

JD Edwards EnterpriseOne

WHITE PAPER

Auditing JD Edwards Security: 6 Common Mistakes

FEBRUARY 2018





For JD Edwards EnterpriseOne:

Auditing JD Edwards Security: 6 Common Mistakes

CONTENTS

INTRODUCTION	3
SEGREGATION OF DUTIES	3
COMMON MISTAKE # 1: INAPPROPRIATE SOD RULES	4
RECOMMENDATIONS:	4
COMMON MISTAKE # 2: OBJECT SHORTSIGHTEDNESS.....	5
RECOMMENDATIONS:	6
COMMON MISTAKE # 3: MISSING FACTS OF FUNCTIONALITY	7
RECOMMENDATIONS:	10
COMMON MISTAKE # 4: LACK OF ATTENTION TO SECURITY DETAILS	11
RECOMMENDATIONS:	12
COMMON MISTAKE # 5: NOT UNDERSTANDING FALSE POSITIVES	13
RECOMMENDATIONS:	15
COMMON MISTAKE # 6: NOT CONSIDERING THE BIGGER PICTURE.....	16
ENVIRONMENTS.....	16
ROLE CHOOSER	16
ROLE SEQUENCING	16
AUDIT YOUR AUDIT RULE SET!	17
IF YOU NEED HELP	17

TRADEMARKS:

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



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

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

sales@qsoftware.com





INTRODUCTION

It's no big secret that JD Edwards security is very complex. But if you've ever tried to audit your JD Edwards security, I'm sure you've found that to be an even greater challenge. (I speak from experience!)

Whether your main objective is to satisfy your auditors, or to implement and test robust controls to protect your organization from fraud, the objective of this paper is to shed light on some of the major issues that make it difficult to produce accurate audit results.

I hope you find it enlightening!

About the Author:

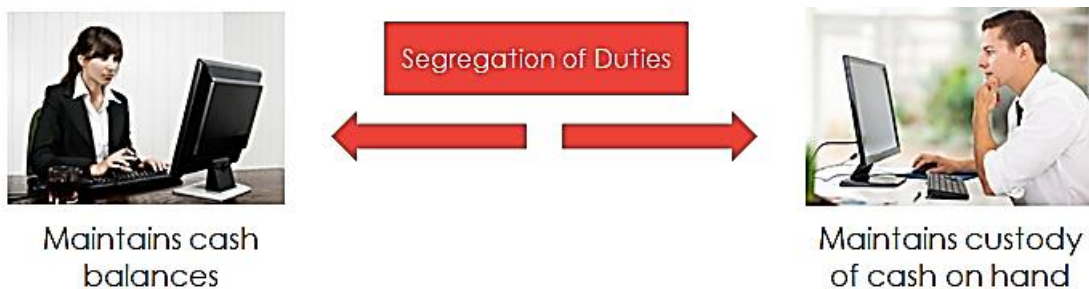
Carrie Curry is Senior Delivery Manager at Q Software, working as a consultant, implementer, trainer and all-round security guru.

She has over 15 years' experience in ERP systems, having previously been employed as a Business Process Analyst, Systems Analyst and ERP Security Team Lead at a multi-billion \$ public company. Developing her own methodologies by trial and error, she constructed a compliant security model on JD Edwards EnterpriseOne, resulting in successful audit outcomes for her employer.

SEGREGATION OF DUTIES

Much of auditing security in JD Edwards revolves around Segregation of Duties (SoD), and identifying whether users can violate your SoD Rules.

Segregation of Duties (SoD) is an internal control designed to prevent error and fraud by ensuring that at least two individuals are responsible for the separate parts of any task. It involves breaking down tasks that might reasonably be completed by a single individual into multiple tasks, so that no one person is solely in control.



Your SoD model is the set of Rules that you define to denote which tasks should be separated, and auditing SoD involves identifying users whose access would enable them to violate your SoD rules.

When you're auditing, one of your main objectives is to ensure that these tasks are always segregated.



COMMON MISTAKE # 1: INAPPROPRIATE SOD RULES

To obtain reliable auditing results, you need to create a good set of Segregation of Duties Rules, geared to your organization and its activities, as well as the software that you use.

This may feel like a daunting task, so it can be tempting to download a set from the Internet or adopt a matrix that you've used at a different company. But that won't work. Another company's rules are highly unlikely to be appropriate for your business, and your processes.

The starting point for effective auditing is to establish an appropriate set of Segregation of Duties Rules.

Segregation of Duties rules should:

- Be unique to your business
- Be based on the processes within your business
- Include the software that is being used within the processes. Many rules are designed to segregate tasks which are carried out using two or three different programs, so it is important to define the rules for your processes, identifying the specific objects used.
- Take into consideration which pieces of the rules are conducted within (and outside of) the software
- Be based on your company's risk tolerance. For example, the risk matrix may consist of 200 rules, but perhaps the risk of fraud, or the potential for loss, incurred by some of the rules is very low. You may decide to check for violations for low-risk rules less frequently, or not at all.

It's fine to take a sample set of rules and use them as a starting point, but you need to adapt them to reflect your processes, and the specific software modules and objects that you use, to arrive at a set that is appropriate and manageable for your company.

