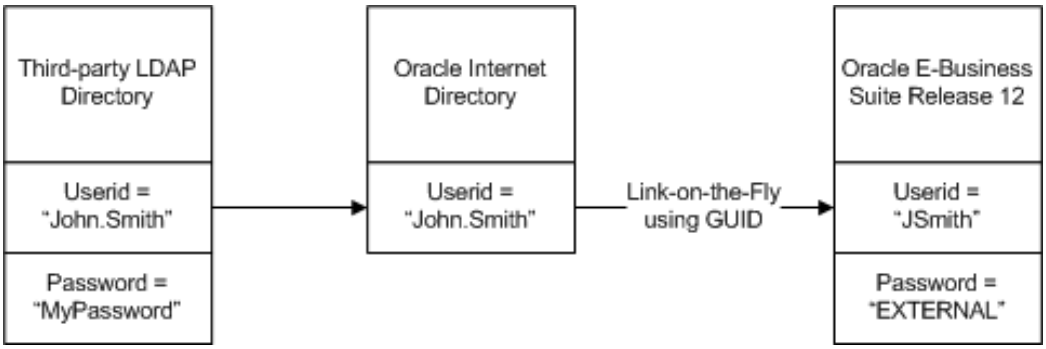
Password Management

Once a single sign-on account in Oracle Internet Directory is linked to an application account in Oracle E-Business Suite, the password for the application account in Oracle E-Business Suite is, as mentioned earlier, replaced with the reserved keyword " EXTERNAL". The password stored in the master user directory for passwords is sufficient for authentication purposes.

Note that Oracle Access Manager delegates user authentication to the third-party single sign-on solution, which in turn authenticates users against the third-party LDAP directory. Users cannot gain access to Oracle E-Business Suite via AppsLocalLogin.jsp. As Oracle Internet Directory passwords will be ignored, it is inadvisable to retain any passwords in Oracle Internet Directory.

> **Note:** If an SSO user's setting of APPS_SSO_LOCAL_LOGIN is changed to Local or Both (for local access), the user's password will need to be changed by an administrator.

The primary role of the third-party LDAP directory here can be represented as shown in the following diagram:



Critical Implementation Decisions

1. Oracle Internet Directory has a powerful and flexible set of configuration options. Most E-Business Suite system and security administrators will be able to use the default Oracle Internet Directory configuration. Security administrators with advanced security requirements may choose to use alternate Oracle Internet Directory configurations. Refer to the 'Directory Deployment' chapter in the Oracle Internet Directory Administrator's Guide.

   Items of particular importance to Oracle E-Business Suite integration are:

   - Identity management realm
   - DIT structure
   - The attribute chosen as the nickname attribute

2. Synchronization between Oracle Internet Directory and third-party LDAP directory.

   Items of particular importance are:

   - Identifying users who need to access Oracle E-Business Suite Release 12.2 and who therefore need to be synchronized between the third-party LDAP directory and Oracle Internet Directory
   - Which attributes to use to synchronize between Oracle Internet Directory and the third-party LDAP directory

3. Which user management option described above to use.

4. Decisions related to single sign-on settings, especially session timeouts for:

   - Oracle Access Manager
   - Third-party single sign-on components

- Oracle E-Business Suite Release 12.2

5. Current third-party LDAP and single sign-on deployment information, including host, port, and administration account information. For this, you may need to refer to documentation from Oracle and third-party LDAP and single sign-on product vendors describing integration with Oracle Access Manager.

Implementation Instructions

1. Depending on the user management options, develop a strategy to reconcile existing accounts in Oracle E-Business Suite Release 12.2 and the third-party LDAP directory.

2. Configure Oracle Access Manager to work with the third-party authentication mechanism.

3. Migrate existing third-party LDAP accounts to Oracle Internet Directory, and configure synchronization between third-party LDAP and Oracle Internet Directory.

4. Configure session timeout setting.

5. Set Oracle E-Business Suite profile options. Refer to "Oracle E-Business Suite Release 12.2 Single Sign-On Profile Options" for further details of relevant profile options.

Variations On This Scenario

A variation of this scenario may have the following characteristics:

- Existing Oracle E-Business Suite Release 12.2 Installation
- Existing Oracle Access Manager and Oracle Internet Directory infrastructure
- No third-party single authentication mechanism or third-party LDAP directory involved

The major difference here is that all steps relating to third-party (non-Oracle) software can be ignored.

Deployment Scenario 4: Multiple Oracle E-Business Suite Instances with Unique User Populations

Starting Point

- Multiple Oracle E-Business Suite Release 12.2 instances are implemented, and each has an existing user population.
- No existing Oracle Access Manager infrastructure is in place

Architectural Requirements

This scenario applies to sites that have more than one Oracle E-Business Suite Release 12.2 instance in use, but no Oracle Access Manager infrastructure in place. The requirement is to enable Oracle Access Manager for the multiple Oracle E-Business Suite instances.

Solution Outline

- Oracle Access Manager, Oracle E-Business Suite AccessGate, and Oracle Internet Directory are used for the integration. Each Oracle E-Business Suite instance delegates user sign-on and authentication to Oracle Access Manager.
- Oracle Access Manager authenticates user credentials against user entries in Oracle Internet Directory, which contains every user's single sign-on account ID and password.
- A single sign-on account needs to be created for every user in Oracle Internet Directory. Existing applications accounts in Oracle E-Business Suite instances need to be linked to the single sign-on account.
- *Optional*: User profile information in Oracle E-Business Suite can be kept synchronized with the information in Oracle Internet Directory.

Solution Details

The single sign-on architecture is the same as that described in the base scenario. In addition, the Link-on-the-Fly feature described in Scenario 3 may be used.

User Management Options

The options for user management in this scenario depend on the characteristics of existing user data in the multiple Oracle E-Business Suite instances.

Option 1: If one of the Oracle E-Business Suite instances is currently serving as the

source of truth for user information for all Oracle E-Business suite instances, it is possible to change this in a two-stage process. First, migrate the existing users from that Oracle E-Business Suite instance into Oracle Internet Directory using the bulk migration tool, and then configure the provisioning process such that any further new users created in that Oracle E-Business Suite instance are automatically provisioned into Oracle Internet Directory.

- Users who already have accounts on the other Oracle E-Business Suite instances will use the Link-on-the-Fly mechanism to link their single sign-on accounts to their application accounts on those instances.

- New users provisioned into Oracle Internet Directory can be selectively provisioned into the other Oracle E-Business Suite instances.

Option 2: If none of the existing Oracle E-Business Suite instances is the master source of truth for user information, it is possible to migrate the existing accounts in all Oracle E-Business Suite instances into Oracle Internet Directory with the following restrictions on the existing data:

- No two users have the same account names across all Oracle E-Business Suite instances.

- If a user has accounts in multiple Oracle E-Business Suite instances, those accounts must be of the same account name.

After the migration, new users can be created from Oracle Internet Directory, and then selectively provisioned into an Oracle E-Business suite instance.

Option 3: If the above options are not feasible, a deployment may choose not to rely on the provisioning process for creating accounts (no SUBSCRIPTION_ADD nor IDENTITY_ADD event enabled in provisioning profile). Every user who needs single sign-on access to an Oracle E-Business Suite is required to have created a single sign-on account in Oracle Internet Directory, and an application account in that Oracle E-Business Suite Release 12.2 instance, via the user enrollment method provided by each system. The Oracle Internet Directory account and Oracle E-Business Suite account are linked via the Link-on-the-Fly process when the user accesses an Oracle E-Business instance for the first time.

Advanced Features

Personalizing the Local Login Page

By default, all regions are displayed on the Oracle E-Business Suite local login page. As with all Oracle Application Framework-based pages, however, it can be personalized. Some of the personalizations that may be desired are:

- Hiding "Register Here" and "Login Assistance" links

- Hiding the language images region
- Hiding the *Cancel* button

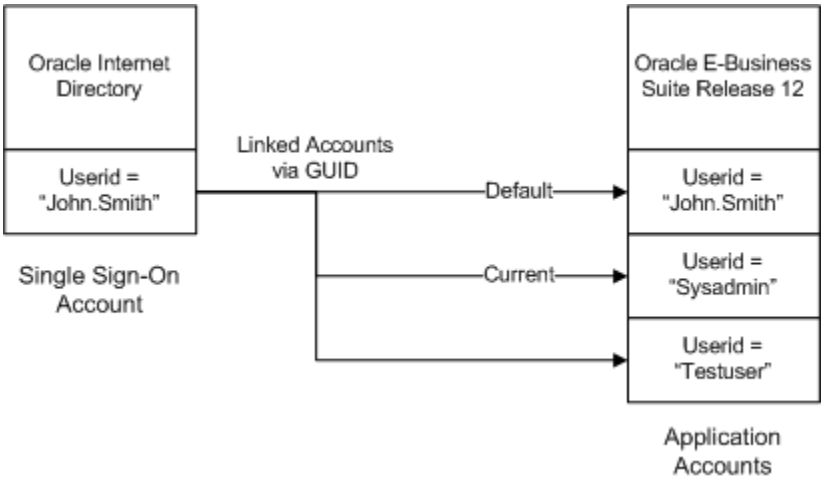
Setup Steps for Login Page Personalization

1. Set the profile FND_PERSONALIZATION_REGION_LINK_ENABLED to *Yes*
2. Select the Functional Administrator responsibility
3. Select the Personalization tab
4. Enter the document path for the Local Login page definition
5. Select a Region to customize
6. This takes you to the Choose Personalization Context page: select *Apply*.
7. The personalization structure is displayed where an item can be selected and its properties changed

For a description of how login pages can be customized, see Custom Login Pages in the section Single Sign-On Profile Options.

Linking Multiple Application Accounts to One Oracle Single Sign-On Account

In most cases, a user's single sign-on account in Oracle Internet Directory will correspond to a single application account in Oracle E-Business Suite Release 12.2. However, there may be special cases where a user has a single sign-on account in Oracle Internet Directory and multiple application accounts in Oracle E-Business Suite Release 12.2. In such a case, it is possible to associate a single sign-on account in Oracle Internet Directory with multiple application accounts in Oracle E-Business Suite Release 12.2:



This feature can be enabled by system administrators via a profile option ('Applications SSO Allow Multiple Accounts'). To utilize this feature, proceeds as follows:

1. Log in to Oracle E-Business Suite using a valid single sign-on account in Oracle Internet Directory.

2. Once logged in, access the 'Single Sign-On Account Settings' page by clicking the 'Account Settings' button from the 'Preferences' page.

3. To associate additional application accounts with an existing single sign-on account, choose 'Add Account' and enter the new application account user name and password when prompted.

4. Verification of the new application account information will result in redirection back to the 'Single Sign-On Account Settings' page, showing the newly linked account.

5. Failure to verify the new account information will result in redirection back to the 'Add Account' page.

The first linked application account is marked as the default application account for the single sign-on account, and is the account the user will be logged into after Oracle single sign-on authentication. If required, the default account can be changed by making the appropriate selection on the 'Single Sign-On Account Settings' page.

After logging into Oracle E-Business Suite via Oracle single sign-on, a user can view all currently linked application accounts using the 'Single Sign-On Account Settings' page, and can if desired switch to another linked application account by selecting that account and clicking on 'Make Current Account'. If this feature is disabled by the system administrator, the 'Add Account' button will not appear on the 'Single Sign-On Account Settings' page and users will not be permitted to link multiple application accounts to their single sign-on account.

Only one single sign-on account in Oracle Internet Directory may be linked to a given application account in Oracle E-Business Suite Release 12.2 at a time; simultaneous linking of multiple single sign-on accounts to a single application account is not supported.

