



Troubleshoot Qlik Sense using logs

Qlik Sense®

2.2

Copyright © 1993-2016 QlikTech International AB. All rights reserved.



Copyright © 1993-2016 QlikTech International AB. All rights reserved.

Qlik®, QlikTech®, Qlik Sense®, QlikView®, Sense® and the Qlik logo are trademarks which have been registered in multiple countries or otherwise used as trademarks by QlikTech International AB. Other trademarks referenced herein are the trademarks of their respective owners.

1 Introduction	13
1.1 Conventions	13
Style coding	13
Environment variables	14
1.2 Additional documentation	14
2 Qlik Sense Repository Service	15
2.1 Update user	15
Procedure	15
Success	15
2.2 Delete user	15
Procedure	15
Success	15
2.3 Open app	16
Procedure	16
Success	16
Errors	16
Your account is inactive	16
2.4 Create app	16
Procedure	16
Success	16
Errors	17
Forbidden	17
2.5 Delete app	17
Procedure	17
Success	17
Errors	17
The Delete operation failed	17
2.6 Publish app	17
Procedure	17
Success	18
2.7 Export app	18
Procedure	18
Success	18
Errors	18
Resource not found	18
2.8 Import app	19
Procedure	19
Success	19
Errors	19
Corrupt data	19
Write failed	19
Resource not found	20
2.9 Reload app	20
Procedure	20

Contents

Success	20
2.10 Duplicate app	20
Procedure	20
Success	21
Errors	21
Resource not found	21
2.11 Add app object	21
Procedure	21
Success	21
2.12 Update app object	21
Procedure	21
Success	22
2.13 Delete app object	22
Procedure	22
Success	22
2.14 Publish app object	22
Procedure	22
Success	22
2.15 Unpublish app object	22
Procedure	22
Success	23
2.16 Add extension	23
Procedure	23
Success	23
2.17 Create extension	23
Procedure	23
Success	23
2.18 Upload extension	23
Procedure	23
Success	24
Errors	24
The process cannot access the file	24
2.19 Delete extension	24
Procedure	24
Success	24
2.20 Add extension content	24
Procedure	24
Success	25
2.21 Delete extension content	25
Procedure	25
Success	25
2.22 Add content library	25
Procedure	25
Success	25

Contents

2.23 Delete content library	25
Procedure	25
Success	26
2.24 Upload content library content	26
Procedure	26
Success	26
Errors	26
Exception of type 'Qlik.Sense.Common.Exceptions.ConflictException' was thrown	26
2.25 Delete content library content	26
Procedure	26
Success	27
2.26 Add user access	27
Procedure	27
Success	27
Errors	27
Validation failed	27
2.27 Update user access	27
Procedure	27
Success	27
2.28 Delete user access	28
Procedure	28
Success	28
2.29 License user access request	28
Procedure	28
Success	28
2.30 License user access	28
Procedure	28
Success	28
Errors	29
Login access denied	29
2.31 Add user access from license	29
Procedure	29
Success	29
Errors	29
Cannot add user access	29
2.32 Add app privilege	30
Procedure	30
Success	30
2.33 Export certificates	30
Procedure	30
Success	30
2.34 Download license	30
Procedure	30
Success	30

Contents

Errors	31
License: Invalid serial number or control number	31
2.35 Add license	31
Procedure	31
Success	31
Errors	31
License: Invalid serial number or control number	31
2.36 Update license	31
Procedure	31
Success	31
Errors	32
License: Invalid serial number or control number	32
2.37 Delete license	32
Procedure	32
Success	32
2.38 Add rule	32
Procedure	32
Success	32
2.39 Update rule	32
Procedure	32
Success	33
2.40 Delete rule	33
Procedure	33
Success	33
2.41 Add stream	33
Procedure	33
Success	33
2.42 Delete stream	33
Procedure	33
Success	34
2.43 Server node registration	34
Procedure	34
Success	34
Errors	34
Cannot reach the following remote host when distributing certificates	34
2.44 Server node configuration	34
Procedure	34
Success	34
2.45 Create task	35
Procedure	35
Success	35
2.46 Update task	35
Procedure	35
Success	35

Contents

2.47 Delete task	35
Procedure	35
Success	35
2.48 Start task	36
Procedure	36
Success	36
Errors	36
Resource not found	36
2.49 Stop task	37
Procedure	37
Success	37
2.50 Synchronize user directory	37
Procedure	37
Success	37
2.51 Start repository	37
Procedure	37
Success	37
Errors	38
Certificates are not correctly installed	38
2.52 Stop repository	38
Procedure	38
Success	38
2.53 Check service status	38
Procedure	38
Success	38
Errors	39
The following service account does not exist	39
2.54 Load plugin	39
Procedure	39
Success	39
2.55 Audit rules	39
Procedure	39
Success	39
2.56 Audit security	39
Procedure	39
Success	40
2.57 Audit license	40
Procedure	40
Success	40
2.58 Audit license rule	40
Procedure	40
Success	40
2.59 License maintenance	40
Procedure	40

Success	41
2.60 Distribute certificate	41
Procedure	41
Success	41
Errors	41
Certificate distribution failed	41
3 Qlik Sense Proxy Service	42
3.1 Start proxy	42
Procedure	42
Success	42
Errors	42
Error occurred while retrieving hardware information	42
Proxy will not be listening to port '443'	43
3.2 Stop proxy	43
Procedure	43
Success	43
3.3 Open connection	43
Procedure	43
Success	43
Errors	44
Web socket connection could not be opened	44
Web exception: Protocol error: Response stream exists	44
Web exception: Protocol error: No response stream exists	44
Web exception when contacting service uri	44
Connection has been transferred to a streaming state to send a single error message	45
Web socket connection could not be opened	45
3.4 Close connection	45
Procedure	45
Success	45
Errors	45
Web socket connection could not be opened	45
Command=Open connection;Result=<different from zero>	46
3.5 Start session	46
Procedure	46
Success	46
Errors	47
Cannot start session	47
3.6 Stop session	47
Procedure	47
Success	47
Errors	47
Cannot stop session	47
3.7 Log out	48
Procedure	48
Success	48

Contents

Errors	48
Command=Logout;Result=<return code not zero>	48
3.8 Log in	48
Procedure	48
Success	49
Errors	49
Logon failed	49
3.9 Install certificate	49
Procedure	49
Success	49
4 Qlik Sense Scheduler Service	50
4.1 Start task	50
Procedure	50
Success	50
Errors	50
Task not found	50
Scheduler is not licensed	51
Scheduler is not Master	51
Found active session for Task	51
Task disabled	51
TaskExecutionSession already exists	51
App is not enabled	52
No slave nodes found to execute Task	52
Unable to create TaskExecutionSession	52
Unexpected exception when starting task	52
Unexpected exception when trying to start task	52
Max number of retries reached for task	52
Not possible to initiate retry	53
4.2 Finish task	53
Procedure	53
Success	53
Errors	53
Failed in TaskCompletedFail	53
Task finished with state <State>	53
Failed to remove session	53
Unexpected exception in TaskCompletedSuccess	54
Unexpected exception in TaskCompletedFail	54
Task failed	54
4.3 Execute task	54
Procedure	54
Success	54
Errors	54
Trying to set task id to state	54
TaskExecutionSession cannot be null	55
Suppressed state change	55

