SQL Audit Log Implementation (v2023.1+)

The [SQL Audit Log](https://medsphere.sharepoint.com/:w:/r/STL/Development/_layouts/15/Doc.aspx?sourcedoc=%7B369EC1DE-F986-458E-A656-461EA09E5B30%7D&file=CAS-127537%20Audit%20Logging%20%26%20Reporting.docx&action=default&mobileredirect=true&CID=C4BDA255-4343-4BA7-B130-3BE6040AFB65&wdLOR=c39CD4D5C-5479-44EA-867F-487A93328415), which supersedes the previous ONCHIT Audit Logging functionaltiy, is a fairly simple mechanism for logging and reading user activity within the product (RCM Cloud & Classic).

The overall design/architecture is comprised of:

* SQL Server/Database (2017+ minimum)
* SQL Audit Objects (added by specialty script)
  + Server Level Audit Object
  + Database Level Audit Object
* SQL Job (added by release script)
* SSRS Report (added by reports installer)

Each of the above items will be described in detail within the course of the document and will include a brief implementation section.

Contents

[Implementation 3](#_Toc132627371)

[SQL Server Instance 3](#_Toc132627372)

[SMI (SQL Managed Instance) 3](#_Toc132627373)

[Logged Events 4](#_Toc132627374)

[SQL Audit Objects 6](#_Toc132627375)

[SQL Job 9](#_Toc132627376)

[SQL Audit Log Report 10](#_Toc132627377)

[Technical Information 12](#_Toc132627378)

# Implementation

Implementation comes in two flavors for the SQL Audit Log:

1. SQL Server Instance (full version installed on a server)
2. SMI (SQL Managed Instance in the cloud)

The SMI implementation has yet to be implemented.

## SQL Server Instance

To implement Audit Logging on SQL Server, follow the steps below:

* Execute the **InsightCS 2022.1.sql** script if not done. This will install the job, tables, registry key, stored procedures, data and configuration necessary for the Audit Log to function (and copy necessary components over to the warehouse automatically, assuming a warehouse exists and is up and running).
* Update and execute the **2022.1 - CAS-127537 - SQL Server Audit Log Specifications.sql** file. You must follow the instructions in the file prior to execution.
  + **Required**: Once installed, you must e*nable both audit objects in SSMS (right-click, enable)*.
  + **Required**: Create the physical path for the files that was specified in the installation script (from the server’s point of view and that the server has access to).
  + **Optional**: You may adjust the Queue Delay and Maximum File Size for the parent SQL Audit Specification if the defaults aren’t satisfactory.
* Update the following:
  + Update the **job** so that the single step is pointing to the proper database (ideally, the warehouse. Only use production if absolutely necessary).
  + Update the **GLOBAL\_REGISTRY** key **SQLAUDIT\_FOLDER:**
    - VALUE1 – Full path where the SQL audit files will be stored. *This must match the same path entered for FILEPATH in the 2022.1 - CAS-127537 - SQL Server Audit Log Specifications.sql file*.
    - VALUE2 – Full path where the *processed* audit files will be stored. This can be a “\Processed” folder directly within the VALUE1 folder.
    - Note that UNC paths work fine.
* Run the reports installer to install the Patient Audit Log report.

## SMI (SQL Managed Instance)

To be created.

# Logged Events

Overall, the process captures, and reports on, *application* & *custom* events as defined in the tables below.

**RCM Cloud**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **ITEM** | **LOGON** | **LOGOFF** | **VIEW** | **INSERT** | **UPDATE** | **DELETE** | **NAV** | **PRINT** |
| Shell | X |  |  |  |  |  |  |  |
| Scheduling Wrkspc |  |  |  | X | X |  | X |  |
| Reg Wrkspc |  |  | X | X | X |  | X |  |
| Medrec Wrkspc |  |  | X |  | X |  | X |  |
| Billing Wrkspc |  |  | X |  | X |  | X |  |
| Collections Wrkspc |  |  | X |  | X |  | X |  |
| Pymt Psting Wrkspc |  |  | X |  | X |  | X |  |
| Bad Debt Wrkspc |  |  | X |  | X |  | X |  |
| ABT QA |  |  |  |  | X |  | X |  |
| Adj Entry QA |  |  |  |  | X |  | X |  |
| Bill Annul QA |  |  |  |  | X |  | X |  |
| Charge Entry QA |  |  |  |  | X |  | X |  |
| Copay Entry QA |  |  |  |  | X |  | X |  |
| Discharge QA |  |  |  |  | X |  | X |  |
| Discharge Xcel QA |  |  |  |  | X |  | X |  |
| External Coder QA |  |  |  |  | X |  | X |  |
| Imaging QA |  |  |  |  | X |  | X | X |
| Mark Bad Debt QA |  |  |  |  | X |  | X |  |
| Mark Refund QA |  |  |  |  | X |  | X |  |
| Merge |  |  |  |  | X | X | X |  |
| Notes Entry QA |  |  |  |  | X |  | X |  |
| OP Bed Ass QA |  |  |  |  | X |  | X |  |
| Pymt Entry QA |  |  |  |  | X |  | X |  |
| Room/Bed Xfer QA |  |  |  |  | X |  | X |  |
| Transfer QA |  |  |  |  | X |  | X |  |
| Refund Rev QA |  |  |  |  | X |  | X |  |
| Visit Cancel QA |  |  |  |  |  | X | X |  |
| Visit Combine |  |  |  | X | X | X | X |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |

**Insight** (Classic)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **APP** | **LOGON** | **LOGOFF** | **VIEW** | **INSERT** | **UPDATE** | **DELETE** | **NAV** | **PRINT** |
| Shell | X | X |  |  |  |  | X |  |
| Abstracting |  |  | X |  | X |  |  |  |
| Acct Bal Xfer |  |  |  |  | X |  |  |  |
| Adj Entry |  |  |  |  | X |  |  |  |
| Adm Cancel |  |  |  |  |  | X |  |  |
| Adm Discharge |  |  |  |  | X |  |  |  |
| Adm Disch Xcel |  |  |  |  | X |  |  |  |
| Adm Transfer |  |  |  |  | X |  |  |  |
| Batch Pymt Entry |  |  |  |  | X |  |  |  |
| Bill Annulment |  |  |  |  | X |  |  |  |
| Charge Entry |  |  |  |  | X |  |  |  |
| Combine Visit |  |  |  |  | X | X |  |  |
| Imaging |  |  |  |  |  |  |  | X |
| Ins Verification |  |  | X |  | X |  |  |  |
| Mr Merge |  |  |  |  | X | X |  |  |
| Nb Registration |  |  |  | X |  |  |  |  |
| Payment Entry |  |  |  |  | X |  |  |  |
| ~~Patient Inquiry~~ |  |  | ~~X~~ |  |  |  |  |  |
| Quick Reg |  |  |  | X |  |  |  |  |
| Reassgn Chrgs |  |  |  |  | X |  |  |  |
| Reassgn Pymts |  |  |  |  | X |  |  |  |
| Registration |  |  | X | X | X |  |  |  |
| Remit. Adv Entry |  |  |  |  | X |  |  |  |
| Utilization Rvw |  |  |  |  | X |  |  |  |
| Visit Un/Combine |  |  |  | X | X |  |  |  |
| Visit Detail |  |  | X |  | X |  |  |  |
| Visit Merge |  |  |  |  | X | X |  |  |
|  |  |  |  |  |  |  |  |  |

# SQL Audit Objects

To capture all the data to a log file, [SQL Server Audit](https://learn.microsoft.com/en-us/sql/relational-databases/security/auditing/sql-server-audit-database-engine?view=sql-server-ver15) is utilized. Both a *server* and *database* level Audit Specification are necessary to log information.

These specification objects are installed as part of a specialty script (discussed in the Implementation section).

**Server Audit** **Specification**

The Server Audit is the parent component of a SQL Server audit and resides in the master database. It’s used to define the queue delay, how SQL Server should react in case auditing fails, where the audit information will be stored, and the file roll over policy. A filter may be applied to the overall audit as well.

Graphical user interface, text, application

Description automatically generated

For the purposes of our audit, the queue delay should be minimal, but set in consideration of the current load that the server will endure. For light-duty servers, a setting of 5000 (5 seconds) would suffice. For heavy-duty servers, a 10,000 – 15,000 (10-15 seconds) setting might yield better results and impact the server less. The risk with a longer delay comes if the server is unexpectedly shuts down; anything that might be in the buffer since the last write operation will be lost.

The Audit destination should be set to File for on-prem or full SQL server based recording, or set to URL for SQL Azure. Both require an appropriate path. When selecting a file based storage, it’s important to also select a maximum file size (10 mb). This will allow the SQL job that monitors the files to move and import the files to the audit table once capped at their maximum size.

Note that for the RCM Cloud/Insight Audit, we are only capturing updates created by the core applications (RCM Cloud & Insight). Because of this, it is required to have a filter on the application2 name as well as only the events and statements we’re interested in. Without the filter, the audit will capture well more data than is useful:

Graphical user interface, text

Description automatically generated

**Database Audit Specification**

The Database Audit is the child component of a SQL Server audit and resides in the database for which events are to be captured. It’s used to define which actions/events are of interest for capture (within the context of the filter specified on the parent).

