JD Edwards EnterpriseOne

WHITE PAPER

Creating an Efficient, Sustainable Security Model:

Best Practices for Role Design

APRIL 2017

ORACLE Gold Partner

Q Software

ERP audit, security & efficiency

www.qsoftware.com

For JD Edwards EnterpriseOne:

Creating an Efficient, Sustainable Security Model: Best Practices for Role Design

CONTENTS

US Headquarters:
5889 Greenwood Plaza Blvd, Suite 401
Greenwood Village, CO 80111
United States
Tel: 720-390-7970

UK & EMEA Headquarters:
Connect House, Kingston Road,
Leatherhead, KT22 7LT United Kingdom
Tel: +44 (0) 1372 700852

ORACLE Gold Partner

Q Software

INTRODUCTION

Designing a good Role Structure is the cornerstone of an efficient, sustainable Security Model.

Well-designed Roles make your model 'future-proof' by giving you flexibility to adapt quickly to business change and scalability to support business growth.

Good Role design also minimizes the amount of time and effort your CNC Admin team needs to spend managing security.

Poorly designed Roles make security difficult to manage and have a negative impact on application performance due to:

- More lines of security in your live system
- More User level security
- Duplication of security elements
- Role overlap.

They also make it difficult to see what Users can access and how, creating more work for your highly trained technical staff, as they need to spend time resolving support calls on access problems and researching information in response to auditors' queries.

In this document, our goal is to show you how to design a resilient Role Structure that is flexible, easy to understand and just as easy to maintain. You can use this approach whether you are planning a new security implementation or re-engineering existing security.

We will also explain how our Security Manager Pro product helps you to implement and manage your Security Model efficiently within JD Edwards EnterpriseOne (E1).

ROLE DESIGN STRATEGY

The objective of Role-Based Access Control is to reduce the complexity and cost of administering security. To achieve this, your Role Structure needs to give you:

- Efficient security management: it should enable you to apply routine E1 Security changes quickly and easily
- Clarity: it must be easy to establish exactly who can access what
- Optimum performance: good Role design minimizes the overall number of security records
- Integrity: working in conjunction with conflict-checking tools, a good Role Structure enables you to eliminate Multiple Role security conflicts and Segregation of Duties violations
- Flexibility: as your business processes evolve or change you need to be able to adapt your Roles and duties with minimum effort to avoid disrupting your business
- Scalability: whether your business grows organically or through acquisitions, you need to be able to roll out security to new sites or business units quickly and efficiently, without having to start from scratch every time.

STEP 1: DEFINING YOUR E1 ROLES AND JOB DUTIES

Like most successful ERP endeavors, Role Design cannot be an exercise for the IT department alone.

Key business users need to take ownership of the Roles and be actively involved in defining the processes and duties, including listing the relevant JD Edwards EnterpriseOne applications and specifying the appropriate security access.

Every Role should have a designated business Role Owner, who should also periodically review the Role to ensure that the access granted is still appropriate for the job requirements.

Involving Business Users is a critical success factor in creating a sustainable Role Structure

We recommend that you set up Role Design Workshops for each functional team (for example Accounts Payable (AP), Accounts Receivable (AR), Customer Service, etc.) to work with your primary and secondary E1 security/admin users. For the best chance of long term success, this should include key business managers (the designated Role Owners) who have knowledge of job duties and the E1 applications needed to perform them.

Using an Excel spreadsheet, list out:

- all employees that use E1 applications
- their job titles
- their job duties
- owner(s) for each job title.

Next, examine the job titles to determine if people with the same titles have the same duties, e.g. do all AP Clerks have the same job duties?

For job titles where people assigned have different duties, create additional job titles to cater for the differences. For example, if you have five AP clerks and one of them has some additional responsibilities, you can create a job title 'AP Clerk' with the common duties that they all perform, plus an additional job title of 'AP Clerk 1' with just the additional duties that the fifth clerk needs.

The goal is to create one E1 Role for each job title, but remember that users can have more than one Role. We can assign the right combination of Roles to individual people to give them all the duties they need to perform their jobs.