Time Zone Support

Oracle Access Manager and the Oracle E-Business Suite database machine system clocks should be accurate, and kept synchronized. If the clocks are inaccurate or out-of-sync, user provisioning flows may be affected.

Be aware of the following points:

- Oracle Access Manager converts all times to GMT. If the orclStartDate attribute is defaulted, it will pick the system date and convert it to GMT.

- The Oracle E-Business Suite database machine runs in the local time zone, so dates are also in the local time zone.

- When a user is provisioned from Oracle Internet Directory, the dates are converted to the local time zone.

Switching User Back to Local Authentication

It may be necessary to switch the user management master from Oracle Internet Directory back to Oracle E-Business Suite for specific users. Credentials for these users will need to be switched back to being authenticated by FND_USER for local authentication. Special procedures to do this are necessary, because the FND User form as well as the User Preferences screen will not allow you to change the password once it has been set to "EXTERNAL".

To preserve the password and allow users to locally log in to Oracle E-Business Suite, follow these steps:

1. Ensure that the profile option 'Applications SSO Login Types' (APPS_SSO_LOCAL_LOGIN) is set to either 'LOCAL' or 'BOTH' for users to whom you want to keep the local access.

2. Use the AFPASSWD utility or FNDCPASS utility to reset the user's password. The new password then needs to be emailed to the user.

For more information about the AFPASSWD and FNDCPASS utilities, refer to the *Applications DBA Duties* chapter of *Oracle E-Business Suite Setup Guide*.

Recommended Nickname (Login Attribute) Setting

The default nickname used for login is "uid", which can be verified in the Oracle Internet Directory Delegated Administration Service Configuration screen, Attribute for Login Name field. "uid" corresponds to User Name in the Oracle Directory Services Manager UI.

Changing the nickname attribute is generally not recommended, but other unique attributes such as email address can be used in special circumstances. Oracle E-Business Suite currently supports setting of the nickname (login attribute) to either *uid* or *mail*.

The attribute set as the nickname in Oracle Internet Directory is mapped to the FND_USER.USER_NAME column in the Oracle E-Business Suite database. If the nickname is changed in Oracle Internet Directory, the Oracle E-Business Suite database must be restarted to force a refresh of the cached value.

Customizing Directory Information Tree (DIT) and Relative Distinguished Name (RDN)

Customizable Directory Information Trees (DIT) and Relative Distinguished Names (RDN) are supported for use with Oracle E-Business Suite single sign-on environments.

Described further in *Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory 11g*, the following parameters can be defined at realm level:

- Name Attribute (NickNameAttribute)
- UserCreateBase: one or more DN where the user entries are located
- Attribute for RDN
- UserSearchBase: in the hierarchical path for all defined UserCreateBases, this is the location to start searching for users of a given username

> **Caution:** Implementing the Custom DIT feature in an existing infrastructure is not recommended, as it may result in data corruption. If there is such a need, contact Oracle Support for details of how to migrate existing data safely.

The Custom DIT feature should not be confused with Multiple Realm support.

Custom DIT Configuration Steps

The Custom DIT feature requires the following configuration steps within Oracle Internet Directory and Oracle E-Business Suite.

In Oracle Internet Directory (see Oracle Internet Directory Administration Guide for details):

1. Create the new DIT structure.
2. Optionally, configure the CommonNameAttribute to be used for the RDN (the default is cn).
3. Specify a single UserSearchBase where all UserCreateBases can be located.

> **Caution:** The current implementation supports only one UserSearchBase. Using more than one may result in incorrect operation.

In Oracle E-Business Suite:

1. Register the Oracle E-Business instance with the desired deployment template. Note that this feature is only relevant for the deployments provisioning users from Oracle E-Business Suite to Oracle Internet Directory.
2. From the APPS account, run the API `fnd_oid_plug.setplugin` from SQL*Plus to configure Oracle E-Business Suite for use with the new user repository.

   For example:

```
sql>fnd_oid_plug.setPlugin(default_user_repository=>'cn=new_reposito
ry,dc=example,dc=com');
```

The Oracle Internet Directory configuration attributes are now stored in Oracle E-Business Suite preferences.

> **Note:** Any configuration changes in OID will require this API to be re-run so that the new values are picked up by Oracle E-Business Suite.

3. Stop and restart the application tier processes

Now, when new users are created in Oracle E-Business Suite, they will also be created in the OID User Repository. This will have no impact to the propagation of users from OID to Oracle E-Business Suite. Note, however, that the same "user" cannot be created in multiple user repositories.

Single Sign-On Profile Options

The logon process by which users are authorized to access Oracle E-Business Suite is significantly modified in an environment where Oracle Access Manager (and the associated Oracle E-Business Suite AccessGate) have been integrated. This section discusses the key changes, in particular the use of profile options.

Overview of Login Pages

In a standalone Oracle E-Business Suite environment, all users and system administrators connect via Oracle E-Business Suite's AppsLogin page. This page redirects users to an Oracle E-Business Suite login page that authenticates their userid and password against the FND_USER table. Oracle E-Business Suite then determines the user's authorization by looking up the application responsibilities against entries in the FND_USER table.

In an environment where Oracle E-Business Suite has been integrated with Oracle Access Manager and Oracle Internet Directory, the following points apply:

- End users connect to Oracle E-Business Suite via the AppsLogin page, which redirects them to the Oracle Access Manager login page. Oracle Access Manager authenticates the Oracle E-Business Suite user's userid and password against Oracle Internet Directory, and redirects the user back to Oracle E-Business Suite, which then determines the user's authorizations by looking up application responsibilities against entries in the Oracle E-Business Suite FND_USER table.

- System administrators and other selected users connect to Oracle E-Business Suite via Oracle E-Business Suite's AppsLocalLogin page, which authenticates their userid and password against the FND_USER table. Oracle E-Business Suite then determines the user's authorizations by looking up application responsibilities

against entries in the FND_USER table. Users in this special user population have their credentials authenticated locally in Oracle E-Business Suite instead of externally in Oracle Access Manager and Oracle Internet Directory.

The login process is controlled by a group of Oracle E-Business Suite profile options, which are described in more detail below.

The key components involved in the login process are as follows.

AppsLogin

`<http://[host]:[port]/OA_HTML/AppsLogin>`

The login route is determined by the profile option "Applications SSO Type" (APPS_SSO). If the Oracle E-Business Suite instance is integrated with Oracle Access Manager, this should be set to "SSWA w/SSO". The user is redirected to the Oracle E-Business Suite AccessGate login page, and after entering his credentials (username and password), he is authenticated against the LDAP server.

AppsLocalLogin

`<http://[host]:[port]/OA_HTML/AppsLocalLogin.jsp>`

The login route is determined by the profile option "Applications SSO Type" (APPS_SSO). If this site level profile is set to "SSWA", the user will be shown the local login page, and after entering his credentials (username and password), he is authenticated against the Oracle E-Business Suite instance.

> **Note:** If APPS_SSO is set to SSWA, the user will be redirected to AppsLocalLogin.jsp regardless of whether or not OAM integration is in effect. When accessing AppsLocalLogin.jsp, the APPS_SSO profile is not used to determine the page to redirect to.

The login page is an Oracle Application Framework-based page, so the regions can be person. Administrators can personalize the page by setting the profile FND_PERSONALIZATION_REGION_LINK_ENABLED to 'Yes'.

By default, all the regions on the login page are displayed. The following items may be personalized:

- User Name
- Password
- Login button
- Cancel button
- Login Assistance Link
- Register Here Link

- Accessibility
- Language Options

Custom Login Pages

System Administrators can create custom login pages. The custom page will need to post to the servlet AuthenticateUser, which requires two attributes: username and password. Once the user is successfully authenticated, the servlet will redirect the user to a destination defined in *requestUrl* or the default *APPSHOMEPAGE*. If the authentication fails, the servlet will redirect the user to the login page with the error message in the parameter *errCode*.

To deploy a custom login page:

1. Place the new servlet in the OA_HTML directory.
2. Create a new function (FND_FORM_FUNCTION) - the web_html value of this function should be populated with file name of your new login page. The function code should begin with 'APPS_LOGIN'.
3. Assign this function to the APPS_LOGIN_DEFAULT menu. As this menu is already granted to all users (including guest), the grant flag is not needed.
4. Update the profile option APPS_LOGIN_FUNCTION with new function name. The drop-down for this profile will query only function codes starting with APPS_LOGIN.

CRMLogin servlet and jtflogin.jsp

```
<http://[host]:[port]/oa_servlets/CRMLogin.jsp>
http://[host]:[port]/OA_HTML/jtflogin.jsp
```

There is a new recommended login flow for the CRM System Administrator Console. You can use the servlet CRMLogin to log in. The servlet checks whether your system is SSO-enabled, and directs you to the appropriate login page. The old login page, jtflogin.jsp, is still supported, but is only recommended in cases where jtflogin.jsp has been customized.

Oracle Applications Manager Login

```
http://[host]:[port]/servlets/weboam/oam/oamLogin
```

> **Important:** Here, "oam" refers to Oracle Applications Manager, **not** Oracle Access Manager.

You will be prompted for the Oracle E-Business Suite user account and password. Log in to an account that has System Administrator and Self-Service System Administrator responsibilities. Upon successful login, the Oracle Applications Manager Console will show the Oracle E-Business Suite system to which you have connected.

Profiles and Profile Categories

The login process is determined by a group of Oracle E-Business Suite profile options, which are divided into several categories and described below. The major components involved in the logon process are as follows.

Profiles for Login and Logout

The profiles described in this category are all related to the login and logout process.

Applications SSO type (APPS_SSO)

Features of this profile:

- Available at site level only (cannot be set for individual servers or users)
- Updatable only by system administrators
- Defined by the lookup type 'APPS_SSO_TYPE'
- Has a default value of 'SSWA'

This profile determines the overall user login and authentication experience, as follows:

Profile Value	Login Via	Authentication	User directory	Integration model	Requires	Home Page
SSWA w/SSO	OAM login page	Oracle Access Manager	OID	EBS is partner application to Oracle SSO	EBS AccessGate installed into EBS instance	Set by APPLICATIONS_HOME_PAGE profile
Portal w/SSO	OAM login page	Oracle Access Manager	OID	EBS is a partner application to SSO	EBS AccessGate installed into EBS instance	Portal home page

Profile Value	Login Via	Authentication	User directory	Integration model	Requires	Home Page
SSWA	EBS login page	EBS	FND_USER	N/A	N/A	Set by APPLICATIONS_HOME_PAGE profile

Note: In the above table, EBS = Oracle E-Business Suite; OID = Oracle Internet Directory; OAM = Oracle Access Manager; SSWA = Self-Service Web Applications.

Self-Service Personal Home Page mode (APPLICATIONS_HOME_PAGE)

This profile determines the default home page for the application, which is the first page a user sees after logging into Oracle E-Business Suite.

Features of this profile:

- Available at site level only (cannot be set for individual users)
- Updatable only by system administrators
- Default value is 'Framework only'

Features of this profile:

Profile Value	Description
Framework only	Navigate to the Oracle E-Business Suite Release 12.2 home page
None	Do not use a personal home page.

Applications Local Login URL (APPS_LOCAL_LOGIN_URL)

This profile specifies which login page is used to perform local access to Oracle E-Business Suite. When the 'Applications SSO type' profile is set to 'SSWA', the application login servlet (AppsLogin) will redirect a user to the login page specified by this profile.

Features of this profile:

- Available at site level only (cannot be set for individual users)
- Updatable only by system administrators
- Default value is 'AppsLocalLogin.jsp'

Applications Portal (APPS_PORTAL)

This profile is used to specify Oracle Portal-related settings.

