

Development - SedonaOffice Product Troubleshooting

12/29/2023 4:33 pm EST

How to get EFT Payment Processing log data

From database tables:

There are two database tables that can contain useful information for diagnosing EFT Processing issues: SEFT_Service_Log and SY_Edit_Log.

SEFT_Service_Log:

This table is the primary logging table used by EFT Processing. The information contained in this table can be viewed from the front-end by opening the EFT Processing form and selecting the menu Tools > View Service Log. However, querying the database table directly is a much more useful way of getting the data you need, since you can filter for the specific log entries you are interested in. You can filter for a date range by using a query similar to the following:

```
SELECT * FROM SEFT_Service_Log WHERE CreatedOn > '2022-03-01' and CreatedOn < '2022-03-15' ORDER BY CreatedOn
```

You can also filter the query for errors by adding a where clause (WHERE IsError = 1), but this will filter out the context leading up to the error.

SY_Edit_Log:

This table is used by EFT Processing to log some events that are initiated manually from the front-end. It is less useful than the SEFT_Service_Log table, but it can provide further context to the events logged in the SEFT_Service_Log table.

From the Logger.txt debug file:

Another useful diagnostic tool is the Logger.txt file. If this file exists on the computer's desktop, EFT Processing will write debugging information to it. Some of the data logged to this file is the same as that logged in the SEFT_Service_Log table, but it also contains much more data that can be invaluable for diagnosing problems. The downside to using this file is that, if it does not already exist before the problem occurs, then no data will be logged, so the problem will need to be reproduced after this file is created.

On the client computer:

To use this file, you will need to login to the computer using the login credentials of the person that will be running SedonaOffice. Then on that person's desktop, create the file "Logger.txt". When SedonaOffice is run and the EFT Processing form is opened, debugging information will be written to this file.

On the server computer:

If the problem you are trying to diagnose is related to the automatic processing of EFT transactions, i.e., those that are performed by the SedonaEFTService program, then you will have to perform the following steps to use the Logger.txt file.

1. Login to the server (you may need administrative privileges to do some of these steps).
2. Create the Logger.txt file on the desktop.
3. Open Task Manager and do the following:
 - a. Stop SedonaEFTService service.
 - b. Open services.
 - c. Double click SedonaEFTService.
 - d. Open the "Log On" tab, select "This account" and enter the login credentials used above.
 - e. Restart SedonaEFTService.
4. Make sure to restore SedonaEFTService to its original state after you have finished with the Logger.txt file.

NOTE: It is recommended that you delete the Logger.txt file after you have gotten all the information from it that you need, since SedonaOffice will continue to write to it as long as it exists.

Tracing VB6 code with Milestone_Tracking table

The SedonaOffice company database contains a stored procedure, Milestone, that can be used to add tracking information to the Milestone_Tracking table. (This stored procedure will create the table if it does not already exist.) This is useful for tracing through the code to determine when an error occurs, especially if you are unable to use the debugger to trap the error, for example, if the error only occurs on a customer's system.

The SedonaOffice VB6 project contains the Milestone_Tracking class that executes the Milestone_Tracking stored procedure to add a tracking record. Below is a block of example code that demonstrates how to initialize the class and add a tracking record.

```
' Create the class and establish a connection to the database

Dim clsMilestone As New Milestone_Tracking

Set clsMilestone.ActiveConnection = asConnection(mCompany)

' Write a tracking record to the database

clsMilestone.Track 1, 1, "Tracking information", "Tracking comment"
```