Risk Management in Role-based Applications

Segregation of Duties in Oracle

Sundar Venkat, Senior Manager, Protiviti
Tai Tam, Accounting Manager, Electronic Arts
Core Competencies – C23
Agenda

• Introductions
• Overview and Session Objectives
• Common Issues in Security Design
• Top-Down SoD, Security Design Methodology and benefits
• About Electronic Arts
• Project Meridian Background and Security Design
• Automation of Segregation of Duties (SoD) Monitoring using Oracle AACG
• Automation of Security Build
• Q & A
Introductions

**Protiviti**

**Sundar Venkat**, Senior Manager  
*Over 10 years of experience in ERP Implementation, Security and GRC Design*

**Electronic Arts**

**Tai Tam**, Accounting Manager  
*Global lead for Segregation of Duties. Over 15 years of experience in the Industry, working in various capacities in Finance, Audit and Compliance*
TOP-DOWN SoD AND SECURITY DESIGN METHODOLOGY AND BENEFITS
Common Issues in Security Design

- Insufficient understanding of the security model of ERP systems leading to a design that is not comprehensive
- Not allocating enough time in the implementation process for security design
- Not identifying and securing sensitive data prior to implementation
- Need to define a lot of manual controls increasing audit cost
Understanding SoD Design Approaches

**Bottom-up**

- No direct relationship between formal SoD policies and Oracle Responsibilities.
- Oracle Responsibilities are defined based on limited design of SoD rules.
- Oracle Responsibilities are not conflict-free.
- One-off results in each SoD test cycle.
- Heavy manual controls.
Alternate SoD Design Approach

**Top-down**

- Business process owners define formal SoD policies. These policies are used as the foundation of SoD design in Oracle ERP environment.
- "Authorized" conflicts are determined at the design level.
- Oracle functions are classified according to formal SoD policies and rules are defined by business process owners.
- Conflict-free Oracle Responsibilities are designed according to these policies.
Alternate SoD Design Approach (continued)

**Top-down**

- Each Responsibility includes a set of functions defining its unique characteristics.
- Oracle functions are categorized into Business Setup, IT Setup, and Transactional setup, ensuring consistency in the separation of the functions by category.
- Good fit for automation.
Security and SoD Design Approach

SoD Control

Automation

Authorized
Conflicts

SoD Rules
Library

Test & Reporting

SoD Policies

SoD Elements

SoD Rules Matrix

Oracle SoD Matrix

Oracle Responsibilities

Model Users

Real Users

Business Rules

Business System
Example of SoD Design Elements

Procure to Pay Transactions & Master Data

Procure to Pay Transaction Elements
- Vendor Invoice Processing
- Expense Report Processing
- Posting to G/L

Master Data Elements
- Vendor Master
- Chart of Accounts
- Bank Master Data
- Tax Master Data
- Organization Master Data
- Period-End Master Data

SoD Policy Elements

<table>
<thead>
<tr>
<th>SoD Policy Element 1</th>
<th>SoD Policy Element 2</th>
<th>SoD Policy Sub-element 1</th>
<th>SoD Policy Sub-element 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transactional Data</td>
<td>Master Data</td>
<td>Vendor Invoice Processing</td>
<td>Vendor Master</td>
</tr>
<tr>
<td>Transactional Data</td>
<td>Master Data</td>
<td>Posting to G/L</td>
<td>Chart of Accounts</td>
</tr>
</tbody>
</table>

Oracle SoD Matrix

<table>
<thead>
<tr>
<th>SoD Policy Element</th>
<th>SoD Policy Lowest-Level Element</th>
<th>Oracle Function</th>
<th>Oracle Function Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transactional Data</td>
<td>Vendor Invoice Processing</td>
<td>AP_APXPAWKB_CHECK_ACTIONS</td>
<td>Payment Actions</td>
</tr>
<tr>
<td>Master Data</td>
<td>Vendor Master</td>
<td>AP_APXVDMVD</td>
<td>Suppliers</td>
</tr>
</tbody>
</table>
Design Steps – Summary

Step 1:
Segregation of Duties (SoD) policies of the enterprise are designed. These policies are system agnostic. Client's business stakeholders provide feedback if SoD policies are relevant and if they represent risks that need to be monitored.

