



White Paper

Configuration of Audit Logs

Fabasoft Folio 2024 Update Rollup 1

Copyright © Fabasoft R&D GmbH, Linz, Austria, 2024.

All rights reserved. All hardware and software names used are registered trade names and/or registered trademarks of the respective manufacturers.

No rights to our software or our professional services, or results of our professional services, or other protected rights can be based on the handing over and presentation of these documents.

Contents

1 Introduction	4
2 Software Requirements	4
3 Audit Data Sources	4
4 Audit Log Configuration in the Object Class	6
5 Security Aspects of the Audit	7
5.1 Reading Audit logs	7
5.2 Data Source	7
6 Example	7
7 Environment Variables	8

1 Introduction

This document describes the configuration and use of audit logs. Using audit logs, it is possible to record access to properties, calls to actions or applications and the review of access rights.

2 Software Requirements

System environment: All information contained in this document implicitly assumes a Microsoft Windows environment or Linux environment.

Supported platforms: For detailed information on supported operating systems and software see the software product information on the Fabasoft distribution media.

3 Audit Data Sources

The audit information is stored using dedicated audit databases. It is written by the Fabasoft Folio Kernel, therefore the Fabasoft Folio Kernel has to be able to connect to the audit database via network.

The data sources to be used (*Service Data Source*, COOSYSTEM@1.1:ServiceDataSource) have to be entered in the *Audit Data Sources* (COOSYSTEM@1.1:domainauditdatasource) property of the *Current Domain*.

Examples of Data Sources:

- "Microsoft SQL Server" Data Source

New Service Data Source (Service Data Source): Edit

Service Data Source

Multilingual Name
SQL Server

Username
audituser

Password
••••••••

Parameters

<input type="checkbox"/>	Parameter Type *	Parameter Value *	Software Component	Show Details (3)
<input type="checkbox"/>	1 Provider	SQLOLEDB	Built-in Settings	
<input type="checkbox"/>	2 Data Source	192.168.100.143	Built-in Settings	
<input type="checkbox"/>	3 Catalog	auditdb	Built-in Settings	

Add Entry

Cancel Apply Next

- “Oracle Database” Data Source

New Service Data Source (Service Data Source): Edit

Service Data Source

Multilingual Name
Oracle Database

Username
audituser

Password
••••••

<input type="checkbox"/>	Parameter Type *	Parameter Value *	Software Component	Show Details (2)
<input type="checkbox"/>	1 Transport	OCI	Built-in Settings	⌵ ⌵
<input checked="" type="checkbox"/>	2 Location	orcl	Built-in Settings	⌵ ⌵

Add Entry

Cancel Apply Next

- “PostgreSQL” Data Source

PostgreSQL (Service Data Source): Edit

Service Data Source

Multilingual Name
PostgreSQL

Username
audituser

Password
•••••

<input type="checkbox"/>	Parameter Type *	Parameter Value *	Software Component	Show Details (4)
<input type="checkbox"/>	1 Transport	PGSI		⌵ ⌵
<input type="checkbox"/>	2 Catalog	auditdb		⌵ ⌵
<input type="checkbox"/>	3 Connect String	PORT=5432		⌵ ⌵
<input type="checkbox"/>	4 Location	192.168.100.83		⌵ ⌵

Add Entry

Cancel Apply Next

To use SSL/TLS encryption for the connection, the `SSLMODE` parameter (supported values are “disable” and “require”) can be added to the client connect string. For example, the connect string `“PORT=5432;SSLMODE=require”` uses port 5432 for the connection and requires the server to use an encrypted connection.

Note:

- To make the audit log highly available it is necessary to configure several data bases in the *Current Domain*. If writing audit information fails on one data source, an automatic failover to another data source is performed.
- If no audit log data source is configured, no audit log entries are written.

4 Audit Log Configuration in the Object Class

For each object class can be configured which access and calls to objects of this object class should be logged. The configuration is done on the "Advanced" tab of an object class in the aggregate list *Audit Log Configuration* (COOSYSTEM@1.1:classauditconf).

In the *Audit Log Configuration* the following settings can be made:

- **Context** (COOSYSTEM@1.1:auditactionattrdef)
Defines the context of the audit. Allowed object classes are:
 - *Action* (COOSYSTEM@1.1:Action)
 - *Property* (COOSYSTEM@1.1:AttributeDefinition)
For example: objcreatedat
 - *Access Type* (COOSYSTEM@1.1:AccessType)
 - *Application* (FSCVAPP@1.1001:Application)
- **Audit Type** (COOSYSTEM@1.1:audittype)
 - "Read Property"
Allowed context: Property.
 - Change Property"
Allowed context: Property.
 - "Change Property (Values Saved)"
Allowed context: Property.
 - "Change Property (Version Saved)"
Allowed context: Property.
 - "Call Action"
Allowed context: Action, Application.
 - "Call Action (Version saved)"
Allowed context: Action, Application.
 - "Call Action With Success"
Allowed context: Action, Application.
 - "Call Action With Error"
Allowed context: Action, Application.
 - "Read Content"
 - „"Access allowed "
 - "Access Denied "
Allowed context: Access Type.
 - *Condition* (COOSYSTEM@1.1:auditcondexpr)
Additional condition in form of a Fabasoft DUCX expression.
- **Software Component** (COOSYSTEM@1.1:auditcomponent)

Note: An object class inherits the audit settings of her base class.

