

Deploy Qlik Sense Enterprise on Windows

Qlik Sense®
February 2024
Copyright © 1993-2024 QlikTech International AB. All rights reserved.



1 Planning your Qlik Sense Enterprise deployment	9
1.1 Qlik product licenses	9
Product activation	9
Unified license	9
Qlik Sense Enterprise	10
Qlik NPrinting	10
Qlik Sense licenses	11
License Enabler File	13
Access assignment	14
1.2 Downloading installation files	17
1.3 Before you install Qlik Sense Enterprise on Windows	18
System requirements for Qlik Sense Enterprise	18
Supported browsers	26
Qlik Sense Enterprise on Windows architecture	28
Performance	62
User accounts	63
1.4 Qlik Sense Enterprise deployment examples	65
Qlik Sense Enterprise on Windows deployments	65
Qlik Sense Enterprise SaaS deployments	66
Qlik Sense Enterprise on Windows multi-cloud deployments	66
Qlik Sense Enterprise on Windows on-premises	66
Qlik Sense Enterprise on Windows deployed to AWS	72
Qlik Sense Enterprise on Windows deployed to Azure	76
Qlik Sense Enterprise on Windows deployed to Google Cloud	79
Qlik Sense Enterprise on Windows deployed to Oracle Cloud	92
Deploying Qlik Sense Enterprise in a multi-cloud environment	94
Distributing apps from Qlik Sense Enterprise on Windows to Qlik Sense Enterprise SaaS	96
2 Installing Qlik Sense Enterprise on Windows	98
2.1 Installing Qlik Sense Enterprise on Windows on a single node	98
Pre-installation	98
Preparing the server	98
Installing Qlik Sense Enterprise on Windows	99
Licensing Qlik Sense	105
Allocating access to users	105
Additional configuration	106
Modifying an object bundles installation	106
Common issues and solutions to problems with your installation	107
2.2 Installing Qlik Sense in a multi-node site	108
Pre-installation	109
Preparing the server	109
Installing the Qlik Sense central node	110
Configuring PostgreSQL multi-node connections	117
Licensing Qlik Sense	117
Allocating access to users	118
Installing a Qlik Sense rim node	119
Connecting and configuring the nodes	122
Additional configuration	124

Modifying an object bundles installation	124
Common issues and solutions to problems with your installation	125
2.3 Creating a file share	126
Creating an NFS file share	127
Changing the file share path	128
2.4 Configuring failover for central node resiliency	129
Failover considerations	129
Create a failover candidate node	130
Manually migrating the central node	132
2.5 Installing and configuring PostgreSQL	132
Databases	132
Installing PostgreSQL	133
Creating a PostgreSQL database	134
Creating login roles	134
Granting permissions for PostgreSQL versions 15 and 16	135
Configuring PostgreSQL	137
2.6 Installing or upgrading PostgreSQL using the Qlik PostgreSQL Installer	139
About the Qlik PostgreSQL Installer	139
Supported scenarios	139
Considerations	140
Installing a new Qlik Sense Repository Database using PostgreSQL 14.8	140
Upgrading an existing Qlik Sense Repository Database using PostgreSQL 14.8 - Central node ..	141
Silently installing or upgrading PostgreSQL using the Qlik PostgreSQL Installer	142
2.7 Installing and configuring PostgreSQL on Azure	147
Databases	147
Setting up a PostgreSQL database in Azure	148
Connecting to the database using psql	148
Installing Qlik Sense	149
2.8 Configuring a proxy for Qlik ADS and HDS communication with Qlik Sense Enterprise SaaS ..	149
2.9 Configuring a proxy for Qlik License Service communication in Qlik Sense Enterprise on	
Windows	151
2.10 Configuring preferred cipher suites for Qlik License Service in Qlik Sense Enterprise on	
Windows	152
2.11 Changing the user account to run Qlik Sense services	153
Using an account without administrator privileges to run the Qlik Sense services during the	
installation of a node	154
Changing the user account type to run the Qlik Sense services on an existing site	155
Changing the Qlik Sense services account password	156
2.12 Uninstalling Qlik Sense Enterprise on Windows	156
2.13 Integrating Qlik Catalog with Qlik Sense Enterprise	157
3 Upgrading Qlik Sense Enterprise on Windows	158
3.1 Patching instead of upgrading	158
3.2 Repairing instead of upgrading	158
3.3 Planning your upgrade	158
Planning your upgrade path	158
Considerations about older versions	159

Considerations about multi-node deployments	161
Considerations about logging	161
Considerations about custom configurations	161
Consideration about your Qlik Sense Repository Database	161
3.4 Running the upgrade application	162
Before you start the upgrade	162
Upgrading a Qlik Sense node	163
3.5 Configuring your node after upgrading	165
3.6 Upgrading after uninstalling Qlik Sense	165
3.7 Patching Qlik Sense	166
3.8 Repairing an installation	167
3.9 Troubleshooting your upgrade	168
4 Running the installer silently	171
4.1 Silent installing	171
Syntax	171
Commands	172
Arguments	172
Shared persistence configuration file syntax	174
Deprecated command line arguments	176
4.2 Silent upgrading	176
Syntax	177
Commands	177
Arguments	178
Deprecated command line arguments	178
4.3 Silent repairing	179
4.4 Silent patching	179
Commands	180
Recreating root certificates	180
Example	180
5 Backup and restore Qlik Sense Enterprise on Windows	181
5.1 Qlik Sense certificates	181
5.2 Qlik Sense Repository Database	181
5.3 Shared persistence file share	182
5.4 Backing up certificates	182
5.5 Restoring certificates	191
5.6 Backing up a Qlik Sense site	201
Backing up the Qlik Sense Repository Database after uninstalling Qlik Sense	202
5.7 Restoring a Qlik Sense site	203
Restoring a Qlik Sense site to a machine with the same hostname	203
Restoring a Qlik Sense site to a machine with a different hostname	205
6 Qlik Sense Enterprise on Windows security	208
6.1 Certificates	208
Certificate revocation check	209
Certificates used by the Qlik Sense Proxy Service	209
6.2 Protecting the platform	212
Network security	212

Server security	213
Process security	214
App security	215
6.3 Authentication	216
Default authentication module	217
Certificate trust	217
Authentication solutions	221
6.4 Authorization	229
Access control	230
6.5 Auditing	232
6.6 Confidentiality	233
6.7 Integrity	233
Database security	233
Data encryption	245
6.8 Availability	247
6.9 Security example: Opening an app	247
6.10 AWS and Azure security	248
Qlik Sense	248
AWS	249
Azure	249
7 Logging	251
7.1 Updated logging framework	251
7.2 Legacy logging framework	251
7.3 Requirements	251
Securing the file system	251
Synchronizing time	251
Setting time zone	251
7.4 Storage	252
Log folder	252
Archived log files	256
7.5 Naming	257
7.6 Rows	257
7.7 Fields	258
Audit activity log	258
Audit security log	261
Server log	265
Qlik Sense engine service log fields	269
7.8 Trace logs	269
Storage	269
Naming	269
Rows	270
Fields	270
7.9 Configuring the logging	281
Appenders	281
7.10 Telemetry logging	288
Enabling telemetry logging	288

Parameter descriptions	289
Reading the logs	289
Important Engine Operations	290
8 Troubleshooting - Deployment	292
8.1 Understand the problem	292
8.2 Use the log files	293
Default log files	293
Archived log files	294
8.3 Qlik Sense client or application problems	294
8.4 Other resources	294
8.5 Cannot find the password for the PostgreSQL database superuser	294
8.6 Cannot find the password for the qliksenserepository database user	295
8.7 Cannot access the hub or the QMC after installation	295
8.8 Error message "No access path" after upgrade	295
8.9 One or more Qlik Sense services did not start after installation	296
8.10 Anti-virus software scanning affects performance	297
8.11 Exit codes	297
8.12 Rim node loses connection to the central node	298
8.13 Repository cannot connect to database after installation	299
8.14 Unable to upgrade Qlik Sense, missing database	299
8.15 Troubleshooting - database not configured for IP address or range	299
8.16 Troubleshooting app distribution in multi-cloud	299
Publishing is a little slow	300
A temporary error occurred	300
An unknown error occurred	300
8.17 Cannot read or write to the logging database	300
8.18 Failed to remove soft deleted records	301
8.19 Issues with Qlik Sense Enterprise when not connected to the internet	305
8.20 The Qlik Sense Client-Managed Mobile app encounters a network error and must close	305
9 Deploying Qlik Sense Client-Managed Mobile	306
9.1 The Qlik Sense Client-Managed Mobile app	306
9.2 Enterprise Mobile Management (EMM) and Qlik Sense Client-Managed Mobile	306
9.3 Qlik Sense Client-Managed Mobile security	307
Authentication	307
Certificates	307
Configuring the certificate validation policy for the Qlik Sense Client-Managed Mobile app	308
9.4 Installing Qlik Sense Client-Managed Mobile	308
Qlik Sense Client-Managed Mobile and VPP	309
Deploying the Qlik Sense Client-Managed Mobile app using AirWatch	309
Deploying the Qlik Sense Client-Managed Mobile app using MobileIron	313
Connecting to Qlik Sense Client-Managed Mobile using MobileIron	319
Deploying Qlik Sense Client-Managed Mobile with Microsoft Azure and Intune	320
Connecting to Qlik Sense from the Qlik Sense Client-Managed Mobile app	325
9.5 Deploying mashups to the Qlik Sense Client-Managed Mobile app	326
Why use mashups in the Qlik Sense Client-Managed Mobile app	327

Restricting access to mashups in the Qlik Sense Client-Managed Mobile app	327
9.6 Customizing Qlik Sense Client-Managed Mobile with AppConfig	328
Configurable settings in AppConfig	328
Setting a mashup as landing page	330

1 Planning your Qlik Sense Enterprise deployment

To successfully plan and prepare for your Qlik Sense deployment, do the following:

Introducing Qlik Sense Enterprise

Get a brief introduction to Qlik Sense Enterprise.

Qlik Sense Enterprise deployment examples

See examples of different ways to deploy Qlik Sense Enterprise.

System requirements for Qlik Sense Enterprise

Review the Qlik Sense Enterprise system requirements.

Qlik product licenses

Understand how Qlik Sense uses license keys and LEF for site licensing.

Understand how Qlik Sense uses tokens for user access allocation (token-based licensing).

Ensure that you have your Qlik Sense license key available.

1.1 Qlik product licenses

Here is a summary of the license options that are available for the different Qlik Sense related products. Licensing allows you to manage the usage of the Qlik Sense software in your organization.

For detailed information on Qlik Sense licensing options, see Qlik's legal terms, product terms, and Licensing Service Reference Guide:

-  [Qlik Legal Terms](#)
-  [Qlik Product Terms](#)
-  [Qlik Licensing Service Reference Guide](#)

Product activation

All Qlik products are entitled and enforced by a License Enabler File (LEF). The LEF is the artifact that is downloaded during the production activation. A Qlik product is licensed and activated using either a serial and control number, or a signed license key. The use of a signed license key is required for Qlik Sense Enterprise SaaS deployments, and for the use of capacity based licenses.

With a signed license key, you need internet access (direct or through a proxy) to access the cloud-based license backend, for user assignments, analytic time consumption, and product activations.



*With a signed license key, license information can be viewed in the QMC after the license key is entered and saved using **Apply**.*

Unified license

You can use a unified license in multiple deployments. A unified license shares the same signed key between:

1 Planning your Qlik Sense Enterprise deployment

- multiple Qlik Sense Enterprise deployments
- multiple QlikView Server deployments
- QlikView Server and Qlik Sense Enterprise deployments

Applying the same signed key to multiple deployments lets you share the same users and access types. Users can access all connected deployments using the same Professional or Analyzer access allocation.

Qlik Sense Enterprise

Qlik Sense Enterprise is the server version of Qlik Sense that you can deploy on a single node, or on multiple nodes. Qlik Sense Enterprise deployments are licensed and activated using a serial and control number, or a signed license key. Your Qlik Sense Enterprise license is based either on access types, or on tokens.

Access types

Access types licenses are the Professional and Analyzer Users licenses (user-based) and Analyzer Capacity licenses (capacity-based). You can combine these for a subscription based license if you use the signed license key when your deployment is activated. You can combine only user-based licenses if you are using a perpetual license.



After changing to a license with a signed key, you cannot return to using the old LEF based license model.

Token

You use tokens to allocate access passes to users so that they can access Qlik Sense. The LEF determines the number of tokens that you can allocate to different access passes. A user without an access pass cannot access apps.

With a Qlik Sense Token license you use tokens to allocate access passes to users. You can allocate user access and login access.



The token license is available only to customers with existing Qlik Sense Token licenses.

Core-based site

Qlik Sense Enterprise core-based sites are licensed based on the number of CPU cores on which the software will operate. A Core means a single processing unit within a processor or CPU, whether physical or virtual, including a vCPU or virtual core, which is capable of executing a single software thread at a time.

Qlik NPrinting

You can install and configure Qlik NPrinting to connect to QlikView documents or Qlik Sense apps. The licensing requirements and procedures are different depending on if you connect Qlik NPrinting to QlikView or Qlik Sense.

Qlik NPrinting versions 16.0.0.0 and later are licensed by a LEF. If you are using an earlier version of Qlik NPrinting, we suggest that you upgrade to Qlik NPrinting versions 16.0.0.0 or later.

1 Planning your Qlik Sense Enterprise deployment



A Qlik Sense token is not required for the Qlik NPrinting service account. However, because you often perform troubleshooting within the Qlik NPrinting service account, it is helpful to assign a token to the Qlik NPrinting service account so that it has access to the Qlik Sense hub.

Qlik Sense licenses

Qlik Sense Enterprise is the server version of Qlik Sense that you can deploy on a single node, or on multiple nodes. Qlik Sense Enterprise licenses are based either on access types, or on tokens.

For detailed information on Qlik Sense licensing options, see Qlik's legal terms, product terms, and Licensing Service Reference Guide:

[Qlik Legal Terms](#)

[Qlik Product Terms](#)

[Qlik Licensing Service Reference Guide](#)

Unified license

You can use a unified license in multiple deployments. A unified license shares the same signed key between:

- multiple Qlik Sense Enterprise deployments
- multiple QlikView Server deployments
- QlikView Server and Qlik Sense Enterprise deployments

Applying the same signed key to multiple deployments lets you share the same users and access types. Users can access all connected deployments using the same Professional or Analyzer access allocation.

Qlik Sense Enterprise

A Qlik Sense Enterprise deployment can be licensed using two different models: the serial and control number and the signed license key. The License Enabler File (LEF) defines the terms of your license and the access types that you can allocate to users. Your Qlik Sense Enterprise license is based either on access types, or on tokens. A core-based license is also available.

With a signed license key, you need internet access (direct or through a proxy) to access the cloud-based license backend, for user assignments, analytic time consumption, and product activations.

User-based and capacity-based licenses

A user-based license grants a predefined number of access allocations that can be assigned to unique and identified users. In Qlik Sense Enterprise, user-based licenses are either Professional and Analyzer Users licenses, or User access passes allocated with a Token license.

A capacity-based license grants a predefined number of time allocations for accessing Qlik Sense Enterprise that can be used by identified or anonymous users. In Qlik Sense, capacity-based licenses are either based on Analyzer Capacity access, or Login access pass allocated with a Token license.

1 Planning your Qlik Sense Enterprise deployment

Access types

Access types licenses are the Professional and Analyzer Users licenses (user-based) and Analyzer Capacity licenses (capacity-based). You can combine these for a subscription based license if you use the signed license key when your deployment is activated. You can combine only user-based licenses if you are using a perpetual license.



After changing to a license with a signed key, you cannot return to using the old serial and control number license model.

Professional and Analyzer Users license

A Professional and Analyzer Users license is composed of Professional and Analyzer access types.

- Professional access (user-based) is allocated to an identified user to allow the user to access streams and apps within a Qlik Sense site. The professional access is intended for users who need access to all features in a Qlik Sense installation. A user with professional access can create, edit, and publish sheets or apps, and make full use of the available features, including administration of a Qlik Sense site.
- Analyzer access is allocated to an identified user to allow the user to access streams and apps in the hub. The analyzer access is intended for users who consume sheets and apps created by others. A user with analyzer access cannot create, edit, or publish sheets or apps, but can create and publish stories, bookmarks and snapshots based on data in apps. The user can also print objects, and export data from an object to Excel.

Analyzer Capacity license

An Analyzer Capacity license is composed of Analyzer Capacity access type.

Analyzer capacity is a consumption-based license type, which is like analyzer access regarding available features. Users can access streams and apps in the hub and consume sheets and apps created by others. Analyzer capacity access allows users to create stories, bookmarks, and snapshots based on data in apps. Creating, editing, or publishing sheets or apps is not possible.

With an analyzer capacity license, you subscribe to analyzer time, a defined number of minutes per month (calendar date). These minutes are shared between users and can be consumed by anyone who is part of the user group, including anonymous users. Consumption is measured in units of six minutes. For each new six-minute period, a unit is consumed.

Token

You use tokens to allocate access passes to users so that they can access Qlik Sense. The License Enabler File (LEF) determines the number of tokens that you can allocate to different access passes. A user without an access pass cannot access apps.



The token license is available only to customers with existing Qlik Sense Token licenses.

There are two types of access passes that can be allocated using tokens:

1 Planning your Qlik Sense Enterprise deployment

- User access pass (user-based) is assigned to unique and identified users allowing them unlimited access to apps, streams, and other resources.
- Login access pass (capacity-based) allocates a block of passes to a group for infrequent or anonymous access. Allows full access for a limited period.

When you allocate tokens, the number of available tokens is reduced. Each access type costs a certain number of tokens, and if the token balance is zero or insufficient, you cannot allocate more to the access types. You can free up tokens and choose to use the tokens differently. The number of tokens for the Qlik Sense site can be increased or decreased by activating a new license.

Core-based site

Qlik Sense Enterprise core-based sites are licensed based on the number of CPU cores on which the software will operate. The license is administered using a License Enabler File (LEF), which limits the maximum number of cores on which the Qlik associative engine and its components may operate. A Core means a single processing unit within a processor or CPU, whether physical or virtual, including a vCPU or virtual core, which can execute a single software thread at a time.

License Enabler File

In Qlik Sense there are two alternative license models: the serial and control number and the signed license key. The License Enabler File (LEF) defines the terms of your license and the access types that you can allocate to users.

When licensing Qlik Sense using a serial and control number, the LEF can be downloaded when the serial number and the control number have been entered in the Qlik Management Console (QMC). The LEF can also be pasted directly into the QMC, if, for example, no network connection is available. There are two license types that can be activated using a serial and control number: Professional and Analyzer Users licenses, and Qlik Token licenses.

When licensing Qlik Sense using a signed key, the LEF file is stored in the License Backend.



If you want to set up Qlik Sense Enterprise SaaS, please contact your Qlik representative or Qlik Support to obtain a valid license for the setup.

Professional and Analyzer Users license

Professional and Analyzer Users licenses grant a predefined number Professional and Analyzer (user-based) access type allocations. The LEF file determines the allocation of the access types.



Analyzer Capacity licenses (capacity based) can only be licensed using a signed key. When combining professional, analyzer, and analyzer capacity access types in the same Qlik Sense Enterprise installation, you must license it using a signed key.

1 Planning your Qlik Sense Enterprise deployment

Token license

You use tokens to allocate access passes to users so that they can access Qlik Sense. The License Enabler File (LEF) determines the number of tokens that you can allocate, and holds the number of tokens available for the central node in a site. This means that a Qlik Sense site needs at least one (1) LEF. A user without an access pass cannot access apps.



The token license is available only to customers with existing Qlik Sense Token licenses.



You cannot use QlikView CAL-based licenses with Qlik Sense as the tokens are not compatible with the Client Access Licenses (CALs) used in QlikView.

Increase in tokens

When the number of tokens in the LEF increases (for example, when buying additional tokens), the new tokens are added to the pool of unallocated tokens that can be used to allocate access passes that allow users to access Qlik Sense.

Decrease in tokens

When the number of tokens in the LEF decreases, the following happens:

1. Unallocated tokens are removed.
2. If step 1 is not enough to meet the decreased number of tokens in the LEF, any tokens that are freed up by removal of access passes cannot be used for new allocations until the number of allocated tokens is below the new number set in the LEF.

Access assignment

Qlik Sense Enterprise licenses are based either on access types, or on tokens. Depending on your license, you can allocate either access types or access passes to users, to allow them to access Qlik Sense.

- Access types licenses are the Professional and Analyzer Users licenses (user-based) and Analyzer Capacity licenses (capacity-based).
With a Professional and Analyzer Users license you can allocate professional access and analyzer access.
With an Analyzer Capacity license you can allocate analyzer capacity access, where consumption is time based (analyzer time).
- With a Qlik Sense Token license you use tokens to allocate access passes to users. You can allocate user access and login access.

Access types

Professional and Analyzer Users licenses and Analyzer Capacity licenses grant a predefined number of access allocations. The License Enabler File (LEF) defines the terms of your license and the access types that you can allocate to users. You can combine these for a subscription based license if you use the signed license key

1 Planning your Qlik Sense Enterprise deployment

when your deployment is activated. You can combine only user-based licenses if you are using a perpetual license. You must use a license with a signed key if you are licensing analyzer capacity access.

Professional access

Professional access is allocated to an identified user to allow the user to access streams and apps within a Qlik Sense site. The professional access is intended for users who need access to all features in a Qlik Sense installation. A user with professional access can create, edit, and publish sheets or apps, and make full use of the available features, including administration of a Qlik Sense site.

For Qlik Sense installations licensed with a serial and control number, if you remove professional access allocation from a user, the access type is put in quarantine, if it has been used within the last seven days. If it has not been used within the last seven days, the professional access is released immediately. You can reinstate quarantined professional access, to the same user, within seven days.

The maximum number of parallel user connections for a single user of this type of access pass is five (5). If you use a license with a signed license key, accessing the QMC also counts and adds to the maximum number of parallel sessions, which is five. To avoid unnecessary session consumption, the root admin should not be allocated any type of access.

When a user with the maximum number of parallel user connections ends a connection (for example, by logging out) five minutes must pass before the user can use the access pass to add another connection (for example, by logging in).

Analyzer access

Analyzer access is allocated to an identified user to allow the user to access streams and apps in the hub. The analyzer access is intended for users who consume sheets and apps created by others. A user with analyzer access cannot create, edit, or publish sheets or apps, but can create and publish stories, bookmarks and snapshots based on data in apps. The user can also create bookmarks, print objects, stories, and sheets, and export data from an object to Excel.

For Qlik Sense installations licensed with a serial and control number, if you remove analyzer access allocation from a user, the access type is put in quarantine, if it has been used within the last seven days. If it has not been used within the last seven days, the analyzer access is released immediately. You can reinstate quarantined analyzer access, to the same user, within seven days.

The maximum number of parallel user connections for a single user of this type of access pass is five (5). When a user with the maximum number of parallel user connections ends a connection (for example, by logging out) five minutes must pass before the user can use the access pass to add another connection (for example, by logging in).

Analyzer capacity access

Analyzer capacity is a consumption-based license type, which is like analyzer access regarding available features. Users can access streams and apps in the hub and consume sheets and apps created by others. Analyzer capacity access allows users to create stories, bookmarks, and snapshots based on data in apps. Creating, editing, or publishing sheets or apps is not possible.

1 Planning your Qlik Sense Enterprise deployment

With an analyzer capacity license, you subscribe to analyzer time, a defined number of minutes per month (calendar date). These minutes are shared between users and can be consumed by anyone who is part of the user group, including anonymous users. Consumption is measured in units of six minutes. For each new six-minute period, a unit is consumed.

Access passes

With a Qlik Sense Token license you use tokens to allocate access passes to users. The License Enabler File (LEF) determines the number of tokens that you can allocate to different access passes. A user without an access pass cannot access apps.

User access pass

This type of access pass allows a unique and identified user to access the hub.

The access pass is valid within an entire Qlik Sense site. For example, if a user first connects to a node in the USA and then, at a later stage, connects to a node in the UK, the user consumes the same access pass, if the two nodes are connected to the same central node.

The maximum number of parallel user connections for a single user of this type of access pass is five (5). When a user with the maximum number of parallel user connections ends a connection (for example, by logging out) five minutes must pass before the user can use the access pass to add another connection (for example, by logging in).

One (1) token corresponds to one (1) access pass. The access passes are allocated using the Qlik Management Console (QMC).



You can have both a user access pass and the possibility to consume login access passes. If you have five active sessions, opening an additional session will consume from your login access passes.

Removing user access pass allocation

When a user access pass is removed, it enters a quarantine for seven (7) days, counting from the last time that the access pass was used. For example, if the access pass is used on January 10, the tokens used to allocate the access pass are not available for new allocations until January 18. During the quarantine period, the original allocation of the access pass can be reinstated, which means that the quarantine period ends and the user can start using the access pass again.

Login access pass

This type of access pass allows an identified or anonymous user to access the hub for a maximum of 60 continuous minutes per 28-day period. If the user exceeds the 60 minutes time limitation, the user connection does not time out. Instead, another login access pass is used. If no more login access passes are available, the user connection is discontinued.

- If an identified user is disconnected, the user can re-connect and continue to use the same access pass, if re-connecting within the 60 minutes.
- If an anonymous user is disconnected, the user gets a new access pass when re-connecting.

1 Planning your Qlik Sense Enterprise deployment

The login access pass tracks the number of logins and runs over 28 days. For example, if 1000 logins are assigned to Group A, the users in Group A can use 1000 logins over 28 days. If 100 logins are consumed on Day 1, the 100 logins are available again on Day 29.

The maximum number of parallel user connections for a single user of this type of access pass is five (5). Note that this only applies to identified users. An anonymous user can only have one (1) user connection. When a user with the maximum number of parallel user connections ends a connection (for example, by logging out) five minutes must pass before the user can use the access pass to add another connection (for example, by logging in). However, a user can have more connections than allowed by a single access pass by consuming additional access passes.

One (1) token corresponds to ten (10) access passes. The access passes are allocated using login access groups in the QMC.



App reloads will extend the session and consume access passes also when the app is not actively used. If a browser page is open with an app, app reloads will result in additional access pass consumption.

Removing login access pass allocation

When a login access group is removed, the tokens used to allocate the access pass become available in accordance to the following procedure:


1. For every ten (10) **unused** login access passes, one (1) token is freed up.
2. For every ten (10) login access passes that leave the **used** state after the period specified in the *Login access pass* (page 16) section above has passed, one (1) token is freed up.

See also:

1.2 Downloading installation files

The Qlik Download Site provides the files you need to install and upgrade Qlik products. You can find the site in Qlik Community under Support > Product News > Product Downloads.

Do the following:

1. Go to  [Product Downloads](#).
2. Select **Qlik Data Analytics** or **Qlik Data Integration**, and then select your product.
3. Use the filters to narrow your list of possible downloads.
4. Click a link in the **Download Link** column in the **Download Assets** table to start the download.

1 Planning your Qlik Sense Enterprise deployment

Example from the **Download Site** where the files have been filtered on product and release.

The screenshot shows the Qlik Sense Download Site interface. At the top, there are navigation buttons: "Qlik Data Analytics", "Qlik Data Integration", "Search All", "Clear Filter", and "Feedback?". Below these are four filter sections: "Visibility" (with options: "All releases and all patches", "Latest release and patch", "All releases with latest patch"), "Product" (with a dropdown menu showing "Qlik Sense Enterprise on Windows" selected and other options like "Qlik Alerting", "Qlik Connector for use with SAP NetWea...", "Qlik Data Transfer", "Qlik Essbase Connector", "Qlik GeoAnalytics Data Package"), "Release" (with a dropdown menu showing "November 2022" selected and other months from "October 2022" to "April 2022"), and "Release Number" (with a dropdown menu showing "Initial Release" selected and other patches from "Patch 17" to "Patch 13"). Below the filters is a "Downloads" section with a table of download links.

Product	Release	Num...	Version	Publish...	End of Supp...	Release N...	Size	Download Link
Qlik Sense Enterpris...	November 20...	Initial Rele...	v14.97.3	08 Nov 2022	https://community...	Release Notes	1,282 MB	Qlik_Sense_setup.exe

1.3 Before you install Qlik Sense Enterprise on Windows

To successfully plan and prepare for your Qlik Sense deployment, do the following:

System requirements

Check that your environment fulfills the system requirements.

Ports

Check that the required ports are available on your system.

Supported browsers

Check that your browsers are supported.

Architecture

Understand the Qlik Sense Enterprise on Windows architecture, and the different node types.

Security

Understand how Qlik Sense Enterprise on Windows uses certificates for security. Certificates are installed by default.

Performance

Basic information on performance to consider before you install Qlik Sense Enterprise on Windows.

User accounts

Understand and set up the various user accounts required to install and run the Qlik Sense Enterprise on Windows services.

If you intend to run Qlik Sense Enterprise on Windows services as a user without administrator privileges, some additional configuration steps are required.



System requirements for Qlik Sense Enterprise

This section lists the requirements that must be fulfilled by the target system in order to successfully install and run Qlik Sense.


1 Planning your Qlik Sense Enterprise deployment

Qlik Sense Enterprise on Windows




Qlik Sense Enterprise on Windows requirements

Item	Requirements
Platforms	<ul style="list-style-type: none">• Microsoft Windows Server 2016• Microsoft Windows Server 2019• Microsoft Windows Server 2022 <p>For development and testing purposes only:</p> <ul style="list-style-type: none">• Microsoft Windows 10 (64-bit version only)• Microsoft Windows 11 <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <i>These operating systems are supported by Qlik Sense. Third-party software may require service packs to be installed.</i></div>
Processors (CPUs)	<p>Multi-core x64 compatible processors</p> <p>Advanced Vector Extensions (AVX) support</p> <p>We recommend that you use at least 4 cores per node in a Qlik Analytics Platform deployment.</p>
Memory	<p>8 GB minimum (depending on data volumes, more may be required)</p> <p>Qlik Sense is an in-memory analysis technology. The memory requirements for the Qlik Sense products are directly related to the amount of data being analyzed.</p>
Disk space	5.0 GB total required to install
Disk share	SMB or NFS
Storage	<ul style="list-style-type: none">• A network file share is required for the storage to be accessible by all servers in the site. In case of a single-server deployment, local disk storage may be sufficient.• Sufficient storage is required for the volume of apps and content used in the deployment. <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <i>Consider periodically running network file share performance tests on Qlik Sense using WinShare and FreeNAS with SMB 3.0. For more information about network file share solutions, contact your IT team for support.</i></div>
Security	<ul style="list-style-type: none">• Microsoft Active Directory• Microsoft Windows Integrated Authentication• Third-party security

1 Planning your Qlik Sense Enterprise deployment

Item	Requirements
WebSockets	Web browsers and infrastructure components (such as proxies and routers) must support WebSockets.
.NET framework	4.8
PowerShell	4.0 or higher <div data-bbox="418 535 1390 786" style="border: 1px solid #ccc; padding: 10px;"> <i>When installing or upgrading Qlik Sense Enterprise Client-Managed, several unsigned PowerShell scripts are executed. If your company has a policy only allowing the execution of signed scripts, you will have to bypass this policy during installation or upgrade. See Set-ExecutionPolicy for more information on the PowerShell execution policy.</i></div>

1 Planning your Qlik Sense Enterprise deployment

Item	Requirements
<p>Repository database</p>	<p>PostgreSQL:</p> <ul style="list-style-type: none"> • 16.x (not included in the installer) • 15.x (not included in the installer) <div data-bbox="497 448 1390 618" style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <p> <i>To use PostgreSQL versions 15 and 16, you must grant all permissions on the public schema to the qlikenserepository user for proper functionality in Qlik Sense databases. .</i></p> </div> <ul style="list-style-type: none"> • 14.x (14.8 included in the installer) • 13.x (not included in the installer) • 12.x (not included in the installer) <p>Note that bundled PostgreSQL 12.5 is not recommended. If you have version 12.5 installed locally when you upgrade Qlik Sense, a warning will be displayed but you can still upgrade. For more information, see Considerations about your Qlik Sense Repository Database.</p> <ul style="list-style-type: none"> • 11.x (not included in the installer) <div data-bbox="418 978 1390 1265" style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <p> <i>If you originally installed Qlik Sense before May 2021, upgrading Qlik Sense from such a version will not upgrade the bundled PostgreSQL database. You must manually upgrade the bundled PostgreSQL database running on the 9.6 version to a newer version.</i></p> <p><i>You can use the Qlik PostgreSQL Installer to upgrade your bundled PostgreSQL database from 9.6 or 12.5 to the 14.8 version.</i></p> </div> <p>PostgreSQL is included in the Qlik Sense setup by default. However, you can also download and install it manually. When installing manually, the PostgreSQL installer will automatically install Microsoft Visual C++ 2015-2019 Redistributable (x64), which is required with PostgreSQL 14.x. For a setup using the default Qlik Sense installer with the bundled repository database, you must first install the Microsoft Visual C++ 2015-2019 Redistributable packages before attempting the Qlik Sense installation.</p> <div data-bbox="418 1534 1390 1668" style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <p> <i>The version of PostgreSQL 14.x installed with Qlik Sense does not include pgAdmin tools. You can download and install them manually if required.</i></p> </div> <p>PostgreSQL is an open source object-relational database management system. It is released under the PostgreSQL license, which is a free and open source software license.</p>
<p>Internet protocol</p>	<ul style="list-style-type: none"> • IPv4 • IPv6 • Dual stack (IPv4 and IPv6)

1 Planning your Qlik Sense Enterprise deployment

Item	Requirements
Network	The configured hostname must resolve to an IP address on the host machine.
Qlik Management Console (QMC), supported browsers	<p>The following browsers are supported for accessing the QMC.</p> <p>Supported Microsoft Windows browsers:</p> <ul style="list-style-type: none">• Microsoft Edge• Google Chrome• Mozilla Firefox (requires hardware acceleration, not supported in virtual environments) <p>CefSharp embedded browser v55 or later (CefSharp allows you to embed the Chromium open source browser inside .Net apps)</p> <p>Supported Apple macOS browsers:</p> <ul style="list-style-type: none">• Apple Safari (the last 3 major versions)• Google Chrome• Mozilla Firefox (requires hardware acceleration, not supported in virtual environments)
QMC, minimum screen resolution	<p>Desktops, laptops, and Apple Mac: 1024x768</p> <p>No mobile or small screen support.</p>
QlikView compatibility	It is not possible to install Qlik Sense on a machine with QlikView Server already installed.

1 Planning your Qlik Sense Enterprise deployment

Item	Requirements
Insight Advisor Chat	<p>Natural Language Processing (NLP) support for Insight Advisor requires a CPU that supports Advanced Vector Extensions (AVX) instructions. To find out if your CPU supports AVX, download Coreinfo v3.5 from Microsoft to view your CPU and memory topology.</p> <p>Coreinfo v3.5 - Dump information on system CPU and memory topology Copyright (C) 2008-2020 Mark Russinovich Sysinternals - www.sysinternals.com ...</p> <p>Intel(R) Core(TM) i7-9850H CPU @ 2.60GHz Intel64 Family 6 Model 158 Stepping 13, GenuineIntel Microcode signature: 000000CA HTT * Hyperthreading enabled HYPERVISOR * Hypervisor is present ... AES * Supports AES extensions AVX * Supports AVX instruction extensions FMA * Supports FMA extensions using YMM state ... Logical Processor to Group Map: ***** Group 0</p>
TLS	1.2



We do not recommend that you install Qlik Sense on domain controller machines, as group policies may prevent Qlik Sense from getting access to required services.





*License activations request access to the Qlik Licensing Service. Open port 443 and allow outbound calls to license.qlikcloud.com.
 Use of a proxy is supported. For more information about setting up a proxy service in Windows, see [Configuring a proxy for Qlik Licensing Service communication in Qlik Sense Enterprise on Windows](#).*

1 Planning your Qlik Sense Enterprise deployment

Qlik Sense Enterprise SaaS

Qlik Sense Enterprise SaaS requirements

Maximum app size (in memory)	5 GB  <i>To monitor your in-memory app size and memory usage over time, use the  App Analyzer for Qlik SaaS.</i> <ul style="list-style-type: none">• <i>This app is provided as-is and is not supported by Qlik Support.</i>• <i>Always use the latest version of the app.</i>• <i>Qlik does not collect any information when using the App Analyzer for Qlik SaaS.</i>
Total cloud storage	*Unlimited
Maximum concurrent reloads	*Unlimited
Maximum reloads per day	*Unlimited
WebSockets	Web browsers and infrastructure components (such as proxies and routers) must support WebSockets.



** Subject to restrictions described in the Qlik Sense License Metrics. You can find this document at [Qlik Product Terms](#).*



When distributing to Qlik Sense SaaS, your Qlik Sense Enterprise on Windows deployment must be either the current version or one of the previous two releases (starting from the June 2018 release).

Qlik Sense Client-Managed Mobile

Qlik Sense Client-Managed Mobile client managed requirements

Qlik Sense Client-Managed Mobile client managed device compatibility	<ul style="list-style-type: none">• 64-bit CPU architecture (ARM)• RAM: 2 GB or more (Dependent on data size)• Screen size: 720x1280 HDPI (267) or better
---	---


1 Planning your Qlik Sense Enterprise deployment

Qlik Sense Client-Managed Mobile client managed compatibility with Qlik Sense	Qlik Sense February 2020 and later releases
Qlik Sense Client-Managed Mobile client managed Apple support	<ul style="list-style-type: none">• iOS 15 or later• iPadOS 15 or later
Qlik Sense Client-Managed Mobile client managed Android support	Android 11 or later



Qlik Sense Desktop

To successfully install and run Qlik Sense Desktop, the requirements listed in this section must be fulfilled.

Qlik Sense Desktop requirements


Operating system	Microsoft Windows 10 (64-bit version only) Microsoft Windows 11
Processors (CPUs)	Intel Core 2 Duo or higher recommended. Advanced Vector Extensions (AVX) support.
Memory	4 GB minimum (depending on data volumes, more may be required). <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <i>Qlik Sense uses an in-memory analysis technology. The memory requirements are directly related to the amount of data being analyzed.</i></div>
Disk space	5.0 GB total required to install
.NET framework	4.8 or higher
Security	Local admin privileges needed to install.
Minimum screen resolution	<ul style="list-style-type: none">• Desktops, laptops and tablets: 1024x768• Small screens: 320x568

1 Planning your Qlik Sense Enterprise deployment

Browser support	<ul style="list-style-type: none">• Microsoft Edge• Google Chrome• Mozilla Firefox
	 <i>By default, Qlik Sense Desktop runs in a window of its own. But you can also open it in a web browser.</i>
	 <i>Mozilla Firefox requires hardware acceleration, not supported in virtual environments.</i>

Qlik DataTransfer

Qlik DataTransfer requirements

Platforms	<ul style="list-style-type: none">• Microsoft Windows Server 2016• Microsoft Windows Server 2019• Microsoft Windows Server 2022 <p>For development and testing purposes only:</p> <ul style="list-style-type: none">• Microsoft Windows 10 (64-bit version only) <div> <i>These operating systems are supported by Qlik Sense. Third-party software may require service packs to be installed.</i></div>
Processors (CPUs)	Multi-core x64 compatible processors. We recommend a minimum of 4 cores.
Memory	8 GB minimum The memory requirements for the Qlik Sense products are directly related to the amount of data being analyzed.
Disk space	2 GB minimum
Storage	Sufficient storage is required for the volume of apps and content used in the deployment.
PowerShell	5.1 or higher
TLS	1.2

Supported browsers

Qlik Sense is designed to work on the platform and web browser combinations described in this section, using default browser settings.

1 Planning your Qlik Sense Enterprise deployment

Each Qlik Sense release is tested for compatibility with the latest publicly available browser versions. Due to the frequency of browser version updates, Qlik does not include specific browser version numbers in the system requirements.

Each Qlik Sense release is compatible with and supported on the latest iOS versions that are publicly available at the time of the Qlik Sense release. Due to the frequency of iOS version updates, Qlik does not include specific iOS version numbers in the system requirements.



Minimum screen resolution for desktops and laptops is 1024x768; tablets is 1024x768; small screens is 320x568.

Supported Microsoft Windows browsers

The following browsers can be used on supported Microsoft Windows and Microsoft Windows Server machines to access the Qlik Management Console (QMC) and the hub:

- Microsoft Edge
- Google Chrome
- Mozilla Firefox (requires hardware acceleration, not supported in virtual environments)

CefSharp embedded browser v55 or later (CefSharp allows you to embed the Chromium open source browser inside .Net apps)

Supported Apple macOS browsers

The following browsers can be used on supported Apple macOS computers to access the Qlik Management Console (QMC) and the hub:

- Apple Safari (the last 3 major versions)
- Google Chrome
- Mozilla Firefox (requires hardware acceleration, not supported in virtual environments)

iOS/iPadOS

The following browsers can be used on supported devices (script editing is not supported on tablet devices):

- Apple Safari (the last 3 major versions)
- VMware browser (using AirWatch per-app VPN)
- MobileIron Web@Work (using MobileIron Tunnel)
- Microsoft Edge

Android

The following browsers can be used on supported devices (script editing is not supported on tablet devices):

- Google Chrome
- Microsoft Edge

Qlik Sense Enterprise on Windows architecture

The Qlik Sense architecture consists of one or more nodes. Each node runs some or all of the software services that perform specific roles in a Qlik Sense site. You can distribute services across nodes for better performance and scalability. The architecture is flexible enough to suit the needs of most organizations, and can vary from small, single-server sites to large, multi-server installations.

A multi-node, distributed architecture offers the most flexibility, consisting of multiple nodes that together form a scalable and high performance site. You define a central node as the main point of control.

Sites

A Qlik Sense site is a collection of one or more nodes (servers) connected to a single repository database, and sharing a single license. Each site also contains a common set of data in the form of apps and configuration data.

Single-node sites

A single node site is the smallest site possible and consists of a single node (single server), which is also the central node of the site. It contains the Qlik Sense services, the repository database, and the file share all on a one server computer.

Multi-node sites

Multi-node sites offer more scalability options for larger organizations. In a multi-node environment, the Qlik Sense site is distributed across two or more nodes that share the same set of data and the same license key. In larger sites, you can configure one or more rim nodes to improve scalability, capacity, and resilience. All rim nodes connect to a central node.

Benefits of multi-node sites include:

- Better scalability, making it easier to increase capacity
- Improved resilience and reliability
- Ability to move apps or roles to specific nodes
- Flexibility to suit customer network deployments

Nodes

A node is a computer that performs a specific role in your Qlik Sense site. You can configure each node to run or combine a different set of Qlik Sense services, so that each node performs a specific role.

Typical node roles:

- Consumer or user node - delivers apps to end users
- Scheduler node - handles all app reloads
- Proxy node - manages authentication, session handling, and load balancing

You can also configure your site for failover so that it is not dependent on the central node. In this case, if there is a failure, then one of the rim nodes in the site becomes the central node.

A typical multi-server Qlik Sense site consists of two main types of nodes:

1 Planning your Qlik Sense Enterprise deployment

- Central node - the minimum configuration. Every site includes a central node.
- Rim node - you can configure rim nodes to perform different roles in your site.

Each node in a Qlik Sense site can:

- Perform different roles
- Deploy a set of Qlik Sense services
- Operate independently

You assign a purpose to each node depending on what you think it will be used for:

- Production
- Development
- Both

Configuring Qlik Sense nodes correctly increases system resilience, reduces the need for maintenance, and increases deployment flexibility.

Storage

Qlik Sense uses the following default storage.

Repository database

A PostgreSQL database that contains the Qlik Sense app metadata, including the paths to the binary files in the file share. This data is referred to as entity data and is usually small in size. The PostgreSQL database can be installed locally or on a remote server and must be accessible to the central node.

File share

A file share is used to store app data as binary files and must be accessible to all nodes in your Qlik Sense site. The file share stores application objects, such as visualizations, dimensions and measures. Apps are stored in the proprietary QVF portable format, for example <App name>.qvf. These files are referred to as binary data and the data model element of the files can be large in size.

You can create a file share either on the same server as the central node or on another server.



*Qlik Sense supports Network File System (NFS) storage. You must enable **Services for NFS** from the **Control Panel > Programs and Features**.*

Clients

You use Qlik Sense clients to communicate and interact with Qlik Sense sites.

Hub

The hub is where you find all the apps you have access rights to. It runs in a web browser. You use the hub to access and publish apps in Qlik Sense. Hub traffic only travels between the node (delivering apps) and the hub client unless the site is on a single node.

Qlik Management Console

You use the Qlik Management Console (QMC) to configure and administer a Qlik Sense site.

1 Planning your Qlik Sense Enterprise deployment

The QMC only communicates logically with the central node. This means that:

- The QMC always uses the Qlik Sense Proxy Service (QPS) on the central node.
- For maximum performance within a multi-node site, you should not allow any user traffic on the central node.

Apps

A Qlik Sense app is a collection of reusable data items (measures, dimensions, and visualizations), sheets, and stories. It is a self-contained entity that includes the data you want to analyze in a structured data model.



In Qlik Sense, the term app is equivalent to the term document in QlikView.

Services

The Qlik Sense services run as Microsoft Windows services, which you can deploy on a single server or on separate server nodes that have dedicated roles in a Qlik Sense site. For example, you could deploy a scheduler node that only runs the scheduler service and manages the reloads of apps.

The Qlik Sense services are as follows.

Qlik Sense Repository Service (QRS)

Required by all Qlik Sense services to run and serve apps, and connects to the repository database. The Qlik Sense Repository Service manages persistence, licensing, security, and service configuration data. The QRS is needed by all other Qlik Sense services to run and serve apps. In a multi-node site, one instance of the Qlik Sense Repository Service (QRS) runs on each node, connecting it to the shared repository database.

In addition, the QRS stores the app structures and the paths to the binary files. The app data is stored as .qvf files in the file share.

Paths

The following table lists the paths used by the Qlik Sense Repository Service (QRS).

List of QRS paths

Executable	<code>%ProgramFiles%\Qlik\Sense\Repository\Repository.exe</code>
Data	<code>%ProgramData%\Qlik\Sense\Repository</code>
Logs	<code>%ProgramData%\Qlik\Sense\Log\Repository</code> <i>See: Logging (page 251)</i>
Repository database	<p>In a default Qlik Sense installation, the repository database is an instance of PostgreSQL installed locally that runs its own database cluster specifically for the repository.</p> <p>All files related to the repository database in a default Qlik Sense installation are stored in the following folder:</p> <code>%ProgramData%\Qlik\Sense\Repository\PostgreSQL</code>

1 Planning your Qlik Sense Enterprise deployment

Bootstrap mode

You can use the following parameters to start the Qlik Sense Repository Service in bootstrap mode when you need to deploy Qlik Sense with a service account that does not have administrator privileges.

See: *Changing the user account to run Qlik Sense services (page 153)*

- `-bootstrap`
Use this parameter to start Qlik Sense Repository Service in bootstrap mode.
- `-bootstrap=install`
Use this parameter to start Qlik Sense Repository Service in bootstrap mode when installing.
- `-bootstrap=uninstall`
Use this parameter when uninstalling Qlik Sense.
- `-iscentral`
Use this flag in addition to the bootstrap flag when installing or configuring a central node.

Do the following:

1. Stop all Qlik Sense services except Qlik Sense Repository Database.
2. Run `repository.exe -bootstrap` from an elevated command prompt. The Qlik Sense Service Dispatcher must be running before the `repository.exe -bootstrap` is executed.
3. Start all Qlik Sense services. You must start the Qlik Sense Service Dispatcher (QSD) before starting the Qlik Sense Repository Service (QRS).



By default, when you are running Qlik Sense with an administrator account, bootstrap is executed each time the Qlik Sense services are restarted. To disable automatic bootstrap in the Qlik Sense repository, you must update the configuration file. By default, the `Repository.exe.config` file can be found in `C:\Program Files\Qlik\Sense\Repository\` on your Qlik Sense machine. Edit the configuration file and change the value of the `DisableAutomaticBootstrap` key to `true`. Restart the Qlik Sense Repository Service using the Windows Services application to enable this new configuration.

Metrics

The following metrics related to the Qlik Sense Repository Service are available in the Performance Monitor in Microsoft Windows. For instructions on how to add metrics to the monitor, see *Selecting the metrics to display (page 40)*.

Qlik Sense Repository Service - Common metrics

The Qlik Sense Repository Service HTTP connection thread pool manages incoming HTTP requests.

- `HttpConnection_ThreadPool_ActiveThreadCount`: The number of active threads in the HTTP connection thread pool. Help assess the current workload and responsiveness of the service.
- `HttpConnection_ThreadPool_CompleterRequestsPerSecond`: The rate at which HTTP requests are being completed (processed and responded to). Measures the request processing efficiency and throughput.
- `HttpConnection_ThreadPool_RequestQueueSize`: The current size of the request queue in the HTTP connection thread pool. Requests are queued when all available threads are busy handling other requests. Helps assess the workload and responsiveness of the service.

1 Planning your Qlik Sense Enterprise deployment

- `HttpConnection_ThreadPool_TotalThreadCount`: The total number of threads (both active and idle) in the HTTP connection thread pool. Reflects the overall capacity and utilization of the thread pool for handling incoming HTTP requests.

Qlik Sense Repository Service - REST API metrics

- # of http DELETE calls: The number of HTTP DELETE requests made to the server. DELETE requests are used to remove a resource on the server.
- # of http GET calls: The number of HTTP GET requests made to the server. GET requests are used to retrieve data or resources from the server.
- # of http POST calls: The number of HTTP POST requests made to the server. POST requests are used to submit data to be processed by the server, often resulting in the creation of a new resource.
- # of http PUT calls: The number of HTTP PUT requests made to the server. PUT requests are used to update or replace an existing resource on the server.
- # of httpstatus 200 (OK): The number of HTTP responses with status code 200. This code indicates a successful request, where the server successfully processed and responded to the client's request.
- # of httpstatus 201 (Created): The number of HTTP responses with status code 201. This code indicates that a new resource was successfully created on the server as a result of the request.
- # of httpstatus 400 (BadRequest): The number of HTTP responses with status code 400. This code indicates that the server could not process the request due to malformed syntax or invalid parameters in the client's request.
- # of httpstatus 401 (Unauthorized): The number of HTTP responses with status code 401. This code indicates that the request requires authentication, and the client either failed to provide valid credentials or lacks authorization for the requested resource.
- # of httpstatus 403 (Forbidden): The number of HTTP responses with status code 403. This code indicates that the server understood the request but refuses to authorize it because the client does not have permission to access the requested resource.
- # of httpstatus 406 (NotAcceptable): The number of HTTP responses with status code 406. This code indicates that the server cannot respond with the requested content type specified in the client's Accept header.
- # of httpstatus 409 (Conflict): The number of HTTP responses with status code 409. This code indicates that the request could not be completed due to a conflict with the current state of the resource.
- # of httpstatus 415 (UnsupportedMediaType): The number of HTTP responses with status code 415. This code indicates that the server refuses to accept the request because the payload format is unsupported (e.g., unsupported media type).
- # of httpstatus 500 (InternalServerError): The number of HTTP responses with status code 500. This code indicates that the server encountered an unexpected condition that prevented it from fulfilling the request.
- # of httpstatus 503 (ServiceUnavailable): The number of HTTP responses with status code 503. This code indicates that the server is temporarily unable to handle the request due to maintenance or overload, and the client should retry later.

1 Planning your Qlik Sense Enterprise deployment

Qlik Sense Repository Database (QRD)

In a default Qlik Sense installation, the Qlik Sense Repository Service (QRS) uses the Qlik Sense Repository Database (QRD) service to read and write data in the repository database. By default a PostgreSQL database is installed locally with your Qlik Sense installation otherwise you can choose to install PostgreSQL on a separate dedicated server.

Paths

The following table lists the paths used by the Qlik Sense Repository Database (QRD) service.

List of QRD paths

Executable	<p>In a default Qlik Sense installation, the repository database is an instance of PostgreSQL that creates its own database cluster.</p> <p>The following folder contains the contains the PostgreSQL executable file for the QRD:</p> <p><i>%ProgramFiles%\Qlik\Sense\Repository\PostgreSQL\&lt;database version&gt;\bin</i></p>
Data	<p><i>%ProgramData%\Qlik\Sense\Repository\PostgreSQL</i></p>
Logs	<p>There are no logs for the QRD service. Instead see the PostgreSQL log files.</p>

Qlik Sense Proxy Service (QPS)

The Qlik Sense Proxy Service (QPS manages site authentication, session handling, and load balancing.

On the central node in a multi-node site, you should have a dedicated Qlik Sense Proxy Service (QPS) for the Qlik Management Console (QMC) and not for the hub.

Paths

The following table lists the paths used by the Qlik Sense Proxy Service (QPS).

List of QPS paths

Executable	<p><i>%ProgramFiles%\Qlik\Sense\Proxy\Proxy.exe</i></p>
Data	<p><i>%ProgramData%\Qlik\Sense\Proxy</i></p>
Logs	<p><i>%ProgramData%\Qlik\Sense\Log\Proxy</i> See: <i>Logging (page 251)</i></p>

Bootstrap mode

You can use the following parameters to start the Qlik Sense Proxy Service in bootstrap mode when you need to deploy Qlik Sense with a service account that does not have administrator privileges.