Step 2:
SoD Elements and Rule-set are designed based on SoD policies defined in Step 1 above. An SoD Rule comprises two policy elements that are conflicting in nature.
Design Steps – Summary (continued)

Step 3:
The Oracle SoD Rule-set represents Oracle Functions and is used as a basis to design the Oracle Responsibilities and Request Groups.

Step 4:
Responsibilities are designed in such a way that conflicting elements are not defined within the same responsibility.
Benefits

- Provides a business view of Oracle Responsibilities and uses business-user friendly language.
- Oracle Functions are grouped into a brief list of business activities.
- The Design templates provide easy drill-down to Oracle Functions from business activities.
- Custom Responsibilities and Request Groups are designed based on business activities.
- The Design includes Responsibility and Request Group matrices showing SoD conflicts.
PROTIVITI'S SECURITY DESIGN & BUILD PROCESS
Process Flow – Design Oracle Responsibilities

START

Design of SoD Rules

SoD Policies, Elements & Rule-sets in Matrix Format

Extract Super User Responsibility Definition

Oracle EBS

Design Responsibilities

Oracle Functions

1. Group functions into business activities
2. Define Activity Type
3. Define default Privilege Type
4. Determine Inquiry functions
5. Determine SoD Conflicts

Design sign off

Responsibility Design review by business users

END
Automated Responsibility Build using proprietary tool

- The tool uses the System Administrator User interface on Oracle E-Business Suite to build responsibilities. No transactions are performed on the database (back-end). This minimizes risks of data inconsistencies when moving responsibilities from one environment to another.
- 'Custom' responsibilities are built using the concept of menu and function exclusions, not customizing seeded responsibilities.
- Pre-defined 'Built' templates available for various releases of Oracle E-business Suite.
- Tool processes large volumes of transactions in a few hours.
Automated Responsibility Build using proprietary tool (continued)

Examples of pre-defined templates include:

- Build 'Custom' Responsibilities and 'Custom' Menus
- Exclude Menus and Functions
- Assign seeded request groups to responsibilities
- Assign Custom Reports, Forms and Functions to responsibilities
- Assign FND Profile Options to responsibilities
- Assign Security Profiles to responsibilities
- Assign Multi Organization Access Controls (MOAC)
- Assign Inventory Organizations to responsibilities
Process Flow – Build Oracle Responsibilities

1. Replicate Oracle responsibilities for each operating unit
2. Exclude menus and functions based on design
3. Assign Request Groups
4. Assign Security rules (if applicable)
5. Assign MOAC Security Profiles (if applicable)
6. Assign Inventory Orgs (if applicable)
CASE STUDY:
PROJECT MERIDIAN – ERP
TRANSFORMATION TO ORACLE R12
Project Meridian is Part of a Larger Effort

Earlier Transformation Projects
(COA, Global Planning, P&L/Reporting, Product MDM, GPO Supply Chain-related)

Phase 1 – Deploy following Oracle modules in R12
Procurement: iClick + iExpense & iProcurement
Finance: General Ledger, Accounts Payable, Indirect Purchasing, Fixed Assets

Phase 2 – Deploy following Oracle modules in R12
Publishing: Inventory, Order Management, Pricing, Supply Chain
Finance: AR, Trade Management, Advanced Collection, Costing
Online Publishing (Digital Order to Cash)

Meridian will switch primary focus to revenue generation processes
Project Meridian – Objectives

- Global Single Instance
- Global Business Process Standardization
- Achieve Operational Efficiency
- Minimize Customization
- Minimize Development Cost
- Cost Efficiency
Project Meridian – Security Design Objectives

Design SoD Rule-set to address risks in new Oracle R12 modules

Minimize SoD Risks on Oracle R12 custom responsibilities using the SoD Rule-set as a basis
ORACLE SECURITY DESIGN & AUTOMATED SoD MONITORING
Identify and rank SoD risk with various Oracle access scenarios in key business process areas.