Contents

Unable to get <Task.Name>	55
4.4 Start master	55
Procedure	55
Success	55
4.5 Start slave	56
Procedure	56
Success	56
4.6 Resume master	56
Procedure	56
Success	56
4.7 Resume slave	56
Procedure	56
Success	56
4.8 Read initial settings	57
Procedure	57
Success	57
4.9 Log hardware information at the startup of the service	57
Procedure	57
Success	57
Errors	57
Failed when logging hardware information	57
4.10 Stop master	57
Procedure	57
Success	58
4.11 Stop slave	58
Procedure	58
Success	58
4.12 Pause master	58
Procedure	58
Success	58
4.13 Pause slave	58
Procedure	58
Success	58
4.14 Settings change for slave	59
Procedure	59
Success	59
5 Qlik Sense Engine Service	60
5.1 Open app	60
Procedure	60
Success	60
Errors	60
Already opened in different mode	60
Already opened	60
Invalid path	61

Contents

Resource not found	61
5.2 Create app	61
Procedure	61
Success	61
Errors	61
Invalid name	61
5.3 Delete app	62
Procedure	62
Success	62
Errors	62
Delete failed	62
5.4 Export app	62
Procedure	62
Success	62
Errors	63
Disk is full	63
File corrupt	63
5.5 Import app	63
Procedure	63
Success	63
Errors	63
File corrupt	63
Internal error	64
5.6 Reload app	64
Procedure	64
Success	64
5.7 Duplicate app	64
Procedure	64
Success	64
Errors	64
Delete failed	64
5.8 Publish app	65
Procedure	65
Success	65
Errors	65
Access denied	65
REST connection failure	65
5.9 Unpublish app	65
Procedure	65
Success	66
Errors	66
Access denied	66
REST connection failure	66
5.10 Replace app	66
Success	66

Contents

5.11 Start engine	67
Procedure	67
Success	67
Errors	67
Registration failure	67
5.12 Stop engine	67
Procedure	67
Success	68
Errors	68
Cache trimmer never stopped	68


1 Introduction

Troubleshooting may be needed when Qlik Sense does not behave as expected (for example, if the system responds with an error message that needs further investigation or does not respond at all when an error occurs).

The log messages produced by Qlik Sense provide important information that can be used to detect security incidents, operational problems, and policy violations.

The description of how to troubleshoot Qlik Sense using logs is based on "use cases", each of which corresponds to a typical user or system action, such as opening an app or stopping a task.

Each use case is described using the sections listed in the following table.

Section	Description
Procedure	This section lists the actions that are performed (and logged) by Qlik Sense when the use case is carried out.
Success	This section lists the log files to which Qlik Sense writes log entries in case of success.
Errors	<p>This section lists errors that may occur when the use case is performed.</p> <p>It is recommended to check the <code><MachineName>_Service_<Service>.txt</code> file in case of an error as all errors are logged in this file. Each error section also includes a description of the actions to perform in case of an error.</p> <div style="border: 1px solid gray; padding: 5px; margin-top: 10px;">  <i>For some use cases, there is no Error section. Information on errors will be added in future releases of the Qlik Sense documentation.</i> </div>

This document is derived from the online help for Qlik Sense. It is intended for those who want to read parts of the help offline or print pages easily, and does not include any additional information compared with the online help.

1.1 Conventions

The following conventions are used in the documentation for Qlik Sense.

Style coding

- Menu commands and dialog options are written in **bold**.
- File names and paths are written in *Italics*.
- Sample code is written in `Lucida Console`.

Environment variables

The paths used in the documentation for Qlik Sense may use environment variables. The variables and the equivalent paths in the Microsoft Windows operating system are listed below.

Environment variable	Microsoft Windows
<i>%LocalAppData%</i>	<i>C:\Users\<username>\AppData\Local</i>
<i>%ProgramData%</i>	<i>C:\ProgramData</i>
<i>%ProgramFiles%</i>	<i>C:\Program Files</i>
<i>%UserProfile%</i>	<i>C:\Users\<username></i>

1.2 Additional documentation

Besides this document, the following related documentation is available for Qlik Sense:

- [Manage Qlik Sense sites](#): Describes how to manage a Qlik Sense site.
- [Plan Qlik Sense deployments](#): Describes Qlik Sense and provides reference information on the architecture, security, logging, and licensing.

2 Qlik Sense Repository Service

This section describes how to use the Qlik Sense logs to troubleshoot problems related to the Qlik Sense Repository Service (QRS).

See also:

- ▢ [Introduction \(page 13\)](#)

2.1 Update user

Procedure

Qlik Sense performs the following procedure:

1. A request is sent to the Qlik Sense Repository Service (QRS).
2. The QRS verifies that the update does not disable any service account that the user is allowed to update.
3. The QRS updates the user.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditSecurity_Repository.txt*

2.2 Delete user

Procedure

Qlik Sense performs the following procedure:

1. A request is sent to the Qlik Sense Repository Service (QRS).
2. The QRS verifies that the user to be deleted is not a service account or the last user with root admin access.
3. The QRS removes the user.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditSecurity_Repository.txt*

2.3 Open app

Procedure

Qlik Sense performs the following procedure:

1. A request is sent to the Qlik Sense Repository Service (QRS).
2. The QRS checks that the app exists.
3. The QRS checks that the user is allowed to open the app.
4. The QRS sends a request to the Qlik Sense Engine Service (QES).
5. The QES checks the access to the data set in the app.
6. The QES loads the app and returns it.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Repository.txt*
- *<MachineName>_AuditActivity_Engine.txt*

Errors

Your account is inactive

Description	Command=Open app;Result=403;ResultText=Error: Security
Message	Your account is inactive. Contact your administrator to activate it. (HTTP code: 403)
Action	Contact the Qlik Sense system administrator to obtain the correct access rights.

2.4 Create app

Procedure

Qlik Sense performs the following procedure:

1. A request is sent to the Qlik Sense Repository Service (QRS).
2. The QRS checks that the user is allowed to create an app.
3. The QRS sends a request to the Qlik Sense Engine Service (QES).
4. The QES creates the app.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Repository.txt*

Errors

Forbidden

Description	Command=Create app;Result=403;ResultText=Error: Security
Message	Forbidden (HTTP code: 403)
Action	The user is not allowed to create an app.

2.5 Delete app

Procedure

Qlik Sense performs the following procedure:

1. A request is sent to the Qlik Sense Repository Service (QRS).
2. The QRS marks the app as deleted in the repository database.
3. The QRS requests the Qlik Sense Engine Service (QES) to delete the app from disk.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Repository.txt*
- *<MachineName>_AuditActivity_Engine.txt*

Errors

The Delete operation failed

Description	Command=Delete app;Result=400;ResultText=Error: PersistenceFailed
Message	The Delete operation failed (HTTP code: 400)
Action	Probable cause of error: The app does not exist.

2.6 Publish app

Procedure

Qlik Sense performs the following procedure:

1. A request is sent to the Qlik Sense Repository Service (QRS).
2. The QRS links the app to a stream.
3. The QRS checks if the published app is renamed.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Repository.txt*

2.7 Export app

Procedure

Qlik Sense performs the following procedure:

1. A request is sent to the Qlik Sense Repository Service (QRS).
2. The QRS provides the Qlik Sense Engine Service (QES) with the information needed to find the app.
3. The app ID is downloaded from the QES to the client.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Repository.txt*
- *<MachineName>_AuditActivity_Engine.txt*

Errors

Resource not found

Description	Command=Export app;Result=9003;ResultText=Error: EngineQix
Message	Resource not found Parameters: QVF header error (read) (HTTP code: 400)
Actions	Proceed as follows: <ol style="list-style-type: none">1. Check that the resource exists in the Qlik Management Console (QMC).2. Verify that you can open the app in the Qlik Sense hub.3. Check if the QVF file exists on the central node or on a rim node in the Qlik Sense site.

