SafePaaS

Oracle E-Business Suite Role and Responsibility Access Management Solution by SafePaaS

INTRODUCTION

Oracle E-Business Suite security model can be configured to grant users access based on Responsibilities as well as Roles. In this document we provide an overview of the Oracle security model fundamentals, describe best practices for implementing access policies and share some key capabilities in SafePaaS to help you automate user access controls management process for users granted access to Oracle E-Business Suite applications through Roles as well as Responsibilities.

Oracle Responsibilities consist of navigation menus that include sub-menus, functions and programs called "concurrent requests". Once a user is granted access to one or more Responsibilities, she/he can access all the functions and programs within all the available menus. There are additional security attributes that can limit user access to data and functions. A key drawback of the Responsibility based security model is that it does not support data level security.

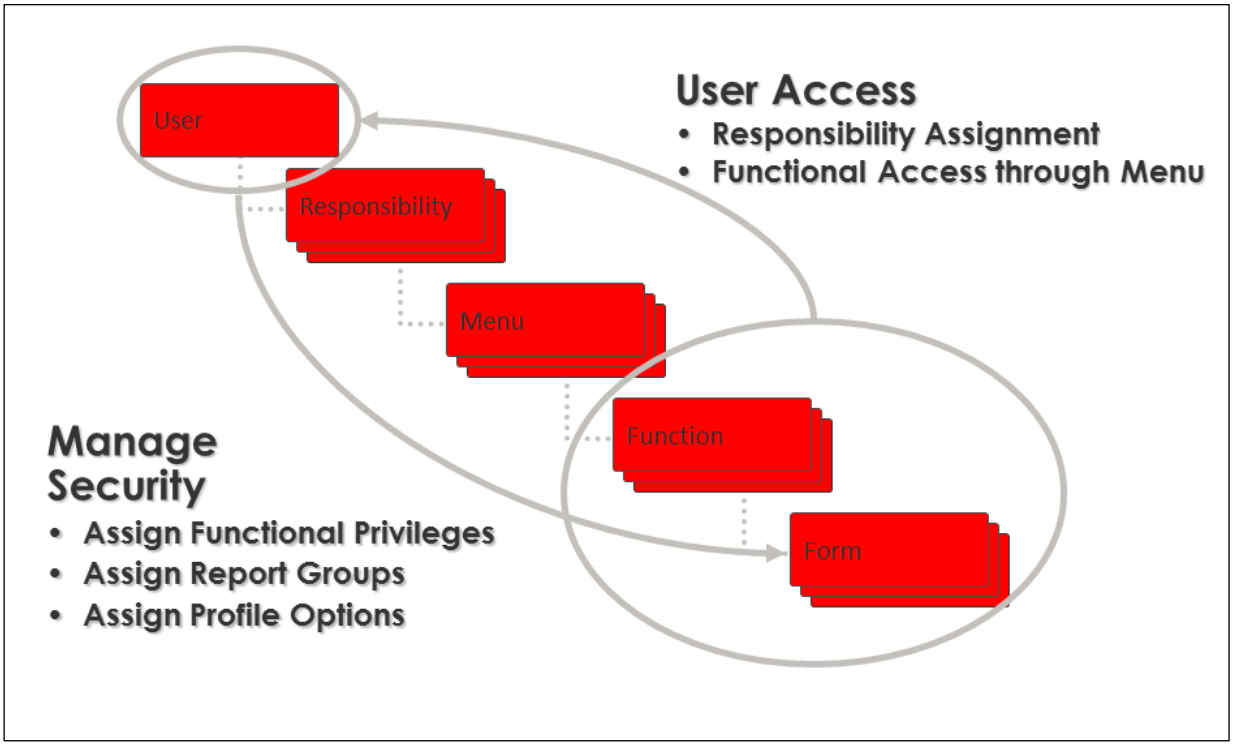
Role-based access control (RBAC) fully introduced in Oracle E-Business Suite R12 offers significant security design improvement over the Responsibilities based option by normalizing access to functions and data through user roles rather than only users. RBAC security model improves access security by controlling what a user can do on each function or sets of data under specific condition, e.g. view and edit are actions, and task flows or rows in data tables are resources.

FUNDAMENTALS OF ORACLE E-BUSINESS SUITE SECURITY MODEL

Oracle E-Business Suite security model is designed to provide application user access to perform business functions through the assignment of one or more "Responsibilities". Multiple users can share the same responsibility. A system administrator can assign users any of the standard ("seeded") responsibilities provided with Oracle Applications, or create new custom responsibilities as required.

A responsibility provides an "entitlement" in which a user operates. This entitlement includes navigation menus, profile option values and concurrent programs. For example, a responsibility can allow access to a restricted list of menu items that a user can navigate to Enter Journals function on an Oracle Form in the General Ledger Module. Reports in a specific application are signed to one or more responsibilities, so the responsibility a user chooses determines the reports that can be submitted.

The following diagram shows the components of the Oracle EBS security model:



We will describe each component of the EBS security model below.

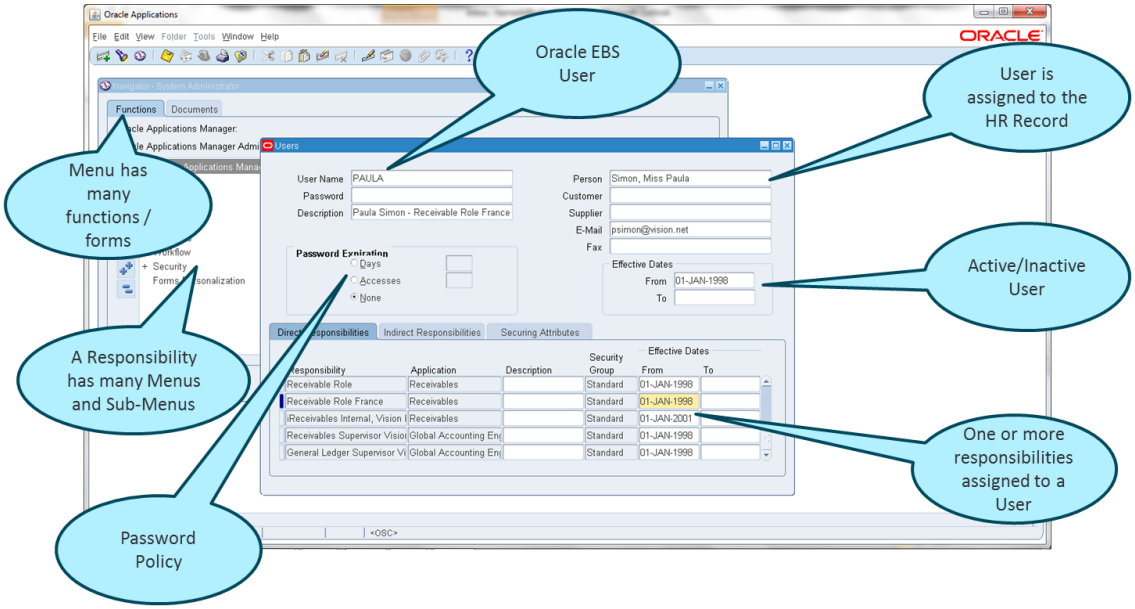
RESPONSIBILITY BASED ACCESS MANAGEMENT

A User is created in Oracle EBS and one or more responsibilities are assigned to the user. The system administrator assigns a temporary password to the user that is changed when the user first signs into Oracle EBS. An Employee HR record can be associated to the user on this form as well. Password expiration option enables you to apply your password security policies. You can choose "Days" to 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. You can also select "Accesses" to enter the maximum allowed number of sign-one's to Oracle Applications 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.