Graphical user interface, application

Description automatically generated

Only two actions are defined

* BATCH\_COMPLETED\_GROUP – Which relates specifically to SQL statements, such as INSERT, UPDATE or DELETE actions
* USER\_DEFINED\_AUDIT\_GROUP – Which relates specifically to custom messages, such as Navigation and Log On/Off actions

Together (again, in the context of the filter), these two actions capture everything needed to produce information for the audit log report.

# SQL Job

A SQL job (SQLAUDIT\_LOG\_IMPORT\_V2022\_Q1\_V1) is utilized for periodically importing, moving and deleting the log files produced by the SQL Audit Specifications described earlier.

The job is scheduled to run every 5 minutes, however, it will only import files that have been completed and closed by the SQL Audit Log. This means that a file may be created by SQL Audit but not yet imported until the file reaches its maximum size (rollover size noted in the SQL Audit configuration) and a new file created.

The job reads the files in the directory setup in the GLOBAL\_REGISTRY SQLAUDIT\_FOLDER key, VALUE1 field. Once completed, each file is imported and moved to the “processed” folder configured in the SQLAUDIT\_FOLDER key, VALUE2 field.

Files that are eventually moved to the “processed” folder are also deleted by the job so no maintenance is required.

# SQL Audit Log Report

The SQL Audit Log report displays information regarding captured SQL Audit Log data which pertains to user activity.

Graphical user interface, table

Description automatically generated

The report contains 6 filters including:

* From & To Dates (inclusive)
* Event – Multi-select for the different types of events (defaults to all)
* User Name – For paring down to individual users
* URN – For selecting a specific patient
* Visit No – for selecting a specific visit

Note that using the URN or Visit No parameter will display all records captured from the product for the specific events listed though this is *not* a full picture of *everything* that has happened in the system to a particular patient or visit, for instance:

Only common RCM Cloud and Classic updates, as noted in the tables within this doc, are captured. There are other functionalities within the product which may update accounts, for example (note that most of these items are not user-based updates):

* Interfaces
* Nightly Executables
* SQL Jobs
* Custom Processes
* SSMS queries
* Even some events within the RCM Cloud and Classic products aren’t captured.

The report displays the following information:

* Action Date
  + The action date is recorded in UTC time and will be displayed as such unless the LOCATION\_REGISTRY LOCAL\_TIME\_BIAS key is configured. Once configured, the time will be displayed according to the offset for the user’s base location. Note that DST is utilized as necessary.
* Action
* Application
* Visit No
* URN
* Message
* User Name
* SQL Statement
  + SQL statements will only display for specific actions as custom messages have no SQL statements.

# Technical Information

The SQL Audit specifications, when created, can be found in SSMS (SQL Server Management Studio). The SQL Server (parent) specification is found in the Server\Security\Audits section. Named RCM Cloud - Audit

Graphical user interface, text, application

Description automatically generated

The  database (child) specification can be found in the Server\Databases\Database\Security\Database Audit Specifications section. Named: RCM Cloud – DB Audit Specification

Graphical user interface, application

Description automatically generated

There exists a large volume of actions that SQL Audit can record, though we’re only interested in a subset. To see the full list of actions, the following SQL queries can be executed:

-- Action IDs for SQL Audit

select action\_id, name from sys.dm\_audit\_actions group by action\_id, name order by action\_id

-- Actions and Event Groups

select \*/\*, containing\_group\_name, action\_id, name\*/ from sys.dm\_audit\_actions order by containing\_group\_name, action\_id

To see data captured from the SQL Audit (stored in the \*.sqlaudit files), run the following query, updating the file path on the server as necessary. The file path can be retrieved from the SQL Audit by viewing the properties of the audit in SSMS (SQL Server Management Studio):

/\*

\*\* Select audit data from files directly (not table)

\*/

SELECT EVENT\_TIME

,SEQUENCE\_NUMBER

,ACTION\_ID

,USER\_DEFINED\_EVENT\_ID

,USER\_DEFINED\_INFORMATION

,STATEMENT

,APPLICATION\_NAME

,SERVER\_PRINCIPAL\_NAME

,SERVER\_INSTANCE\_NAME

,DATABASE\_NAME

,SCHEMA\_NAME

,OBJECT\_NAME

,HOST\_NAME

FROM sys.fn\_get\_audit\_file('E:\AuditTest\\*.sqlaudit', DEFAULT, DEFAULT)

ORDER BY EVENT\_TIME DESC

While the Action Id recorded and imported from the Audit is a string, it is originally a number. This is why the filter on the parent SQL Audit object filters the actions by numbers. To determine what action equates to what number, a stored procedure exists in the IcsDev database that helps convert the code to it’s numeric equivalent:

/\*

\*\* Convert ACTION\_ID string to integer for filter

\*/

ConvertSqlAuditActionId 'trbc'

The filter on the parent audit object cannot hold comments, so this may help in understanding what the filter is doing:

/\*

\*\* Full audit filter

\*/

([APPLICATION\_NAME] like '%RcmCloud%' OR [APPLICATION\_NAME] like '%InsightCS%')

and

(

([STATEMENT] LIKE '%UPDATE %' OR [STATEMENT] LIKE '%DELETE %' OR [STATEMENT] LIKE '%INSERT %' OR [STATEMENT] LIKE '%EXECUTE %')

or

([ACTION\_ID] = 541934402 /\*BCM\*/

AND

(

([STATEMENT] LIKE '%TRANCOUNT%' AND ([STATEMENT] LIKE '%COMMIT TRAN%' OR [STATEMENT] LIKE '%ROLLBACK TRAN%'))

OR

([STATEMENT] LIKE '%IMPLICIT\_TRANSACTIONS%')

)

)

or

( [ACTION\_ID] = 1128419924 /\*TRBC\*/ OR [ACTION\_ID] = 1129468500 /\*TRRC\*/ OR [ACTION\_ID] = 1128485460 /\*TRCC\*/  OR [ACTION\_ID] = 1430340693 /\*UDAU\*/)

)

**Supporting Tables**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **SQLAUDIT\_ACTION\_MSTR – Actions within the audit log (Exec, Rpc Strt, etc…)** | | | | |
| **KEY** | **Column Name** | **UDDT/**  **Data Type** | **Length** | **Nullable** |
| PK | ACTION\_CD | Varchar | 4 | 0 |
|  | ABBR\_TXT | Datetime2 | 15 | 0 |
|  | FULL\_TXT | Int | 60 | 0 |
|  | ENABLE\_FLG | Varchar | 4 | 1 |
|  |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **SQLAUDIT\_EVENT\_MSTR – Events for the audit log (Logon, View, Insert, etc…)** | | | | |
| **KEY** | **Column Name** | **UDDT/**  **Data Type** | **Length** | **Nullable** |
| PK | EVENT\_ID | Int | 4 | 0 |
|  | ABBR\_TXT | Datetime2 | 12 | 0 |
|  | FULL\_TXT | Int | 60 | 0 |
|  | ENABLE\_FLG | Varchar | 4 | 1 |
|  |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **SQLAUDIT\_HISTORY – Event information from the audit.** | | | | |
| **KEY** | **Column Name** | **UDDT/**  **Data Type** | **Length** | **Nullable** |
| PK | ROW\_NO | Bigint |  | 0 |
|  | EVENT\_TIME | Datetime2 |  | 0 |
|  | SEQUENCY\_NUMBER | Int |  | 0 |
|  | ACTION\_ID | Varchar | 4 | 0 |
|  | SERVER\_PRINCIPAL\_NAME | Sysname |  | 1 |
|  | SERVER\_INSTANCE\_NAME | Sysname |  | 0 |
|  | DATABASE\_NAME | Sysname |  | 1 |
|  | SCHEMA\_NAME | Sysname |  | 1 |
|  | OBJECT\_NAME | Sysname |  | 1 |
|  | STATEMENT | Nvarchar | 4000 | 1 |
|  | USER\_DEFINED\_EVENT\_ID | Smallint |  | 1 |
|  | USER\_DEFINED\_INFORMATION | Nvarchar | 4000 | 1 |
|  | APPLICATION\_NAME | Nvarchar | 128 | 1 |
|  | HOST\_NAME | Sysname |  |  |
|  |  |  |  |  |

Stored procedures include:

* SQLAUDIT\_LOG\_CUSTOM\_EVENT\_V2022\_Q1\_V1 that allows the logging of custom events for the SQL Server Audit.
* SQLAUDIT\_LOG\_IMPORT\_V2022\_Q1\_V1 which users powershell and SQL commands to:
* Move the audit files from a storage directory to a “processed” directory for importing.
* Imports the data to the SQL\_AUDIT\_HISTORY table.
* Deletes the files once the data is imported. This ensures the values are imported only once and then discarded.

A simple, SQL scheduled job calls the SQLAUDIT\_LOG\_IMPORT\_V2022\_Q1\_V1 procedure every so often (recommended 1-5 minutes) for import.

A GLOBAL\_REGISTRY key, SQLAUDIT\_FOLDER is utilized by the SQLAUDIT\_LOG\_IMPORT\_V2022\_Q1\_V1 stored procedure to determine where to find and process the files:

* VALUE1 – Indicates the existing path/directory where the Audit Log files are initially stored. This is the same path as the Destination set in the SQL Server Audit parent object.
* VALUE2 – Indicates the existing path/directory where the Audit Log files should be moved prior to processing.

Note that a special script is used to create the SQL Server Audit (parent audit) and the Database Audit (child audit) specifications. They are not created automatically from the release script as there is specific configuration that must be done.