RECOMMENDATIONS:

Consult appropriate resources

It's advisable to work with your internal and external auditors to make sure that your risk matrix reflects the risks that they are auditing for.

Security experts can also be of assistance. For example, at Q Software we have compiled a set of the SOD rules that are most commonly used by JDE customers in various lines of business, and which can be adapted and refined to meet your needs.

Research the process behind Segregation of Duties rules

There is a wealth of information available from audits firms and bodies such as ISACA and the IIA which can help you understand the risk matrix and how to define your rules.

Recruit business involvement

Most important of all – involve the business!



It's often the case that IT, specifically CNC or system administrators, are responsible for the set-up of SoD rules and reporting, but they don't necessarily understand every application and how it functions.

Business users are the ones who understand the processes, so it is very important to involve them in identifying the risks and defining appropriate Segregation of Duties rules.

COMMON MISTAKE # 2: OBJECT SHORTSIGHTEDNESS

When you're setting up your SoD Rules, you need to include ALL the ways that changes to data can be effected.

Example: Supplier Master Data

If part of the rule involves updates to supplier master data, you need to include all the applications that can update it, including, for example, Batch Upload. Even if you think that you don't use that program and it hasn't been provisioned to people, it's possible that someone may find it and use it to manipulate data, so it's important to check for it within your rules.



When defining your rules, you should aim to be proactive, which means listing all the ways that JD Edwards can affect the data, and including them from the outset.

For Supplier Master data, typically you would include application P04012: Work with Supplier Data.

But there are other applications which can affect supplier data, such as:

P01012: Address Book

P01013: Name and Address Change

P0401Z1: Work with Batch Supplier Master



Work With Addresses

Alpha Name ☐ Display Phone

Search Type ☐ Display Address

Records 1 - 1

309 Name and Address Change

Customer Number e Curry

Tax ID

Name and Address Change

Alpha Name

Mailing Name

Address Line 1

Address Line 2

Address Line 3

Address Line 4

City

State

Postal Code

Country

E-Mail Address

Work With Batch Supplier Master

Batch Number

Records 1 - 10

User ID	Batch Number	Line Number	Address Number	Tr Ac
<input checked="" type="radio"/> AP5933301	10514	100993	1.000	42451 A
<input type="radio"/> AP5933301	10519	100994	1.000	42452 A
<input type="radio"/> AP5933301	10523	100995	1.000	42453 A
<input type="radio"/> DL811727	13367	102485	1.000	4100 A
<input type="radio"/> IC8893739	15283	103390	1.000	8893739 A
<input type="radio"/> IC9027533	15194	103347	1.000	20081 A
<input type="radio"/> IC9027533	15223	103365	1.000	20091 A
<input type="radio"/> JG5941598	13134	102274	1.000	4370 UB
<input type="radio"/> JG5941598	13137	102277	2.000	4370 UA
<input type="radio"/> JG5941598	13138	102278	1.000	4370 UB

These three items are easily overlooked, and they may not currently be provisioned to users, but you should include them in your rules so that you can proactively check for them in case someone does start to use them.

Be proactive: include all the applications that can affect the relevant data.

If you have a tight security approval process, where access is approved before it is provisioned, you may think this is not so important.

But most companies only conduct a Security Workbench review against approvals once or twice a year, so there could be a long gap between the access being provisioned and the review taking place.

By being proactive and including all these applications up front, you're making sure that your SoD rules check for them in the event that they are provisioned at a later stage.

RECOMMENDATIONS:

Involve functional consultants / super users

These people will be able to help you identify all the objects which can update the various types of data.

Work with your auditor

Your external auditor will be able to give you a list of the objects that they are checking for, so that you can align your list to theirs.

Consult Oracle guides

The Oracle Knowledge Base has a lot of great information about different types of applications and their usage.

Use the object librarian and xref

These can also help you to identify the objects.

Always check the 55-59 product code and consult developers

Ask your developers whether there is anything you need to be aware of regarding that list.

Developers may often clone an object and make some changes to it, but the impact of the changes isn't always obvious from the description. So, it's important to investigate that and consider them in your SoD rules.

