July/August 2007 Volume VIII Issue 4

JDEtips Journal

JDEtips On Technical / Security

Changes in the JD Edwards® Security Model to Support Sarbanes Oxley

By Bill Loban

Editor's Note: *Who's been making changes to your database? If your company uses a proxy user; i.e., "sign-on security", your audit trail will show one user only for all changes made. That's not exactly what the auditors are looking for when they ask to see which individual users made which changes. So how do you get around this issue? Bill Loban provides the steps you'll need to make to the JD Edwards security model in order to reflect individual users (and keep those auditors happy!)*

Introduction

Since the Sarbanes-Oxley Act of 2002 became federal law, the need for publicly traded companies to keep track of who makes changes to their databases has become increasingly important. The JD Edwards security model employed by many companies uses a proxy user, also known as "sign on security." In this model, each user has a unique logon to the application, but access to the database is through the proxy user (JDE or PSFT). This creates a problem for audits because when the auditors want to see which users are making changes to the database, they see only one user, PSFT or JDE, depending on the release of the software. While this security model allows for less database administration, as only the PSFT / JDE logon needs to be maintained in the database, it does not meet the needs of SOx.

This paper will outline the steps needed to change the security model so that individual user IDs are reflected when tracking changes made to the database:

1. Unified Logon

Unified logon is a setup in JD Edwards that allows users to pass the ID with which they log into the domain into the JD Edwards application. This ID is stored in the Active Directory associated with each domain.

2. Active Directory

The users in the Active Directory are set up to match the user IDs established in the JD Edwards application.

3. SQL Server

The individual user IDs do not reside in the SQL Server database, as all users are accessing the database as the JDE / PSFT user. Individual users are created in the database using the same IDs used when logging into the domain. This allows the database to be accessed using the domain credentials.

4. Adding the Unified Logon Service

To support the Unified Logon, an additional service needs to be created for the JD Edwards application to use. This service is created by accessing the UniLogonSetup.exe, located in the x:\PeopleSoft\ddp\release\system\bin32 directory on the Enterprise server (see Figure 1).

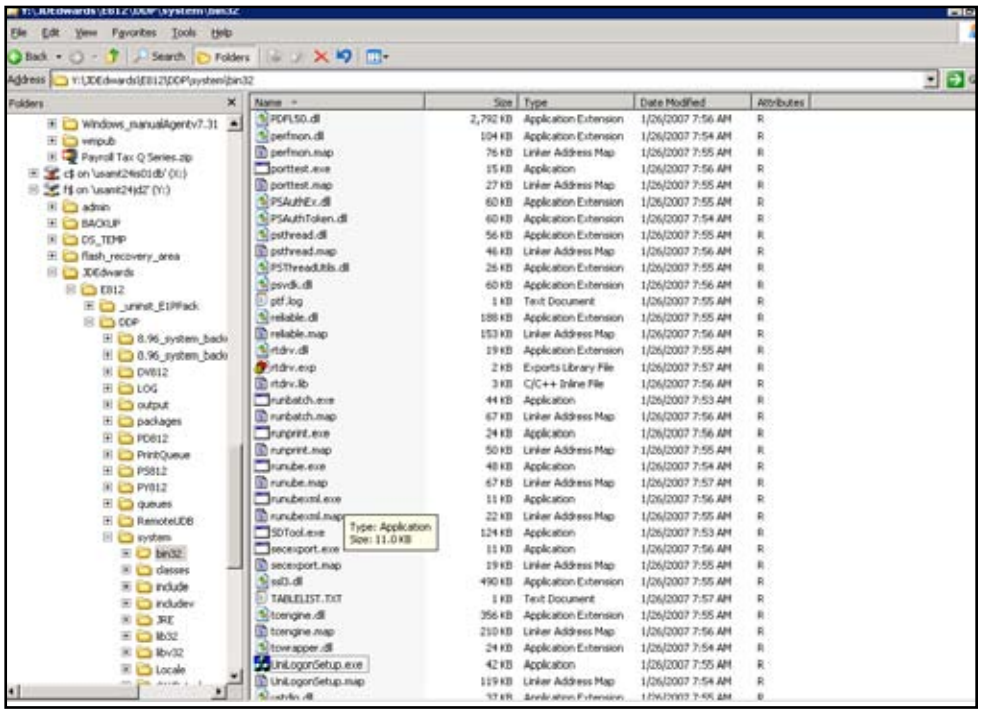

Figure 1 – Accessing the UniLogonSetup.exe

By invoking the UniLoginSetup application (see Figure 1), you'll bring up a window that allows you to enter the specifics of the service (see Figure 2).