5 Security Aspects of the Audit

5.1 Reading Audit logs

To be able to read the audit log of an object class the *Read Audit Log* permission (COOSYSTEM@1.1:AccTypeReadAuditLog) is needed. The audit log can be read using the *Audit Log* property (COOSYSTEM@1.1:objauditlog) of an object.

Note: If dedicated audit data sources are used, successful reading of data is required for each data source to be able to read audit information.

5.2 Data Source

If a user accesses a data source, for the first write access he or she needs the permission for creating tables. The table (fscauditlogentrylist) is created automatically and audit information is written to this table. Further on, only write and read access for this table is required.

6 Example

Configuration of the object class (e.g. *Text Document*, "Advanced" tab):

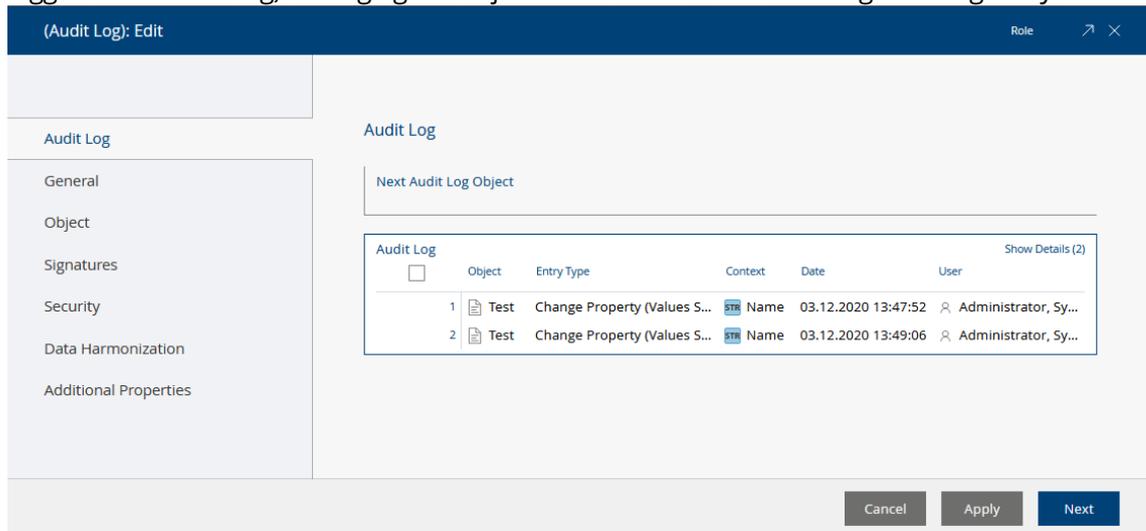
The screenshot shows the 'Text Document (Object Class for Documents): Edit' configuration window. The 'Advanced' tab is selected. The 'Audit Log Configuration' section is active, displaying a table with the following entries:

	Context	Audit Type	Condition	Software Component
1	STR COOSYSTEM@1.1:objname	Change Property (Va...		
2	▶ FSCVENV@1.1001:RenameO...	Call Action		

Buttons for 'Add Entry' and 'Show Details (2)' are visible. Below this, the 'Licensing Apps' section is empty, with 'Add Entry' and 'Search and Add' buttons. At the bottom right, there are 'Cancel', 'Apply', and 'Next' buttons.

- In the first configuration line is determined, that all changes of the *Name* property (COOSYSTEM@1.1:objname) are logged and the values are saved (the old and the new value are

logged in the audit log). Changing the object name leads to the following audit log entry:



- In the second configuration line is determined, that all calls of the *Rename Object* application (FSCVENV@1.1001:RenameObjectApp) are logged.

7 Environment Variables

- **AUDITENTRYLIMIT**
The value of this variable determines the maximum number of entries in an audit log object.
Default value: 2500 entries
- **AUDITSESSIONLIMIT**
The value of this variable determines the maximum number of open connections to the audit data source (per Fabasoft Folio Kernel instance).
Default value: 0 (unlimited).
- **AUDITTIMEOUT**
When logging an action, the value of this variable determines the period of time after which a new entry concerning this action is generated in the audit log object.
Default value: 60 s.