COMMON MISTAKE # 3: MISSING FACTS OF FUNCTIONALITY

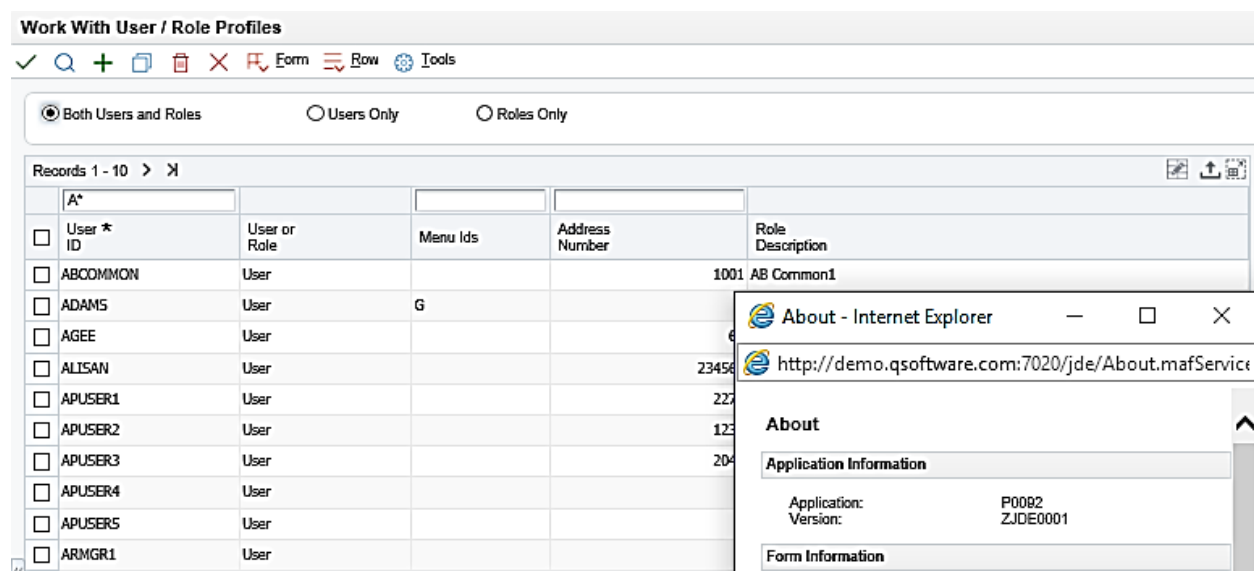
So, you've created a good set of rules and identified all the objects, but do you understand all the functionality that the objects allow?

Example 1: Pre-production access

Business Analysts may have full access in pre-production, but 'View Only' access in Production.

But some of the objects update tables that exist in the System Data Source – so they can affect the Development, the Prototype and the Production environments. Therefore, it is important to examine which tables exist in the System Data Source, the related applications, and who has access to them.

The screenshot below shows an example with application P0092, where the security table is held in the System Data Source. Business Analysts could update and add users, but didn't think that they were affecting Production.



Work With User / Role Profiles

✓ 🔍 + 📄 🗑️ ✕ 📄 Form 📄 Row ⚙️ Tools

☒ Both Users and Roles ☐ Users Only ☐ Roles Only

Records 1 - 10 > <

User *	User or Role	Menu Ids	Address Number	Role Description
<input type="checkbox"/> ABCCOMMON	User			1001 AB Common1
<input type="checkbox"/> ADAMS	User	G		
<input type="checkbox"/> AGEE	User			
<input type="checkbox"/> ALISAN	User		23456	
<input type="checkbox"/> APUSER1	User		227	
<input type="checkbox"/> APUSER2	User		123	
<input type="checkbox"/> APUSER3	User		204	
<input type="checkbox"/> APUSER4	User			
<input type="checkbox"/> APUSER5	User			
<input type="checkbox"/> ARMGR1	User			

About - Internet Explorer

http://demo.qsoftware.com:7020/jde/About.mafService

About

Application Information

Application: P0092
Version: ZJDE0001

Form Information

Example 2: User Defined Object (UDO) Security

There are various ways that you can update this table.

You can use an exit off the P00950 Security Workbench to call up the UDO Security Workbench application. In there, most of the security is related to user functionality, but, in the newer versions of JD Edwards, Data Browser is also provisioned in the UDO security workbench:

View	User / Role	User / Role Name	Object Type	Object Type Description
<input type="checkbox"/>	*PUBLIC	*PUBLIC	DATABROWSE	Data Browser
<input type="checkbox"/>	*PUBLIC	*PUBLIC	SEARCH	EnterpriseOne Search
<input type="checkbox"/>	*PUBLIC	*PUBLIC	SEARCH	EnterpriseOne Search
<input type="checkbox"/>	*PUBLIC	*PUBLIC	SEARCH	EnterpriseOne Search
<input type="checkbox"/>	*PUBLIC	*PUBLIC	SEARCH	EnterpriseOne Search

You can effect change by having access to the UDO Security Workbench, but the table can also get updated from application P98220U:

Short Description	User Defined Object Status	User Defined Object Type	User Defined Object Name
<input type="checkbox"/> Batch Journal Review			CAF0011A_1607110002JDE
<input type="checkbox"/> Batch Journal Review			CAF0011A_1607110002JDE
<input type="checkbox"/> Batch Out Bal Journal Review			CAF007032A_1607120002JDE
<input type="checkbox"/> Test Layout			CAF01012B_1603020001CJUS
<input type="checkbox"/> New Layout Addresss			CAF01012B_1603020002CJUS
<input type="checkbox"/> AB Test			CAF01012B_1603020003CJUS
<input type="checkbox"/> Customer Revision			CAF03013A_1607270001JDE

In this application there is a row exit where you can go in and provision access and it updates the same table as if you did it from the UDO security workbench.

So, there are two places where you can effect these changes, and you need to understand whether both applications are being used to apply updates. Do you need to use both methods and does it potentially incur additional risk if you allow people to provide access through P98220U? If so, you may want to consider restricting all updates from this application.

Example 3: Don't rely on the Application Description

Descriptions can be misleading!

For example, Application P0911B is described as General Journal Review, so people often assume that it only allows people to review the entries – when, in fact, it allows people to effect change.

