SedonaOffice - 6.2 PCI Compliance Update

01/04/2024 10:51 am EST

> The process of upgrading SedonaOffice from 5.7 to 6.1 has historically been very complicated and time-consuming. It is complicated because it is a multi-step process that must be performed in a specific order, to assure success. It is time-consuming because if it is not done in the proper order, or there are certain types of errors in the database, it can result in data errors that require SedonaOffice data support to make the necessary corrections.

The complex nature of this process is due to the fact that the method of processing EFT payments changed significantly from 5.7 to 6.1. This requires that a number of changes be made to the database before it is ready for use by the new EFT payment processes.

The PCIComplianceUpdate program was created to make the necessary changes to the database so that it could be transitioned from version 5.7, which uses a legacy Forte procedure, to version 6.1, which uses the Forte RestAPI for handling EFT payments.

The changes made for these DevOps work items are an attempt to begin to simplify this process so that it can be performed by the customer with minimal involvement by SedonaOffice support personnel.

1. Previous Upgrade Procedure

The previous procedure for upgrading from 5.7 to 6.1 required that the following steps be performed, in order.

1. Run PCIComplianceUpdate (version 5.7) to tokenize the credit cards. (This needs to be done with the legacy Forte credentials.)
2. Request RestAPI credentials from Forte and enter them into EFT Setup.
3. Upgrade SedonaOffice to version 6.1.
4. Run PCIComplianceUpdate (version 6.1) to tokenize the banks.

2. New Upgrade Procedure

The new upgrade procedure requires the following steps.

1. Upgrade SedonaOffice to version 6.1.
2. Run PCIComplianceUpdate to tokenize the credit cards. (This needs to be done with the legacy Forte credentials.)

3. Request RestAPI credentials from Forte and enter them into EFT Setup.

4. Run PCIComplianceUpdate again to tokenize the banks.

Although this procedure is very similar to the previous procedure, there are two distinct differences. First, the upgrade needs to be done first, so that the latest version of PCIComplianceUpdate is installed. This version of PCIComplianceUpdate will be used to do all tokenizations. Second, a number of database checks have been added to PCIComplianceUpdate, that will check for and whenever possible, correct, any data issues that could potentially cause errors during the tokenization process.

These checks are described in the following sections.

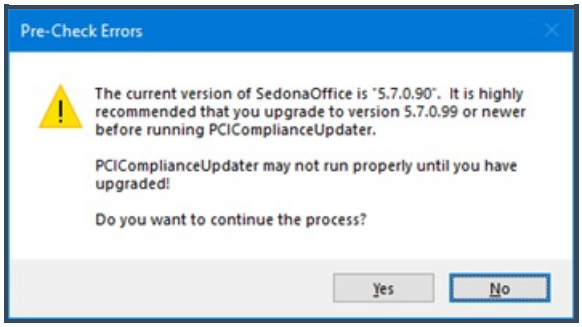
2.1 When Tokenizing Credit Cards

During the first execution of PCIComplianceUpdate, the following data checks are performed, before any credit cards are tokenized. This is to verify that there are no data errors that could result in tokenization failures, or loss of data.

After these checks are performed, the credit card tokenization is done in the same manner as it was in the earlier version of PCIComplianceUpdate.

2.1.1 Check SedonaOffice Version

The SedonaOffice version is checked to make sure that it is at least 5.7.0.99. Past experience has shown that earlier versions of the SedonaOffice database could be missing necessary data that might result in a failure to tokenize credit cards and banks. Therefore, this check is performed to warn the user of this situation. The following message is displayed if an earlier version of SedonaOffice is detected.



Of course, if the upgrade to 6.1 was performed before PCIComplianceUpdate was run, this check should not be necessary; it is only done as a precaution.

2.1.2 Create Backups

Because this process may make changes to some database tables, backups are made of these tables so that if a problem arises during this process, the database can be restored to its initial state. The tables that are backed up are the following:

- AR_ACH
- AR_ACH_Direct

- AR_Bank
- AR_Customer_CC
- AR_Customer_Bank

The names of these backup table names follow this convention:

- <table>_PRE6_<year_month_day> (e.g. AR_ACH_PRE6_2021_06_30)

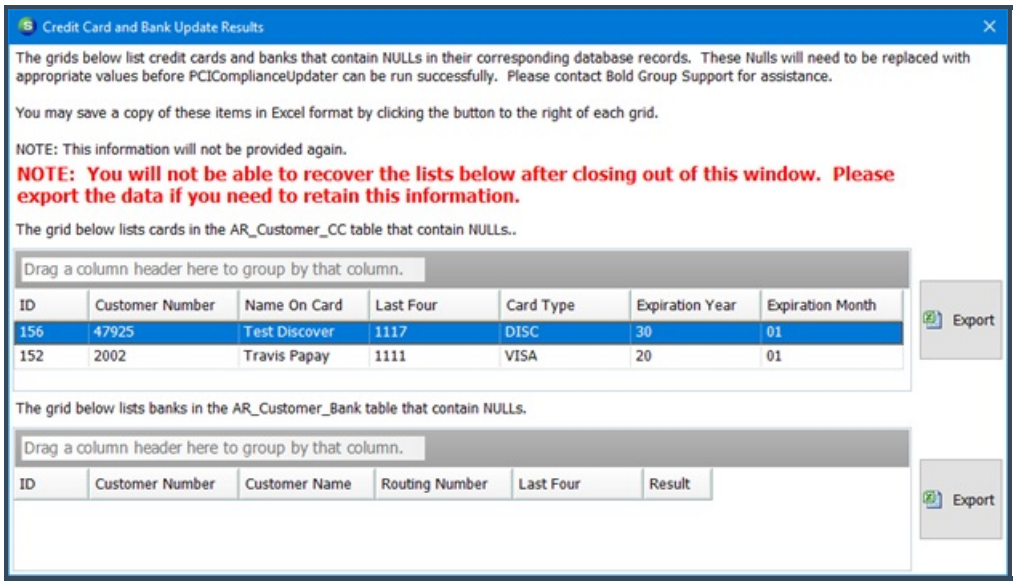
2.1.3 Replace NULLs

In earlier versions of the database, the Merchant_Id field in the AR_ACH and AR_Customer_CC tables may be NULL. These are replaced with the Merchant_Id of the branch to which each of the entries is associated.