Features of this profile:

- Available at site level only (cannot be set for individual users)
- Updatable only by system administrators
- Defines the portal entry page

Applications Post-Logout URL (APPS_SSO_POSTLOGOUT_HOME_URL)

This profile can be used to specify where the user should be redirected after logging out of the Oracle E-Business Suite instance. Profile changes take effect for newly created sessions only.

Features of this profile:

- Available at site and user level
- Default value is NULL
- May be any valid URL

> **Note:** Product groups may programmatically set the post-logout URL, overriding any site or user level profile settings.

Profiles for Linking Accounts

The profile options described in this category control how Oracle E-Business Suite user accounts are linked to single sign-on accounts.

Applications SSO Auto Link User (APPS_SSO_AUTO_LINK_USER)

This profile determines whether Oracle E-Business Suite Release 12.2 will automatically link an authenticated single sign-on account to an application account of the same account name, without prompting the user for authentication information for the application account during login.

Features of this profile:

- Available at site level only (cannot be set for individual users)
- Updatable only by system administrators
- Has possible values of:
  - 'Enabled' – Allow auto link
  - 'Disabled' – Do not allow auto link (the default)
  - 'Create User and Link' - To create and link user on-demand

When automatic linking is enabled for users, they must meet two criteria: have the same name as the SSO user, and a USER_GUID of null or 1. FND Users with a different name, or with a USER_GUID that is not null and not 1, cannot be linked in this way.

> **Note:** As the user with GUID=1 cannot be linked on the fly, the only way to link this user is with APPS_SSO_AUTO_LINK_USER.

Applications SSO Link Same Names (APPS_SSO_LINK_SAME_NAMES)

This profile indicates whether the Oracle E-Business Suite Release 12.2 instance should link a newly-created Oracle E-Business Suite user to an existing Oracle Internet Directory account with the same name.

- Available at site level only (cannot be set for individual users)
- Updatable only by system administrators
- Has possible values of:
  - 'Enabled' – Link users with the same user name
  - 'Disabled' – Do not link users with the same user name

Applications SSO Allow Multiple Accounts (APPS_SSO_ALLOW_MULTIPLE_ACCOUNTS)

This profile indicates whether the Oracle E-Business Suite Release 12.2 instance allows linking of one Oracle Internet Directory user to multiple Oracle E-Business Suite user accounts.

Features of this profile:

- Available at site level only (cannot be set for individual users)
- Updatable only by system administrators
- Has possible values of:

- 'Y' – Allow multiple accounts to be linked
- 'N' – Do not allow multiple accounts to be linked (the default)

The 'Link additional account' operation uses this profile, which has the following implications:

- If the APPS_SSO_ALLOW_MULTIPLE_ACCOUNTS profile is set to 'Y' in the 'Single Sign-On Account Settings' page (accessible from the 'User Preferences' page), the 'Add Account' button will be shown.
- If the profile is set to the default value of 'N', the 'Add Account' button will not be shown, and the 'Link account' page will therefore not permit linking of multiple accounts.

Profiles for Password Settings

The profile options in this category specify how passwords are managed in a single sign-on Oracle E-Business Suite environment.

Applications SSO Login Types (APPS_SSO_LOCAL_LOGIN)

Features of this profile:

- Available at both site and user level (can be set for individual users)
- Updatable only by system administrators
- Determines whether a user's password is managed:
  - Externally in Oracle Internet Directory
  - Locally in Oracle E-Business Suite
  - In both Oracle Internet Directory and Oracle E-Business Suite

Valid values are defined in the Lookup Type, 'FND_SSO_LOCAL_LOGIN':

- 'SSO' – Login is only allowed through single sign-on. The password is set to 'EXTERNAL' after a single sign-on account and an application account are linked.
- 'LOCAL' – Login is only allowed via Oracle E-Business Suite local login. Passwords must be retained in the Oracle E-Business Suite and the account cannot be linked to any Oracle Internet Directory user.
- 'BOTH' – Login can be through both single sign-on and Oracle E-Business Suite. Since changes to the Oracle E-Business Suite password can be synchronized to Oracle Internet Directory, but not vice versa, a user's single sign-on password will

not necessarily be synchronized with his Oracle E-Business Suite password.

The default site level value is 'BOTH'. The user level value, applicable for example to the 'SYSADMIN' and 'GUEST' accounts, is set to 'LOCAL'.

The 'SYSADMIN' and 'GUEST' user profile options should not be changed. The 'SSADMIN' uer is a standard account that can only be used for local login, and cannot be used to log in via single sign-on. Once a password is set to 'EXTERNAL' Oracle E-Business Suite, it is no longer possible to use the original password to log in locally. For the password to be changed if the profile is updated to allow LOCAL access, the AFPASSWD utility or FNDCPASS utility will need to be run by a system administrator.

> **Important:** Regardless of whether the user credentials are correct, a 'LOCAL' user cannot be linked on the fly, and the linking page will display the error: *FND-9921: Unable to link account. This E-Business Suite user account is marked as a local account*. The user can then choose to enter a different (non-local) account to link to.

For more information about the AFPASSWD and FNDCPASS utilities, refer to the *Basic DBA Tasks* chapter of *Oracle E-Business Suite Maintenance Guide*.

Applications Local Change Password URL (APPS_LOCAL_CHANGE_PWD_URL)

This profile stores the location of the page where Self-Service users can change their Oracle E-Business Suite password. The page specified should only allow the password to be changed by a user whose 'APPS_SSO_LOCAL_LOGIN' profile has the value of either 'BOTH' or 'LOCAL' (i.e. not 'SSO').

> **Note:** For 'SSO' and 'Both' users an API is used to determine whether the password can be changed locally, or if the APPS_SSO_CHANGE_PWD URL should be used. The criteria are whether the password can be synchronized to OID.

Features of this profile:

- Available at site level only (cannot be set for individual users)
- Updatable only by system administrators
- Default value is 'AppsChangePassword.jsp'

Applications SSO Change Password URL (APPS_SSO_CHANGE_PWD_URL)

This profile points to the LDAP self-service user interface for password changes. When an Oracle E-Business Suite Self-Service change password page determines that a user's password in stored in LDAP, it can redirect the user to the location stored in this profile.

For example, the password may be stored in Oracle Identity Management.

Features of this profile:

- Available at site level only (cannot be set for individual users)
- Updatable only by system administrators

Profiles for Provisioning Settings

The profile options in this category determine how provisioning (automatic updating of user accounts) is carried out in a single sign-on Oracle E-Business Suite environment.

Applications SSO LDAP Synchronization (APPS_SSO_LDAP_SYNC)

This profile determines whether provisioning is enabled for a particular FND_USER account. User information associated with an FND_USER account will be provisioned with Oracle Internet Directory only if the APPS_SSO_LDAP_SYNC profile of the user is set to 'Y'.

Features of this profile:

- Available at site and user level (can be set for individual users)
- System administrators can change setting at both site and user levels
- End users can change this from user level profiles.
- Default site level value is 'Y'
- User level values for 'SYSADMIN' and 'GUEST' accounts are set to 'N'

The site level value is provided to obviate the need for every user to define a user level value, and has the following important characteristics:

- Setting the site level value (to 'Y' or 'N') does not globally enable (or disable) provisioning.
- Since provisioning with Oracle Internet Directory is the most common deployment scenario, this profile is shipped with a default site level value of 'Y'.
- For any user accounts that are not to be provisioned, this profile should be overridden with a user level value of 'N'.
- To provision users from FND to Oracle Internet Directory, APPS_SSO_LDAP_SYNC needs to be enabled and the Oracle Internet Directory provisioning profile set.
- If an existing user's APPS_SSO_LOCAL_LOGIN profile has 'LOCAL' value, the

user modifications are *not* provisioned, regardless of this profile value. Profile APPS_SSO_LOCAL_LOGIN has higher precedence than APPS_SSO_LDAP_SYNC at user level.

> **Important:** Linking a single enterprise user account to multiple Oracle E-Business Suite (FND_USER) user accounts can have undesirable consequences, such as data from one application overwriting data from another. Therefore, after the first FND_USER account is linked, all accounts subsequently linked to the same enterprise account will have the APPS_SSO_LDAP_SYNC user level profile value set to 'N'. Users who still wish to change the user level value of this profile can do so via the 'Single Sign-On Account Settings' page.

Applications SSO Enable OID Identity Add Event (APPS_SSO_OID_IDENTITY)

This profile determines whether users created in Oracle Internet Directory are automatically created in Oracle E-Business Suite and subscribed to the given Oracle E-Business Suite instance. You can enable this profile to allow the automatic subscriptions for users created in Oracle Internet Directory.

Features of this profile:

- Available at site level only (avoids the need for every user to define a user level value)
- System administrators can change setting at site level
- Default site level value is 'Disabled'

The default site level value of 'Disabled' means that users created in Oracle Internet Directory will not be automatically created in Oracle E-Business Suite. The reason for this is that significant numbers of users from different sources may be created in Oracle Internet Directory quite rapidly, and typically not all will also need to be created in Oracle E-Business Suite.

When the profile 'Applications SSO Enable OID Identity Add Event' value is set to 'Enabled', users created in Oracle Internet Directory are automatically both created in Oracle E-Business Suite and subscribed to the Oracle E-Business Suite instance.

Applications SSO User Creation And Updating Allowed (APPS_SSO_USER_CREATE_UPDATE)

This profile is for Oracle internal use only.

Configuring Directory Integration Platform Provisioning Templates

This section describes how to configure an Oracle E-Business Suite Release 12.2 instance as a provisioning integrated application with Oracle Access Manager. The goal is to

keep user information synchronized between Oracle Internet Directory and Oracle E-Business Suite Release 12.

Configure and Create a Provisioning Profile

Bidirectional provisioning between Oracle E-Business Suite and Oracle Internet Directory is built around the Oracle Directory Integration Platform, as described further in the Oracle Internet Directory Release Administrator's Guide.

A key feature of this solution is the provisioning integration service, which enables automatic provisioning (updating between the systems) of account creation or changes of user attributes. The provisioning process between each Oracle E-Business Suite instance and Oracle Internet Directory is controlled by a provisioning profile.

When changes are made in Oracle Internet Directory that match an application's provisioning profile event subscription criteria, the Provisioning Integration Service is the agent that sends the relevant new data to that application. Going in the other direction, the Provisioning Integration Service filters changes coming from an application (according to the application's provisioning profile's permitted events criteria), and transmits applicable ones to Oracle Internet Directory.

One of the advantages of this solution is a high level of flexibility at deployment time, i.e. the provisioning profile is highly customizable. Configuration of the profile is carried out by either using the `oidprovtool`, or by instantiating an LDIF template file that contains the requisite values for the particular deployment.

Profile Creation Prerequisites

Before a profile can be created, the relevant Oracle E-Business Suite instance must be registered with Oracle Internet Directory. This involves creating a unique application identity for the instance in Oracle Internet Directory.

Oracle E-Business Suite instances are created at the following location in the directory information tree (DIT): "cn=E-Business,cn=Products,cn=OracleContext, <Identity Management Realm>"

The created application identity (combination of dn and password) also needs to be stored in Oracle E-Business Suite. Note that the registered application identity and password can be used by the application administrator to connect to Oracle Internet Directory for certain tasks, such as querying the provisioned profile details between this application instance and Oracle Internet Directory.