A very common SoD rule is to separate the tasks of accessing Journal Entries and posting them. But because people think that the General Journal Review doesn't allow changes, they omit this object from the SoD rule.

Document Type	Document Number	Document Company	G/L Date	Explanation	Domestic Amount	Foreign Amount	Reverse or Void	Currency Code
##	3826	61000	31/12/2016	Record audit adjust				
%	3066	00001		Model percent reven				
%	3069	00001		Reclass Advertising E				
AE	1067	00000	30/06/2017	Post Due From Acco				
AE	1068	00000	30/06/2017	Post Due From Acco				
AE	3033	00000	30/06/2017	Post Due From Acco				
AE	3034	00000	30/06/2017	Post Due From Acco				
AE	3035	00000	30/06/2017	Post Due From Acco				
AE	3828	00001	30/06/2017	Post Due From Acco				
AE	3866	61000	30/06/2017	Post Due From Acco				

But if a user selects one of the records and clicks the green check box, it opens the Journal Entry in Application P0911 (and there is also a row exit into the Journal Entry program). So, if the user has full access to General Journal Review, he could go into the Journal Entry and use the Delete button to delete the record.

Account Number *	Amount	Account Description
200.1291	1,500.00	Intercompany Accounts R
1.1291	1,500.00	Intercompany Accounts R

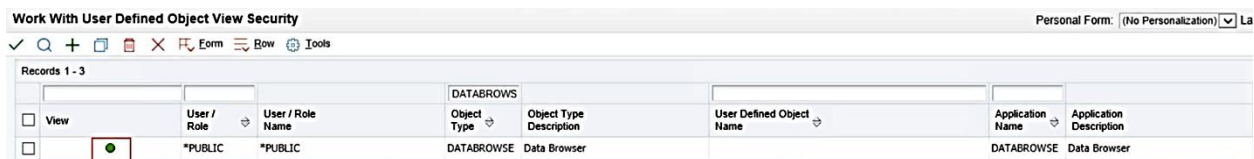


So, don't rely on the Object Description. Make sure you walk through the application and verify that a user can only review the data, and can't take any exits that will allow him to amend the records. In this instance, you may also wish to check that people who are responsible for reviewing don't have any access to P0911, to ensure that the review screen truly only gives 'View only' access.

Example 4: Be aware of new functionality introduced in the later JDE releases.

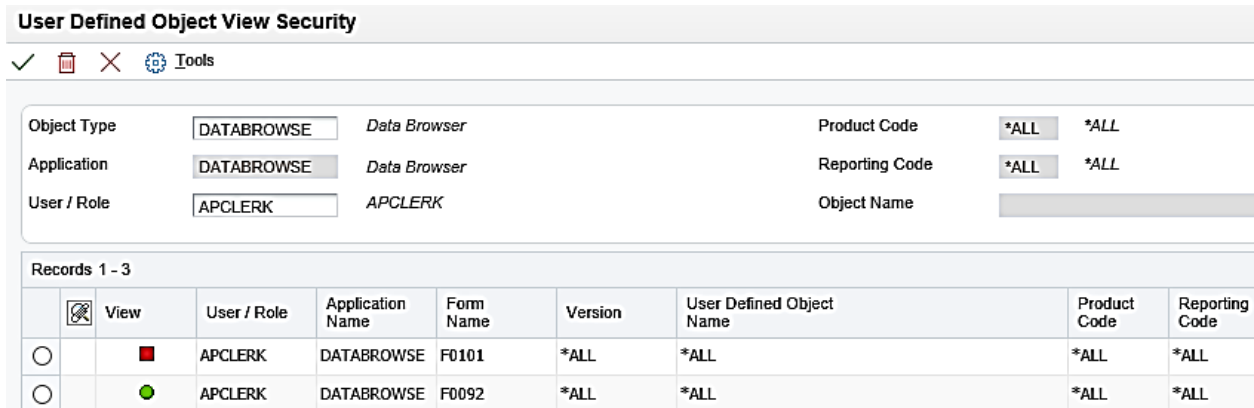
Many people think that security isn't changing much between different versions of JD Edwards, but this is not the case!

In recent releases, there is a new Data Browser UDO, which is provisioned in the UDO security workbench. In this screen, it looks like *PUBLIC has View Only access.



Work With User Defined Object View Security								Personal Form: (No Personalization) ▼ La	
Records 1 - 3									
	View	User / Role	User / Role Name	Object Type	Object Type Description	User Defined Object Name	Application Name	Application Description	
<input type="checkbox"/>	<input checked="" type="checkbox"/>	*PUBLIC	*PUBLIC	DATABROWSE	Data Browser		DATABROWSE	Data Browser	

But in later releases of JDE, you can now restrict Data Browser access by table. As shown below, I can now allow a user to Data Browse F0092, but not F0101 (for example):



User Defined Object View Security									
Records 1 - 3									
	View	User / Role	Application Name	Form Name	Version	User Defined Object Name	Product Code	Reporting Code	
<input type="radio"/>	<input checked="" type="checkbox"/>	APCLERK	DATABROWSE	F0101	*ALL	*ALL	*ALL	*ALL	
<input type="radio"/>	<input checked="" type="checkbox"/>	APCLERK	DATABROWSE	F0092	*ALL	*ALL	*ALL	*ALL	

So be aware that new functionality is being introduced in new releases, which may impact what users can and can't access.