See: *Changing the user account to run Qlik Sense services (page 153)*

- `-bootstrap`
Use this parameter to start Qlik Sense Proxy Service in bootstrap mode.
- `-bootstrap=install`
Use this parameter to start Qlik Sense Proxy Service in bootstrap mode when installing.

1 Planning your Qlik Sense Enterprise deployment

- `-bootstrap=uninstall`
Use this parameter when uninstalling Qlik Sense.

Do the following:

1. Stop Qlik Sense services.
2. Run `proxy.exe -bootstrap` from an elevated command prompt.
3. Start Qlik Sense services.

Metrics

The following metrics related to the Qlik Sense Proxy Service are available in the Performance Monitor in Microsoft Windows. For instructions on how to add metrics to the monitor, see *Selecting the metrics to display* (page 40).

Qlik Sense Proxy Service metrics

- # of ActiveClientWebsockets: The number of active WebSockets between the client and the QPS.
- # of ActiveConnections: The number of active connections from the client. A connection is a stream (or a socket) between a Qlik Sense client and the QPS. This stream is often connected to another stream, which runs from the QPS to the Qlik Sense Repository Service (QRS) or the Qlik Sense Engine Service (QES). The two streams allow the client to communicate with the QRS or the QES.
- # of ActiveEngineWebsockets: The number of active WebSockets between the QPS and the target Qlik Sense service.
- # of ActiveSessions: The number of active sessions in the QPS.
A Qlik Sense user gets a proxy session when the user has been authenticated. The session is terminated after a certain period of inactivity.
- # of ActiveStreams: The number of active data streams (or sockets), either from the browser to the QPS or from the QPS to the QRS or the QES.
- # of HandleCount: The number of handles currently open by the QPS. Handles represent resources (like files, devices, or objects) used or managed by the service.
- # of LoadBalancingDecisions: The number of users who currently have at least one engine session.
- # of PrintingLoadBalancingDecisions: The number of users who have been load balanced to the Qlik Sense Printing Service (QPR).
- # of Tickets: The number of issued login tickets that have not yet been consumed.
- # of WorkerThreads: The current number of worker threads used by the QPS. Worker threads handle tasks and requests within the proxy service, such as managing user sessions or processing incoming requests.



The metrics are also available as entries in the Performance log for the QPS, see Performance log (page 276).

Qlik Sense Scheduler Service (QSS)

The Qlik Sense Scheduler Service (QSS) manages the scheduled reloads of apps, as well as other types of reload triggering based on task events. Depending on the type of deployment, the Qlik Sense Scheduler Service runs as manager, worker, or both on a node.

1 Planning your Qlik Sense Enterprise deployment

Manager

There is only one manager Qlik Sense Scheduler Service within a site and it is always located on the central node, where the manager Qlik Sense Repository Service runs. The central node must have the Qlik Sense Scheduler Service installed even if more QSS nodes are added because the QSS on the central node coordinates all QSS activities within the site.

The manager QSS handles all task administration. For example, which tasks to execute and when to execute a specific task. When the time comes to execute a task, the manager QSS sends the task ID to a worker QSS within the site. The load balancing operation performed by the manager QSS determines which worker QSS to distribute the task ID to.

When a worker QSS completes a task, it returns the task state (successful or fail) to the manager QSS. The manager QSS uses the task state to perform task chaining. It uses the task state to determine if other events are affected by the state of the completed task and need to be executed. You configure task chaining in the Qlik Management Console (QMC).

If the worker QSS fails to perform the task, the manager QSS repeatedly requests the same or another worker QSS to perform the task until it has been completed or until the maximum number of attempts has been reached.

Worker

If a Qlik Sense Scheduler Service (QSS) runs on a rim node, the QSS is considered to be a worker QSS. When receiving a task ID from the manager QSS, the worker QSS reads the task from the local repository database and executes the task. When a worker QSS completes a task, it returns the task state (successful or fail) to the manager QSS.

Tasks

Tasks are used to perform a wide variety of operations and can be chained together in any arbitrary pattern. The tasks are handled by the Qlik Sense Scheduler Service (QSS) and managed in the Qlik Management Console (QMC).

Reload

The reload task is used to fully reload the data in an app from the source. Any old data is discarded.

Paths

The following table lists the paths used by the Qlik Sense Scheduler Service (QSS).

List of QSS paths

Executable	<i>%ProgramFiles%\Qlik\Sense\Scheduler\Scheduler.exe</i>
Data	-
Logs	<i>%ProgramData%\Qlik\Sense\Log\Scheduler</i> <i>See: Logging (page 251)</i>

Bootstrap mode

You can use the following parameters to start the Qlik Sense Scheduler Service in bootstrap mode when you need to deploy Qlik Sense with a service account that does not have administrator privileges.

1 Planning your Qlik Sense Enterprise deployment

See: *Changing the user account to run Qlik Sense services (page 153)*

- `-bootstrap`
Use this parameter to start Qlik Sense Scheduler Service in bootstrap mode.
- `-bootstrap=install`
Use this parameter to start Qlik Sense Scheduler Service in bootstrap mode when installing.
- `-bootstrap=uninstall`
Use this parameter when uninstalling Qlik Sense.

Do the following:

1. Stop Qlik Sense services.
2. Run `scheduler.exe -bootstrap` from an elevated command prompt.
3. Start Qlik Sense services.

Metrics

The following metrics related to the Qlik Sense Scheduler Service (QSS) are available in the Performance Monitor in Microsoft Windows. For instructions on how to add metrics to the monitor, see *Selecting the metrics to display (page 40)*.

Qlik Sense Scheduler Service metrics

- # of connected workers: The number of currently connected QSS worker instances.
- # of running engines: The number of Qlik Sense Engine Service (QES) instances running on a worker. This metric is only available on the node where the QES instances run.
- # of running processes: The total number of processes currently active within the QSS.
- # of running tasks as understood by the manager: The number of tasks currently running, managed by the QSS manager.
- # of running tasks on the worker: The number of tasks currently running on the worker node.
- # of task messages dispatched by the worker: The number of messages related to tasks sent out by the worker.
- # of task messages received by the manager: The number of messages related to tasks received by the QSS manager.
- # of task retries: The number of times tasks have been retried after initial failure.
- # of tasks that has completed successfully when executed by the worker: The number of tasks successfully completed by worker nodes.
- # of tasks that has failed when executed by the worker: The number of tasks that failed when executed by worker nodes.
- # of tasks the manager has acknowledged as completed: The number of tasks acknowledged as completed by the QSS manager.
- # of tasks the manager has acknowledged as failed: The number of tasks acknowledged as failed by the QSS manager.
- # of times the settings has been updated: The number of times settings related to QSS have been modified.

1 Planning your Qlik Sense Enterprise deployment

- # of tasks we've attempted to start: The number of attempted task starts.
- # of tasks we've attempted to stop: The number of attempted task stops.

Qlik Sense Engine Service (QES)

The Qlik Sense Engine Service (QES) handles all application calculations and logic. In a multi-node site, we recommend that you have a dedicated Qlik Sense Engine Service (QES) on the central node that you use specifically for the Qlik Management Console (QMC) and not for the hub.

Paths

The following table lists the paths used by the Qlik Sense Engine Service (QES).

List of QES paths

Executable	<i>%ProgramFiles%\Qlik\Sense\Engine\Engine.exe</i>
Data	<i>%ProgramData%\Qlik\Sense\Engine</i>
Logs	<i>%ProgramData%\Qlik\Sense\Log\Engine</i> See: <i>Logging (page 251)</i>
Configuration	<i>%ProgramData%\Qlik\Sense\Engine\Settings.ini</i> This file contains the QES settings. The file is created when the service first runs.

Qlik Sense Printing Service (QPR)

This service manages export in Qlik Sense. In a multi-node site, one instance of the Qlik Sense Printing Service (QPR) runs on each node. Export requests from clients are directed to the printing services in the multi-node site using round robin load balancing. If the first export request is load balanced to the QPR on node 1, the second export request is load balanced to the QPR on node 2, and so on.

Paths

The following table lists the paths used by the Qlik Sense Printing Service (QPR).

List of QPR paths

Executable	<i>%ProgramFiles%\Qlik\Sense\Printing\Printing.exe</i>
Data	<i>%ProgramData%\Qlik\Sense\Printing</i>
Logs	<i>%ProgramData%\Qlik\Sense\Log\Printing</i> See: <i>Logging (page 251)</i>

Qlik Sense Service Dispatcher (QSD)

This is a service controller used to launch and manage the following Qlik Sense services:

- Broker Service: acts as an interface to and an intermediary between services started by the Qlik Sense Service Dispatcher(QSD). The service is launched and managed by the Qlik Sense Service Dispatcher (QSD) when required.

1 Planning your Qlik Sense Enterprise deployment

- Data Profiling Service: is used to access and modify the app load data model. It communicates directly with the Qlik Sense Engine Service (QES) on the node. The service is launched and managed by the Qlik Sense Service Dispatcher (QSD) when required.
- Hub Service: controls which content a user is allowed to see based on their access rights as defined in the QMC. The service is launched and managed by the Qlik Sense Service Dispatcher(QSD) when required.
- Web Extension Service: is used to control web extensions such as visualizations, mashups, and widgets. The service is launched and managed by the Qlik Sense Service Dispatcher (QSD) when required.
- Capability Service: is used to handle Qlik Sense .NET SDK system feature configuration.
- Converter Service: is used by the QlikView converter tool.
- On-demand App Service: generates on-demand apps that load subsets of data from very large data sets.
- Hybrid Deployment Service (HDS): manages target deployments and credentials related to hybrid connectivity between environments, specifically the distribution of apps from the QSE.
- Hybrid Setup Console (HSC): serves the HSC user interface which is used to configure target deployments and app distribution.
- App Distribution Service (ADS): distributes apps and associated metadata to defined distribution targets, based on policy based app distribution rules.
- Precedents Service: captures user-learned feedback from Insight Advisor.

Paths

The following table lists the paths used by the Qlik Sense Service Dispatcher (QSD) and the services that are launched and managed by the QSD.

List of QSD paths

Executables	<ul style="list-style-type: none">• QSD: <i>%ProgramFiles%\Qlik\Sense\ServiceDispatcher\ServiceDispatcher.exe</i>• Services that are launched and managed by the QSD: <i>%ProgramFiles%\Qlik\Sense\ServiceDispatcher\node\node.exe</i>
--------------------	---

1 Planning your Qlik Sense Enterprise deployment

Logs	<ul style="list-style-type: none">• Broker Service: <code>%ProgramData%\Qlik\Sense\Log\BrokerService</code>• Data Profiling Service: <code>%ProgramData%\Qlik\Sense\Log\DataProfiling</code>• Hub Service: <code>%ProgramData%\Qlik\Sense\Log\HubService</code>• Web Extension Service: <code>%ProgramData%\Qlik\Sense\Log\WebExtensionService</code>• On-demand App Service: <code>%ProgramData%\Qlik\Sense\Log\OdagService</code>• Capability Service: <code>%ProgramData%\Qlik\Sense\Log\CapabilityService</code>• Hybrid Deployment Service: <code>%ProgramData%\Qlik\Sense\Log\HybridDeploymentService</code> For the Hybrid Deployment Service, you can modify some of the settings via the <code>appsettings.json</code> file.• Hybrid Setup Console: <code>%ProgramData%\Qlik\Sense\Log\HybridSetupConsole</code>• App Distribution Service: <code>%ProgramData%\Qlik\Sense\Log\AppDistributionService</code> For the App Distribution Service, you can modify some of the settings via the <code>appsettings.json</code> file.• Precedents Service: <code>%ProgramData%\Qlik\Sense\Log\PrecedentsService</code> <p>See: <i>Logging (page 251)</i></p>
-------------	---

Qlik License Service

The Qlik License Service is included in Qlik Sense Enterprise February 2019 and later releases and is used when Qlik Sense is activated using a signed key license. The Qlik License Service stores the information about the license, and communicates with a License Back-end Service, hosted by Qlik, for product activations and entitlement management. Port 443 is used for accessing the License Back-end Service and retrieving license information.

In a Qlik Sense Enterprise on Windows multi-node deployment, the Qlik License Service is installed on every node. You can manage the status of the Qlik License Service by starting and stopping the Qlik Sense Service Dispatcher, listed in the list of services running in the Windows machine.

Deployment examples of nodes running Qlik Sense services

You can deploy Qlik Sense services to run individually or combine them on dedicated server nodes.

- Complete: A single-node deployment that includes all Qlik Sense services.
- Consumer node: A node that delivers Qlik Sense apps to end users. It includes the Qlik Sense Engine Service service, the Qlik Sense Proxy Service, and the Qlik Repository service.
- Proxy node: A node that manages Qlik Sense authentication, session handling, and load balancing. It includes the QRS, and the QPS services.
- Engine node: A node that provides the analytical power of Qlik Sense to the client. It includes the QRS, and the QES services.
- Proxy and engine node: A combined node that includes the QRS, QPS, and QES service.
- Scheduler: A node that manages scheduled reloads of Qlik Sense apps and other types of reload triggering. It includes the QRS, QSS, and QES services. In order to perform reloads the QSS requires the QES to be running on the same node.

1 Planning your Qlik Sense Enterprise deployment

Service dependencies

This section describes the dependencies related to the Qlik Sense services (for example, dependencies on the operating system and other software).

Repository database

The Qlik Sense Repository Service (QRS) connects to the repository database to store and retrieve data necessary for the Qlik Sense services on the node on which the QRS is running. In a default Qlik Sense installation, the Qlik Sense Repository Service (QRS) uses the Qlik Sense Repository Database (QRD) service to read and write data in the repository database. A PostgreSQL database is used by default.

File share

The file share stores the binary files for the Qlik Sense apps.

Directory service

The QRS and Qlik Sense Proxy Service (QPS) communicate with a configured directory service (for example, Microsoft Active Directory) using, for example, LDAP or ODBC.

Start and restart of services

When a node starts up, the Qlik Sense services are started automatically.

Start-up behavior

The Qlik Sense Repository Database (QRD) and Qlik Sense Repository Service (QRS) are started first.

When any other Qlik Sense service starts, it contacts its local QRS to get configuration parameters. If the service has not been configured to run, it periodically checks back with the local QRS.

Manual start

If you need to start services manually, start them in the following order:

1. Qlik Sense Repository Database (QRD)
2. Qlik Sense Service Dispatcher (QSD)
3. Qlik Sense Repository Service (QRS)
4. Qlik Sense Proxy Service (QPS), Qlik Sense Engine Service (QES), Qlik Sense Scheduler Service (QSS), and Qlik Sense Printing Service (QPR) in no specific order

The start-up order is important. During start-up the QRS must be able to contact the Qlik License Service, which is managed by the QSD. The other services are dependent on the QRS. The QSD must therefore be running when the QRS is started.

Selecting the metrics to display

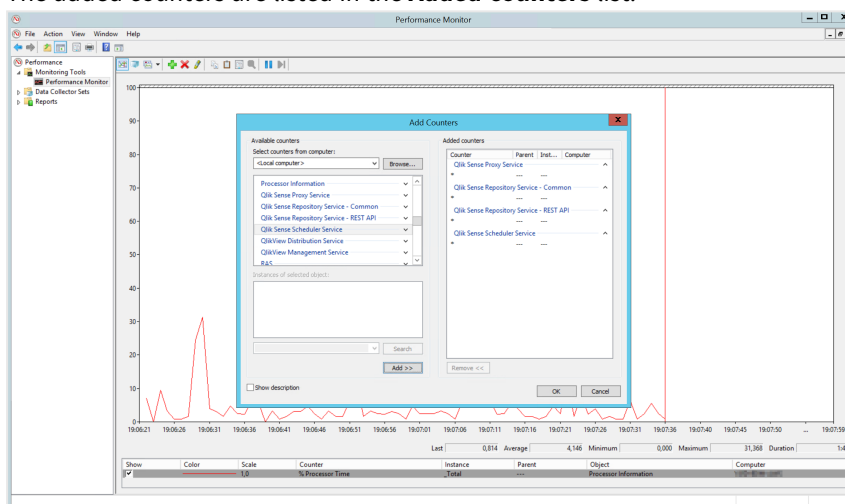
Performance counters in Windows provide a way to monitor various aspects of system performance. Each type of performance data is represented by a performance counter. Select which metrics to display for the Qlik Sense services in the Windows Performance Monitor.

Do the following:

1. Select **Start>Run**.
2. Enter *perfmon* and click **OK**.

1 Planning your Qlik Sense Enterprise deployment

3. In the left panel, expand **Monitoring Tools**.
4. Select **Performance Monitor**.
The **Performance Monitor** is displayed in the right panel.
5. Click the + (plus) icon in the toolbar at the top of the **Performance Monitor**.
The **Add Counters** dialog is displayed.
6. Select the computer to add counters from in the **Select counters from computer:** drop-down list.
The **Available counters** list is populated with counters.
7. In the **Available counters** list, locate the following counter sets :
 - Qlik Sense Proxy Service
 - Qlik Sense Repository Service - Common
 - Qlik Sense Repository Service - REST API
 - Qlik Sense Scheduler Service
8. Click the + (plus) sign next to a counter set to expand the set.
9. In the **Performance Monitor**, select the counters to display.
10. Click **Add >>** to add the counters.
11. The added counters are listed in the **Added counters** list.



12. Click **OK**.

The counters you added are now displayed in the **Performance Monitor**.

Multi-cloud services

. The services that you need to run in a multi-cloud deployment are managed by Qlik and they are running on Qlik-owned infrastructure.

Services on Windows deployments

The services listed below are required if you use the multi-cloud capabilities in a Qlik Sense Enterprise on Windows deployment.

1 Planning your Qlik Sense Enterprise deployment

Multi-cloud services

Service	Description
App Distribution Service	Distributes apps and associated metadata to defined distribution targets, based on policy-based app distribution rules.
Hybrid Deployment Service	Stores configuration details including credentials and URLs for all target environments in a multi-cloud deployment.
Hybrid Setup Console Service	Multi-cloud Setup Console UI functions for managing target environments configured in a multi-cloud deployment including credentials and service URLs.
Resource Distribution Service	Publishes installed extensions and themes to the Resource Library in each cloud environment.



Ports

Qlik Sense Enterprise uses ports to communicate between web browsers (users) and proxies, and between services in single and multi-node deployments.




Ports overview

The following tables are an overview of the ports used in a Qlik Sense deployment.



Communication ports

Service	Inbound	Outbound	Internal only
Qlik Sense Proxy Service (QPS) 	80 (HTTP) 443 (HTTPS) 4243 (REST API)	4239 (QRS websocket) 4242 (QRS REST API) 4747 (Engine) 4899 (Printing) 4900 (Broker) 4949 (Data profiling) 7070 (Logging service)	4244 (Windows authentication)
Qlik Sense Engine Service (QES) 	4747 (QES listen port)	7070 (Logging service)	4748 (notifications from QRS)

1 Planning your Qlik Sense Enterprise deployment

<p>Qlik Sense Repository Service (QRS)</p> 	<p>4242 (REST API)</p> <p>4239 (from QPS - websocket)</p> <p>4240 (When QRS is run in test mode)</p> <p>4444 (Setup API - inbound on rim nodes)</p>	<p>4242 (REST API)</p> <p>4243 (Proxy REST API)</p> <p>4240 (When QRS is run in test mode)</p> <p>4444 (Setup API – outbound on central node)</p> <p>4747 (Engine)</p> <p>4748 (Engine notification API)</p> <p>5050 (Scheduler manager API)</p> <p>7070 (Logging service)</p> <p>9200 (License Service)</p>	<p>4570 (Certificate unlock)</p>
<p>Qlik Sense Scheduler Service (QSS)</p> 	<p>5050 (Manager REST API)</p> <p>5151 (Worker REST API)</p> <p>5252 (Monitoring API - optional)</p>	<p>4242 (QRS REST API)</p> <p>7070 (Logging Service)</p> <p>5050 (Worker to Manager)</p> <p>5151 (Manager to Worker)</p>	<p>No additional ports.</p>
<p>Qlik Sense Repository Database (QRD)</p> 	<p>4432 (default listen port for database connections)</p>	<p>-</p>	<p>No additional ports.</p>

1 Planning your Qlik Sense Enterprise deployment

<p>Qlik Sense Printing service (QPR)</p> 	4899 (QPR listen port)	-	443 (Sense web server - proxy) 4242 (QRS REST API) 8088 (CEF debugging) 2770 (Internal WebSocket - proxy) One port in range 50101 - 50131 (GRPC proxy port) One port in range 52051 - 52081 (GRPC web renderer port)
<p>Qlik License Service</p> 	-	443 (HTTPS)	9200


1 Planning your Qlik Sense Enterprise deployment

Broker service	4900	3003 (Converter service) 4555 (Chart sharing) 4949 (Data profiling) 4950 (Precedents service) 9028 (Hub service) 9031 (Capability service) 9032 (About Service) 9041 (Connector registry proxy - server) 9051 (Connector registry proxy - desktop) 9054 (Precedents service) 9079 (Depgraph service) 9080 (Web extension service) 9081 (Qlik Notifier Service) 9082 (Qlik Mobility Registrar) 9090 (DownloadPrep) 9098 (On-demand app service) 21060 (Resource Distribution Service) 46277 (Deployment based warnings service) 64210 (Cayley.io—Open source graph database layer used by Precedents service)	-
Data profiling service	4949 (listen port for REST API and websocket)		4242 (QRS REST API) 4747 (QES)

1 Planning your Qlik Sense Enterprise deployment

App Distribution Service	-	5926	No additional ports.
Hybrid Deployment Service	-	5927	No additional ports.
Hybrid Setup Console - HSC	5929	-	No additional ports.
Logging Service	7080 7081	-	-
Qlik Catalog Service	4850	-	-
NL Parser Service	4952	-	-
NL Broker Service	4951	-	-
NLApp Search Service	4953		
BotChannel Service	4954		

Other ports

Service	Purpose
Qlik Sense Service Dispatcher (QSD) 	Starts up the following services: <ul style="list-style-type: none"> • Qlik License Service • Broker service • Data profiling service • App Distribution Service



To allow access to the file share, ensure that you open the Microsoft Windows SMB port 445.

Ports used internally within a node

The ports in the following table are used between Qlik Sense services that run on the same node. In most cases, the ports do not have to be open through any firewalls.

Internal ports

Service	Port	Direction	Purpose
Converter Service	3003	Internal	This port is used by the Converter Service which is utilized by QlikView converter.

1 Planning your Qlik Sense Enterprise deployment

QPS	4243	Inbound	<p>Qlik Sense Proxy Service (QPS) REST API listen port.</p> <p>If web ticketing is used for security, this port is used by the software or service that requests tickets for users. If the software or service is remote, this port needs to be open to the location from which it is called.</p>
QRD	4432	Internal	<p>Default listen port for the Qlik Sense Repository Database (QRD).</p> <p>With shared persistence, this port is used to listen for connections from the Qlik Sense Repository Service (QRS).</p>
Chart Sharing Service	4555	Internal	<p>This port is used by the Chart Sharing Service for chart sharing between Qlik Sense users. The service is launched and managed by the Qlik Sense Service Dispatcher (QSD) when required.</p> <p>This port uses HTTPS for communication.</p>
QRS	4570	Internal	<p>Certificate password verification port, only used within multi-node sites by Qlik Sense Repository Services (QRSs) on rim nodes to receive the password that unlocks a distributed certificate. The port can only be accessed from localhost and it is closed immediately after the certificate has been unlocked. The communication is always unencrypted.</p>
QES	4748	Internal	<p>This callback port is used by the Qlik Sense Repository Service (QRS) for sending HTTP events to the Qlik Sense Engine Service (QES).</p>
Data Profiling Service	4949	Internal	<p>This port is used by the Data Profiling Service to access and modify the app load data model. It communicates directly with the Qlik Sense Engine Service (QES) on the node.</p>
Broker Service	4900	Internal	<p>Default listen port for the Broker Service.</p>
Hub Service	9028	Internal	<p>Default listen port for the Hub Service.</p>
Capability Service	9031	Internal	<p>This port is used by the Capability Service to handle Qlik Sense system feature configuration.</p>
About Service	9032	Internal	<p>Default listen port for inbound calls to the About Service.</p>
Depgraph Service	9079	Internal	<p>This port is used by the Service Dispatcher launched microservices.</p>
Web Extension Service	9080	Internal	<p>Default listen port for the Web Extension Service.</p>

1 Planning your Qlik Sense Enterprise deployment


DownloadPrep	9090	Internal	This port is used by the Service Dispatcher launched microservices.
On-demand App Service	9098	Internal	Default listen port for the On-demand App Service.
Connector registry proxy (server)	9041	Internal	This port is used by the distributed connectivity service for discovering and listing connectors.
Connector registry proxy (desktop)	9051	Internal	This port is used by the distributed connectivity service for discovering and listing connectors.
Qlik Notifier Service	9081	Internal	This port is used by the Qlik Notifier Service, which handles push notifications to mobile devices. It is installed on each node in a Qlik Sense Enterprise deployment.
Qlik Mobility Registrar	9082	Internal	This port is used by the Qlik Mobility Registrar, which is installed on each node in a Qlik Sense Enterprise deployment.

Ports used from user web browser

The default ports are exposed to the Qlik Sense users and need to be open through any firewalls in the site.

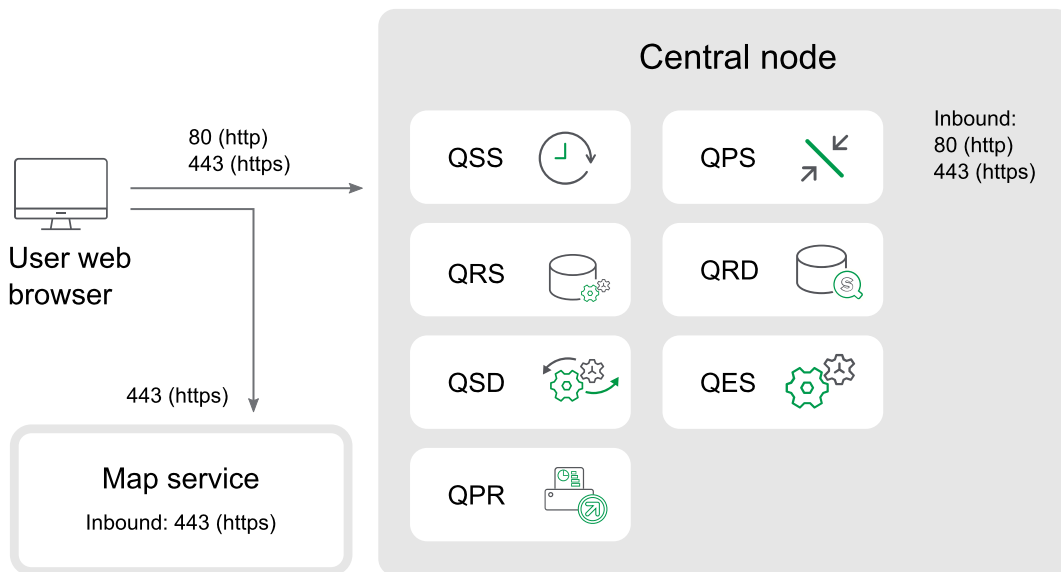
Web browser ports

Service	Port	Direction	Purpose	Host
QPS	443	Inbound	Inbound user web traffic when using HTTPS.	Qlik Sense Proxy Service (QPS) in the site.
QPS	80	Inbound	Inbound user web traffic when using HTTP (optional).	Qlik Sense Proxy Service (QPS) in the site.
Map	443	Inbound	User web traffic for standard map background. For users hosting their own map server, use the name of the host server.	maps.qlikcloud.com
Map	443	Inbound	User web traffic for satellite map background.	ibasemaps-api.arcgis.com

 As of December 7, 2021, the server is *ibasemaps-api.arcgis.com*. Previously this was *services.arcgisonline.com*.

1 Planning your Qlik Sense Enterprise deployment

The following diagram shows the ports used for the communication between a web browser and as single node site.



Ports used between nodes and Qlik Sense services

The ports in this section are used for communication between the Qlik Sense services.

In a single node site, all ports listed in this section are used by the various services, but do not need access through firewalls.

In a multi-node site, the ports in use vary depending on the services installed and running on each node. The ports need to be open in any firewalls between the nodes, but do not have to be open to the Qlik Sense users.

Minimum ports used for communication in multi-node sites

The following ports must always be open between the nodes in a multi-node site. The ports must be open to allow for service health, and some specific operations.



Inbound ports indicate the listening ports for the services running on each node. Firewall rules must allow inbound traffic to these ports. Outbound ports indicate the destination of the communication from one node to other nodes in the environment. Firewall rules must allow the node to send outbound traffic to these outbound ports.

Service	Port	Direction	Purpose
QRS	4242	Bi-directional between the central node and all proxy nodes	This port is used for a number of operations including new user registration.

1 Planning your Qlik Sense Enterprise deployment

QRD	4432	Inbound from Qlik Sense nodes to the repository database	The default listen port used by all nodes in a site for connecting to the Qlik Sense Repository Database.
QRS	4444	Between the central node and all rim nodes	<p>This port has two functions:</p> <ul style="list-style-type: none">• Security distribution port, only used by Qlik Sense Repository Services (QRSs) on rim nodes to receive a certificate from the primary QRS on the central node. The communication is always unencrypted, but the transferred certificate package is password-protected.• Qlik Sense Repository Service (QRS) state port, used to fetch the state of a QRS in a Qlik Sense site. The state is fetched using <code>http://localhost:4444/status/servicestate</code>. The returned state is one of the following:<ul style="list-style-type: none">• 0: Initializing. Once the node has been initialized, the node state changes into one of the other states.• 1: Certificates not installed. There are no certificates installed on the node. The node stays in this state until it has received the certificate and the certificate password.• 2: Running. The node is up and running and all APIs have been initiated.

Ports used between manager and worker schedulers

The ports in the following table are used when a worker Qlik Sense Scheduler Service (QSS) is used.

Ports between manager and worker schedulers

Service	Port	Direction	Purpose
QSS	5050	Inbound (from scheduler nodes only)	This port is used by the manager QSS on the central node to issue commands to and receive replies from worker QSS nodes.
QSS	5151	Inbound (from the central node only)	A worker QSS runs on a worker scheduler node and is accessed only by the manager QSS on the central node.

Ports used between a proxy node and an engine node

The ports in the following table define the minimum needed to allow regular user traffic and load balancing between a proxy node and an engine node.

1 Planning your Qlik Sense Enterprise deployment

Ports between proxy and engine nodes

Service	Port	Direction	Purpose
QES	4747	Inbound (from proxy nodes)	Qlik Sense Engine Service (QES) listen port. This is the main port used by the QES. The port is used via the Qlik Sense Proxy Service (QPS) for communication with the Qlik Sense clients.
QRS	4239	Inbound (from proxy nodes)	Qlik Sense Repository Service (QRS) WebSocket port. The port is used via the Qlik Sense Proxy Service (QPS) by the Qlik Sense hub to obtain apps and stream lists.
QRS	4242	Inbound (from proxy nodes)	Qlik Sense Repository Service (QRS) REST API listen port. This port is mainly accessed by local Qlik Sense services. However, the port must be open to all proxy nodes in a multi-node site to deliver images and static content.
Data Profiling Service	4949	Inbound (from proxy nodes)	This port is used by the Data Profiling Service when accessing and modifying the application load model. The service is launched and managed by the Qlik Sense Service Dispatcher (QSD) when required. The port is access via the Qlik Sense Proxy Service (QPS).
Broker Service	4900	Inbound (from proxy nodes)	Default listen port for the Broker Service.
Hub Service	9028	Inbound (from proxy nodes)	Default listen port for the Hub Service. Open for local services such as the broker service on the engine node.

Ports used between a proxy node and a node running the printing service

The Qlik Sense Printing Service (QPR) may be installed on the same node as other services or on a separate node. The ports in the following table must be accessible between a QPS and all QPRs to which the QPS can load balance traffic.

Ports between proxy and printing nodes

Service	Port	Direction	Purpose
QPR	4899	Inbound (from proxy nodes)	Qlik Sense Printing Service (QPR) port. This port is used for printed export in Qlik Sense. The port is accessed by any node that runs a QPS.

Qlik Sense Desktop ports

The following ports are used by Qlik Sense Desktop.


1 Planning your Qlik Sense Enterprise deployment

Desktop ports

Component	Port	Direction
Qlik associative engine (including authentication)	9076	Internal
DataPrep Service	9072	Internal
Broker Service (Desktop)	4848	Internal/inbound
Capability Service	9075	Internal
About Service	9078	Internal
Broker Service	9070	Internal
NPrinting	9073	Internal
Hub Service	9071	Internal
Converter Service	9077	Internal
Dependency Graph Service	9033	Internal
Web Extension Service	9034	Internal
Connector Registry Proxy	9051	Internal
NL Broker Service	9055	Internal
NL Parser Service	9056	Internal

Qlik DataTransfer ports

Qlik DataTransfer uses the following ports:

Service	Port	Direction
Secure web browser communication (HTTPS)	443	Outbound
 <i>You must open this port in your firewall.</i>		
Data Upload service	5505	Internal
Engine service	5506	Internal
Connector Registry proxy	5507	Internal

Ports examples

This section provides examples of the ports that are used in different Qlik Sense deployments.

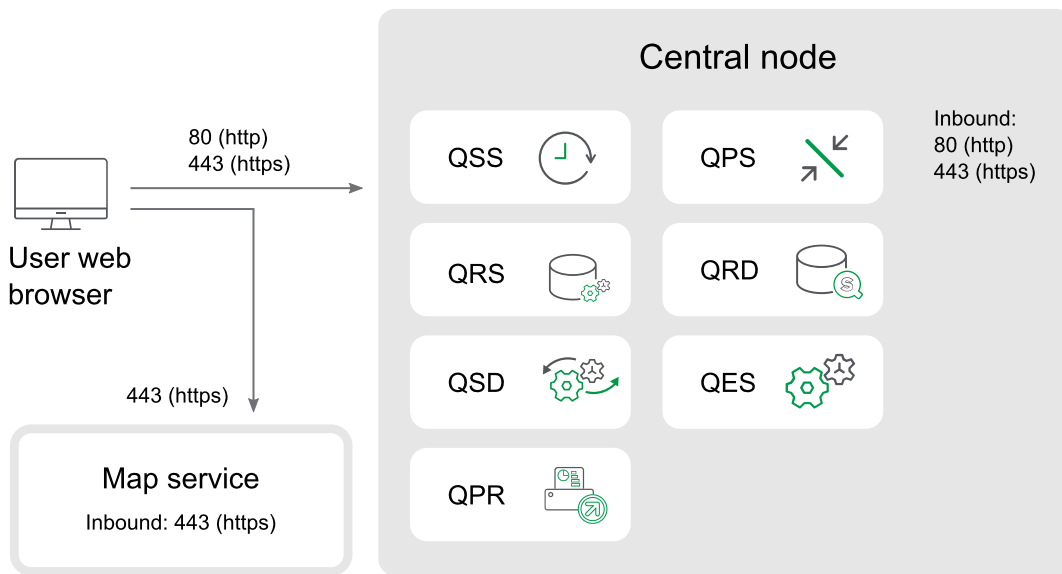


The diagrams in this section do not show all outbound proxy node ports. For a full list of proxy node ports see the Ports overview (page 42) table.

Single node site

This example shows the ports that are used in a single node site.

1 Planning your Qlik Sense Enterprise deployment

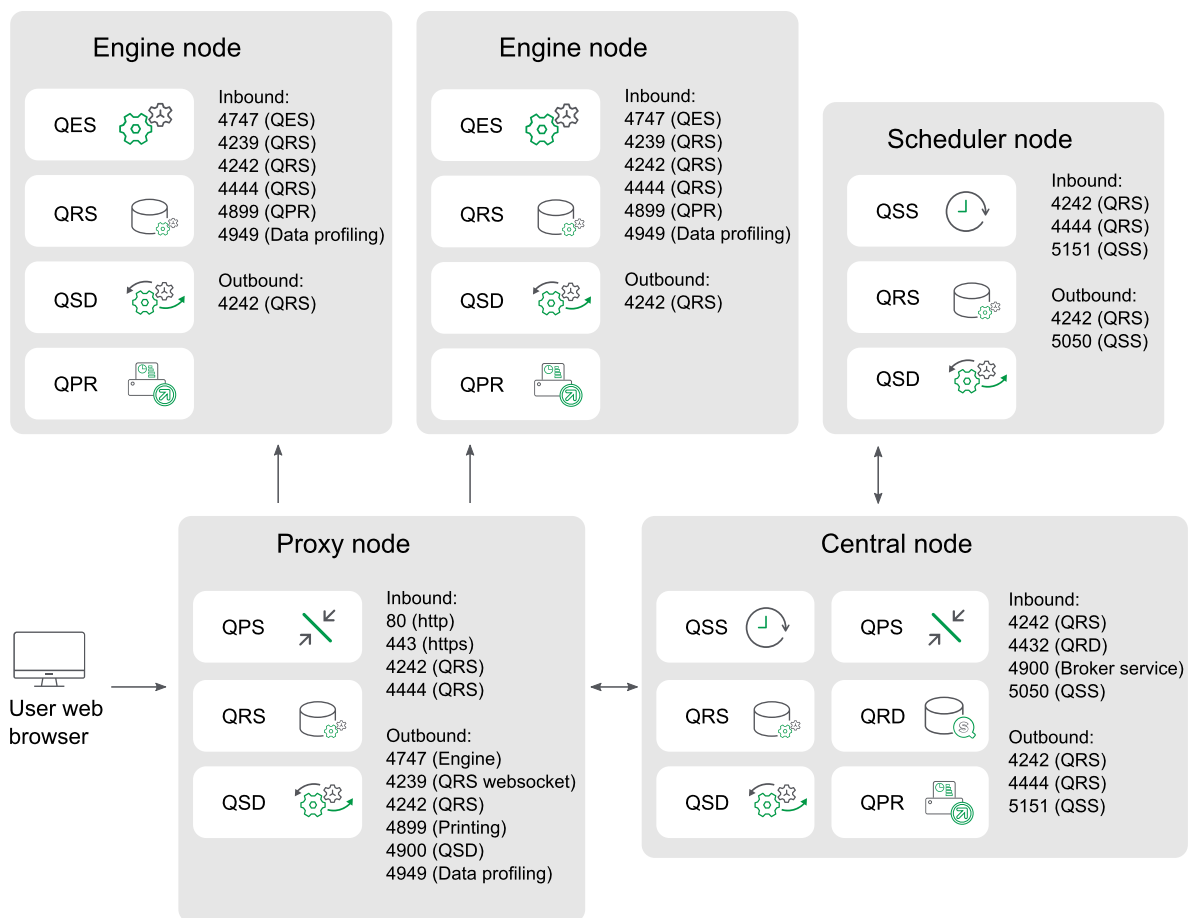


Multi-node site

The following is an example of the ports that are used in a multi-node site that consists of five nodes.

Inbound ports indicate the listening ports for the services running on each node. Firewall rules must allow inbound traffic to these ports. Outbound ports indicate the destination of the communication from one node to other nodes in the environment. Firewall rules must allow the node to send outbound traffic to these outbound ports.

1 Planning your Qlik Sense Enterprise deployment

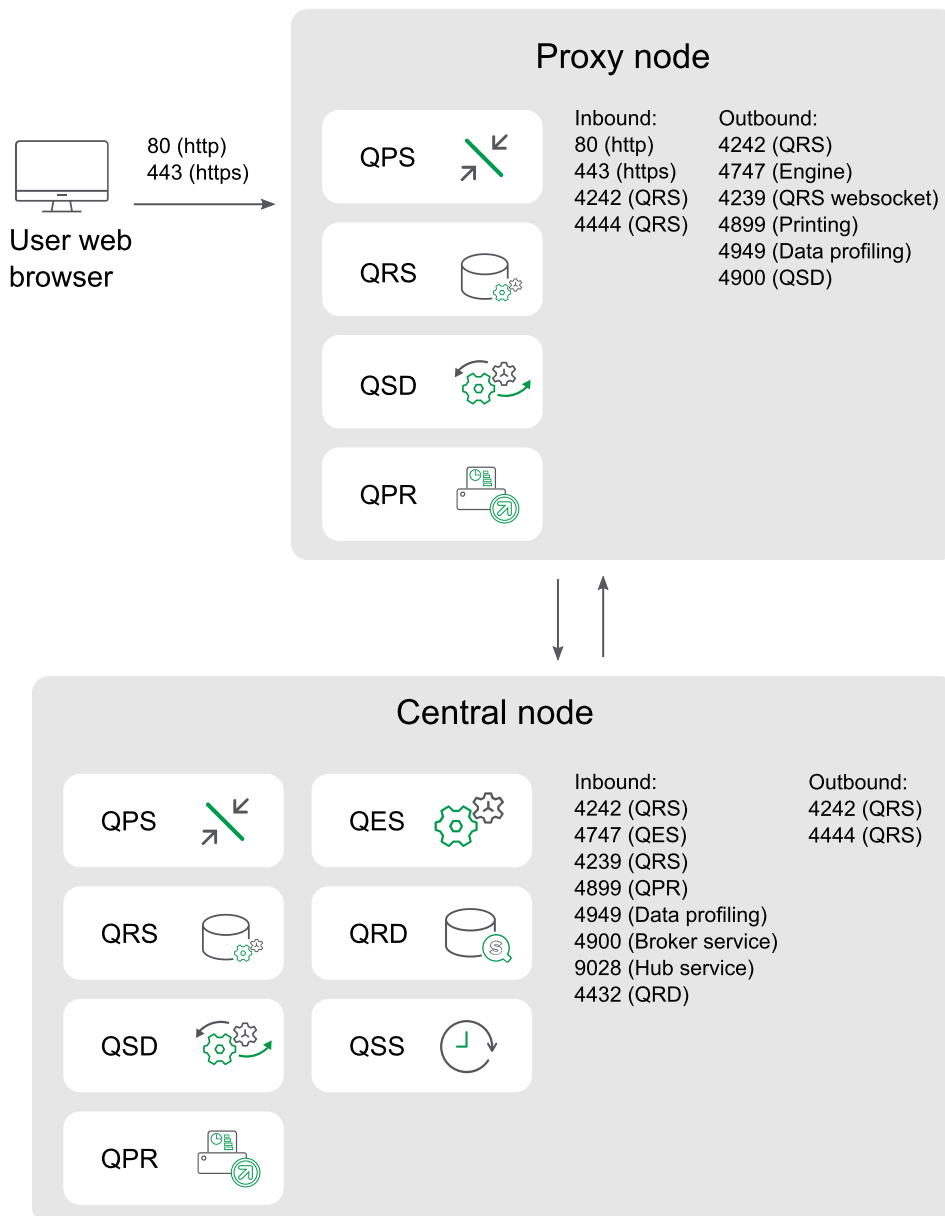


Proxy node in demilitarized zone

This example shows the ports that are used in a multi-node site when deploying a proxy node in a demilitarized zone.

Inbound ports indicate the listening ports for the services running on each node. Firewall rules must allow inbound traffic to these ports. Outbound ports indicate the destination of the communication from one node to other nodes in the environment. Firewall rules must allow the node to send outbound traffic to these outbound ports.

1 Planning your Qlik Sense Enterprise deployment

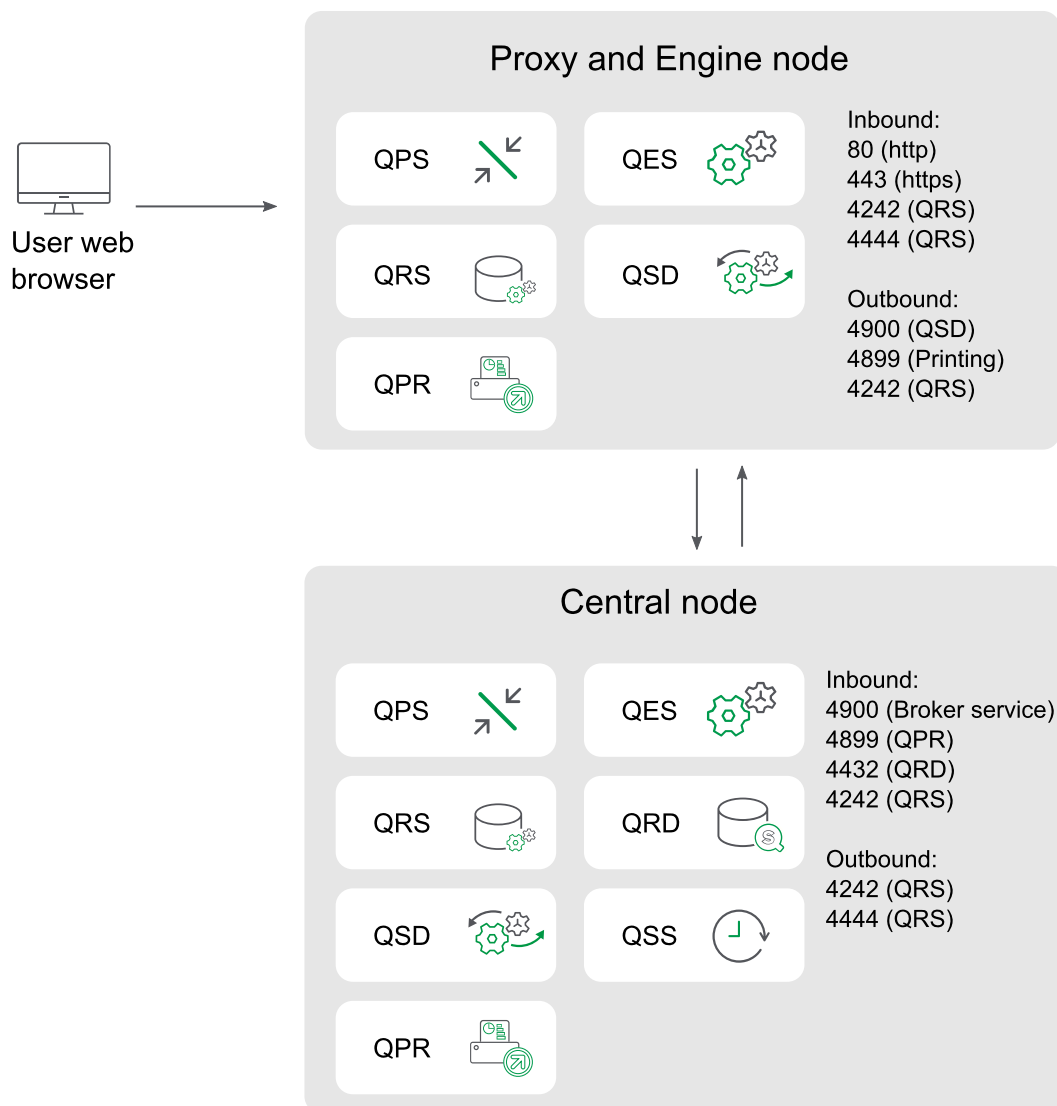


Separate proxy and engine node

This example shows the ports that are used in a multi-node site when deploying a separate proxy and engine node. The proxy load balancing excludes the engine on the central node.

Inbound ports indicate the listening ports for the services running on each node. Firewall rules must allow inbound traffic to these ports. Outbound ports indicate the destination of the communication from one node to other nodes in the environment. Firewall rules must allow the node to send outbound traffic to these outbound ports.

1 Planning your Qlik Sense Enterprise deployment

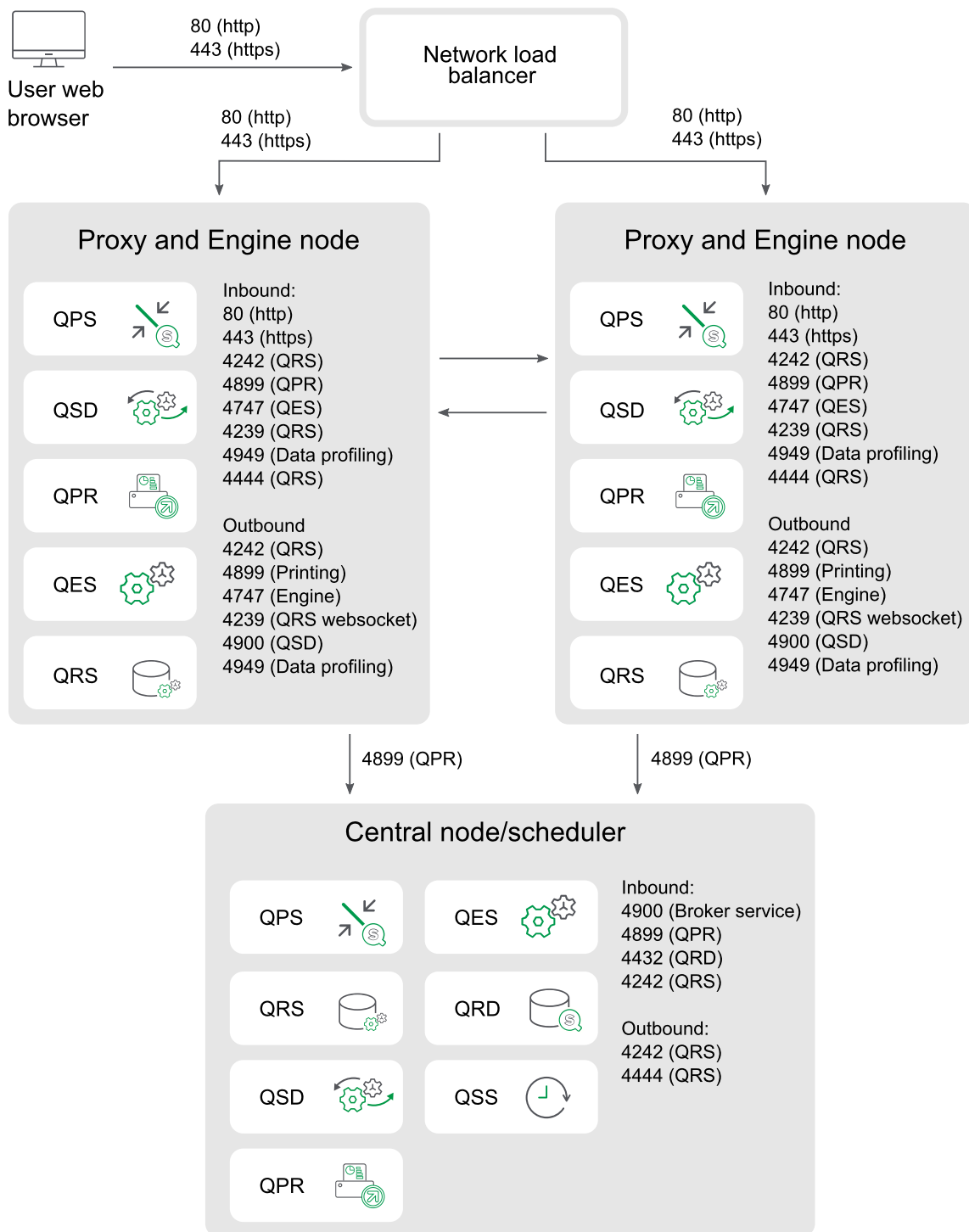


High availability proxy and engine nodes

This example shows the ports that are used in a multi-node site when deploying more than one proxy and engine node. The proxy load balancing excludes the engine on the central node.

Inbound ports indicate the listening ports for the services running on each node. Firewall rules must allow inbound traffic to these ports. Outbound ports indicate the destination of the communication from one node to other nodes in the environment. Firewall rules must allow the node to send outbound traffic to these outbound ports.