Provisioning Profiles - Configuring Provisioning Events

CREATION, MODIFICATION, and DELETION events can be enabled or disabled individually. Four event types are currently used:

- SUBSCRIPTION_ADD
- IDENTITY_ADD

- IDENTITY_MODIFY
- IDENTITY_DELETE

Each of these is described below:

SUBSCRIPTION_ADD

This event is generated by either Oracle Internet Directory or Oracle E-Business Suite Release 12.

Oracle Internet Directory maintains a subscription list for each Oracle E-Business instance that has registered with Oracle Internet Directory. The subscription list maintains a list of all single sign-on user accounts that need to access the associated Oracle E-Business Suite instance.

- Oracle Internet Directory and the associated Oracle E-Business Suite instance jointly maintain the accuracy of the subscription list.
- When a single sign-on account is created in Oracle Internet Directory, and subsequently added to the subscription list of an Oracle E-Business Suite instance (see "Manual Subscription Management With Provsubtool" for how this is done), a SUBSCRIPTION_ADD event is generated in Oracle Internet Directory. If this event is enabled in the Oracle Internet Directory to Oracle E-Business Suite direction, a new application account will be created and linked to the single sign-on account.
- When Oracle Internet Directory receives an IDENTITY_ADD event (see below) from an Oracle E-Business Suite instance, it adds the user to the subscription list of that Oracle E-Business Suite instance.
- When Link-on-the-Fly is performed on an Oracle E-Business Suite Release 12 instance, the Oracle E-Business Suite instance will send a SUBSCRIPTION_ADD event to Oracle Internet Directory.
- When an IDENTITY_MODIFY (see below) event is generated in Oracle Internet Directory, Oracle Internet Directory will check the subscription lists of all registered Oracle E-Business Suite Release 12 instances, and only send the event to an Oracle E-Business Release 12 instance if the modified user appears on its subscription list.

IDENTITY_ADD

This event is generated by either Oracle E-Business Suite or Oracle Internet Directory when a new user is created. If this event is enabled from Oracle E-Business Suite to Oracle Internet Directory direction, after Oracle Internet Directory receives this event, it will create an Oracle single sign-on account in Oracle Internet Directory and add the account to the subscription list of that Oracle E-Business Suite Release 12 instance. The other way, if this event is enabled from Oracle Internet Directory to E-Business Suite and profile 'Applications SSO Enable OID Identity Add Event' is 'Enabled', it has the same affect as SUBSCRIPTION_ADD event generated by Oracle Internet Directory.

IDENTITY_MODIFY

This event is generated by either Oracle Internet Directory or Oracle E-Business Suite when a user account is modified. If this event is enabled in either direction, the receiving system will apply the modification to the account on that system.

IDENTITY_DELETE

This event is generated by Oracle Internet Directory when an Oracle single sign-on account is deleted. If this event is enabled from the Oracle Internet Directory to Oracle E-Business Suite direction, after an Oracle E-Business Suite Release 12 instance receives this event, it will end-date the application account linked to the Oracle single sign-on account.

Provisioning Direction

Each event can be enabled in:

- One direction:
  - From Oracle Internet Directory to Oracle E-Business Suite only
  - From Oracle E-Business Suite to Oracle Internet Directory only
- Both directions:
  - From Oracle Internet Directory to Oracle E-Business Suite
  - From Oracle E-Business Suite to Oracle Internet Directory

Attribute List

For each direction, and each type of event, the list of provisioned attributes can be customized as required (removing an attribute from the attribute list would disable sending that attribute). The "Supported Attributes" section lists the attributes that are currently supported for each direction, and also as the mapping between Oracle Internet Directory attributes and application table and column names.

Polling Interval

By default, Oracle Internet Directory sends out provisioning events every 60 seconds; this value can be increased or decreased by using oidprovtool, or by editing the `orclodipprofileschedule` attribute value in the provisioning template (see below). The polling interval should be set with caution; provisioning that is not frequent enough for site activity may have an impact on operations, while provisioning that is more frequent than necessary will result in needless network traffic.

Creating a Profile

Once the values of the configurable variables for a profile have been decided, there are two methods available to create the profile in Oracle Internet Directory. The first is `oidProvTool` (see Appendix A of the *Oracle Internet Directory Administrator's Guide*

Release 10g). The second option is to instantiate an LDIF template, which captures the configuration choices. The instantiated templates can then be loaded into Oracle Internet Directory using the `ldapmodify` command. The template method is described in detail below.

Creating a Profile From a Provisioning Template

Creating the provisioning profile consists of the following steps:

1. Create a suitable template based on deployment choices. The sample templates shipped can be used as examples and starting points.
2. Instantiate the template with deployment specific values, to generate an LDIF file.
3. Load the LDIF file into Oracle Internet Directory.

Once the LDIF file is loaded, Oracle Internet Directory will start sending and polling provisioning events to and from the Oracle E-Business Suite instance for which the profile was created. It takes the provisioning service approximately two minutes to detect that a new profile has been added or an existing one has changed. The new or updated profile is then read by the service.

Four types of provisioning are provided by the registration utility:

- **BiDirectional Provisioning**: Set by specifying "-provisiontype=1" as a command line argument during OID registration. This is the default provisioning type set by the registration utility.
- **InBound Provisioning**: Set by specifying "-provisiontype=2" as a command line argument during OID registration
- **OutBound Provisioning**: Set by specifying "-provisiontype=3" as a command line argument during OID registration.
- **BiDiNoCreation Provisioning**: Set by specifying "-provisiontype=4" as a command line argument during OID registration.

To decide on the right template to use, an Oracle E-Business Suite administrator needs to determine the direction or directions of provisioning, and which provisioning events need to be enabled in each direction. The deployment scenarios discussed in this section may be used as a reference.

For example, if the Oracle E-Business Suite instance only needs to send events to Oracle Internet Directory, then an INBOUND provisioning profile should be created. If the Oracle E-Business Suite instance only needs to receive provisioning events from Oracle Internet Directory, then an OUTBOUND profile should be created.

If provisioning events may need to be sent in both directions, a bidirectional profile (BOTH) should be created.

Note: Oracle recommends using the base provisioning profile templates provided with Oracle E-Business Suite. Best-efforts support will be provided for customizations to the standard provisioning profile templates. Customers may wish to engage Oracle Consulting for assistance with specific customization requirements and issues.

Administering the Provisioning Process

The monitoring and other administration tasks for the provisioning process are normally performed by Oracle Internet Directory system administrators. Refer to Oracle Internet Directory Release Administrator's Guide for more details.

Each of the following sections is denoted with OID (for topics related to OID) or EBS (for topics related to E-Business Suite).

Maintaining DIP Server Log Files (OID)

The main DIP log file is located in the `$ORACLE_HOME/ldap/log/odisrv<instance number>.log` directory. The <instance number> is a unique integer id, e.g. 1, assigned by a system administrator when specifying the instance parameter as part of the oidctl command line used to start the DIP server.

The provisioning profile logs are located in the `$ORACLE_HOME/ldap/odi/log directory`. Each log file name is of the form: `<ApplicationName>_<RealmName>_[I/E].[trc/aud].`

where:

- I = INBOUND provisioning event (from Oracle E-Business Suite to Oracle Internet Directory)
- E = OUTBOUND provisioning event (from Oracle Internet Directory to Oracle E-Business Suite)
- .trc = Trace file, which grows until the file size is approximately 10MB. When the maximum file size is reached, the current trace file is backed up (and a timestamp appended) and a new trace file started. All old trace files are kept in the same directory.
- .aud = Audit file, which records all the events from the time the profile was created and therefore grows continually. This file consequently needs to be archived periodically. The system administrator needs institute a policy to back up and archive audit files. This will involve temporarily disabling the profile, archiving the audit file, then re-enabling the profile. If archiving is not required, the old audit file can simply be deleted.

Note: For more information, refer to *Oracle Internet Directory Release Administrator's Guide*.

Enabling or Disabling a Profile (OID)

Use the `oidProvTool`. Refer to the *Oracle Internet Directory Administrator's Guide, Release 10g* for usage of this tool.

Changing Profile Characteristics in an Existing Deployment (OID)

If any properties of the provisioning profile are to be changed, the following steps must be performed.

1. Delete the existing profile, using `oidProvTool`.

2. Use `oidProvTool` to create a new profile that suits the current requirements.

The DIP server may take approximately two minutes to detect changes to the provisioning profile entries, i.e. read the new profile configuration entry and then begin processing events based on the new configuration.

Creating Custom Workflow Subscriptions (EBS)

Customization of data synchronized between Oracle Internet Directory and the Oracle E-Business Suite can be achieved by creating custom Workflow Business Event Subscriptions.

The required steps are:

1. Create the procedure that creates or updates the desired attributes. See example code below.

2. Create a new subscription for the relevant Workflow Business Event. Listed below are the Business Events provided, and how they are used:

   - **oracle.apps.global.user.change** – this event is raised whenever a FND_USER is updated by any source.

   - **oracle.apps.fnd.identity.add** – this event is raised whenever the E-Business Suite instance receives an IDENTITY_ADD event from OID, i.e. when a new user is created in OID.

   - **oracle.apps.fnd.identity.modify** – this event is raised whenever the E-Business Suite instance receives an IDENTITY_MODIFY event from OID, i.e. when a user is updated in OID.

- **oracle.apps.fnd.identity.delete** – this event is raised whenever the E-Business Suite instance receives an IDENTITY_DELETE event from OID, i.e. when a user is deleted from OID.

- **oracle.apps.fnd.subscription.add** – this event is raised whenever the E-Business Suite instance receives a SUBSCRIPTION_ADD event from OID, i.e. when a user added to the subscription list in OID.

- **oracle.apps.fnd.subscription.delete** – this event is raised whenever the E-Business Suite instance receives a SUBSCRIPTION_DELETE event from OID, i.e. when a user is deleted from the subscription list in OID. Currently, this subscription does nothing in the E-Business Suite. Administrators may customize this behavior by adding their own subscriptions.

- **oracle.apps.fnd.ondemand.create** – this event is raised when a user is created on demand from SSO.

Example code for a custom Workflow subscription rule function

```
create or replace package custom_update_user AS
  function disable_fnd_user (p_subscription_guid   in   raw,
                             p_event   in out nocopy wf_event_t)
return varchar2;
end custom_update_user;


create or replace package body custom_update_user as

function disable_fnd_user (p_subscription_guid in  raw,
                           p_event   in out nocopy wf_event_t)
return varchar2 is

l_event_name          varchar2(256);
l_event_key           varchar2(256);
l_change_source       varchar2(256);
l_change_source       varchar2(256);
l_orcl_guid           fnd_user.user_guid%type;
l_ent_type            varchar2(256);
l_oid_user_enabled    boolean;
l_end_date            date;

if (p_event.GetValueForParameter('CHANGE_SOURCE') = 'OID') then
        l_event_key := p_event.GetEventKey();
        l_ent_type :=
wf_entity_mgr.get_entity_type(p_event.GetEventName());
        l_orcl_guid      :=
wf_entity_mgr.get_attribute_value(l_ent_type, l_event_key, 'ORCLGUID');
        l_end_date := wf_entity_mgr.get_attribute_value(l_ent_type,
l_event_key, 'ORCLACTIVEENDDATE');
     if (l_end_date <= sysdate) then
           fnd_user_pkg.DisableUser(username => l_event_key);
     end if;
end if;
        return(wf_rule.default_rule(p_subscription_guid, p_event));

exception when others
then
      return(wf_rule.error_rule(p_subscription_guid, p_event));
end disable_fnd_user;

end custom_update_user;
```

Customizing SSO Workflow Business Events (EBS)

Oracle Internet Directory provisioning events are processed in Oracle E-Business Suite using Workflow Business Events. The Workflow Business Events have subscriptions that are enabled by default and if disabled will change the default behavior. The event subscriptions that an administrator may want to disable are:

- **Event:** oracle.apps.fnd.identity.add **Subscription:** assign_def_resp

  This event subscription will add the default responsibility "Preferences" when provisioning a new user from Oracle Internet Directory to Oracle E-Business Suite.