RECOMMENDATIONS:

Involve functional consultants / super users

These people will be able to help answer questions about functions and how they operate.

Consult Oracle guides

The Oracle Knowledge Base has a lot of great information about different types of applications and their usage.

Consult functional specifications and test scripts

These should give you more information about Custom Objects.

Set up quick discovery session with a super user

Ask them to execute the process so that you can document the applications and ask questions about the functionality. You can also conduct negative tests on functionality; for example, if I don't have access to P0911, can I still get into the General Journal Review?

To get the most accurate results from your audit, it's very important to break down the rules and objects and understand the functionality behind them.

COMMON MISTAKE # 4: LACK OF ATTENTION TO SECURITY DETAILS

When conducting an audit, some people only check for Action Security. As shown in the example below, the APCLERK has both Action Security (Type 1) and Application Security (Type 3), so it's better to check for a combination of Application and Action Security.

Work With User/Role Security

Personal Form: (No Personalization) Layout: (No Layout) Query: /

✓ Q X Form Tools

Records 1 - 5

<input type="checkbox"/> User / Role	Object Name	Object Description	Security Type	Description	View	Add	Change	Delete	OK/Select	Copy	Scroll to End	Run	Install
<input type="checkbox"/> APCLERK	P0411												
<input type="checkbox"/> APCLERK	P0411	A/P Standard Voucher Entry	1	Action Security		Y	Y	N	Y	Y	Y		
<input type="checkbox"/> APCLERK	P0411	A/P Standard Voucher Entry	3	Application Security								Y	Y

Why? Because if the user can't open and run the application, you don't want to waste time analyzing the Action Security for potential violations. So, the first consideration is:

Can this user run and install this application?

And if so:

Do they have the Action Security?

And you need to check for ALL ways they can change the record – ie Add, Change, Delete and Copy.

If you're not checking for Application Security where Run = Yes, you're also missing a lot of reports. You can't apply Action Security to UBEs, so you need to check for Application Security to find out if users can run them, and you may also wish to check Processing Options Security.

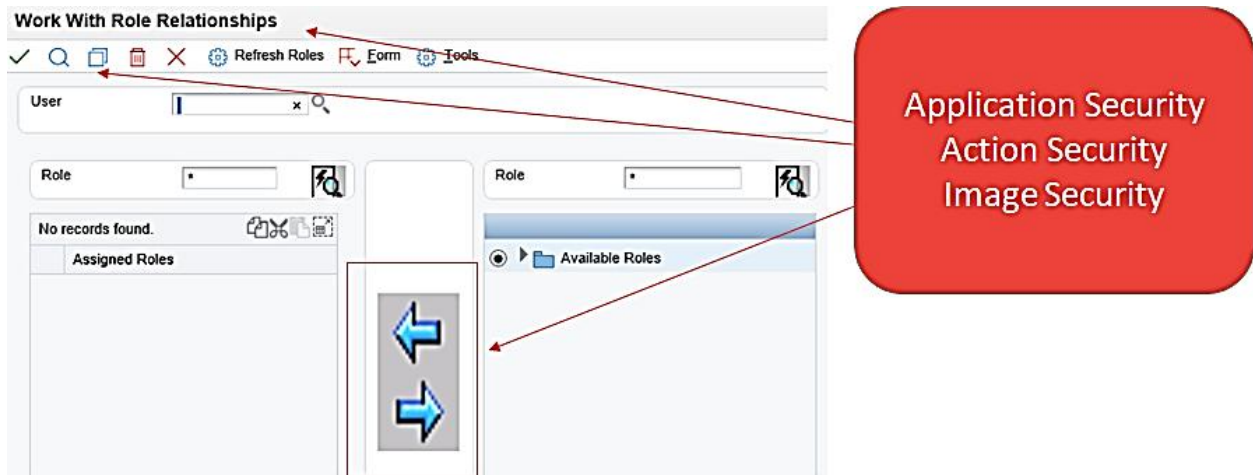
Understanding how Security Types work together will help you get more accurate audit results

New Security Types:

The later releases have introduced new security types, so it's also important to understand them and to know what to check for.

Example: Image Security

The screenshot below shows P95921, where Roles are assigned to users.



Here, people often only check for Application and Action Security. But in the later versions of JDE, we have the arrow images which are used to add or remove Roles from users, so you need to check for Image Security in combination with the other 2 types, i.e.

Application Security: Can I open this application?

Action Security: Can I Modify

Image Security: Can I click on the image to add and remove Roles?

RECOMMENDATIONS:

Do your homework!

When it comes to JD Edwards security, it's worth taking the time to understand the various security types and how they work together. Watch out for new security types and updated functionality in new releases.

Consult the Oracle Tools guide

This is a great source of information about all the different security types for the specific JDE versions.

Understand the changes in security between versions

This is particularly important for Version 9.0 and above. Keep up to date on the new security types and how they work, as well as how your environments work.

Attend Oracle / Quest / Q Software webinars

These organizations run educational sessions to help you keep up to date on new security functionality.