The user cannot sign onto Oracle Applications 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 Applications because this information helps to provide an audit trail. You can deactivate an Oracle Applications 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.

You can select the name of a responsibility you wish to assign to the application user. A responsibility is uniquely identified by application name and responsibility name. 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.

The following screen shot shows the User Assignment Form in Oracle EBS:



MENUS AND FUNCTIONS

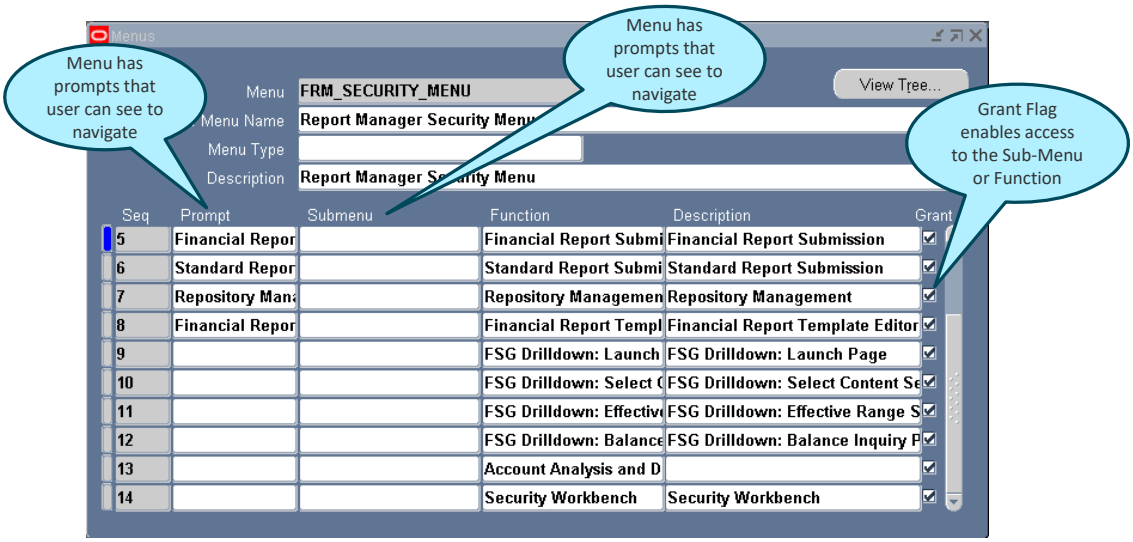
A menu is a hierarchical arrangement of application functions (forms). In the definition of a responsibility, the specified menu defines what is displayed in the navigator. The specified menu does not necessarily define the functions that can be accessed by the responsibility, which are granted. A menu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

Menus have prompts that are displayed for a user to navigate. Menu prompts that have unique first letters enable the users to type the first letter of the menu prompt to choose a menu entry.

A menu can also call another menu (sub-menu) and allow the user to select menu entries from that sub-menu.

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.

The following screen shot shows the Menu Setup in Oracle EBS:



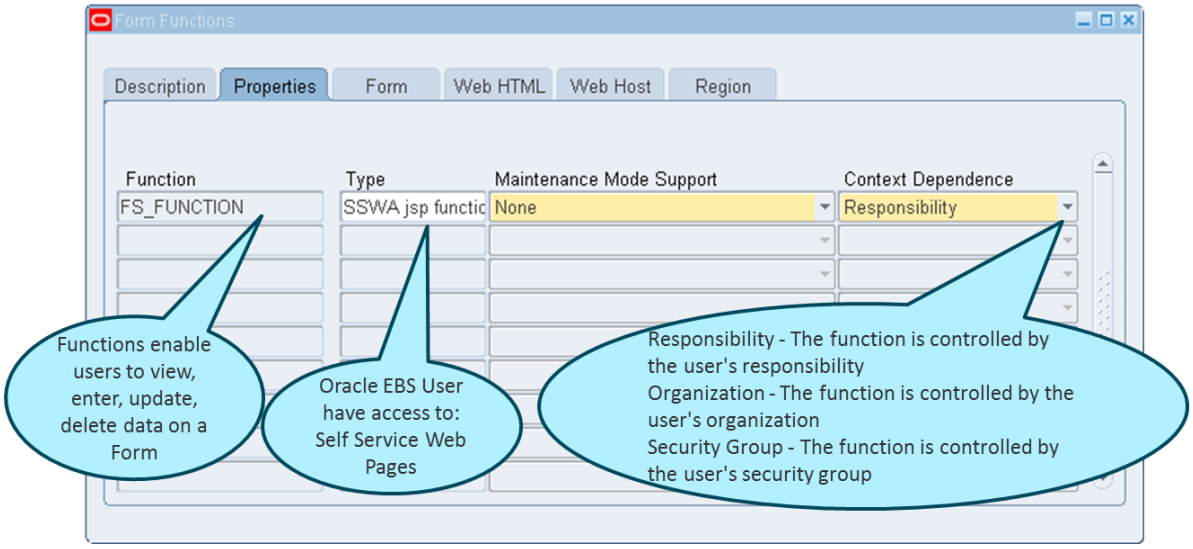
A function is a part of an application's functionality that is registered under a unique name for the purpose of assigning it to, or excluding it from, a responsibility. There are many types of functions. A function's type describes its use. The function types include: Form – for user to access an Oracle Form, Sub function - added to menus (without prompts) to provide security functionality for forms or other functions, JSP - Functions used for some products in the Oracle Self-Service Web Applications.

Function behavior is context dependent. 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 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
• 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.