1 Planning your Qlik Sense Enterprise deployment

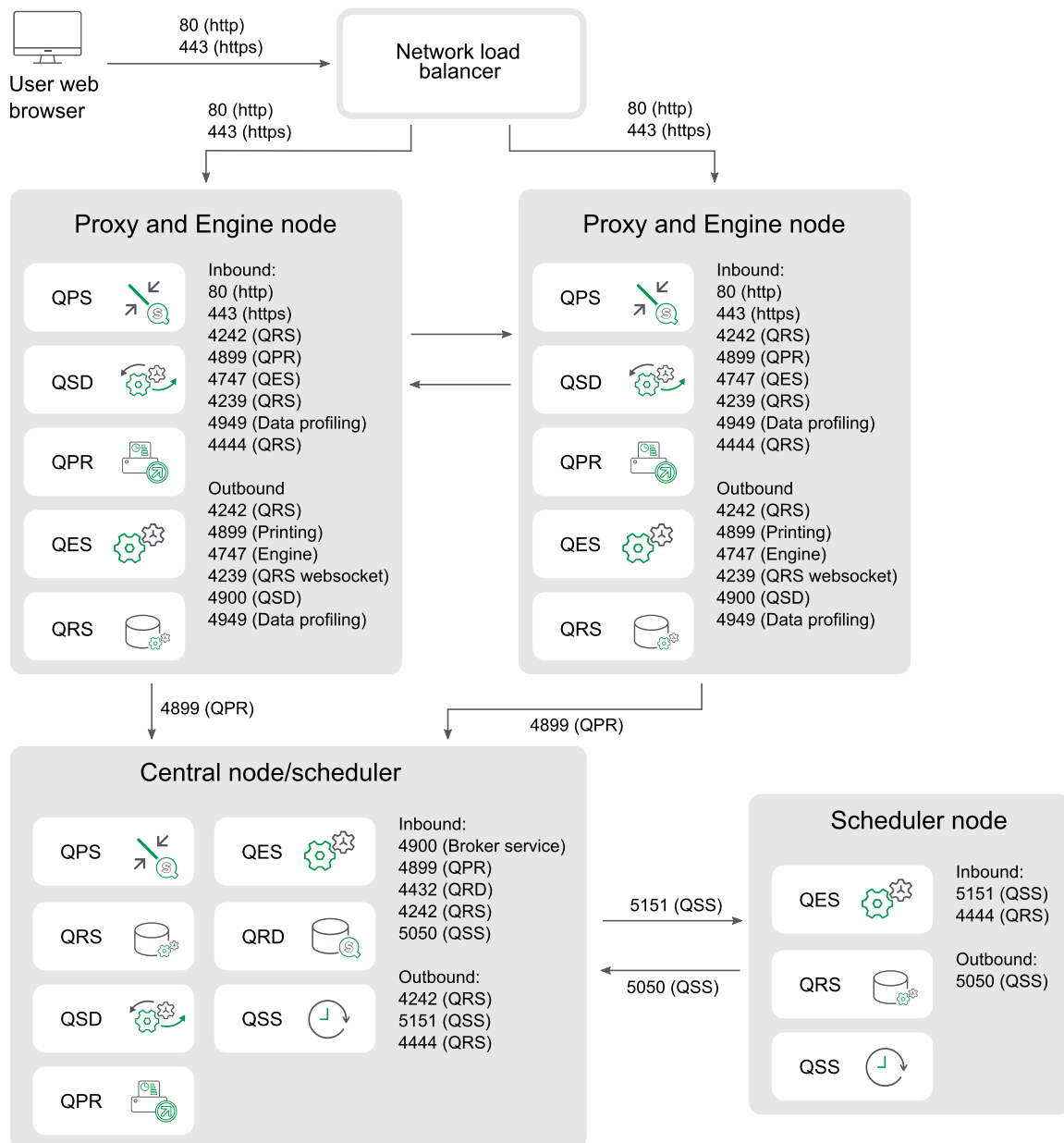


Separate scheduler node and high availability proxy and engine nodes

This example shows the ports that are used in a multi-node site when deploying a separate scheduler node and more than one proxy and engine node. The proxy load balancing excludes the engine on the central node.

1 Planning your Qlik Sense Enterprise deployment

Inbound ports indicate the listening ports for the services running on each node. Firewall rules must allow inbound traffic to these ports. Outbound ports indicate the destination of the communication from one node to other nodes in the environment. Firewall rules must allow the node to send outbound traffic to these outbound ports.

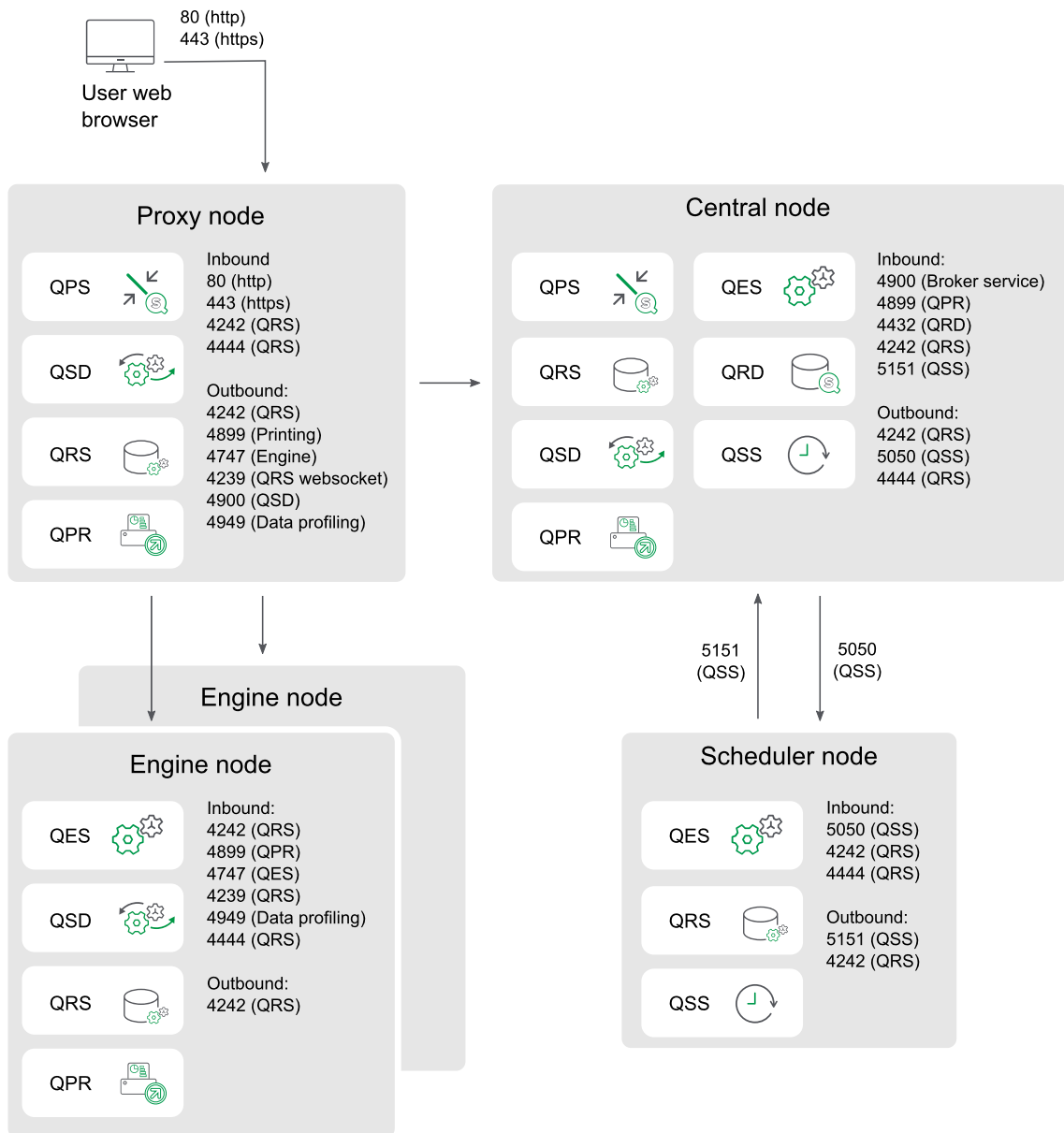


Separate proxy and scheduler nodes and high availability engine nodes

This example shows the ports that are used in a multi-node site when deploying separate proxy and scheduler nodes and more than one engine node. The proxy load balancing excludes the engine on the central node.

1 Planning your Qlik Sense Enterprise deployment

Inbound ports indicate the listening ports for the services running on each node. Firewall rules must allow inbound traffic to these ports. Outbound ports indicate the destination of the communication from one node to other nodes in the environment. Firewall rules must allow the node to send outbound traffic to these outbound ports.

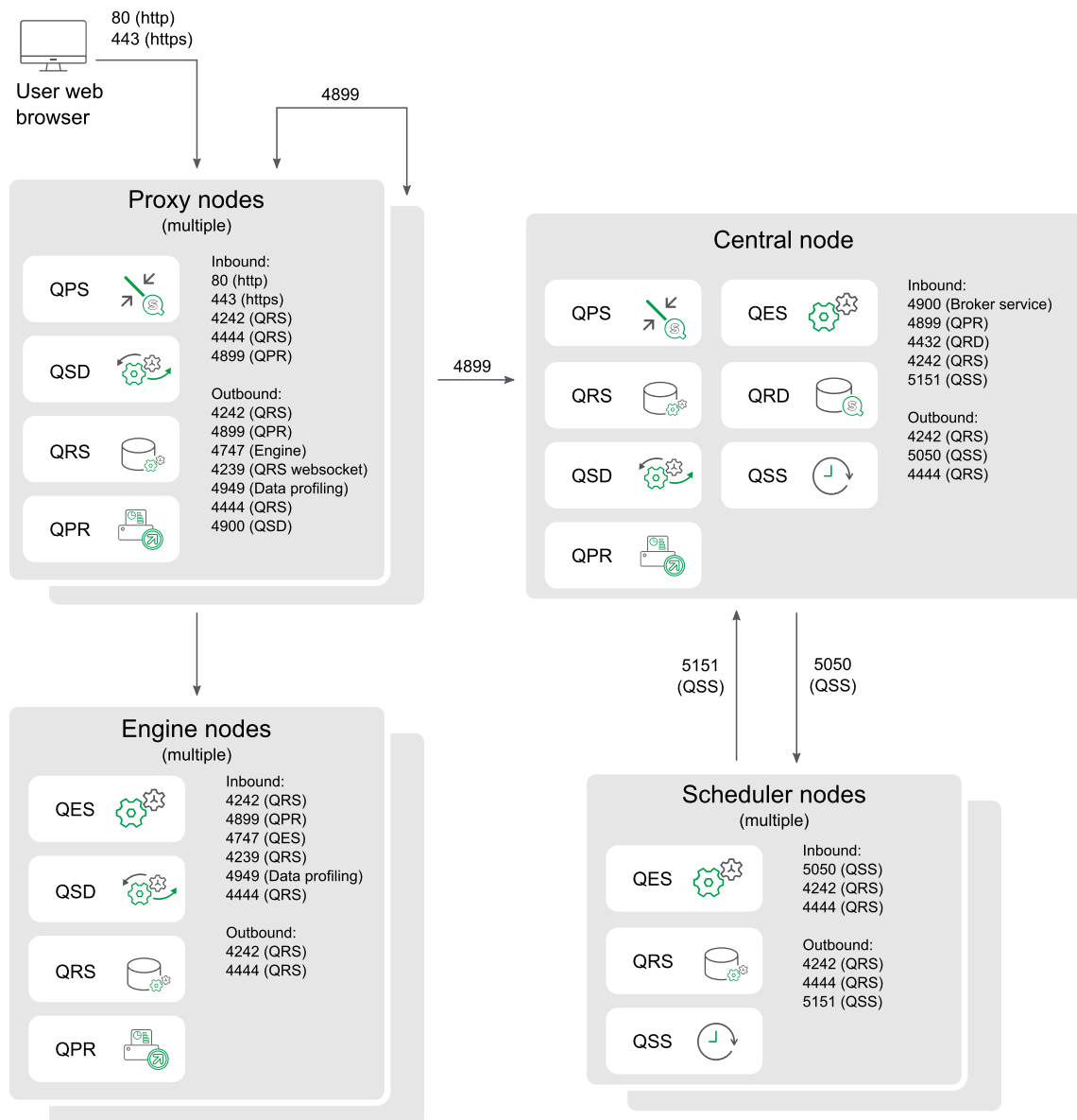


Generic scale out

This example shows the ports that are used in a multi-node site when scaling the site by adding additional proxy, engine, or scheduler nodes. The proxy load balancing excludes the engine on the central node.

1 Planning your Qlik Sense Enterprise deployment

Inbound ports indicate the listening ports for the services running on each node. Firewall rules must allow inbound traffic to these ports. Outbound ports indicate the destination of the communication from one node to other nodes in the environment. Firewall rules must allow the node to send outbound traffic to these outbound ports.



Persistence

A Qlik Sense site stores data to both a repository database and a file share. The repository database stores system and app meta data, while the file share stores binary application data such as data models and app content. In a single node deployment, both the repository database and the files share are usually located on the same machine as the Qlik Sense services. In a multi-node deployment, a cluster is formed around a single repository database and file share. In many cases these may be on separate dedicated servers to improve resilience or performance.

1 Planning your Qlik Sense Enterprise deployment



For best performance we recommend that you locate all your Qlik Sense servers in the same geographic location or data center as the repository database and the file share with a network latency below 4 milliseconds.

File share

In a Qlik Sense site, a file share is used to store the binary application data, including data models and the app content. It can be located on any one of the nodes in the Qlik Sense site or on a dedicated server for better resilience and performance. You create this folder before you install Qlik Sense. See: *Creating a file share (page 126)*

The requirements for the share are:

- The Qlik Sense nodes in the cluster must have network latency below 4 milliseconds to connect to the file share server. Performance can degrade if this is not the case.
- The bandwidth to the file share must be appropriate for the amount of traffic on the site. The frequency and size of the apps being saved after reloading, and opened into memory, drives this requirement. 10 Gigabit networking is recommended.
- The file share can run on:
 - A Windows Server OS. The Windows server may have storage allocated to it from a storage area network (SAN), use local disks, or virtual storage in the case of a virtual machine.
 - A non-Windows device such as a Linux server or hardware NAS device that supports SMB 3.0.
 - An NFS (Network File System) share hosted on a Windows Server. See: *Creating an NFS file share (page 127)*.



Qlik periodically runs network file share performance tests on Qlik Sense using WinShare, and FreeNAS with SMB 3.0. For more information on network file share solutions, contact your Qlik representative.

- The file storage must have a single read and write master. Storage can be replicated to standby storage, but only one location can be used to read and write to.



If your storage is on-premises, Qlik recommends using an enterprise SAN storage solution. SAN storage can be customized for your specific storage requirements and is reliable for throughput and latency to large blocks of storage.

If you are using a Virtual Private Cloud (VPC) deployment, consider the size and uses of your apps. A Windows-based file server will run better in a storage-optimized virtual machine than a general purpose instance. Consider using block storage services instead of Windows-based to take advantage of scaling and resilience offered by those providers.

The <Qlik Sense shared folder>\Apps folder contains app data and includes subfolders such as *DataPrepAppCache*, which stores QVD files for caching loaded data, and a server-named subfolder that is automatically created when apps are imported from the QMC.

1 Planning your Qlik Sense Enterprise deployment

Repository database

In a Qlik Sense site, a PostgreSQL repository database is used to store all data for the Qlik Sense Repository Service, including system and meta data. It can be located on one of the nodes in the Qlik Sense site or on a dedicated server for better resilience and performance. If you want to install it on a dedicated server, you do this before installing Qlik Sense.

You have two options for the repository database:

- Install as a local database on a central node. This option can be used for both single-node and multi-node deployments, and is done during installation using the Qlik Sense setup program.
- Install as a remote database on a separate server. This option provides higher performance and resilience, and is the recommended approach in a multi-node deployment. See: *Installing and configuring PostgreSQL (page 132)*

The requirements for the database are:

- The Qlik Sense nodes in the cluster must have network latency below 4 milliseconds to connect to the repository database server. Performance can degrade if this is not the case.
- If you run a PostgreSQL database on a dedicated server, it must use PostgreSQL version 12.x.



PostgreSQL can be run on various platforms including Windows, Linux, or cloud-hosted services such as Amazon RDS. If you use Linux or Amazon RDS, it is your responsibility to install and configure a running instance of PostgreSQL for Qlik Sense to use.

Performance

This topic aims to provide some basic information on performance to consider before you install Qlik Sense Enterprise on Windows. There are several different considerations to think about when planning your Qlik Sense Enterprise on Windows deployment:

- Size of deployment - small single-node, medium, or large multi-node site?
- Number of nodes in your site?
- Local or dedicated repository database?
- Local or network file share?
- Number of CPU cores required for each node?
- RAM required for each node?

We also recommend scalability testing and engaging with Qlik consulting services for larger deployments.

Capacity and performance

Qlik Sense supports up to a maximum of 12 nodes. In addition to the number of nodes, there are other factors that contribute to total capacity:

- Workload
- Hardware speed

1 Planning your Qlik Sense Enterprise deployment

- Network speed

For example, if the disk speed of the file share and the central node is too slow, you may expect low performance during some operations, such as importing or duplicating apps.

DMZ deployments

All nodes in a site, including nodes without an engine, require access to both the database and file share. In demilitarized zone (DMZ) deployments this may require opening additional ports, or taking an alternative approach, compared to a DMZ deployment with synchronized persistence.

Geographical deployments

The current persistence model does not support geographical deployments. For best performance we recommend that you locate all your Qlik Sense servers in the same geographic location or data center as the repository database and the file share with a network latency below 4 milliseconds.

Central node dependencies

The central node is responsible for handling a number of vital operations on your site. If the central node fails, some operations will fail to run, including:

- Manager scheduler - responsible for triggering reloads
- License distribution - allowing new users to obtain a license
- Extension objects

To reduce the dependency on the central node you can configure one or more nodes as a failover candidate. For more information, see *Configuring failover for central node resiliency (page 129)*.

User accounts

In order to successfully install and deploy Qlik Sense you must set up some user accounts before you start your Qlik Sense installation.

Windows user accounts are created and configured using your Windows server administration tools.

If you choose to manually install and configure your PostgreSQL repository database, users are created and configured using your PostgreSQL database administration tools. If you choose to have Qlik Sense install the repository database for you, the Qlik Sense setup wizard will create the users during installation.

The following are the users that you may need to create before you install Qlik Sense:

- Windows Qlik Sense services administrator
- Windows Qlik Sense services user that is not an administrator
- PostgreSQL database superuser
- Qlik Sense Repository Database administrator

You must create the required Windows user accounts before you install Qlik Sense because you are prompted to enter them during the installation. If you choose to install as a Windows local administrator and wish to change to a Windows dedicated Qlik Sense service user after installation, see *Changing the user account to run Qlik Sense services (page 153)*.

1 Planning your Qlik Sense Enterprise deployment

When you create your Windows user accounts you must set a password for each one. Windows user account passwords may expire in accordance with the Windows domain security rules settings. If you do not update the passwords for each Windows service setting, the services will stop working. To avoid this, you can select the **Password never expires** check box in the Windows user profile, if your security protocol allows it.

Windows Qlik Sense services administrator

We recommend that you use a dedicated Windows user account to run the Qlik Sense services. If your dedicated Windows Qlik Sense services user is an administrator, you can login as that user to install Qlik Sense. If your dedicated Windows Qlik Sense services user is not a local administrator, you must use an administrator account to install Qlik Sense.

Windows Qlik Sense services user that is not an administrator

If you wish to use a dedicated Windows user account that is not an administrator to run the Qlik Sense services, you must create that account before you install Qlik Sense. The Windows Qlik Sense services user runs the following services:

- Qlik Sense Repository Service
- Qlik Sense Proxy Service
- Qlik Sense Engine Service
- Qlik Sense Scheduler Service
- Qlik Sense Printing Service
- Qlik Sense Service Dispatcher

For more information about services, see *Services (page 30)*.

The Windows Qlik Sense services user that is not an administrator must meet the following requirements:

- Be a member of the **Qlik Sense Service Users** and **Performance Monitor Users** groups. You add the Windows Qlik Sense services user that is not an administrator to these groups after you install Qlik Sense.
- Have **Log on as a service** rights.
- Only be used for Qlik SenseWindows services. This is necessary to avoid conflicts with other Windows services in the same computer.

PostgreSQL database superuser

The PostgreSQL database superuser is a role that bypasses all permission checks, except the right to log in. It is not a Windows, or Qlik Sense user, it is a PostgreSQL user configured in the repository database.

If you choose to install the PostgreSQL database manually, you are prompted to create a PostgreSQL database superuser and password during installation. That user ID and password are used to connect your PostgreSQL database. For details about creating users with the PostgreSQL administration tools, see *Installing and configuring PostgreSQL (page 132)*.

If you choose to install the Qlik Sense Repository Database locally during the Qlik Sense installation, the PostgreSQL installation is done automatically.

1 Planning your Qlik Sense Enterprise deployment



When you install Qlik Sense, if you select the *Install local database option*, the QSR, SenseServices, and QSMQ databases are created automatically. These databases also share the same PostgreSQL login role. For more information, see *Installing and configuring PostgreSQL (page 132)*

Qlik Sense Repository Database administrator

The Qlik Sense Repository Database administrator role has full access to the Qlik Sense Repository Database that contains all configuration data for the Qlik Sense site. It is not a Windows, or Qlik Sense user, it is a PostgreSQL user configured in the repository database.

If you choose to install PostgreSQL manually, the Qlik Sense Repository Database administrator is also created manually using the PostgreSQL administration tools. For details about creating users with the PostgreSQL administration tools, see *Installing and configuring PostgreSQL (page 132)*. You must enter the location of the Qlik Sense Repository Database and the login credentials for the Qlik Sense Repository Database administrator during the Qlik Sense setup on the **Shared persistence database connections settings** page.

If you choose to install the Qlik Sense Repository Database locally using the Qlik Sense setup, you are prompted to set a user name and password for the Qlik Sense Repository Database administrator during the setup.

You must keep that password for backup and restore activities. It may also be needed for support.

1.4 Qlik Sense Enterprise deployment examples

This section provides examples of different ways to deploy Qlik Sense Enterprise. The examples are not guidelines or best practices for how to install your deployment, rather they are to provide a high-level view of how different organizational needs are achieved by different deployment scenarios.

Qlik Sense Enterprise on Windows deployments

A Qlik Sense Enterprise on Windows deployment can be a single-node (server) site or a multi-node site. All Qlik Sense sites must have a repository database, a single Qlik license key, and a common set of configuration data and apps.

Single-node sites

A single-node site is the smallest site possible and consists of a single node (single server). In a single-node site, site administration, app development, and app consumption happens on the same node. All Qlik Sense services, the repository database, and the file share are hosted on the same node.

Multi-node sites

Multi-node sites offer more scalability options for large organizations. In a multi-node deployment, the Qlik Sense site is distributed across several nodes. Nodes are connected to a common repository database, they share a common set of data, and they share the same license key. In larger sites, you can add nodes to improve scalability, capacity, and resilience. In a multi-node site, there is at least one central node and one or more rim nodes that are connected to the central node.

Benefits of multi-node sites include:

- Better scalability, making it easier to increase capacity.
- Improved resilience and reliability.
- Ability to move apps or roles to specific nodes.
- Flexibility to suit customer network deployments.

Deploying Qlik Sense Enterprise on Windows on a cloud platform

You can deploy Qlik Sense Enterprise on Windows on a cloud infrastructure platform, such as Amazon AWS and Microsoft Azure, to take advantage of cloud-native scalability, low maintenance storage options, and high reliability.

Qlik Sense Enterprise SaaS deployments

Qlik Sense Enterprise SaaS is a full SaaS deployment option for Qlik Sense Enterprise. The cloud infrastructure is hosted and managed by Qlik. Qlik Sense Enterprise SaaS may be deployed independently or as part of a multi-cloud deployment. See, *Qlik Sense Enterprise on Windows multi-cloud deployments (page 66)*.

Qlik Sense Enterprise on Windows multi-cloud deployments

Qlik Sense multi-cloud deployments refer to a deployment where one or more cloud instances are connected to a Qlik Sense Enterprise on Windows site. Qlik Sense multi-cloud deployments are multi-node sites.

Benefits of multi-cloud sites include:

- Central node is managed on-premises.
- Users consume and develop apps from the cloud.
- Authentication for both on-premises and cloud is handled by a single identity provider.

Qlik Sense Enterprise on Windows on-premises

You can configure a Qlik Sense Enterprise on Windows deployment to meet the specific needs of your organization. As your requirements for performance and scalability increase, so too will the size of your deployment.

The following terms are used in the deployment scenarios:

- Central node: the central point for managing all nodes in a site.
- Failover candidate node: a redundant node that becomes the central node if the original central node fails.

1 Planning your Qlik Sense Enterprise deployment

- Scheduler or Reload node: reloads apps on a schedule, but does not serve content to users.
- Consumer node: serves apps to users, but is not used to create, process, or reload data.
- Development node: allows users to create and reload new apps, but does not serve normal consumer traffic.
- Proxy node: provides load balancing of user traffic to other nodes but does not contain a Qlik Sense Engine Service (QES).



An alternative to using a proxy node is to have a proxy installed on each consumer node and balance the traffic using a hardware load balancer.

Deployment scenarios

This section provides four deployment scenarios of Qlik Sense Enterprise on Windows deployments. The deployments described here are examples of a small, medium, large, and extra-large Qlik Sense Enterprise on Windows scenarios. These examples provide an approximation of the type of workload a particular deployment might need to handle. The figures are not intended to set a minimum or maximum limit on your deployment.

If you expect to have performance demands higher than any of the figures below (such as more reloads or apps) then contact your Qlik partner and perform a full sizing exercise. For more general scalability and performance information, see *Performance (page 62)* and [QMC performance - best practices](#).

The following table provides some basic assumptions for each type of deployment scenario:

Deployment type assumptions

Item	Single-node (small)	Multi-node (medium)	Multi-node (large)	Multi-node (extra-large)
Apps	50	100	1000	1000
Active apps per day	25	50	125	125
Total users (from UDC)	500	1000	50000	50000
Concurrent users (equals active users within the same hour)	50	100	500	1000
Maximum concurrent users in the QMC	2	2	5	10
Average app size (in gigabytes)	0.1	0.1	0.1	0.1
Maximum app size (in gigabytes)	1	2	5	5
Content creation (objects per hour)	20	40	50	50
Reloads per hour	10	20	400	400

1 Planning your Qlik Sense Enterprise deployment



The difference between a large and extra-large deployment in our examples is the number of concurrent users.

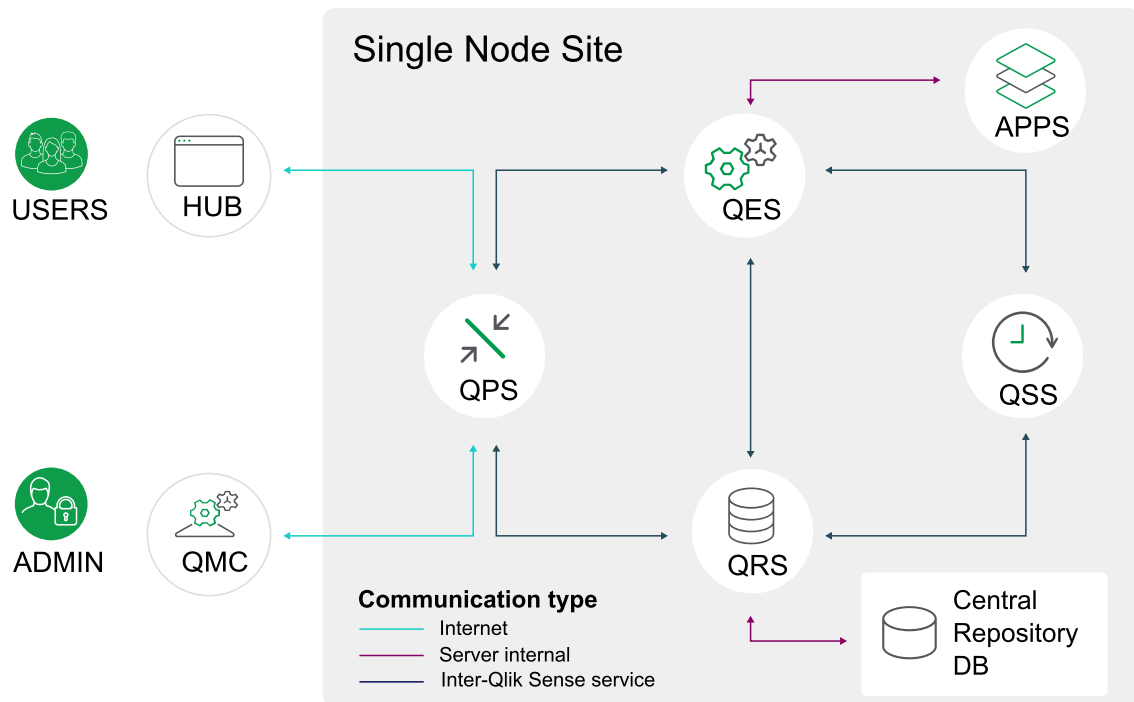
Single-node (small)

The following deployment example shows a Qlik Sense Enterprise on Windows single-node site.

In a single-node site, the Qlik Sense services are all running on the same node.

- Qlik Sense Repository Service
There is only one instance of the Qlik Sense Repository Service (QRS) running, and it has direct access to the central repository database.
- Qlik Sense Scheduler Service
The Qlik Sense Scheduler Service (QSS) acts as both manager and worker.

This kind of deployment works best in a single time zone, where data reloads can be done at night.



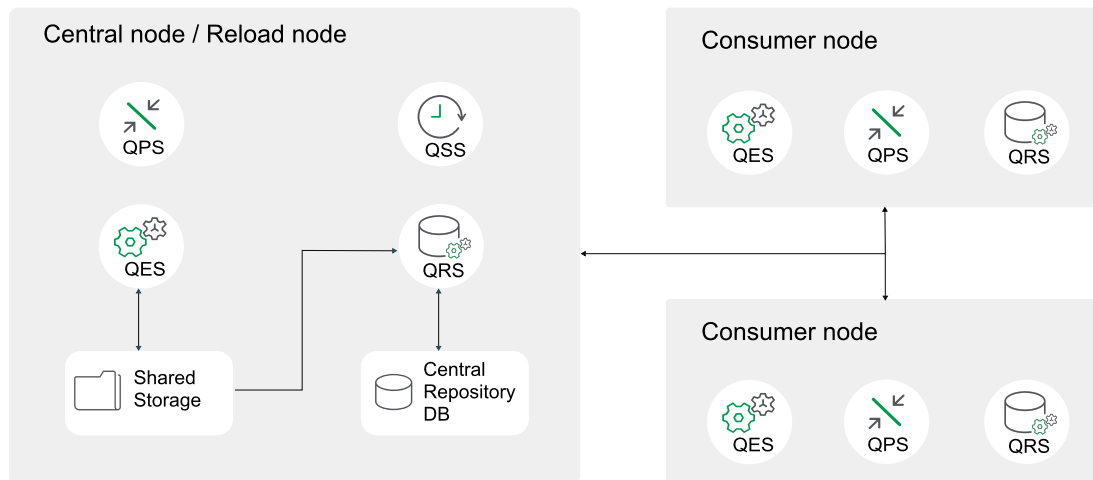
Multi-node (medium)

The following deployment example shows a medium-size, multi-node Qlik Sense Enterprise on Windows production deployment comprising three nodes:

- One central node/reload node on which the Qlik services are running.
- Two consumer nodes to load balance user demand.

In this configuration, the central repository database, the file share, and the other Qlik Sense services are running on the central node. The two consumer nodes handle app consumption.

1 Planning your Qlik Sense Enterprise deployment



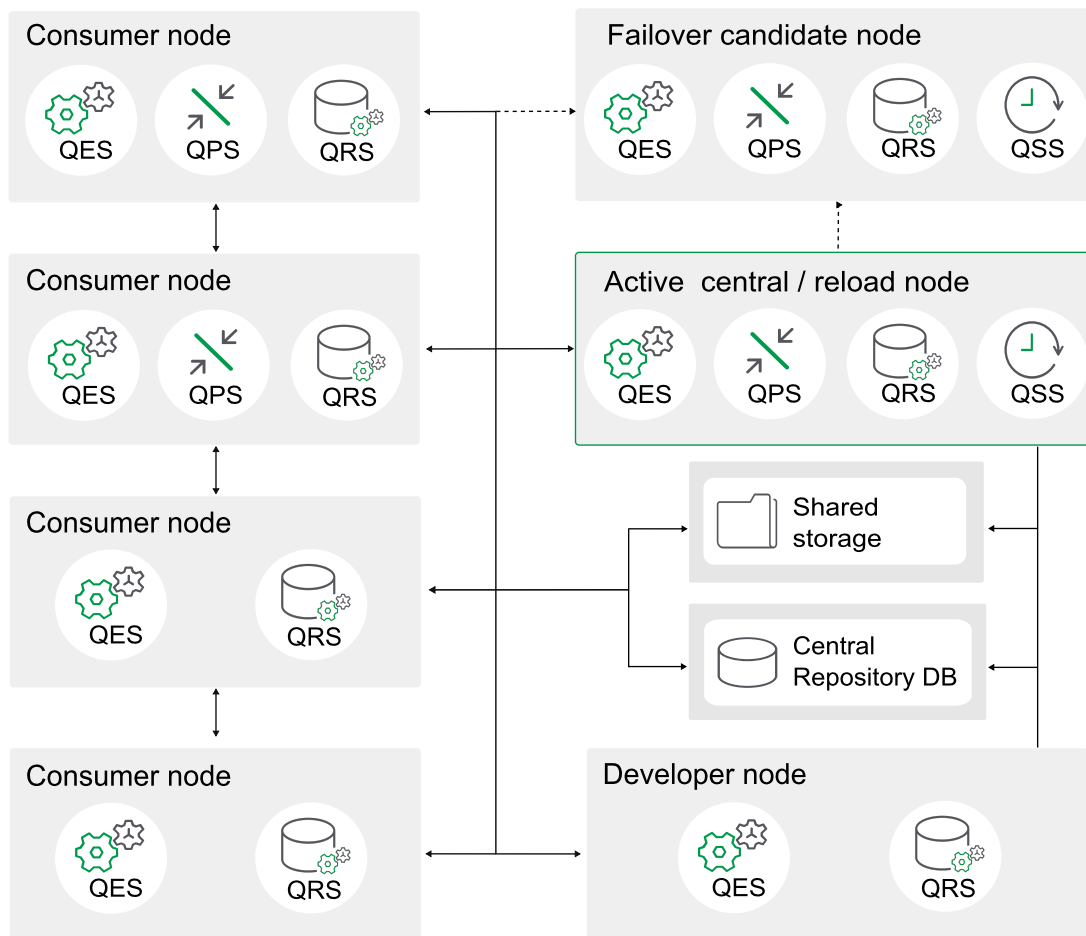
Multi-node (large)

The following deployment example shows a large, multi-node Qlik Sense production deployment.

A deployment like this provides the ability to scale up app reloads and user load. This deployment consists of the following nodes:

- Central node/reload node to handle the services
- Failover candidate node to handle the services if the central node fails.
- Four consumer nodes to load balance user demand.
- One developer node for app development.
- The repository database (PostgreSQL) and the file share are installed on separate, dedicated servers.

1 Planning your Qlik Sense Enterprise deployment



The central node and failover node must have all services installed. Configure the proxy service on consumer nodes to handle user traffic, and on both the central and failover nodes to handle admin traffic.



The Qlik services on both central and failover nodes are always active.

Multi-node (extra large)

The following deployment example shows an extra large, multi-node Qlik Sense production deployment consisting of seven consumer nodes providing the ability to scale up app reloads and user load. Groups of consumer nodes are dedicated to different size apps. Each consumer node can be configured with security and custom load balancing rules to restrict the size of the apps they can serve.

To ensure that the system can cope with the load, you can pre-load some apps in memory. For example, you could pre-load all medium and large sized apps, ensuring that they can be loaded in less than two seconds, even during peak hours.

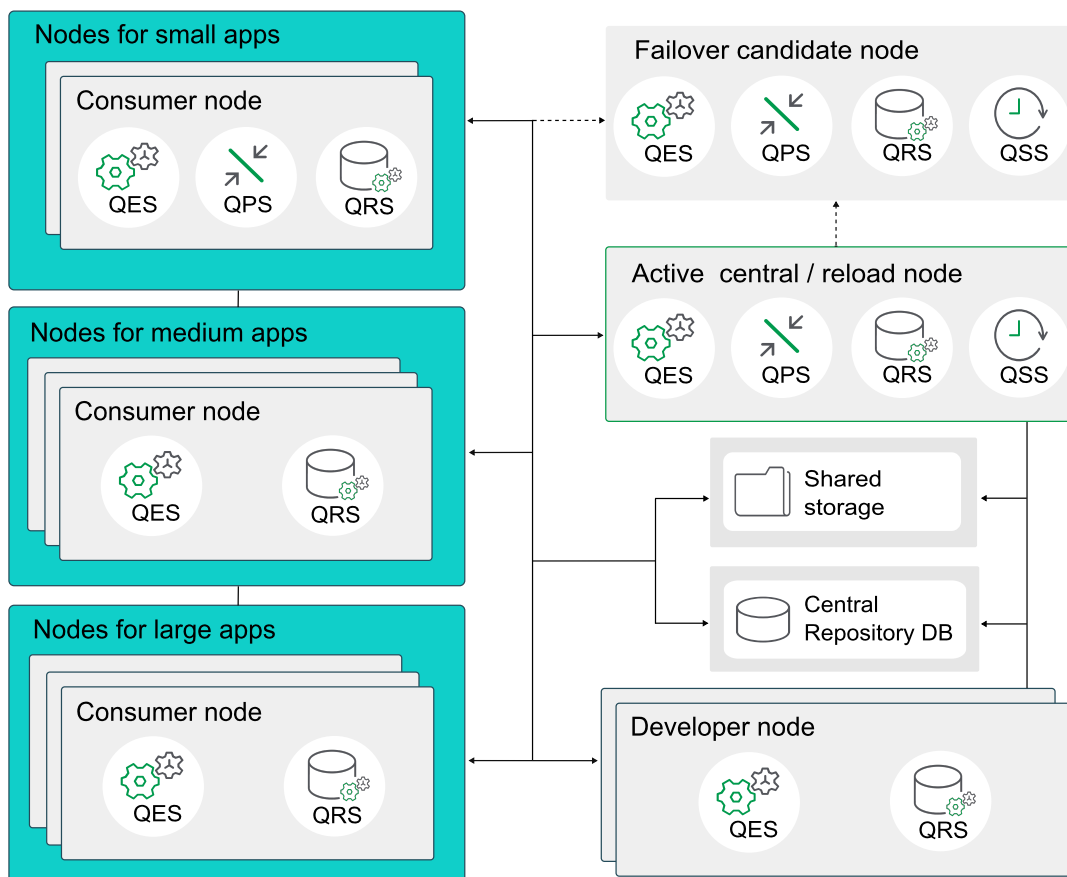
1 Planning your Qlik Sense Enterprise deployment



With very large deployments, development of applications can be resource intensive. It may therefore be appropriate to have a separate deployment dedicated to app development. If you prefer to keep developer and consumer nodes in the same deployment, ensure the resource limits are suitable for the developer nodes. This includes reload time, hyper cube timeout, and amount of RAM.

This deployment consists of the following nodes:

- Active central node/reload node to handle the services.
- Failover candidate node to handle the services if the central node fails.
- Seven consumer nodes with node clusters dedicated to app size.
- Two developer nodes for app development.
- The repository database (PostgreSQL) and the file share are installed on separate, dedicated servers.



The central node and failover node must have all services installed. Configure the proxy service on consumer nodes to handle user traffic, and on both the central and failover nodes to handle admin traffic.



The Qlik services on both central and failover nodes are always active.

Qlik Sense Enterprise on Windows deployed to AWS

In an Amazon Web Services (AWS) deployment, you install Qlik Sense Enterprise on an Amazon virtual private cloud infrastructure that is flexible, high performance, and quick to set up.

Deploying Qlik Sense Enterprise on AWS will enable you to quickly add new applications in a simple and scalable manner. You can do this with a basic knowledge of AWS security and scalability options but without the need to follow complex on-premise installation and configuration procedures. Using AWS will enable you to get your Qlik Sense infrastructure up and running in fraction of the time required for an on-premise deployment, and will enable you to scale your deployment quickly and easily, regardless of unexpected changes in demand.

You can deploy Qlik Sense to AWS manually, or you can use an Amazon Machine Image (AMI) available in the AWS Marketplace that includes Qlik Sense preinstalled. However, predefined images do not include a file share, so can only support single node Qlik Sense deployments.

Benefits of using AWS cloud

- A quick and effective way of deploying Qlik Sense to the cloud.
- Simple and cost-effective, reducing overall deployment times.
- Quick and easy to deploy Qlik Sense applications.
- Fewer hardware management overheads.
- Scalable, elastic storage that can be expanded and contracted on demand.
- Geographic deployment to multiple regions around the world makes lower latency possible.
- A reliable and high performance platform.

Components

To successfully deploy Qlik Sense on AWS cloud you need a basic understanding of the architecture and services available in an AWS deployment. As part of a Qlik Sense deployment on AWS, you need the following components:

- An Amazon AWS account
- Amazon Management Console - available when you log in to your AWS account.
- VPC - Amazon Virtual Private Cloud
- EC2 - Amazon Elastic Cloud instance running on a VPC. Lets you scale your deployment by adding or removing servers as your requirements change.

AWS services

You should also have a basic understanding of other AWS services that you can use for managing resources and as data stores for your Qlik Sense applications:

- RDS - Managed relational database service as an alternative to a PostgreSQL repository database. Provides high availability without the same complexity.

1 Planning your Qlik Sense Enterprise deployment

- S3 - Simple Storage Service. Scalable, object-based cloud storage.
- Dynamo DB - NoSQL database service
- Elastic IP - remapping of IP addresses
- EMR - Elastic MapReduce. Managed Hadoop service
- Redshift - Data warehouse
- Cloud formation - for managing resources automatically

For more information about AWS services, see the  [Amazon AWS](#) website.

Microsoft Windows versions

Your AWS instance needs to be running a Microsoft operating system onto which you can install a Qlik Sense instance. Qlik Sense supports the following Windows operating systems for an AWS deployment:

- Microsoft Windows Server 2016
- Microsoft Windows Server 2019
- Microsoft Windows Server 2022

Qlik Sense Enterprise

Install a single-node Qlik Sense server on your EC2 instance.

Qlik Sense Enterprise configuration:

Use the QMC to configure the following:

- Licensing
 - Tokens (only token-based license)
 - User access (token-based license) or Professional access (user-based license)
 - CPU cores
- Security groups

Create a proxy setup for allowing HTTP access.

Other considerations

When you deploy Qlik Sense to AWS for the first time you should also consider the following.

Security

To configure security on an AWS deployment you need a good understanding of how to set up AWS security groups, key pairs, and also security groups in Qlik Sense. You use the Amazon Management Console to configure AWS security and the QMC to configure all security and authentication settings in Qlik Sense Server.

For more information about security, see *AWS and Azure security (page 248)*, and for more on Qlik Sense security, see *Qlik Sense Enterprise on Windows security (page 208)*

Connectivity

AWS web services that you can use as data stores for Qlik Sense applications to retrieve data from when building applications:

1 Planning your Qlik Sense Enterprise deployment

- Amazon DynamoDB – NoSQL database
- Amazon RDS – managed relational database service
- Amazon Redshift – data warehouse as a service
- Amazon Simple Storage Service (S3) – scalable, object-based cloud storage
- AWS Elastic Map Reduce (EMR) – managed Hadoop service

In an AWS deployment you can use the following connectivity mechanisms to connect to different data sources:

- ODBC connection
- OLE DB connection
- REST API connection
- Native connector to a specific source

Connectivity scenarios:

- Qlik Sense instance that uses both data stored in Amazon RDS and Amazon Redshift.
- Qlik Sense instance that uses data coming from an AWS data source as well as a combination between flat files and web based data sources (i.e. a web service data feed).
- Hybrid Qlik Sense instance - uses data stored in AWS data sources as well as data stored on premise.

Scalability

As environments grow in terms of number of users, number and size of applications, and number of data sources it is important to understand how to size the environment correctly and how to scale the environment accordingly. You need to create a multi-node environment to effectively scale up or down, by creating dedicated servers for different purposes. You can then allocate resources correctly across the following Qlik Sense services.

- Engine Service – The QIX engine, provides in-memory Associative Data Indexing and calculation supporting analysis
- Proxy Service – Manages authentication, handles user sessions and load balancing
- Repository Service – Manages Qlik Sense applications, controls access, and handles configuration
- Scheduling Service – Manages reloads of Qlik Sense applications and other scheduled tasks
- Service Dispatcher – Launch and manage the data profiling service for the data load model

For more information about scalability, see the [Qlik Sense Performance Benchmark](#) technical brief.

AWS deployment example

AWS provides a cloud infrastructure with all the services and computing power you need to provide a reliable cloud deployment platform for Qlik Sense that can perform regardless of unexpected changes in demand, and concurrency.

Qlik Sense single-node deployment on AWS

Components in a typical Qlik Sense single-node deployment on AWS:

1 Planning your Qlik Sense Enterprise deployment

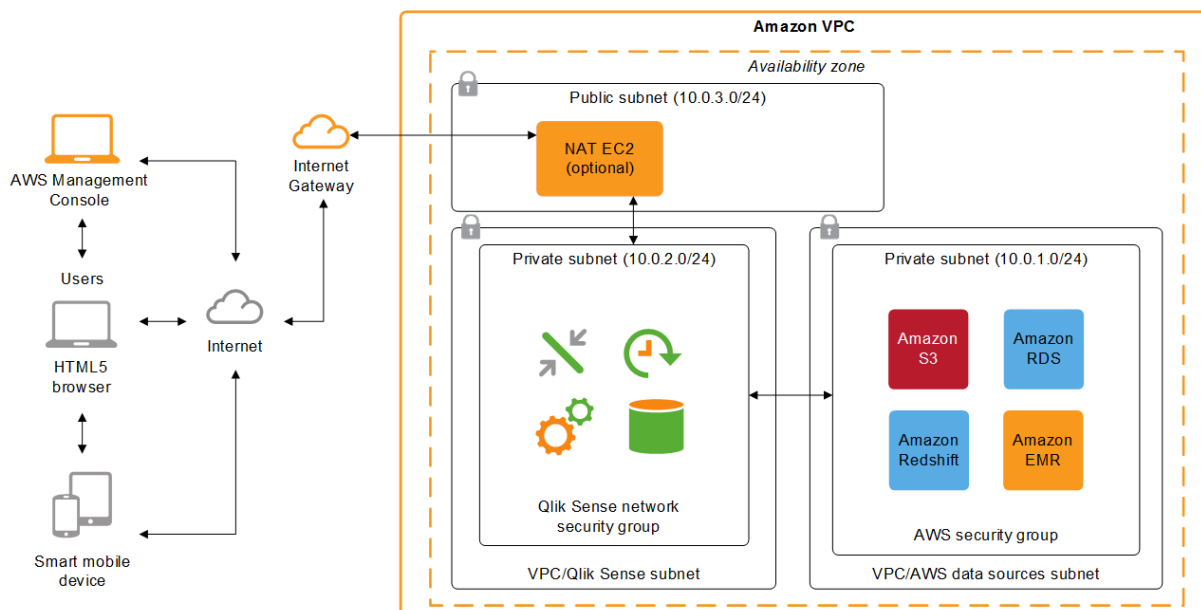
- VPC - Virtual Private Cloud. A logically isolated virtual network that shares a common security configuration that you define.
- Subnet - you need at least one subnet within the VPC. This could be a public, or private subnet.
 - Public subnet - subnet with direct access to the internet.
 - Private - a subnet that cannot be reached from the internet.
- RDS - Relational database service. Use this for the repository to provide high availability without the same complexity as a PostgreSQL database.
- NAT instance (optional) - restricts traffic to private subnets but allows outgoing traffic to the internet. For example, if an EC2 instance is launched inside the private network it can access the internet.
- Windows Server instance - deployed inside the default subnet to host your Qlik Sense installation.
- Security groups - act as a virtual firewall controlling which IP addresses can gain access to your instance. Use the Amazon Management Console to create a security group called *Qlik Sense*.
- Key pair - a `qlik_sense.pem` file that you create and store locally. This file handles authentication when you connect to your AWS instance.
- IAM - Identity and Access Management. You need IAM to manage the fine-grained permissions required for access to different AWS services.
- Qlik Sense Server node - a single node deployed on Windows Server inside the default subnet.

Deployment options:

- Qlik Sense node in a public subnet with direct Internet access.
- Qlik Sense node in a private subnet without Internet access.

The decision whether to choose a public or private subnet in your deployment depends on your overall solution requirements.

The following example shows a complete Qlik Sense Enterprise, single node deployment on Amazon Virtual Private Cloud.



Qlik Sense Enterprise on Windows deployed to Azure

In a Microsoft Azure deployment, you install Qlik Sense Enterprise on an Azure cloud infrastructure that is flexible, high performance, and is quick to set up.

Deploying Qlik Sense Enterprise on Azure will enable you to quickly add new applications in a simple and scalable manner. You can do this with a basic knowledge of Azure security and scalability options but without the need to follow complex on-premise installation and configuration procedures. Using Azure will enable you to get your Qlik Sense infrastructure up and running in a fraction of the time required for an on-premise deployment, and will enable you to scale your deployment quickly and easily, regardless of unexpected changes in demand.

You can deploy Qlik Sense to Azure manually, or you can use a Virtual Hard Disk (VHD) available in the Azure Marketplace that includes Qlik Sense pre-installed. However, predefined images do not include a file share, so can only support single node Qlik Sense deployments.

Benefits of using Microsoft Azure cloud

- A quick and effective way of deploying Qlik Sense to the cloud.
- Simple and cost-effective, reducing overall deployment times.
- Quick and easy to deploy Qlik Sense applications.
- Microsoft Server Message Block (SMB) 3.0 file system - This makes the Qlik Sense file share highly resilient to failures, and AWS does not offer a similar alternative.
- Scalable, reliable and high performance cloud platform.
- Microsoft security and networking functionality.
- Geographic deployment to multiple regions around the world makes lower latency possible.
- A reliable and high performance platform.

Components

To successfully deploy Qlik Sense on Azure cloud you need a basic understanding of the architecture, and services available in an Azure deployment. As part of a Qlik Sense deployment on Azure, you need the following components:

- Azure Virtual Machine
- Azure SMB 3.0 file system storage
- Azure Virtual Network
- Azure Resource Group
- Azure Resource Manager

Azure services

You should also have a basic understanding of other Azure services that you can use for managing resources and as data stores for your Qlik Sense applications:

- Azure Portal
- Azure Active Directory and Identity Management
- Azure SQL Database – SQL Server 2016 on the Cloud

1 Planning your Qlik Sense Enterprise deployment

- Azure SQL Data Warehouse – Enterprise level scale-out, massively parallel processing, highly scalable database for both relational and non-relational data.
- Azure Storage – scalable cloud storage (Blob Storage, Table Storage, Azure Queues and Azure Files)
- Azure HDInsight – elastic map reduce (Hadoop as Service)

For more information about Azure services, see the  [Microsoft Azure](#) website.

Microsoft Windows versions

Your Azure instance needs to be running a Microsoft operating system onto which you can install a Qlik Sense instance. Qlik Sense supports the following Windows operating systems for an Azure deployment:

- Microsoft Windows Server 2016
- Microsoft Windows Server 2019
- Microsoft Windows Server 2022

Qlik Sense Enterprise

Install a single-node Qlik Sense server on your Azure instance.

Qlik Sense Enterprise configuration:

Use the QMC to configure the following:

- Licensing
 - Tokens (only token-based license)
 - User access (token-based license) or Professional access (user-based license)
 - CPU cores
- Security groups

Create a proxy setup for allowing HTTP access.

Other considerations

When you deploy Qlik Sense to Azure for the first time you should also consider the following.

Security

Use the Resource Manager to configure Azure security and the QMC to configure all security groups and authentication settings in Qlik Sense.

For more information about security, see *AWS and Azure security (page 248)*, and for more on Qlik Sense security, see *Qlik Sense Enterprise on Windows security (page 208)*

Connectivity

Qlik Sense applications can use the following Azure web services as data stores:

- Azure SQL Database – SQL Server 2016 on the Cloud
- Azure SQL Data Warehouse – enterprise level scale-out, massively parallel processing, highly scalable database for both, relational and non-relational data
- Azure Storage – scalable cloud storage (Blob Storage, Table Storage, Azure Queues and Azure Files)
- Azure HDInsight – elastic map reduce (Hadoop as Service)

1 Planning your Qlik Sense Enterprise deployment

In an Azure deployment you can use the following connectivity mechanisms to connect to different data sources:

- ODBC connection
- OLE DB connection
- REST API connection
- Native connector to a specific source

Connectivity scenarios:

- Qlik Sense instance that uses data stored in Azure SQL Database and Azure SQL Data Warehouse.
- Hybrid Qlik Sense instance - uses data stored in Azure data sources as well as data stored on premise.

Scalability and sizing

As your environment grows in terms of number of users, number and size of applications, and the number of data sources, it is important to understand how to size and scale your environment correctly. Resources need to be allocated correctly across the following Qlik Sense services:

- Engine Service – The QIX engine, provides in-memory Associative Data Indexing and calculation supporting analysis
- Proxy Service – Manages authentication, handles user sessions and load balancing
- Repository Service – Manages Qlik Sense applications, controls access, and handles configuration
- Scheduling Service – Manages reloads of Qlik Sense applications and other scheduled tasks
- Service Dispatcher – Launch and manage the data profiling service for data load model and chart sharing between two users

For more information about scalability, see the [Qlik Sense Performance Benchmark](#) technical brief.

Azure deployment example

Microsoft Azure provides a cloud infrastructure with all the services and computing power you need to provide a reliable, cloud deployment platform for Qlik Sense that can perform regardless of unexpected changes in demand, and concurrency.