COMMON MISTAKE # 5: NOT UNDERSTANDING FALSE POSITIVES

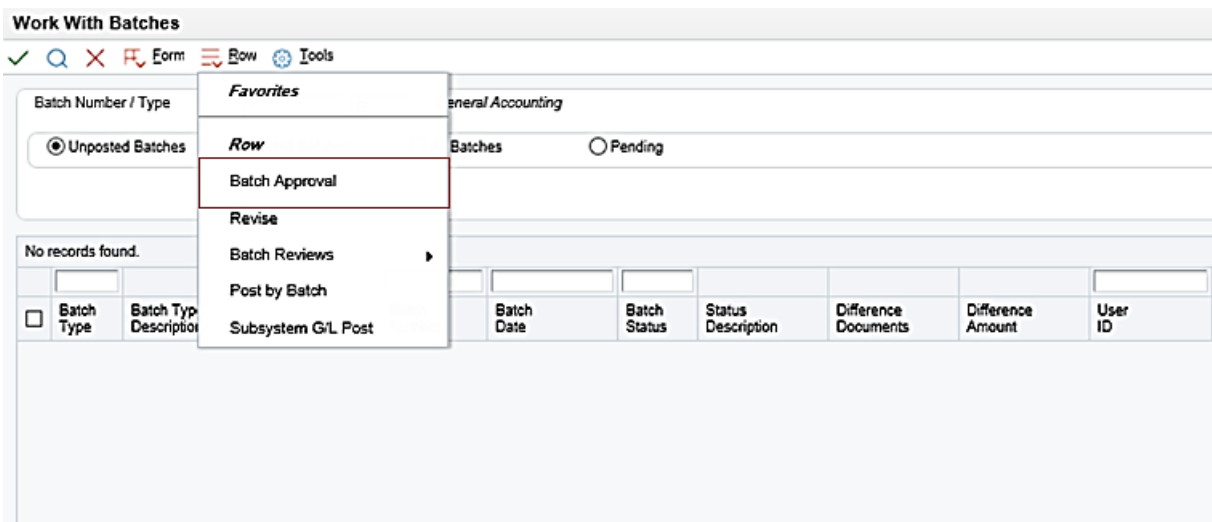
If you don't understand the functionality of applications, it's easy to waste time chasing false positives.

Here are 2 examples of this that I see frequently.

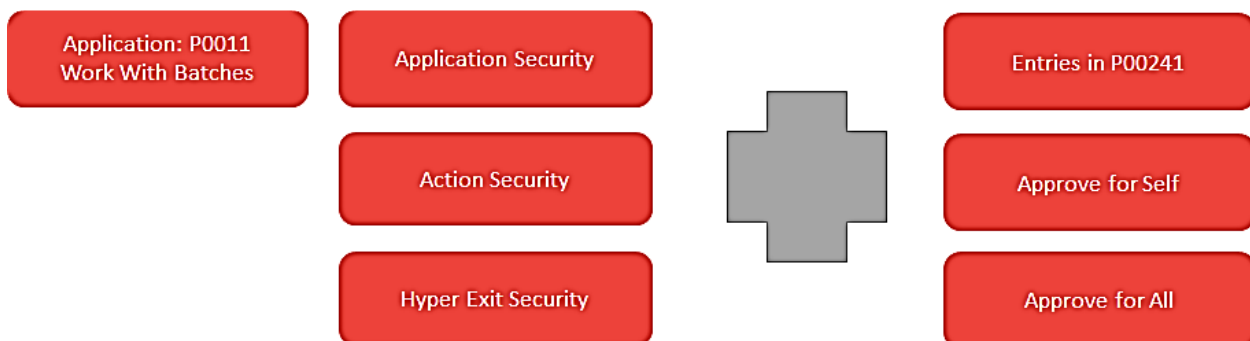
Example 1: Does a user really have access to approve batches?

With application P0011: Work with Batches, when trying to ascertain if a user can approve a batch, people most often look for Application Security and Action Security. You should also look for Hyper Exit Security, or where it may have been denied.

On this screenshot of P0011, you can see that there are no Add, Change or Delete buttons at the top. Batch approval is initiated via a Row Exit – hence the need to check for Hyper Exit Security:

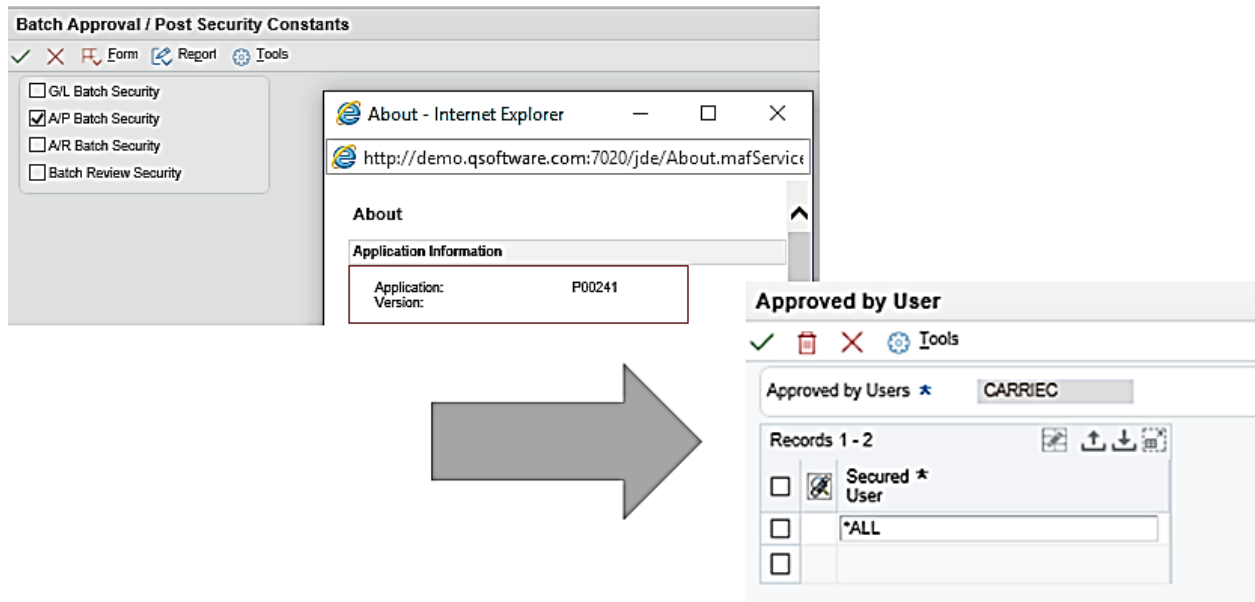


But these checks alone may not truly reveal whether a user has access to approve a batch.



To be able to approve a batch, may also be setup in P00241: Batch Approval/Post Security Constants, so you also need to check:

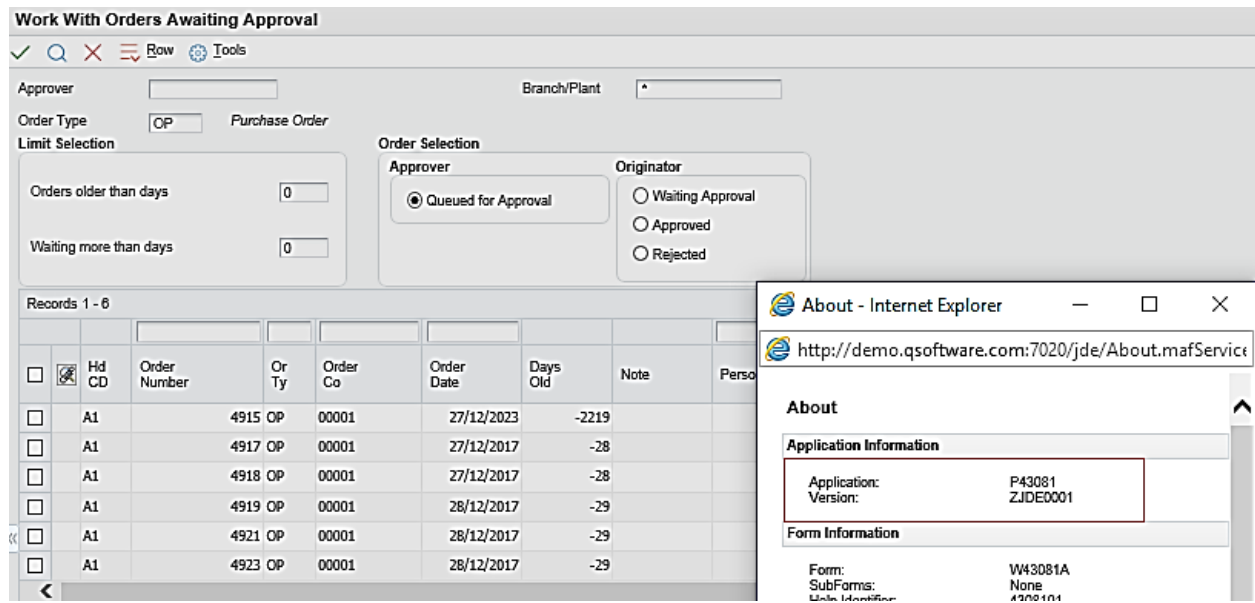
- is Batch Security is activated for the relevant batch type there and, if so
- is the user set up to approve:
 - for that batch type?
 - for self?
 - for all?