2.8 Import app

Procedure

Qlik Sense performs the following procedure:

1. A request is sent to the Qlik Sense Repository Service (QRS).
2. If you have access to the Qlik Sense system where the app was created, the following happens:
 - a. The QRS stores metadata in the repository database.
 - b. The QRS contacts the Qlik Sense Engine Service (QES).
 - c. The app is migrated (if necessary).
3. The QES imports the app (including objects) and persists it.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Repository.txt*
- *<MachineName>_AuditActivity_Engine.txt*

Errors

Corrupt data

Description	Command=Import app;Result=11;ResultText=Error: EngineQix
Message	Corrupt data Parameters: QVF File corrupt (HTTP code: 500)
Action	The format of the app is invalid and it cannot be imported.

Write failed

Description	Command=Import app;Result=9000;ResultText=Error: EngineQix
Message	Write Failed Parameters: REST client response error (HTTP code: 500)
Action	Proceed as follows: <ol style="list-style-type: none">1. Check that the Qlik Sense Engine Service (QES) is up and running as the request may have timed out.2. Verify that you can open the app where it was created in Qlik Sense.3. Verify that you can load app data in the data load editor and that you can reload the app in the Qlik Management Console (QMC).4. Check the reload history of the app.

Resource not found

Description	Command=Import app;Result=2;ResultText=Error: EngineQix
Message	Resource not found Parameters: QVF object error (HTTP code: 500)
Action	Provide a working QVF file for import.

2.9 Reload app

The data in an app can be reloaded in different ways:

- The user manually reloads the app data in the Qlik Management Console (QMC).
- The app data is reloaded by a scheduled task in the repository database.
- The user manually loads data in the data load editor. Information on such reloads is logged in *<MachineName>_AuditActivity_Engine.txt*.

See: *Reload app (page 64)*

Procedure

Qlik Sense performs the following procedure:

1. A request is sent to the Qlik Sense Repository Service (QRS).
2. The QRS contacts the Qlik Sense Scheduler Service (QSS).
3. The QSS starts the reload task.
4. The QSS contacts the Qlik Sense Engine Service (QES) and initiates a reload of the app.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Repository.txt*
- *<MachineName>_AuditActivity_Engine.txt*
- *<MachineName>_AuditActivity_Scheduler.txt*

2.10 Duplicate app

Procedure

Qlik Sense performs the following procedure:

1. A request is sent to the Qlik Sense Repository Service (QRS).
2. The QRS checks that the app exists.
3. The QRS checks that the user is allowed to duplicate the app.

4. The QRS sends a request to the Qlik Sense Engine Service (QES).
5. The QES makes a copy of the app.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Repository.txt*
- *<MachineName>_AuditActivity_Engine.txt*

Errors

Resource not found

Description	Command=Duplicate app;Result=-2146233074;ResultText=Error: Resource not found [LOCERR_PERSISTENCE_NOT_FOUND]:Resource not found(QVF header error (read))
Message	Duplicate failed. App 'Test' duplicated to 'Test(1)'. Additional info: 'Resource not found [LOCERR_PERSISTENCE_NOT_FOUND]:Resource not found(QVF header error (read))',
Action	Probable cause of error: The app has been deleted from disk. Try to import the app again.

2.11 Add app object

Procedure

Qlik Sense performs the following procedure:

1. The Qlik Sense Engine Service (QES) contacts the Qlik Sense Repository Service (QRS).
2. The app object is added to the repository database in a bulk operation.
3. The QES persists the app object.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Repository.txt*

2.12 Update app object

Procedure

Qlik Sense performs the following procedure:

1. The Qlik Sense Engine Service (QES) contacts the Qlik Sense Repository Service (QRS).
2. The app object is updated in the repository database in a bulk operation.
3. The QES updates the persisted app object.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Repository.txt*

2.13 Delete app object

Procedure

Qlik Sense performs the following procedure:

1. The Qlik Sense Engine Service (QES) contacts the Qlik Sense Repository Service (QRS).
2. The app object is set to be deleted in the repository database in a bulk operation.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Repository.txt*

2.14 Publish app object

Procedure

Qlik Sense performs the following procedure:

1. A request is sent to the Qlik Sense Repository Service (QRS).
2. The QRS sets the app object as published in the repository database.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Repository.txt*

2.15 Unpublish app object

Procedure

Qlik Sense performs the following procedure:

1. A request is sent to the Qlik Sense Repository Service (QRS).
2. The QRS sets the app object as unpublished in the repository database.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Repository.txt*

2.16 Add extension

Procedure

Qlik Sense performs the following procedure:

1. A request is sent to the Qlik Sense Repository Service (QRS).
2. The QRS adds the extension.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Repository.txt*

2.17 Create extension

Procedure

Qlik Sense performs the following procedure:

1. A request is sent to the Qlik Sense Repository Service (QRS).
2. The QRS creates the extension.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Repository.txt*

2.18 Upload extension

Procedure

Qlik Sense performs the following procedure:

1. A request is sent to the Qlik Sense Repository Service (QRS).
2. The QRS uploads the extension.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Repository.txt*

Errors

The process cannot access the file

Description	Command=Upload extension;Result=500;ResultText=Error: IO
Message	The process cannot access the file '<Filename>' because it is being used by another process. (HTTP code: 500)
Action	Try the following measures: <ul style="list-style-type: none">• Wait for any ongoing process in the Qlik Management Console (QMC) to finish and then upload the extension again.• Restart the QRS.

2.19 Delete extension

Procedure

Qlik Sense performs the following procedure:

1. A request is sent to the Qlik Sense Repository Service (QRS).
2. The QRS deletes the extension.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Repository.txt*

2.20 Add extension content

Procedure

Qlik Sense performs the following procedure:

1. A request is sent to the Qlik Sense Repository Service (QRS).
2. The QRS adds the extension content.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Repository.txt*

2.21 Delete extension content

Procedure

Qlik Sense performs the following procedure:

1. A request is sent to the Qlik Sense Repository Service (QRS).
2. The QRS deletes the extension content.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Repository.txt*

2.22 Add content library

Procedure

Qlik Sense performs the following procedure:

1. A request is sent to the Qlik Sense Repository Service (QRS).
2. The QRS adds the content library.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Repository.txt*

2.23 Delete content library

Procedure

Qlik Sense performs the following procedure:

1. A request is sent to the Qlik Sense Repository Service (QRS).
2. The QRS deletes the content library.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Repository.txt*

2.24 Upload content library content

Procedure

Qlik Sense performs the following procedure:

1. A request is sent to the Qlik Sense Repository Service (QRS).
2. The QRS uploads the content library content.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Repository.txt*

Errors

Exception of type 'Qlik.Sense.Common.Exceptions.ConflictException' was thrown

Description	Command=Upload content library content;Result=409;ResultText=Error: Conflict
Message	Exception of type 'Qlik.Sense.Common.Exceptions.ConflictException' was thrown. (HTTP code: 409)
Action	The library content already exists. Rename or replace the library content.