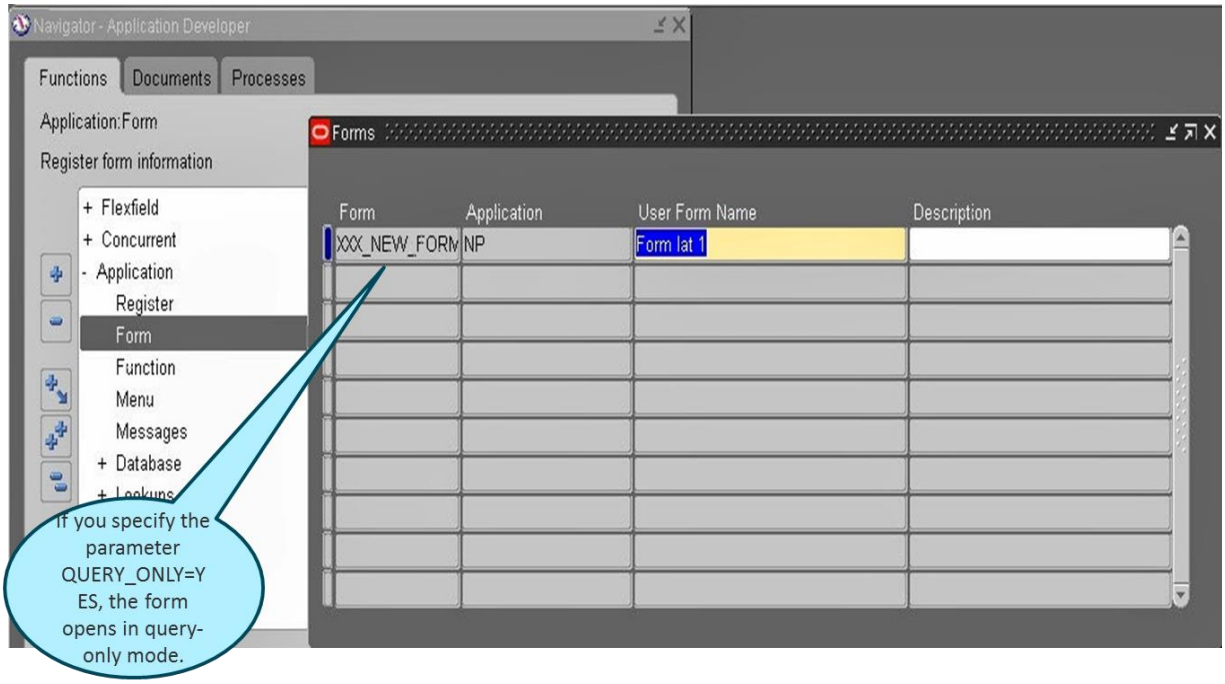
The following screen shot shows the Function Setup Form in Oracle EBS:



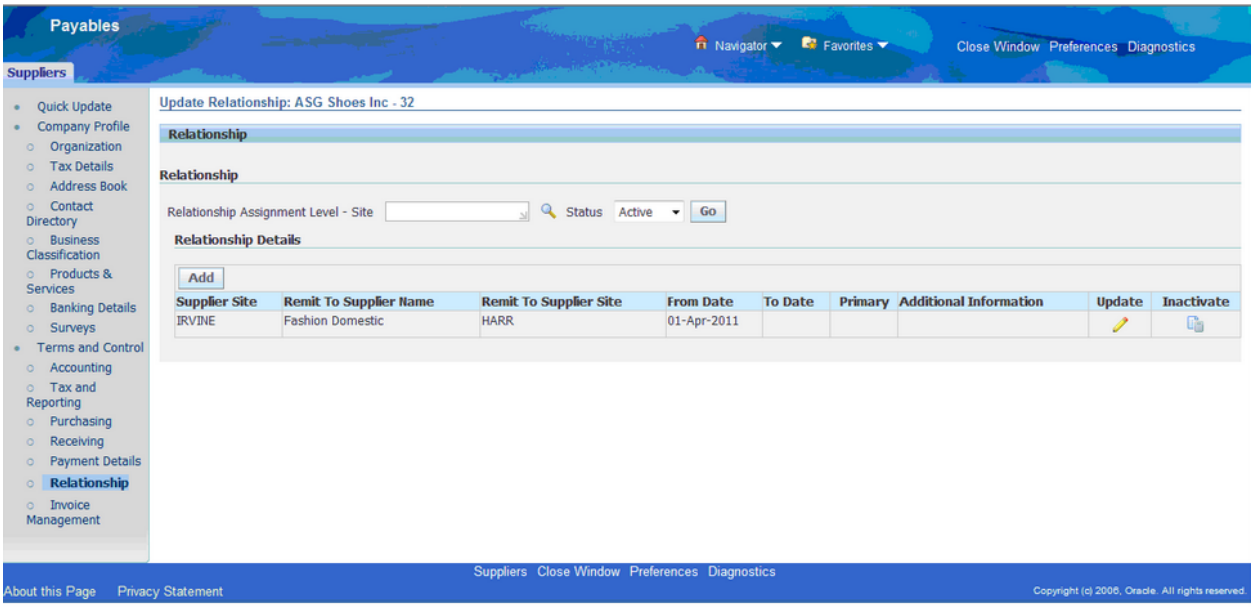
FORMS, HTML PAGES AND PERSONALIZATION

Forms and HTML Pages are invoked by functions in Oracle EBS when the user navigates to them using the Navigator window. Oracle E-Business suite provides granular security controls on Oracle Forms or Self-Service Web Pages accessed by users through the security components described above.

Forms appear in the Navigator window. A form 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 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 functions (subfunctions), 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. When you define a form function in the Form Functions window or call an existing form function using FND_FUNCTION.EXECUTE or APP_NAVIGATE.EXECUTE, you can add the string: QUERY_ONLY=YES as shown below:



A growing number of Oracle applications use the HTML pages to provide user access to application functions. For example, Oracle Self-Service Web Applications that use HTML pages include Self-Service Expenses, Self-Service Human Resources, and Internet Procurement. You can navigate to functions in an HTML-based application using tabs, subtabs, and side navigation elements. As you drill down through these elements, locator links display your navigation path. When you navigate through records in a table, you can use the Previous and Next links. When you navigate through steps of a task, you can use the Back and Next buttons. The following screen shot shows the HTML Page for supplier management in Oracle EBS R12:



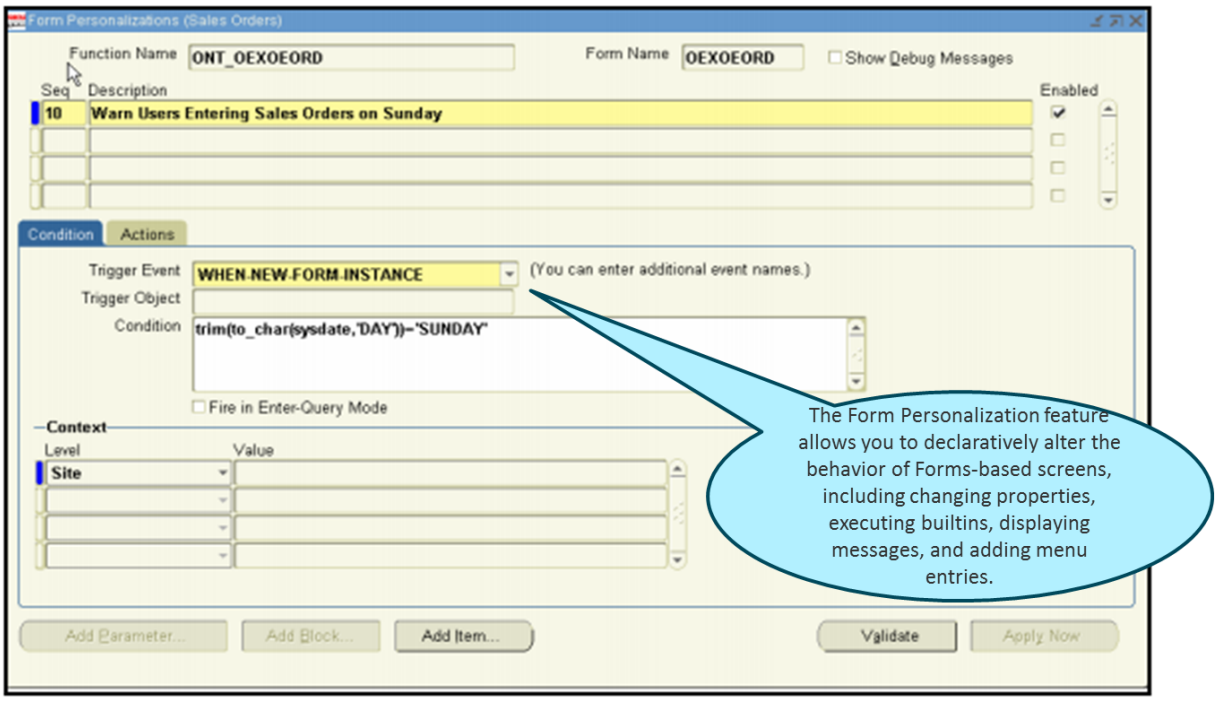
The Form Personalization feature allows you to declaratively alter the behavior of Forms-based screens, including changing properties, executing built-ins, displaying messages, and adding menu entries.