Qlik Sense single-node deployment on Azure

Components in a typical Qlik Sense deployment on Azure:

- Azure Virtual Network (VNet) - a logically isolated area of the Azure cloud where you can launch Azure resources in a virtual network that you define.
- Subnet - you need at least one subnet (either public or private) within the Virtual Network. This could be a public or private subnet.
 - Public subnet - subnet with direct access to the internet.
 - Private - a subnet that cannot be reached from the internet.
- Virtual Machine - A Windows Server virtual machine instance deployed in the default subnet onto which you can install and configure your instance of Qlik Sense server.

1 Planning your Qlik Sense Enterprise deployment

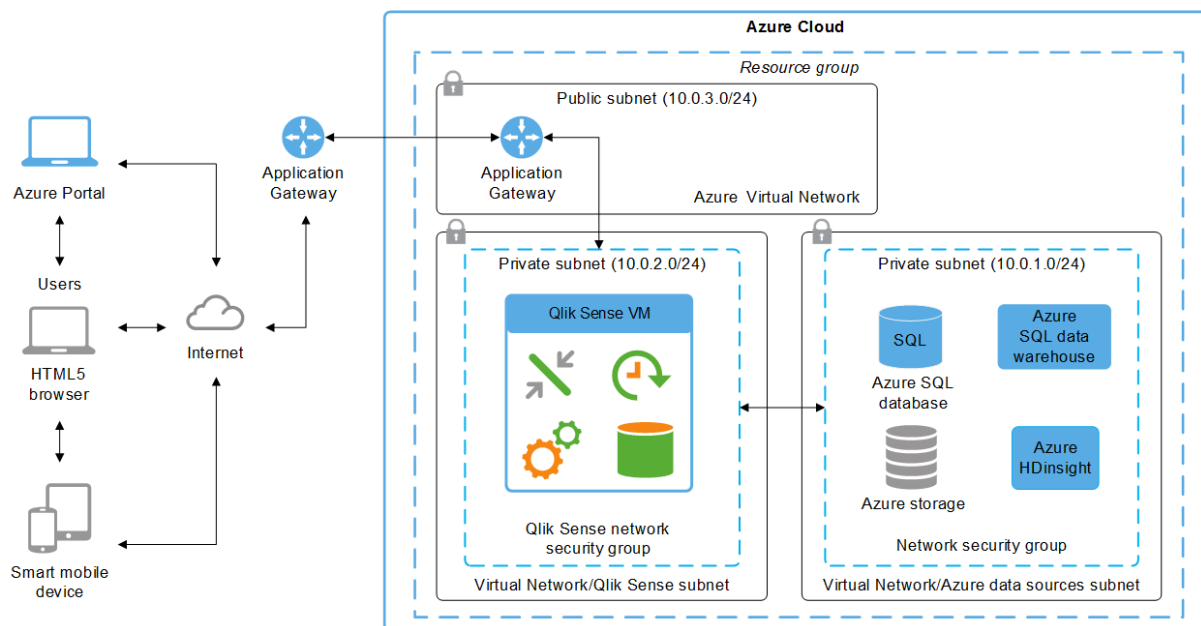
- Resource Group/Resource Manager - enables you to deploy, manage, and monitor the different components in your Microsoft Azure solution as a group. This makes it easier to deploy, update or delete components in a single, coordinated operation using the Resource Manager.
- Network Security groups - a list of Access Control List (ACL) rules that allow or deny network traffic to the Virtual Machine instances in a Virtual Network.
- Azure Active Directory and Identity Management - depending on the expected administration of the environment, integration with Azure Active Directory and Identity Management may be needed to manage fine-grained permissions for access to various Azure services involved in the deployment process.
- Qlik Sense Server node - a single node deployed on Windows Server inside the default subnet.

Deployment options:

- Qlik Sense node in a public subnet with direct Internet access.
- Qlik Sense node in a private subnet without Internet access.

The decision whether to choose a public or private subnet in your deployment depends on your overall solution requirements.

The following example shows a complete Qlik Sense Enterprise, single node deployment on Azure Cloud.



Qlik Sense Enterprise on Windows deployed to Google Cloud

In a Google Cloud deployment, you install Qlik Sense Enterprise on Windows on a Google Cloud infrastructure that is flexible, high performance, and is quick to set up.

Deploying Qlik Sense Enterprise on Windows on Google Cloud will enable you to quickly add new applications in a simple and scalable manner. You can do this with a basic knowledge of Google Cloud security and scalability options but without the need to follow complex on-premise installation and configuration

1 Planning your Qlik Sense Enterprise deployment

procedures. Using Google Cloud will enable you to get your Qlik Sense infrastructure up and running in a fraction of the time required for an on-premise deployment, and will enable you to scale your deployment quickly and easily, regardless of unexpected changes in demand.

Components


To successfully deploy Qlik Sense Enterprise on Windows on Google Cloud you need a basic understanding of the architecture, and services available in a Google Cloud deployment. As part of a Qlik Sense deployment on Google Cloud, you need the following components:

- Compute Engine (GCE)
- Persistent Disk
- VPC - Virtual Private Cloud
- Cloud VPN

Google Cloud services

You should also have a basic understanding of other Google Cloud services that you can use for managing resources and as data stores for your Qlik Sense applications:

- Cloud Deployment Manager
- Cloud SQL
- Persistent Disk
- Google BigQuery

For more information about Google Cloud services, see the  [Google Cloud](#) website.

Microsoft Windows versions

Your Google Cloud instance needs to be running a Microsoft operating system onto which you can install a Qlik Sense instance. Qlik Sense supports the following Windows operating systems for a Google Cloud deployment:

- Microsoft Windows Server 2016
- Microsoft Windows Server 2019
- Microsoft Windows Server 2022

Qlik Sense Enterprise

Install a single-node Qlik Sense Enterprise on Windows server on your Google Cloud instance.

Qlik Sense Enterprise configuration:

Use the QMC to configure the following:

- Licensing
 - Tokens (only token-based license)
 - User access (token-based license) or Professional access (user-based license)
 - CPU cores
- Security groups

Create a proxy setup for allowing HTTP access.

1 Planning your Qlik Sense Enterprise deployment


Other considerations

When you deploy Qlik Sense to Google Cloud for the first time you should also consider the following.

Security

Use the QMC to configure all security groups and authentication settings in Qlik Sense. You can make your on-premise Active Directory available in the VPC through an IPsec VPN tunnel to an external IdP with Google Cloud VPN.

For more information about security, see *Qlik Sense Enterprise on Windows security (page 208)*

For more information about Google Cloud security, see  [Google Cloud Security](#).

Connectivity

Qlik Sense applications can use the following Google Cloud web services as data stores:

- Cloud SQL for PostgreSQL
PostgreSQL version 11.5 is required.
- Google BigQuery
- Persistent Disk

In a Google Cloud deployment you can use the following connectivity mechanisms to connect to different data sources:

- ODBC connection
- OLE DB connection
- REST API connection
- Native connector to a specific source

Connectivity scenarios:

- Qlik Sense instance that uses data stored in Cloud SQL and Google BigQuery.
- Hybrid Qlik Sense instance - uses data stored in Google Cloud data sources as well as data stored on premise.



The following setup is not supported by the Qlik Sense Service Dispatcher: Adding an SSL connection towards a GCP Postgres database used for the Repository database to configure the repository service to communicate with the instance using certificate authentication with the provided Google certificate files.

Workaround: Install the Cloud SQL Auth proxy on each instance and reconfigure Qlik Sense on all services to communicate with the "localhost" instance of the proxy, which then hosts the communication secured towards the cloud.

1 Planning your Qlik Sense Enterprise deployment

Scalability and sizing

As your environment grows in terms of number of users, number and size of applications, and the number of data sources, it is important to understand how to size and scale your environment correctly. Resources need to be allocated correctly across the following Qlik Sense services:

- Engine Service – The QIX engine, provides in-memory Associative Data Indexing and calculation supporting analysis
- Proxy Service – Manages authentication, handles user sessions and load balancing
- Repository Service – Manages Qlik Sense applications, controls access, and handles configuration
- Scheduling Service – Manages reloads of Qlik Sense applications and other scheduled tasks
- Service Dispatcher – Launch and manage the data profiling service for data load model and chart sharing between two users

For more information about scalability, see the [Qlik Sense Performance Benchmark](#) technical brief.

Preparing your Google Cloud platform to install Qlik Sense Enterprise on Windows

This section describes how to set up and configure a single Google Compute Engine instance before installing Qlik Sense Enterprise on Windows as a single node site. A Compute Engine instance is a virtual machine that is hosted on Google Cloud infrastructure. To deploy Qlik Sense Enterprise on Windows in a production-ready, multi-node environment, you need to deploy your Qlik Sense Enterprise on Windows across several virtual machines.

Before you begin


- Download Qlik Sense Enterprise on Windows from [Product Downloads](#). For more information, see *Downloading installation files (page 17)*.
- Obtain a Qlik product license. To learn about the licensing options, see *Qlik product licenses (page 9)*.
- Log into your [Google Cloud Platform console](#).

Considerations for a multi-node deployment

Your business requirements determine the best way to deploy Qlik Sense Enterprise on Windows on Google Cloud. These instructions describe the basic configuration for a single Google Compute Engine instance. For larger multi-node sites, you should determine the architecture requirements of your multi-node deployment before you begin. Here are some important things to consider when planning a multi-node deployment:

- Ensure your Google Cloud region is geographically close to your on-premises network and data sources when possible. This reduces network latency.
- Deploy your Google Cloud cluster across zones. This helps you build a highly available solution. For more information, see, [Overview of the high availability configuration](#).
- Take advantage of the native Google Cloud Platform components, such as Google Cloud SQL Databases, for your Qlik Sense Enterprise on Windows deployment. For more information see, [Cloud SQL features](#).

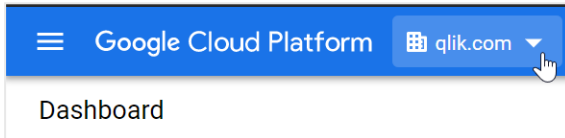
1 Planning your Qlik Sense Enterprise deployment

 *Storage layer resiliency is a requirement for any multi-node environment, and it is a requirement to use central node failover with Qlik Sense Enterprise on Windows. See [Storage layer resiliency \(page 129\)](#) to understand the requirement.*

Create a project

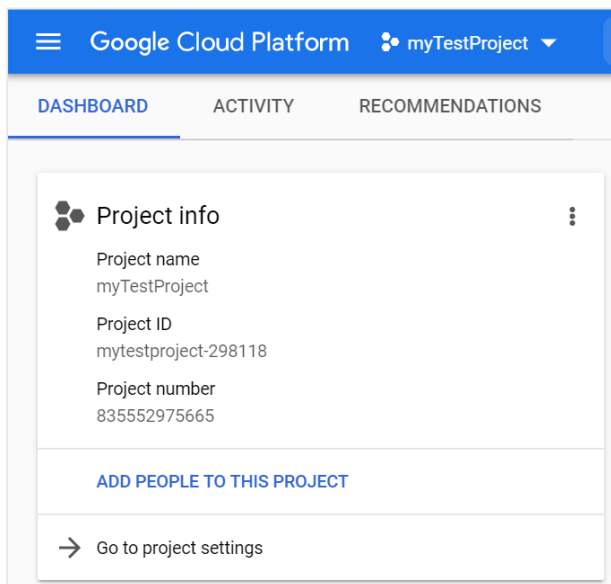
A Google Cloud project is a container for organizing the Google Cloud resources that you create for your Qlik Sense Enterprise on Windows deployment. For more information about Google Cloud projects, see [Google Cloud Projects](#).

1. From the Google Cloud Platform console dashboard, click the scope picker.



2. From the scope picker window, Click **New Project**.
3. Provide a unique name for the project.
4. Select the appropriate billing account.
5. Select the organization this project will belong to.
6. Select the project location. The location refers to the folder.
7. Click **Create**.

When finished, the project is displayed on your dashboard on the **Project info** card, and the scope picker shows the newly created project.



Create a virtual private cloud network

A Google Virtual Private Cloud (VPC) network is an isolated virtual network that is hosted on the Google Cloud network. For more information about Google VPC networks, see [Google VPC network overview](#).

1 Planning your Qlik Sense Enterprise deployment

1. From the Services menu, in the **Networking** section, select **VPC network > VPC networks**.



If you are creating a new project, you are prompted to enable the Compute Engine API to create a VPC.

2. Click **Create VPC Network**.
3. Enter a name and description for the VPC.
4. Under **Subnets**, select **Automatic**.



Automatic creation mode creates a range of IP address for each region. Custom creating mode requires that you create subnets manually.

5. Leave the firewall rules unselected.
6. Select **Regional** Dynamic routing mode.
7. Leave the **Maximum Transmission Unit (MTU)** at 1460.
8. Click **Create**.



It is possible to create the firewall rules at the same time, but in this procedure you create the firewall rules separately.

When finished, the VPC networks page shows your VPC and lists the subnets, along with their IP ranges and region.

Name ↑	Region	Subnets	MTU ⓘ	Mode	IP address ranges	Gateways	Firewall Rules	Global dynamic routing	Flow logs
▶ default		24	1460	Auto ▼			4	Off	
▼ my-test-vpc		24	1460	Auto ▼			0	Off	
	us-central1	my-test-vpc			10.128.0.0/20	10.128.0.1			Off
	europa-west1	my-test-vpc			10.132.0.0/20	10.132.0.1			Off
	us-west1	my-test-vpc			10.138.0.0/20	10.138.0.1			Off
	asia-east1	my-test-vpc			10.140.0.0/20	10.140.0.1			Off
	us-east1	my-test-vpc			10.142.0.0/20	10.142.0.1			Off
	asia-northeast1	my-test-vpc			10.146.0.0/20	10.146.0.1			Off
	asia-southeast1	my-test-vpc			10.148.0.0/20	10.148.0.1			Off
	us-east4	my-test-vpc			10.150.0.0/20	10.150.0.1			Off
	australia-southeast1	my-test-vpc			10.152.0.0/20	10.152.0.1			Off
	europa-west2	my-test-vpc			10.154.0.0/20	10.154.0.1			Off
	europa-west3	my-test-vpc			10.156.0.0/20	10.156.0.1			Off
	southamerica-east1	my-test-vpc			10.158.0.0/20	10.158.0.1			Off
	asia-south1	my-test-vpc			10.160.0.0/20	10.160.0.1			Off
	northamerica-northeast1	my-test-vpc			10.162.0.0/20	10.162.0.1			Off
	europa-west4	my-test-vpc			10.164.0.0/20	10.164.0.1			Off
	europa-north1	my-test-vpc			10.166.0.0/20	10.166.0.1			Off
	us-west2	my-test-vpc			10.168.0.0/20	10.168.0.1			Off
	asia-east2	my-test-vpc			10.170.0.0/20	10.170.0.1			Off
	europa-west6	my-test-vpc			10.172.0.0/20	10.172.0.1			Off
	asia-northeast2	my-test-vpc			10.174.0.0/20	10.174.0.1			Off
	asia-northeast3	my-test-vpc			10.178.0.0/20	10.178.0.1			Off
	us-west3	my-test-vpc			10.180.0.0/20	10.180.0.1			Off
	us-west4	my-test-vpc			10.182.0.0/20	10.182.0.1			Off
	asia-southeast2	my-test-vpc			10.184.0.0/20	10.184.0.1			Off

1 Planning your Qlik Sense Enterprise deployment

Create VPC firewall rules

Google Cloud VPC firewall rules control inbound and outbound connections to and from your Compute Engine instance. For more information on VPC firewall rules, see [VPC firewall rules overview](#).

1. Click the VPC network you created in the steps above to open the VPC network details page.
2. Above the list of subnets, select **Firewall rules**.
3. Click **Add firewall rule** for each new firewall rule.
4. Create the following firewall rules:

Type	Targets	Filters	Protocols / ports	Action	Priority	Logs
Ingress	Apply to all	IP ranges: 10.128.0.0/9	icmp	Allow	1000	off
Ingress	Apply to all	IP ranges: 10.128.0.0/9	all	Allow	1000	off
Ingress	Apply to all	IP ranges: 0.0.0.0/0	tcp:3389	Allow	1000	off
Ingress	Apply to all	IP ranges: 0.0.0.0/0	tcp:80,443	Allow	1000	off

5. Click **Create**.

A summary of the firewall rules is displayed.

The screenshot shows the 'Firewall rules' tab in the Google Cloud console. It features a table with columns for Name, Type, Targets, Filters, Protocols / ports, Action, Priority, Logs, Hit count, and Last hit. Four test rules are listed:

Name	Type	Targets	Filters	Protocols / ports	Action	Priority	Logs	Hit count	Last hit
my-test-firewall-allow-icmp	Ingress	Apply to all	IP ranges: 10.128.0.0/9	icmp	Allow	1000	Off	—	—
my-test-firewall-allow-internal	Ingress	Apply to all	IP ranges: 10.128.0.0/9	all	Allow	1000	Off	—	—
my-test-firewall-allow-rdp	Ingress	Apply to all	IP ranges: 0.0.0.0/0	tcp:3389	Allow	1000	Off	—	—
my-test-firewall-allow-webaccess	Ingress	Apply to all	IP ranges: 0.0.0.0/0	tcp:80,443	Allow	1000	Off	—	—



These rules are only for testing and for the initial setup of your Compute Engine instance. You might need to change your rules for a production environment. For example, the my-test-firewall-allow-rdp rule opens port 3389 to the internet. In a production environment, you might want to use a VPN or a private connection instead.

1 Planning your Qlik Sense Enterprise deployment

Create a storage disk

Google Cloud storage disks are attached to Compute Engine instance to act as the boot disk and to provide persistent storage. For more information about all of the Google Cloud storage options, see [Storage options](#).

You can create the disk independently from the Compute Engine instance or at the same time as you create the instance. These instructions separate the two tasks to keep the procedures distinct.

1. From the Services menu, in the **Compute** section, select **Compute Engine > Disks**.
2. Click **Create Disk**.
3. Enter a name for the disk.
4. Provide an optional description.
5. For the remaining details, use the following values:

Field	Value
Type	Standard persistent disk It is not required to select Replicate this disk within region for this example. To learn about Regional persistent disks, see Regional persistent disks .
Region	Select an appropriate region.
Zone	Select a zone within your region.
Snapshot schedule	No schedule
Source type	Image
Source image	windows-2019-dc-v20201208
Size (GB)	50
Encryption	Google-managed key

6. Click **Create**.

Your disk takes a moment to be created. When ready, it is listed on the Disks overview page.

Disks								
CREATE DISK REFRESH DELETE								
Filter table								
<input type="checkbox"/>	<input checked="" type="checkbox"/>	Name ↑	Type	Size	Zone(s)	In use by	Snapshot schedule	Actions
<input type="checkbox"/>	<input checked="" type="checkbox"/>	my-test-disk	Standard persistent disk	500 GB	us-east1-b		None	⋮



The source image and the size of the disk depend on your requirements. The values above are sufficient for a single-node test site. Before creating your disk, understand your requirements.

1 Planning your Qlik Sense Enterprise deployment

Create a Compute Engine instance

A Google Cloud Compute Engine instance is a virtual machine (VM) that is hosted on Google Cloud infrastructure. You can group instances into cluster to take advantage of Google Cloud autoscaling and load balancing. For more information about VMs, see [Virtual machine instances](#).



In this procedure, you only need to create a single instance for a single-node site. In a multi-node site, follow the same instruction to create your rim nodes.

1. From the Services menu, in the **Compute** section, select **Compute Engine > VM instances**.
2. Click **Create**.
3. Enter a name for the Compute Engine instance.
4. Select the same **Region** and **Zone** as you did previously when creating the disk.
5. For **Machine Configuration**, on the **General-purpose** tab, select the **E2** series and **e2-medium (2vCPU, 4 GB memory)** machine type.



These settings are sufficient for a non-production single-node deployment. For specific deployments, choose the machine configuration that meets your requirements.

6. Clear **Confidential VM service**.
7. Clear **Container**.
8. On the Boot disk card, click **Change**.
9. On the **Existing disks** tab, select the disk you created previously.
10. Click **Select**.
11. For **Identity and API access**, select **Compute Engine default service account** as the Service account.
12. For **Access scopes**, select **Allow default access**.
13. For **Firewall**, leave the options unchecked.



HTTP and HTTPS traffic is allowed through the firewall rules you created earlier.

14. Expand the **Management, security, disks, networking, sole tenancy** section.
15. On the **Networking** tab, select your VPC network. Use the default values for the remaining options.
16. Click **Done**.
17. Click **Create**.

When finished, the Compute Engine instance is available from the **VM instances** overview page.

Name	Zone	Recommendation	In use by	Internal IP	External IP	Connect
<input checked="" type="checkbox"/> my-test-vm	northamerica-northeast1-a			10.162.0.2 (nic0)	34.95.25.14	RDP

1 Planning your Qlik Sense Enterprise deployment

Connect to the Compute Engine instance through a Remote Desktop Protocol (RDP)

1. From the Services menu, in the **Compute** section, select **Compute Engine > VM instances**. The Compute Engine instance you created is listed on the overview page.
2. From the **Connect** column, click the down arrow beside **RDP**.
3. Select **Set Windows password**.
4. Optionally change the default username, then click **Set**.
5. Copy the automatically generated password.
6. Open the Windows RDP client.
7. Enter the **External IP** in the RDP.
8. Log into the instance using the username and password you saved from above.



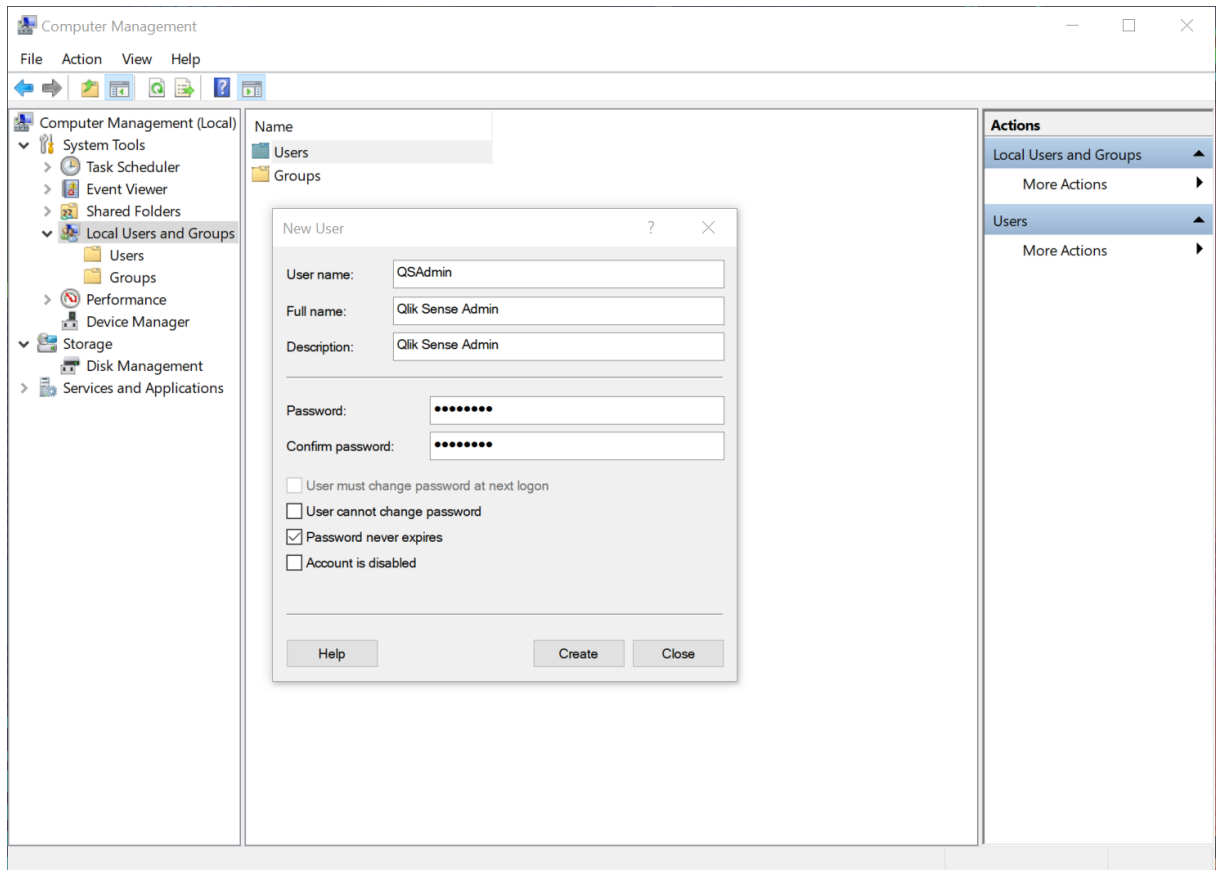
When you connect, you must accept the certificates to continue.

Create Windows user accounts

On your virtual machine, you must set up an administrator service account to install and run Qlik Sense. You can also create additional local user accounts to access Qlik Sense Enterprise on Windows after installation.

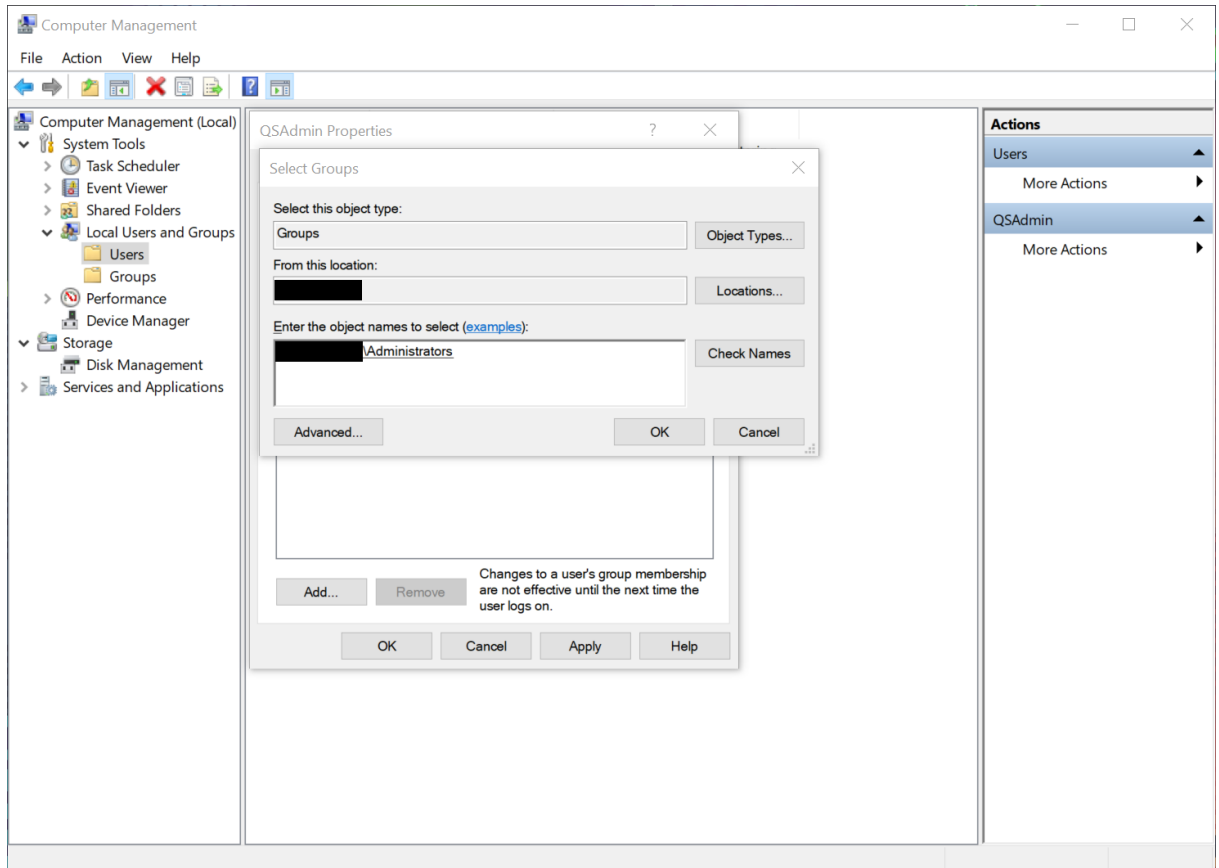
1. From the Windows virtual machine you created, open the Computer Management application.
2. Under **Local Users and Groups**, right-click on **Users**, then select **New User**.
3. Enter a user name and a password. Optionally add a full name and description.
4. Clear **User must change password at next logon**.
5. Select **Password never expires**.

1 Planning your Qlik Sense Enterprise deployment



6. Click **Create**.
7. Right-click the user you created, then select **Properties**.
8. Click **Member Of**.
9. Click **Add**.
10. Type *Administrators*, then click **Check Names**. The field will populate with the Administrators group.

1 Planning your Qlik Sense Enterprise deployment



11. Click **OK**.
12. Under **Member of**, Click **Users**.
13. Click **Remove**.
14. Click **OK**.



To create the local users that can log into Qlik Sense, follow the steps from above for each new user. Do not assign them to the Administrators group.



As the administrator, it is good practice to record and save the user details that you create, like user role descriptions and account settings.

Install Qlik Sense Enterprise on Windows on the Google Compute Engine instance

After the instance is configured, and the user accounts are created, you can install Qlik Sense Enterprise on Windows on your Compute Engine instance.

- *Installing Qlik Sense Enterprise on Windows on a single node (page 98)*
- *Installing Qlik Sense in a multi-node site (page 108)*

1 Planning your Qlik Sense Enterprise deployment



When installing Qlik Sense Enterprise on Windows, you are prompted to enter the server name or address. The correct server name should appear in the field automatically. Do not use a fully qualified domain name.

Google Cloud Platform deployment example

Google Cloud Platform provides a cloud infrastructure with all the services and computing power you need to provide a reliable, cloud deployment platform for Qlik Sense that can perform regardless of unexpected changes in demand and concurrency.

Qlik Sense Enterprise on Windows single-node deployment on Google Cloud Platform

Components in a typical Qlik Sense Enterprise on Windows deployment on Google Cloud Platform:

- VPC - Virtual Private Cloud
A logically isolated virtual network that shares a common security configuration that you define.
- Subnet
You need at least one subnet within the VPC. This could be a public, or private subnet.
- Cloud SQL for PostgreSQL
You can use the PostgreSQL option of the Cloud SQL database service to provide high availability as an alternative to the PostgreSQL repository database embedded in Qlik Sense. PostgreSQL version 11.5 is required.
- Persistent Disk
Block storage that may be attached to instances of Google Compute Engine. Use this to store apps, log files and other common content.
- Compute Engine (GCE)
You deploy a GCE inside the default subnet to host your Qlik Sense installation.
- Cloud VPN
You can make your on-premise Active Directory available in the VPC through an IPsec VPN tunnel to an external IdP.
- Google BigQuery
You can access data sources using the BigQuery data warehouse.
- Qlik Sense Server node
A single Qlik Sense Enterprise on Windows node deployed on Windows Server in the Compute Engine. This includes the Engine, Proxy, Repository and Scheduler services.
- TCP Load Balancer
You can put your resources behind a single anycast IP and scale your resources with intelligent autoscaling if you have deployed several nodes running the proxy service.

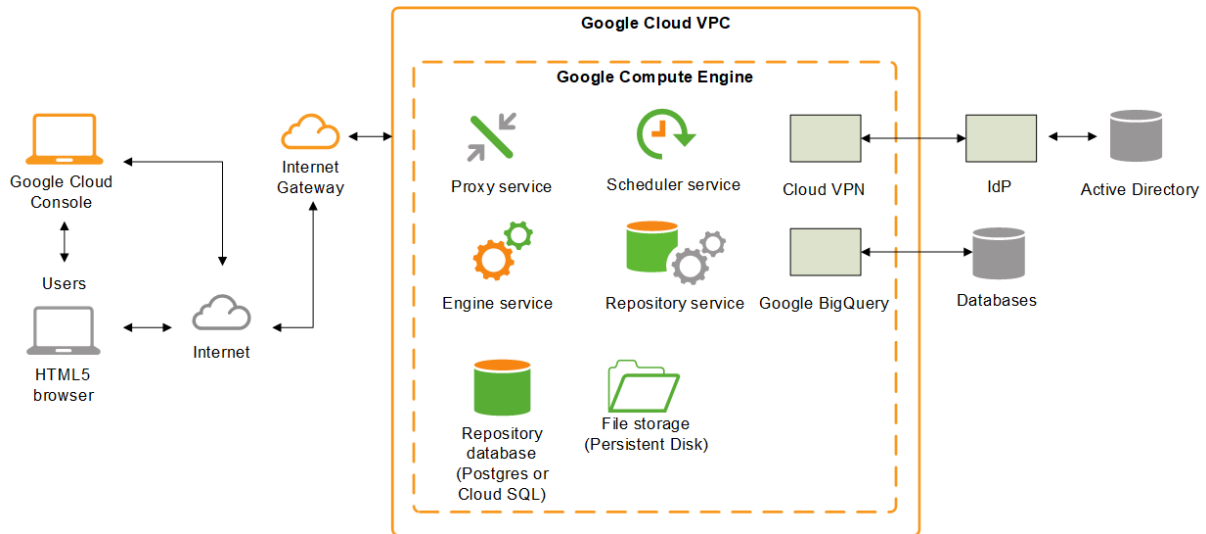
Deployment options:

- Qlik Sense node in a public subnet with direct Internet access.
- Qlik Sense node in a private subnet without Internet access.

The decision whether to choose a public or private subnet in your deployment depends on your overall solution requirements.

1 Planning your Qlik Sense Enterprise deployment

The following example shows a complete Qlik Sense Enterprise on Windows single node deployment on Google Cloud.



Qlik Sense Enterprise on Windows deployed to Oracle Cloud

In an Oracle Cloud deployment, you install Qlik Sense Enterprise on Oracle Cloud Infrastructure (OCI) which is flexible, high performance, and quick to set up.

Deploying Qlik Sense Enterprise on Oracle will enable you to quickly add new applications in a simple and scalable manner. You can do this with a basic knowledge of Oracle security and scalability options but without the need to follow complex on-premise installation and configuration procedures. Using Oracle will enable you to get your Qlik Sense infrastructure up and running in fraction of the time required for an on-premise deployment, and will enable you to scale your deployment quickly and easily, regardless of unexpected changes in demand.

You can only deploy Qlik Sense to Oracle Cloud manually at this time.

Benefits of using OCI

- A quick and effective way of deploying Qlik Sense to the cloud.
- Simple and cost-effective, reducing overall deployment times.
- Quick and easy to deploy Qlik Sense applications.
- Scalable, elastic storage that can be expanded and contracted on demand.
- Oracle security and networking functionality.
- Geographic deployment to multiple regions around the world makes lower latency possible.
- A reliable and high performance platform which is autoscalable.
- Oracle Autonomous DB in the same cloud.

Components

To successfully deploy Qlik Sense on OCI you need a basic understanding of the architecture and services available in an OCI deployment. As part of a Qlik Sense deployment on OCI, you need the following components:

1 Planning your Qlik Sense Enterprise deployment

- Oracle Compute instances
- Oracle Virtual Cloud Network (VCN)

OCI services

You should also have a basic understanding of other OCI services that you can use for managing resources and as data stores for your Qlik Sense applications:

- OCI IAM Policies
- Virtual Cloud Networks (VCN)
- OCI block volumes
- OCI Compute
- Bastion

For more information about OCI services, see  [Oracle Cloud Infrastructure Documentation](#).

Microsoft Windows versions

Your OCI instance needs to be running a Microsoft operating system onto which you can install a Qlik Sense instance. Qlik Sense supports the following Windows operating systems for an OCI deployment:

- Microsoft Windows Server 2016
- Microsoft Windows Server 2019
- Microsoft Windows Server 2022

Qlik Sense Enterprise

Install a single-node Qlik Sense server on your OCI instance.

Qlik Sense Enterprise configuration:

Use the QMC to configure the following:

- Licensing
 - Tokens (only token-based license)
 - User access (token-based license) or Professional access (user-based license)
 - CPU cores
- Security groups

Create a proxy setup for allowing HTTP access.

Other considerations


When you deploy Qlik Sense to OCI for the first time you should also consider the following.

Security

Use the OCI IAM to configure OCI users and IAM policies and the QMC to configure all security groups and authentication settings in Qlik Sense.

For more information about security, see:

1 Planning your Qlik Sense Enterprise deployment

-  [Oracle Cloud Infrastructure Security](#)
- *Qlik Sense Enterprise on Windows security (page 208)*

Connectivity

Qlik Sense applications can use the following Oracle web services as data stores:

- Oracle Autonomous DB
- Oracle Base Database Service (Standard and Enterprise Edition)
- MySQL Database Service
- Essbase
- Oracle Exadata Database Service

In an OCI deployment you can use the following connectivity mechanisms to connect to different data sources:

- ODBC connection
- OLE DB connection
- REST API connection
- Native connector to a specific source

Connectivity scenarios:

- Qlik Sense instance that uses Oracle Database instances specified above.
- Hybrid Qlik Sense instance - uses data stored in Oracle data sources as well as data stored on premise.

Scalability

As environments grow in terms of number of users, number and size of applications, and number of data sources it is important to understand how to size the environment correctly and how to scale the environment accordingly. You need to create a multi-node environment to effectively scale up or down, by creating dedicated servers for different purposes. You can then allocate resources correctly across the following Qlik Sense services.

- Engine Service – The QIX engine, provides in-memory Associative Data Indexing and calculation supporting analysis
- Proxy Service – Manages authentication, handles user sessions and load balancing
- Repository Service – Manages Qlik Sense applications, controls access, and handles configuration
- Scheduling Service – Manages reloads of Qlik Sense applications and other scheduled tasks
- Service Dispatcher – Launch and manage the data profiling service for the data load model

For more information about scalability, see the [Qlik Sense Performance Benchmark](#) technical brief.

Deploying Qlik Sense Enterprise in a multi-cloud environment

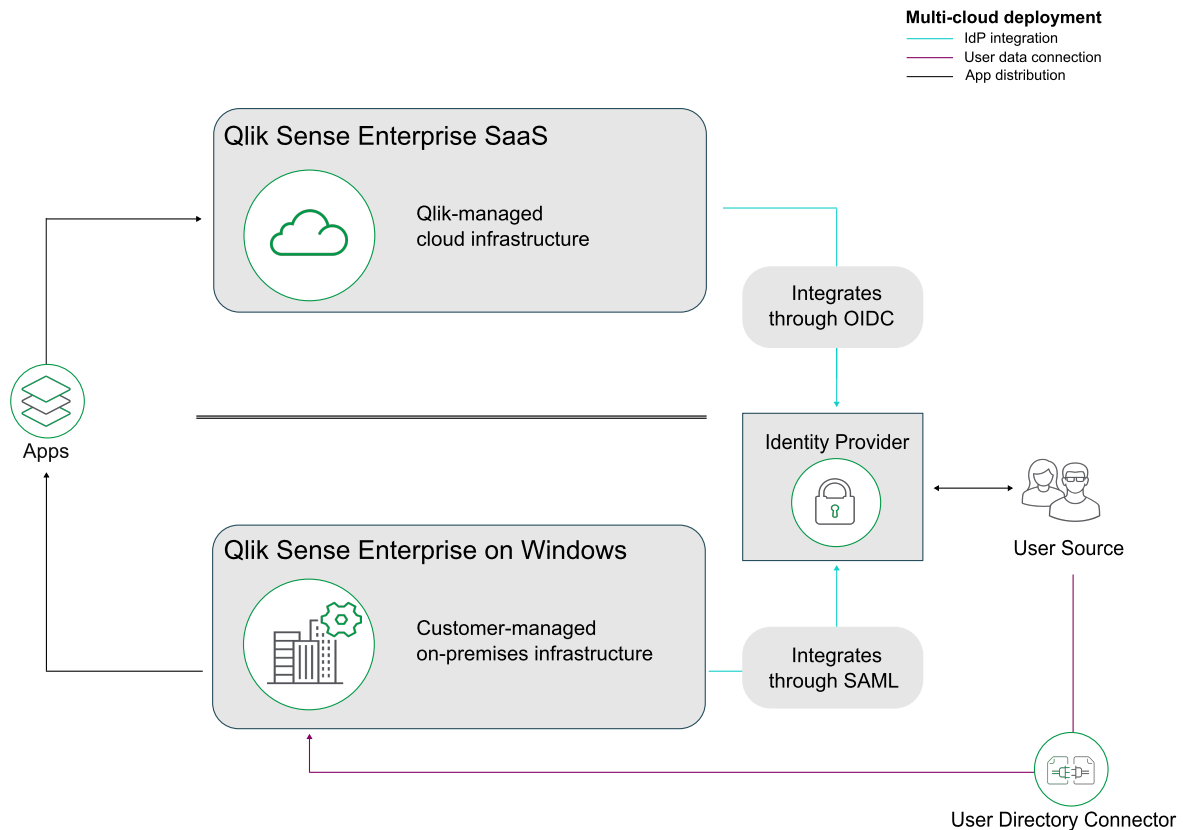
A Qlik Sense Enterprise multi-cloud deployment lets you deploy both Qlik Sense Enterprise on-premises and Qlik Sense Enterprise on cloud so users can develop apps on-premises and access them from the cloud.

The primary components of a multi-cloud deployment are:

1 Planning your Qlik Sense Enterprise deployment

- Qlik Sense Enterprise on Windows, deployed on-premises.
- Qlik Sense Enterprise SaaS, deployed to a Qlik-managed cloud infrastructure.

After you install your Qlik Sense Enterprise on Windows on-premises, you can choose to configure multi-cloud with Qlik Sense Enterprise SaaS. Once your multi-cloud deployment is configured, you can distribute Qlik Sense apps that you create in Qlik Sense Enterprise on Windows to the cloud for consumption. Users are integrated through the identity provider and consume a single license regardless of whether they connect through the Qlik Sense Enterprise on Windows hub or the cloud hub.



The characteristics of a multi-cloud deployment are:

- Qlik Sense Enterprise on Windows installed on-premises.
- Access to Qlik Sense Enterprise SaaS.
- An identity provider that supports OIDC and SAML to integrate user authentication between on-premises and cloud, or a local bearer token.
- A license that includes multi-cloud (enabled).

Who should consider a multi-cloud deployment:

- You have Qlik Sense Enterprise on Windows running on-premises and you want to expand to the cloud.
- You want to develop apps in a Windows environment and allow users who cannot access your Windows environment to consume apps.
- You want to scale out the number of users consuming apps using cloud resources.

1 Planning your Qlik Sense Enterprise deployment

For an example of how to distribute an app from Qlik Sense Enterprise on Windows to Qlik Sense Enterprise SaaS, see *Distributing apps from Qlik Sense Enterprise on Windows to Qlik Sense Enterprise SaaS* (page 96).

Distributing apps from Qlik Sense Enterprise on Windows to Qlik Sense Enterprise SaaS

When you publish apps to streams in Qlik Sense Enterprise on Windows, you may also want to distribute the apps to your Qlik Sense Enterprise SaaS deployment. By making configurations in Qlik Sense Enterprise on Windows and in your cloud tenant, you can automate the distribution to cloud so that apps are automatically distributed to your Qlik Sense Enterprise SaaS deployment when you publish them to a stream.

Prerequisites

To distribute an app from Client-Managed Qlik Sense to qlik you need the following:

- Qlik Cloud tenant.
- A license that includes multi-cloud. Either of the following
 - The same signed license key for Qlik Sense Enterprise on Windows and Qlik Cloud.
 - Different signed license keys, where the Qlik Sense Enterprise on Windows license has the cloud attributes enabled.

Configurations

The following is a high-level description of the configuration steps.

1. Set up a deployment in the Qlik Management Console (QMC) of your Qlik Sense Enterprise on Windows server.
2. Create a multi-cloud identity provider configuration in the tenant.
3. Create the distribution policy in the QMC.

Setting up a deployment

Do the following:

1. From the QMC start page, open **Cloud distribution**.
2. Click **Deployment setup**.
3. Click **Set up new** in the bottom-left corner.
4. Enter a deployment name.
You will use this name in the distribution policy.
5. Enter the **API endpoint**, that is, your tenant address.
Example: *https://my-tenant.eu.qlikcloud.com*.
6. For **Audience**, enter *qlik.api*.
7. Select **Use local bearer token**.



Using a local bearer token simplifies setup. If you do not use it, you need to enter **Client ID**, **Client secret**, and **Token endpoint** instead.

8. Click **Apply**.

1 Planning your Qlik Sense Enterprise deployment

9. Click **Copy to clipboard**.

You need the local bearer token in the identity provider configuration.



Before you can test the connection you need to create the identity provider configuration in the next step.

Creating the identity provider configuration

Do the following:

1. Open the Management Console in your Qlik Sense Enterprise SaaS tenant and select **Identity provider** in the menu to the left.
2. Click **Create new**.
The **Create identity provider configuration** window is opened.
3. Under **Type**, select *Multi-cloud*.
4. Optionally, enter a description.
5. In the **Local bearer token** box, paste the token you copied in the deployment setup.

Creating the distribution policy

Distribution policies are used to determine whether a published app can be distributed to one or more of the deployments in Qlik Cloud. Only published apps can be distributed.

Do the following:

1. In the QMC of Qlik Sense Enterprise on Windows, open **Cloud distribution** and select **Distribution policies**.
2. Click **Create new**.
3. In the **Create rule from template** list, select *Distribution_App*.
4. Name the distribution policy.
5. Under **Basic**, verify that the resource filter value is *App_** and the action **Distribute** is selected.
6. In the rule editor keep the values *subject*, *name*, *=*, and *value*. In the last field, you only need to add the name of the deployment you created earlier (in *Setting up a deployment (page 96)*). Let's assume it is *deployment*.
The **Conditions** box in the **Advanced** section should then have the following string:
`((subject.name="deployment"))`.
7. Click **Validate rule**.
The rule syntax is checked, and, if valid, a confirmation is displayed.
8. Click **Apply** to save the rule.

This is a simple example of app distribution where all apps that are published to a stream also are distributed to your Qlik Sense Enterprise SaaS tenant.

2 Installing Qlik Sense Enterprise on Windows

When you install Qlik Sense you have several deployment options depending on the size and requirements of your organization. Before you begin the installation process, choose the appropriate architecture for your needs. Consider scalability and performance and factors such as how many apps you want to run, how many concurrent users you need, or how many reloads you want per hour.

Deployment recommendations

Size of organization	Qlik Sense deployment
Small	Single-node
Medium	Single-node or multi-node
Large	Multi-node

2.1 Installing Qlik Sense Enterprise on Windows on a single node

The simplest and most basic installation of Qlik Sense Enterprise on Windows is to install all Qlik services on a single server, or node, to create a single-node site. This kind of deployment is best suited for smaller organizations, where users are in a single time zone, and where app reloads can be done overnight when there are no users accessing the site.

Pre-installation

Do the following:

1. Check the [system requirements for Qlik Sense Enterprise](#).
2. Check the [supported browsers](#).
3. Check the [ports](#) page to see which ports are used in your deployment.
4. Check the [Qlik Sense licenses](#) to see your licensing options.


Preparing the server

Do the following:

1. Log in to the server where you will install Qlik Sense Enterprise on Windows.



If the server is running anti-virus software, learn how [anti-virus software scanning can affect performance](#) and how to solve it.

2. Create the required [user accounts](#) on the server.
3. Download the Qlik Sense Enterprise on Windows installer from  [Product Downloads](#). For more information, see *Downloading installation files (page 17)*.

2 Installing Qlik Sense Enterprise on Windows

4. Create a [file share](#). The file share stores Qlik application data and must be accessible to the central node, as well as all rim nodes in a multi-node site.
5. Install Microsoft .NET Framework version 4.8 on the server. If not installed, the Qlik Sense installer will prompt you for it.



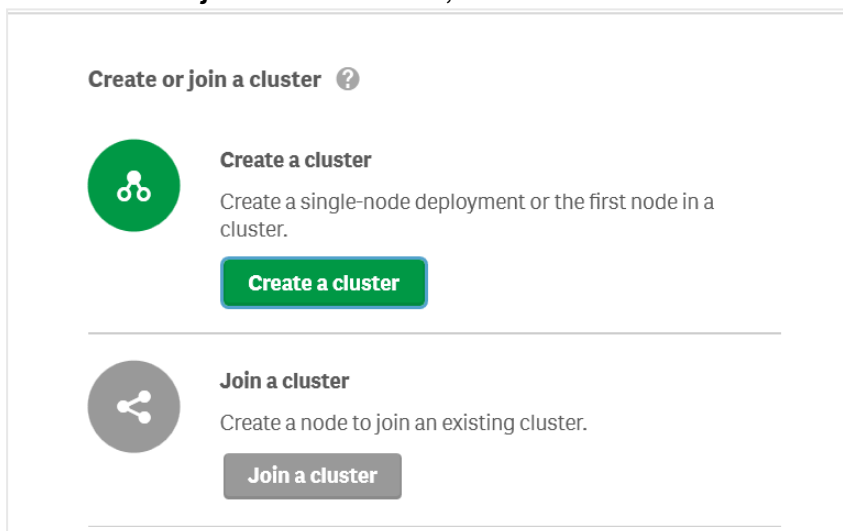
For both single node and multi-node deployments, you can connect to an existing Qlik Sense repository database. If you have an existing repository database on your server but you do not plan to use it, remove it before you start the installation.

Installing Qlik Sense Enterprise on Windows

In a single-node site, all Qlik services are installed on a single Windows server.

Do the following:

1. Right-click the installer file and select **Run as administrator**. Click **Install** to start the installation.
2. On the **License Agreement** window, read and accept the Qlik User License Agreement.
3. On the **Create or join a cluster** window, select **Create cluster**.



For both single-node and multi-node sites, you begin by creating the central node in a cluster. A single-node site contains only the central node, whereas a multi-node site has additional rim nodes connected to the central node.

4. On the **Select the host name** window, enter the address to your computer or select the pre-defined value from the list.

Accepted address values

IP address

Server or machine name

Fully qualified machine name

Example

10.1.123.234


rd-bdm-win2019

rd-bdm-win2019.rdlund.qliktech.com


2 Installing Qlik Sense Enterprise on Windows

Select the host name

The central node uses certificates to securely communicate with other nodes in the deployment. Enter the host name for this computer (node). Use the same host name format that other computers will use to connect to this node.

Computer host name 

Select a host name from the list below (optional):

-  usott-bdm2.lan



Ensure that the recommended server node name displayed in the **Computer host name** field matches the one you will use to access this node, otherwise, enter an appropriate address or fully qualified machine name. You can verify your computer host name by navigating to `Control Panel\All Control Panel Items\System`.



For additional support and to learn about common issues related to the Qlik Sense Enterprise on Windows installation, see [Common issues and solutions to problems with your installation](#) (page 107).

5. On the **Set up the repository database** window, you can create a new repository database on your current computer or connect to an existing repository database. Learn more about [persistence, file shares, and the repository databases](#), and [database security](#) for shared persistence.



Do not use postgres as the database username.

- To create a new repository database on the current computer, click **Create a new database**. On the **Configure a new database (1/2)** window, set a database user password and adjust the advanced settings if required. On the **Configure a new database (2/2)** window, set a database superuser password.

2 Installing Qlik Sense Enterprise on Windows

Configure a new database (1/2)

Set a password for the repository database user.

Database user password

Confirm password

▼ Advanced settings ?

Listen addresses ?

IP ranges ?

Max connections ?

Database port ?

Database user ?

- To connect to an existing repository database, click **Connect to an existing PostgreSQL database**.

Enter the database hostname and database user password. The port and database user are filled in with default values. Change these only if you changed them during the initial database configuration.

Enter database credentials

To connect to an existing repository database, enter the PostgreSQL database host name and provide the database user password.

Database host name ?

Database port ?

Database user ?

Database user password



For additional support and to learn about common issues related to the Qlik Sense Enterprise on Windows installation, see [Common issues and solutions to problems with your installation](#) (page 107).

6. On the **Provide information for the Qlik services (1/2)** window, enter the username and password to run the services on the current computer.

2 Installing Qlik Sense Enterprise on Windows

Under **Additional server settings** you can set the database maximum connection pool size and listening ports for http and https. https is the default option, to use http you need to select **Enable http**.

Database maximum connection pool size: This value is calculated by the installer, based on the server's configuration. The default value is usually the best option.



The service credentials username must be in the form domain\username.



If you enter a username that is more than 20 characters long, it must be in User Principal Name (UPN) format, and must include the full domain name. For example, longusername@full.domain.name.

Provide information for the Qlik services (1/2)

Settings and account information needed for Qlik Sense services installed on the host computer to work properly.

Windows service account credentials ?

Username

Password

Additional server settings ?

Database maximum connection pool size ?

Https port number (https://rd-fpc-dd24.rdlund.qliktech.com:443)

Http port number (http://rd-fpc-dd24.rdlund.qliktech.com:80)

 Enable http

For additional support and to learn about common issues related to the Qlik Sense Enterprise on Windows installation, see [Common issues and solutions to problems with your installation \(page 107\)](#).