2.25 Delete content library content

Procedure

Qlik Sense performs the following procedure:

1. A request is sent to the Qlik Sense Repository Service (QRS).
2. The QRS deletes the content library content.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Repository.txt*

2.26 Add user access

Procedure

Qlik Sense performs the following procedure:

1. A request is sent to the Qlik Sense Repository Service (QRS).
2. The QRS verifies that the login/user access can be added.
3. The QRS adds the login/user access.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditSecurity_Repository.txt*

Errors

Validation failed

Description	Command=Add User Access;Result=400;ResultText=Error: Bad Request
Message	<BrokenRules>
Action	Make sure that you are allowed to add the login/user access.

2.27 Update user access

Procedure

Qlik Sense performs the following procedure:

1. A request is sent to the Qlik Sense Repository Service (QRS).
2. The QRS verifies that the login/user access can be updated.
3. The QRS updates the login/user access.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditSecurity_Repository.txt*

2.28 Delete user access

Procedure

Qlik Sense performs the following procedure:

1. A request is sent to the Qlik Sense Repository Service (QRS).
2. The QRS verifies that the login/user access can be deleted.
3. The QRS deletes the login/user access.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditSecurity_Repository.txt*

2.29 License user access request

Procedure

Qlik Sense performs the following procedure:

- Qlik Sense checks that the user has a valid login access pass. This is done every time a user requests access to a resource (such as an app).

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditSecurity_Repository.txt*

2.30 License user access

Procedure

Qlik Sense performs the following procedure:

- The Qlik Sense Repository Service (QRS) checks the Qlik Sense license to determine if the user can be given an access pass.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditSecurity_Repository.txt*

Errors

Login access denied

Description	Command=License user access;Result=403;ResultText=Error: Access denied
Message	Login access denied for SessionID: 'xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx', Hostname: 'xx.xx.xx.xx', OperationType: 'UsageDenied'
Action	Contact the Qlik Sense system administrator to obtain the correct access rights.

2.31 Add user access from license

Procedure

Qlik Sense performs the following procedure:

1. A request is sent to the Qlik Sense Repository Service (QRS).
2. The QRS verifies that the user access can be added.
3. The QRS identifies the user access.
4. The QRS adds the user access.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditSecurity_Repository.txt*

Errors

Cannot add user access

Description	Command=Add user access from license;Result=400;ResultText=Error: Bad Request
Message	<BrokenRules>
Action	Check the following: <ul style="list-style-type: none">• That the license has not expired• That the license is not blacklisted• That there are enough tokens available in the Qlik Sense license

2.32 Add app privilege

Procedure

Qlik Sense performs the following procedure:

- The Qlik Sense Repository Service (QRS) adds app privileges to the appropriate user groups.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditSecurity_Repository.txt*.

2.33 Export certificates

Procedure

Qlik Sense performs the following procedure:

- The Qlik Sense Repository Service (QRS) exports the certificates.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditSecurity_Repository.txt*

2.34 Download license

Procedure

Qlik Sense performs the following procedure:

- The Qlik Sense Repository Service (QRS) validates the license and downloads it.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditSecurity_Repository.txt*

Errors

License: Invalid serial number or control number

Description	Command=Download license;Result=400;ResultText=Error: ValidationFailed
Message	License: Invalid serial number or control number (HTTP code: 400)
Action	Provide a valid Qlik Sense license.

2.35 Add license

Procedure

Qlik Sense performs the following procedure:

- The Qlik Sense Repository Service (QRS) adds the license provided by the user.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditSecurity_Repository.txt*

Errors

License: Invalid serial number or control number

Description	Command=Update license;Result=400;ResultText=Error: ValidationFailed
Message	License: Invalid serial number or control number (HTTP code: 400)
Action	Provide a valid Qlik Sense license.

2.36 Update license

Procedure

Qlik Sense performs the following procedure:

- The Qlik Sense Repository Service (QRS) adds the license provided by the user.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditSecurity_Repository.txt*

Errors

License: Invalid serial number or control number

Description	Command=Update license;Result=400;ResultText=Error: ValidationFailed
Message	License: Invalid serial number or control number (HTTP code: 400)
Action	Provide a valid Qlik Sense license.

2.37 Delete license

Procedure

Qlik Sense performs the following procedure:

- The Qlik Sense Repository Service (QRS) deletes the license.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Repository.txt*

2.38 Add rule

Procedure

Qlik Sense performs the following procedure:

1. A request is sent to the Qlik Sense Repository Service (QRS).
2. The QRS adds the rule.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Repository.txt*

2.39 Update rule

Procedure

Qlik Sense performs the following procedure:

1. A request is sent to the Qlik Sense Repository Service (QRS).
2. The QRS updates the rule.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Repository.txt*

2.40 Delete rule

Procedure

Qlik Sense performs the following procedure:

1. A request is sent to the Qlik Sense Repository Service (QRS).
2. The QRS deletes the rule.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Repository.txt*

2.41 Add stream

Procedure

Qlik Sense performs the following procedure:

1. A request is sent to the Qlik Sense Repository Service (QRS).
2. The QRS adds the stream.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Repository.txt*

2.42 Delete stream

Procedure

Qlik Sense performs the following procedure:

1. A request is sent to the Qlik Sense Repository Service (QRS).
2. The QRS deletes the stream.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Repository.txt*

2.43 Server node registration

Procedure

Qlik Sense performs the following procedure:

- The Qlik Sense Repository Service (QRS) establishes a connection to the specified server address.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Repository.txt*

Errors

Cannot reach the following remote host when distributing certificates

Description	Command=Server node registration;Result=400;ResultText=Error: REST
Message	Cannot reach the following remote host when distributing certificates: http://<ServerAddress>:4444/setup/certificateDistribution (HTTP code: 400)
Action	Provide a valid server address.

2.44 Server node configuration

Procedure

Qlik Sense performs the following procedure:

- The Qlik Sense Repository Service (QRS) updates the configuration for a server node.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Repository.txt*

2.45 Create task

Procedure

Qlik Sense performs the following procedure:

1. The Qlik Sense Repository Service (QRS) creates a task and stores it in the repository database.
2. The task information is synchronized by the QRS to the rim nodes.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Repository.txt*

2.46 Update task

Procedure

Qlik Sense performs the following procedure:

1. The Qlik Sense Repository Service (QRS) updates the task and stores it in the repository database.
2. The updated task information is synchronized by the QRS to the rim nodes.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Repository.txt*

2.47 Delete task

Procedure

Qlik Sense performs the following procedure:

1. The Qlik Sense Repository Service (QRS) deletes the task and removes it from the repository database.
2. The deletion of the task is synchronized by the QRS to the rim nodes.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Repository.txt*

2.48 Start task

Procedure

Qlik Sense performs the following procedure:

- The Qlik Sense Repository Service (QRS) requests the Qlik Sense Scheduler Service (QSS) to execute the task.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Repository.txt*

