avaloq
An NEC Company

Workflow and Order Processing

Customization Upgrade Guide
Release 5.4
Area: Business Process Management

Latest version of this document

The latest version of this document can be found at https://docs.avaloq.com

Feedback

Please send any feedback to documentation@avaloq.com

Copyright Avaloq Evolution Ltd. All rights reserved.

The information in this document is provided for informational purposes only, is subject to change without notice and is not warranted to be error-free. No part of this document may be used, reproduced or transmitted in any form or by any means unless authorized by Avaloq Evolution Ltd through a written licence agreement. Further, this document does not grant any rights to, or in, the products mentioned therein and no rights of any kind relating to such products will be granted except pursuant to written agreements with Avaloq Evolution Ltd.

Avaloq Evolution Ltd. Allmendstr. 140 ¦ CH-8027 Zürich ¦ Switzerland

Version history

Version / date	Section	Description of the change
5.4v0 / 20 April 2022		This is a new document for release 5.4.

Contents

1 Introduction

This document describes monitoring and analysis tasks you should perform on release 4.9 or later, early in the lifecycle of the release (i.e. with enough time before you upgrade to release 5.5 or later), as a result of a security enhancement to the workflow engine.

Starting from Avaloq Core release 5.1, before executing a workflow action, the workflow engine now checks that the workflow action is visible for the session user. The workflow action is visible for the session user if either the user has the necessary access right or security is disabled. The new behaviour can already be enabled on releases 4.9 and 4.10 by setting the `avq.wfc` base parameter's `enbl_wfa_author_chk` item to "+" (default: "-").

In the past the visibility check was not deemed necessary because workflow actions were only executed via graphical user interfaces, where the workflow actions available to the user were the ones that were visible for the user. The visibility check was applied to calculate the set of available workflow actions for a given user, but not for the actual execution of the workflow action chosen from this set of available actions. This is no longer enough because workflow actions are now executed via APIs from external systems and services, as well as from graphical user interfaces.

So, starting in Avaloq Core release 5.1 (or 4.9, if the `avq.wfc` base parameter's `enbl_wfa_author_chk` item is set to "+"), the workflow engine performs visibility checks before executing workflow actions. If this check fails, a "No Permission" exception is raised.

This change in behaviour might break your customization. For some reason, a user might have been able in the past to execute a workflow action to which they had no direct access. With the new visibility check before the workflow action execution, this will no longer work. In this case, you must grant the appropriate access right to the user.

To check whether the current session user is authorized to execute a particular workflow action, you can use the SEC data dictionary's `user_is_wfc_action_visible` function.

To give clients enough time to adapt to this stricter behaviour of the workflow engine, there is a base parameter item available in release 4.9 and later to suppress the raising of the exception and only log unauthorized access for analysis. This is the `avq.sec` base parameter's `log_no_permit_on_start_wfa_in_intf_version` item. It contains a comma-separated list of interface versions for which the exceptions are only logged. Available values are:

- 2 (for the SmartClient)
- 3 (for "external" applications like Web Banking, Front Workplace, RM Workplace, and RM Cockpit)

Its default value is "2,3". In other words, by default, all exceptions get only logged on release 5.4 or earlier.

Avaloq does not currently plan to remove this base parameter item, but with release 5.5 (or maybe later, but no earlier) its default value will be changed to "" so that exceptions will get raised. It will still be possible for clients to deliberately set the value back to another value to suppress the raising of exceptions, explicitly accepting the risk of unauthorized workflow executions in the future. Our recommendation, however, is to enable hard security checks as soon as possible and therefore perform the necessary analysis on a lower release.

The exceptions are logged in table SEC_UI_ERR_LOG, which is used to store all security related UI exceptions.

You can use the reporting task TASK_SEC_UI_ERR_LOG (task ID: 1539) with the **Action** input parameter set to "doc_mgr#.doc#start_wfc_action" to display the generated logging information, and to find out which users executed a workflow action for which they did not have the necessary access rights.

Note that the workflow engine skips the workflow action visibility check in the following cases:

- It skips the visibility check for a workflow action that:
  - Is an auto-exec workflow action
  - Is started in a program executed by a background process; this covers all batch processing, for example, programs run from a process queue
  - Is started from a context action (in this case, the security is checked by the context action framework)
  - Is started within a task execution (in this case, the security is checked by the task framework)
  - Is started within the AFP contract administration delivered by Avaloq (in this case, the security is checked by the AFP framework)
  - Is started from a work item order of a master order (in this case, the security is checked by the master order workflow)
  - Is executed within the workflow, for example by an `exec` micro-command that triggers another workflow action (in this case, the security is checked by the master order workflow)

  These cases are so-called "trusted callers". There might be more types of trusted callers that have not been discovered yet.
- It skips the visibility check if WFC_ACTION.KEEP_ERR is set (these are considered to be error actions that are used for moving actions to a suspension state).

Please use the Avaloq Service Request Management portal to open a ticket if you get a "No Permission" exception and you do not agree with that exception because you think the workflow engine should skip the security check for that particular use case (i.e. the case should be registered in the list of trusted callers). In your ticket:

- Provide a detailed description of the use case (step-by-step instructions and an export of the generated log in table SEC_UI_ERR_LOG).
- Explain the reasons why you think the workflow engine should skip the security check for that particular use case.