7. On the **Provide information for the Qlik services (2/2)** window, enter the path or URL to the file share you created earlier, for example, \\<domain>\QlikShare. The file share can be a local folder, or it can be hosted on another computer. Learn more about [persistence, file shares, and the repository databases](#).

Provide information for the Qlik services (2/2)



Enter the path to the file share. The file share holds data and resources used by the Qlik services. It must be accessible by the Qlik Sense service user on all nodes in the cluster.

Root folder


\\USOTT-BDM2\share 

▼ Advanced settings

Apps folder

\\USOTT-BDM2\share\Apps  

Archived logs folder

\\USOTT-BDM2\share\ArchivedLogs  

Static content folder


\\USOTT-BDM2\share\StaticContent  




For additional support and to learn about common issues related to the Qlik Sense Enterprise on Windows installation, see [Common issues and solutions to problems with your installation](#) (page 107).


- On the **Installation location** window, enter the location to install Qlik Sense Enterprise on Windows, or choose the default location on the C:\ drive.
- On the **Ready to install** window, select the installer options.

Ready to install

Create desktop shortcuts 

Start the Qlik Sense services when the setup is complete 

Include object bundles

Install supported object bundles 

Dashboard bundle

Visualization bundle

Help us improve

Qlik collects systems and usage data to optimize, support, and improve our products and services. This data is anonymized. If you prefer to not share this data, clear this box. [Learn more about our privacy policy here.](#)

- In the **Ready to install** section, clear the **Start the Qlik Sense services when the setup is complete** check box if you want to use a dedicated service account to run the Qlik Sense

2 Installing Qlik Sense Enterprise on Windows

services.

- In the **Include object bundles** section, optionally install the object bundles. Then, select which object bundles you want to install from the list of those available for your Qlik Sense Enterprise on Windows installation.



If you are installing object bundles, read and accept the object bundles license agreement.

- In the **Help us improve** section, select if you want to anonymously share system data with Qlik.
- Click **Next**.



To add or remove object bundles after an installation, see [Modifying an object bundles installation](#) (page 106)



For additional support and to learn about common issues related to the Qlik Sense Enterprise on Windows installation, see [Common issues and solutions to problems with your installation](#) (page 107).

10. On the **License Agreement** window, accept the license agreement.

License Agreement

Qlik Sense Object Bundle Terms and Conditions

- Licenses. The Qlik Sense Object Bundles consist of open source software or Qlik provided objects ("Objects") that are separately licensed in accordance with the license terms set forth herein. Any other license or customer agreement for other Qlik Products expressly do not apply to the Objects.
- Visualization Bundle Objects and Licenses.
 - Bar & Area Chart**
 - Copyright © 2019 QlikTech International AB
 - barsPlus, Copyright © 2016 Larry Woodside, (<https://github.com/LarryWoodside/barsPlus>)
 - License: [MIT License](#)
 - Bullet Chart**
 - Copyright © 2019 QlikTech International AB
 - D3-Bullet-Chart, Copyright © 2015 Kevin McGovern, (<https://github.com/mcgovey/D3-Bullet-Chart>)
 - License: [MIT License](#)
 - Funnel Chart**
 - Copyright © 2019 QlikTech International AB
 - License: [Qlik License](#)
 - Grid Chart**

I accept the license agreement

- Click **Install**.

2 Installing Qlik Sense Enterprise on Windows

11. You will see a message indicating that Qlik Sense Enterprise on Windows has been installed successfully.

Licensing Qlik Sense

Before you can start using Qlik Sense you must activate your site license.

Do the following:

1. Open the Qlik Management Console (QMC) by entering the QMC address in your browser.
By default, the QMC address is `https://<QPS server name>/qmc`.
The QMC displays the **Site license properties** screen the first time you open it.
2. In the **Signed license key** field, paste the unique license key that was sent to you via email upon completion of your purchase.
3. Click **Apply**.
Successfully licensed is displayed.

You have activated your Qlik Sense site license.



For more information, see [How to Apply a Signed License Key \(SLK\) to Qlik Sense Enterprise on Windows](#).

You are ready to connect to a user directory (optional), allocate user access or professional access, and set up permissions.

Allocating access to users

Your license is either based on access types, with professional access allocation as an option, or on tokens, with user access allocation as an option.

Access types license

Your Qlik Sense license includes a number of professional access allocations that are used to grant users in your organization access to Qlik Sense.

Do the following:

1. In the QMC, from the **Start** menu, click **License management**.
The **License usage summary** screen is displayed.
2. Click the **Professional access allocations** tab.
3. Click the **+ Allocate** button.
The **Users** screen is displayed.
4. Select the users that you want to provide access to from the list and click **Allocate**.



Allocate is disabled if the number of allocations available is insufficient for the number of selected users.

2 Installing Qlik Sense Enterprise on Windows

The users that you allocated access to appear in the **Professional access allocations** overview table.

Token-based license

Your Qlik Sense license includes a number of tokens that are used to allocate Qlik Sense access to users in your organization.

Do the following:

1. In the QMC, from the **Start** menu, click **License management**.
The **License usage summary** screen is displayed.
2. Click the **User access allocations** tab.
3. Click the **+ Allocate** button.
The **Users** screen is displayed.
4. Select the users that you want to provide access to from the list and click **Allocate**.



Allocate is disabled if the number of tokens available for allocation is insufficient for the number of selected users.

The users that you allocated access to appear in the **User access allocations** overview table.

Additional configuration

After you have installed and verified that Qlik Sense is running correctly, you may find the following configuration information useful:

- Load balancing - create load balancing rules in the QMC to improve resilience and performance in a multi-node site.
- Host allow list - configure the virtual proxy advanced settings to add your own hosts names to the allowlist.
- User imports (UDC) - configure the user directory connector to retrieve users from a user directory.

Qlik Sense is ready to be used. But before getting started, back up your files so that you can recover from a system crash, see *Backup and restore Qlik Sense Enterprise on Windows (page 181)*

Modifying an object bundles installation

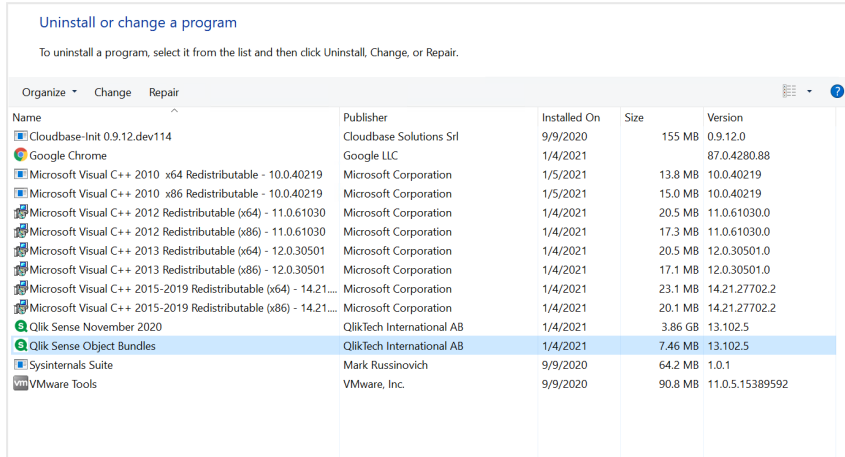
You can add or remove object bundles from your Qlik Sense deployment at any moment. If you have a multi-node installation, object bundles are installed on the central node.



You can see which extensions are installed in your deployment by checking the **Extensions** section in the Qlik Management Console (QMC).

Do the following:

1. In **Windows Control Panel**, open **Programs and Features**.
2. In the list of programs, double-click the object bundle that you want to modify.



3. The Object Bundle Setup Wizard opens. Click **Next**.
4. Select **Change**.
5. On the **Custom setup** screen, click on the bundle icon to select how to modify the bundle installation:
 - If the bundle is installed, select **Entire feature will be unavailable** to uninstall it.
 - If the bundle is not installed, select **Entire feature will be installed on local hard drive** to install it.Then, click **Next**.
6. Click **Change**.
Once the modification is complete, you are required to manually restart the Qlik Sense Repository Service.
7. Click **Finish** to close the Object Bundle Setup Wizard.
8. Manually restart the Qlik Sense Repository Service to apply the changes.

You can verify that the changes have been correctly applied by checking the **Extensions** section in the QMC.

Common issues and solutions to problems with your installation

Windows user accounts and service credentials

- [Changing the Qlik Sense Service Account and what to consider](#)
- [Interactive Logon Rights for Qlik Sense Installations](#)
- [How to: Change the Qlik Sense Proxy certificate if the service account does not have local administrative permissions](#)
- [Qlik Sense repository service fails to start due to error " 'SeSecurityPrivilege " error](#)

Host name

- [🔗 Qlik Sense: Change hostname \(and certificates\) after an installation](#)
- [🔗 Authentication fails repeatedly when using external DNS alias locally on the server](#)
- [🔗 How to change the certificate used by Qlik Sense Hub and QMC](#)

Shared persistence

- [🔗 Qlik Sense Share Storage options and requirements](#)
- [🔗 How to change the share path in Qlik Sense \(Service Cluster\)](#)
- [🔗 Qlik Sense Logging Service does not have support for SSL Database traffic encryption](#)
- [🔗 Configure Qlik Sense to use a dedicated PostgreSQL server](#)

Qlik Sense extension bundles

- [🔗 Installing and removing Qlik Sense Extension bundle](#)

Anti-virus software

- [🔗 Antivirus exceptions for Qlik Sense- McAfee, Symantec and Other Anti-Virus exclusions absolutely required](#)

2.2 Installing Qlik Sense in a multi-node site

A Qlik Sense multi-node deployment offers more configuration options than single node deployments. In a multi-node site, you can distribute Qlik Sense services across one or more server nodes to optimize scalability and performance.

Consider the pre-installation items

Pre-installation (page 109)

Prepare the Windows server

Preparing the server (page 109)

Install Qlik Sense on the central node

Installing the Qlik Sense central node (page 110)

Configure PostgreSQL multi-node connections on the central node

Configuring PostgreSQL multi-node connections (page 117)

License Qlik Sense on the central node

Licensing Qlik Sense (page 117)

Allocate access to users

Allocating access to users (page 118)

Install Qlik Sense on the rim nodes

Installing a Qlik Sense rim node (page 119)

Configure the rim nodes in QMC

Connecting and configuring the nodes (page 122)

Optionally, assign a node to be a failover candidate

Configuring failover for central node resiliency (page 129)

Optionally, perform additional configuration

Additional configuration (page 124)

Pre-installation

Follow this work-flow when installing Qlik Sense in a multi-node site:

Preparing a large, enterprise multi-node deployment requires careful planning, so first ensure that you have considered all the architecture and configuration options available.

1. Check the [system requirements for Qlik Sense Enterprise](#).
2. Check the [supported browsers](#).
3. Check the [ports](#) page to see which ports are used in your deployment.
4. Check the [Qlik Sense licenses](#) to see your licensing options.

For more information on multi-node architecture and configuration options see:


- *Planning your Qlik Sense Enterprise deployment (page 9)*
- *Before you install Qlik Sense Enterprise on Windows (page 18)*

Preparing the server

1. Log in to the server where you will install Qlik Sense Enterprise on Windows.



If the server is running anti-virus software, learn how [anti-virus software scanning can affect performance](#) and how to solve it.

2. Create the required [user accounts](#) on the server.
3. Download the Qlik Sense Enterprise on Windows installer from  [Product Downloads](#).
For more information, see *Downloading installation files (page 17)*.
4. Create a [file share](#). The file share stores Qlik application data and must be accessible to the central node, as well as all rim nodes in a multi-node site.
5. Install Microsoft .NET Framework version 4.8 on the server. If not installed, the Qlik Sense installer will prompt you for it.



For both single node and multi-node deployments, you can connect to an existing Qlik Sense repository database. If you have an existing repository database on your server but you do not plan to use it, remove it before you start the installation.

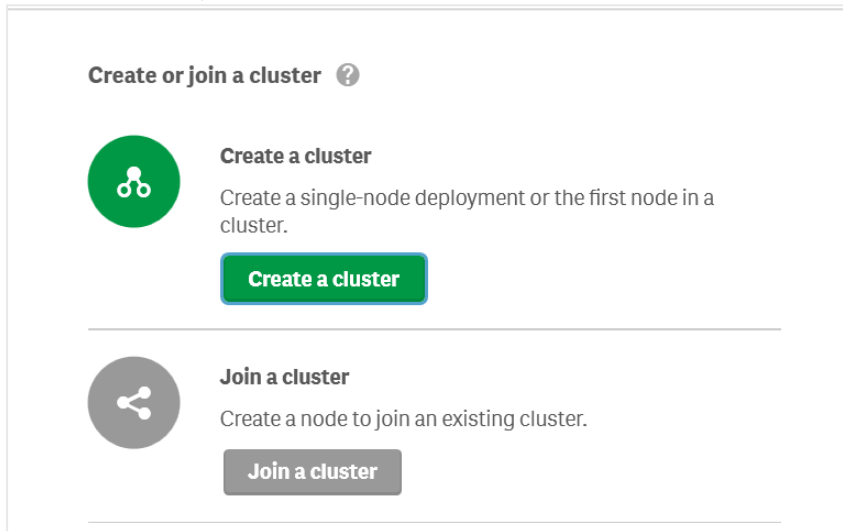
Installing the Qlik Sense central node



When installing a central node, you may also wish to [configure a failover candidate](#). You only have the option to create a failover candidate when you are creating a node.

Do the following:

1. Right-click the installer file and select **Run as administrator**. Click **Install** to start the installation.
2. On the **License Agreement** window, read and accept the Qlik User License Agreement.
3. On the **Create or join a cluster** window, select **Create cluster**.



For both single-node and multi-node sites, you begin by creating the central node in a cluster. A single-node site contains only the central node, whereas a multi-node site has additional rim nodes connected to the central node.


4. On the **Select the host name** window, enter the address to your computer or select the pre-defined value from the list.

Accepted address values	Example
IP address	10.1.123.234
Server or machine name	rd-bdm-win2019
Fully qualified machine name	rd-bdm-win2019.rdlund.qliktech.com


2 Installing Qlik Sense Enterprise on Windows

Select the host name

The central node uses certificates to securely communicate with other nodes in the deployment. Enter the host name for this computer (node). Use the same host name format that other computers will use to connect to this node.

Computer host name 

Select a host name from the list below (optional):

-  usott-bdm2.lan



Ensure that the recommended server node name displayed in the **Computer host name** field matches the one you will use to access this node, otherwise, enter an appropriate address or fully qualified machine name. You can verify your computer host name by navigating to `Control Panel\All Control Panel Items\System`.



For additional support and to learn about common issues related to the Qlik Sense Enterprise on Windows installation, see [Common issues and solutions to problems with your installation](#) (page 125).

5. On the **Set up the repository database** window, you can create a new repository database on your current computer or connect to an existing repository database. Learn more about [persistence, file shares, and the repository databases](#), and [database security](#) for shared persistence.



Do not use postgres as the database username.

- To create a new repository database on the current computer, click **Create a new database**. On the **Configure a new database (1/2)** window, set a database user password and adjust the advanced settings if required. On the **Configure a new database (2/2)** window, set a database superuser password.

2 Installing Qlik Sense Enterprise on Windows

Configure a new database (1/2)

Set a password for the repository database user.

Database user password

Confirm password

▼ Advanced settings

Listen addresses

IP ranges

Max connections

Database port

Database user

- To connect to an existing repository database, click **Connect to an existing PostgreSQL database**.

Enter the database hostname and database user password. The port and database user are filled in with default values. Change these only if you changed them during the initial database configuration.

Enter database credentials

To connect to an existing repository database, enter the PostgreSQL database host name and provide the database user password.

Database host name

Database port

Database user

Database user password

Make a note of these values as you will need them again when you install a rim node.



All Qlik Sense servers must be in the same geographic location or data center as the repository database and the file share.

2 Installing Qlik Sense Enterprise on Windows



For additional support and to learn about common issues related to the Qlik Sense Enterprise on Windows installation, see *Common issues and solutions to problems with your installation* (page 125).

6. On the **Database configuration** window, under **Advanced settings**, configure the listen addresses, IP ranges, and max connections from other nodes, then click **Next**.

This is an optional step if you install a local repository database. You can also configure the database service listener directly in your PostgreSQL repository database. See: *Installing and configuring PostgreSQL* (page 132)

Enter the following values:

Database configuration values

Field name	Description	Example Value
Listen addresses	The IP address(es) to listen on. Use the value * to allow access for all IP addresses. If entering multiple listen addresses use a comma separated list.	*
IP ranges	To allow all servers to access the repository database, use the value 0.0.0.0/0 (for all IPv4 addresses) and ::/0 (for all IPv6 addresses). If entering multiple IP addresses use a comma separated list.	0.0.0.0/0, ::/0
Max connections	Specifies the maximum number of concurrent connections to the database. The default value for a single server is 100. In a multi-node environment, this should be adjusted to the sum of all repository connection pools + 20. By default, this value is 110 per node.	110



This screen does not appear if you are using a remote PostgreSQL database.

7. On the **Provide information for the Qlik services (1/2)** window, enter the username and password to run the services on the current computer.

Under **Additional server settings** you can set the database maximum connection pool size and listening ports for http and https. https is the default option, to use http you need to select **Enable http**.

Database maximum connection pool size: This value is calculated by the installer, based on the server's configuration. The default value is usually the best option.



The service credentials username must be in the form domain\username.

2 Installing Qlik Sense Enterprise on Windows



If you enter a username that is more than 20 characters long, it must be in User Principal Name (UPN) format, and must include the full domain name. For example, `longusername@full.domain.name`.

Provide information for the Qlik services (1/2)

Settings and account information needed for Qlik Sense services installed on the host computer to work properly.

Windows service account credentials

Username

Password

Additional server settings

Database maximum connection pool size

Https port number (https://rd-fpc-dd24.rdlund.qliktech.com:443)

Http port number (http://rd-fpc-dd24.rdlund.qliktech.com:80)

 Enable http

For additional support and to learn about common issues related to the Qlik Sense Enterprise on Windows installation, see [Common issues and solutions to problems with your installation](#) (page 125).

8. On the **Provide information for the Qlik services (2/2)** window, enter the path or URL to the file share you created earlier, for example, `\\<domain>\QlikShare`. The file share can be a local folder, or it can be hosted on another computer. Learn more about [persistence, file shares, and the repository databases](#).

2 Installing Qlik Sense Enterprise on Windows

Provide information for the Qlik services (2/2)



Enter the path to the file share. The file share holds data and resources used by the Qlik services. It must be accessible by the Qlik Sense service user on all nodes in the cluster.

Root folder



\\USOTT-BDM2\share 

▼ Advanced settings

Apps folder

\\USOTT-BDM2\share\Apps  

Archived logs folder

\\USOTT-BDM2\share\ArchivedLogs  

Static content folder

\\USOTT-BDM2\share\StaticContent  



For additional support and to learn about common issues related to the Qlik Sense Enterprise on Windows installation, see [Common issues and solutions to problems with your installation](#) (page 125).

9. On the **Installation location** window, enter the location to install Qlik Sense Enterprise on Windows, or choose the default location on the C:\ drive.
10. On the **Repository Database Superuser Password** window, enter a password for the PostgreSQL repository database superuser.





This screen does not appear if you are using a remote PostgreSQL database, or if you are installing a rim node (Join cluster option).

11. On the **Ready to install** window, select the installer options.


2 Installing Qlik Sense Enterprise on Windows

Ready to install

Create desktop shortcuts 

Start the Qlik Sense services when the setup is complete 

Include object bundles

Install supported object bundles 

Dashboard bundle

Visualization bundle

Help us improve

Qlik collects systems and usage data to optimize, support, and improve our products and services. This data is anonymized. If you prefer to not share this data, clear this box. Learn more about our privacy policy [here](#).

- In the **Ready to install** section, clear the **Start the Qlik Sense services when the setup is complete** check box if you want to use a dedicated service account to run the Qlik Sense services.
- In the **Include object bundles** section, optionally install the object bundles. Then, select which object bundles you want to install from the list of those available for your Qlik Sense Enterprise on Windows installation.



If you are installing object bundles, read and accept the object bundles license agreement.

- In the **Help us improve** section, select if you want to anonymously share system data with Qlik.
- Click **Next**.



To add or remove object bundles after an installation, see [Modifying an object bundles installation](#) (page 124)



For additional support and to learn about common issues related to the Qlik Sense Enterprise on Windows installation, see [Common issues and solutions to problems with your installation](#) (page 125).

12. You will see a message indicating that Qlik Sense Enterprise on Windows has been installed successfully.



If you selected local system as the user account type in the **Service Credentials** screen, but wish to use a dedicated service account to run the Qlik Sense services, change the user account type and manually start the Qlik Sense services now. See *User accounts* (page 63)

Configuring PostgreSQL multi-node connections

For multi-node sites, you must set the connection pool limit. This limit is determined by the `max_connections` setting in the `postgresql.conf` file. The settings value depends on the number of nodes in your site.



If you reach the connection pool limit, PostgreSQL rejects additional connections.

Do the following:

1. Stop the Qlik Sense services.
2. Navigate to the `postgresql.conf` file in `C:\ProgramData\Qlik\Sense\Repository\PostgreSQL<version>` of your Qlik Sense installation.
3. Open the file in a text editor as an administrator.
4. Make the following configuration changes:

PostgreSQL configuration changes

Setting	Description	Example Value
<code>max_connections</code>	Specifies the maximum number of concurrent connections to the database. The default value for a single server is 100. In a multi-node environment, this should be adjusted to the sum of all repository connection pools + 20. By default, this value is 110 per node.	110

5. Save your changes.
6. Restart the Qlik Sense services.

You are now ready to license your Qlik Sense installation.

Licensing Qlik Sense

Before you can start using Qlik Sense you must activate your site license.

Do the following:

1. Open the Qlik Management Console (QMC) by entering the QMC address in your browser. By default, the QMC address is `https://<QPS server name>/qmc`. The QMC displays the **Site license properties** screen the first time you open it.

2 Installing Qlik Sense Enterprise on Windows

2. In the **Signed license key** field, paste the unique license key that was sent to you via email upon completion of your purchase.
3. Click **Apply**.
Successfully licensed is displayed.

You have activated your Qlik Sense site license.



For more information, see  [How to Apply a Signed License Key \(SLK\) to Qlik Sense Enterprise on Windows](#).

You are ready to connect to a user directory (optional), allocate user access or professional access, and set up permissions.

Allocating access to users

Your license is either based on access types, with professional access allocation as an option, or on tokens, with user access allocation as an option.

Access types license

Your Qlik Sense license includes a number of professional access allocations that are used to grant users in your organization access to Qlik Sense.

Do the following:

1. In the QMC, from the **Start** menu, click **License management**.
The **License usage summary** screen is displayed.
2. Click the **Professional access allocations** tab.
3. Click the **+ Allocate** button.
The **Users** screen is displayed.
4. Select the users that you want to provide access to from the list and click **Allocate**.



Allocate is disabled if the number of allocations available is insufficient for the number of selected users.

The users that you allocated access to appear in the **Professional access allocations** overview table.



In a multi-node site, all nodes share the same license, so you only need to activate your license once on the central node.

Token-based license

Your Qlik Sense license includes a number of tokens that are used to allocate Qlik Sense access to users in your organization.

Do the following:

1. In the QMC, from the **Start** menu, click **License management**.
The **License usage summary** screen is displayed.
2. Click the **User access allocations** tab.
3. Click the **+ Allocate** button.
The **Users** screen is displayed.
4. Select the users that you want to provide access to from the list and click **Allocate**.



Allocate is disabled if the number of tokens available for allocation is insufficient for the number of selected users.


The users that you allocated access to appear in the **User access allocations** overview table.



In a multi-node site, all nodes share the same license, so you only need to activate your license once on the central node.

Installing a Qlik Sense rim node

A multi-node site consists of a central node and at least one rim node. Rim nodes let you designate specific nodes to handle specific roles in your site. Before you install Qlik Sense Enterprise on Windows on a new server, you must:

- Create the required [user accounts](#) on the server.
- Download the Qlik Sense Enterprise on Windows installer from  [Product Downloads](#).
For more information, see *Downloading installation files (page 17)*.

Do the following:

1. Right-click the installer file and select **Run as administrator**. Click **Install** to start the installation.
2. On the **License Agreement** window, read and accept the Qlik User License Agreement.
3. On the **Create or join a cluster** screen, select **Join a cluster** to install a rim node that connects to a central node.

Create or join a cluster ?



Create a cluster

Create a single-node deployment or the first node in a cluster.

Create a cluster



Join a cluster

Create a node to join an existing cluster.

Join a cluster

4. On the **Set up the repository database** window, select **Connect to an existing PostgreSQL database**.

Set up the repository database ?



Connect to an existing PostgreSQL database

Use an existing PostgreSQL database.

Connect to an existing PostgreSQL database

Enter database credentials

To connect to an existing repository database, enter the PostgreSQL database host name and provide the database user password.

Database host name ?

rd-bdm-inst.rdlund.qliktech.com

Database port ?

5432

Database user ?

qliksenserepository

Database user password



For additional support and to learn about common issues related to the Qlik Sense Enterprise on Windows installation, see *Common issues and solutions to problems with your installation* (page 125).

2 Installing Qlik Sense Enterprise on Windows

5. On the **Provide information for the Qlik services (1/2)** window, enter the username and password to run the services on the current computer.

Under **Additional server settings** you can set the database maximum connection pool size and listening ports for http and https. https is the default option, to use http you need to select **Enable http**.

Database maximum connection pool size: This value is calculated by the installer, based on the server's configuration. The default value is usually the best option.



The service credentials username must be in the form domain\username.



If you enter a username that is more than 20 characters long, it must be in User Principal Name (UPN) format, and must include the full domain name. For example, longusername@full.domain.name.

Provide information for the Qlik services (1/2)

Settings and account information needed for Qlik Sense services installed on the host computer to work properly.

Windows service account credentials ?

Username

Password

▼ **Additional server settings** ?

Database maximum connection pool size ?

Https port number (https://rd-fpc-dd24.rdlund.qliktech.com:443)

Http port number (http://rd-fpc-dd24.rdlund.qliktech.com:80)


 Enable http



For additional support and to learn about common issues related to the Qlik Sense Enterprise on Windows installation, see [Common issues and solutions to problems with your installation](#) (page 125).

6. On the **Installation location** window, enter the location to install Qlik Sense Enterprise on Windows, or choose the default location on the C:\ drive.
7. On the **Read to install** window, clear **Start the Qlik Sense services when the setup is complete** if you want to use a dedicated service account to run the Qlik Sense services. In the **Help us Improve** section, select if you want to anonymously share system data with Qlik.

Ready to install

Start the Qlik Sense services when the setup is complete 

Help us improve

Qlik collects systems and usage data to optimize, support, and improve our products and services. This data is anonymized. If you prefer to not share this data, clear this box. Learn more about our privacy policy [here](#).



For additional support and to learn about common issues related to the Qlik Sense Enterprise on Windows installation, see *Common issues and solutions to problems with your installation* (page 125).

8. You will see a message indicating that Qlik Sense Enterprise on Windows has been installed successfully.



If you selected *Local System* as the [user account](#) type in the **Service Credentials** screen, but wish to use a dedicated service account to run the Qlik Sense services, change the user account type and manually start the Qlik Sense services now.

Connecting and configuring the nodes

After installing a central node and a rim node, configure the central node to connect to the rim node. Before you can verify that a rim node is running correctly you must connect it to the central node. Use the QMC on the central node to register a rim node.

To configure a central node to connect to a rim node:

1. On the central node, open the QMC, and click **Nodes**.
2. Click **Create new**.
3. In the **Edit node** window, enter the following configuration details about the node you want to connect to:

Qlik Sense node configuration

Field name	Description	Example value
Name	Provide a suitable name for the node.	For example, <i>Consumer node 1</i>
Host name	Enter the full URL of the node you want to connect to.	For example, <i><domain>-<server-name>.qliktech.com</i>

2 Installing Qlik Sense Enterprise on Windows

Node purpose	<p>Choose a suitable purpose for the node:</p> <ul style="list-style-type: none">• Production• Development• Both	<p>For example, choose Production for a scheduler node or Development for a developer node used for creating apps.</p> <p>Check that your license supports the node purpose that you have chosen.</p>
Node configuration	<p>Select this node as a failover candidate.</p>	<p>For example, if you select this node as a failover candidate it means that this node can perform the same role as the central node if the central node fails. See: <i>Configuring failover for central node resiliency (page 129)</i></p>
Service activation	<p>Select the services you want to run on this server node:</p> <ul style="list-style-type: none">• Repository• Engine• Printing• Proxy• Scheduler	<p>For example, if you are installing a consumer node, select the Repository and Engine services.</p> <p>For more information on which services to run on different types of nodes, see: <i>Qlik Sense Enterprise on Windows architecture (page 28)</i> and <i>Services (page 30)</i></p>

4. Click **Apply**. The central node generates a certificate that you use to register the rim node. If the central node cannot connect to the rim node you will see a **Node registration** error message. If you get this error, first check that you have opened port 4444 on the central and rim nodes to allow certificates to be sent.
5. The **Install certificates** pop-up window then opens providing you with a URL and a password to authorize the certificate on the rim node.
6. On the rim node, paste the URL into a new browser window.

2 Installing Qlik Sense Enterprise on Windows

7. On the **Install certificates** page (in your browser), enter the password and click **Submit**. If successful, you see the **Successfully licensed** message.
8. Follow the same authorization procedure for each node that you want to add to your deployment.
9. To verify that all rim nodes are configured correctly, open the QMC, click **Nodes** and you can see the status of all the nodes in your deployment.

Verify your installation

To verify that Qlik Sense has installed correctly:

1. Open the Qlik Management Console (QMC).
2. Open the Qlik Sense Hub.

If the QMC and Hub open without any security warnings displayed in the browser, then you have installed Qlik Sense correctly.

Additional configuration

After you have installed and verified that Qlik Sense is running correctly, you may find the following configuration information useful:

- Load balancing - create load balancing rules in the QMC to improve resilience and performance in a multi-node site.
- Host allow list - configure the virtual proxy advanced settings to add your own hosts names to the allowlist.
- User imports (UDC) - configure the user directory connector to retrieve users from a user directory.

Qlik Sense is ready to be used. But before getting started, back up your files so that you can recover from a system crash, see *Backup and restore Qlik Sense Enterprise on Windows (page 181)*



If you are installing custom connectors in a multi-node setup, the custom connectors must be installed on each node.

Modifying an object bundles installation

You can add or remove object bundles from your Qlik Sense deployment at any moment. If you have a multi-node installation, object bundles are installed on the central node.

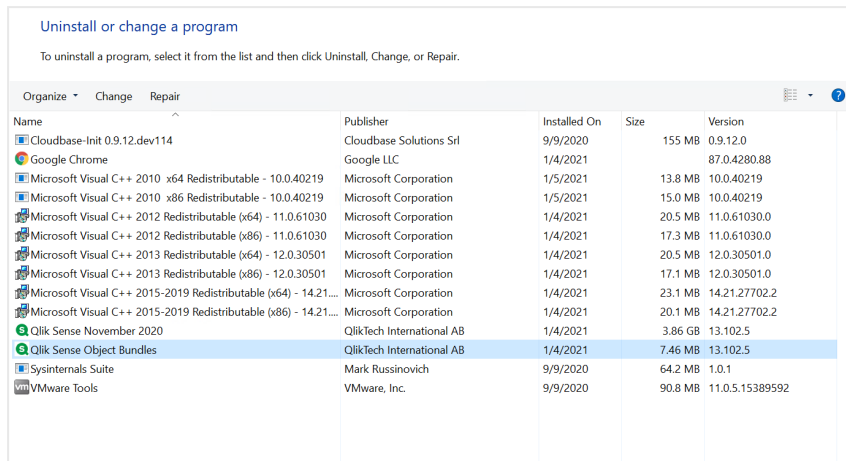


*You can see which extensions are installed in your deployment by checking the **Extensions** section in the Qlik Management Console (QMC).*

Do the following:

1. In **Windows Control Panel**, open **Programs and Features**.
2. In the list of programs, double-click the object bundle that you want to modify.

2 Installing Qlik Sense Enterprise on Windows



3. The Object Bundle Setup Wizard opens. Click **Next**.
4. Select **Change**.
5. On the **Custom setup** screen, click on the bundle icon to select how to modify the bundle installation:
 - If the bundle is installed, select **Entire feature will be unavailable** to uninstall it.
 - If the bundle is not installed, select **Entire feature will be installed on local hard drive** to install it.Then, click **Next**.
6. Click **Change**.
Once the modification is complete, you are required to manually restart the Qlik Sense Repository Service.
7. Click **Finish** to close the Object Bundle Setup Wizard.
8. Manually restart the Qlik Sense Repository Service to apply the changes.

You can verify that the changes have been correctly applied by checking the **Extensions** section in the QMC.

Common issues and solutions to problems with your installation

Windows user accounts and service credentials

- [Changing the Qlik Sense Service Account and what to consider](#)
- [Interactive Logon Rights for Qlik Sense Installations](#)
- [How to: Change the Qlik Sense Proxy certificate if the service account does not have local administrative permissions](#)
- [Qlik Sense repository service fails to start due to error " 'SeSecurityPrivilege " error](#)

Host name

- [Qlik Sense: Change hostname \(and certificates\) after an installation](#)
- [Authentication fails repeatedly when using external DNS alias locally on the server](#)
- [How to change the certificate used by Qlik Sense Hub and QMC](#)

Shared persistence

- [Qlik Sense Share Storage options and requirements](#)
- [How to change the share path in Qlik Sense \(Service Cluster\)](#)
- [Qlik Sense Logging Service does not have support for SSL Database traffic encryption](#)
- [Configure Qlik Sense to use a dedicated PostgreSQL server](#)

Qlik Sense extension bundles

- [Installing and removing Qlik Sense Extension bundle](#)

Anti-virus software

- [Antivirus exceptions for Qlik Sense- McAfee, Symantec and Other Anti-Virus exclusions absolutely required](#)

2.3 Creating a file share

Creating a file share or shared folder is a prerequisite before you install Qlik Sense. The file share stores all binary app data, including the data models and dashboard sheets. It must be accessible to all nodes in your Qlik Sense site. You can create a file share either on the same server as the central node or on a separate server. If you have a large multi-node site we recommend that you configure the file share on a dedicated server for better resilience and performance.

If you create the file share on a separate server then you can follow the same steps as for a central node but you must ensure that the same Windows domain user that you use to run the Qlik services has read and write access to the file share folder.

To create a file share and share the folder with specific users:

1. Create a local folder on your server computer. For example, create a folder called *QlikShare* on the C:\ drive.
2. Right click the folder, and then click **Properties**.
3. Click the **Sharing** tab, and then click **Share**.
4. Enter the name of your Windows user, and click **Add**.
5. In the **Permission level** column, select **Read/Write**, then click **Share**.



Make a note of the network path shown in the confirmation screen as you use this later during setup of your shared persistence storage folders. The network path will be in the following format: \\server-name\QlikShare

Ensure that permissions on the folder, subfolders, and files are set to full control for the user account you selected.

Do the following:

2 Installing Qlik Sense Enterprise on Windows

1. Click the **Security** tab.
2. Select the user account you want to use for the installation.
3. Click **Advanced** and verify that your current user has full control, and that this permission applies to the folder, subfolders, and files.
4. Click the **Effective Access** tab, then click **Select a user** and enter your user account name.
5. Click **View effective access**, then check the **Permission** column that your user has full control.



A file share cannot be used for more than one Qlik Sense site. If two sites use a single file share, there is a potential for file locking issues, and there can be incorrect meta-data references because data in the file share is being updated by two independent sites.

Creating an NFS file share

You can host an NFS file share on a Microsoft Windows Server and map it to your Qlik Sense Enterprise on Windows installation. Before you create an NFS file share, you must first enable the **Server for NFS** feature on the Windows server.

To enable Server for NFS, do the following:

1. Open the Windows Server Manager.
2. Click **Manage**.
3. Click **Add Roles and Features**.
4. In the **Add Roles and Features Wizard**, select the following:

Configuration setting	Instruction
Installation Type	Select Role-based or feature-based installation
Server Selection	Use the pre-selected server.
Server Roles	Under Files And Storage Services > File and iSCSI Services , select Server for NFS .
Features	Ensure that Client for NFS is selected.
Confirmation	Click Install .
Results	Verify the install is successful. You will need to restart the server.

To create a NFS file share, do the following:

1. Create a new folder on your Windows Server.
2. Right-click the folder, then click **Properties**.
3. Under the **NFS Sharing** tab, select **Share this folder**.
4. Enter a **Share name**.
5. Click **Permissions**.

2 Installing Qlik Sense Enterprise on Windows

6. Click **Add** to add IP addresses or host names that can connect to the NFS file share, and set their **Type of access** as needed. Click **OK**
7. Click **Apply**, then **OK**.



Your NFS file share is listed under **Shares** in the Windows Server Manager. You can access the NFS share information by right-clicking the share, then selecting **properties**. When you install Qlik Sense Enterprise, use the NFS file share remote path when setting up your shared persistent storage. Enter the remote path in the following format: `\\server-name\share-name`.

Changing the file share path

You can change the Qlik Sense file share path after installing Qlik Sense Enterprise on Windows by using the `QlikSenseUtil.exe`, which is included with the Qlik Sense installer.

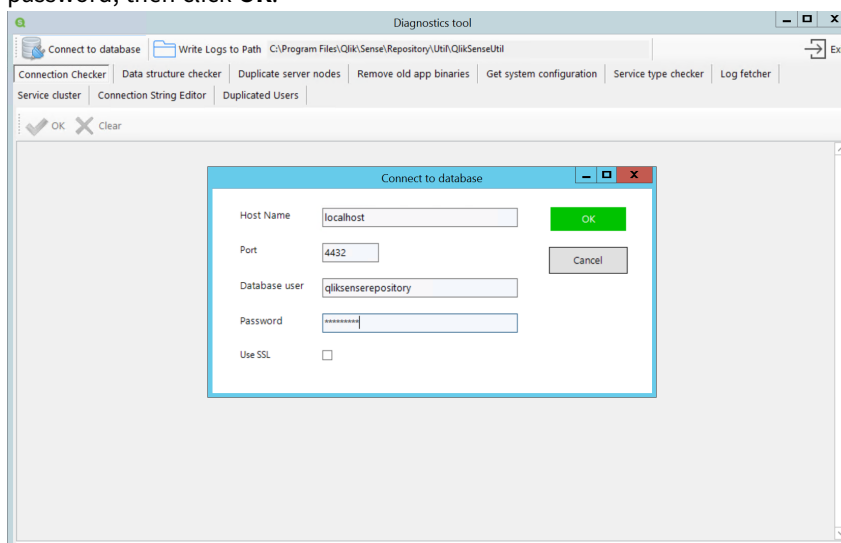
To change the file share path, do the following:

1. On the file share server, create a new file share.
2. Copy the share path to a notepad. You need to add the new share path to the `QlikSenseUtil.exe` later.
3. Stop all Qlik services except the Qlik Sense Repository Service.
4. Move the `Apps`, `ArchivedLogs`, and `StaticContent` folders from your existing file share to your new file share.
5. Run the `QlikSenseUtil.exe` as administrator.



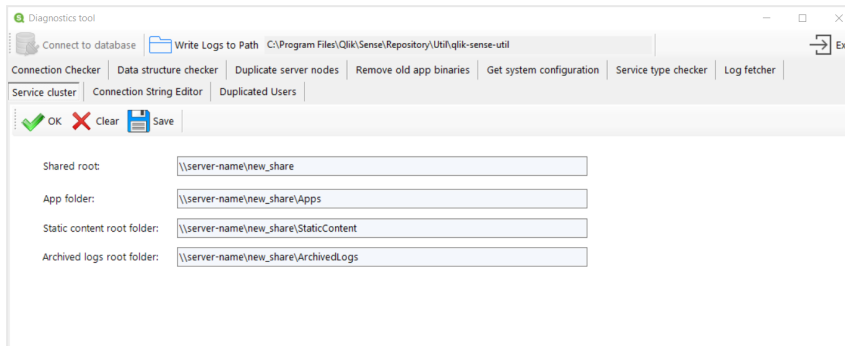
If you used the default installation path for Qlik Sense, `QlikSenseUtil.exe` is located: `%Program Files%\Qlik\Sense\Repository\Util\QlikSenseUtil\QlikSenseUtil.exe`

6. From the **QlikSenseUtil** window, click **Connect to database**.
7. On the **Connect to database** window, enter the Qlik Sense repository database username and password, then click **OK**.



8. Select **Service cluster**, then click **OK** to retrieve the current file share path.

9. Replace the current file paths with the new file paths, then click **Save**.
10. Verify that the file share paths have been updated correctly by clicking **Clear**, and then **OK**. The fields should populate with the new file paths.



11. Restart the Qlik services that you closed.
12. Restart the QMC.

2.4 Configuring failover for central node resiliency

In a multi-node site, you can assign a node to be a failover candidate. A failover candidate can perform the same role as the central node in the case where the central node fails. A multi-node site with a designated failover candidate can help you achieve a more resilient and highly-available deployment.

Failover considerations

Before you create a failover candidate node, it is important to consider your deployment architecture. A failover candidate node can help you maintain a resilient and highly available deployment by minimizing the downtime of your site if the central node fails. However, the failover candidate node provides failover capacity only for the Qlik Sense services running on the central node. If you want to create a highly available deployment, you must add resiliency to the storage layer as well.



Each node in your multi-node site must meet the minimum system requirements. For a complete list, see [System requirements](#).

Storage layer resiliency

If the storage components reside on the central node when it fails, they become unavailable because the failover candidate node does not provide failover for the storage components. You can add resiliency to your repository database and the file share by deploying them on a separate node from your central node. Other options to add resiliency are:

- Deploy a standalone database on a virtual machine and take advantage of the resiliency options provided by the virtualization platform.

2 Installing Qlik Sense Enterprise on Windows

- Host the file share in a network file location or a storage area network (SAN), or use resilient storage provided by a cloud platform.

For information about database replication and failover, see *Database replication and failover* (page 243).

Create a failover candidate node

When you create your multi-node site, you first create the central node and then you join additional nodes to the cluster. From the QMC, you can set one of these non-central nodes to be the failover candidate. The failover candidate will take over the responsibility of the central node if it fails. To set a failover candidate node, see [Create a node](#).



The failover candidate node can have different functions depending on your deployment. For example, the node that is designated to be the failover node can be the scheduler node in your multi-node site, as long as it has the required Qlik services to also be the failover candidate node.

Once you add more nodes to your site, you can assign one or more of them to be a failover candidate. For a node to be a failover candidate, it must run the following services:

- Qlik Sense Repository Service
- Qlik Sense Engine Service
- Qlik Sense Proxy Service
- Qlik Sense Scheduler Service

Automatic failover

In a multi-node site, each node regularly checks the central node for a heartbeat. If after 10 minutes (the default timeout period is 10 minutes) there is no response from the central node, the site will automatically fail over to the failover candidate node. If there is more than one failover candidate node, the first node to get a lock on the database field becomes the central node. If the node that was previously the central node comes back online, it becomes a failover candidate node.

You can view the status of the nodes that make up your multi-node site in the QMC. The default view does not include the node type, but you can customize the node information that is displayed. To see the node information, and to configure the information displayed about each node, see [Nodes](#).

The default timeout period for the central node is 10 minutes, but you can change it in the QMC. To change the default timeout, see [Cluster settings](#).

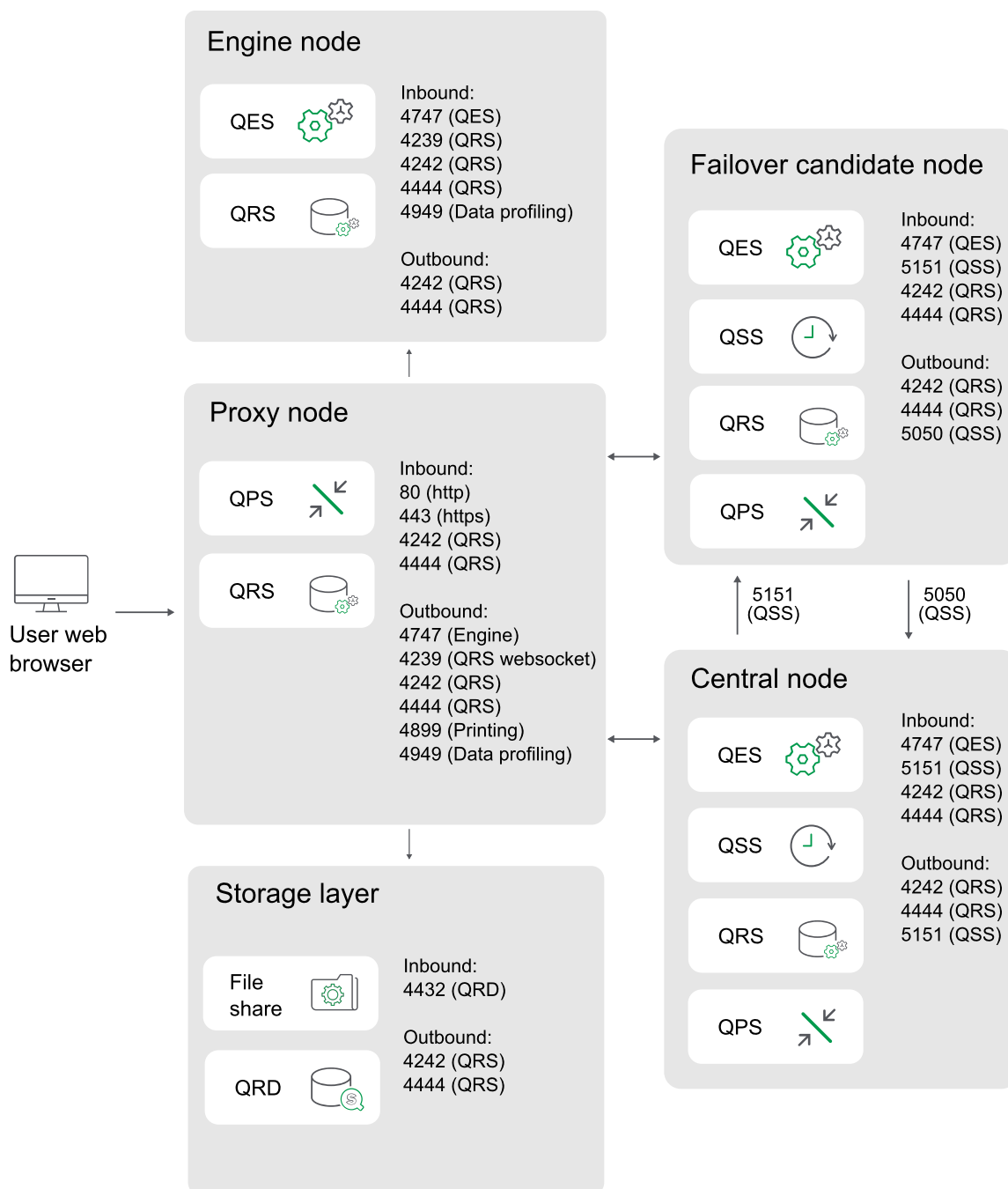
Failover candidate node with inbound and outbound ports

As mentioned above, the failover candidate can have different roles, depending on your organizational needs. The example below shows a multi-node site with a single failover candidate node that is running as the worker scheduler in this site. The failover candidate must have the same inbound and outbound ports open as the central node. As the failover node acts as the worker scheduler, therefore, port 5151 and 5050 must be open inbound and outbound on their respective nodes for scheduling jobs to the failover candidate node.

2 Installing Qlik Sense Enterprise on Windows



Inbound ports indicate the listening ports for the services running on each node. Firewall rules must allow inbound traffic to these ports. Outbound ports indicate the destination of the communication from one node to other nodes in the environment. Firewall rules must allow the node to send outbound traffic to these outbound ports.



For a complete list of inbound and outbound ports for all services, and to see more deployment examples, see *Ports* (page 42).

Manually migrating the central node

You cannot use the QMC to change which node in your site is the central node. You can, however, use the QRS REST API to do this. Before manually reassigning a failover candidate node to a central node role, you must ensure that it is running the necessary services for the central node.

Use the following REST API calls:

- Run a GET to `/qrs/serverNodeConfiguration` to return a list of server GUIDs.
- Run an empty POST to `/qrs/failover/tonode/{serverNodeConfigurationID}` where `{serverNodeConfigurationID}` is the ID of the node you want to become the central node.

2.5 Installing and configuring PostgreSQL

To improve performance in a Qlik Sense multi-node deployment, you have the option to install your repository (QSR), SenseServices, QSMQ, and Licenses databases on a dedicated, remote PostgreSQL server.



The Qlik PostgreSQL Installer offers an easy way to install the PostgreSQL, see: [Installing or upgrading PostgreSQL using the Qlik PostgreSQL Installer \(page 139\)](#).

Databases

The QSR, SenseServices, QSMQ, and Licenses databases share the same login role and must be installed on the same PostgreSQL instance. If you already have a PostgreSQL database installed as part of a previous deployment, you can continue to use it.



In Qlik Sense Enterprise, configuring all the components of a Multi-Cloud deployment is optional. However, all deployments, whether Multi-Cloud or on-premise require the installation of the SenseServices database and QSMQ databases.



If Qlik Sense uses a PostgreSQL database on a dedicated infrastructure, then it can use supported versions of PostgreSQL. To know which versions of PostgreSQL are supported in Qlik Sense, see: [System requirements for Qlik Sense Enterprise](#). You can run the instance of PostgreSQL on platforms including Windows, Linux or cloud hosted services, such as Amazon RDS. However, Qlik will only offer configuration support when PostgreSQL is running on Windows. If you use Linux or Amazon RDS, it is your own responsibility to install and configure a running instance of PostgreSQL for Qlik Sense to use.

The Qlik Sense repository database (QSR)

The QSR is the primary database in your Qlik Sense deployment.

2 Installing Qlik Sense Enterprise on Windows

If you want to install the QSR database on a dedicated PostgreSQL server, you must install and configure PostgreSQL before you install Qlik Sense, as you will need to enter the PostgreSQL server/host details in the Qlik Sense installer.

The Qlik Sense services database (SenseServices)

The SenseServices database contains schemas for each of the Qlik Sense services and allows growth independently of the Qlik Sense Repository Database, while still sharing the same PostgreSQL instance and login role.

The Qlik Sense message queue database (QSMQ)

The QSMQ database provides a light-weight method of passing messages internally between services in Qlik Sense Enterprise. The NOTIFY and LISTEN functionality in PostgreSQL allows services to be notified about new messages that have been written to the messaging table.

The licenses service database (Licenses)

The licenses database contains a local copy of license data to allow faster response times and more robustness. It is only accessed by the licenses service.

To install a dedicated PostgreSQL server with QSR, SenseServices, QSMQ, and Licenses database:

- Install PostgreSQL
- Create the PostgreSQL databases, and configure login roles.
- Edit the configuration files to allow access from Qlik Sense nodes.
- Verify that the database has installed and is running correctly.

Installing PostgreSQL

Before installing a dedicated PostgreSQL server instance, check that your server fulfills the system requirements on www.postgresql.org.



If you are installing on Microsoft Azure with an Azure database for PostgreSQL, follow the instructions in [Installing and configuring PostgreSQL on Azure](#) (page 147).

To install PostgreSQL on a dedicated server:

1. Log in to the server where you want to install PostgreSQL as an administrator.
See: [User accounts](#) (page 63)
2. Download PostgreSQL EnterpriseDB version 12.x from the [PostgreSQL](#) website.
To know which versions of PostgreSQL are supported in Qlik Sense, see: [System requirements for Qlik Sense Enterprise](#)
3. Run the **PostgreSQL setup wizard**.
4. On the **Installation Directory** and **Data Directory** screens, accept the default paths.
5. On the **Password** screen, create a password for the PostgreSQL superuser.
You will use this password when you connect to the PostgreSQL database and you will also be prompted for it when you run the Qlik Sense setup.

6. On the **Port** screen, specify port 4432. This port is required for communication between all the nodes in a site.
7. In the **Advanced Options** screen, accept the default locale.
8. In the **Ready to Install** screen, click **Next** to run the setup.
9. After running the setup, you have the option to install *Stack Builder*. Clear the check box if you want to install this later.
10. Click **Finish** to complete the installation.

When you install PostgreSQL EnterpriseDB, the pgAdmin tool is included.

Creating a PostgreSQL database

You can create a repository QSR, SenseServices, QSMQ, and Licenses database manually with the pgAdmin tool or using a script.

To create a new, empty PostgreSQL database using the pgAdmin tool:

1. Open the *pgAdmin* tool.
2. In the *pgAdmin Browser*, under **Servers**, right-click the PostgreSQL node and then click **Connect Server**.
3. Enter your PostgreSQL superuser password to make a connection. A green status bar appears in the lower right corner of your screen when the server connection is successful.
4. Right-click the **Databases** node, click **Create**, and then click **Database**.
5. Enter the name of the database you are creating, and then click **Save**.