Errors

Resource not found

Description	Command=Start task;Result=-1;ResultText=Error: -----System.Net.WebException: Unable to connect to the remote server --> System.Net.Sockets.SocketException: No connection could be made because the target machine actively refused it 127.0.0.1:5050 at System.Net.Sockets.Socket.EndConnect(IAsyncResult asyncResult) at System.Net.ServicePoint.ConnectSocketInternal(Boolean connectFailure, Socket s4, Socket s6, Socket& socket, IPAddress& address, ConnectSocketState state, IAsyncResult asyncResult, Exception& exception) --- End of inner exception stack trace --- at System.Net.HttpWebRequest.EndGetRequestStream(IAsyncResult asyncResult, TransportContext& context) at System.Net.HttpWebRequest.EndGetRequestStream(IAsyncResult asyncResult) at System.Net.WebClient.UploadBitsRequestCallback(IAsyncResult result)-----
Message	Trailing task exception in SchedulerClient.StartTask('1xx1111x-111x-111x-1xx1-111xxx1x1x11'): '↓-----↓System.Net.WebException: Unable to connect to the remote server --> System.Net.Sockets.SocketException: No connection could be made because the target machine actively refused it 127.0.0.1:5050↓ ↓ at System.Net.Sockets.Socket.EndConnect(IAsyncResult asyncResult)↓ ↓ at System.Net.ServicePoint.ConnectSocketInternal(Boolean connectFailure, Socket s4, Socket s6, Socket& socket, IPAddress& address, ConnectSocketState state, IAsyncResult asyncResult, Exception& exception)↓ ↓ --- End of inner exception stack trace ---↓ ↓ at System.Net.HttpWebRequest.EndGetRequestStream(IAsyncResult asyncResult, TransportContext& context)↓ ↓ at System.Net.HttpWebRequest.EndGetRequestStream(IAsyncResult asyncResult)↓ ↓ at System.Net.WebClient.UploadBitsRequestCallback(IAsyncResult result)↓-----'
Action	Check that the Qlik Sense Scheduler Service (QSS) is up and running.

2.49 Stop task

Procedure

Qlik Sense performs the following procedure:

- The Qlik Sense Repository Service (QRS) requests the Qlik Sense Scheduler Service (QSS) to stop the execution of the task.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Repository.txt*

2.50 Synchronize user directory

Procedure

Qlik Sense performs the following procedure:

- The Qlik Sense Repository Service (QRS) synchronizes the user directory.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Repository.txt*

2.51 Start repository

Procedure

Qlik Sense performs the following procedure:

1. Validate the command line parameters for the QRS.
2. The QRS loads the logging framework.
3. The QRS validates or initializes the certificates.
4. The QRS establishes a connection to the Qlik Sense Repository Database (QRD).
5. If needed, the QRS performs the migration steps of the repository database schema.
6. Hardware information is collected.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_Service_Repository.txt*

Errors

Certificates are not correctly installed

Description	Command=Start Repository;Result=-1;ResultText=Certificates are not correctly installed
Message	Initializing certificates.
Action	The error is related to an invalid certificate. Contact your system administrator for support regarding the certificates.

2.52 Stop repository

Procedure

Qlik Sense performs the following procedure:

1. The internal services are stopped.
2. The background threads are stopped.
3. All Qlik Sense services, except for the Qlik Sense Repository Database (QRD), are stopped.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_Service_Repository.txt*

2.53 Check service status

Procedure

Qlik Sense performs the following procedure:

- The Qlik Sense Repository Service (QRS) checks that the communication with the specified Qlik Sense service is working.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_Service_Repository.txt*

Errors

The following service account does not exist

Description	Command=Check service status;Result=403;ResultText=Error: Security
Message	The following service account does not exist: <Service.Name> (HTTP code: 403)
Action	Restart the specified Qlik Sense service.

2.54 Load plugin

Procedure

Qlik Sense performs the following procedure:

- The Qlik Sense Repository Service (QRS) loads the specified plugins during the installation process.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_Service_Repository.txt*

2.55 Audit rules

Procedure

Qlik Sense performs the following procedure:

- The Qlik Sense Repository Service (QRS) fetches the specified rules from the repository database.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditSecurity_Repository.txt*

2.56 Audit security

Procedure

Qlik Sense performs the following procedure:

- The Qlik Sense Repository Service (QRS) fetches the specified security rules from the repository database.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditSecurity_Repository.txt*

2.57 Audit license

Procedure

Qlik Sense performs the following procedure:

- A Qlik Sense Repository Service (QRS) thread runs in the background.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditSecurity_Repository.txt*

2.58 Audit license rule

Procedure

Qlik Sense performs the following procedure:

- A Qlik Sense Repository Service (QRS) thread runs in the background and monitors the license usage.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditSecurity_Repository.txt*

2.59 License maintenance

Procedure

Qlik Sense performs the following procedure:

- A Qlik Sense Repository Service (QRS) thread runs in the background.

The thread continuously checks the proxy sessions linked to the license in order to:

- Prevent users from over-consuming license tokens

- Check if any changes linked to the license prevent the user from consuming a login pass
- Check if the proxy session has timed out for the login access pass

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditSecurity_Repository.txt*

2.60 Distribute certificate

Procedure

Qlik Sense performs the following procedure:

- The Qlik Sense Repository Service (QRS) distributes a certificate to the rim node.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_Service_Repository.txt*

Errors

Certificate distribution failed

Description	Command=Distribute certificate;Result=-2146233088;ResultText=Error: Cannot reach the following remote host when distributing certificates: http://<ServerAddress>:4444/setup/certificateDistribution
Message	Certificate distribution failed
Action	Provide a valid server address.

3 Qlik Sense Proxy Service

This section describes how to use the Qlik Sense logs to troubleshoot problems related to the Qlik Sense Proxy Service (QPS).

See also:

- ▢ [Introduction \(page 13\)](#)

3.1 Start proxy

Procedure

Qlik Sense performs the following procedure:

1. The Qlik Sense Proxy Service (QPS) creates a proxy.
2. The QPS starts the proxy.
3. Hardware information is collected.
4. The QPS writes the hardware information in the logs.
5. The QPS installs the certificates.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Proxy.txt*
- *<MachineName>_Service_Proxy.txt*

Errors

Error occurred while retrieving hardware information

Description	Command=Start proxy;Result= -1;ResultText=Error occurred while retrieving hardware information
Message	The message may vary.
Action	Check the registry settings.

Proxy will not be listening to port '443'

Description	Port collisions may occur when internal services are started during the startup of the QPS: Command=Start internal service;Result=-2147467259;ResultText=Error: Only one usage of each socket address
Message	Proxy will not be listening to port '443' (most likely bound by another process)
Action	Proceed as follows: <ol style="list-style-type: none">1. Make sure that port 443 is available for Qlik Sense to use.2. Restart the Qlik Sense services.

3.2 Stop proxy

Procedure

Qlik Sense performs the following procedure:

1. The Qlik Sense Proxy Service (QPS) notifies the Qlik Sense Repository Service (QRS).
2. The QPS settings and notification poller threads are stopped.
3. The QPS checks that the QRS has installed the certificate.
4. The internal services are stopped.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_Service_Proxy.txt*

3.3 Open connection

Procedure

Qlik Sense performs the following procedure:

- Open a socket to the Qlik Sense Engine Service (QES). This happens every time a user opens an app in Qlik Sense.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Proxy.txt*
- *<MachineName>_Service_Proxy.txt*

Errors

Web socket connection could not be opened

Description	Command=Open connection;Result=400;ResultText=Error: Web socket connection could not be opened
Message	Connection '<ID of connection>' has been transferred to a streaming state to send a single error message
Action	<p>Proceed as follows:</p> <ol style="list-style-type: none"> 1. Check if the QES is up and running. If not, start the QES. 2. Check that it is possible to connect to the QES.

Web exception: Protocol error: Response stream exists

Description	Command=Open connection;Result=400;ResultText=<different from zero>;ResultText=Error: <error message>
Message	Web exception: Protocol error: Response stream exists. Remote endpoint '{0}' + various exception message
Action	<p>Check the following:</p> <ul style="list-style-type: none"> • That the QES is up and running. • That you can connect to the QES on port 4747 locally on the server.