**Don't be tempted to resort to User level security.
You'll regret it later!**

If you find yourself thinking along the lines of 'Sally needs AP Clerk and some of the AP Supervisor tasks. But I can't let her have access to all the Supervisor tasks,' you must NOT:

- assign some additional security at the user level or
- make some changes to the Supervisor role and hope it doesn't mess up the other Supervisors.

The best solution is to give Sally an additional Role which contains the appropriate Supervisor duties.

Creating additional Roles in this way makes it easier to find out exactly what users can access and to ensure that you only grant them the access they need to do their jobs. With 35% of ERP customers experiencing fraud every year, you need to be sure that your processes and data are fully secured. The most efficient and effective way to sustain watertight security is to manage it exclusively at Role level and avoid User level security altogether.

Granting security at the User level has other significant disadvantages:

- **It degrades application performance** by significantly increasing the size of your security table (F00950).
- **It increases your audit costs** because it will force your auditors to go through all Users to determine if any User level access has been granted which supersedes Role level security.
- **It increases the security management workload.** When your security administrator needs to make simple changes to Role security, they will have to spend time checking if anyone has User level security that also needs to be taken into account.

STEP 2: DEFINING JOB DUTIES AND MAPPING THEM TO E1 APPLICATIONS

The next task for the Workshop team is to go through the job duties and list all the E1 applications (including versions if necessary) that are required to perform each duty, as well as the types of security needed. For example:

- Job Duty: Voucher Entry
- Applications: P0411, P0411S, P0411SV
- Security Types:
    - Type 1 (Action) Add, Change, Delete
    - Type 3 (Application) Run

You may also create duties that are defined as 'Full Access' and 'Inquiry Only'. For example, you may have a duty for Batch Processing (P0011) where some users have Add, Change, Delete access and others have read only access (Ok/Select, Scroll to End). These can be defined as two distinct duties: Batch Processing (Full Access) and Batch Processing (Inquiry Only).

When you create your duties and list the E1 applications required to perform them, we suggest that you include as few applications as possible. Your security model will retain greater flexibility with duties that can be reused by all users under any changing business circumstances.

Duties	Voucher Entry	Payment Processing	Receipt Entry	Receipt Matching	Management Reports	General Journal Review	Batch Processing
Applications	P0411B	P04570	P4312	P0413	R04413	R09801	P0011
	P0411	R04571	P43214	P0413M	R04423	P0911	
	P0411S	R04572				P0911B	
	P0411SV						

The next step is to map all Job Roles to their job duties:

Roles	AP CLERK I	AP CLERK II	AP CLERK III	AP MANAGER
Duties				
Voucher Entry	Full Access	Full Access	View Only	View Only
Payment Processing	View Only	Full Access	Full Access	View Only
Receipt Entry	Full Access	Full Access	Full Access	View Only
Receipt Matching	View Only	Full Access	Full Access	View Only
Management Reports	No Access	No Access	Full Access	View Only
General Journal Review	View Only	Inquiry Only	Full Access	View Only
Batch Processing	No Access	No Access	Full Access	View Only

Your completed matrix should include every E1 Role, mapped to job duties which are mapped to applications and their security definitions.

Data Roles

You also may need to define data access restrictions, for example where staff should only be able to access data relating to a particular business unit or geographical location.

If data access restrictions affect many Roles (rather than just a few), you can create an E1 Role for each distinct data restriction. For example, if data viewing needs to be restricted by country, you might create Data Roles called 'US Data' and 'Canada Data'. You can then assign the appropriate Data Role to individual users along with their Job Roles

Next you need to define the restriction, and for this we recommend you to use Inclusive Row Security. For example, for Business Unit Security, you can create Row Security for *ALL Objects and the data alias MCU and define the range of data that is accessible.

This approach significantly reduces the Role maintenance overhead. For example, without Data Roles, you might end up with separate Roles for 'APClerk-US' and 'APClerk-Canada'. With Data Roles, you only need one Role for AP Clerk and Data Roles for US and Canada.