To create a new, empty PostgreSQL database by running a script in the pgAdmin tool:

1. Open the **Query Tool**. First select an existing database, such as **postgres**, to display the **Query Tool** option in the **Tools** menu.
2. Execute the following script:

```
CREATE DATABASE "<databasename>" ENCODING = 'UTF8'; --creates an empty database.
```

Replace <databasename> with QSR for the repository database, SenseServices for the SenseServices database, QSMQ for the message queue database, Licenses for the license service.

Creating login roles

You need to create login roles for users when you create a PostgreSQL database. You can create login roles using the pgAdmin tool or by running a script.

The QSR, SenseServices, QSMQ, and Licenses login role

To create login roles using the pgAdmin tool:

1. Right-click the **Login/Group Roles** node. To create a new database user, click **Create**, and then click **Login/Group Role**.
2. In the **Create - Login/Group Role** window, in the **General** tab, enter the name *qliksenserepository*.
3. In the **Privileges** tab, enable **Can login?** and leave the other default privileges unchanged.
4. In the **Definition** tab, enter a password of your choice, and click **Save**.
When you run the Qlik Sense setup, in the **Shared persistence database connections settings** screen,

2 Installing Qlik Sense Enterprise on Windows

you are asked to enter the **Database user** password that you created here so that Qlik Sense can connect to the repository database.

5. Make *qliksenserepository* the owner of the **QSR**, **SenseServices**, **QSMQ**, and **Licenses** databases. To do this, right-click the **QSR**, **SenseServices**, **QSMQ**, and **Licenses** databases you created earlier, and then click **Properties**.
6. In the **General** tab, in the **Owner** drop-down, select *qliksenserepository* as **Owner** of the **QSR**, **SenseServices**, **QSMQ**, and **Licenses** databases and click **Save**.

To create login roles by running a script in the pgAdmin tool:

Open the **Query Tool**. Select an existing database, to display the **Query Tool** option in the **Tools** menu.

Run the following script:

```
CREATE ROLE "qliksenserepository" WITH LOGIN NOINHERIT NOSUPERUSER NOCREATEDB NOCREATEROLE  
NOREPLICATION VALID UNTIL 'infinity'; -- change <qliksenserepository_user_pass> to your  
password for the repository service user  
ALTER ROLE "qliksenserepository" WITH ENCRYPTED PASSWORD '<qliksenserepository_user_pass>';  
GRANT qliksenserepository TO postgres;
```

```
ALTER DATABASE "QSR" OWNER TO "qliksenserepository";  
ALTER DATABASE "SenseServices" OWNER TO "qliksenserepository";  
ALTER DATABASE "QSMQ" OWNER TO "qliksenserepository";  
ALTER DATABASE "Licenses" OWNER TO qliksenserepository;
```

```
GRANT TEMPORARY, CONNECT ON DATABASE "QSMQ" TO PUBLIC;  
GRANT ALL ON DATABASE "QSMQ" TO postgres;  
GRANT CREATE ON DATABASE "QSMQ" TO "qliksenserepository";  
GRANT TEMPORARY, CONNECT ON DATABASE "SenseServices" TO PUBLIC;  
GRANT ALL ON DATABASE "SenseServices" TO postgres;  
GRANT CREATE ON DATABASE "SenseServices" TO "qliksenserepository";
```

```
GRANT TEMPORARY, CONNECT ON DATABASE "Licenses" TO PUBLIC;  
GRANT ALL ON DATABASE "Licenses" TO postgres;  
GRANT CREATE ON DATABASE "Licenses" TO qliksenserepository;
```



Include a password for qliksenserepository as you will be prompted for this when you install Qlik Sense.

Granting permissions for PostgreSQL versions 15 and 16

To use PostgreSQL versions 15 and 16 with Qlik Sense, you need to run the following script to grant the necessary permissions. This ensures that the *qliksenserepository* user has the required access to the public schema in all Qlik Sense databases.

You can run the script via the command line (psql) or in pgAdmin. Follow the instructions below for your preferred method.

SQL script

```
CREATE DATABASE "QSR" ENCODING = 'UTF8' TEMPLATE=template0;  
CREATE DATABASE "SenseServices" ENCODING = 'UTF8' TEMPLATE=template0;
```

2 Installing Qlik Sense Enterprise on Windows

```
CREATE DATABASE "QSMQ" ENCODING = 'UTF8' TEMPLATE=template0;
CREATE DATABASE "Licenses" ENCODING = 'UTF8' template=template0;
CREATE ROLE "qliksenserepository" WITH LOGIN NOINHERIT NOSUPERUSER NOCREATEDB NOCREATEROLE
NOREPLICATION VALID UNTIL 'infinity';
--Replace `your_password` with the desired password for the repository user
ALTER ROLE "qliksenserepository" WITH ENCRYPTED PASSWORD 'your_password';
GRANT qliksenserepository TO postgres;

ALTER DATABASE "QSR" OWNER TO "qliksenserepository";
ALTER DATABASE "SenseServices" OWNER TO "qliksenserepository";
ALTER DATABASE "QSMQ" OWNER TO "qliksenserepository";
ALTER DATABASE "Licenses" OWNER TO "qliksenserepository";

GRANT TEMPORARY, CONNECT ON DATABASE "QSMQ" TO PUBLIC;
GRANT ALL ON DATABASE "QSMQ" TO postgres;
GRANT CREATE ON DATABASE "QSMQ" TO "qliksenserepository";

GRANT TEMPORARY, CONNECT ON DATABASE "SenseServices" TO PUBLIC;
GRANT ALL ON DATABASE "SenseServices" TO postgres;
GRANT CREATE ON DATABASE "SenseServices" TO "qliksenserepository";

GRANT TEMPORARY, CONNECT ON DATABASE "Licenses" TO PUBLIC;
GRANT ALL ON DATABASE "Licenses" TO postgres;
GRANT CREATE ON DATABASE "Licenses" TO "qliksenserepository";

GRANT TEMPORARY, CONNECT ON DATABASE "QSR" TO PUBLIC;
GRANT ALL ON DATABASE "QSR" TO postgres;
GRANT CREATE ON DATABASE "QSR" TO "qliksenserepository";

--Connect to each database and grant full permissions on the public schema to
qliksenserepository
\c QSR
GRANT ALL ON SCHEMA public TO qliksenserepository;
\c Licenses
GRANT ALL ON SCHEMA public TO qliksenserepository;
\c QSMQ
GRANT ALL ON SCHEMA public TO qliksenserepository;
\c SenseServices
GRANT ALL ON SCHEMA public TO qliksenserepository;
```

Running the SQL script using psql

1. Save the SQL script in a text file with a .sql extension (for example, script.sql).
2. Open a command prompt or terminal and run the script with the following command, replacing the placeholders with your actual values:

```
psql -h localhost -U postgres -p 4432 -f c:\path\to\your\script.sql
```

Running the SQL script using pgAdmin


1. Open the pgAdmin application, connect to the PostgreSQL server, and open the Query Tool.
2. Copy and paste the SQL script into the Query Tool and execute it.
3. Finally, for each database (QSR, QSMQ, SenseServices, and Licenses), run the following command to


2 Installing Qlik Sense Enterprise on Windows

grant permissions on the public schema:
GRANT ALL ON SCHEMA public TO qliksenserepository;

Configuring PostgreSQL

To allow communication between your PostgreSQL repository database and your Qlik Sense nodes, edit the `pg_hba.conf` and `postgresql.conf` configuration files.

 Make a backup copy of the `postgresql.conf` and `pg_hba.conf` files before you start, so that you have the option to revert back to the original settings.

 The paths in the instructions are adapted to a default PostgreSQL installation used as database on a dedicated server. A PostgreSQL database installed by Qlik Sense has the following database path:
`%ProgramData%\Qlik\Sense\Repository\PostgreSQL\<version>\.`

postgresql.conf

The `postgresql.conf` file enables you to specify general parameters for your PostgreSQL server, such as for auditing, authentication, and encryption. Edit this file to control which Qlik Sense nodes can access your PostgreSQL database server.

To edit the `postgresql.conf` file:

1. Navigate to the `postgresql.conf` file in `C:\Program Files\PostgreSQL\<version>\data` of your PostgreSQL installation.
2. Open the file in a text editor as an administrator.
3. Make the following configuration changes:

PostgreSQL configuration changes

Setting	Description	Example Value
<code>listen_addresses</code>	Enter the IP address(es) to listen on. If entering multiple listen addresses, use a comma separated list. Enter * to listen for connections from all IP addresses.	*
<code>max_connections</code>	Specifies the maximum number of concurrent connections to the database. The default value for a single server is 100. In a multi-node environment, this should be adjusted to the sum of all repository connection pools + 20. By default, this value is 110 per node.	

4. Save your changes.

For more detailed information about setting these parameters, see the [PostgreSQL](#) documentation.

pg_hba.conf

The `pg_hba.conf` file handles client authentication. Each record specifies a connection type, such as a client IP address range, database name, user name, and the authentication method used.

To edit the `pg_hba.conf` file:

1. Navigate to the `pg_hba.conf` file in `C:\Program Files\PostgreSQL\<version>\data` of your PostgreSQL installation.

2. Open the file in a text editor as an administrator.

3. Locate the following line:

```
host      all      all      127.0.0.1/32      md5
```

This line determines which servers can access the repository database server. The default address setting, `127.0.0.1/32`, only allows local host to access the database.

4. Replace `127.0.0.1/32` with a sub net specification that covers all the IP addresses of the nodes in your site.

When specifying these settings, add one row for each node, using `/32` as a suffix for each address, or add a sub net that covers all addresses using, for example, `/24` as a suffix:

- IPv4 (32-bit addresses):
 - To specify a single address: `192.168.1.0/24`, or `172.20.143.89/32`
 - For a small network: `172.20.143.0/24`, or `10.6.0.0/16` for a larger one.
 - To allow access from all IPv4 addresses: `0.0.0.0/0`
- IPv6 (128-bit numeric addresses):
 - For a single host: `::1/128` (in this case the IPv6 loopback address)
 - For a small network: `fe80::7a31:c1ff:0000:0000/96`
 - To allow access from all IPv6 addresses: `::/0`



When you add the IPv6 connections and use hostname in the address column, both the forward and reverse `nslookup` of the client machine must return valid values for PostgreSQL to accept the connection from the client. For more information refer to the [PostgreSQL documentation](#).

5. Save your changes.

For more information on how to set a more restrictive IP address, see the [PostgreSQL](#) documentation.

You have installed and configured a PostgreSQL database on a separate server. You are now ready to resume your installation of Qlik Sense.

2.6 Installing or upgrading PostgreSQL using the Qlik PostgreSQL Installer

The Qlik PostgreSQL Installer (QPI) facilitates your installation or upgrade of the bundled PostgreSQL database. The QPI can be used when installing a completely new standalone database based on PostgreSQL 14.8, or when upgrading and unbundling an existing bundled database from PostgreSQL 9.6 and PostgreSQL 12.5 to PostgreSQL 14.8.

About the Qlik PostgreSQL Installer

As of March 2023, PostgreSQL 9.6 is no longer supported. Using the QPI when upgrading to PostgreSQL 14.8 simplifies the upgrade. The QPI also makes the installation of a new database easier.

When you launch the QPI, it detects any already existing PostgreSQL setup and starts the upgrade process. If no bundled PostgreSQL setup is detected, the QPI starts the installation process.

The default installation path is `C:\Program Files\PostgreSQL\14`. Data for the new database is stored at `C:\Program Files\PostgreSQL\14\data`.

You can change these paths, as long as they don't use the old path: `C:\Program Files\Qlik\Sense\Repository\PostgreSQL\12.5`. The change of paths is made to decouple PostgreSQL from Qlik Sense and facilitate maintenance and upgrading. With this change there is no dependency on upgrades to Qlik Sense Enterprise on Windows. Any attempt to use the old installation path is automatically prevented to avoid issues with the database being interpreted as bundled by the installer when upgrading.

The QPI only supports Qlik Sense Enterprise on Windows February 2022 and above.

When upgrading, the database backup is stored in `C:\ProgramData\Qlik\Sense\Repository\PostgreSQL\backup\9.6`. The 9.6 folder is renamed to `9.6_deprecated`, to avoid accidental references to the old 9.6 folder.



Using the Qlik PostgreSQL Installer not only upgrades PostgreSQL; it also unbundles PostgreSQL from your Qlik Sense Enterprise on Windows installation. This allows for direct control of your PostgreSQL instance and facilitates maintenance without a dependency on Qlik Sense. Further Database upgrades can then be performed independently and in accordance with your corporate security policy when needed, as long as you remain within the supported PostgreSQL versions. See

 [How To Upgrade Standalone PostgreSQL](#).

Supported scenarios

You can use the QPI in the following scenarios:


- Clean installation of PostgreSQL 14.8. The QPI allows to install a standalone PostgreSQL 14.8 database on a new server for a new environment, or to migrate an existing database to a separate host.

2 Installing Qlik Sense Enterprise on Windows

- Upgrade an existing bundled database from PostgreSQL 9.6 and 12.5 to 14.8. The QPI only supports Qlik Sense Enterprise on Windows February 2022 and above. It's recommended that you upgrade to the latest supported version of Qlik Sense before using the QPI.



Upgrading earlier versions of the Qlik Sense Repository Database (QRD) is not supported through this method. Instead, upgrade to a version of Qlik Sense that supports 9.6 to automatically migrate through the minor version.

For more information about the QPI, see:  [Upgrading and unbundling the Qlik Sense Repository Database using the Qlik PostgreSQL Installer.](#)

Considerations

Before you start installing or upgrading, consider the following:

- Using the QPI on a patched Qlik Sense version might lead to unexpected issues. If you have a patch installed, follow these steps:
 1. Uninstall all patches for your current Qlik Sense version (for example, revert Feb 2022 Patch X to Feb 2022 IR).
 2. Use QPI to upgrade the PostgreSQL database.
 3. Finally, upgrade Qlik Sense to your desired version.

Alternatively:


1. Upgrade Qlik Sense to an IR (initial release) of a later version that supports QPI.
 2. Use QPI to upgrade the PostgreSQL database.
- The QPI will not detect a standalone PostgreSQL database installed by any other means other than through an official Qlik installer package.
 - The QPI itself does not provide a rollback feature.
 - When using the QPI to upgrade the bundled PostgreSQL database, it not only upgrades the database to 14.8, but also unbundles it from Qlik Sense Enterprise on Windows installation.
 - The user who runs the QPI needs to have sufficient permissions to execute the QPI and install services.
 - The user who runs the QPI needs to have sufficient permissions to create folders and files in the backup destination.
 - The backup destination needs to have sufficient free disk space to dump the existing database.
 - The backup destination is not a network path or virtual storage folder. It's recommended to store the backup on the main drive.
 - There will be downtime during this operation, plan accordingly.

Installing a new Qlik Sense Repository Database using PostgreSQL 14.8

Do the following:

1. Stop all services on rim nodes.
2. Run the Qlik PostgreSQL Installer.
3. Click **Install**.

2 Installing Qlik Sense Enterprise on Windows

4. Accept the **Qlik Customer Agreement**.
5. Set the **Local database settings** and click **Next**. You will use these details to connect other nodes to the same cluster.
6. Set the **Database superuser password** and click **Next**.
7. Review the settings and click **Install** and then **Finish**.
8. Do one of the following:
 - a. Restore your existing QRD database, see: *Restoring a Qlik Sense site (page 203)*.
 - b. Start installing Qlik Sense Enterprise Client-Managed, see:  [How to configure Qlik Sense to use a dedicated PostgreSQL database](#).
9. When the installation is complete, close the installer.
10. Click **Finish** to complete the installation.

Upgrading an existing Qlik Sense Repository Database using PostgreSQL 14.8 - Central node



It is required to first upgrade Qlik Sense to the latest supported version with QPI. Qlik Sense Enterprise on Windows February 2022 and above are supported versions for the QPI.

Do the following:

1. Stop all services on the rim nodes.
2. On your central node, stop all services except the Qlik Sense Repository Database service.
3. Run the Qlik PostgreSQL Installer. The installer will detect the existing database.
4. Click **Upgrade**.
5. Accept the **Qlik Customer Agreement**.
6. Enter your **Database superuser password** and click **Next**.
7. Set the backup location and click **Next**.
8. Click **Upgrade** and then **Finish**.
A new service named *postgresql-x64-14* is now installed. It's run by the **Network Service** account. Don't change the name of the service and don't modify the account running it.
9. Start all services except the Qlik Sense Repository Database service.
10. Start all services on the rim nodes.
11. Validate that all services and nodes are operating as expected.
12. Once the upgrade and all validation are complete, remove the old Qlik Sense Repository Database from the server.



This step is required. Failing to remove the old service will lead to upgrade or patching issues.

1. Open a command prompt and run:
c:\cd ProgramData\Package Cache
2. From there, do one of the following:

2 Installing Qlik Sense Enterprise on Windows

- If you upgraded from 9.6, run:
c:\ProgramData\Package Cache\dir /s PostgreSQL.msi
The folder containing *PostgreSQL.msi* is displayed.
- If you upgraded from 12.5, run:
c:\ProgramData\Package Cache\dir /s PostgreSQL125.msi
The folder containing *PostgreSQL125.msi* is displayed.

3. Right-click the *PostgreSQL.msi* or *PostgreSQL125.msi* file and select **Uninstall** from the menu.

If the upgrade was unsuccessful and you are missing data in the Qlik Management Console or elsewhere, [contact Qlik Support](#).

Silently installing or upgrading PostgreSQL using the Qlik PostgreSQL Installer

When running a silent installation, PostgreSQL is installed with no dialogs at all. This means that all features, properties, and user selections must be known before performing a silent installation. All setup options that are available in the user interface of the installer can be performed with silent operations.

Do the following:

1. Open a **Command Prompt** window.
2. Navigate to the folder containing the *Qlik_PostgreSQL_setup.exe* file.
3. Enter *Qlik_PostgreSQL_setup.exe* followed by the preferred silent installation *Syntax (page 143)*.

Example installation command:

```
Qlik_PostgreSQL_setup.exe -silent acceptqca=1 iprange=0.0.0.0/0,::/0  
dbport=4432 maxconnections=200 listenaddresses=*  
dbusername=qliksenserepository dbuserpassword=your_password  
dbsuperuserpassword=your_password
```

Details about the commands and their arguments can be found below.



Elevation will take place if run from an unelevated process and the UAC is on.

4. Validate that all services and nodes are operating as expected.

If you are upgrading, you also need to remove the old Qlik Sense Repository Database from the server once the upgrade and all validation are complete.



This step is required. Failing to remove the old service will lead to upgrade or patching issues.

Do the following:

1. Open a **Command Prompt** window and run:
c:\cd ProgramData\Package Cache
2. From there, do one of the following:
 - If you upgraded from 9.6, run:
c:\ProgramData\Package Cache\dir /s PostgreSQL.msi

2 Installing Qlik Sense Enterprise on Windows

The folder containing *PostgreSQL.msi* is displayed.

- If you upgraded from 12.5, run:
c:\ProgramData\Package Cache\dir /s PostgreSQL125.msi
The folder containing *PostgreSQL125.msi* is displayed.

3. Right-click the *PostgreSQL.msi* or *PostgreSQL125.msi* file and select **Uninstall** from the menu.
4. Reinstall the PostgreSQL binaries.



- If you don't reinstall the binaries, you'll encounter errors when running service configuration scripts.
- This step is optional if Qlik Sense is immediately upgraded after using Qlik PostgreSQL Installer, as the correct binaries will be automatically installed.

- a. Open a **Command Prompt** as administrator.
- b. Navigate to the location of the previously located .msi file.
Example: cd C:\ProgramData\Package Cache\{GUID}
- c. Run the following command replacing PostgreSQL125.msi with the appropriate .msi based on your previous version used:

- For Qlik Sense August 2023 and later:
msiexec.exe /qb /i "PostgreSQL14.msi" SKIPINSTALLDBSERVICE="1"
INSTALLDIR="C:\Program Files\Qlik\Sense"
- For Qlik Sense February 2022 to May 2023:
msiexec.exe /qb /i "PostgreSQL125.msi" SKIPINSTALLDBSERVICE="1"
INSTALLDIR="C:\Program Files\Qlik\Sense"



If you installed in a custom directory, adjust *INSTALLDIR* accordingly. For example, if installed in D:\Qlik\Sense, enter *INSTALLDIR="D:\QLIK\Sense"*.

This will reinstall the binaries without installing a database.

Syntax

Syntax

<pre>Qlik_PostgreSQL_setup.exe [-silent] {-log=path\filename} {acceptqca=1 0} {iprange=IpAddresses} {maxconnections=max connection number} {listenaddresses=ipv4/ipv6 addresses} {dbusername=[domain\username]} {dbuserpassword=password} {dbsuperuserpassword=password} {dbhost=hostname} {dbport=port} {dbinstallationpath=binary path} {dbdatapath=data folder} {qpiconfig="configfilepath"}</pre>	Fresh installation mode.
<pre>Qlik_PostgreSQL_setup.exe [-silent] {-log=path\filename} {acceptqca=1 0} {dbsuperuserpassword=password} {dbhost=hostname} {dbport=port} {dbinstallationpath=binary path} {qpiconfig="configfilepath"}</pre>	Upgrade mode.

2 Installing Qlik Sense Enterprise on Windows

```
Qlik_PostgreSQL_setup.exe -? or -h
```

Brings up the on-screen silent setup help.

Commands

Commands

Command	Type	Purpose
-silent	-	Command line-driven setup without UI (mandatory).
-log	[log file name with path]	Log file directory and log file name. Note: The user must have access to this directory

Arguments

Arguments are separated by a space and presented in the form [Argument]="[Value]". The double quotes can normally be omitted but may be needed, for example, when a path contains spaces.

The default values are the same as those used in the setup user interface.

Arguments

Argument	Values	Purpose	Scope
acceptQCA	1 0	Accepts the Qlik Customer Agreement (QCA). Note: This argument is mandatory when installing or upgrading, and you must accept the QCA to install successfully.	Install/Upgrade
iprange	[IP addresses]	IP addresses separated by comma. For all IPV4 and all IPV6 addresses, enter 0.0.0.0/0,:::0/0.	Install
maxconnections	[max connection number]	The maximum number of concurrent connections to the database.	Install
dbinstallationpath	[path to custom install directory]	Defines the directory if the default install directory will not be used (C:\Program Files\PostgreSQL\14).	Install/Upgrade

2 Installing Qlik Sense Enterprise on Windows

dbdatapath	[path to custom data directory]	Defines the directory if the default data directory will not be used (C:\Program Files\PostgreSQL\14\data).	Install/Upgrade
dbbackuppath	[path to custom backup directory]	Database backup folder path.	Upgrade
listenaddresses	[IP addresses/ranges]	Addresses that the database service should listen to. For example: 0.0.0.0/0, ::/0. To listen to all addresses, enter *.	Install
dbusername	[username]	Username of the repository database user.	Install
dbuserpassword	[password]	Password of the repository database user.	Install
dbsuperuserpassword	[password]	Password of the Postgres database super user.	Install/Upgrade
dbhost	[address of the database server]	Postgres database server address. If left empty, localhost is used.	Install/Upgrade
dbport	[database port]	Postgres database port.	Install/Upgrade
qpiconfig	[path to configuration file including the filename]	Path to a configuration file where settings for silent installation are stored.	Install/Upgrade

Examples of silent installation commands

An example of a silent installation command:

```
Qlik_PostgreSQL_setup.exe -silent acceptqca=1 iprange=0.0.0.0/0,::/0
dbport=4432 maxconnections=200 listenaddresses=*
dbusername=qlikenserepository dbuserpassword=your_password
dbsuperuserpassword=your_password
```

Another example of a silent installation command specifying a log file:

```
Qlik_PostgreSQL_setup.exe -silent -log="c:\temp\qpilog\qpiinstall.log"
acceptqca=1 dbport=4432 maxconnections=300 listenaddresses=*
dbusername=qlikenserepository dbuserpassword=your_password
dbsuperuserpassword=your_password
```

An example of a silent installation command that points to a configuration file containing the settings to be used in the installation:

2 Installing Qlik Sense Enterprise on Windows

```
Qlik_PostgreSQL_setup.exe -silent -log="c:\temp\qpilog\qpiinstall.log"  
acceptqca=1 qpiconfig="C:\temp\qpicfg\QpiConfig.cfg"
```

QPI configuration file

The Qlik PostgreSQL Installer configuration file is in XML format. The file must follow the structure provided in the example below. Substitute the placeholder values with your own.

If a value is left empty, the default value will be used. The default values are the same as those used in the setup user interface.

```
<?xml version="1.0"?>  
<CommandLineArguments xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
xmlns:xsd="http://www.w3.org/2001/XMLSchema">  
  <DbUserName>qlikenserepository</DbUserName>  
  <DbUserPassword>your_password</DbUserPassword>  
  <DbSuperUserPassword>your_password</DbSuperUserPassword>  
  <DbHost></DbHost>  
  <DbPort>4432</DbPort>  
  <ListenAddresses>*</ListenAddresses>  
  <IpRange>0.0.0.0/0,::/0</IpRange>  
  <MaxConnections>300</MaxConnections>  
  <DbInstallationPath></DbInstallationPath>  
  <DbDataPath></DbDataPath>  
  <DbBackupPath></DbBackupPath>  
</CommandLineArguments>
```

Configuration file syntax

Configuration file syntax

Setting	Description
DbUserName	Username of the repository database user. The default is <i>qlikenserepository</i> .
DbSuperUserPassword	Password of the Postgres database super user.
DbUserPassword	The password of the repository database user.
DbHost	Postgres database server address. If left empty, localhost is used.
DbPort	Postgres database port.
ListenAddresses	Addresses that the database service should listen to. For example: 0.0.0.0/0, ::/0. To listen to all addresses, enter *.
IpRange	IP addresses separated by comma. For all IPV4 and all IPV6 addresses, enter 0.0.0.0/0,::/0.
MaxConnections	The maximum number of concurrent connections to the database.
DbDataPath	Data path for the Postgres database. The default is <i>C:\Program Files\PostgreSQL\14\data</i> .
DbBackupPath	Backup path for the database upgrade. The default is <i>C:\ProgramData\Qlik\Sense\PostgreSQL\{dbversion}\backup</i> .
DbInstallationPath	Database installation folder path. The default is <i>C:\Program Files\PostgreSQL\14\</i> .

Troubleshooting

Troubleshooting typically starts by interpreting the installation log files. The log files are, by default, stored in the local temp directory of the current user.

2.7 Installing and configuring PostgreSQL on Azure

This topic describes how to install and configure PostgreSQL on Microsoft Azure.

For more general instructions regarding PostgreSQL installation and configuration, see *Installing and configuring PostgreSQL* (page 132).

Databases

You can improve performance in a Qlik Sense multi-node deployment by installing your repository (QSR), SenseServices, QSMQ, and Licenses databases on a dedicated, remote PostgreSQL server.



In Qlik Sense Enterprise, configuring all the components of a Multi-Cloud deployment is optional. However, all deployments, whether Multi-Cloud or on-premise, require the installation of the SenseServices database and QSMQ databases.

The Qlik Sense repository database (QSR)

The QSR is the primary database in your Qlik Sense deployment.

If you want to install the QSR database on a dedicated PostgreSQL server, you must install and configure PostgreSQL before you install Qlik Sense, as you will need to enter the PostgreSQL server/host details in the Qlik Sense installer.

The Qlik Sense services database (SenseServices)

The SenseServices database contains schemas for each of the Qlik Sense services and allows growth independently of the Qlik Sense Repository Database, while still sharing the same PostgreSQL instance and login role.

The Qlik Sense message queue database (QSMQ)

The QSMQ database provides a light-weight method of passing messages internally between services in Qlik Sense Enterprise. The NOTIFY and LISTEN functionality in PostgreSQL allows services to be notified about new messages that have been written to the messaging table.

The licenses service database (Licenses)

The licenses database contains a local copy of license data to allow faster response times and more robustness. It is only accessed by the licenses service.



The QSR, SenseServices, QSMQ, and Licenses databases share the same login role and must be installed on the same PostgreSQL instance.

2 Installing Qlik Sense Enterprise on Windows



If you already have a PostgreSQL database installed as part of a previous deployment, then you can continue to use it.



If Qlik Sense uses a PostgreSQL database on a dedicated infrastructure, then it can use PostgreSQL version 12.x. You can run the instance of PostgreSQL on platforms including Windows, Linux or cloud hosted services, such as Amazon RDS. However, Qlik will only offer configuration support when PostgreSQL is running on Windows. If you use Linux or Amazon RDS, it is your own responsibility to install and configure a running instance of PostgreSQL for Qlik Sense to use.

Setting up a PostgreSQL database in Azure

Before you install Qlik Sense, you need to set up a database in Azure.

Do the following:

1. Go to the Azure portal: <https://portal.azure.com>.
2. Search for **Azure Database for PostgreSQL**.
3. For the **PostgreSQL server** input fields, enter your values. The following three values must be filled in:
 - **Server name:** <your unique instance name, example: qliksensedemo>
 - **Server admin login name:** postgres
 - **Version:** 12
4. Under **Connection security**, click **Add client IP** to add the connection to the allowlist.
5. Disable SSL encryption.

Connecting to the database using pgadmin 4.x

Do the following:

1. If not already installed, download and install the *pgAdmin* tool from the following site: <https://www.pgadmin.org/download/pgadmin-4-windows/>.
2. Create a connection to the instance you checked out, in this case: *qliksensedemo.postgres.database.azure.com*.
3. Enter user: *postgres@qliksensedemo*
4. Enter the password that you used when setting up the database.



When installing the Sense database, you need to specify the user as *qliksenserepository@dbinstance*, while remaining as *qliksenserepository* in the Azure PostgreSQL instance.

5. Once connected to the Azure instance, open up a database and open the query tool.
6. In the **Query Editor**, add the following lines of code:
-- one by one, for creating the DB

```
CREATE DATABASE "QSR" ENCODING = 'UTF8';
```

2 Installing Qlik Sense Enterprise on Windows

```
CREATE DATABASE "SenseServices" ENCODING = 'UTF8';
CREATE DATABASE "QSMQ" ENCODING = 'UTF8';
CREATE DATABASE "Licenses" ENCODING = 'UTF8'; //one at a time

-- from here the whole script

CREATE ROLE "qliksenserepository" WITH LOGIN NOINHERIT NOSUPERUSER NOCREATEDB
NOCREATEROLE NOREPLICATION VALID UNTIL 'infinity'; -- change <qliksenserepository_user_
pass> to your password for the repository service user
ALTER ROLE "qliksenserepository" WITH ENCRYPTED PASSWORD '<qliksenserepository_user_
pass>';
GRANT qliksenserepository TO postgres;

ALTER DATABASE "QSR" OWNER TO "qliksenserepository";
ALTER DATABASE "SenseServices" OWNER TO "qliksenserepository";
ALTER DATABASE "QSMQ" OWNER TO "qliksenserepository";
ALTER DATABASE "Licenses" OWNER TO qliksenserepository;

GRANT TEMPORARY, CONNECT ON DATABASE "QSMQ" TO PUBLIC;
GRANT ALL ON DATABASE "QSMQ" TO postgres;
GRANT CREATE ON DATABASE "QSMQ" TO "qliksenserepository";
GRANT TEMPORARY, CONNECT ON DATABASE "SenseServices" TO PUBLIC;
GRANT ALL ON DATABASE "SenseServices" TO postgres;
GRANT CREATE ON DATABASE "SenseServices" TO "qliksenserepository";

GRANT TEMPORARY, CONNECT ON DATABASE "Licenses" TO PUBLIC;
GRANT ALL ON DATABASE "Licenses" TO postgres;
GRANT CREATE ON DATABASE "Licenses" TO qliksenserepository;
```

Installing Qlik Sense

Now that you have set up the PostgreSQL database on Azure, you can install Qlik Sense.

Do the following:

1. Follow the installation instructions in *Installing Qlik Sense Enterprise on Windows on a single node (page 98)*
2. The following values must be used on **Shared persistence database connection settings**:
 - **Database host name:** *qliksensedemo.postgres.database.azure.com*
 - **Database port:** *5432*
 - **Database user:** *qliksenserepository@qliksensedemo*

When you have installed Qlik Sense, your setup is complete.

2.8 Configuring a proxy for Qlik ADS and HDS communication with Qlik Sense Enterprise SaaS

You can handle the communication between Qlik Sense Enterprise on Windows and Qlik Sense Enterprise SaaS with a proxy.

2 Installing Qlik Sense Enterprise on Windows

In a Qlik Sense Enterprise on Windows multi-node deployment, the services App Distribution Service (ADS) and Hybrid Deployment Service (HDS) for distributing apps from Qlik Sense Enterprise on Windows to Qlik Sense Enterprise SaaS are installed on every node. You can manage the status of these services by starting and stopping the Qlik Sense Service Dispatcher, listed in the list of services running in the Windows machine.

With Qlik Sense May 2021 or later, you can configure the communication between Qlik Sense Enterprise on Windows and Qlik Sense Enterprise SaaS to be handled by a proxy.

In Qlik Sense Enterprise on Windows, configuration of a proxy for the ADS and HDS is done using command line parameters. Only HTTP schema is supported.

Do the following:

1. Stop the Qlik Sense Service Dispatcher, which handles the execution of the ADS and HDS.
2. Navigate to the *appsettings.json* files, which by default are located in:

%Program Files%\Qlik\Sense\AppDistributionService\appsettings.json

%Program Files%\Qlik\Sense\HybridDeploymentService\appsettings.json

3. Locate the following sections:
For ADS: QRS, Elastic, Engine, TempContent, HDS
For HDS: QRS, TokenRequest, Elastic
4. Add or edit as needed to have the following code:

```
"Proxy": {  
  "Server": "http://myproxy.example.com ",  
  "Port": 8888,  
  "UserId": "",  
  "EncryptedPassword": ""  
}
```

Where *http://myproxy.example.com* is the address of your company's proxy, and 8888 is the port used by the proxy.



You can specify an IP address rather than a domain name as the proxy URI, for example, *http://10.76.124.124*.

- An admin is required to enter "EncryptedPassword" by encrypting the password string using *C:\Program Files\Qlik\Sense\Licenses\Encrypt-Password.ps1* on the machine where Qlik Sense is installed.
 - If an admin wants to enter a plain text password, they must change the key from "EncryptedPassword" to "Password".
5. Save and close the *appsettings.json* file.
 6. Restart the Qlik Sense Service Dispatcher.
 7. If you have a multi-node installation, repeat these steps for all the nodes in your installation.

2.9 Configuring a proxy for Qlik License Service communication in Qlik Sense Enterprise on Windows

You can handle the communication between the Qlik License Service and the License Back-end with a proxy.

The Qlik License Service is included in Qlik Sense Enterprise February 2019 and later releases and is used when Qlik Sense is activated using a signed key license. The Qlik License Service stores the information about the license, and communicates with a License Back-end Service, hosted by Qlik, for product activations and entitlement management. Port 443 is used for accessing the License Back-end Service and retrieving license information.

In a Qlik Sense Enterprise on Windows multi-node deployment, the Qlik License Service is installed on every node. You can manage the status of the Qlik License Service by starting and stopping the Qlik Sense Service Dispatcher, listed in the list of services running in the Windows machine.

With Qlik Sense June 2019 or later you can configure the communication between Qlik License service and the Qlik License Back-end to be handled by a proxy.

In Qlik Sense Enterprise on Windows, configuration of a proxy for the Qlik License Service is done using command line parameters. Both HTTP and HTTPS schema are supported.

With Qlik Sense June 2020 or later NTLM and basic authentication capabilities to the licenses service when communicating over a HTTP tunnel are available. This allows you to require authentication on tunneling proxies to configure a more secure environment.

Do the following:

1. Stop the Qlik Sense Service Dispatcher, which handles the execution of the Qlik License Service.
2. Navigate to the *services.conf* file, which by default is located in:

```
%Program Files%\Qlik\Sense\ServiceDispatcher\services.conf
```

3. Locate the section [licenses.parameters], which by default contains the following lines:

```
[licenses.parameters]
-qsefw-mode
-app-settings="..\Licenses\appsettings.json"
```

4. Add the line `-proxy-uri=http://myproxy.example.com:8888` as shown below:

```
[licenses.parameters]
-qsefw-mode
-proxy-uri=http://myproxy.example.com:8888
-app-settings="..\Licenses\appsettings.json"
```

Where "http://myproxy.example.com" is the address of your company's proxy, and "8888" is the port used by the proxy.



You can specify an IP address rather than a domain name as the proxy URI, for example - `proxy-uri=http://10.76.124.124:1337`.

2 Installing Qlik Sense Enterprise on Windows

5. If your external proxy requires an encrypted password to be applied, browse to `%ProgramFiles%\Qlik\Sense\Licenses` and run `Encrypt-Password.ps1` [password for proxy access].

Example:

```
Encrypt-Password.ps1 123456
```

Copy the generated encrypted password and use it in the next step.

6. To require authentication on tunneling proxies add the following lines to the `services.conf` file:
 - If encryption is required for your proxy password:
 - proxy-uri=[the uri of the proxy]
 - proxy-auth-mode=ntlm|basic|(leave empty for no authentication)
 - proxy-user=[username without domain]
 - proxy-encrypted-password=[password]
 - proxy-domain=[the domain] (only for NTLM)
 - If you are not using encryption:
 - proxy-uri=[the uri of the proxy]
 - proxy-auth-mode=ntlm|basic|(leave empty for no authentication)
 - proxy-user=[username without domain]
 - proxy-password=[password]
 - proxy-domain=[the domain] (only for NTLM)
7. Save and close the `services.conf` file.
8. Restart the Qlik Sense Service Dispatcher.
9. If you have a multi-node installation, repeat these steps for all the nodes in your installation.



After a version upgrade, you may need to add the settings back to the `services.conf` file, see Error message "No access path" after upgrade (page 295).

2.10 Configuring preferred cipher suites for Qlik License Service in Qlik Sense Enterprise on Windows

You can rank the preferred cipher suites that Qlik License Service uses to encrypt and decrypt the signed key license.

The Qlik License Service is included in Qlik Sense Enterprise on Windows February 2020 and in later releases.

The Qlik License Service uses Mutual TLS Authentication (mTLS) to ensure requests coming from both the server and client are trusted. The Qlik License Service listens on port 9200.



TLS 1.2 is supported since June 2017.

The following list shows the supported cipher suites:

- `TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256`
- `TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384`
- `TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256`

2 Installing Qlik Sense Enterprise on Windows

- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384
- TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305
- TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305
- TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA
- TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA
- TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA
- TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA
- TLS_RSA_WITH_AES_128_GCM_SHA256
- TLS_RSA_WITH_AES_256_GCM_SHA384
- TLS_RSA_WITH_AES_128_CBC_SHA
- TLS_RSA_WITH_AES_256_CBC_SHA
- TLS_ECDHE_RSA_WITH_3DES_EDE_CBC_SHA
- TLS_RSA_WITH_3DES_EDE_CBC_SHA

To configure the preferred cipher suites for the Qlik License Service, do the following:

1. Open the *service.conf* file.
The default path is `%Program Files%\Qlik\Sense\ServiceDispatcher\service.conf`.
2. Go to the following section:
`[licenses.parameters]`
`-qsefw-mode`
`-app-settings="..\Licenses\appsettings.json"`
3. Add a comma-separated list of ciphers to this section, as shown below:
`[licenses.parameters]`
`-qsefw-mode`
`-cipher-suites=TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256,TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA`
`-app-settings="..\Licenses\appsettings.json"`
4. Save the file and close.
5. Restart the Qlik Sense Service Dispatcher, which handles execution of the Qlik License Service.
6. If you have a multi-node environment, repeat the steps above for each node.

2.11 Changing the user account to run Qlik Sense services

Before you install, change or upgrade your Qlik Sense installation, you must choose or create an administrator or non-administrator account to run the Qlik Sense services. For example, your company policy may require you to run the Qlik Sense services as a user without administrator privileges.



If you want to upgrade from Qlik Sense 3.1 SR2 or later to Qlik Sense June 2017 you must use a service user account (local or domain) and not a Local System account to run the services. If you use a Local System account to upgrade, you will get an error. See: [Upgrading Qlik Sense Enterprise on Windows \(page 158\)](#).

Using an account without administrator privileges to run the Qlik Sense services during the installation of a node

To install a rim node in this way you need to run an additional bootstrap command from an elevated command prompt to register the rim node on the central node.



If you are installing a central node you can follow the same procedure as a regular administrator installation.

To install a node:

1. Log in to the computer where you plan to install Qlik Sense as an administrator.
See: *User accounts (page 63)*
2. Download the *Qlik_Sense_setup.exe* file.
See: *Downloading installation files (page 17)*
3. On the **Create or join a cluster** screen, select **Join cluster**.
4. On the **Shared persistence database connections settings** screen, ensure that you specify the correct hostname and password to the repository database that you want to connect to.
See: *Installing Qlik Sense in a multi-node site (page 108)*
5. On the **Service Credentials** screen, enter your non-administrator user account, user name, and password. For example, enter your user name as follows: `.\senseserviceuser` or `domain\senseserviceuser`.



If you enter a username that is more than 20 characters long, it must be in UPN format, and must include the full domain name. For example, `longusername@full.domain.name`.

On the final screen of the installation program, you do not have the option to start the Qlik Sense services, instead the following message is displayed: **The service user does not have administrator privileges. See the documentation for more information.**

Next, run the bootstrap command in an elevated command prompt while registering the rim node with a certificate.

To run the bootstrap command:

1. On the rim node, open an elevated command prompt window. The bootstrap command elevates your rights enabling you to perform tasks that require an administrator, such as installing certificates and adding performance counters.
2. In the command prompt, navigate to the installed location: `Program Files\Qlik\Sense\Repository` and run the `Repository.exe -bootstrap` command. The Qlik Sense Service Dispatcher must be running before the `Repository.exe -bootstrap` is executed. While the bootstrap is running, in the QMC on the central node, register the rim node with a certificate that is generated.
3. On the central node, register the rim node in the QMC, see: *Connecting and configuring the nodes (page 122)*. After you have registered the rim node the bootstrap process will terminate.

4. Exit the command prompt.
5. In Windows, **Services**, start all Qlik Sense services. You must start the Qlik Sense Service Dispatcher (QSD) before starting the Qlik Sense Repository Service (QRS).

Changing the user account type to run the Qlik Sense services on an existing site

Follow the instructions in this section if you used an administrator user account when installing Qlik Sense, and later wish to change to use an account without administrator privileges to run the Qlik Sense services.

Do the following:

1. In Windows, either create a new or use an existing domain or local user account to run the Qlik Sense services.
2. If the service account user does not have administrator privileges, you must add the user to the following groups in **Computer Management > System Tools > Local Users and Groups > Groups**.
 - Qlik Sense Service Users
 - Performance Monitor Users

The service account user also needs access to shared folders.

3. Open the **Control Panel** and then select **System and Security > Administrative Tools > Services**.
4. Stop all services except the **Repository Database**.
5. Assign **Full control** permission for the dedicated service account to the folder `%ProgramData%\Qlik\Sense`.
6. As an administrator, open an elevated command prompt.
7. Navigate to the `Program Files\Qlik\Sense\Proxy` folder and run `Proxy.exe -bootstrap`.
8. Navigate to the `Program Files\Qlik\Sense\Scheduler` folder and run `Scheduler.exe -bootstrap`.
9. Navigate to the `Program Files\Qlik\Sense\Repository` folder and run `Repository.exe -bootstrap`. If you are changing the user account on your primary or central node, run `Repository.exe -bootstrap -iscentral`. The Qlik Sense Service Dispatcher must be running before the `Repository.exe -bootstrap` is executed.
10. Close the elevated command prompt.
11. Change the log on credentials for each of the Qlik Sense services as follows:
 - a. Right-click the service and select **Properties**.
 - b. Select the **Log On** tab and then **This account**.
 - c. Enter the credentials for the dedicated service account and click **OK**.



If you are using a user account with administrative privileges, keep the Qlik Sense Repository Database running under the Local System account. Do not change the account.



Depending on your setup some of the services may not be available.

2 Installing Qlik Sense Enterprise on Windows

12. Start the Qlik Sense Service Dispatcher, and then the Qlik Sense Repository Service (QRS).
13. Start the rest of the Qlik Sense services.

For additional support on changing the Qlik Sense Service account or troubleshooting installation issues, see [🔗 Changing the Qlik Sense Service Account and what to consider](#) and [🔗 Interactive Logon Rights for Qlik Sense installation](#).

Changing the Qlik Sense services account password

In some situations, you may be required to change your service account password that is used to run your Qlik Sense services. For example, if the company password policy requires that you change the password at a set interval, or you want to change your password for security reasons. To prevent issues with the QMC, you must change the service account password and update the monitoring data connections. First, change your service account password.

Do the following:

1. Change your service account password.
2. Open the Windows Services app.
3. For each Qlik service, right-click, then select **Properties**.
4. On the **Log On** tab, update the password.
5. Once the password is updated on each service, restart the service.

Next, update the monitoring data connections in the QMC.

Do the following:

1. Open the QMC.
2. Click **Data connections**.
3. For each **monitoring_apps_*** data connection, click **Edit**, then update the password.

Your Qlik Sense services account password is now updated.

2.12 Uninstalling Qlik Sense Enterprise on Windows

Before uninstalling Qlik Sense, consider the following information:

- Uninstalling Qlik Sense removes the Qlik License from the local PostgreSQL database, but it does not remove it from the server.
- To completely remove all files from the server during the uninstallation, check **Remove Qlik Sense demo apps, certificates and data folders**.
- If any updates have been applied to Qlik Sense since the initial installation, uninstalling Qlik Sense will also remove the updates.
- In a multi-node site, the rim nodes are dependent on the central node. Uninstalling the central node will cause rim nodes to fail.
- When using the *Qlik_Sense_setup.exe* file to uninstall Qlik Sense, it must be the same version that you used to install Qlik Sense.

2 Installing Qlik Sense Enterprise on Windows

Do the following:

1. To uninstall Qlik Sense, open the **Windows Control Panel** and select **Uninstall a program**. Select **Qlik Sense** from the list of programs, then click **Uninstall**.

A confirmation screen is displayed asking if you are sure that you want to uninstall Qlik Sense from your server.

2. To remove all files from the current server, select **Remove Qlik Sense demo apps, certificates and data folders**.



*If you plan to reinstall Qlik Sense on the same server, leave the check box unselected.
Restoring a Qlik Sense site (page 203) Restoring a Qlik Sense site (page 203)*

3. Click **Uninstall**.
 - If User Account Control (UAC) is disabled, the uninstallation starts.
 - If UAC is enabled, the **User Account Control** dialog is displayed. Click **Yes** to start.

When finished, the uninstall dialog confirms that Qlik Sense has been uninstalled successfully.

4. Click **Finish**.

You have now uninstalled Qlik Sense.

2.13 Integrating Qlik Catalog with Qlik Sense Enterprise

From February 2021 you can find the documentation to configure the integration of Qlik Catalog with Qlik Sense Enterprise in the *Integration Guide* on the [Catalog Installation Guides](#) page.

3 Upgrading Qlik Sense Enterprise on Windows

Upgrading your Qlik Sense Enterprise on Windows deployment can be done by running the Qlik Sense installer application. Upgrading replaces your current version of Qlik Sense Enterprise on Windows with a newer version. For any type of deployment, a successful upgrade requires a bit of planning. This guide will help you plan and run your upgrade, and configure your deployment when ready. At the end of this section, you will find a troubleshooting section if you have any problems during the upgrade.

3.1 Patching instead of upgrading

A patch applies a software update or a software fix to a current version, without upgrading the entire deployment.

If you want to patch your current version, see *Patching Qlik Sense (page 166)*.

3.2 Repairing instead of upgrading

A repair checks your current deployment for missing files, shortcuts, or registry values, and repairs them without changing your current version.

If you want to repair your current version, see *Repairing an installation (page 167)*.

3.3 Planning your upgrade

A successful upgrade requires some planning. Before you can upgrade, you must know your upgrade path, that is, you need to know which version you currently have and which version you want to upgrade to. You also need to collect other information about your deployment, the details of which are covered in this guide.



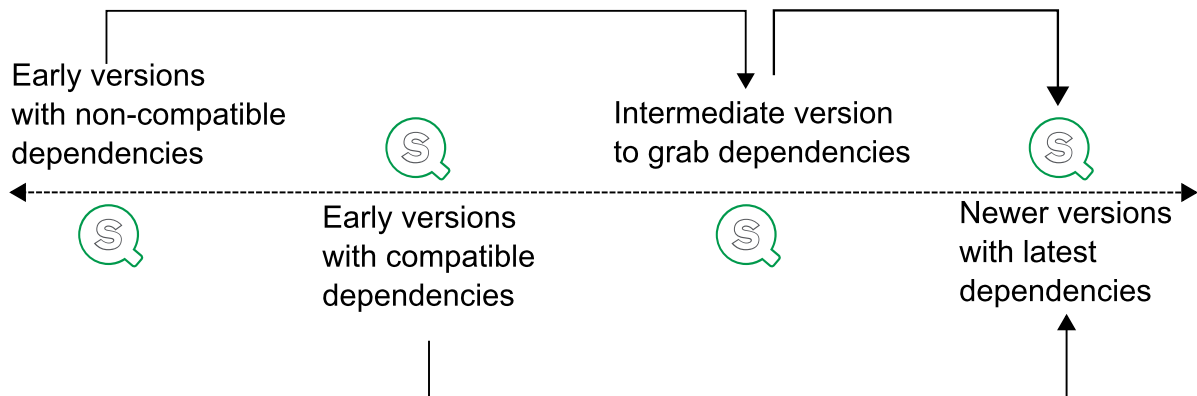
*Do not uninstall your current version of Qlik Sense Enterprise. If you have already uninstalled Qlik Sense, see *Upgrading after uninstalling Qlik Sense (page 165)*.*

Planning your upgrade path

Not all versions have a direct upgrade path to the latest or newer version of Qlik Sense Enterprise. It is good practice to use the release notes as a guide to determine when major changes between versions are introduced. You might be required to upgrade to an intermediate version to pick up dependencies before upgrading to your desired version of Qlik Sense Enterprise.

Qlik Sense Enterprise upgrade path when skipping versions

Qlik Sense upgrade path



Considerations about older versions

Some Qlik Sense Enterprise versions introduced significant changes. The table below lists major changes you should consider when upgrading. This list will help you decide which intermediate version you might need to upgrade to before upgrading to later versions.



Example: Qlik Sense Enterprise February 2021 supports only .NET 4.8. This means that to upgrade from older versions, you should first upgrade to the version before the change, in this case Qlik Sense Enterprise November 2020, then upgrade to the version with the major change. Doing this will reduce the likelihood of encountering errors during the upgrade.

Versions	Considerations
Qlik Sense 3.1 SR1 and earlier	Must upgrade to June 2017 before upgrading to later versions.
Qlik Sense June 2017 and later	Support only the shared persistence model.
Qlik Sense September 2017 and later	Does not support configuring centralized logging through the installer.
Qlik Sense November 2017 and later	Does not support soft delete records.

3 Upgrading Qlik Sense Enterprise on Windows

Versions	Considerations
later	
Qlik Sense February 2020 and later	Changed NodeJS version. Must re-create certificates generated with versions June 2019 and earlier.
Qlik Sense February 2021 and later	Support only .NET framework 4.8 or higher. Must upgrade to November 2020 first. This version also requires Visual C++ 2019 distributable package.
Qlik Sense February 2022 and later	Does not support centralized logging.
Qlik Sense August 2022 and later	<p>PostgreSQL 9.6 is not supported when upgrading. However, if you have PostgreSQL 9.6 installed locally when you upgrade Qlik Sense, it will still be used after the upgrade, even if it is no longer supported.</p> <p>The easiest way to upgrade to PostgreSQL 12.5 is to use the Qlik PostgreSQL Installer, which upgrades the database from 9.6 to 12.5 – if the database has been installed by Qlik Sense. Other versions of PostgreSQL need to be installed manually.</p> <p>If you have PostgreSQL 12.5 installed locally when you upgrade Qlik Sense, the database will still be used after an upgrade.</p> <p>For other versions of the database, the installed version will continue to be used after an upgrade.</p>
Qlik Sense August 2023 and later	<p>PostgreSQL 9.6 is not supported when upgrading. If you have PostgreSQL 9.6 installed locally when you upgrade Qlik Sense, you will not be able to proceed with the upgrade until you have upgraded the PostgreSQL database to a newer version.</p> <p>If you have PostgreSQL 12.5 installed locally when you upgrade Qlik Sense, a warning will be displayed but you can still upgrade. The existing the PostgreSQL database will still be used after the upgrade.</p> <p>The easiest way to upgrade to PostgreSQL 14.8 is to use the Qlik PostgreSQL Installer, which upgrades the database from 9.6 or 12.5 to 14.8 – if the database has been installed by Qlik Sense. Other versions of PostgreSQL need to be installed manually.</p> <p>If you have PostgreSQL 14.8 installed locally when you upgrade Qlik Sense, the database will still be used after an upgrade.</p> <p>For other versions of the database, the installed version will continue to be used after an upgrade.</p>

Considerations about multi-node deployments

A single node site is simpler to upgrade than a multi-node site. If you have more than one node in a site, you must upgrade each node separately.