Web exception: Protocol error: No response stream exists

Description	Command=Open connection;Result=400;ResultText=<different from zero>;ResultText=Error: <error message>
Message	Web exception: Protocol error: No response stream exists when contacting '{0}' + various exception message
Action	<p>Check the following:</p> <ul style="list-style-type: none"> • That the QES is up and running. • That you can connect to the QES on port 4747 locally on the server.

Web exception when contacting service uri

Description	Command=Open connection;Result=400;ResultText=<different from zero>;ResultText=Error: <error message>
Message	Web exception when contacting service uri '{0}' + various exception message
Action	Check that service at URI is available.

Connection has been transferred to a streaming state to send a single error message

Description	Command=Open connection;Result=400;ResultText=<different from zero>;ResultText=Error: <error message>
Message	Connection '{0}' has been transferred to a streaming state to send a single error message + various exception message
Action	Check the following: <ul style="list-style-type: none">• That the QES is up and running.• That you can connect to the QES on port 4747 locally on the server.

Web socket connection could not be opened

Description	Command=Open connection;Result=400;ResultText=<different from zero>;ResultText=Error: <error message>
Message	Web socket connection could not be opened
Action	Check the following: <ul style="list-style-type: none">• That the QES is up and running.• That you can connect to the QES on port 4747 locally on the server.

3.4 Close connection

Procedure

Qlik Sense performs the following procedure:

- Close the socket to the Qlik Sense Engine Service (QES). This happens every time a user closes an app (for example, by closing a tab in a browser).

Success

In case of success, log entries are written in the following files throughout the procedure:

- <MachineName>_AuditActivity_Proxy.txt

Errors

Web socket connection could not be opened

Description	Command=Open connection;Result=400;ResultText=Error: Web socket connection could not be opened
--------------------	--

Message	Connection '<ID of connection>' has been transferred to a streaming state to send a single error message
Action	Proceed as follows: <ol style="list-style-type: none">1. Check if the QES is up and running. If not, start the QES.2. Check that it is possible to connect to the QES.

Command=Open connection;Result=<different from zero>

Description	Command=Open connection;Result=<different from zero>;ResultText=Error: <error message>
Message	Search for any log message related to Close connection. If you find a log entry similar to the one described above, evaluate the information in the Message field.
Action	Proceed as follows: <ol style="list-style-type: none">1. Analyze the memory consumption and run netstat (TCP sockets in illegal state). If you encounter any issues, restart the QPS as connections (that is, web sockets to the QES) are disposed by the Qlik Sense Proxy Service (QPS).2. If the problem persists, check that the QES is up and running.3. Contact Qlik support.

3.5 Start session

Procedure

Qlik Sense performs the following procedure:

- A new proxy session starts when a user accesses Qlik Sense.

Success

In case of success, log entries are written in the following files throughout the procedure:

- <MachineName>_AuditActivity_Proxy.txt

Errors

Cannot start session

Description	<p>If a user cannot start a new proxy session, it is most likely related to:</p> <ul style="list-style-type: none"> • The configuration of the Qlik Sense Proxy Service (QPS) • Installation-specific issues <p>Search for any log message with the following information:</p> <p>Command=Start session;Result=<different from zero>;ResultText=Error: <error message></p>
Message	If you find a log entry similar to the one described above, evaluate the information in the Message field.
Action	If there are no log entries for Start session, check the configuration of the QPS by examining the log entries for the Start proxy command.

3.6 Stop session

Procedure

Qlik Sense performs the following procedure:

- A proxy session stops when it times out or when the user closes the browser tab where the session is running.

Success

In case of success, log entries are written in the following files throughout the procedure:

- <MachineName>_AuditActivity_Proxy.txt

Errors

Cannot stop session

Description	<p>Search for any log message with the following information:</p> <p>Command=Stop session;Result=<different from zero>;ResultText=Error: <error message></p>
Message	If you find a log entry similar to the one described above, evaluate the information in the Message field.

Action	<p>If there are no log entries for Stop session, check the configuration of the Qlik Sense Proxy Service (QPS) by examining the log entries for the Start proxy command.</p> <p>As a last resort, restart the QPS.</p>
---------------	--

3.7 Log out

Procedure

Qlik Sense performs the following procedure:

- The user is logged out when the proxy session times out or when the user actively logs out or the closes all browser tabs related to Qlik Sense.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditSecurity_Proxy.txt*

Errors

Command=Logout;Result=<return code not zero>

Description	Command=Logout;Result=<return code not zero>;ResultText=Error: <error message>
Message	If you find a log entry similar to the one described above, evaluate the information in the Message field.
Action	<p>Proceed as follows, if a user repeatedly experiences problem when logging out:</p> <ol style="list-style-type: none"> 1. Restart the Qlik Sense Proxy Service (QPS). 2. Analyze the status of the sockets by running netstat and search for sockets in CLOSE_WAIT, SYN_SENT, or FIN_WAIT_2 state.

3.8 Log in

Procedure

Qlik Sense performs the following procedure:

1. Check if the user is linked to a valid user directory.
2. Check if the user exists in the repository database.
3. Check if the user has access to Qlik Sense.
4. Check if the user has access linked to the license.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditSecurity_Proxy.txt*

Errors

Logon failed

Description	Command=Login;ResultCode=403;ResultText=Error: Access Denied
Message	Login failed for user '<username>' wrong credentials?
Action	Proceed as follows: <ul style="list-style-type: none">• Check that the user exists in the repository database.• Check that the user has access rights to Qlik Sense linked to a license.• Check that the user is not blocked.

3.9 Install certificate

Procedure

Qlik Sense performs the following procedure:

- The Qlik Sense Proxy Service (QPS) waits for the certificates to be installed in the repository database.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditSecurity_Proxy.txt*

4 Qlik Sense Scheduler Service

This section describes how to use the Qlik Sense logs to troubleshoot problems related to the Qlik Sense Scheduler Service (QSS).

See also:

- ▢ [Introduction \(page 13\)](#)

4.1 Start task

Procedure

Qlik Sense performs the following procedure:

1. A request is sent to the Qlik Sense Repository Service (QRS).
2. The master Qlik Sense Scheduler Service (QSS) on the central node communicates with each slave QSS to see which ones are available to perform the task.
3. The task is given to the currently available slave QSS with the least load according to the load balancing rules.

Success

In case of success, log entries are written in the following files throughout the procedure:

- `<MachineName>_AuditActivity_Scheduler.txt`
- `<MachineName>_AuditActivity_Repository.txt`

Errors

Task not found

Description	Command=Start task;Result=5;ResultText=Error: Failed
Message	Task not found
Action	<p>Try the following measures:</p> <ul style="list-style-type: none"> • If the repository database is accessible, search the database for the task. • Check if the task has been successfully synchronized to the rim nodes. • Create a new, identical task and see if it works.

Scheduler is not licensed

Description	Command=Start task;Result=10;ResultText=Error: Failed
Message	Scheduler is not licensed. Not allowed to run Tasks
Action	Provide a valid Qlik Sense license.

Scheduler is not Master

Description	Command=Start task;Result=15;ResultText=Error: Failed
Message	Scheduler is not Master. Not allowed to run Tasks
Action	Make sure that the QSS on the central node is "Master" or "Master and Slave".

Found active session for Task

Description	Command=Start task;Result=20;ResultText=Error: Failed
Message	Found active session for Task. Task not started
Action	Try the following measures: <ul style="list-style-type: none">• Wait until the task has completed.• Stop the task.• Restart the QSS.