Where data restrictions are only required for a few Roles, you can add the security directly to the Roles. Later we'll explain how Security Manager Pro enables you to create a single reusable component that defines this security. For now, treat your data access the same as a job duty. Using your matrix, map each role to its data access duties as appropriate.

USING **SECURITY MANAGER PRO** TO CREATE AND MANAGE YOUR SECURITY MODEL

However you choose to manage your security, the process that we've outlined above will help you to design a good Role-Based Security model.

Here's a summary of the main tasks:

- List each user, their job title and job duties
- Create E1 Roles based on job titles
- Create job duties based on user job requirements
- Define the E1 Applications that make up job duties
- Define Data Roles
- Map job duties and data duties to their E1 Roles.

In this section, we'll show you how quickly and easily you can create and deploy your Security Model using Security Manager Pro.

Before we dive into that, it will be helpful to explain some key concepts in our approach that simplify security management and enable you to validate your security before you write it to your live security table.

Prebuild Security:

With standard E1, you add security records one at a time for a specific User or Role, and the records are added directly to the live P00950 security table.

With Security Manager Pro, as you define security settings they are held in a Prebuild table – a custom table held within your JD Edwards environment.

You start by defining the security for the Duties (see below), then add the relevant Duties to your Roles. You can then assign the Roles to Users as appropriate.

When you're ready to build your live security (i.e. write the security records into the F00950), you can first run automated checks to identify Segregation of Duties (SoD) violations or Multiple Role conflicts that would be introduced by the proposed access. This gives you chance to investigate and resolve any conflicts BEFORE they are introduced to your live environment.

As you refine your security settings you can build the live security Role by Role or for a group of Roles. When settings are updated, the affected Roles are flagged to show where the changes have not yet been built into the F00950.

Prebuild security allows you to keep access clean by proactively checking for Segregation of Duties violations and Multiple Role Conflicts before they enter the system.

Components: Duty-based groups of security records

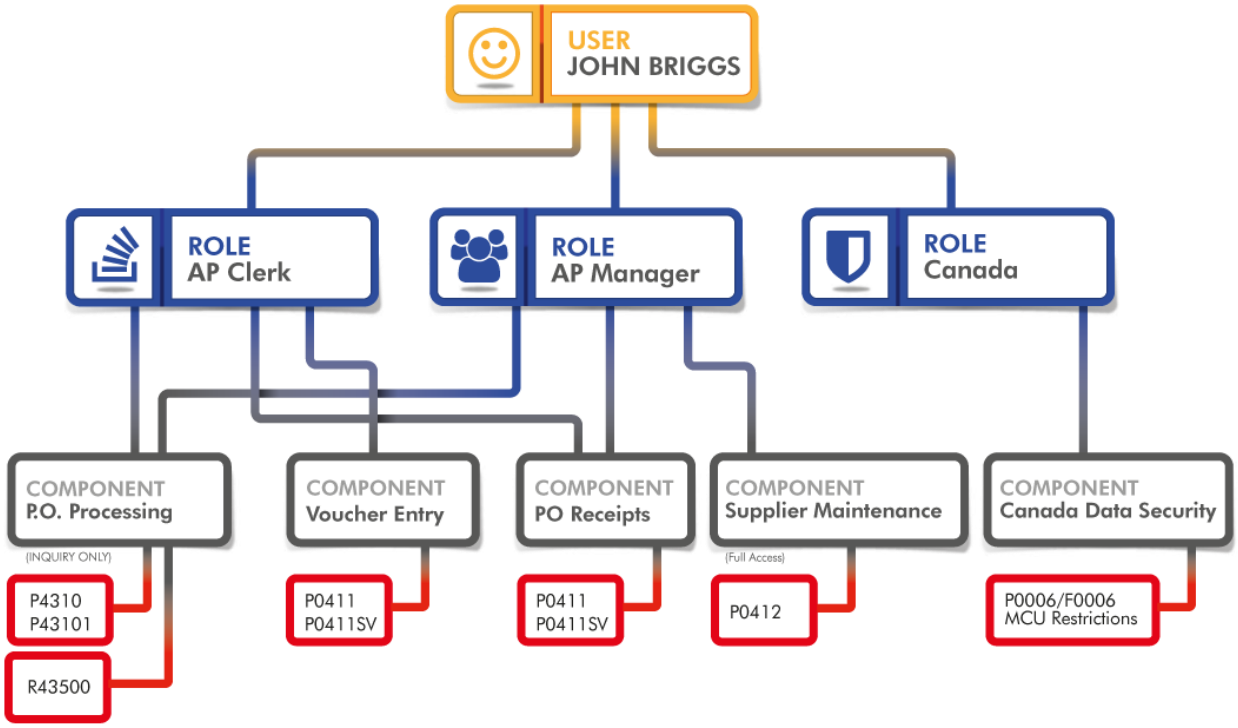
Our use of Components delivers much greater efficiency than is possible with Role-based security in standard E1.