- **Event:** oracle.apps.fnd.identity.add **Subscription:** hz_identity_add

  This event subscription will create TCA records when provisioning a new user from Oracle Internet Directory to Oracle E-Business Suite.

- **Event:** oracle.apps.fnd.identity.modify **Subscription:** hz_identity_modify

  This event subscription will modify TCA records when updates are made to a user in Oracle Internet Directory.

Maintaining the Workflow Attribute Cache (EBS)

Data is synchronized between Oracle Internet Directory and Oracle E-Business Suite using a Workflow attribute cache. The data resides in this table until manually removed by the system administrator. It is recommended that periodically the API WF_ENTITY_MGR.FLUSH_CACHE should be executed to remove obsolete data. This API deletes cached records that match the specified entity information provided. When passing a specific entity_type (for example, 'USER'), the specific entity_key_value should also be passed. The special entity_type "*ALL*" will truncate the entire table.

Parameters for procedure wf_entity_mgr.flush_cache

Name	Type	Direction	Default	Description
p_entity_type	varchar2	In	Null	Entity type to be deleted, for example 'USER'
p_entity_key_value	varchar2	In	Null	Entity value to be deleted, for example 'SCOTT'

Changing E-Business Suite Database Account Password

The APPS database account password is used to register a provisioning profile in Oracle Internet Directory for a specific Oracle E-Business Suite instance. If the APPS database account password for that instance is changed using the AFPASSWD utility or FNDCPASS utility, the Oracle Internet Directory provisioning profile must to be updated with the new information. This can be done by running the Oracle Internet Directory `oidprovtool` command-line utility.

For more information about the AFPASSWD and FNDCPASS utilities, refer to the *Basic DBA Tasks* chapter of *Oracle E-Business Suite Maintenance Guide*..

oidprovtool Usage

The command syntax for this tool is:

```
oidprovtool operation=modify \
ldap_host=<OID Server hostname> \
ldap_port=<OID Server Port> \
ldap_user_dn="cn=orcladmin" \
application_dn="<LDAP distinguished name of application>" \
interface_connect_info=<Oracle E-Business Suite connect info of the
format, host:port:Sid:username:password>
```

For example:

```
oidprovtool operation=modify \
ldap_host=infra30qa  ldap_port=3060 \
ldap_user_dn=cn="orcladmin" \
application_dn="orclApplicationCommonName=ebizqa,cn=EBusiness,cn=Product
s,cn=OracleContext,dc=com" \
interface_connect_info=ebiz30qa:1521:ebizqa:apps:password
```

Example output:

```
orclODIPProfileName=EA3EFF8640819A51F0301990304E5D0B_EA960F743D5D7552F03
01990304E34B3, cn=Provisioning Profiles, cn=Changelog
Subscriber,cn=Oracle Internet Directory
The Provisioning Profile for the Application has been modified.
```

For further details about the `oidprovtool` utility, see: *Oracle Internet Directory Administrator's Guide*.

Manual Subscription Management With Provsubtool

Provsubtool Subscription Management Tool

Depending on how your Oracle E-Business Suite Single Sign-On profile options have been configured, it may be necessary to manage subscriptions for some of your users manually.

The Oracle Internet Directory `provsubtool` command-line utility is used to manage application-specific subscription lists in Oracle Internet Directory. The tool can be used by the application administrator or the Identity Management Realm administrator (such as orcladmin).

In case you do not have execute permission to the tool shipped as `$ORACLE_HOME/ldap/odi/bin/provsubtool.orc`, the file should be copied to $ORACLE_HOME/bin or another suitable location for which you have both write and execute permissions.

Specific uses of this tool are to:

- Add or remove users from application-specific subscription lists in bulk mode or batch mode.

- Add users to the application-specific subscription lists when 'Applications SSO Enable OID Identity Add Event' profile value is 'Disabled'. This profile controls the automatic subscription for users created in Oracle Internet Directory.
- List the memberships of a particular subscription list for an application.
- Read from a file of a list of simple user login names (nickname attribute values) or user DNs and add or remove them from the appropriate subscription list as specified.

Command Line Parameters

Parameter Name	Required or Optional	Default Value	Parameter Description
LDAP_HOST	Optional	Local host	LDAP server host
LDAP_PORT	Optional	389	LDAP Server port
APP_DN	Required	None	Application Identity DN, for example: `orclapplicationcommonname=Financials,cn=EBusiness,cn=Products,cn=OracleContext,<Identity Realm>`
APP_PWD	Required	None	Application DN password
REALM_DN	Required	None	DN of the identity Management Realm, for example: `dc=ganseycorp,dc=com`
LIST_NAME	Optional	ACCOUNTS	The Subscription List Name. By default, ACCOUNTS is created for Oracle E-Business Suite instances.
OPERATION	Required	None	ADD, REMOVE, LIST. The LIST option will list all the current members of the subscription list.
FILE_NAME	Optional	members.lst	File containing the user list either as simple names or DNs
FILE_TYPE	Optional	0	0 = Simple Names 1 = DNs

Parameter Name	Required or Optional	Default Value	Parameter Description
LOG_FILE	Optional	report.log	Output log file. The output from the command is written to a file specified by the parameter "LOG_FILE". If no filename is specified, the default of report.log is used.
DEBUG	Optional	0	Debugging On/Off (0 or 1)
MAX_ERRORS	Optional	1000	Abort operation after this number of errors have occurred. If the numbers of errors exceed the value specified by the " MAX_ERRORS" parameter (during a bulk operation when trying to add many users together in a batch), the command will fail.

Manually Adding and Removing Users

For an Oracle Financials E-Business Suite instance registered in Oracle Internet Directory as: `orclapplicationcommonname=Financials,cn=EBusiness,cn=Products,cn=OracleContext,<Identity Realm>` for the ID realm: `dc=ganseycorp,dc=com`

To add a user whose nickname is "john.smith" to the default subscription list "ACCOUNTS", you would add the line "john.smith" (without the quotes) to an input file, in this case with the default name of `members.lst`, and then execute the command:

```
provsubtool ldap_host=LDAP_HOST ldap_port=LDAP_PORT \
app_dn="orclapplicationcommonname=Financials,cn=EBusiness,\
cn=Products,cn=OracleContext,dc=ganseycorp,dc=com" \
realm_dn="dc=ganseycorp,dc=com"
list_name=ACCOUNTS \
operation=ADD \
file_name=members.lst
file_type=0 \
app_pwd=tea4two
```

To remove a user, you would follow the same procedure, simply substituting the operation REMOVE for the operation ADD:

```
provsubtool ldap_host=LDAP_HOST ldap_port=LDAP_PORT \
app_dn="orclapplicationcommonname=Financials,cn=EBusiness,cn=Products,cn
=OracleContext,dc=ganseycorp,dc=com" \
realm_dn="dc=ganseycorp,dc=com"
list_name=ACCOUNTS \
operation=REMOVE \
file_name=members.lst
file_type=0 \
app_pwd=tea4two
```

Migrating Data between Oracle E-Business Suite and Oracle Internet Directory

The Oracle E-Business Suite Release 12.2 user migration utilities include:

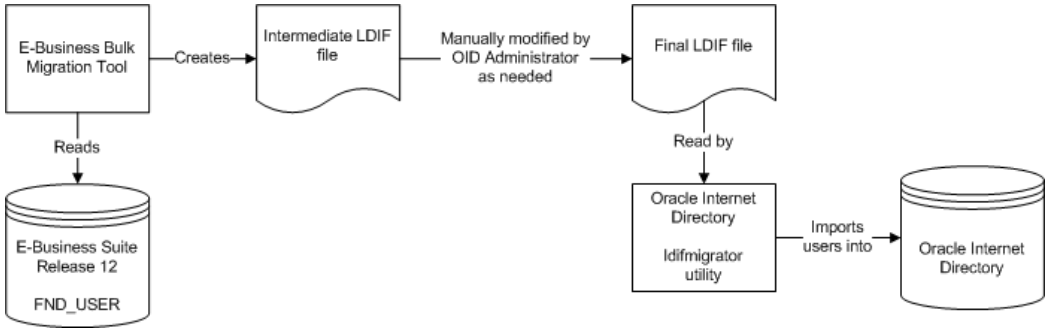
- The *AppsUserExport* tool, which exports existing application accounts from Oracle E-Business Suite Release 12.2 into an intermediate LDIF file. This tool is a Java program that is invoked from the command line on an Oracle E-Business Suite application tier machine.

- The *LDAPUserImport* tool, which reads an LDIF file, creates new Oracle E-Business Suite application accounts as needed, and imports the data. This tool is invoked from the command line. LDAPUserImport is provided for bulk migration of existing Oracle Internet Directory accounts into Oracle E-Business Suite Release 12.2.

See below for details of the migration process between Oracle E-Business Suite Release 12.2 and Oracle Internet Directory, and the usage of these tools.

Migrating Existing Application Accounts in Oracle E-Business Suite Release 12.2 to Oracle Internet Directory

An Oracle E-Business Suite administrator can use AppsUserExport to export a selected set of application accounts from the Oracle E-Business Suite native user directory (FND_USER) into an intermediate LDIF file. An Oracle Internet Directory administrator then uses the Oracle Internet Directory `ldifmigrator` utility to convert this intermediate LDIF file into a final LDIF file, based on Oracle Internet Directory deployment choices. The Oracle Internet Directory administrator then loads the final LDIF file into Oracle Internet Directory using the `bulkload` utility.



The migration process and intermediate LDIF format are explained further in the section *Migrating Data from Other Directories* in Oracle Internet Directory Administrator's Guide. In addition, usage of the `ldifmigrator` tool is described in Oracle Identity Management User Reference.

Note: If you plan to implement a password hashing scheme for Oracle E-Business Suite user passwords, you should perform the export of application accounts and bulk load to Oracle Internet Directory before you implement the password hashing scheme. After you implement password hashing, the AppsUserExport utility can no longer include the passwords when exporting Oracle E-Business Suite user information. See: Using AFPASSWD to Migrate to a Password Hashing Scheme, *Oracle E-Business Suite Maintenance Guide*.

If you have already implemented password hashing but you want to export Oracle E-Business Suite users to Oracle Internet Directory, then the following options are available.

- **Option 1**
  1. Ensure that passwords can be synchronized from Oracle E-Business Suite to Oracle Internet Directory by setting the Oracle Internet Directory provisioning profile for deployment from Oracle E-Business Suite to Oracle Internet Directory and enabling the Applications SSO LDAP Synchronization (APPS_SSO_LDAP_SYNC) profile option.
  2. Run AppsUserExport with the `-g` option to create and copy your users' GUID values and import them into Oracle Internet Directory.
  3. Expire these users' passwords using the AFCPEXPIRE.sql script.
  4. Direct users to log in to Oracle E-Business Suite using the local login and change their passwords. Once a user's password is reset, the user should be able to log in through single sign-on.
- **Option 2**
  1. Export users using AppsUserExport without the `-g` option and import the users into Oracle Internet Directory.
  2. The Identity Management administrator should determine the best approach to set the users' passwords and communicate this approach to the users.

The next section focuses on application-specific tasks.