For each function (a form running in a particular context based on parameters passed to it), you can specify one or more Rules. Each Rule consists of an Event, an optional Condition, the Scope for which it applies, and one or more Actions to perform. An Event is a trigger point within a form, such as startup (WHEN-NEW-FORM-INSTANCE), or when focus moves to a new record (WHEN-NEW-RECORD-INSTANCE). There are standard events that almost every form sends, and certain forms send additional product-specific events. The Scope is evaluated based on the current runtime context to determine if a Rule should be processed or not. The Scope can be at the Site, Responsibility, User, or Industry level. Each Rule can have one or more Scopes associated with it. The Condition is an optional SQL code fragment that is evaluated when the Event occurs; if it evaluates to TRUE then the Actions are processed. Each Action consists of one of the following:

- Setting a Property, such as making a field Required or hiding a Tab page
- Executing a Builtin, such as GO_BLOCK, DO_KEY or EXECUTE_FUNCTION
- Displaying a Message
- Enabling a Special menu entry

Once Rules are defined, when the target function is run then the Rules are automatically applied as events occur within that form.

The following screen shot shows Personalization:



PROFILE OPTIONS

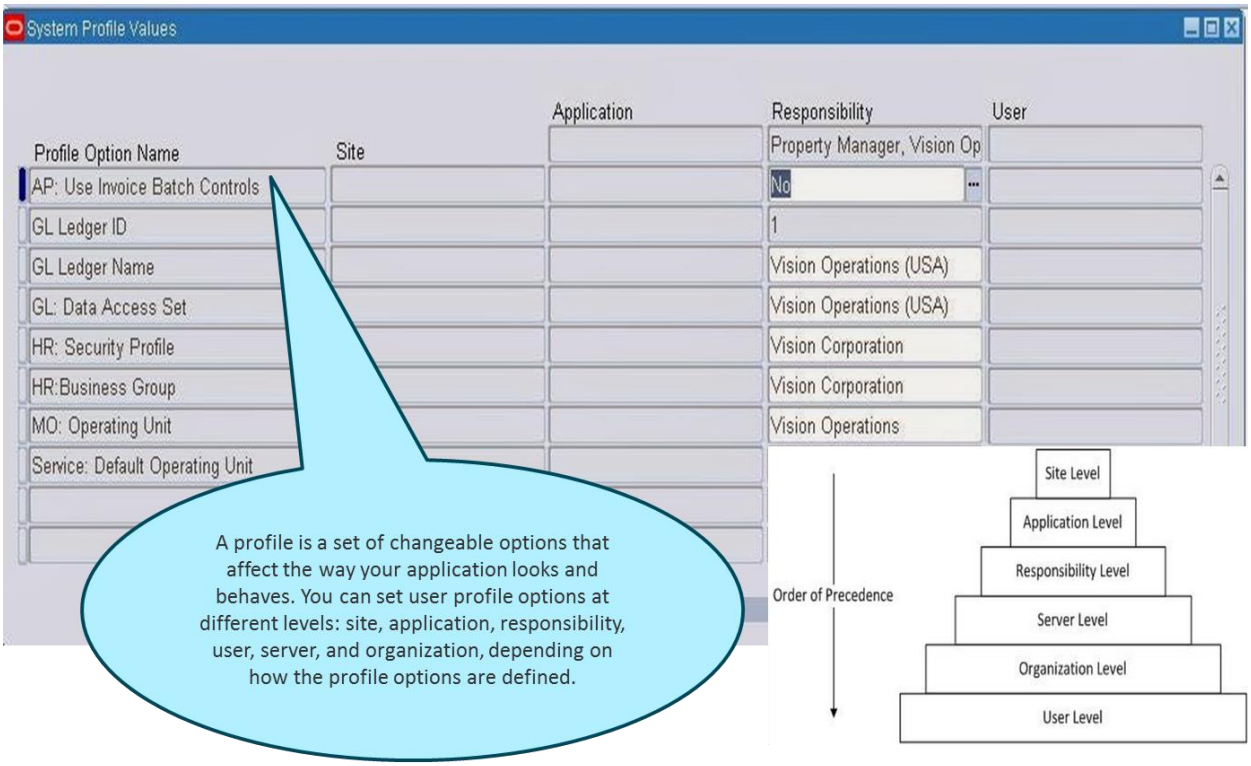
A user profile is a set of changeable options that affect the way your application looks and behaves. User security access can be controlled by setting user profile options to the values you want. You can set user profile options at four different levels: site, application, responsibility, and user. If you change a user profile option value, your change takes effect as soon as your users log on again or change responsibilities.

When you set a user profile, you provide Oracle Applications with standard information (such as printer) that describes a user, responsibility, application, or site. You can set values for user profile options at each profile level:

- Site Option settings pertain to all users at an installation site.
- Application Option settings pertain to all users of any responsibility associated with the application.
- Responsibility Option settings pertain to all users currently signed on under the responsibility.
- User Option settings pertain to an individual user, identified by their application username.

The values you set at each level provide run-time values for each user's profile options. An option's run-time value becomes the highest-level setting for that option.

When a profile option may be set at more than one level, site has the lowest priority, superseded by application, then responsibility, with user having the highest priority. For example, a value entered at the site level may be overridden by values entered at any other level. A value entered at the user level has the highest priority, and overrides values entered at any other level as shown below:



ROLE BASED ACCESS CONTROLS SECURITY MODEL

Role Based Access Control (RBAC) is the next level of security model available in Oracle E-Business Suite. It 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 policies 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.

SEGREGATION OF DUTIES MANAGEMENT WITH RBAC

Roles are managed through User Management (UMX) HTML pages. RBAC is less prone to errors and relatively simple to maintain. This keeps the system more secure and lowers the cost of maintaining security through security simplification. With the implementation of RBAC you can reduce the cost of administration of your E-Business suite Environment. You can improve security and compliance by developing effective SOD compliant model that reduces cost of managing security violations. You can empower business users with the delegated administration functionality the task managing access to their business areas and provisioning their user's roles based on their business needs

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.