During the Role Design process above, we defined Job Duties that individuals perform as part of their Job Roles.

We also identified that many of the Duties (e.g. Voucher Entry) need to be made available to several different Roles.

With Security Manager Pro you can create a Security Component for each Duty; i.e. a group of security records which define the E1 applications and the type of security access needed to perform the Duty.

The Component can then be added to every Role that needs to perform that Duty; you just add the Component, and that automatically brings all the relevant security information with it. If the Duty changes, you only need to update the Component. You DON'T need to check and change every Role affected; the changes will automatically be applied to all the affected Roles and Users when you rebuild the security.



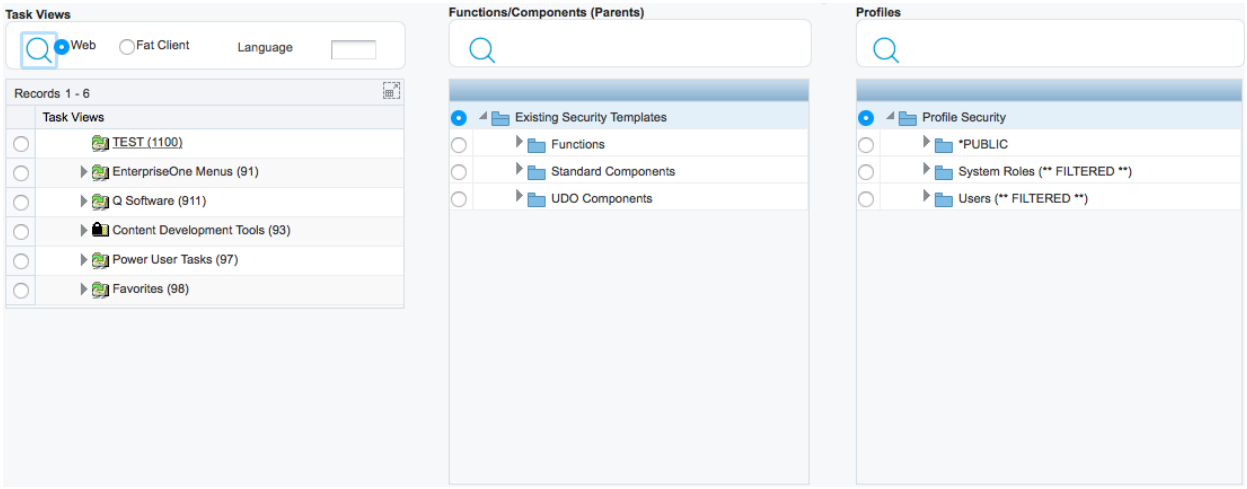
Grouping security records into Duty (Business Process) based Components cuts the workload of security set-up and management by up to 80%

Security Manager Pro: fully integrated into your JD Edwards environment

Security Manager Pro is built using the E1 Software Development Tool Kit. All its objects are JD Edwards objects, created on a fat client development workstation, then generated, checked in and deployed, just like all JD Edwards applications. This means that all your Role, User and Security data is stored within your JDE system rather that in insecure spreadsheets.

Security Manager Pro has three main user interfaces:

- Menu
- Security Model
- Roles and Users.



The **Menu interface** (Task Views) allows you to create or edit Task Views. It also allows you to generate security from menus or vice versa. You can also easily copy Task Views between environments.