Task 1: Exporting Application Accounts into Intermediate LDIF File

Determine which accounts to migrate

Having determined which accounts to export, the application administrator can then specify whether an account is migrated by utilizing the following profiles:

- **Applications SSO Login Types (APPS_SSO_LOCAL_LOGIN)** – An account will not be migrated if the user level profile value of the account is 'LOCAL', i.e. the account is a local account.

- **Applications SSO LDAP Synchronization (APPS_SSO_LDAP_SYNC)** – An account will not be migrated if the user level profile value of the account is 'N', i.e. the account is marked to *not* synchronize with Oracle Internet Directory.

Oracle E-Business Suite ships a number of standard accounts, such as SYSADMIN and GUEST. These accounts should not be migrated. To enforce this, the SYSADMIN and GUEST accounts are pre-seeded with Applications SSO Login Types (APPS_SSO_LOCAL_LOGIN) set to 'LOCAL' and Applications SSO LDAP Synchronization (APPS_SSO_LDAP_SYNC) set to 'N'.

> **Important:** Accounts with user_id less than 10 can only be logged into locally, and **not** via single sign-on (you can check for these with the query `select user_name from FND_USER where user_id < 10`).

Use AppsUserExport to extract user information

Use the AppsUserExport tool to extract application user information into an intermediate LDIF file. This tool is invoked from the command line.

> **Note:** The list of attributes migrated to Oracle Internet Directory from the E-Business Suite is currently limited to those listed in "Supported Attributes".

To invoke the AppsUserExport tool, ensure your environment is set up correctly, and use the following syntax. Note that all parameters can if desired be entered on the same command line; they are shown here on different lines (using the UNIX '\' continuation character) for clarity.

```
java oracle.apps.fnd.oid.AppsUserExport \ [-v]
-dbc <dbcfile> \
-o <outputfile> \
-pwd <apps schema pwd> \
-g
[-l <logfile>]
```

where:

`[-v]` - Run in verbose mode

`<dbcfile>` - Full path to the dbcfile

`<outputfile>` - Intermediate LDIF file

`<apps schema pwd>` - Apps schema password

`-g` - Create and copy users GUIDs to OID

`<logfile>` - Log file (default is <outputfile>.log)

For example:

```
java oracle.apps.fnd.oid.AppsUserExport \
-v \
-dbc $FND_SECURE/myebiz.dbc \
-o users.txt \
-pwd password \
-g \
-l users.log
```

> **Warning:** The resulting data file and log file may contain confidential information, such as the start and end dates for a user's account, and should therefore be secured appropriately.

Task 2: Converting Intermediate LDIF File to Final LDIF File

Before performing loading data into Oracle Internet Directory, the Oracle Internet Directory administrator needs to ensure that:

- The extracted data file is copied from the Oracle E-Business Suite instance to Oracle Internet Directory.

- If the provisioning profile has been set up for the Oracle E-Business Suite instance and the profile mode is either OUTBOUND or BOTH (i.e. you have enabled any provisioning events from Oracle Internet Directory to Oracle E-Business Suite), the profile will need to be temporarily disabled during the migration process.

  To convert the intermediate LDIF file to the final LDIF format:

  The *intermediate LDIF file* created by AppsUserExport has two variables that an Oracle Internet Directory administrator needs to instantiate using the Oracle Internet Directory `ldifmigrator` utility:

  - **s_UserContainerDN** – DN of the entry under which all users are added, for example cn=users,dc=us,dc=oracle,dc=com.

  - **s_UserNicknameAttribute** – The nickname attribute used for user entries in the subscriber, such as uid.

For example:

```
ldifmigrator "input_file=data.txt" \
"output_file=data.ldif" \
"s_UserContainerDN=cn=users,dc=us,dc=oracle,dc=com" \
"s_UserNicknameAttribute=uid"
```

> **Important:** Note that the variable names above are case sensitive.

If you encounter problems running any of the Oracle Internet Directory command line

tools such as `oidprovtool` or `ldapsearch`, refer to the Oracle Internet Directory Administrator's Guide for more information.

Task 3: Loading Final LDIF file into Oracle Internet Directory

Once the final LDIF file has been generated, the user data is ready to be loaded into Oracle Internet Directory using the Oracle Internet Directory `bulkload` tool. This section describes the minimum command-line options required to perform this task; note that additional options exist for more advanced requirements.

> **Note:** For further details, see the section *Using Bulk Tools* in Oracle Internet Directory Administrator's Guide.

Before performing a bulk load:

1. Use `oidprovtool` with operation=DISABLE to disable the profile before the migration is started. For example:

```
oidprovtool operation=disable \
ldap_host=testsys1.example.com \
ldap_port=3060 \
ldap_user_dn=cn=orcladmin \
application_dn="
orclApplicationCommonName=beta,cn=EBusiness,cn=Products,cn=OracleCon
text,dc=example,dc=com" \
profile_mode=BOTH
```

> **Important:** Do not add spaces after any of the commas in the application_dn parameter.

2. Before using the `bulkload` utility to load the LDIF file, stop all OID processes by running the following command from the OID Oracle Home:

   ```
   $ORACLE_HOME/opmn/bin/opmnctl stopall
   ```

   Note the OID password, which should be the same as the instance and orcladmin passwords. You will be prompted for this when running the utility.

3. You must ensure that no OID processes are running before continuing with the `bulkload` command. If any other OID processes such as `odisrv` are still running, stop them manually using the command:

   ```
   oidctl connect=<SID> server=<servername> instance=<#> stop
   ```

The user namespaces contained in an LDIF file that is to be bulk loaded must be unique and non-overlapping. When bulk loading users into OID, the potential for *collisions* (duplicate users) exists. Collisions can result when integrating multiple sources into a single OID instance, or by running the `bulkload` utility more than once for the same LDIF file. As collisions can lead to numerous problems, you should follow the steps below to ensure that they do not occur:

1. Run the `bulkload` utility with the `check` and `generate` options to verify that

there are no duplicate users. For example:

```
bulkload connect=<connect string> check=true generate=true
file=<full path to LDIF file>
```

2. Check the log file for duplicate users.

3. If the log file indicates duplicate users, manually remove these users from the LDIF file.

4. Rerun Step 1 to verify all duplicates have been successfully removed.

5. Once all duplicates are removed, run the `bulkload` utility with the `-load` option to load the users.

   For example:

   ```
   bulkload connect=<connect string> load=true file=<full path
   to LDIF file>
   ```

   Note: For further details of the `bulkload` utility, see the relevant version of Oracle Internet Directory Administrator's Guide.

Importing Multiple LDIF Files

It is possible to use `bulkload` to import multiple LDIF files. The most common scenario is one in which multiple LDIF files are generated from different Oracle E-Business Suite instances. Consolidating user information from each Oracle E-Business Suite instance into a single Oracle Internet Directory can reduce the administrative overhead of managing multiple user repositories.

The user namespaces from each Oracle E-Business Suite instance's LDIF file must be unique and non-overlapping. For example, if username "John.Brown" exists in the LDIF file to be imported from Oracle E-Business Suite instance A, it must not exist in the LDIF file to be imported from Oracle E-Business Suite instance B. If these usernames do not correspond to the same user, then the username should be updated in Oracle E-Business Suite instance B. This will both distinguish between the two users and eliminate the duplication. Otherwise, the username must be removed from the LDIF file from instance B.

Once the LDIF file for Oracle E-Business Suite instance A has been bulk loaded into OID, then the procedure should be done for the LDIF file for Oracle E-Business Suite instance B. By removing the duplicate users from the LDIF file, only the unique users from Oracle E-Business Suite instance B should bulk-loaded into OID. If a third Oracle E-Business Suite instance is to be bulk-loaded, the same procedure should be carried out: after removing the duplicate users from the LDIF file, only the users unique to Oracle E-Business Suite instance C will be bulk-loaded into OID.

Using ldapadd instead of bulkload

For small amounts of data, you may use the `ldapadd` tool instead of the `bulkload`

tool. For example:

```
ldapadd -h <ldaphost> -p <ldapport> -D "cn=orcladmin" -w <password> -f
data.ldif -v
```

> **Warning:** As some operating systems also include an ldapadd executable (which will not work with OID), it is advisable to specify the full path of $ORACLE_HOME/bin/ldapadd to ensure the correct one is used.

The main practical difference between these two tools is that `bulkload` is optimized for rapid processing of large numbers (possibly hundreds of thousands) of userid changes, whereas `ldapadd` is intended for making a small number of changes one by one.

For further details about using `ldapadd`, see *Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory*.

Sample Intermediate LDIF File

The following sample is an excerpt from an intermediate LDIF file:

```
# user name = 001
dn:: Y249MDAxLCAlc19Vc2VyQ29udGFpbmVyRE4l
sn:: MDAx
%s_UserNicknameAttribute%:: MDAx
description:: VGVzdGluZyBPSUQgc3luYw==
mail:: MDAxQG9yYWNsZS5jb20=
facsimileTelephoneNumber:: NjUwLTU1NS0xMTEx
orclActiveStartDate: 2003040316242131
orclIsEnabled: ENABLED
userPassword: {MD5}IB8AtcpdZaHBGOXjJDFRTA==
orclGuid: B9A5009B1603A500E030028A9F9E7C98
objectClass: inetOrgPerson
objectClass: orclUserV2
```

Password Restrictions and Bulk Loading

- Passwords stored in Oracle Internet Directory are case-sensitive. Mixed-case passwords in Oracle E-Business Suite are migrated with the case preserved.

- The passwords in the LDIF file are encrypted using the MD5 hashing method If errors occur while importing the LDIF file into OID check the hashing method used by OID. If is not MD5, using ODM reset the import hashing method to MD5 and try importing the LDIF file.

- When you export users from Oracle E-Business Suite and create an LDIF file, the passwords are encrypted and so the bulk loader cannot verify if they follow OID password policy. Therefore, the password policy cannot be enforced when such users are bulk-loaded into Oracle Internet Directory.

Task 4: Update lastchangenumber and Restart OID Processes

1. Start all OID processes

```
$ORACLE_HOME/opmn/bin/opmnctl startall
```

2. Update the `lastchangenumber` attribute of the profile.

   First, find the current last change number in Oracle Internet Directory with the `ldapsearch` command:

   ```
   $ORACLE_HOME/bin/ldapsearch -h <host> -p <port> -D <bindDN> \
   -w <bindDN pwd> -s base -b "" "objectclass=*" \
   lastchangenumber
   ```

   Next, use the `oidprovtool` command to update the `lastchangenumber` attribute to the number `n` that was discovered in the last step:

   ```
   oidprovtool operation=MODIFY \
   ldap_host=<ldap_host> \
   ldap_port=<ldap_port> \
   ldap_user_dn=<user to connect to LDAP> \
   ldap_user_password=<user password> \
   application_dn=<dn of the registered app for which the profile is
   modified> \
   orclLastAppliedChangeNumber=<n>
   ```

   For example:

   ```
   oidprovtool operation=MODIFY \
   ldap_host=testsys1.example.com \
   ldap_port=3060 \
   ldap_user_dn=cn=orcladmin \
   application_dn="
   orclApplicationCommonName=testsys1,cn=EBusiness,cn=Products,cn=Oracl
   eContext,dc=example,dc=com" \
   orclLastAppliedChangeNumber=100
   ```

3. Use `oidprovtool` with operation=ENABLE to enable the profile.

Task 5: Create Subscriptions for Bulkloaded Users

The `bulkload` tool does *not* automatically subscribe users to the parent Oracle E-Business Suite instance. To create the subscriptions for your bulkloaded users, run the following SQL statement on your Oracle E-Business Suite database:

```
select user_name from FND_USER where
FND_profile.VALUE_SPECIFIC('APPS_SSO_LOCAL_LOGIN', user_id)<>'LOCAL' and
FND_profile.VALUE_SPECIFIC('APPS_SSO_LDAP_SYNC', user_id)='Y'
```

You can save the results of this query in a text file using your SQL client's capabilities. See the section "Manual Subscription Management With Provsubtool" for details on how to run `provsubtool` to add these users to the subscription list.