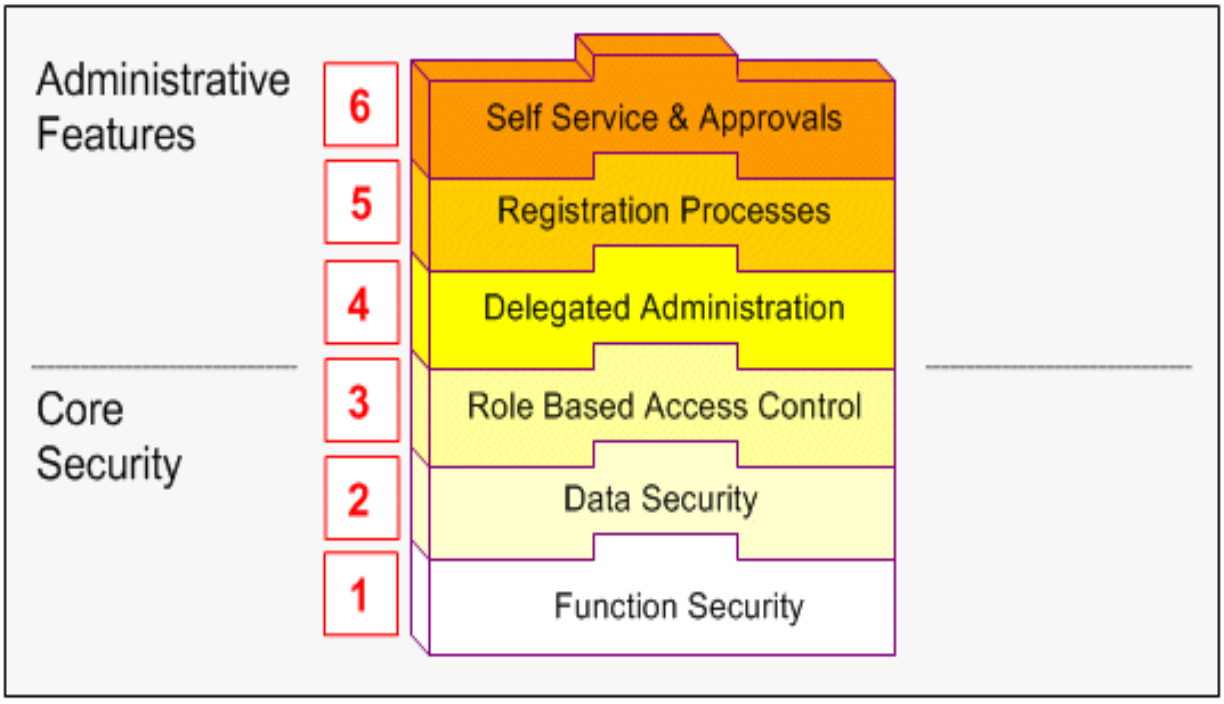
USER MANAGEMENT WITH RBAC

RBAC enables you to manage user security based on Oracle's Function and Data Security models. Administrative features build upon Core Security and include Delegated Administration, Registration Processes, and Self Service and Approvals.

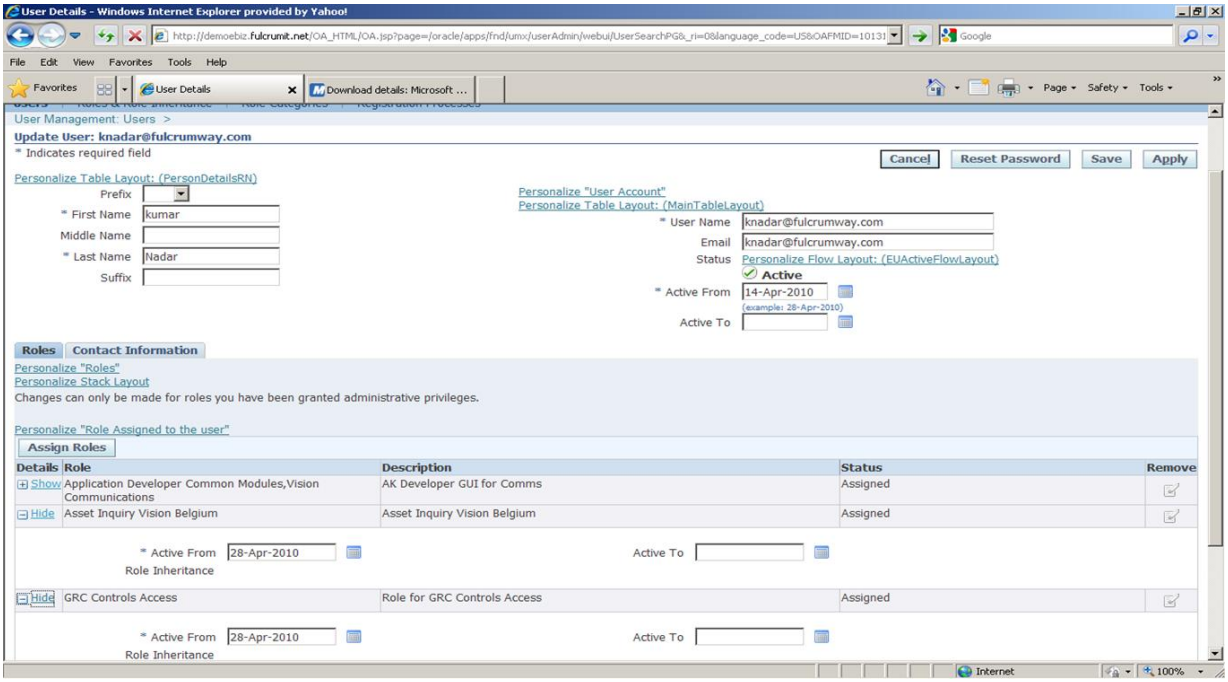
Core Security and Administrative Features are implemented in successive layers and each builds upon the one that precedes it. Organizations can optionally uptake the various layers, depending on the degree of automation and scalability that they wish to build upon the existing Function and Data Security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

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:



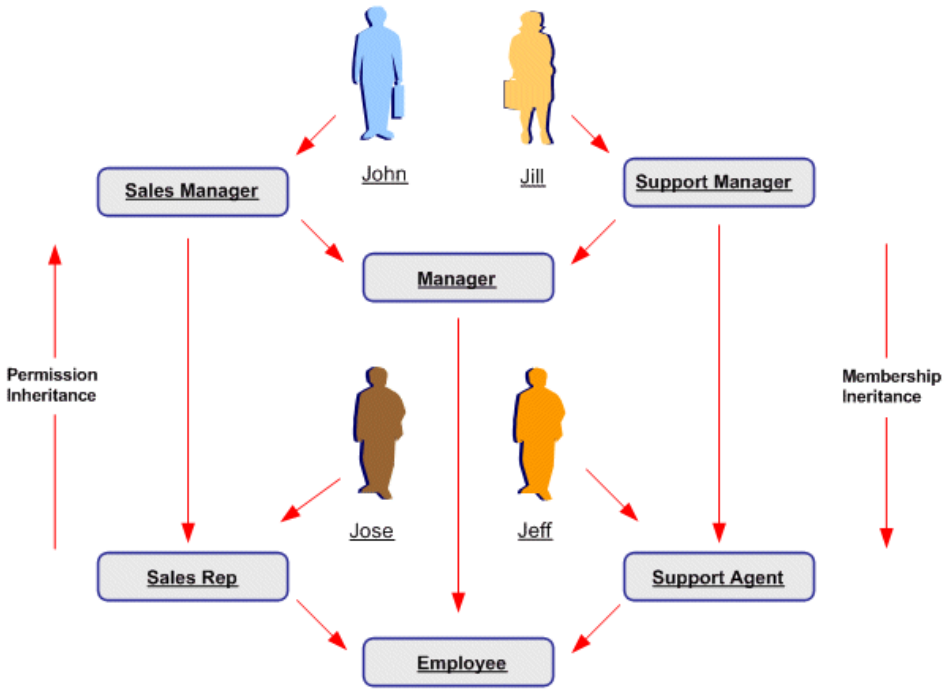
The following screen shot shows the User Management page in Oracle EBS:



ROLE INHERITANCE

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:

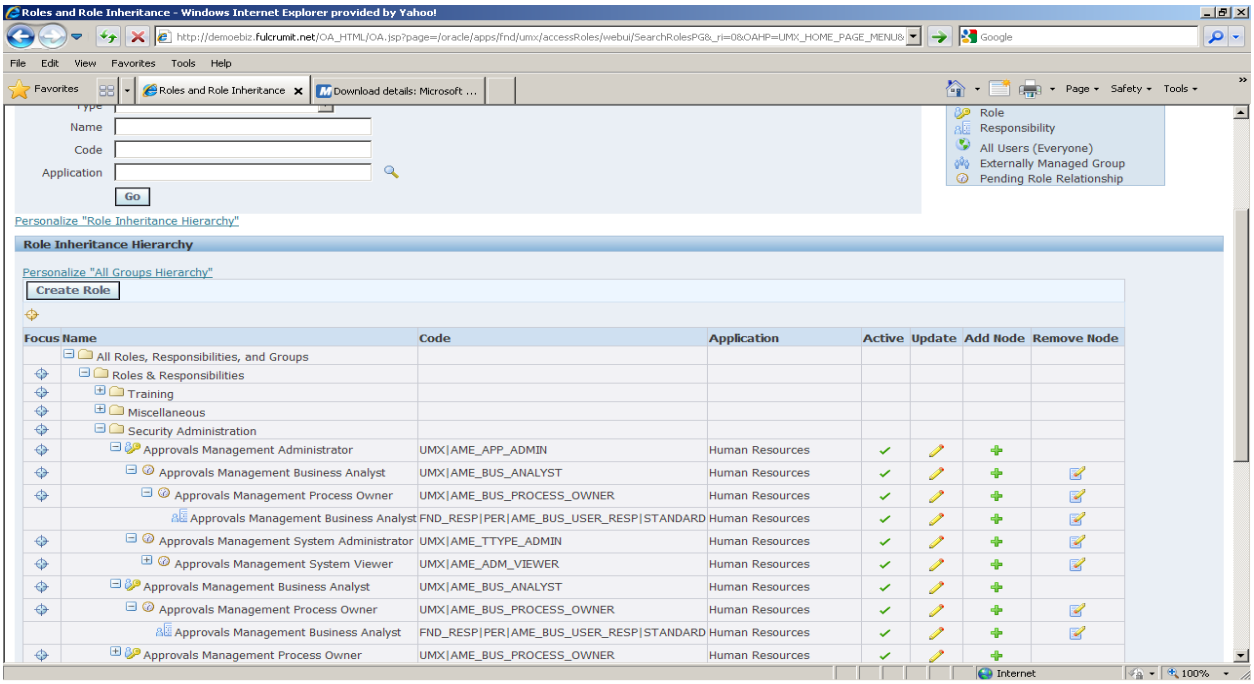
Role Inheritance Hierarchy



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.

The follow screen shows the Roles inheritance:



DELEGATED USER MANAGEMENT

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.

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.

SAFEPAAS ACCESS CONTROLS MANAGEMENT SOLUTION

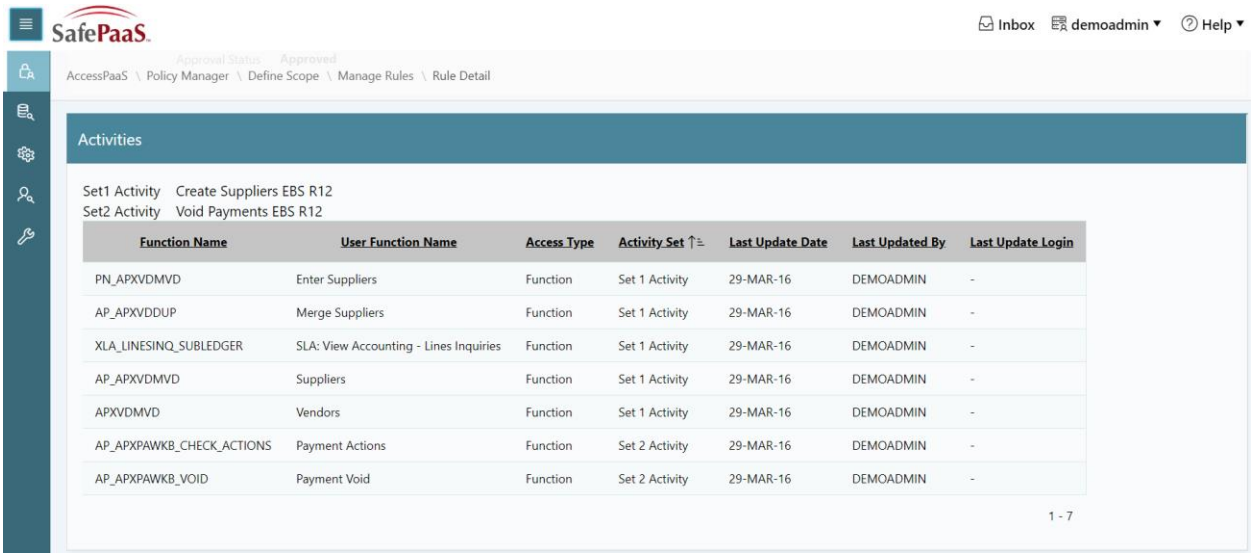
SafePaaS is a trusted Access Platform-as-a-Service (AccessPaaS) available in the Cloud for the Modern Digital Enterprise Access Management for Any App, Any Device and Any Data Source. You can configure the security model for any application data source to enable central access management. The security snapshot is extracted and analyzed for access policy compliance. The solution includes pre-configured security model for popular enterprise applications including Oracle E-Business Suite Roles and Responsibilities, PeopleSoft, J D Edwards, SAP, Microsoft and Workday.