The **Security Model interface** (Functions/Components) is where you create your Security Components – ie define the E1 applications and the types of access needed for a Duty.

You can then create Security Functions equating to your E1 Roles, and add the Components (Duties) that are required for the Roles, as specified in your Role Design matrix.

The **Role/User Interface** (Profiles) is where you will associate your Functions/Components to the E1 Roles. This is also where you check for conflicts and SoD Violations, before writing the security records to the F00950.

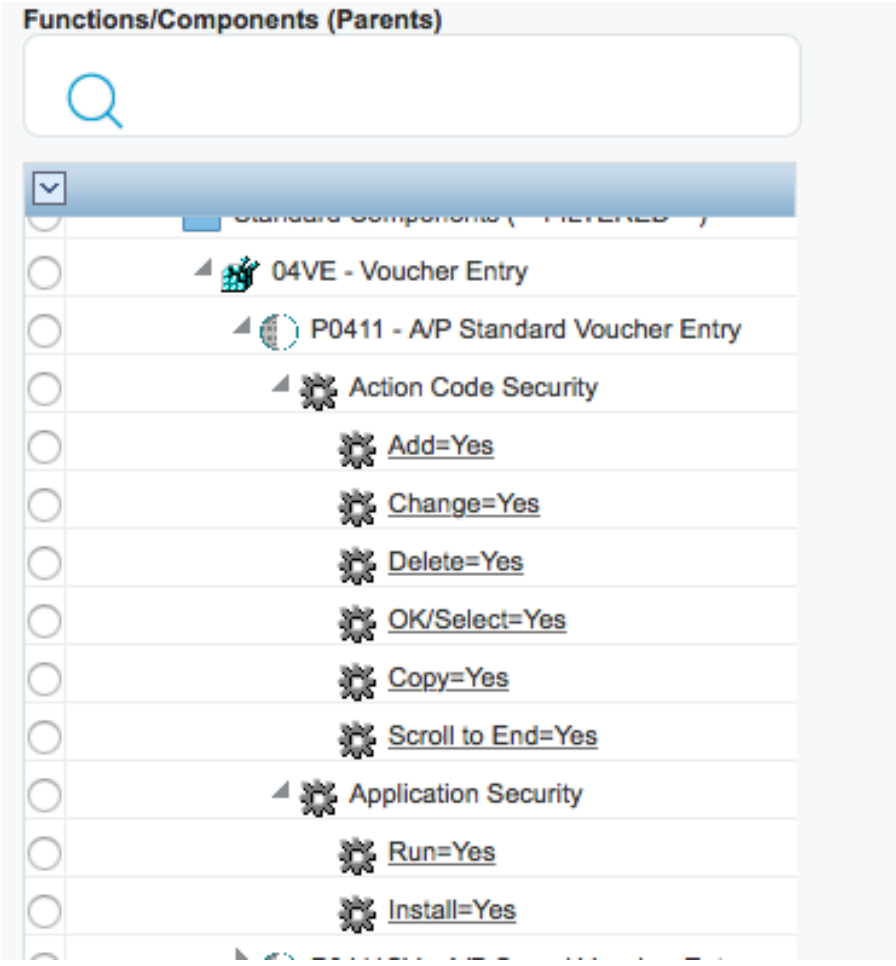
So now we're ready to look at how to use Security Manager Pro to create the Security Model that you and your business users designed.

Step 1: Build Components equating to your Duties

For each Duty that you have defined in your Role Design matrix, create a Component that contains all the identified E1 applications and their security definitions.