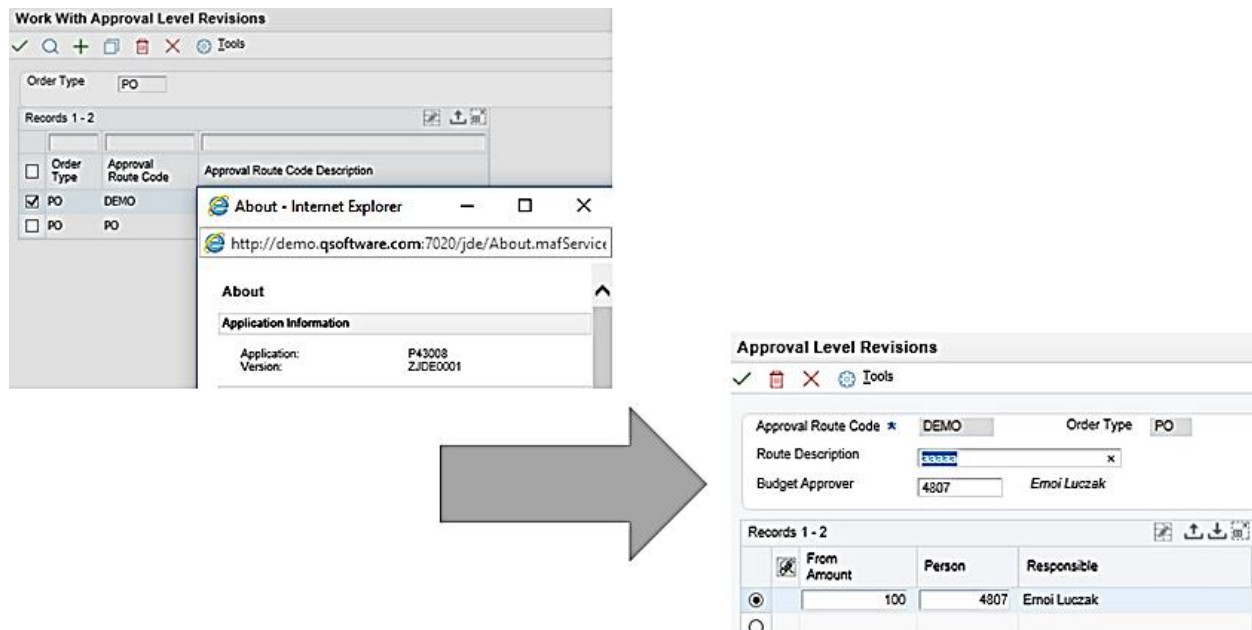
Example 2: Can a user approve Purchase Orders?

This is a similar scenario to the batches example above.

Again, the screenshot of P43081 shows that there are no Add, Change or Delete buttons at the top.



But you also need to check if you are using P43008: Work with Approval Level Revisions. A user needs to be set up in this application and its table to be able to approve purchase orders in P43081.



Application and Access Security to an object does not necessarily mean that a user can perform approvals in these applications.

RECOMMENDATIONS:

Involve users, super users and functional consultants

They will be able to help you understand the various configuration factors.

Perform a walk through with the user

This will enable you to identify the applications within each step.

Perform a negative test

Test to see whether you are able approve if you only have access to the application itself, or do you get an error message?

Account for the various security types within your SoD rule setup

Application and Action Security are two of the main ones, but you may also want to go down to the Form level, Hyper Exit Security level and look at the Table Data or Row Security.

To eliminate false positives, you really need to be able to look at combinations such as these outlined above. This may seem daunting, but if you're using a security tool to do your auditing, it becomes much easier.



COMMON MISTAKE # 6: NOT CONSIDERING THE BIGGER PICTURE

ENVIRONMENTS

Once you get the details worked out, you need to go back up to a higher level and look at your environments.

Not all Roles have access to all environments

If you see a Role in the Security Workbench, it doesn't necessarily mean that it has access to Production. You need to check which environments are assigned to that Role.

Even though Roles have access only in a Pre-Production environment, they can still effect Production

As discussed above, some objects update tables that are in the System Data Source, so they can affect all environments, so you need to bear that in mind when you look at Production versus Pre-Production access

Create a role strategy that is easy to understand and makes sense for the business

When you're reviewing your reports, it should be very clear what access people in Production should have to Pre-Production, and vice-versa. Good Role Strategy and Environment Strategy is very important to keeping your audit results clean and clear so that people can understand them.

ROLE CHOOSER

Do you allow users to choose the Roles they sign in with? When you're reviewing your reports, you need to take into consideration whether people can choose their Roles or not, and how this affects your SoD violations.

Oracle has recently recommended that UXOne Roles shouldn't be included in *ALL, which means that you need to grant standard security to a UXOne Role, so you need to bear that in mind when reviewing your SoD.

ROLE SEQUENCING

Understand how sequencing can affect access

To avoid wasting time on complicated analysis that you don't need to do, it helps to have a good understanding of how sequencing can affect access. When you design your Roles, this will help you to sequence them so that they make sense together.

Take sequencing into account, especially if Role Chooser is enabled

When you're looking at your violations, you need to understand how sequencing works and be able to identify whether people can effect change or violate an SoD rule if they sign in with an individual role, with *ALL, or both.

Often people put a lot of effort into analyzing their SoD violations without understanding the effects that Role Sequencing and Role Chooser have on users' ability to access objects. Looking at the bigger picture can save you a lot of work!



AUDIT YOUR AUDIT RULE SET!

Things change all the time, so it's important to review and adapt your audit rules to reflect changes to business processes, job responsibilities, custom program developments, and upgrades.

It may help to allocate responsibility for the various audit rules to specific team members, and have them review their areas once or twice a year to identify whether changes are needed.

IF YOU NEED HELP

I hope that this information has given you some insights to help you navigate your way through the complexities of auditing JD Edwards,

Most of us have found that it can be a very challenging task (to put it mildly!), but specialized tools can relieve much of the pain and improve the reliability of the results. If you would like to find out more about our auditing solutions for JD Edwards EnterpriseOne, please [click here](#)



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, Oracle E-Business Suite and Oracle ERP Cloud. 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 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