- Each node in a multi-node site must run the same version of Qlik Sense.
- You should upgrade the central node first.
- All nodes should be offline when you upgrade.
- Use the same login account for the upgrade as you did with the original installation. If you use a different login account, the node will not find the certificates on the node.

Considerations about logging

As of Qlik Sense February 2022, centralized logging is no longer supported. See *Logging (page 251)* for general information and log locations.

Considerations about custom configurations

If your current deployment includes custom configuration files, they are retained during the upgrade process. The option **Keep current settings** is selected by default in the **Ready to upgrade** dialog. Clear the selection if you don't want to retain the custom configuration files.

The following files are by default retained during an upgrade:

- `%ProgramFiles%\Qlik\Sense\Repository\Repository.exe.config`
- `%ProgramFiles%\Qlik\Sense\Proxy\Proxy.exe.config`
- `%ProgramFiles%\Qlik\Sense\Scheduler\Scheduler.exe.config`
- `%ProgramFiles%\Qlik\Sense\ServiceDispatcher\services.conf`

Consideration about your Qlik Sense Repository Database

In earlier versions of Qlik Sense, the Qlik Sense Repository Database used PostgreSQL version 9.6, which no longer is supported. Between Qlik Sense Enterprise on Windows May 2021 and Qlik Sense Enterprise on Windows May 2023, the Qlik Sense Repository Database used PostgreSQL version 12.5. PostgreSQL was only deployed during fresh installations of these versions.

Starting at Qlik Sense Enterprise on Windows August 2023, the bundled PostgreSQL has been upgraded to version 14.8. However, PostgreSQL 14.8 will only be deployed during a fresh installation Qlik Sense Enterprise on Windows August 2023 and above.

If you are upgrading Qlik Sense Enterprise on Windows from an earlier version that uses PostgreSQL version 9.6, the version is no longer supported from August 2023 and you will be required to upgrade your PostgreSQL database first. If you have PostgreSQL 12.5 installed locally when you upgrade Qlik Sense, a warning will be displayed but you can still upgrade. The existing the PostgreSQL database will still be used after the upgrade.

The easiest way to upgrade to PostgreSQL 14.8 is to use the Qlik PostgreSQL Installer, which upgrades the database from 9.6 or 12.5 to 14.8 – if the database has been installed by Qlik Sense. For more information, see *Installing or upgrading PostgreSQL using the Qlik PostgreSQL Installer (page 139)*.

3 Upgrading Qlik Sense Enterprise on Windows

If you have PostgreSQL 14.8 installed locally when you upgrade Qlik Sense, the database will still be used after an upgrade. For other versions of the database, the installed version will continue to be used after an upgrade. All your data and settings are migrated to the new version. However, if you have custom configurations for your PostgreSQL installation they must be recreated after the upgrade.



If you want to deploy a PostgreSQL database on a dedicated infrastructure, then you can use any supported version of PostgreSQL. Check the [System requirements for Qlik Sense Enterprise on Windows](#) to know which versions of PostgreSQL are supported. You can run the instance of PostgreSQL on platforms including Windows, Linux or cloud hosted services, such as Amazon RDS. However, Qlik will only offer configuration support when PostgreSQL is running on Windows. If you use Linux or Amazon RDS, it is your own responsibility to install and configure a running instance of PostgreSQL for Qlik Sense.

Here are some other important things to note regarding your Qlik Sense Repository Database upgrade:

- The version of PostgreSQL that is included in the Qlik Sense June 2017 or later does not include pgAdmin tools. For information about manually installing the PostgreSQL database, see *Installing and configuring PostgreSQL* (page 132).
- The Qlik Sense installer cannot use SSL encryption for establishing a connection to PostgreSQL. When SSL encryption is enabled, the installer does not recognize any already installed PostgreSQL databases, as a result, the installation cannot be completed. You should temporarily disable SSL during installation or upgrade.



*If you have uninstalled Qlik Sense before upgrading, but you maintained your PostgreSQL database, you must create a database dump file and restore the PostgreSQL database manually. You also need to manually reconfigure any custom parameters. If you have already uninstalled Qlik Sense, see *Upgrading after uninstalling Qlik Sense* (page 165).*

3.4 Running the upgrade application

To upgrade your deployment, you run the Qlik Sense installer on each node in your Qlik Sense deployment, starting with the central node.

Before you start the upgrade

You need to download the installer, verify the system requirements, and ensure your system is ready to upgrade.

1. Check your [system requirements](#).
2. [Download](#) your installer and the release notes and save them locally on the node you are upgrading.
3. Log into the central node server as an admin user with a password.



The admin user password cannot be blank. If the admin user does not have a password, create one before starting the upgrade.

4. Make sure that the database service is running.
5. Verify that your Qlik Sense services are running with a service account, not a local account. To change the service account user, see *Changing the user account to run Qlik Sense services* (page 153).
6. Get your Qlik Sense Repository Database superuser password, as this is needed during the upgrade. You created this password when you first installed Qlik Sense.
7. Create a [backup](#) of the Qlik Sense deployment before upgrading.
8. You can optionally remove the root certificate from the central node and all certificates from the rim nodes if you want the QPS to recreate them during the upgrade.



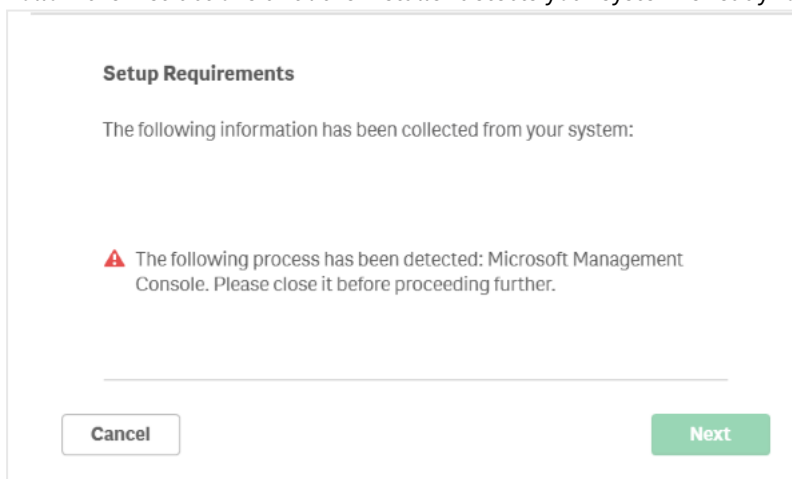
If you have any problems during or after the upgrade, refer to the *Troubleshooting your upgrade* (page 168) topic to find solutions to common upgrade problems.

Upgrading a Qlik Sense node

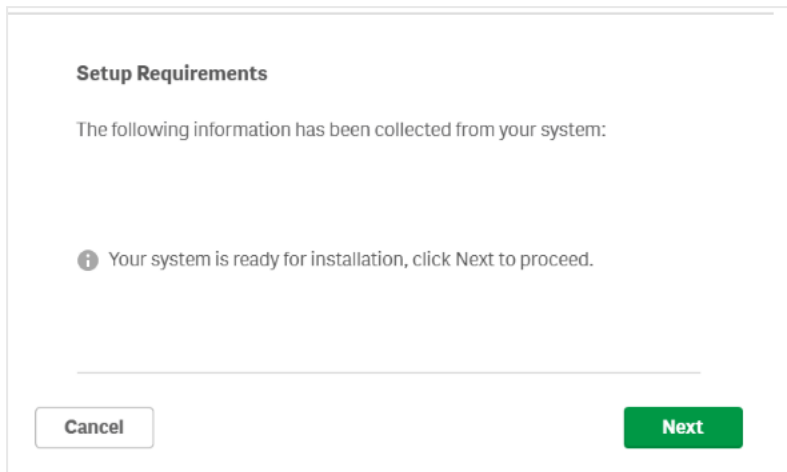
For a multi-node upgrade, start with the central node. When finished, repeat these steps for each node in your site.


Do the following:

1. From the Windows services app, stop the Qlik Sense services.
2. Right-click the *Qlik_Sense_setup.exe* file that you saved locally and click **Run as administrator**. The installer checks to see if any applications that are running need to be closed before starting. Follow the instructions until the installer detects your system is ready for upgrading.



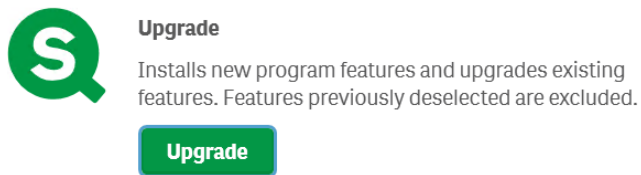
3 Upgrading Qlik Sense Enterprise on Windows




 *The installer also checks your root certificate. If it is unsupported, you must remove it before upgrading. Click the **Remove** button to remove unsupported certificates.*

3. Select **Upgrade**.

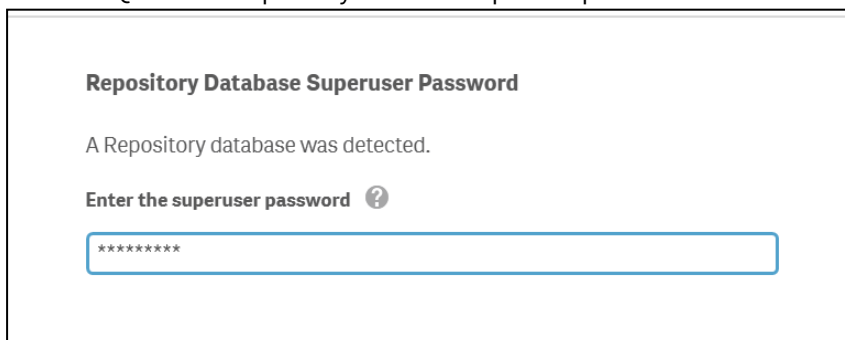
This will install Qlik Sense May 2021 on your computer. An earlier version has been detected and it will be removed during the upgrade process.



4. On the **License Agreement** window, read and accept the Qlik User License Agreement.
5. On the **Service Credentials** page, enter the username and password for the Qlik Sense service account.

 *If the username is a member of a domain, enter the user name as <domain>/<username>.*

6. Enter the Qlik Sense Repository Database superuser password.



7. On the **Ready to upgrade** window, select the installer options.

- In the **Ready to upgrade** section, select the relevant options.
- In the **Supported object bundles** section, optionally install the object bundles. Then, select which object bundles you want to install from the list of those available for your Qlik Sense Enterprise on Windows installation.



If you are installing object bundles, read and accept the object bundles license agreement.

- In the **Help Us Improve** section, select if you want to anonymously share system data with Qlik.
 - Click **Next** when you have selected your options, then click **Upgrade**.
8. Click **Finish**.

3.5 Configuring your node after upgrading

After you successfully upgrade your central node, you can configure some of the things that might have changed during the upgrade. If you have issues with your upgrade, see the *Troubleshooting your upgrade (page 168)* topic to find a solution to your problem.

1. Check that your Qlik Sense services are running.
[Checking the status of Qlik Sense services](#)
2. Log into the QMC. On the **Apps** overview page, check if any apps need to be migrated.
[Apps](#)
3. Log into one of the rim nodes and complete the [upgrade steps](#) as you did on the central node.
[Nodes](#)



If the installer detects unsupported certificates on a non-central node, you must remove all of them before upgrading.

3.6 Upgrading after uninstalling Qlik Sense

If you have already uninstalled Qlik Sense, then you have to install a new version rather than upgrade an existing version. During an uninstallation, the PostgreSQL database is not removed by the installer application and it can be restored. These instructions will help you restore the database before installing a new version of Qlik Sense.

Do the following:

1. Navigate to `%ProgramData%\Qlik\Sense\Repository\PostgreSQL`.
2. Copy the PostgreSQL folder to a temporary location outside of the `%ProgramData%` folder.
3. Download and install a compatible version of PostgreSQL from the [PostgreSQL](#) website.
For more information, see *Installing and configuring PostgreSQL (page 132)*.
To check the system requirements, see *System requirements for Qlik Sense Enterprise (page 18)*.

4. Open a Command Prompt as a non-admin user.



The pg_ctl.exe command in the next step should not be run as an administrator.

5. Navigate to `%ProgramFiles%\PostgreSQL\<database version>\bin`, and run the following commands:

```
pg_ctl.exe start -w -D "C:\SenseDB\<PostgreSQL-version>"
set PGUSER=postgres
set PGUSER=postgres
set PGPASSWORD=password
pg_dumpall.exe > [<path to dump file>]
pg_ctl.exe stop -w -D "C:\SenseDB\<PostgreSQL-version>"
```
6. In the **Command Prompt** window, navigate to the folder containing the `Qlik_Sense_setup.exe` file.
7. Run the following command to install Qlik Sense and restore your Qlik Sense Repository Database.
`Qlik_Sense_setup.exe databasedumpfile=<path_to_dump_file>`



The path to the dump file must be entered as an absolute path. If you enter a relative path, the install will fail.

8. Now you can install a new version of Qlik Sense.
For more information, see *Installing Qlik Sense Enterprise on Windows on a single node (page 98)*.

3.7 Patching Qlik Sense

You can update your Qlik Sense deployment when a patch of the software is available for installation. A patch primarily includes software updates and fixes that are applied to the existing Qlik Sense version.



Patches are installed without the need to remove earlier updates or the major release. Qlik Sense patches are cumulative. By installing the latest patch, updates and fixes introduced in previous patches are also installed.

When you uninstall a patch, the individual updates from the installed version of Qlik Sense are removed.

In a multi-node site, all nodes must run the same version of Qlik Sense. We recommend installing patches with all nodes offline, and starting with the central node.

Before you install a patch Qlik Sense, do the following:

- Review *System requirements for Qlik Sense Enterprise (page 18)*.
- Download the `Qlik_Sense_update.exe` file.
- Make sure you have logged on with Administrator rights using an account that has an actual password defined, that is, not a blank password.
- Create a backup of your Qlik Sense deployment. If Qlik Sense is installed on a Virtual Machine (VM) it may be sufficient to take a snapshot of the machine before upgrading. For more information, see *Backing up a Qlik Sense site (page 201)*.



When updating a rim node, ensure that you use the same log-in account as was used for the initial installation of that node. Failure to do so means that the central node will not find the certificates installed on the node and you will need to perform a clean installation of the node.

Do the following:

1. Stop the Qlik Sense services on all nodes.
2. Run the setup to install a patch on the central node.
When the installation is complete, the **Summary** is displayed.
3. Click **Finish** to close the **Summary**.



If the patch did not install successfully, the **Failed** screen is displayed. For more detailed information, see the installation log located in your **temp** folder accessed with environment variable %temp%.

You have successfully applied a patch to your Qlik Sense deployment.

4. Start the Qlik Sense services.
5. Repeat this procedure for each of the remaining nodes.



You cannot repair an installation using the repair option on the setup program once patches have been applied. The repair option is only available for the original software version, so any patches installed must be uninstalled before you can use the repair option.



Follow the same procedure to uninstall patches.

3.8 Repairing an installation

The **Repair** option restores all missing files, shortcuts and registry values without any credentials being changed.



If patches have been applied to Qlik Sense, the Repair option is disabled. You must uninstall all patches before you can use the **Repair** option, as it will restore the installation to the original installed version.

Do the following:

1. To start repairing the installation, open the **Control Panel** and select **Uninstall a program**. Then select **Qlik Sense** from the list of programs and click **Change**.

The **Qlik Sense Setup maintenance** screen is displayed.



You can also perform this action by double-clicking the Qlik_Sense_setup.exe file. In that case, you must use the correct version of the setup file when repairing your Qlik Sense installation, that is, the same version used when installing Qlik Sense.

2. Click **Repair**.
The **Ready to repair** screen is displayed.
3. Click **Repair**.
 - If UAC is enabled, the **User Account Control** screen is displayed.
 - If UAC is disabled, the repair process starts.
4. Click **Yes** to start repairing your Qlik Sense installation.



This is only applicable if UAC is enabled.

The progress is displayed.

When finished, click **Repair Summary** to confirm that Qlik Sense has been restored successfully. Click **Back**.

5. Click **Finish**.

You have now successfully repaired your Qlik Sense installation.

3.9 Troubleshooting your upgrade

This section contains links to common problems and solutions based on real user support cases reported in the Qlik Community. When you encounter a problem during a upgrade, there are several steps to take to begin troubleshooting the issue.

Start with the [troubleshooting overview](#) section to learn how to investigate problems and find errors in the log files. Here, you'll also find troubleshooting topics that may address your upgrade issue.

Failed upgrade or patch

If the upgrade fails before finishing, there are several possible causes. The most common reason is that a required component is missing. Depending on which version you are upgrading to, you may need to resolve the missing component before continuing.

Common error messages that indicate that you are missing a component:

- This version of Qlik Sense requires a 'SenseServices' database for multi cloud capabilities...
- This version of Qlik Sense requires a 'QSMQ' database for multi cloud capabilities...
- This version of Qlik Sense requires a 'License' database for license capabilities...

For details on how to resolve the issue, see the links below.

3 Upgrading Qlik Sense Enterprise on Windows

- [Repair the missing database](#)
- [Fix the QMSQ database ownership information](#)
- [Re-create all of the databases](#)

Common error code when an upgrade of patch fails:

- [Patch was uninstalled with exit code: -1](#)

Connecting to QMC and the Hub

After you complete the upgrade, you might be unable to connect to the Management Console or the Hub. This is a common problem for users who upgrade from older versions to Qlik Sense November 2017 or later. The most common reason for this issue is that the Qlik Sense Repository Service does not delete soft records in Qlik Sense November 2017 or later. This results in broken references from the repository service. For details on how to resolve the issue, see the link below.

- [Can't access Hub or Management Console after upgrade](#)

Repository service and PostgreSQL database

After completing your upgrade, the repository service might fail to start. The most common reason for this issue is that the repository service is passing incorrect information. For details on how to resolve the issue, see the links below.

- [Fatal exception password authentication failed for user "postgres"](#)
- [SeSecurityPrivilege error](#)
- [How to create a PostgreSQL database dump file after uninstalling Qlik Sense](#)

Windows user accounts and service credentials

After completing the upgrade, the Qlik Sense services may fail to start. This could be caused by Microsoft Windows updates, authentication problems, or service timeout. For details on how to resolve the issue, see the links below.

- [Error 1053: The service did not respond to the start control request in a timely fashion when trying to start the services](#)
- [How to change the Qlik Sense Service Account and what to consider](#)
- [Manual stop-start order for Qlik Sense services](#)

Certificates and licenses

Certificates let the services and nodes in your Qlik Sense deployment communicate with each other. After an upgrade, when there are issues with individual nodes, the cause is usually problems with the certificates. For details on how to resolve the issue, see the links below.

- [Rim node remains offline after upgrade because of certificates](#)
- [Rim nod remains offline after upgrade because can't connect to license service](#)
- [How to change the Qlik Sense Proxy certificate if the service account does not have local administrative permissions](#)

Antivirus software

Qlik Sense requires access to certain ports. An upgrade may fail or your deployment may have issues due to anti-virus software preventing Qlik Sense access to required ports. For details on how to resolve the issue, see the link below.

- [Exclude Qlik Sense folders from anti-virus scanning](#)

4 Running the installer silently

All install, upgrade, repair, and patching options that are available in the user interface of the installer can be performed with silent operations.

- *Silent installing (page 171)*
- *Silent upgrading (page 176)*
- *Silent repairing (page 179)*
- *Silent patching (page 179)*

4.1 Silent installing

When running a silent installation, Qlik Sense is installed with no dialogs at all. This means all features, properties and user selections have to be known before performing a silent installation. All setup options that are available in the user interface of the installer can be performed with silent operations.

Do the following:

1. Select **Start > All Programs > Accessories > Command Prompt**.
The **Command Prompt** window is displayed.
2. In the **Command Prompt** window, navigate to the folder containing the *Qlik_Sense_setup.exe* file.
3. Enter *Qlik_Sense_setup.exe* followed by the silent installation syntax preferred.





Elevation will take place if run from an unelevated process and the UAC is on.

Syntax

Syntax

<pre>Qlik_Sense_setup.exe [-silent] [-uninstall] {-log path\filename} {layout=path} {accepteula=1 0} {desktopshortcut=1 0} {skipstartservices=1 0} {installdir=path} {userwithdomain=domain\user} {userpassword=password} {dbpassword=password} {hostname=www.machinename.domain.com} {cleanup=1 0} {sharedpersistenceconfig="configfilepath"} {senddata=1 0} {skipvalidation=1 0}</pre>	-
<pre>Qlik_Sense_setup.exe -? or -h</pre>	Brings up the on-screen silent setup help.


Commands

Commands		
Command	Type	Purpose
<code>-silent</code> (or <code>-s</code>)	-	Command line-driven setup without UI (mandatory).
<code>-uninstall</code>	-	Uninstall the product silently. It must be used with <code>-silent</code> command.
<code>-log</code> (or <code>-l</code>)	[log file name with path]	Log file directory and log file name. <div style="border: 1px solid gray; padding: 5px;">  <i>The user must have access to this directory.</i> </div>
<code>-layout</code>	[destination directory]	Extracts files (including <code>.msi</code> files) to the destination directory. <div style="border: 1px solid gray; padding: 5px;">  <i>This argument should not be combined with other command line arguments.</i> </div>



Arguments

Arguments are separated by a space and presented in the form `[Argument]=[Value]`. The double quotes can normally be omitted but may be needed, for example, when a path contains spaces.

The default values are the same as those used in the setup user interface.

Arguments		
Argument	Values	Purpose
<code>accepteula</code>	1 0	Accepts the Qlik User License Agreement. <div style="border: 1px solid gray; padding: 5px;">  <i>This argument is mandatory when installing or upgrading, and you must accept the QULA to install successfully.</i> </div>
<code>desktopshortcut</code>	1 0 (defaults to 1 on clean installs)	Installs desktop shortcuts.
<code>skipstartservices</code>	1 0 (defaults to 0 on clean installs, otherwise the current state.)	Skips starting services after the installation has finished.

4 Running the installer silently

installdir	[path to custom install directory]	Defines the directory if the default install directory will not be used (<i>%ProgramFiles%\Qlik\Sense</i>).
userwithdomain	[domain\username]	Adds the username to run the Qlik Sense services.
userpassword	[password]	Adds the password to run the services.
dbpassword	[password]	Adds the password for the database superuser that creates the user that runs the database.
hostname	[address of the central node]	Define the address for the central node. The central node uses certificates to communicate securely with other servers. Leave blank to use the default.
cleanup	1 0 (defaults to 0 on uninstall)	Deletes Qlik Sense certificates and any files in the <i>ProgramData\Qlik\Sense</i> directory after the uninstall is completed. <div style="border: 1px solid gray; padding: 5px; margin-top: 10px;">  <i>This argument must be used with the silent install commands.</i> <i>Example: -silent -uninstall cleanup=1.</i> </div>
sharedpersistenceconfig (or spc)	[path to configuration file including the filename]	Activates setup of shared persistence as storage method. All settings for shared persistence must be in the configuration file referenced here. <div style="border: 1px solid gray; padding: 5px; margin-top: 10px;">  <i>This is a parameter must be configured to install successfully.</i> </div> <p><i>Shared persistence configuration file syntax (page 174)</i></p>
senddata	1 0 (defaults to 0)	Shares system data with Qlik in anonymous form.

4 Running the installer silently

skipvalidation	1 0 (defaults to 0)	Skips password validation process for service user and shared folder access. For silent installation, database connection tests are also skipped.
bundleinstall	dashboard,visualization	Includes the dashboard and visualization bundles.



If you enter a username that is more than 20 characters long, it must be in UPN format, and must include the full domain name. For example, longusername@full.domain.name.

Example 1: To install Qlik Sense

```
Qlik_Sense_setup.exe -s spc="\\configpath\spc.cfg"
userwithdomain=mydomain\myUser userpassword=myPassword
dbpassword=mydbpassword accepteula=1
```

Example 2: To install Qlik Sense while redirecting the installation and log files to a different location

```
Qlik_Sense_setup.exe -s -l c:\mylogpath spc="\\configpath\spc.cfg"
installdir=c:\mycustompath userwithdomain=mydomain\myUser
userpassword=myPassword dbpassword=mydbpassword accepteula=1
```

Shared persistence configuration file syntax

Configure the shared persistence storage model using the `sharedpersistenceconfig` argument and point to a configuration file that contains the settings to be used in the installation.

Example:

```
Qlik_Sense_setup.exe -s spc="\\configpath\spc.cfg"
userwithdomain=domain\yourserviceuser userpassword=yourserviceuserpassword
dbpassword=yoursuperuserpassword accepteula=1
```

The configuration file is in XML format. You need to create the file according to the example described here.

```
<?xml version="1.0"?>
<SharedPersistenceConfiguration xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <DbUserName>username</DbUserName>
  <DbUserPassword>password</DbUserPassword>
  <DbHost>IP or Hostname</DbHost>
  <DbPort>4432</DbPort>
  <RootDir>\\server\share</RootDir>
  <StaticContentRootDir>\\server\share\StaticContent</StaticContentRootDir>
  <ArchivedLogsDir>\\server\share\ArchivedLogs</ArchivedLogsDir>
  <AppsDir>\\server\share\Apps</AppsDir>
  <CreateCluster>true</CreateCluster>
  <InstallLocalDb>false</InstallLocalDb>
  <ConfigureDbListener>true</ConfigureDbListener>
```

```
<ListenAddresses>*/ListenAddresses>
<IpRange>0.0.0.0/0,::/0</IpRange>
<MaxConnections>100</MaxConnections>
<!--<JoinCluster>true</JoinCluster>-->
<HttpPortNumber>80</HttpPortNumber>
<HttpsPortNumber>443</HttpsPortNumber>
<EnableHttpPort>false</EnableHttpPort>
<DbMaximumConnectionPoolSize>90</DbMaximumConnectionPoolSize>
</SharedPersistenceConfiguration>
```

Configuration file syntax

Configuration file syntax

Setting	Description
DbUserName	User name of the repository database user.
DbUserPassword	Password of the repository database user.
DbHost	Hostname of the machine running the repository database.
DbPort	Port used to communicate with the repository database.
RootDir	Root directory for the file share to use as content storage. We recommend that you keep the content in this folder's sub-directories, but this can be changed in the StaticContentRootDir and ArchivedLogsDir settings.
AppsDir	Directory to store apps in.
StaticContentRootDir	Root directory for all static content of the site.
ArchivedLogsDir	Directory to save archived log files in.
CreateCluster JoinCluster	Set CreateCluster to true if you want to create a new cluster, or set JoinCluster to true if you want to join an existing cluster. You can only use one of these settings in the configuration file. The other setting needs to be removed, or commented out like <!--<JoinCluster>true</JoinCluster>-->.
InstallLocalDb	Set to true if you want to install a local PostgreSQL database on the node when you create a new cluster. This setting can only be used together with the CreateCluster setting.
ConfigureDbListener	Set to true if you want to configure the PostgreSQL database installed by Qlik Sense to listen to database connections from other nodes. You need to configure the ListenAddresses and IpRange settings.
ListenAddresses	Addresses that the database service should listen to. You can supply a comma separated list of IPv4 or IPv6 addresses, or 0.0.0.0 (for all IPv4 addresses), ::/0 (for all IPv6 addresses) or * (for all addresses).

Setting	Description
IpRange	Subnet specification that covers the IP addresses of all nodes in your site. Either add one row for each node, using /32 as suffix for each address, or add a subnet that covers all addresses using, for example, /24 as suffix. To allow all servers to access the repository database, use 0.0.0.0/0. If entering multiple IP addresses or ranges, use a comma separated list. A range can be either IPv4 or IPv6.
MaxConnections	Specify the maximum number of concurrent connections to the database. The default value is 100. If you have a multi-node site multiple this value by the number of nodes in the cluster. For example, <code><MaxConnections>100</MaxConnections></code> is a single server deployment.
HttpPortNumber	Listening port when using http. Default port: 80.
HttpsPortNumber	Listening port when using https. Default port: 443.
EnableHttpPort	By default, the https port is used. Select EnableHttpPort to use the http port.
DbMaximumConnectionPoolSize	Maximum connection pool size for the repository database. Minimum value: 90, maximum value:1000.

Deprecated command line arguments

The use of the following command line arguments is no longer recommended.

Deprecated arguments

Argument	Purpose
<code>rimnode</code>	Determines the Repository role.
<code>-rimnodetype</code> (or <code>-rnt</code>)	Installs all the features required for the rim node type selected. The node type can be any one of: Complete, Proxy, Engine, Scheduler.
<code>SetupLocalLoggingDb</code>	Centralized logging
<code>QLogsWriterPassword</code>	Centralized logging
<code>QLogsReaderPassword</code>	Centralized logging
<code>QLogsHostname</code>	Centralized logging
<code>QLogsPort</code>	Centralized logging

4.2 Silent upgrading

You can silently upgrade the current Qlik Sense installation. All setup options that are available in the user interface of the installer can be performed with silent operations.

Do the following:

1. Select **Start > All Programs > Accessories > Command Prompt**.
The **Command Prompt** window is displayed.
2. In the **Command Prompt** window, navigate to the folder containing the *Qlik_Sense_setup.exe* file.
3. Enter *Qlik_Sense_setup.exe* followed by the silent installation syntax preferred.
4. If applicable, .NET Framework 4.8 will be installed. You must reboot the system after it has finalized.
When the system has re-started, repeat the steps above.
5. If applicable, recreate the Qlik Sense root certificate according to this Support article: [Recreating Qlik Sense root CA certificate when upgrading to June 2019 and above](#).
Applicable for all Qlik Sense deployments originally installed with version June 2019 or earlier.



Note that elevation will take place if run from an unelevated process and the UAC is on.



If your current deployment includes custom configuration files, they are by default retained during the upgrade process. See: *keepcurrentsettings* (page 178) and *Considerations about custom configurations* (page 161).

Syntax

Syntax

<code>Qlik_Sense_setup.exe [-silent] {-log path\filename}</code>	-
<code>{accepteula=1 0} {desktopshortcut=1 0} {skipstartservices=1 0}</code>	
<code>{installdir=path} {userpassword=password} {dbpassword=password}</code>	
<code>{keepcurrentsettings=1 0}</code>	
<code>Qlik_Sense_setup.exe -? or -h</code>	Brings up the on-screen silent setup help.

Commands

Commands


-silent (or -s)	-	Command line-driven setup without UI.(mandatory).
-log (or -l)	[log file name with path]	Log file directory and log file name.
		The user must have access to this directory.

Arguments

Arguments are separated by space and presented in the form [Argument]="[Value]". The double quotes can normally be omitted but may be needed, for example, when a path contains spaces.

The default values are the same as those used in the setup user interface.

Arguments

accepteula	1 0	Accepts the Qlik User License Agreement.  <i>This argument is mandatory, and you must accept the QULA to upgrade successfully.</i>
desktopshortcut	1 0 (defaults to 1 on clean installs)	Installs desktop shortcuts.
skipstartservices	1 0 (defaults to 0 on clean installs, otherwise the current state.)	To skip starting services after the installation has finished.
installdir	[path to custom install directory]	Need only be defined if the default install directory will not be used (%ProgramFiles%\Qlik\Sense).
userpassword	[password]	The password of the user used to run the services.
dbpassword	[password]	Password for the database superuser that creates the user that runs the database.
bundleinstall	dashboard,visualization	Includes the dashboard and visualization bundles.
keepcurrentsettings	1 0 (defaults to 1)	In Upgrade, persists current service settings for the following service config files: Repository.exe.config, Proxy.exe.config, Scheduler.exe.config, and Services.conf.

The default values are the same as those used in the setup user interface.

Example: Upgrading the installation

This example shows how to silently upgrade an installation and add desktop shortcuts.

```
Qlik_Sense_setup.exe -s desktopshortcut=1 accepteula=1
```

Deprecated command line arguments

For a list of the command line arguments that are no longer recommended, see [Installing silently](#).

4.3 Silent repairing

You can silently repair the current Qlik Sense installation. All setup options that are available in the user interface of the installer can be performed with silent operations.


Do the following:

1. Select **Start > All Programs > Accessories > Command Prompt**.
The **Command Prompt** window is displayed.
2. In the **Command Prompt** window, navigate to the folder containing the *Qlik_Sense_setup.exe* file.
3. Enter *Qlik_Sense_setup.exe* followed by the silent installation syntax preferred.

Syntax

Syntax	
<code>Qlik_Sense_setup.exe [-silent] [-repair] {-log path\filename}</code>	-
<code>Qlik_Sense_setup.exe -? or -h</code>	Brings up the on-screen silent setup help.

Commands

Commands	
<code>-silent (or -s)</code>	Trigger the silent mode (mandatory).
<code>-repair</code>	Repair the product silently.
<code>-log (or -l)</code>	<p>Log file directory and log file name.</p> <div style="border: 1px solid gray; padding: 5px; margin: 5px 0;">  <i>The user must have access to this directory.</i> </div> <p>If this option is not defined, the log file will be stored with the default name in the default location.</p>

Example:

This example shows how to silently repair the Qlik Sense installation.

```
Qlik_Sense_setup.exe -s -repair
```

4.4 Silent patching

When a software patch is available for your Qlik Sense installation, you can use the command line tool to silently install the updates. Patches include software updates and fixes that are applied to the existing Qlik Sense version.

Commands


Use the following commands to silently run patch updates.

List of commands

Command	Description
install	Runs a command line-driven install without a user interface. For feedback, see the log files, and the return values.
uninstall	Runs a command line-driven uninstall without a user interface. For feedback, see the log files, and the return values.
startservices	Used with [install], or [uninstall], this command determines whether the services should be started automatically or not.
log=[path to logfile]	Specifies the location for the patch to writes log files.
unpack=[path]	Unpacks the patch contents without installing.
help (or -h, /h, -?, /?)	Opens the help dialog.

To troubleshoot silent patching, start by examining the installation log files. The default location of the log files is: `C:\Users\[username]\AppData\Local\Temp`.

Recreating root certificates

If applicable, after the patching has completed, recreate the Qlik Sense root certificate according to this Support article:  <https://support.qlik.com/articles/000094071>.

Applicable for the following and later Qlik Sense patches:

- February 2019 Patch 8
- April 2019 Patch 8
- June 2019 Patch 11
- September 2019 Patch 7
- November 2019 Patch 6

Example

The following command is an example of the syntax you can use for running a patch update file:

```
qlik_sense_update.exe install startservices
```

This command installs the update, and restores the services to the same state they were in before the update.

5 Backup and restore Qlik Sense Enterprise on Windows

To ensure that your Qlik Sense site can be recovered in the event of a system failure or when a node in your deployment needs to be moved or replaced, we recommend that you create regular backups. These backups are used to restore your Qlik Sense site when needed.

To back up a deployment running Qlik Sense 3.2.x or earlier, refer to the documentation for the release that you are running.

To backup a Qlik Sense site, you must back up the following:

- Qlik Sense certificates
- Qlik Sense Repository Database
- Shared persistence file share

5.1 Qlik Sense certificates

Qlik Sense uses certificates to secure communication between components that are installed on different computers. It is recommended that you back up the certificates on the central node in a Qlik Sense site immediately after installation, so that they can be restored if needed.

Backed up certificates can be used to restore certificates on the same node as they were exported from. A backed up server certificate can also be moved from one node of a Qlik Sense site to another node in the same site. For more information, see *Restoring certificates (page 191)*.

For more information about how to back up the Qlik Sense certificates, see *Backing up a Qlik Sense site (page 201)*.

5.2 Qlik Sense Repository Database

The Qlik Sense Repository Database is a PostgreSQL database that contains system data and meta data about apps. The Qlik Sense Repository Database can reside on the central node or on another computer. If the Qlik Sense Repository Database was installed during setup it will be located on the central node. If the Qlik Sense Repository Database was installed manually, it may be located on another computer.

The Qlik Sense Repository Database should be backed up on a regular basis to avoid data loss.

For more information about how to back up the Qlik Sense Repository Database, see *Backing up a Qlik Sense site (page 201)*.

For more information about how to restore the Qlik Sense Repository Database, see *Restoring a Qlik Sense site (page 203)*.

5.3 Shared persistence file share

The shared persistence file share is used to store Qlik Sense app data, such as visualizations, and dimensions and measures. It also stores static content, such as images and extensions, as well as system logs. It is accessible to all nodes in your Qlik Sense site. The file share can reside either on the same server as the central node or on another server.

The file share should be backed up on a regular basis to avoid data loss.

For more information about how to back up the file share, see *Backing up a Qlik Sense site* (page 201).

For more information about how to restore the file share, see *Restoring a Qlik Sense site* (page 203).



Rim nodes maintain local log files that may be worth backing up in order to identify and investigate issues. It may also be worth backing up any general operating system data that may be required.

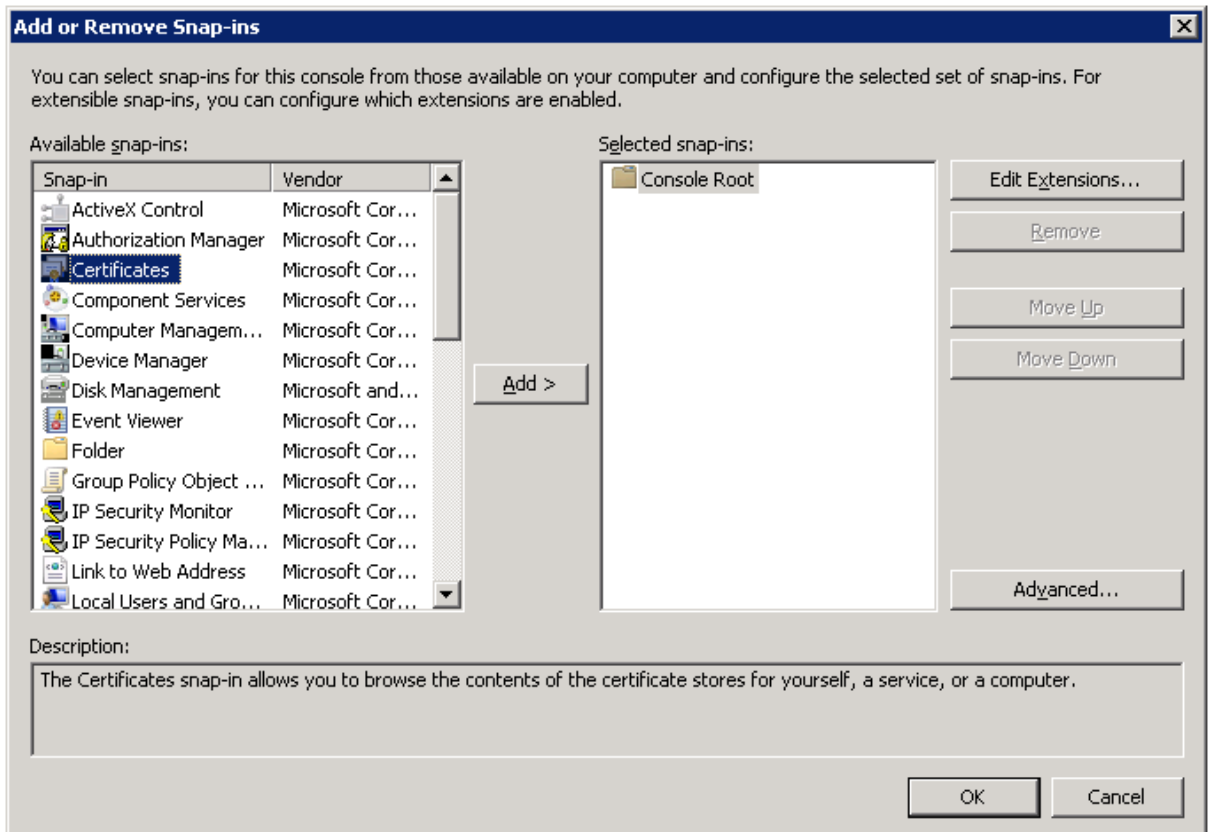
5.4 Backing up certificates

To be able to recover from a system crash, you should create a backup of the certificates on the central node of your Qlik Sense site.

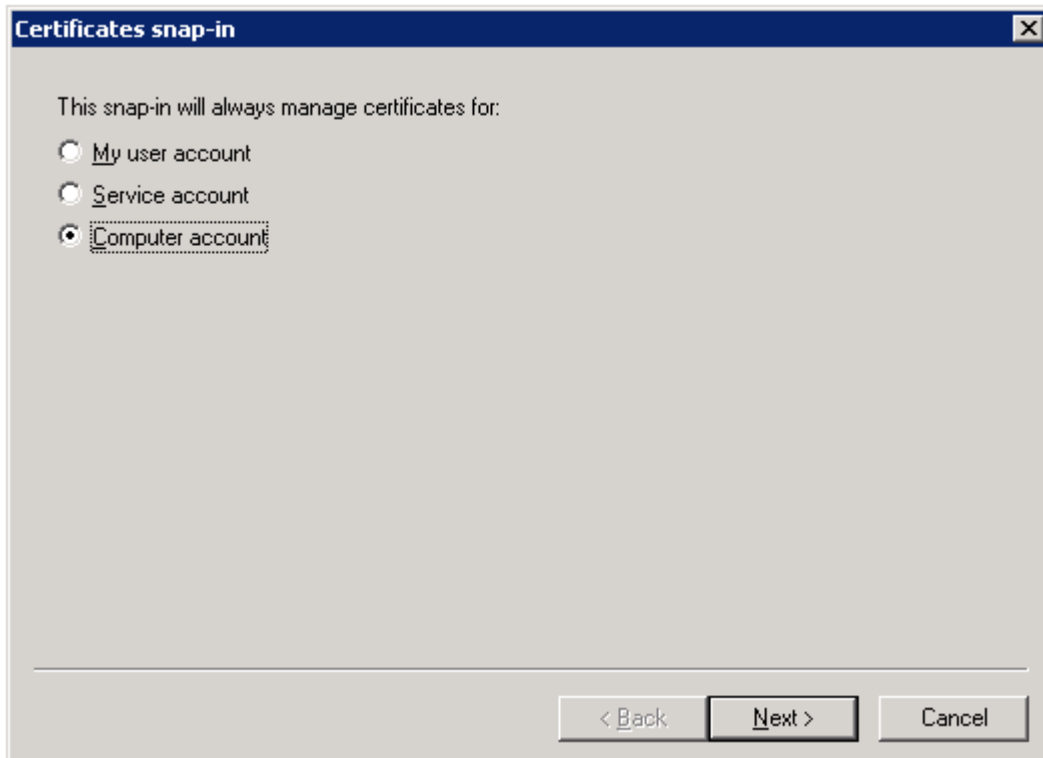
Do the following:

1. From the Windows start menu, type `mmc` to find the Microsoft Management Console (mmc) . Launch the mmc as the user that runs the Qlik Sense services.
2. Select **File>Add/Remove Snap-in**.

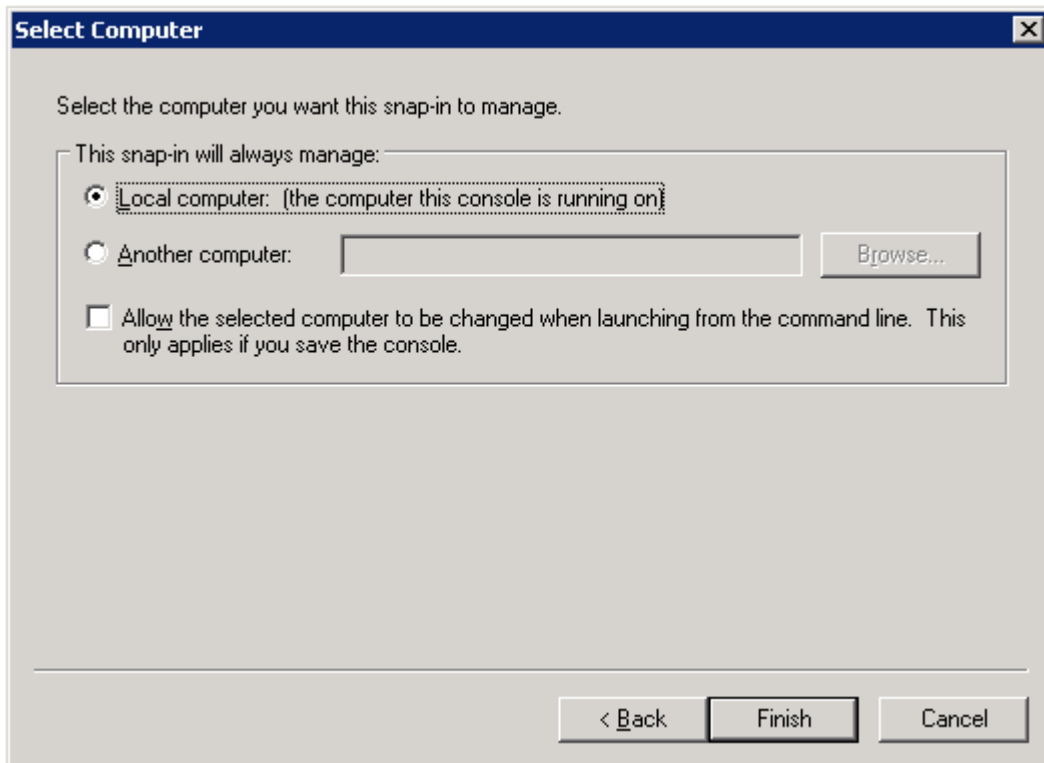
3. Double-click **Certificates**.



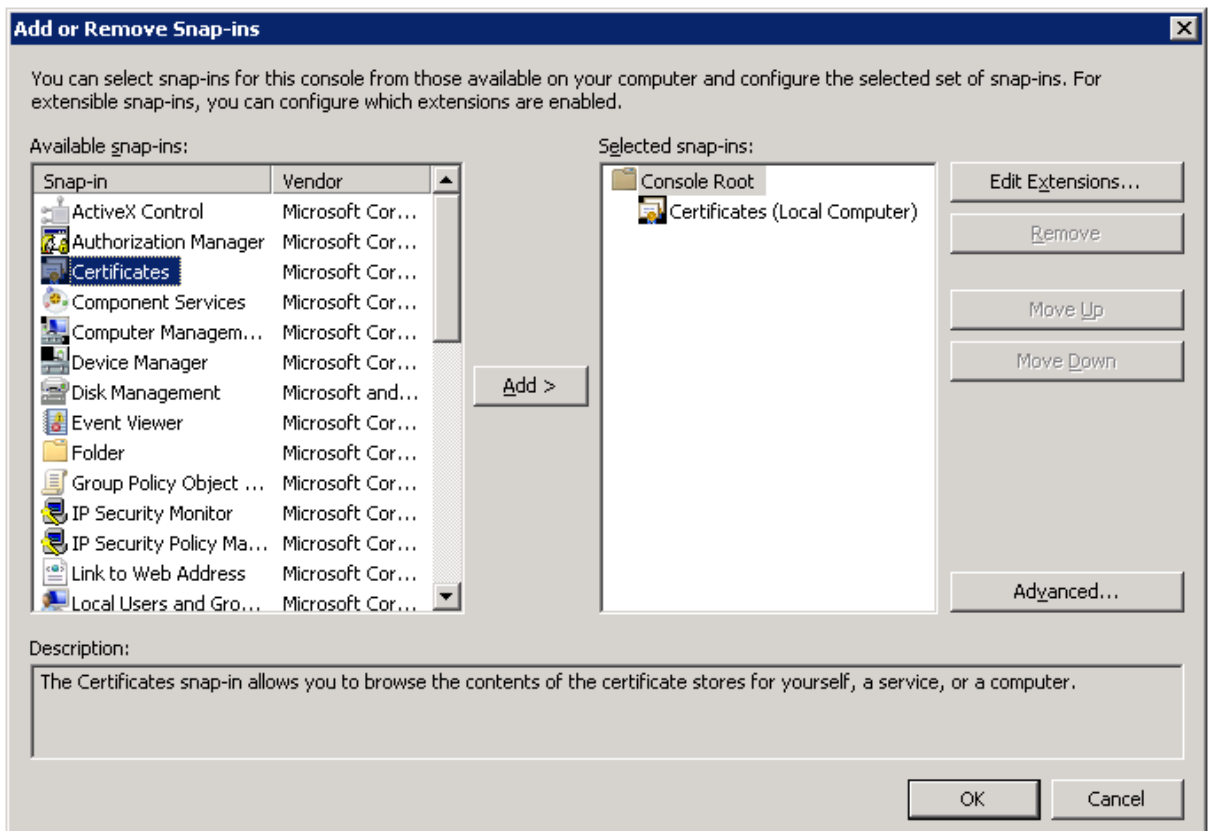
4. Select **Computer account**, then click **Next**.



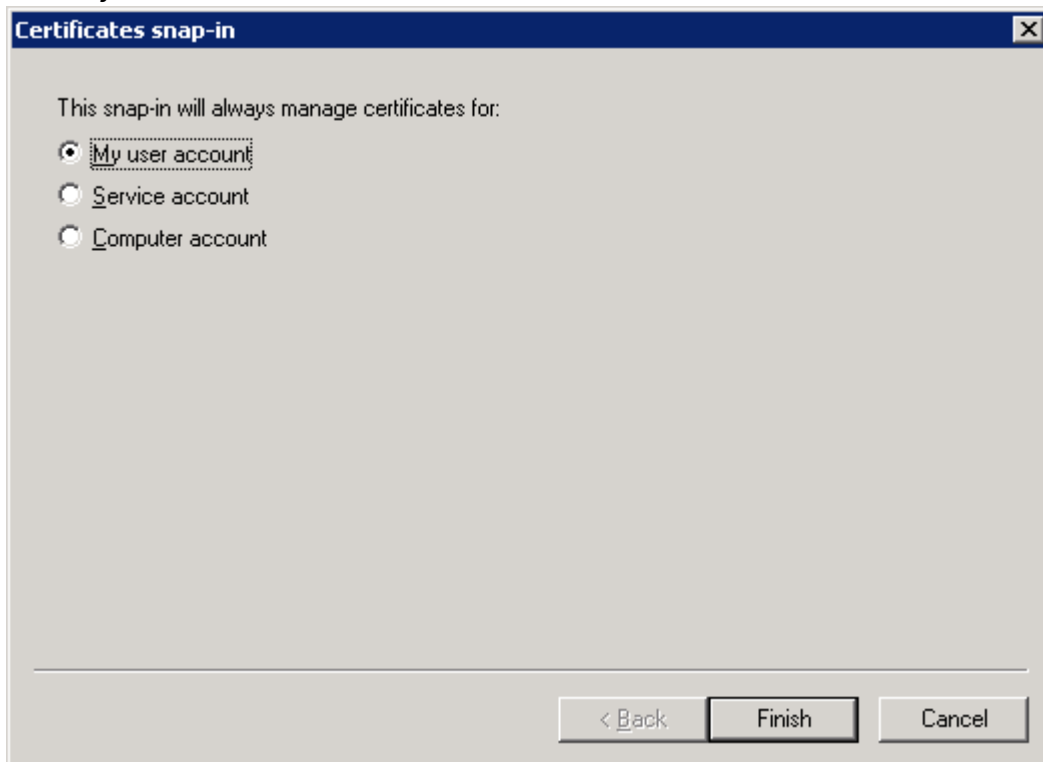
5. Select **Local computer**, then click **Finish**.