Duties	Voucher Entry
Applications	P0411B
	P0411
	P0411S
	P0411SV

Using Voucher entry as an example, the following screen shot shows how you create the Component:



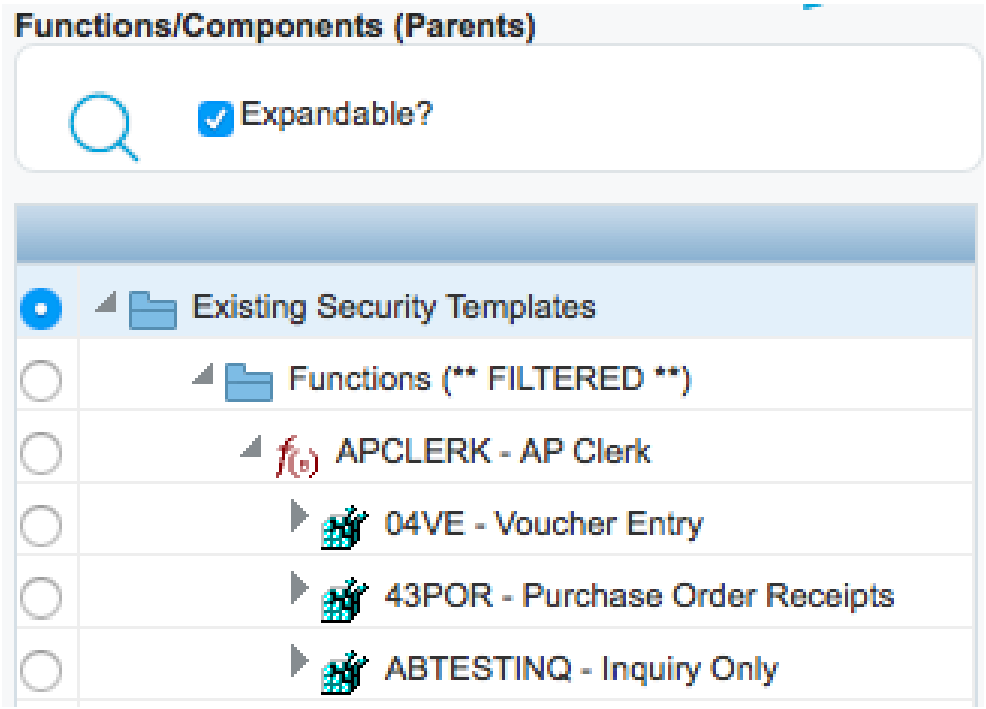
Step 2: Build Functions equating to your E1 Roles

Next create a Function for each E1 Role that you have defined in your matrix.

You can then add all the relevant Components to each Function.

Roles	AP CLERK I	AP CLERK II	AP CLERK III	AP MANAGER
Duties				
Voucher Entry	Full Access	Full Access	View Only	View Only
Payment Processing	View Only	Full Access	Full Access	View Only
Receipt Entry	Full Access	Full Access	Full Access	View Only
Receipt Matching	View Only	Full Access	Full Access	View Only
Management Reports	No Access	No Access	Full Access	View Only
General Journal Review	View Only	Inquiry Only	Full Access	View Only
Batch Processing	No Access	No Access	Full Access	View Only