- Financial Close Process – General Ledger
- Procure to Pay Process – AP, Purchasing and Fixed Assets
- Order to Cash Process – AR, Pricing, Customer Master and Sales Invoicing

<table>
<thead>
<tr>
<th>#</th>
<th>Process</th>
<th>SoD Rule Descriptions</th>
<th>FY11 Rankings</th>
</tr>
</thead>
<tbody>
<tr>
<td>29</td>
<td>Procure to Pay</td>
<td>Supplier Master; AP Payments</td>
<td>Low</td>
</tr>
<tr>
<td>30</td>
<td>Procure to Pay</td>
<td>Supplier Master; Payables Invoice Entry; AP Payments</td>
<td>High</td>
</tr>
<tr>
<td>31</td>
<td>Procure to Pay</td>
<td>Supplier Master; Purchase Order Entry; Receiving Transactions</td>
<td>High</td>
</tr>
<tr>
<td>32</td>
<td>Procure to Pay</td>
<td>Supplier Master; Requisition Entry; Receiving Transactions</td>
<td>High</td>
</tr>
<tr>
<td>33</td>
<td>Order to Cash</td>
<td>AR Approve Adjustments; AR Cash Receipts</td>
<td>Moderate</td>
</tr>
<tr>
<td>37</td>
<td>Order to Cash</td>
<td>AR Approve Adjustments; Sales Order</td>
<td>Moderate</td>
</tr>
<tr>
<td>38</td>
<td>Order to Cash</td>
<td>AR Approve Adjustments; Sales Pricing</td>
<td>Moderate</td>
</tr>
<tr>
<td>39</td>
<td>Order to Cash</td>
<td>AR Cash Receipts; AR Customer Master</td>
<td>Moderate</td>
</tr>
<tr>
<td>40</td>
<td>Order to Cash</td>
<td>AR Cash Receipts; AR Debit Memo</td>
<td>High</td>
</tr>
<tr>
<td>41</td>
<td>Order to Cash</td>
<td>AR Cash Receipts; AR Sales Invoicing</td>
<td>High</td>
</tr>
<tr>
<td>42</td>
<td>Order to Cash</td>
<td>AR Cash Receipts; Sales Agreements</td>
<td>Low</td>
</tr>
</tbody>
</table>
Automated Monitoring of SoD Using AACG (continued)

2. Develop SoD rules with applicable Oracle functional elements covering the Oracle access scenarios and build them in AACG.
Automated Monitoring of SoD Using AACG (continued)

3. Set up the AACG Global Conditions and Global Path Conditions to automatically exclude certain operating units, responsibilities, users or functions from being included in the conflict analysis.

4. Run AACG conflict analysis on selected key SoD rules to detect SoD violations on a regular basis (e.g., quarterly or annual).
Automated Monitoring of SoD Using AACG (continued)

5. Analyze the conflict extract reports to eliminate false positives and identify true intra and inter responsibility conflicts.

6. Work with the business owners to determine proper remediation actions such as remove certain functions from the responsibilities and/or change the user assignments.
Benefits from Monitoring SoD Using AACG

1. AACG provides an auditable framework and process for SoD control
2. Automated process in assessing SoD conflicts raises confidence level of the external auditors
3. Discover SoD conflicts related to hidden functions which manual reviews won't likely detect
4. SoD rule-set provides solid guidelines for business owners to consider when approving user access
5. SoD rules can be set up with any combination of functions or access points to fit different business scenarios
6. Detect any type of conflicts at any time
## Benefits of Automated SoD Monitoring Using Oracle AACG

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Policy Listing</strong></td>
<td>Stores a repository of SoD rules for Oracle E-business suite across Financials, Procure to Pay, Order to Cash, Human Resources, etc.</td>
</tr>
<tr>
<td><strong>SoD Detection</strong></td>
<td>Identifies SoD conflicts based on Oracle ERP environment</td>
</tr>
<tr>
<td><strong>Authorized Conflicts</strong></td>
<td>Provides the ability to configure exceptions</td>
</tr>
<tr>
<td><strong>Reporting</strong></td>
<td>Detects what access users have and what users can do; generates conflict reports for both within Oracle responsibility and multiple responsibilities assigned to users</td>
</tr>
<tr>
<td><strong>Continuous Monitoring</strong></td>
<td>Acts as an effective monitoring tool and helps prevent fraud by limiting what users can do</td>
</tr>
</tbody>
</table>