The solution includes a comprehensive Rules Repository, Reviewed by all the major audit firms. It includes over 250 access rules for Oracle E-Business Suite, covering over 2,500 access points. Additionally, you also have access to 1,000+ objects to monitor EBS configuration and transactions based on user defined rules that ensure effective process controls throughout the enterprise. You can deploy SafePaaS within days to detect access risks in Oracle EBS applications and rapid remediate risks with built-in Security Model Simulation. Rules driven role design and user assignment tools simulate corrective actions to reduce policy violations. Approved actions can be automatically executed within Oracle EBS to streamline the remediation process.

ACCESS POLICY DEFINITION

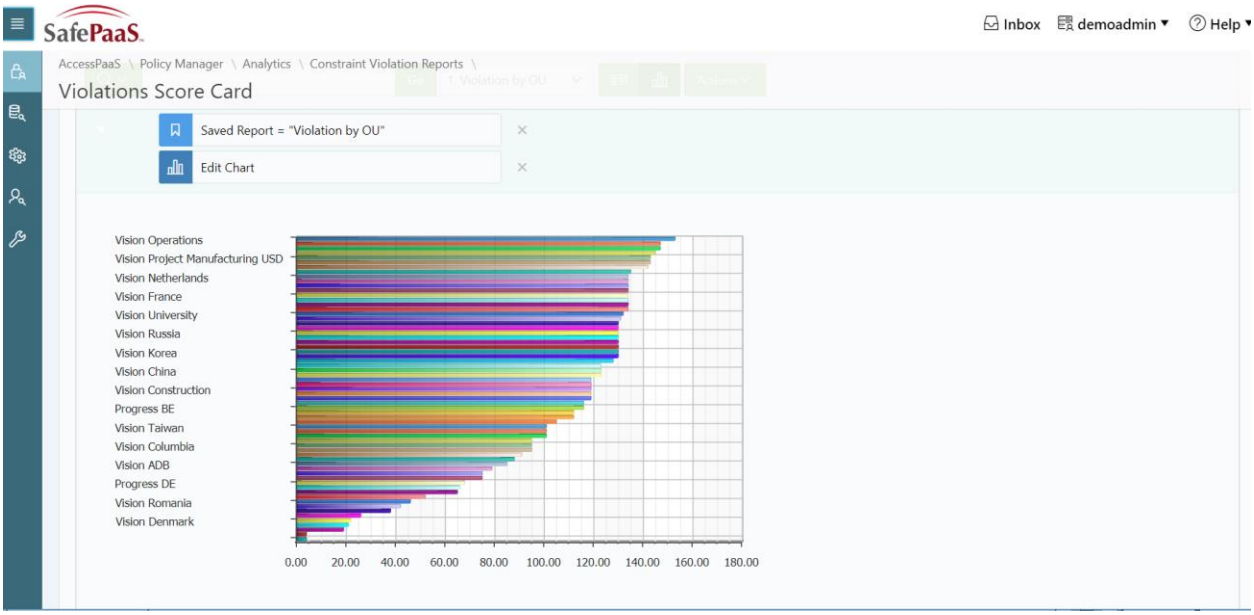
You can define access policies based on the "entry points" within Oracle E-Business Suite which are one or more Functions that enable a user to perform a business activity such as Create Supplier, Approve Purchase Order, Pay AP Invoice, etc.

The following screen shows the Segregation of Duties activity details for the rule "Create Supplier and Void Payments":



SEGREGATION OF DUTY VIOLATIONS (SOD) MANAGEMENT

Once the SOD Rules are defined, SafePaaS enables you to take a "snapshot" of the Oracle EBS security model and test the entire security model including users, roles and responsibilities based on the menu function and other security attributes to detect any violation of the SOD Rules. You can manage SOD violations by removing "global" false positives such as end-dated users and responsibilities, read-only functions and menus without grant flags. In addition, we can close violation if certain application controls such as page personalization, profile options, and configuration or transaction controls mitigate the access risks. The following screen shot show the SOD violations by business operating unit

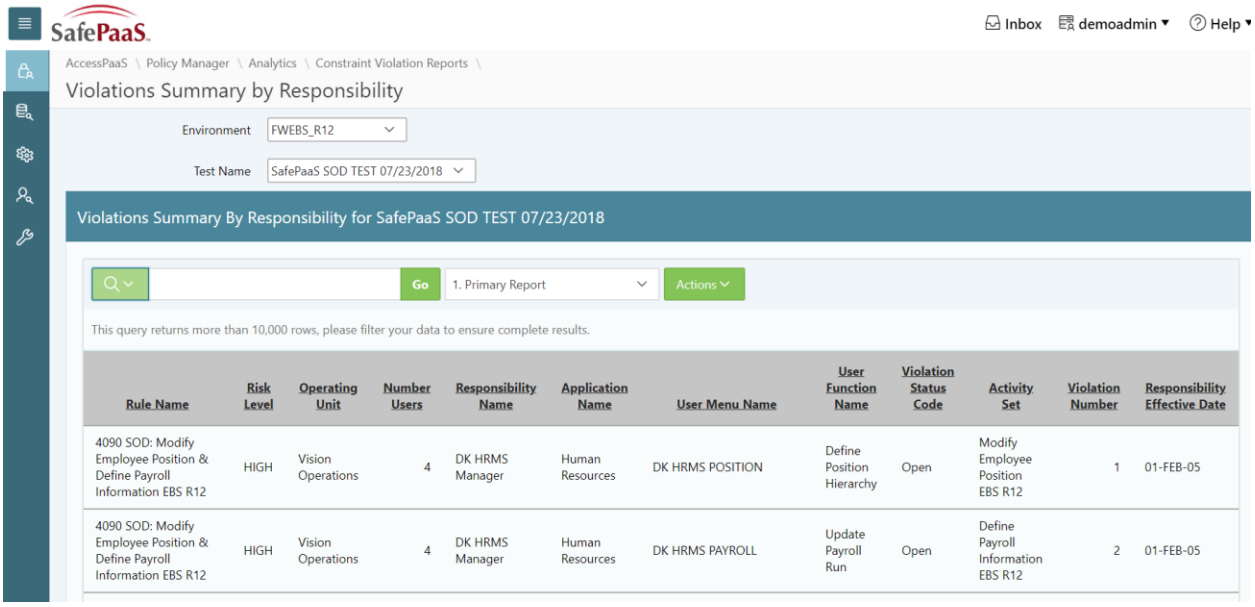


SOD Violation within Oracle Responsibility Security Model

The root cause of many SOD violation in Oracle E-Business Suite is inherent risk in "seeded" Oracle menus that customer use to build Responsibilities for their users. Top level Oracle menus contain access to master data, configurations as well as transactions to execute the key activities within a business process. There are two methods, Custom and Exclusion method available to limit functional access in the menus. However, without SOD simulation capabilities in Oracle EBS, and thousands of functional combinations in multi-level menus, it's very difficult to determine the inherent SOD risks.