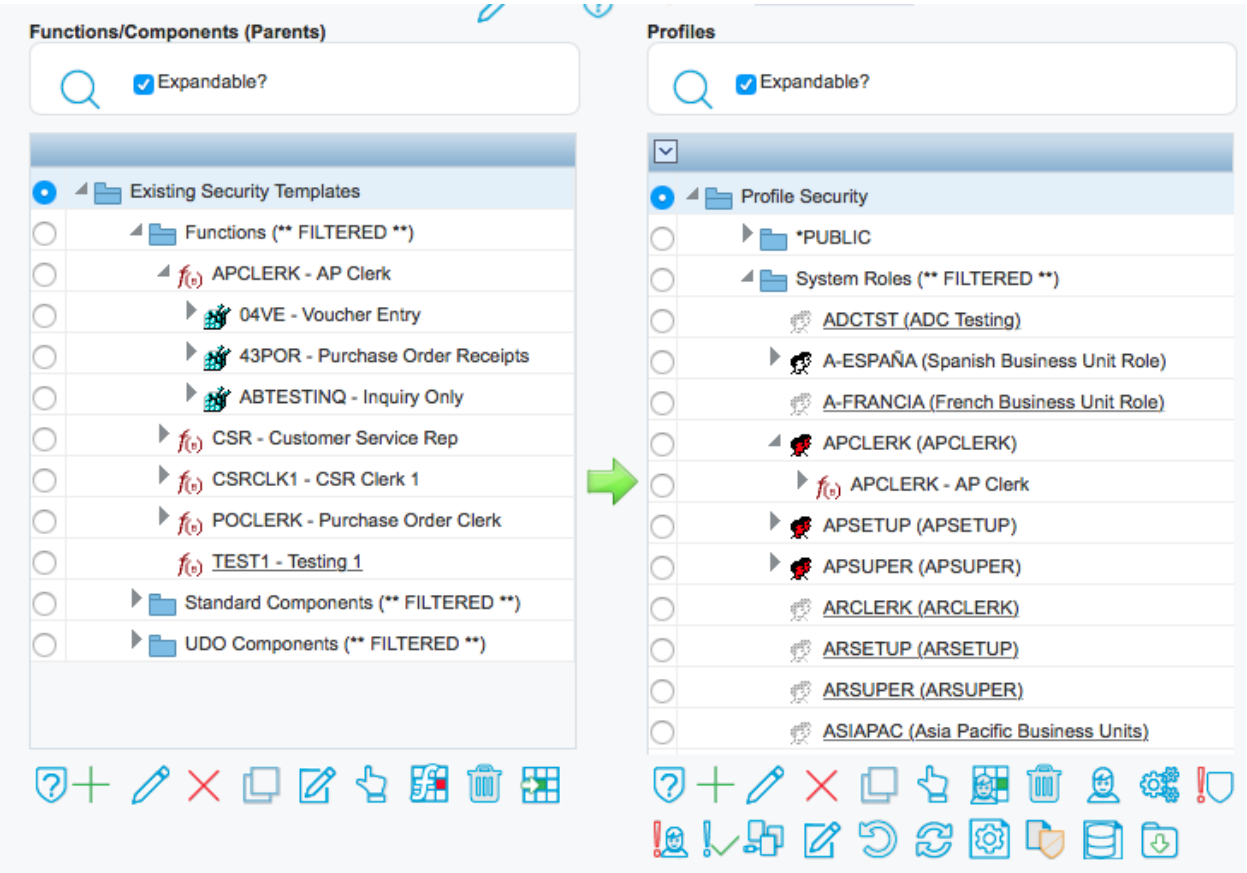
In the example below, we've created a function APCLERK and added three components.



Step 3: Associate the Security Functions with their E1 Roles

Now we're ready to add these security records to the E1 Role APCLERK.

All you have to do is highlighting the APCLERK Function in the Functions/Components interface and the APCLERK Role in the Profiles interface, then click:



As you can see above, the red Role icon next to APCLERK in the Profiles interface shows you that security for that Role needs to be rebuilt.

Once the Function is associated with the E1 Role, if you update the Function (e.g. for example if you add or remove Components), the Role is automatically flagged red to indicate that changes have occurred since the last security build.

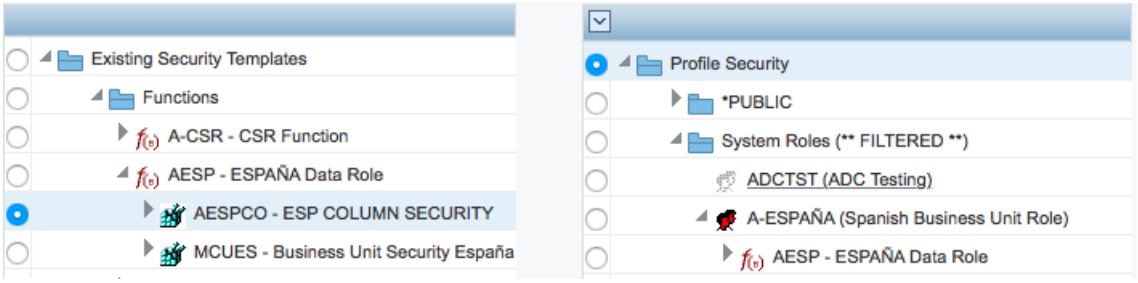
Data Roles

Creating Components and Functions for your data security is no different from the examples shown above.

You can create a Component for each distinct data restriction which defines the range of data that is accessible.