Migrating Existing Accounts from Oracle Internet Directory to Oracle E-Business Suite Release 12

The `LDAPUserImport` command-line utility takes an LDIF file generated from Oracle Internet Directory, and inserts appropriate data into the Oracle E-Business Suite schema. It can be used for bulk migration of existing accounts from Oracle Internet Directory to Oracle E-Business Suite. `LDAPUserImport` updates both FND and TCA

schema.

> **Warning:** Importing user accounts and related information into Oracle E-Business Suite is a resource-intensive operation that may take a significant amount of time, as large amounts of business events and DML statements are issued in the process.

Task 1: Export Oracle Internet Directory users into LDIF file Using ldifwrite

The Oracle Internet Directory `ldifwrite` command-line utility is used to create an LDIF file that can be loaded into the Oracle E-Business Suite schema via the `LDAPUserImport` command-line utility.

Syntax and usage details for `ldifwrite` are described in the Oracle Internet Directory Administrator's Guide.

General syntax of the command is:

```
ldifwrite –c <db connect string> -b <base dn> -f <LDIF file>
```

For example:

```
ldifwrite -c asdb -b "cn=Users,dc=us,dc=example,dc=com" -f output.ldif
```

> **Note:** Do not modify the output file `output.ldif` in any way before proceeding with Task 2 below.

Task 2: Import LDAP Users into Oracle E-Business Suite using LDAP UserImport

The `LDAPUserImport` tool is run from the command line via the following steps:

> **Note:** The list of attributes migrated to the Oracle E-Business Suite from Oracle Internet Directory is limited to those described later in " Supported Attributes".

1. Ensure the environment is set up properly.

2. Invoke the `LDAPUserImport` tool with the following syntax: Note that all parameters can be entered on the same command line; for clarity, they are shown on different lines here (using the UNIX '\' continuation character).

```
java oracle.apps.fnd.oid.LDAPUserImport \
[-v] \
–dbc <dbcfile> \
-f <ldiffile> \
-n <nicknameattribute> \
-b <size in integer> \
-d \
[-l <logfile>] \
[-tcaRecord <N or Y>] \
[-defresp <N or Y>]
```

where:

[-v] - Run in verbose mode.

<dbcfile> - Full path to the dbc file.

<ldiffile> - LDIF file.

<nicknameattribute> - Name of the attribute used as the nicknameattribute in OID.

<logfile> - Log file name. If not specified, the default is LDAPUserImport.log.

-defresp - Indicates whether to assign the default responsibility 'Preferences SSWA' to the uploaded users or not. Default is 'Y' (assign the responsibility).

For example:

```
java oracle.apps.fnd.oid.LDAPUserImport \
-v \
-dbc $FND_SECURE/myebiz.dbc \
-f users.ldif \
-n uid \
-l users.log
```

If the OID user already exists in the Oracle E-Business Suite instance, the following actions are taken:

1. The duplicate record is ignored.
2. The log file is updated with a reference to the duplicate record.
3. Processing continues to the next OID record.

Enabling and Disabling Users

Enabling and disabling events for users are raised and consumed differently in Oracle Internet Directory and E-Business Suite.

Oracle E-Business Suite to Oracle Internet Directory

New user accounts whose start date are in the future or end date in the past are currently not provisioned from Oracle E-Business Suite to Oracle Internet Directory. Such pending user accounts have a corresponding place holder record created in the Oracle Internet Directory: this record is either deleted or activated once the account request has been processed.

> **Important:** The IDENTITY_MODIFY event must be enabled in Oracle Internet Directory to allow users to be enabled at the time of approval.

If an existing Oracle E-Business Suite user account is end-dated, the corresponding Oracle Internet Directory account is not affected. This is because the Oracle Internet Directory user may still require access to other partner applications. If no such access is needed, the relevant account will need to be disabled within Oracle Internet Directory.

Oracle Internet Directory to Oracle E-Business Suite

The status of an account in Oracle Internet Directory is propagated to Oracle E-Business Suite as being either *enabled* or *disabled*. The application account start and end date are not updated, and users with local access to the applications should not be affected.

The default functionality can be customized by creating a Workflow subscription for the event oracle.apps.fnd.identity.modify. See section "Creating Custom Workflow Subscriptions" for details.

User accounts deleted from the Oracle Internet Directory are end-dated in Oracle E-Business Suite, in order to maintain an audit trail.

Synchronizing Oracle HRMS with Oracle Internet Directory

The Oracle HR Agent can be utilized to manage Oracle Human Resources employees in Oracle Internet Directory, or to create E-Business Suite accounts automatically for new employees.

Definitions and Distinctions

An Oracle E-Business Suite *user* is someone who needs to be able to log into Oracle E-Business Suite. That user might need to file expense reports, view payslips, or file purchase requisitions. All Oracle E-Business Suite users have userids and records in the FND_USER repository, and have associated responsibilities that govern the functions and data that they can access.

An *employee* is someone whose information is managed by the Human Resources module in Oracle E-Business Suite. Oracle Human Resources tracks information such as employee numbers, manager hierarchies, and other personally identifiable information like birthdates.

Not all employees are users, and vice versa. For example, a retailer might use Oracle E-Business Suite's Human Resources modules to manage employee information for their cashiers, but those cashiers may not be authorized to log into Oracle E-Business Suite at all.

From an organizational standpoint, this distinction enables the HR department to manage employees and the IT department to manage Oracle E-Business Suite accounts. Following on from the example above, consider a scenario where the cashiers are permitted to view their payslips via the Self-Service Human Resources module. In such a case, the same person would be represented both in the Human Resources module and in the FND_USER repository. For Oracle E-Business Suite environments that are not integrated with Oracle Internet Directory, user records need to be individually maintained in each location.

Creating Employee Entries in Oracle Internet Directory

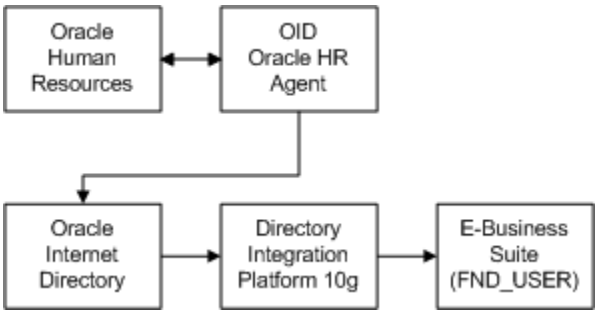
It is possible to use the Oracle Internet Directory Human Resources connector to push employee information from Oracle HR to Oracle Internet Directory:

> **Note:** Refer to the Oracle Identity Management Integration Guide for more information.

A subset of employee data can be exported from Oracle Human Resources into Oracle Internet Directory. The connector includes both a prepackaged integration profile, and an Oracle Human Resources agent that handles communication with Oracle Internet Directory.

The Oracle Human Resources connector can be scheduled to run at any time, configuring it to extract incremental changes from the Oracle Human Resources system.

Administrators can set and modify mapping between column names in Oracle Human Resources and attributes in Oracle Internet Directory. Since it is possible to provision users from Oracle Internet Directory to Oracle E-Business Suite, the following flow can be configured:



This architecture would support a business flow where a new employee is registered in E-Business Suite Human Resources by the HR department. That employee's information is then propagated via Oracle Internet Directory to FND_USER, where an IT administrator grants the appropriate Oracle E-Business Suite responsibilities to the user account.

> **Important:** The opposite direction is not supported. It is not possible to have an employee created in Oracle HR based upon a new user entry in Oracle Internet Directory.

Supported Attributes

The following two tables list, respectively, the attributes that may be provisioned from Oracle Internet Directory to Oracle E-Business Suite, and from Oracle E-Business Suite to Oracle Internet Directory.

> **Note:** This is a subset of the attributes listed in the provisioning templates.

Attributes Provisioned from Oracle Internet Directory to Oracle E-Business Suite

Oracle Internet Directory Attribute name	FND_USER Column Name	TCA Table and Column Names
UID and [nickname]*	USER_NAME	
DESCRIPTION	DESCRIPTION	
FACSIMILETELEPHONENUMBER	FAX	
MAIL	EMAIL_ADDRESS	HZ_CONTACT_POINTS.EMAIL_ADDRESS (CONTACT_POINT_TYPE is 'EMAIL')
SN		HZ_PARTIES.PERSON_LAST_NAME
TELEPHONENUMBER		HZ_CONTACT_POINTS.RAW_PHONE_NUMBER (CONTACT_POINT_TYPE is 'PHONE' and CONTACT_POINT_PURPOSE is 'BUSINESS'
STREET		HZ_LOCATIONS. ADDRESS1
POSTALCODE		HZ_LOCATIONS.POSTAL_CODE
PHYSICALDELIVERYOFFICENAME		HZ_PARTY_SITES.MAILSTOP

Oracle Internet Directory Attribute name	FND_USER Column Name	TCA Table and Column Names
ST		HZ_LOCATIONS.STATE
L		HZ_LOCATIONS.CITY
GIVENNAME		HZ_PARTIES.PERSON_FIRST_NAME
HOMEPHONE		HZ_CONTACT_POINTS.PHONE_NUMBER (CONTACT_POINT_TYPE is 'PHONE' and CONTACT_POINT_PURPOSE is 'PERSONAL')
C		HZ_LOCATIONS.COUNTRY

* Refer to "Recommended Nickname (Login Attribute) Setting" for more information

Attributes Provisioned from Oracle E-Business Suite to Oracle Internet Directory

FND_USER	Oracle Internet Directory
USER_NAME	UID and [nickname]*
DESCRIPTION	DESCRIPTION
EMAIL_ADDRESS	MAIL
FAX	FACSIMILETELEPHONENUMBER
END_DATE	ORCLACTIVEENDDATE
START_DATE	ORCLACTIVESTARTDATE
START_DATE/END_DATE	ORCLISENABLED
ENCRYPTED_USER_PASSWORD	USERPASSWORD

* Refer to "Recommended Nickname (Login Attribute) Setting" for more information. Also refer to "Configuring Directory Integration Platform Provisioning Templates" for

details of the provisioning process.

References and Resources

This section lists some important resources for additional information that will be needed when planning and undertaking integration of Oracle E-Business Suite into a single sign-on environment. These should be used in conjunction with the references given in the chapter.

References

See the Oracle Fusion Middleware Documentation Library for a description of:

- Oracle Access Manager architecture and configuration
- Oracle WebLogic Server architecture and configuration
- The various single sign-on choices available for use with Oracle Fusion Middleware

Also see My Oracle Support Knowledge Document 1388152.1, *Overview of Single Sign-On Integration Options for Oracle E-Business Suite*, to find the recommended integration for your version of Oracle E-Business Suite and a reference to the detailed setup instructions and steps needed to perform this integration.

Glossary of Terms

CN

Common Name. May include a user name.

DN

Distinguished Name The DN uniquely identifies a user in the directory. It comprises all of the individual names of the parent entries, back to the root.

DIP

Directory Integration Platform, the infrastructure that keeps user information bidirectional synchronized between Oracle Internet Directory, Oracle E-Business Suite Release 12, and third-party LDAP servers.

DIT

Directory information tree. A hierarchical tree-like structure consisting of the DNs of the entries.