It's a common practice to assign users more than one Responsibilities, since the Oracle Responsibilities are based on Oracle Modules like General Ledger, Payables, Receivables, Inventory, etc... Business users need access to multiple modules to complete their tasks. For example, a typical member of the finance team may have five or more responsibilities to perform job functions. The combination of Responsibilities further increases the likelihood and impact the SOD risks and makes is difficult to remediate the risk since the source of violations can not be analyzed without advanced analytics.

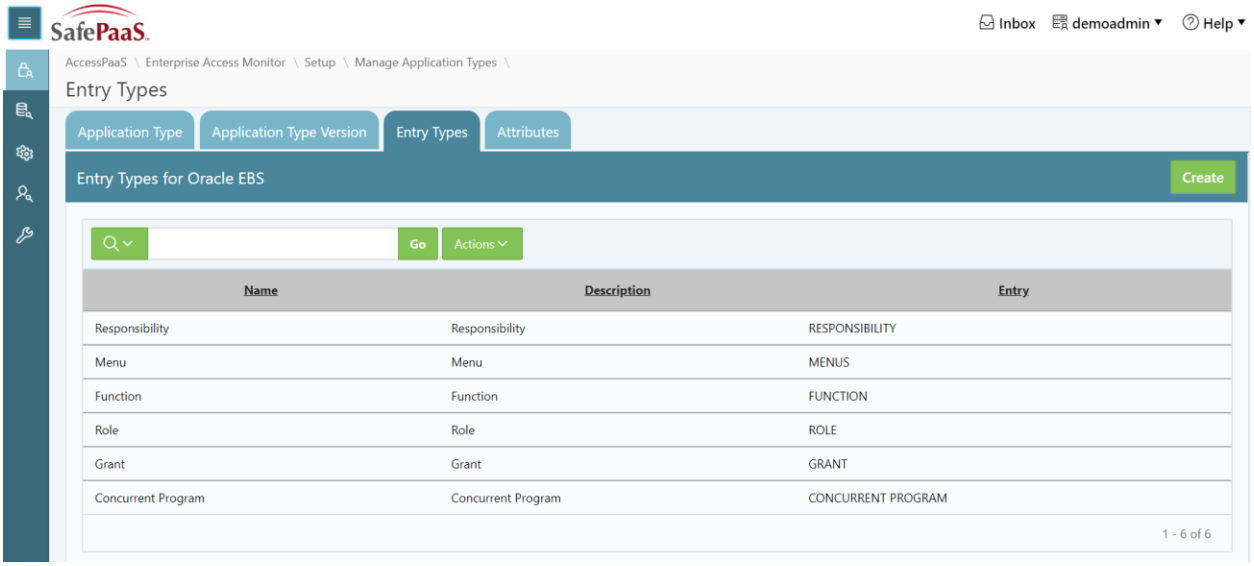
SafePaaS provides Responsibility Violation Management reports that can pin point the root cause of the SOD conflict along with menu and function level access path details to ensure timely and accurate correction actions. See screen shot of Violation Report below that show the menu and functions within the Responsibility DK HRMS Manager with inherent SOD risk to Modify Employee Position and Define Payroll Information in EBS:



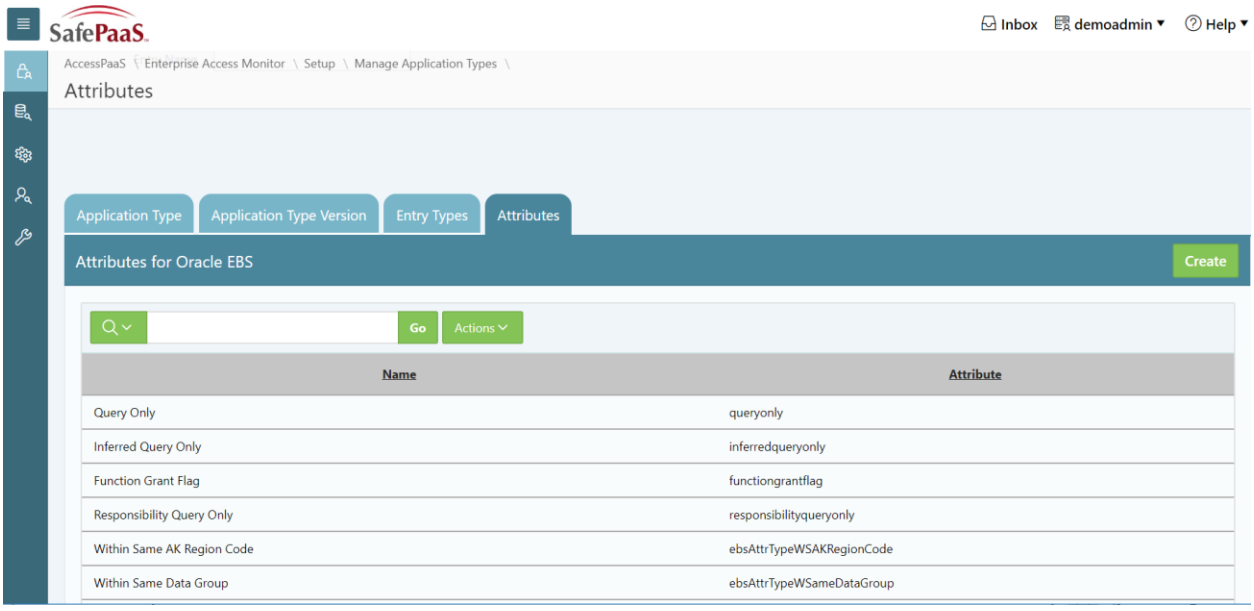
Rule Name	Risk Level	Operating Unit	Number Users	Responsibility Name	Application Name	User Menu Name	User Function Name	Violation Status Code	Activity Set	Violation Number	Responsibility Effective Date
4090 SOD: Modify Employee Position & Define Payroll Information EBS R12	HIGH	Vision Operations	4	DK HRMS Manager	Human Resources	DK HRMS POSITION	Define Position Hierarchy	Open	Modify Employee Position EBS R12	1	01-FEB-05
4090 SOD: Modify Employee Position & Define Payroll Information EBS R12	HIGH	Vision Operations	4	DK HRMS Manager	Human Resources	DK HRMS PAYROLL	Update Payroll Run	Open	Define Payroll Information EBS R12	2	01-FEB-05

SOD Violations Management within Oracle RBAC Security Model

SOD violation detection in Oracle RBAC requires access rule configuration in SafePaaS that includes advanced security option such as role inheritance and grant flag setup in Oracle menus. SafePaaS Enterprise Access Monitor provides the ability to configure the RBAC security model and detect violation in Roles using a common rule set for Responsibilities and RBAC security models in Oracle E-Business Suite. The following screenshot show the Oracle EBS RBAC security model configuration:



Oracle RBAC support advance security attributes to implement ANSI standards. SafePaaS RBAC configuration enables the Rules engine to ensure these attributes are analyzed during when the SOD rules are tested against the Roles based Access Model, as shown below:



References

Oracle E-Business Suite System Administrator's Guide - Configuration
Oracle E-Business Suite System Administrator's Guide – Security
SafePaaS - AccessPaaS Users Guide
SafePaaS - Protect Your Business and Reputation by Securing ERP Application Access