To create Data Roles, you then add one or more of these data security Components to a Data Function and then associate the Function with its designated E1 Role.

In the example below, I've created a business unit (MCU) Component and a column security Component that only allow access to data related to Spanish Business Units.

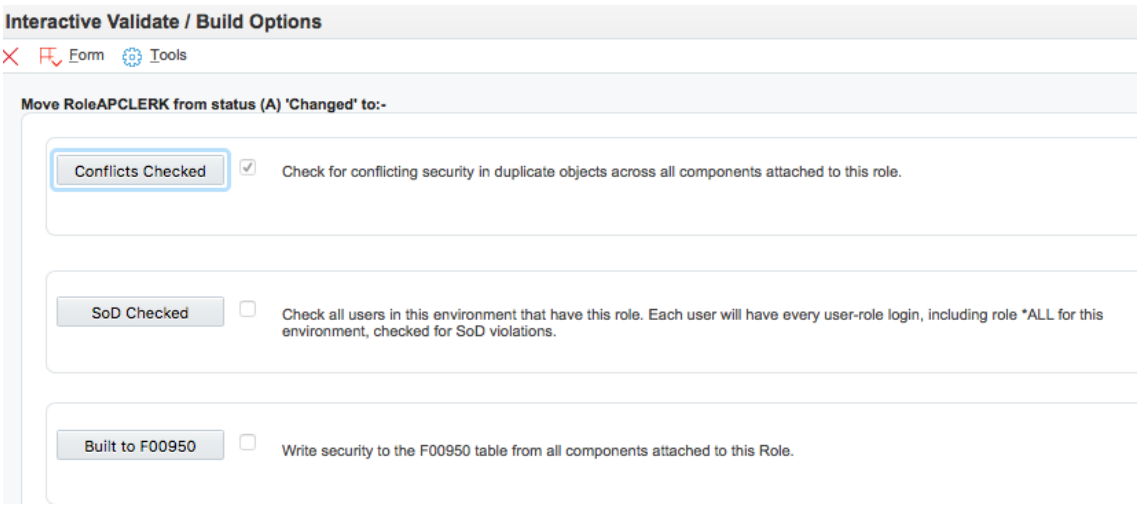


I've attached these two Components to a Data Function called 'AESP – ESPAÑA Data Role,' then associated the Function to an E1 Role called A-ESPAÑA. Now I can assign the Data Role to all the relevant Users.

By having Data Roles, I don't have to add the data security to every role. I can establish one global Role for AP CLERK, and assign Data Roles to individual users as appropriate for their location/business unit.

Step 4: Proactive Conflict / SoD Check and Build

Now all that's left to do is to check for potential conflicts and update the F00950. You can do this on a Role by Role basis, or for many Roles in one Build.



Where Users have Multiple Roles, the validation process checks to see if the Roles contain conflicting security, by looking at the Net Effect on access granted by the Multiple Roles.

The SoD check tells you if your changes would permit a Role or Users assigned to the Role to violate your SoD rules.

Once the required checks are completed and any issues are resolved, you can run the Build, which writes all the appropriate security records to the F00950.

CLOSING THOUGHTS

Security Manager Pro makes your security make sense. By taking it out of a grid and providing you with a user interface that allows you to group and categorize the security, it is much easier and quicker to create a new model.

It also provides tools to import and your existing security so that you can redesign your model without starting from scratch.

The Role Design workshops and the resulting documentation are key to the success of any security project. If you find that your existing security is too difficult to manage, you can use these same guidelines to redesign your security model to gain control and visibility of your security.

Q Software

An Independent Software Vendor and Oracle Gold Partner, Q Software delivers security and compliance solutions and services for users of JD Edwards EnterpriseOne, JD Edwards World, and Oracle E-Business Suite. Our products help customers to protect their businesses from fraud whilst significantly reducing the cost, effort and complexity of managing risk and demonstrating regulatory compliance.

If you would like to discuss any of the points raised in this document or find out more about how our products and services can help you, please email your Q Software contact or sales@qsoftware.com or visit www.qsoftware.com