

# SVCLOG Files

Last Modified on 07/22/2021 5:33 am EDT

## V9.x and later

### Overview

SVCLOG trace files preserve logs for diagnostic and debugging purposes. This logging feature is enabled for each execution engine, from its respective configuration file.

- Administration site
- Flowtime site
- Background Runtime Service
- Job Execution Service

The trace files are created in the path that you specify in the *traceDirectory* parameter in the configuration file.

#### IMPORTANT

Creating SVCLOG files consumes server resources, such as disk space and CPU time. We recommend that you enable this logging feature for a limited time, and closely monitor server performance.

Each SVCLOG has a source and a switch, which are required for creating a log file successfully. The source determines the kind of log to be created, eventViewer, SVC trace, or Database, and for which component, Runtime execution, Jobs execution, or HotOperations execution plan). The switch for each SVCLOG source should be turned on for the source to work.

The `initializeData` property in SVCLOG files determines what kind of events to collect. In the eventViewer trace logs, it determines the scope of messages from the application to display. It can be set to the following:

- All: shows all the messages from the component.
- Warning: shows both error and warning messages.
- Error: shows only the error messages.

### Tips

- If workflow execution is always enabled, for example, in development or testing environments, we recommend that you add a scheduled task that deletes obsolete log files.
- We recommend that you use the same path for the log files (this is not required).
- Use a *service trace viewer* tool to open the SVCLOG trace files.
- When you add the configuration sections to the config files, make sure you use the appropriate product version. If you have Cora SeQuence 7.x, use 7.0.0.0. If you have Cora SeQuence 8.x, use 8.0.0.0, and so on.

### Enable SVCLOG tracing

1. Open the appropriate configuration file for the execution engine for which, you want to create SVCLOG trace files.
2. Edit the configuration file.
  - a. Locate the relevant section (see the *execution engine configuration files* below).
  - b. Update the *traceDirectory* path.

- c. Remove the comment-out syntax by removing .
  - d. Save the configuration file.
3. Restart the service, BRS or JES.
- When you save a `web.config` file, the site automatically resets. You do not have to run IISReset.

### Disable SVCLOG tracing

1. Open the appropriate configuration file for the execution engine for which, you want to disable SVCLOG tracing.
2. Edit the configuration file.
  - a. Locate the relevant section (see the *execution engine configuration files* below).
  - b. Add the comment-out syntax by adding .
  - c. Save the configuration file.
3. Restart the service, BRS or JES.

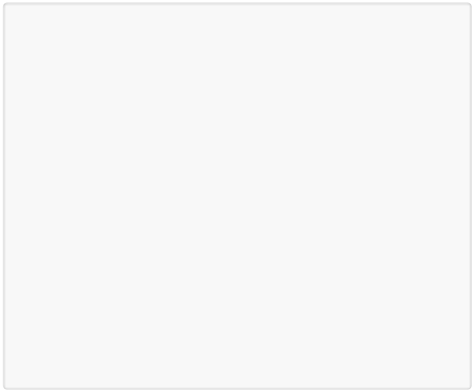
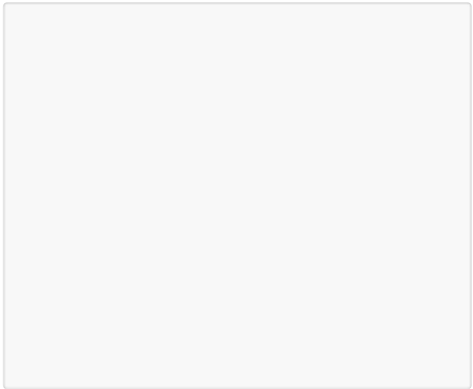
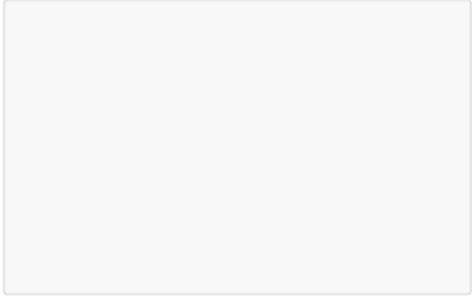
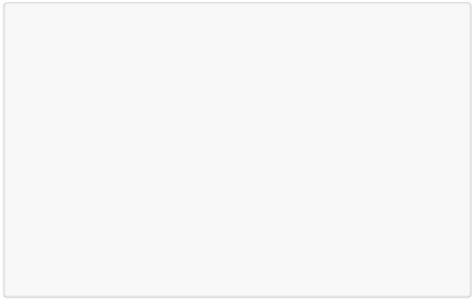
When you save a `web.config` file, the site automatically resets. You do not have to run IISReset.

#### IMPORTANT

Make sure while configuring the SVCLOG that the user running the services has access to the trace directory.

### Available SVCLOG files

Log name	Description	Components	Syntax
SVC runtime log	Audits the execution process, the result of if-else activities, and other activities' executions. It also displays the claims authentication process and the claims that return from the identity provider. For details, see <a href="#">this article</a> .	<ul style="list-style-type: none"> <li>• Flowtime</li> <li>• Administration</li> <li>• JES</li> <li>• BRS</li> <li>• WebAPI</li> </ul>	
SVC host log	Audits the launch of the Cora SeQuence engine and stop state.	<ul style="list-style-type: none"> <li>• Flowtime</li> <li>• Administration</li> <li>• JES</li> <li>• BRS</li> <li>• WebAPI</li> </ul>	

Log name	Description	Components	Syntax
SVC jobs log	Audits all jobs executed by JES and their status. For example, all steps of the email listener job, until an email is retrieved.	<ul style="list-style-type: none"> <li>• JES</li> </ul>	
SVC HotOperations log	Audits the execution of the HotOperations plan, evaluation of the allocation rules, and the teams the case goes to following the evaluation.	<ul style="list-style-type: none"> <li>• Flowtime</li> <li>• Administration</li> <li>• JES</li> <li>• BRS</li> <li>• WebAPI</li> </ul>	
SVC Kafka logs	Audits Kafka activity with Cora SeQuence.	<ul style="list-style-type: none"> <li>• Flowtime</li> <li>• Administration</li> <li>• JES</li> <li>• BRS</li> <li>• WebAPI</li> </ul>	
OBS trace	Collects all information related to the user or the group management in Cora SeQuence.	<ul style="list-style-type: none"> <li>• Administration</li> </ul>	

Log name	Description	Components	Syntax
Authorization	Saves Cora SeQUENCE failed authentications or access requests in the database.	<ul style="list-style-type: none"> <li>• Flowtime</li> <li>• Administration</li> <li>• JES</li> <li>• BRS</li> <li>• WebAPI</li> </ul>	
SVC import logs	Collects and displays the list of objects imported during workflow import, including errors that occurred during import.	<ul style="list-style-type: none"> <li>• Administration</li> </ul>	Saved automatically under <code>shared resources/import logs</code>
SVC wcf trace	Collects and displays the errors captured from the Cora SeQUENCE WCF service.	<ul style="list-style-type: none"> <li>• Flowtime</li> <li>• Administration</li> <li>• JES</li> <li>• BRS</li> <li>• WebAPI</li> </ul>	

**NOTE**

The following Powershell functions can enable the SVC Runtime log and SVC Jobs log respectively in the Cora SeQUENCE 9.X installation:

- PowerShell Function: `Enable-CoraSeSequenceRuntimeLogs`
- PowerShell Function: `Enable-CoraSeSequenceJobsLogs`