Task disabled

Description	Command=Start task;Result=25;ResultText=Error: Failed
Message	Task disabled
Action	Enable the task.

TaskExecutionSession already exists

Description	Command=Start task;Result=30;ResultText=Error: Failed
Message	TaskExecutionSession already exists for App <App.Name>
Action	Try the following measures: <ul style="list-style-type: none">• Wait until the task that is reloading the app has completed.• Stop the task that is reloading the app.• Restart the QSS.

App is not enabled

Description	Command=Start task;Result=35;ResultText=Error: Failed
Message	App <App.Name> is not enabled
Action	The app was not correctly enabled by the Migration Service. Try to re-migrate the app.

No slave nodes found to execute Task

Description	Command=Start task;Result=40;ResultText=Error: Failed
Message	No slave-nodes found to execute Task: <Task.Name>
Action	<p>Proceed as follows:</p> <ol style="list-style-type: none"> 1. Check the status of the slave nodes to determine if they can perform a reload. 2. Check that the app to reload exists on the slave nodes.

Unable to create TaskExecutionSession

Description	Command=Start task;Result=45;ResultText=Error: Failed
Message	Unable to create TaskExecutionSession
Action	Restart the QSS.

Unexpected exception when starting task

Description	Command=Start task;Result=50;ResultText=Error: Failed
Message	Unexpected exception when starting task. Exception message: <Exception.Message>
Action	Restart the QSS.

Unexpected exception when trying to start task

Description	Command=Start task;Result=55;ResultText=Error: Failed
Message	Unexpected exception when trying to start task. Exception message: <Exception.Message>
Action	Restart the QSS.

Max number of retries reached for task

Description	Command=Start task;Result=80;ResultText=Error: Failed
Message	Max number of retries (<Amount>) reached for task (id/name) <Task.ID>/<Task.Name>
Action	Restart the QSS.

Not possible to initiate retry

Description	Command=Start task;Result=90;ResultText=Error: Failed
Message	Not possible to initiate retry. TaskExecutionSession is null
Action	Restart the QSS.

4.2 Finish task

Procedure

Qlik Sense performs the following procedure:

- The Qlik Sense Scheduler Service (QSS) on the slave node that performed the task communicates the result to the master QSS.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Scheduler.txt*
- *<MachineName>_AuditActivity_Repository.txt*

Errors

Failed in TaskCompletedFail

Description	Command=Finished task;Result=60;ResultText=Error: Failed
Message	Failed in TaskCompletedFail. Forcing task to Error
Action	Check if the task has been successfully synchronized to the rim nodes.

Task finished with state <State>

Description	Command=Finished task;Result=65;ResultText=Error: Failed
Message	Task finished with state <State>
Action	Check if the task has been successfully synchronized to the rim nodes.

Failed to remove session

Description	Command=Finished task;Result=70;ResultText=Error: Failed
Message	Failed to remove session with Id <Session.ID> for TaskId <Task.ID>
Action	Restart the QSS.

Unexpected exception in TaskCompletedSuccess

Description	Command=Finished task;Result=50;ResultText=Error: Failed
Message	Unexpected exception in TaskCompletedSuccess. Exception message: <Message>
Action	Restart the QSS.

Unexpected exception in TaskCompletedFail

Description	Command=Finished task;Result=50;ResultText=Error: Failed
Message	Unexpected exception in TaskCompletedFail. Exception message: <Message>
Action	Restart the QSS.

Task failed

Description	Command=Finished task;Result=0;ResultText=Error: Failed
Message	Task failed
Action	Check the task for indications of why it failed. If the task is a reload task, check the engine logs and script logs for indications of why the reload failed.

4.3 Execute task

Procedure

Qlik Sense performs the following procedure:

- The Qlik Sense Scheduler Service (QSS) logs different states in the task execution chain during the execution of a task.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Scheduler.txt*

Errors

Trying to set task id to state

Description	Command=Task execution;Result=55;ResultText=Error: Failed
Message	Trying to set task (id) <Task.ID> to state <State>. Not allowed. No statechange will take place!

Action	Proceed as follows: <ol style="list-style-type: none"> 1. Re-run the task. 2. Restart the QSS.
---------------	--

TaskExecutionSession cannot be null

Description	Command=Task execution;Result=90;ResultText=Error: Failed
Message	TaskExecutionSession cannot be null. Unable to proceed with state change due to deleted task with proposed state: <State>
Action	Proceed as follows: <ol style="list-style-type: none"> 1. Re-run the task. 2. Restart the QSS.

Suppressed state change

Description	Command=Task execution;Result=85;ResultText=Error: Failed
Message	Suppressed state change to <State> for <Task.Name> with Id <Task.ID>
Action	No action is required. The state was suppressed when the task was stopped.

Unable to get <Task.Name>

Description	Command=Task execution;Result=5, 75;ResultText=Error: Failed
Message	Unable to get <Task.Name> with Id <Task.ID> for AuditActivityLog, only logging Id and Name for Task not for App.
Action	Proceed as follows: <ol style="list-style-type: none"> 1. Re-run the task. 2. Restart the QSS.

4.4 Start master

Procedure

Qlik Sense performs the following procedure:

- The master scheduler starts when the Qlik Sense Scheduler Service (QSS) starts on the central node.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_Service_Scheduler.txt*

4.5 Start slave

Procedure

Qlik Sense performs the following procedure:

- The slave scheduler starts when the Qlik Sense Scheduler Service (QSS) starts.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_Service_Scheduler.txt*

4.6 Resume master

Procedure

Qlik Sense performs the following procedure:

- Resume a paused master Qlik Sense Scheduler Service (QSS).

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_Service_Scheduler.txt*

4.7 Resume slave

Procedure

Qlik Sense performs the following procedure:

- Resume a paused slave Qlik Sense Scheduler Service (QSS).

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_Service_Scheduler.txt*

4.8 Read initial settings

Procedure

Qlik Sense performs the following procedure:

- Log the initial settings when the Qlik Sense Scheduler Service (QSS) starts.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_Service_Scheduler.txt*

4.9 Log hardware information at the startup of the service

Procedure

Qlik Sense performs the following procedure:

- Log hardware information during the startup of the Qlik Sense Scheduler Service (QSS).

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_Service_Scheduler.txt*

Errors

Failed when logging hardware information

Description	Command=Start scheduler;Result=50;ResultText=Error: Failed
Message	Failed when logging hardware information
Action	Restart the QSS.

4.10 Stop master

Procedure

Qlik Sense performs the following procedure:

- The master scheduler stops when the Qlik Sense Scheduler Service (QSS) shuts down on the central node.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_Service_Scheduler.txt*

4.11 Stop slave

Procedure

Qlik Sense performs the following procedure:

- The slave scheduler stops when the Qlik Sense Scheduler Service (QSS) shuts down.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_Service_Scheduler.txt*

4.12 Pause master

Procedure

Qlik Sense performs the following procedure:

- Pause a running master Qlik Sense Scheduler Service (QSS).

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_Service_Scheduler.txt*

4.13 Pause slave

Procedure

Qlik Sense performs the following procedure:

- Pause a running slave Qlik Sense Scheduler Service (QSS).

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_Service_Scheduler.txt*

4.14 Settings change for slave

Procedure

Qlik Sense performs the following procedure:

- Log the change of settings for a slave Qlik Sense Scheduler Service (QSS).