2.1.4 Check for NULLs in Credit Card and Bank tables

There may be NULLs in some fields in the AR_Customer_CC and AR_Customer_Bank tables. If any nulls are found, the following form is displayed reporting this. These will need to be corrected before tokenizations can be performed.

PCIComplianceUpdate will not proceed beyond this step, until all nulls have been replaced with valid values.



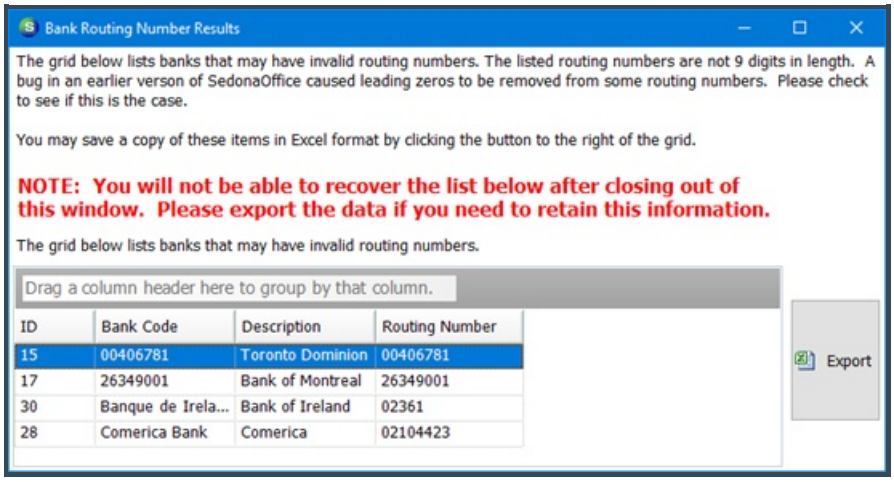
2.1.5 Check for Duplicate Merchant Ids

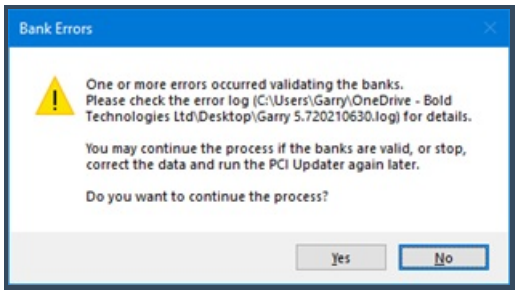
Earlier versions of SedonaOffice allowed duplicate merchant Ids to be entered into EFT Setup. A check is performed for this situation and the following message is displayed if it is found.

PCIComplianceUpdate will not proceed beyond this step until all duplicate merchant Ids have been removed.

2.1.6 Check for Invalid Bank Routing Numbers

Earlier versions of SedonaOffice had a bug that stripped leading zeros from bank routing numbers, resulting in routing numbers that contained few than nine digits. This check looks for routing numbers that are not nine digits in length and if any are found the following form is displayed. It is the responsibility of the user to verify whether or not these routing numbers are valid and correct them if necessary. The user is given the option to proceed if the routing numbers are determined to be valid.





2.1.7 Delete any Duplicate Banks

This check looks for duplicate banks (i.e., multiple banks with the same routing number) in the AR_Bank table. If any are found, the duplicates are removed from the AR_Bank table, and any references to the duplicates in the AR_Customer_Bank and AR_ACH tables are replaced.

2.1.8 Update AR_Bank Code

This step updates the AR_Bank table to set the Code field to match the Routing_Number field.

2.1.9 Update Z-Transactions

The final check, before tokenizing the credit cards, is to make sure that all AR_ACH entries for Z-transactions follow the convention expected by the 6.1 version of EFT payment processing. If any Z-transactions are found that do not match the 6.1 conventions, those entries are updated appropriately.

2.1.10 Tokenize Credit Cards

At this point, any credit cards that do not have tokens are tokenized. This is done using the legacy Forte credentials, in the same manner that it was done in the previous version of PCIComplianceUpdate.

2.2 Update Forte Credentials

2.3 When Tokenizing Banks

Banks are tokenized when running PCIComplianceUpdate the second time, after updating the EFT credentials to the Forte RestAPI credentials. This procedure is almost identical to the procedure that was followed in the earlier version of PCIComplianceUpdate; most of the data issues that could affect bank tokenizations should have already been addressed in the credit card tokenization phase. However, there are a couple of additional steps that have been added. These are described in the following sections.

2.3.1 Create Backups

As is the case during the credit card tokenization process, before the banks are tokenized, backups are made of tables that may be modified during the tokenization process. The tables that are backed up are the following:

- AR_ACH
- AR_ACH_Direct
- AR_Bank
- AR_Customer_CC
- AR_Customer_Bank

The names of these backup table names follow this convention:

- <table>_POST6_<year_month_day> (e.g. AR_ACH_ POST6_2021_06_30)

2.3.2 Tokenize Banks

2.3.3 Update Held Transactions

After the banks have been successfully tokenized, a check is performed to look for any EFT payments that have a hold date that is later than the date on which the upgrade is performed. If any are found, the associated AR_ACH records are updated so that the payments will be submitted at the appropriate time by the 6.1 version of the EFT payment process.