To add Users to the Unified Login service, click the Add User button and enter the user name as it is set up in the Active Directory (see Figure 3).

Click the Install Service button to add the service.

5. Enterprise Server JDE.INI changes
In the [Security] section of the jde.ini on the Enterprise server, add the following line to turn on Unified Logon:

```
SecurityMode=1 (1 = Unified Logon Server)
```

Since the security server holds the proxy user and password, the security server needs to be removed from the SecuityServer entry. Therefore, in the [Security] section, remove the server name from the SecurityServer= entry.

Next, verify the SecurityServer kernel has 1 auto start process.

By setting the Security kernel to an auto start process, the kernel will be started when users first log on, which ultimately speeds up the login process as the application will not have to wait for the kernel to start. The code to do this is as follows:

```
[JDENET_KERNEL_DEF4]
krnlName=SECURITY KERNEL
dispatchDLLName=jdekrnl.dll
dispatchDLLFunction=_JDEK_DispatchSecurity@28
maxNumberOfProcesses=2
numberOfAutoStartProcesses=1
```

6. Client JDE.INI changes
In the [Security] section of the jde.ini on the client workstation, add the following lines:

```
UnifiedLogon=1 (1 = on)
ShowUnifiedLogon=0 (0=no check box for environment selection)
UnifiedLogonServer=EnterpriseServer
```

7. ODBC Changes
Presently, the ODBCs are set up to use SQL Server authentication. This must be changed because the individual users will now be authenticated using their Windows logon (see Figure 4).

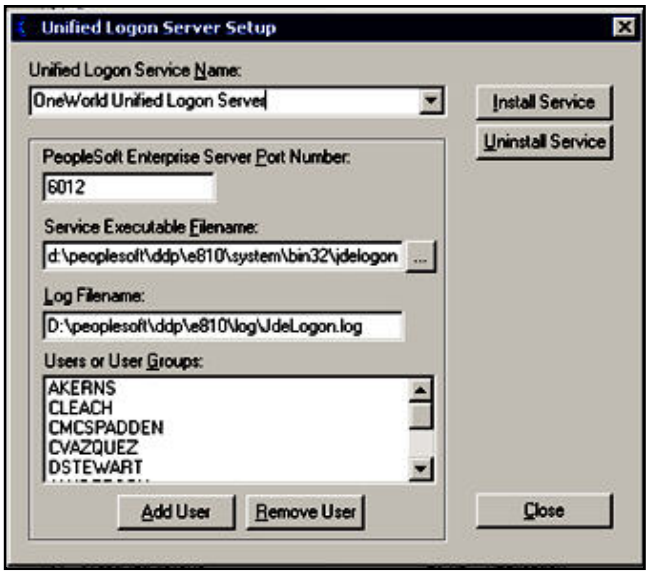

Figure 2 – Unified Logon Server Setup

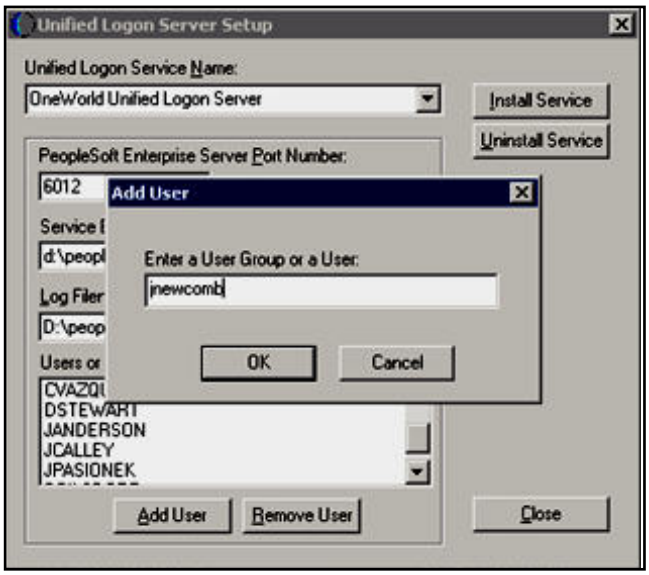

Figure 3 – Adding Users to Unified Logon