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Scheduler.txt*

5 Qlik Sense Engine Service

This section describes how to use the Qlik Sense logs to troubleshoot problems related to the Qlik Sense Engine Service (QES).

See also:

- ▢ [Introduction \(page 13\)](#)

5.1 Open app

Procedure

Qlik Sense performs the following procedure:

1. Check if the app is already open.
2. If not, load the QVF file.
3. Read the app properties from the file.
4. Read the script from the file.
5. Open the app.

Success

In case of success, log entries are written in the following files throughout the procedure:

- `<MachineName>_AuditActivity_Engine.txt`

Errors

Already opened in different mode

Description	Command=Open app;Result=1009;ResultText=Error: App already open in different mode
Message	<Appld>
Action	Make sure that the app is not opened without data when you try to open it with data or vice versa.

Already opened

Description	Command=Open app;Result=1002;ResultText=Error: App already opened
Message	<Appld>
Action	Make sure that the app is not already open.

Invalid path

Description	Command=Open app;Result=4;ResultText=Error: Invalid Path
Message	Malformed parameters
Action	Make sure that the app name, password, user name, and serial number are UTF-8 encoded strings.

Resource not found

Description	Command=Open app;Result=9003;ResultText=Error: Resource not found
Message	QVF header error (read)
Action	Probable cause of error: The app has been deleted from disk. Try to import the app again.

5.2 Create app

Procedure

Qlik Sense performs the following procedure:

1. Check if the app name is valid.
2. Create the app.
3. Save the QVF file.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Engine.txt*

Errors

Invalid name

Description	Command=Create app;Result=3001;ResultText=Error: App save failed
Message	<AppName>
Action	Make sure that the app name does not contain any of the following: <ul style="list-style-type: none"> • Special characters: <>:\"/ ?* • Device names that are reserved in Microsoft Windows (for example, "COM1" and "LPT1")

5.3 Delete app

Procedure

Qlik Sense performs the following procedure:

1. Unload all app instances for the app.
2. Delete the QVF file.
3. Request the Qlik Sense Repository Service (QRS) to delete the file.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Engine.txt*

Errors

Delete failed

Description	Command=Delete app;Result=9002;ResultText=Error: Delete failed
Message	<REST client response error>
Action	As there is no response from the QRS, check if it has stopped running or if it cannot process requests.

5.4 Export app

Procedure

Qlik Sense performs the following procedure:

1. Open the QVF file.
2. Copy to a new QVF file.
3. Export the app contents.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Engine.txt*

Errors

Disk is full

Description	Command=Export app;Result=9009;ResultText=Error: The disk it out of space
Message	CopyQvfFile
Action	Free up some space on the disk.

File corrupt

Description	Command=Export app;Result=3002;ResultText=Error: File corrupted
Message	CopyQvfFile
Action	Make sure that the file that is exported to can be written to.

5.5 Import app

Procedure

Qlik Sense performs the following procedure:

1. Check if the app can be migrated to the current version of Qlik Sense (if needed).
2. Open a copy of the app to import.
3. Remove unwanted objects (such as connections) from the copy.
4. Request the Qlik Sense Repository Service (QRS) to save the app.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Engine.txt*

Errors

File corrupt

Description	Command=Import app;Result=9008;ResultText=Error: App was created in a newer product version
Message	Migration
Action	Make sure that the app was not created in a newer version of Qlik Sense.

Internal error

Description	Command=Import app;Result=10;ResultText=Error: Internal error
Message	Could not import file (ImportApp)
Action	During import, the QVF file is copied to the local <i>VApp</i> folder first. If this fails, the folder is probably write-protected. Therefore check for any write-protection on the folder.

5.6 Reload app

Procedure

Qlik Sense performs the following procedure:

- Execute the reload script.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Engine.txt*

5.7 Duplicate app

Procedure

Qlik Sense performs the following procedure:

1. Compare the app IDs of the source and target apps.
2. Request the Qlik Sense Repository Service (QRS) to open the app.
3. Copy the QVF file.
4. Import the copy of the file.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Engine.txt*

Errors

Delete failed

Description	Command=Duplicate app;Result=8;ResultText=Error: Invalid parameters
--------------------	---

Message	Could not copy file. SrcAppId and TargetAppId are identical
Action	Make sure that the source and target IDs are not identical.

5.8 Publish app

Procedure

Qlik Sense performs the following procedure:

1. Check if the app is a session app. If so, publishing is not supported. A session app only exists in the Qlik Sense Engine Service (QES) memory and is not linked to any file.
2. Save the object in the repository database.
3. Request the Qlik Sense Repository Service (QRS) to publish the app.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Engine.txt*

Errors

Access denied

Description	Command=Publish app;Result=8;ResultText=Error: Access Denied
Message	Could not save objects
Action	Check if the disk is write-protected as the publish operation could not flush the objects needed before publishing the app.

REST connection failure

Description	Command=Publish app;Result=18;ResultText=Error: Rest connection failed
Message	REST client response error
Action	As there is no response from the QRS, check if it has stopped running or if it cannot process requests.

5.9 Unpublish app

Procedure

Qlik Sense performs the following procedure:

1. Check if the app is a session app. If so, publishing is not supported. A session app only exists in the Qlik Sense Engine Service (QES) memory and is not linked to any file.
2. Save the object in the repository database.
3. Request the Qlik Sense Repository Service (QRS) to unpublish the app.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Engine.txt*

Errors

Access denied

Description	Command=Unpublish app;Result=8;ResultText=Error: Access Denied
Message	Could not save objects
Action	Check if the disk is write-protected as the publish operation could not flush the objects needed before unpublishing the app.

REST connection failure

Description	Command=Unpublish app;Result=18;ResultText=Error: Rest connection failed
Message	REST client response error
Action	As there is no response from the QRS, check if it has stopped running or if it cannot process requests.

5.10 Replace app

Qlik Sense performs the following procedure:

1. A request is sent to the Qlik Sense Repository Service (QRS).
2. The QRS updates the repository database.
3. The QRS contacts the Qlik Sense Engine Service (QES).

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_AuditActivity_Repository.txt*
- *<MachineName>_AuditActivity_Engine.txt*

5.11 Start engine

Procedure

Qlik Sense performs the following procedure:

1. The Qlik Sense Engine Service (QES) sets the current execution mode (that is, server or desktop).
2. The QES parses the command line arguments.
3. The QES configures the logging.
4. The QES detects the operating system and version.
5. The QES initializes the collate and memory structures.
6. The QES sets the termination handlers, starts the internal threads, and checks the license.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_Service_Engine.txt*

Errors

Registration failure

Description	Command=Start engine;Result=-1;ResultText=Error:Server crash
Message	Server crashed on registration
Action	<p>Proceed as follows:</p> <ol style="list-style-type: none"> 1. Check the status of the Migration Service. 2. Check that the default <i>\App</i> folder exists. 3. Check the status of the network controller. 4. Check the status of the network port.

5.12 Stop engine

Procedure

Qlik Sense performs the following procedure:

1. The Qlik Sense Engine Service (QES) stops the cache trimmer.
2. The QES clears the cache.

3. The QES terminates the process handlers.
4. The QES exits the threads.

Success

In case of success, log entries are written in the following files throughout the procedure:

- *<MachineName>_Service_Engine.txt*

Errors

Cache trimmer never stopped

Description	Command=Stop engine;Result=-1;ResultText=Warning
Message	Cache Trimmer never stopped
Action	Contact Qlik support.