GUID

Global Unique Identifier, a token used to identify a user's accounts in multiple systems during the single sign-on and enterprise level user management processes.

Identity Management Realm

A collection of identities, all of which are governed by the same administrative policies. In an enterprise, all employees having access to the intranet may belong to one realm, while all external users who access the public applications of the enterprise may belong to another realm. An identity management realm is represented in the directory by a specific entry with a special object class associated with it.

LDAP

The Lightweight Directory Access Protocol is a Internet-standard protocol and schema for user directories, and has gained widespread acceptance. LDAP was conceived as a standard, extensible directory access protocol for communication between suitably configured clients and servers. As a lightweight implementation of the International Standardization Organization (ISO) X.500 standard for directory services, LDAP requires a minimal amount of networking software on the client side, which makes it particularly attractive for Internet-based, thin client applications. Currently Oracle E-Business Suite Release 12 is certified to synchronize directly with Oracle Internet Directory only. However, Oracle Internet Directory can itself synchronize with one or more external, third-party user directories.

Oracle Access Manager

An Oracle Fusion Middleware component that can be integrated with Oracle E-Business Suite to provide a single sign-on solution.

Oracle E-Business Suite AccessGate

A Java Enterprise Edition application that can be used as part of a single sign-on solution for Oracle E-Business Suite. AccessGate is responsible for mapping a single sign-on user to an Oracle E-Business Suite user, and creating the Oracle E-Business Suite session for that user.

Oracle Internet Directory

Oracle Internet Directory is a general-purpose directory service runs as an application on the Oracle database and enables retrieval of information about dispersed users and network resources. It combines LDAP Version 3 with the high performance, scalability, robustness, and availability of the Oracle database. It communicates with the database (which may be on the same or on a different operating system) via Oracle Net, Oracle's operating system-independent database connectivity solution. As noted above, Oracle E-Business Suite is certified to synchronize directly with Oracle Internet Directory only, but Oracle Internet Directory can itself synchronize with one or more external, third-party user directories. For more information, see Oracle Internet Directory Release 11g Administrator's Guide.

Nickname Attribute

The attribute used to uniquely identify a user in the entire directory. The default value for this is uid. Oracle E-Business Suite uses this to resolve a simple user name to the complete distinguished name. The user nickname attribute cannot be multi-valued--that is, a given user cannot have multiple nicknames stored under the same attribute name.

Partner Application

An application that works within the Oracle single sign-on framework. It is designed (or has been modified) to delegate responsibility for user authentication to Oracle Access Manager. Oracle E-Business Suite Release 12.2 can be deployed as a partner application.

Provisioning

Refers to the process by which user information is synchronized between Oracle Internet Directory and Oracle E-Business Suite. How provisioning is set up depends both on site requirements and the configuration in use.

Provisioning Profile

Metadata that controls details of the provisioning process between Oracle Internet Directory and an Oracle E-Business Suite instance. A provisioning profile is required for each application that sends or receives provisioning events to or from Oracle Internet Directory.

Single Sign-On

Technology that allows a user to sign on once and gain access to multiple applications, instead of having to sign on to each application separately. In the context of Oracle E-Business Suite Release 12.2, refers to use of Oracle Access Manager to perform authentication, rather than the native FND_USER table.

Users

Individuals who have access to one or more software applications at a particular enterprise. Users are "global" entities, i.e. their existence and attributes exist outside the context of any particular software application.

User Directory

Software services that store the list of users and their attributes. Oracle E-Business Suite currently has its own proprietary user directory (the FND_USER table). There are also general purpose user directories that manage user information and expose it to integrated applications through a standard interface.

The Lightweight Directory Access Protocol (LDAP, see above for definition) is an example of a user directory.

Index

A

Access Control with Oracle User Management, 2-1
Account Creation by Administrators
  Access Control with Oracle User Management, 2-8
AFPASSWD utility
  using to change APPS database account password, 6-67
  using to reset user's password when switching back to local authentication, 6-46
  using to set local password, 6-24
  using when profile is updated to allow LOCAL access, 6-56
Allowed JSPs
  Definition, 4-77
Allow Unrestricted JSP Access
  profile option for Allowed JSPs feature, 4-78
Application users
  assigning one or more responsibilities, 4-1
  changing passwords, 4-23
  defining, 4-23
  disabling application password, 4-23
  reporting on active users, 4-60
  start dates, 4-26
  username characteristics, 4-23
AppsUserExport
  user migration utility, 6-71
Audit Groups Window, 5-25
Auditing database row changes
  AuditTrail, 5-1
Auditing user activity
  Sign-On Audit, 5-1
Audit Installations Window, 5-23
Audit reports
  brief explanation, 5-5
  listing, 5-1
Audit Tables Window, 5-27
AuditTrail
  archiving data, 5-15
  audit groups, 5-6
  audit set, 5-6
  changing audit tables, 5-6
  description, 5-6
  introduction, 5-1
  reporting, 5-14
  setting up, 5-7
  tables, 5-8
  views, 5-9
authorization
  in single sign-on, 6-10

C

Case sensitivity
  in user passwords, 4-2
collisions
  In OID bulkload, 6-75
configurable user name policy, 3-26
Connectors
  Oracle Identity Manager component, 6-3
cookie
  definition, 4-74
criterion

in role administration, 3-33

D

Database Connection Tagging, 5-1, 5-40
Data Security, 4-15
  Access Control with Oracle User Management, 2-3
Data Security Policies
  Defining Data Security Policies, 3-14
Delegated Administration
  Access Control in Oracle E-Business Suite, 2-6
  Defining Delegated Administration Privileges for Roles
    Organization Administration, 3-9
    Role Administration, 3-9
    User Administration, 3-9

E

end-dating
  roles, 3-32

F

FGA for RBAC
  See Fine Grained Access for RBAC
Fine Grained Access for RBAC, 3-33
FND_CONNECTION_TAGGING
  profile option for Database Connection Tagging, 5-41
FNDCPASS utility, 4-3
  using to change APPS database account password, 6-67
  using to migrate passwords to non-reversible hash, 4-3
  using to reset user's password when switching back to local authentication, 6-46
  using to set local password, 6-24
  using when profile is updated to allow LOCAL access, 6-56
Form Functions Window, 4-28
Forms
  Define Menu, 4-28, 4-33
  Monitor Application Users, 5-21
  Responsibility, 4-19
Function Security
  Access Control with Oracle User Management, 2-2
  implementation, 4-13
Function Security Function Report, 4-58
Function Security Menu Report, 4-58
Function Security Menu Viewer
  Menu Viewer, 4-36
Function Security Navigator Report, 4-58

G

GUEST User
  password restriction, 4-2
Guest user account, 4-3

H

HRMS Security, 4-2

I

ICX_SESSION_COOKIE_DOMAIN
  Oracle Applications Session Cookie Domain Profile, 4-75

L

LDAPUserImport
  user migration utility, 6-71
limited administrator
  in FGA for RBAC, 3-33

M

Menus
  compiling, 4-14, 4-57
  defining, 4-33
  defining a menu entry, 4-33
  entering arguments, 4-30
  menu prompts, 4-33
  Menu Viewer, 4-36
  role in function security, 4-1
  sequence numbers, 4-33
Menus Window, 4-33
Menu Viewer, 4-36
Monitoring users
  Sign-On Audit, 5-5
Monitor Users Window, 5-21

N

Non-reversible hash password scheme, 4-2

O

OAM WebGate agent
used in Oracle E-Business Suite integration, 6-4
Oracle Access Manager
OAM, 6-1
Oracle Access Manager WebGate
definition, 6-5
Oracle E-Business Suite AccessGate, 6-2
Oracle E-Business Suite security
defining a responsibility, 4-19
ORACLE ID
assigning to responsibility, 4-21
Oracle Identity Management
integration with Oracle E-Business Suite, 6-3
Oracle Identity Manager
use with Oracle E-Business Suite, 6-2
Oracle Internet Directory
OID, 6-1
Oracle User Management Setup Tasks
Defining Role Categories, 3-1
Organization Administration Privileges
Access Control in Oracle E-Business Suite , 2-6
Organization Contacts
Registering External Organization Contacts, 3-40

P

Page Access Tracking
in Proxy User mode, 3-39
Password
Resetting User Passwords, 3-30
Passwords
case-sensitive, 4-2
People
Maintaining People and Users, 3-28
Permissions
Assigning Permissions to Roles, 3-3
provisioning integrated application
definition, 6-14
provisioning profiles
definition, 6-14
proxy users, 3-38

R

Registration Processes
Access Control with Oracle User Management, 2-8
Creating and Updating Registration Processes, 3-23
Reports
Active Responsibilities, 4-59
Active Users, 4-60
Reports and Sets by Responsibility, 4-61
Signon Audit Concurrent Requests, 5-30
Signon Audit Forms, 5-32
Signon Audit Responsibilities, 5-34
Signon Audit Unsuccessful Logins, 5-36
Signon Audit Users, 5-38
Users of a Responsibility, 4-58
REpresentational State Transfer security services
definition, 4-62
Requests for Additional Access, 3-41
Access Control with Oracle User Management, 2-8
Responsibilities, 4-1
Application name, 4-20
deactivating, 4-20
defining, 4-19
major components, 4-4
predefined, 4-5
reporting on active responsibilities, 4-59
reporting on reports and report sets, 4-61
reporting on users of, 4-58
Start date, 4-20
Responsibilities Window, 4-19
REST services
See REpresentational State Transfer security services
Role Administration Privileges
Access Control in Oracle E-Business Suite, 2-6
Role Based Access Control (RBAC)
Access Control with Oracle User Management, 2-3
Role Categories
Access Control with Oracle User Management, 2-3
Defining Role Categories, 3-1
Role Inheritance Hierarchies

Access Control with Oracle User Management, 2-3
Defining Role Inheritance Hierarchies Deployment Options, 3-15
Roles
Assigning Permissions to Roles, 3-3
Assigning Roles to and Revoking Roles From Users, 3-32
Creating and Updating Roles, 3-2
Defining Delegated Administration Privileges for Roles
Organization Administration, 3-9
Role Administration, 3-9
User Administration, 3-9

S

Security Administrator
Proxy User Management, 2-8
Security groups, 4-9
Security Groups
defining (for HRMS only), 4-22
Security Groups Window, 4-22
Security in HRMS, 4-2
seeded user name policies, 3-27
Self Service Account Requests
Access Control with Oracle User Management, 2-8
Self-Service and Approvals
Access Control in Oracle E-Business Suite, 2-16
Self-Service Registration, 3-40
Session time-out, 4-4
Sign-On Audit
audit levels, 5-3
examples using, 5-4
introduction, 5-1
monitoring users, 5-5, 5-21
purging obsolete data, 5-40
reporting on users, 5-38
reporting on users and forms, 5-32
reporting on users and requests, 5-30
reporting on users and responsibilities, 5-34
reporting on users and unsuccessful logins, 5-36
reports, 5-5
setting up, 5-3
using, 5-2
single sign-out
definiton, 6-2

T

target systems, 6-3
txkEBSAuth.xml
Oracle E-Business Suite AccessGate configuration script, 6-7

U

UMX_SYS_ACCT
data security object, 2-7, 3-7, 3-37
Upgrading
preserving custom menus, 4-15
User Accounts
Creating, Inactivating, and Reactivating User Accounts, 3-30
User Administration Privileges
Access Control in Oracle E-Business Suite, 2-6
Users, 4-1
Assigning Roles to and Revoking Roles From Users, 3-32
case-sensitive passwords, 4-2
Maintaining People and Users, 3-28
Resetting User Passwords, 3-30
User session limits, 4-4
Users Window, 4-23