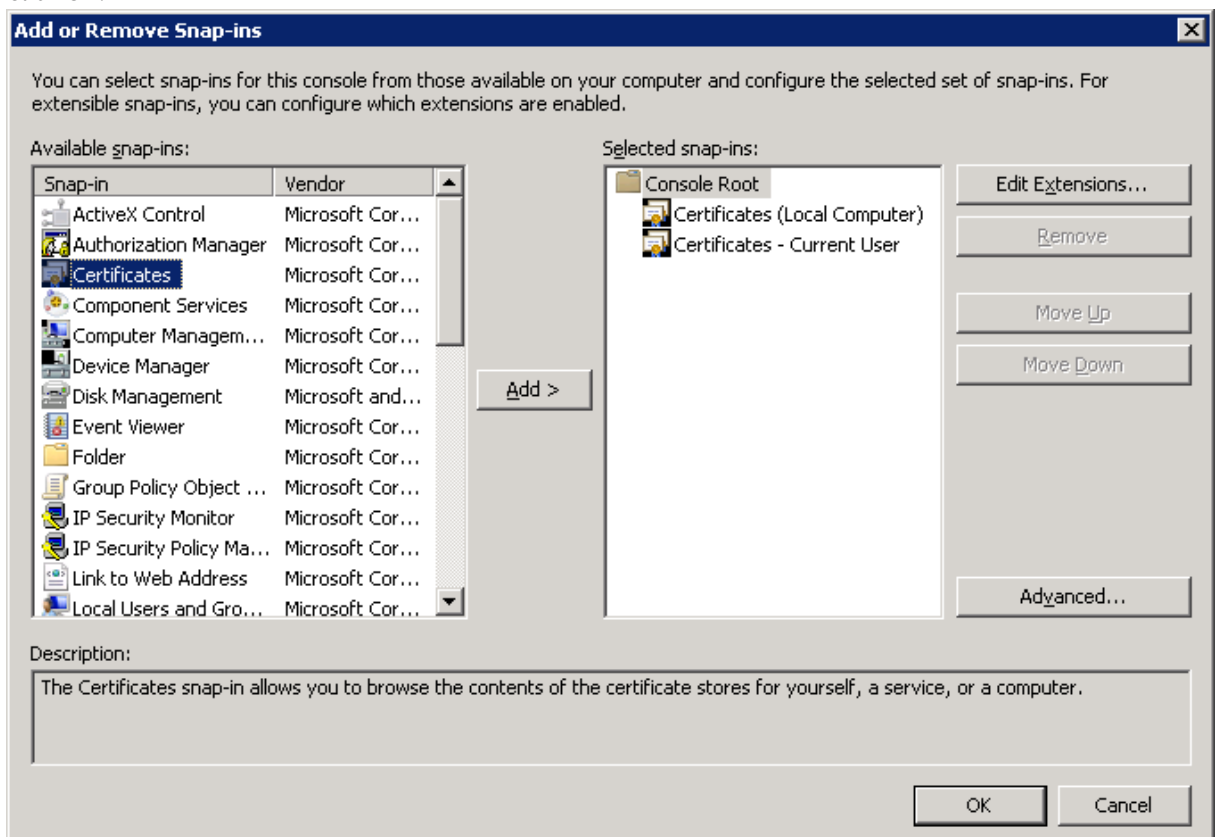
6. Double-click **Certificates**.



7. Select **My user account** and click **Finish**.



8. Click **OK**.



9. Complete this step for each of the following certificates:

5 Backup and restore Qlik Sense Enterprise on Windows

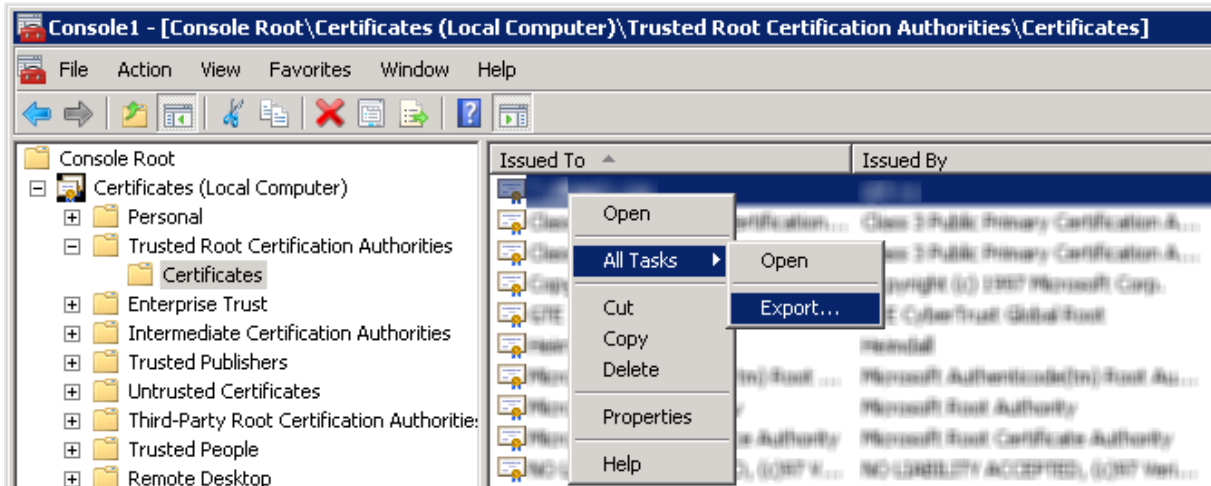
Certificate	Location	Issued to	Issued by
certificate authority	<i>Certificates (Local Computer) > Trusted Root Certification Authority > Certificates</i>	<server-name>-CA	<server-name>-CA
server certificate	<i>Certificates (Local Computer) > Personal > Certificates</i>	<server-name>	<server-name>-CA
client certificate	<i>Certificates (Current User) > Personal > Certificates</i>	QlikClient	<server-name>-CA
QlikServiceCluster certificate	<i>Certificates (Local Computer) > Personal > Certificates</i>	QlikServiceCluster	<server-name>-CA

- a. Expand the certificate location for the certificate you want to export.



For example, to export the Certificate Authority, expand Certificates (Local Computer) > Trusted Root Certification Authority > Certificates.

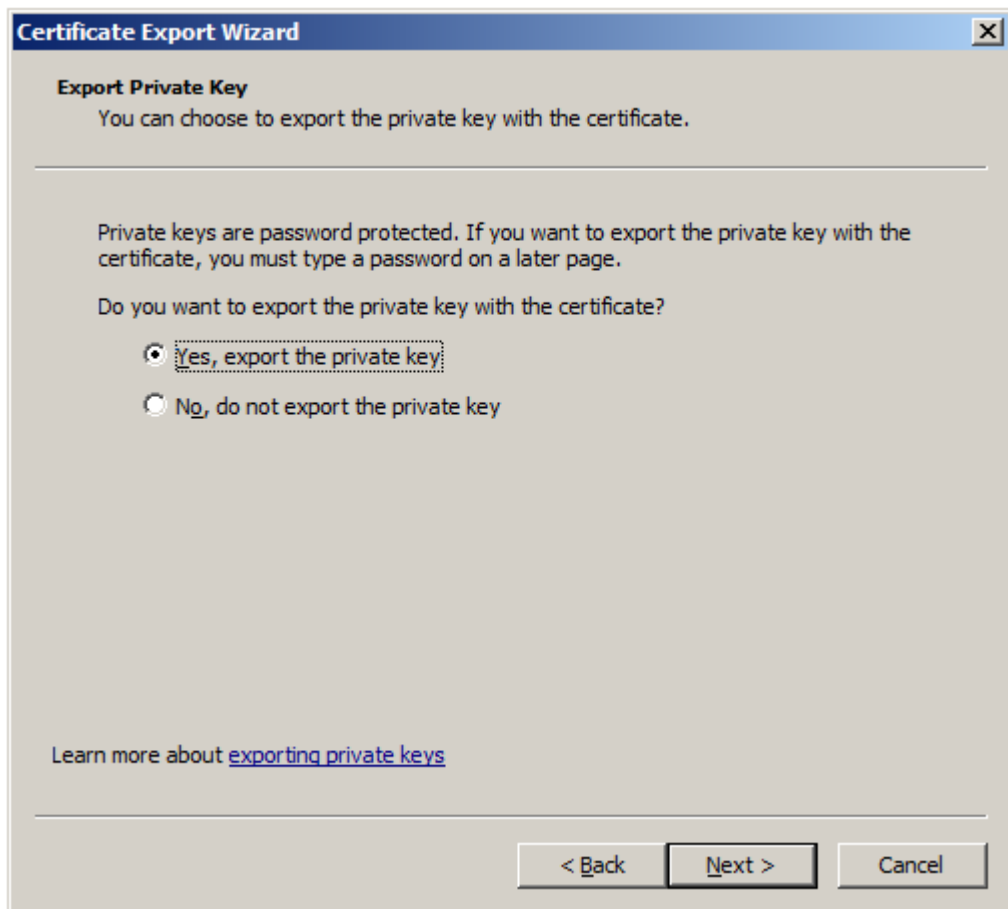
- b. Right-click the certificate you want to export, then select **All Tasks>Export**.



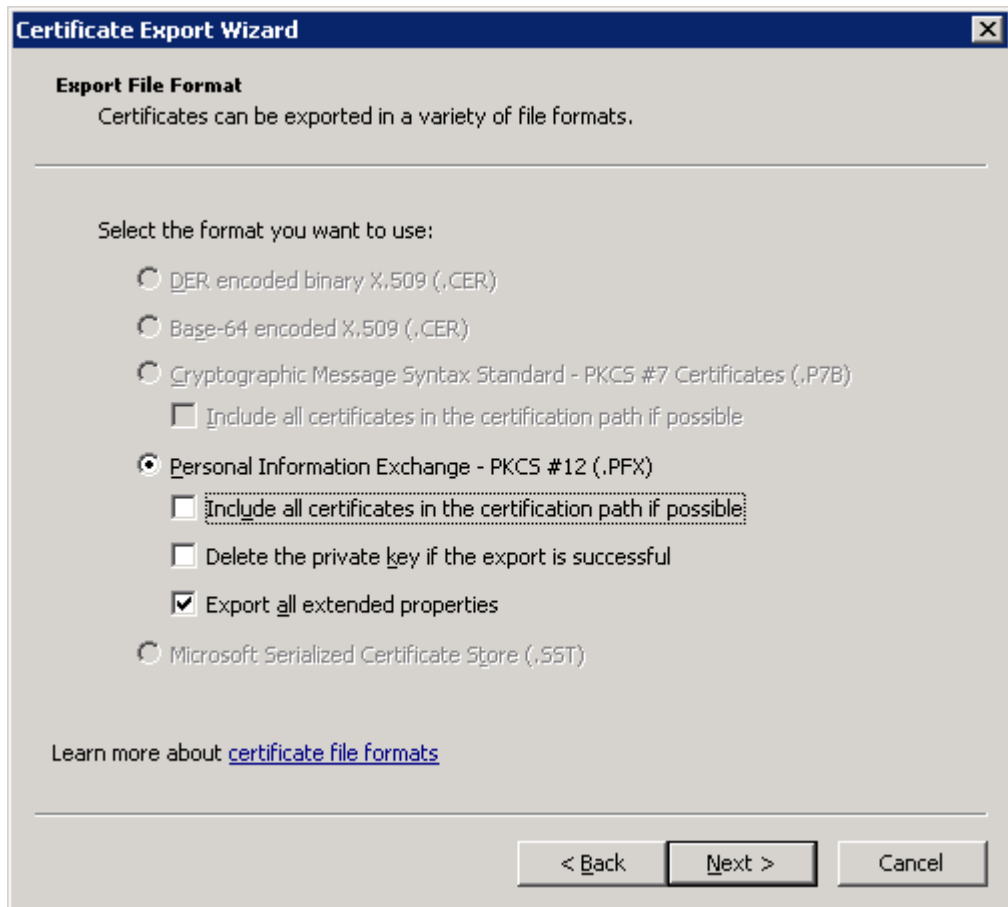
- c. Click **Next**.



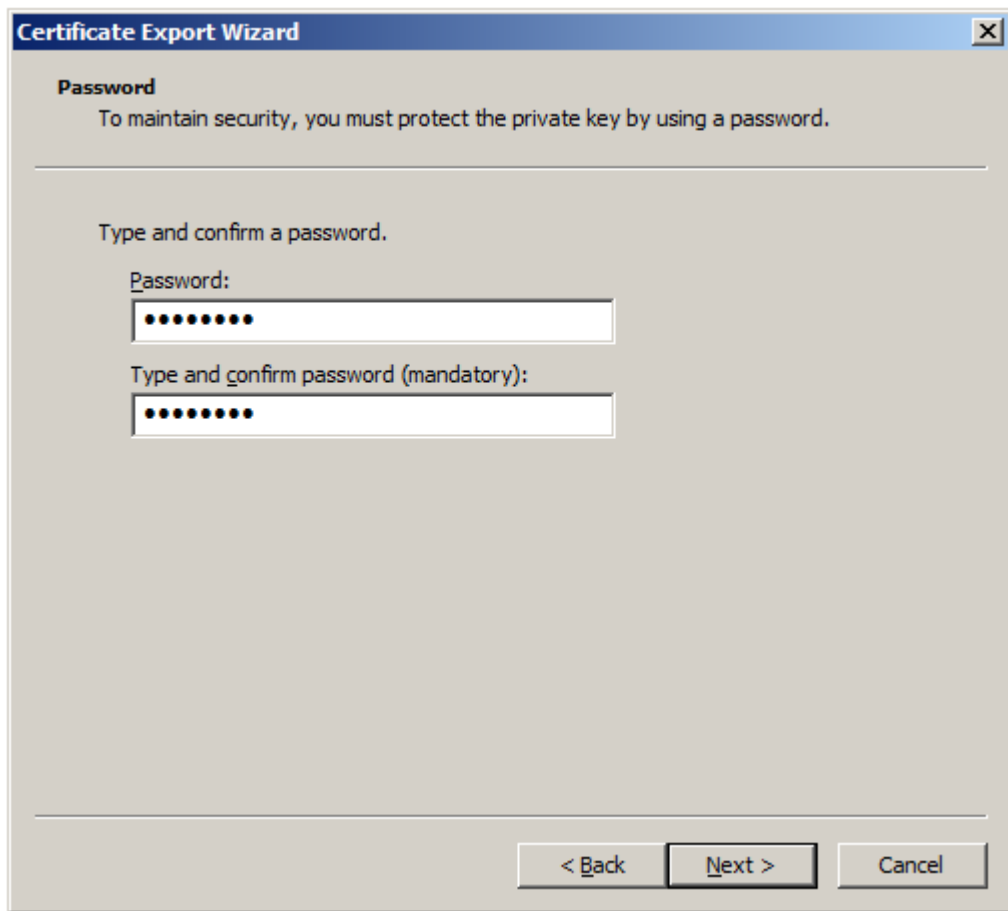
- d. Select **Yes, export the private key**, then click **Next**.



- e. Select **Personal Information Exchange**.
- f. Check the **Export all extended properties** box, then click **Next**.



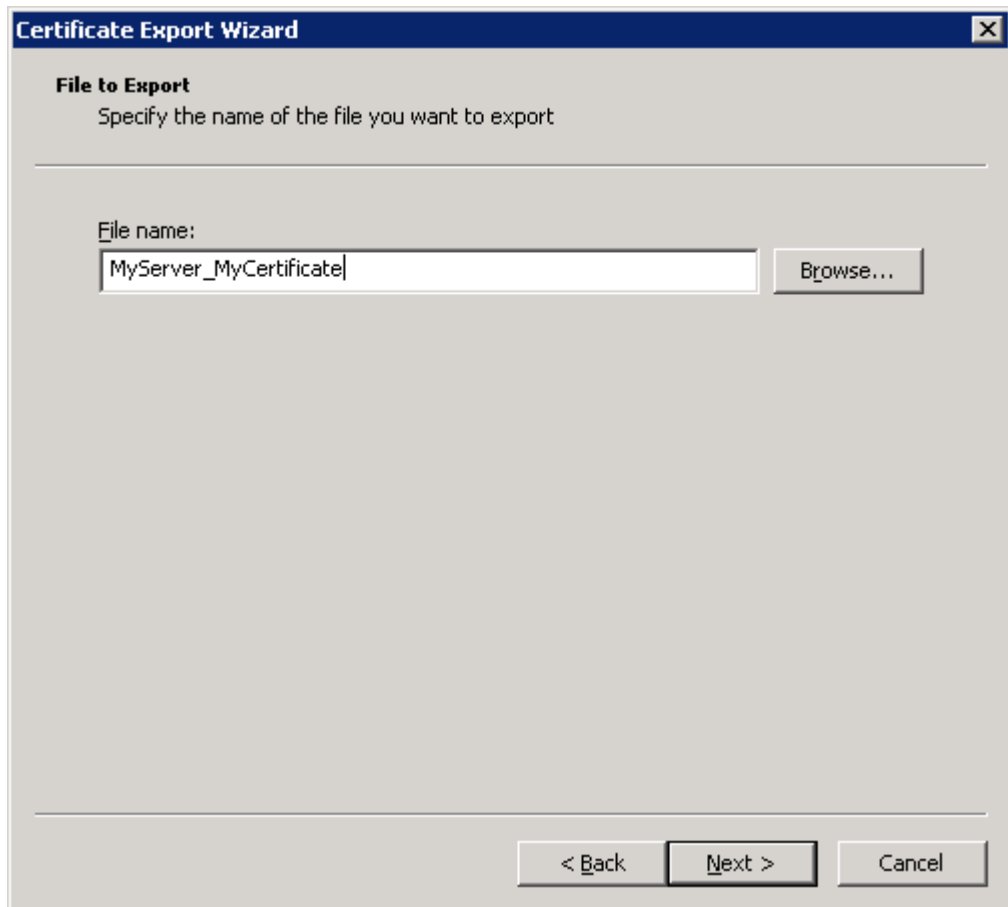
- g. Enter and confirm a password, then click **Next**. You may need to check the **Password** box before entering your password, depending on the Windows Server version. The password is needed when importing the certificate.



- h. Enter a file name for the *.pfx* file, then click **Next**.



We recommend to include the server name in the file name to avoid confusion with other certificate files.



- i. Click **Finish**.

The *.pfx* file that contains the CA for all nodes in the Qlik Sense site is stored in the selected location.



Make sure to complete this step for each certificate.

10. Close the MMC console.



You do not need to save these changes before closing.

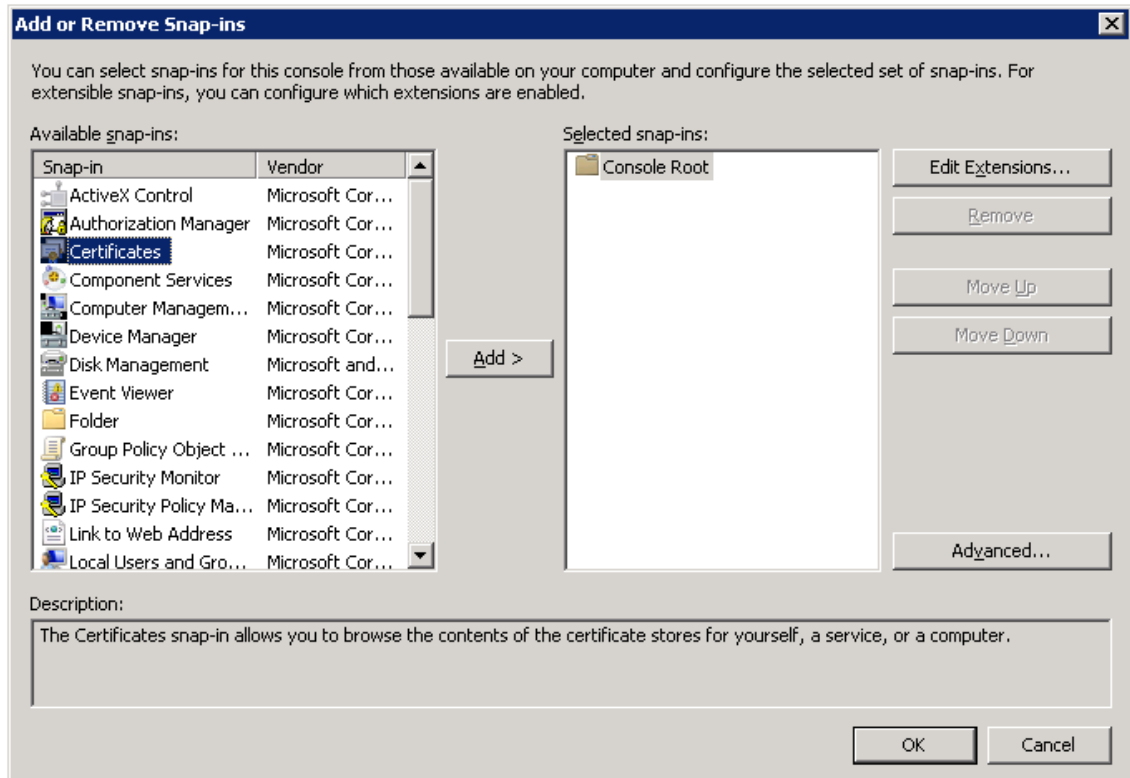
5.5 Restoring certificates

In case of a system crash, the certificates may need to be restored on the central node of your Qlik Sense site.

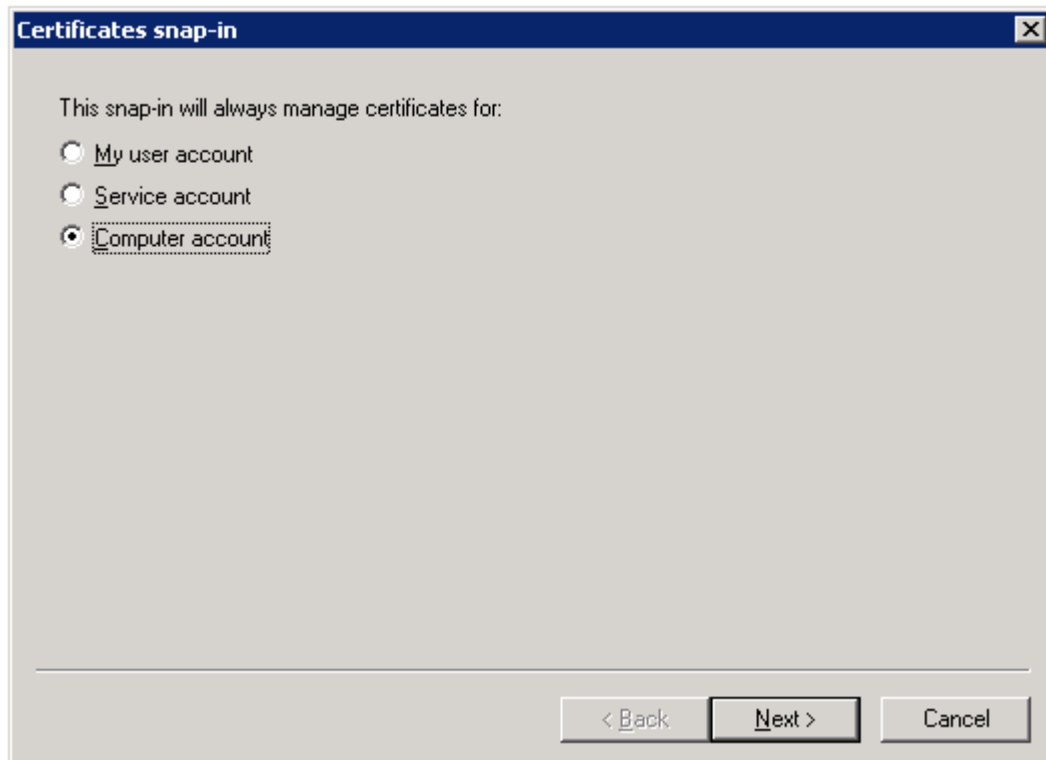
Do the following:

1. Open the Windows Services application to stop all Qlik Sense services except the Qlik Sense Repository Database (QRD) service.
2. From the Windows start menu, type *mmc* to find the Microsoft Management Console (mmc) . Launch the mmc as the user that runs the Qlik Sense services.

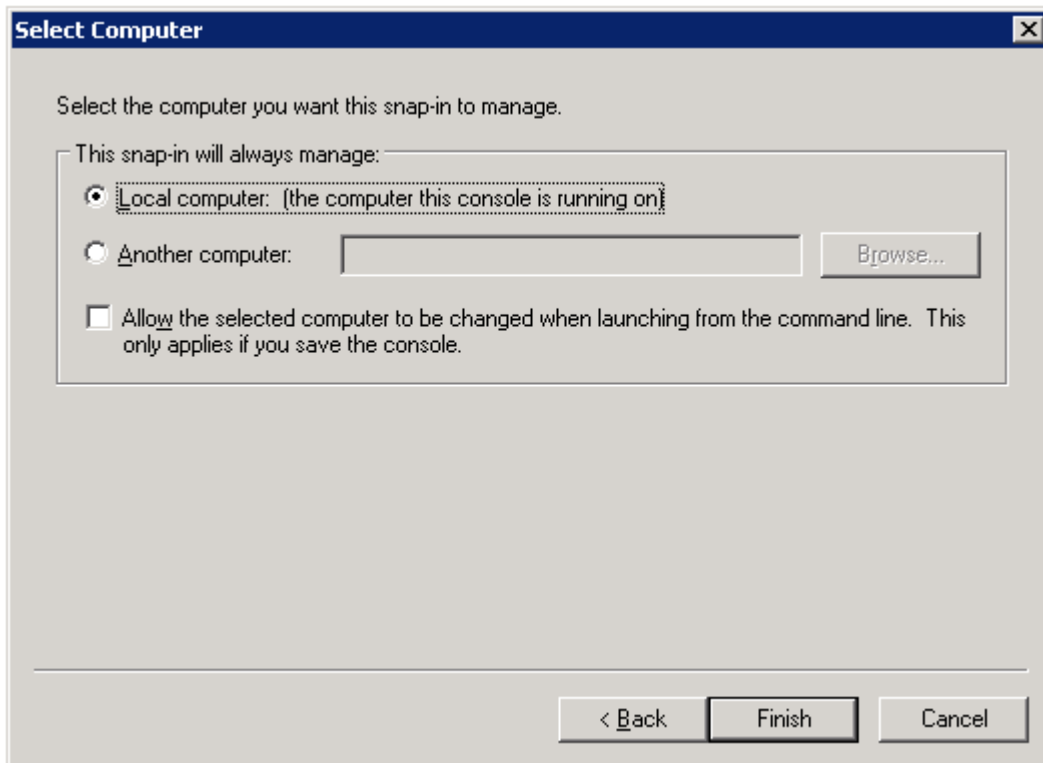
3. Select **File>Add/Remove Snap-in**.
4. Double-click **Certificates**.



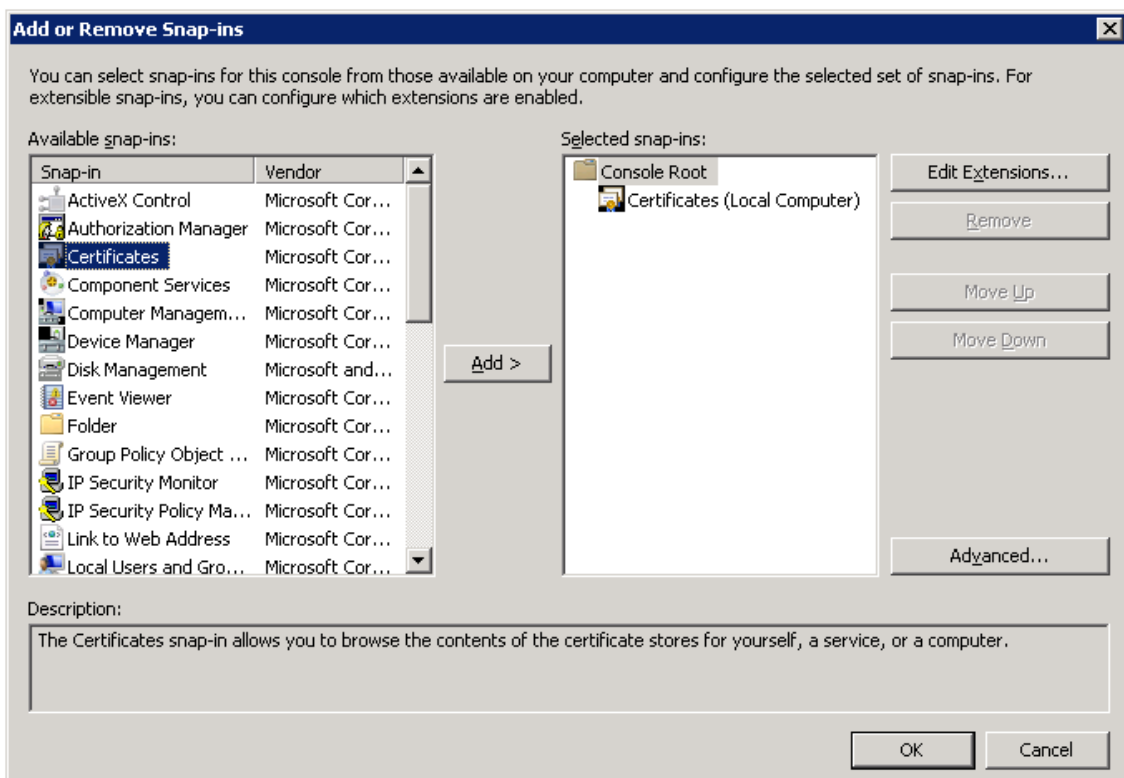
5. Select **Computer account** and click **Next**.



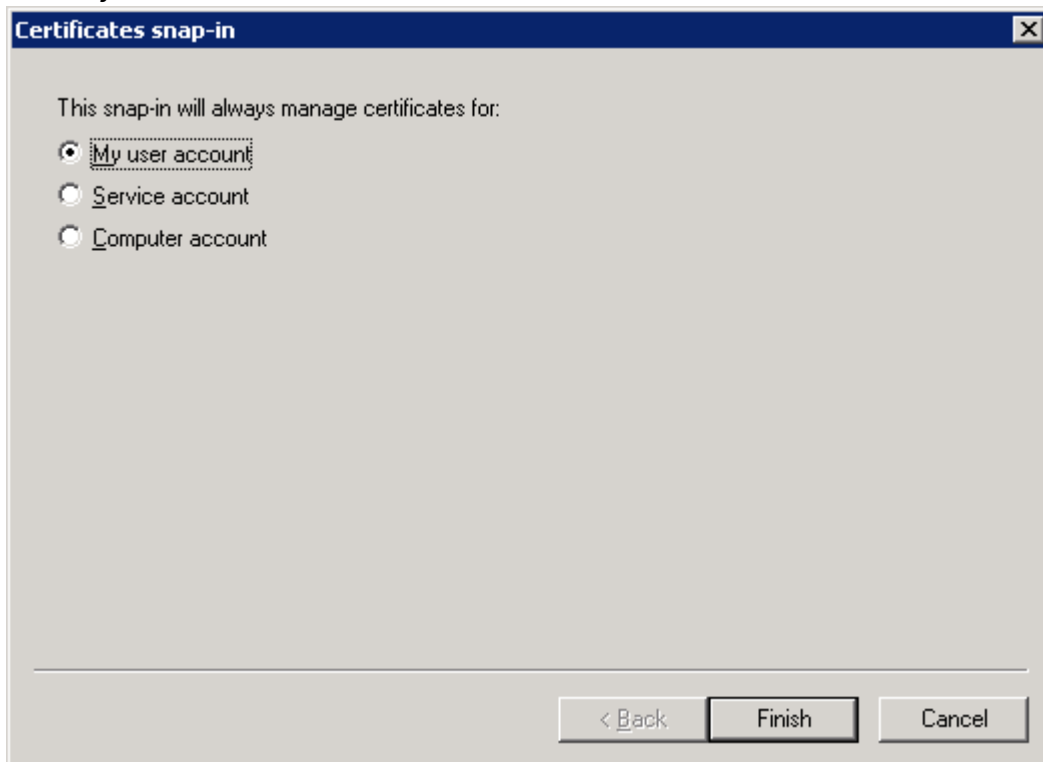
6. Select **Local computer** and click **Finish**.



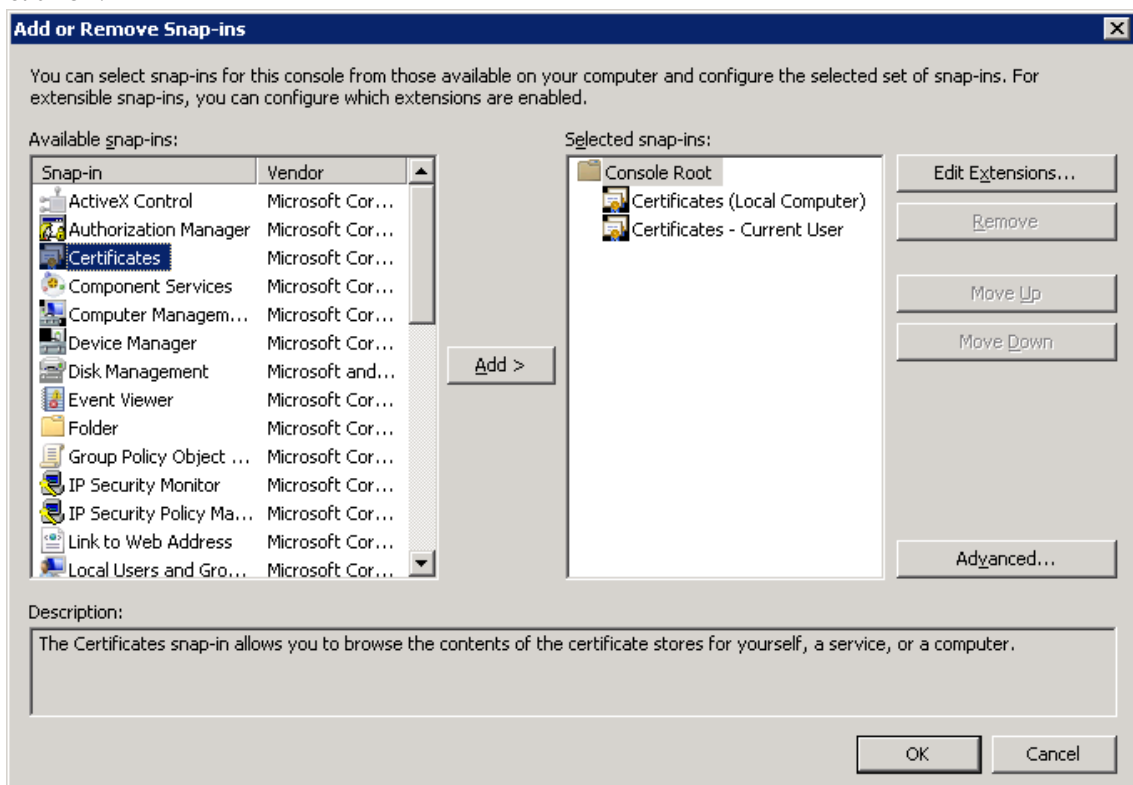
7. Double-click **Certificates**.



8. Select **My user account** and click **Finish**.



9. Click **OK**.



10. Complete this step for each of the backup certificates. Make sure to import the backup certificate to the correct location.

5 Backup and restore Qlik Sense Enterprise on Windows

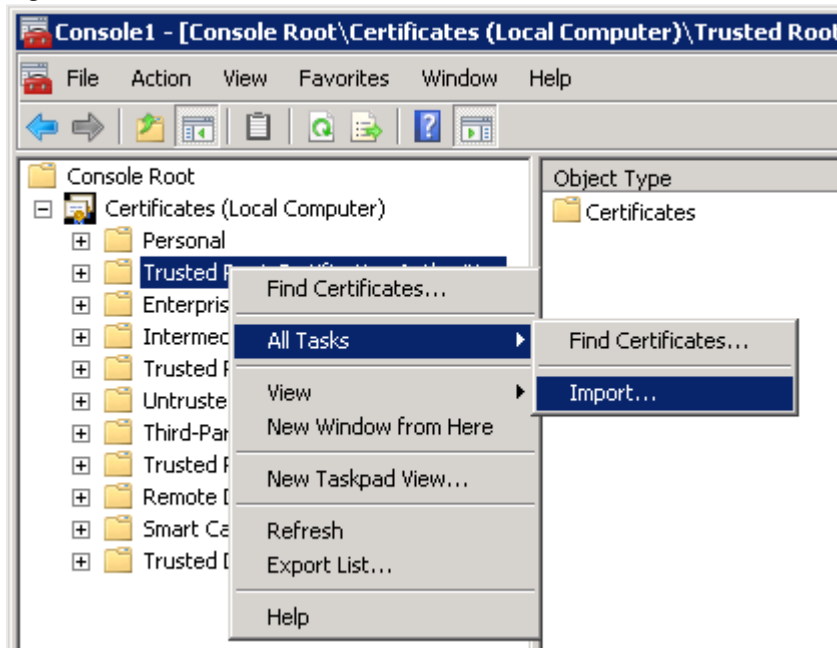
Backup certificate	Import Location	Issued to	Issued by
The backup certificate for the Certificate Authority.	<i>Certificates (Local Computer)</i> > <i>Trusted Root Certification Authority</i> > <i>Certificates</i>	<server-name>-CA	<server-name>-CA
The backup server certificate.	<i>Certificates (Local Computer)</i> > <i>Personal</i> > <i>Certificates</i>	<server-name>	<server-name>-CA
The backup client certificate.	<i>Certificates (Current User)</i> > <i>Personal</i> > <i>Certificates</i>	QlikClient	<server-name>-CA
The backup QlikServiceCluster certificate.	<i>Certificates (Local Computer)</i> > <i>Personal</i> > <i>Certificates</i>	QlikServiceCluster	<server-name>-CA

- a. Expand the certificate location for the certificate you want to import.

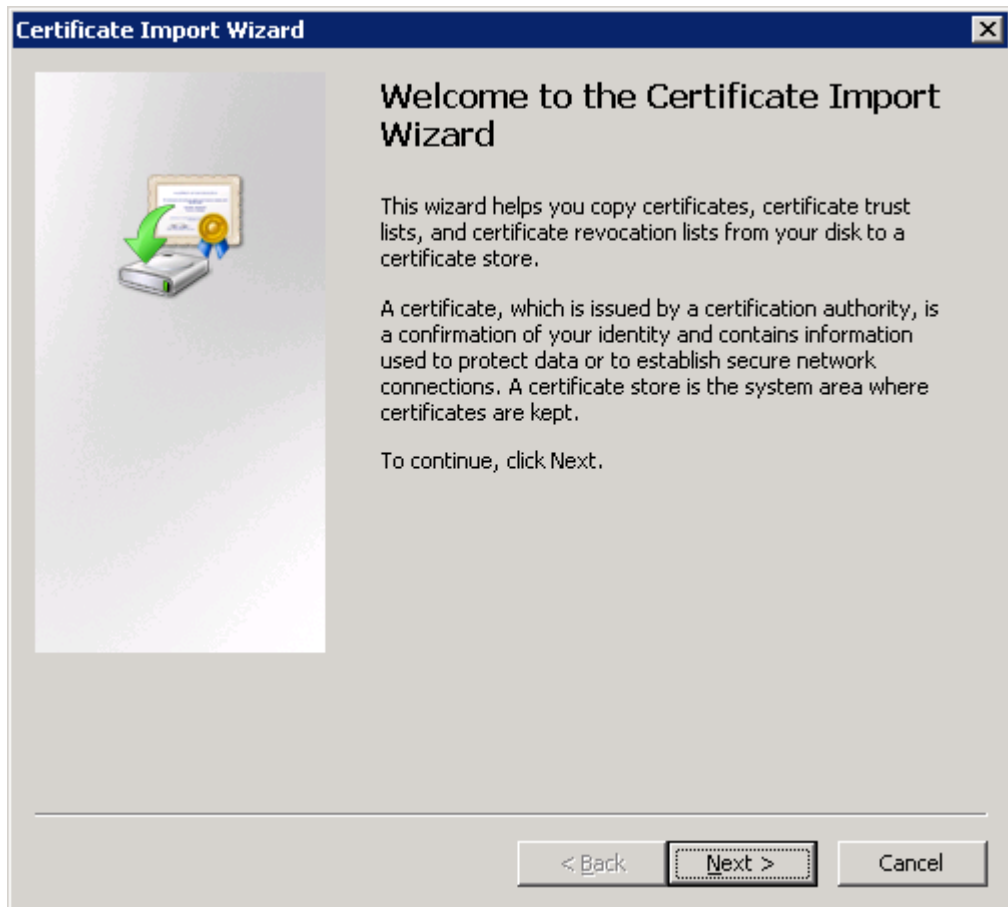


For example, to import the Certificate Authority, expand *Certificates (Local Computer)* > *Trusted Root Certification Authority* > *Certificates*.

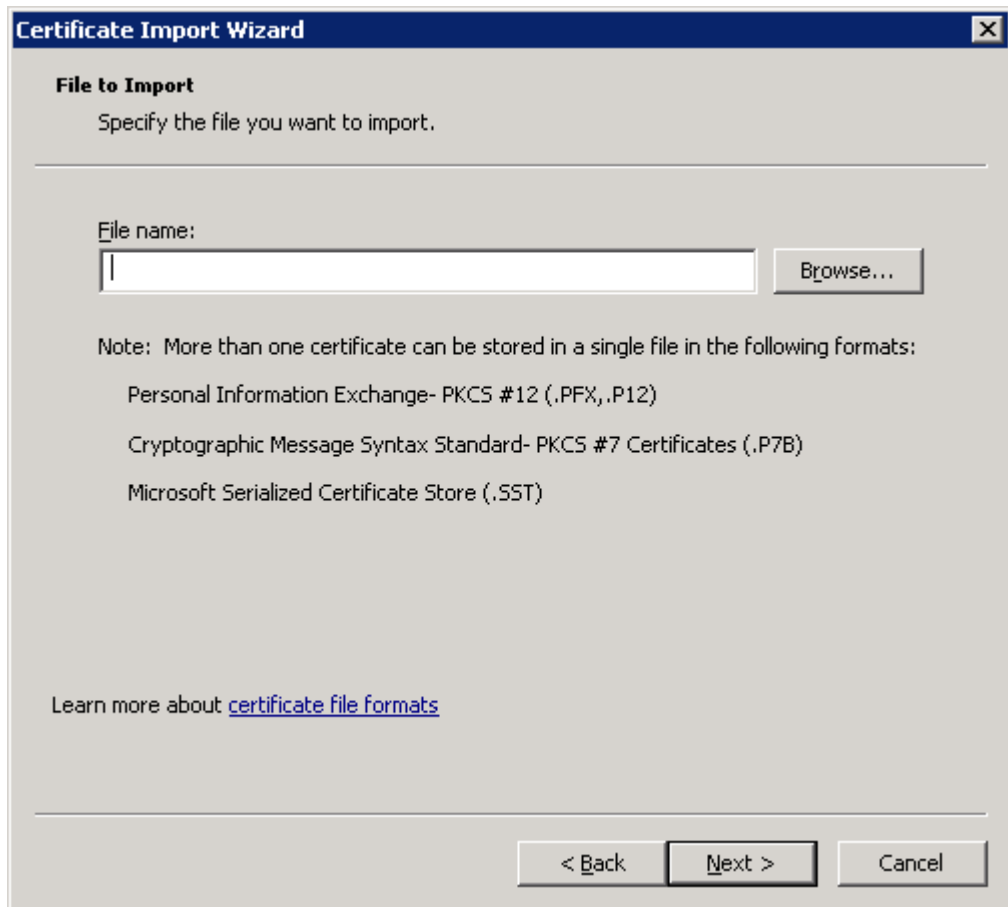
- b. Right-click the certificate folder, then select **All Tasks>Import**.



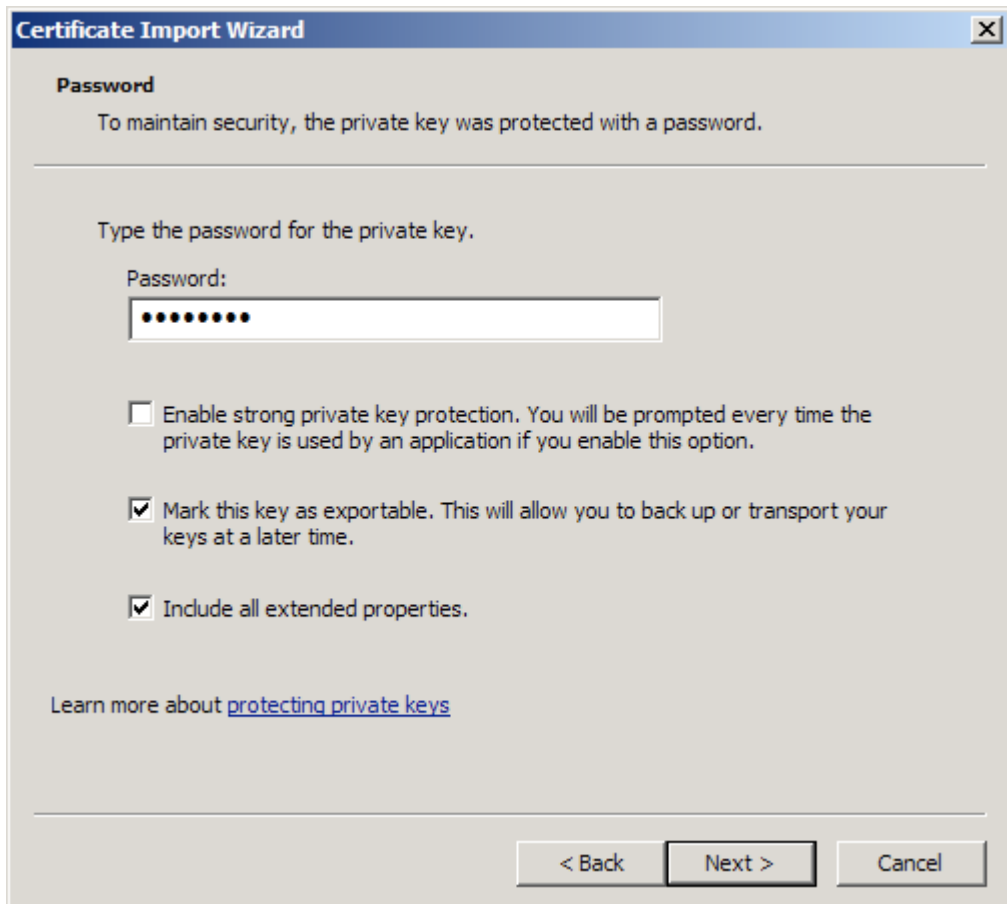
- c. Click **Next**.



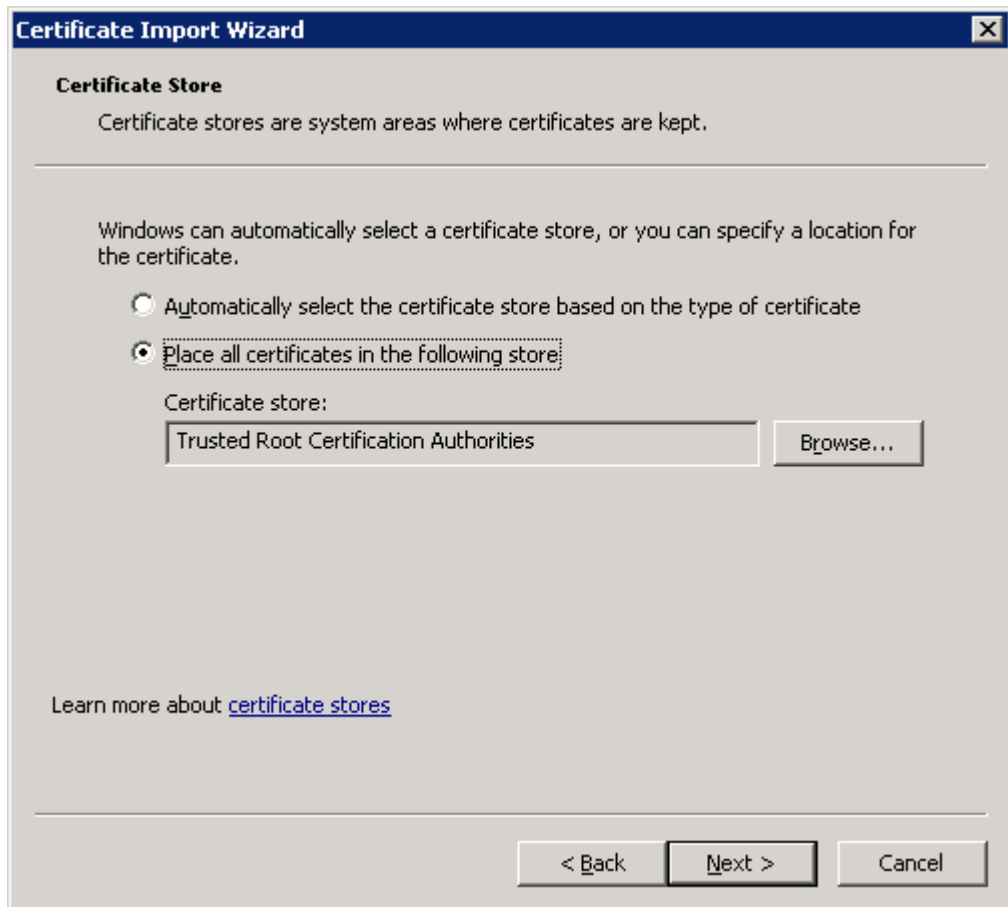
- d. Browse to the file that contains the backed up certificate (.pfx), then click **Next**.



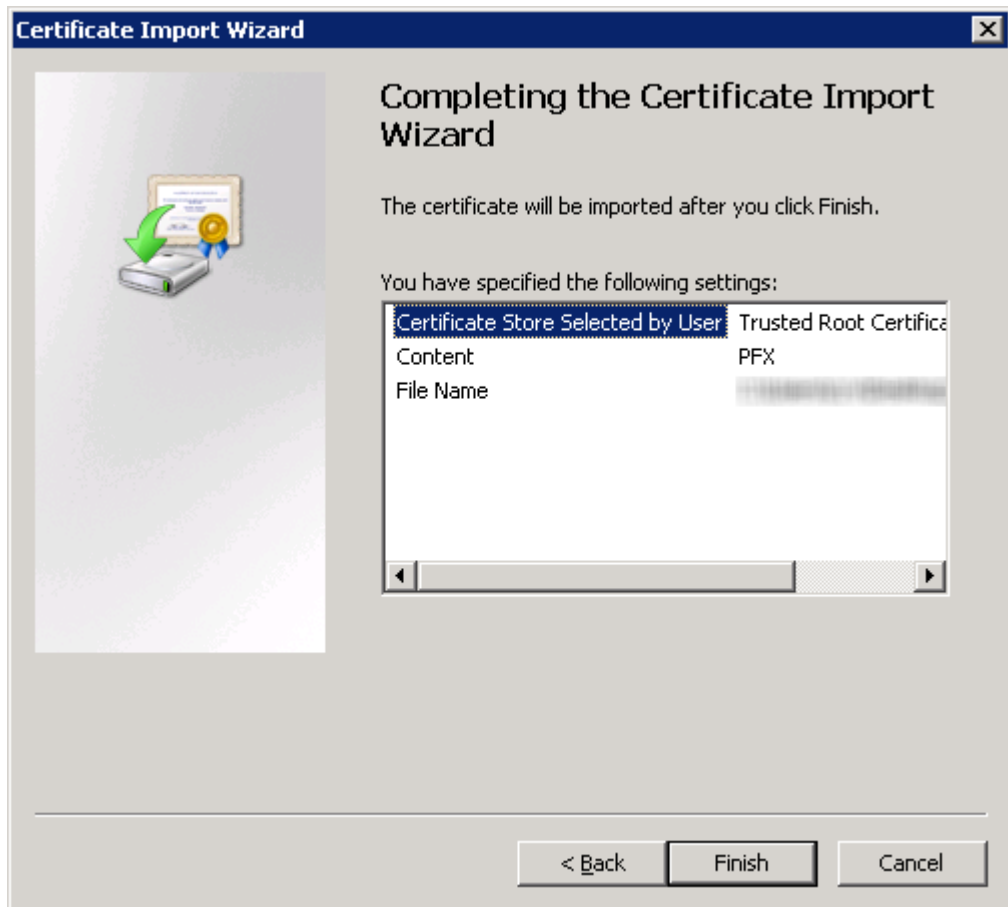
- e. Enter the password that was given when the file was exported for the *.pfx* file.
- f. Select **Mark this key as exportable** and **Include all extended properties**, then click **Next**.



- g. Select **Place all certificates in the following store**, then click **Next**.



- h. Click **Finish**.



You may need to refresh the mmc to see the imported certificate.



Make sure to complete this step for each certificate.

11. Close the MMC console.
12. Start the Qlik Sense services. If the services are started manually, start them in the following order:



If you are restoring the certificates as part of the Restoring a Qlik Sense site (page 203) procedure, do not start the Qlik Sense services.

- a. Qlik Sense Service Dispatcher(QSD)
- b. Qlik Sense Repository Service (QRS)
If the user running Qlik Sense services is not local administrator on the machine, you need to start Repository.exe from an elevated command prompt using the -bootstrap parameter.
Services (page 30)
- c. Qlik Sense Proxy Service (QPS), Qlik Sense Engine Service (QES), Qlik Sense Scheduler Service (QSS), and Qlik Sense Printing Service (QPR) in no specific order

The start-up order is important. During start-up the QRS must be able to contact the Qlik License Service, which is managed by the QSD. The other services are dependent on the QRS. The QSD must therefore be running when the QRS is started.

5.6 Backing up a Qlik Sense site

Backing up a Qlik Sense site includes backing up the following:

- Repository database (QSR): The database contains all configuration data for the site
- SenseServices database (if you have a linked cloud environment)
- QSMQ database (if you have a linked cloud environment)
- Licenses (optional, license assignments restored from License Backend Server after applying license key - Signed License Key)
- Log data: Local log files
- The file share: The shared folder in that contains application data, such as data models used in the Qlik Sense apps, and QVD files

To restore your Qlik Sense deployment you will also need a back up of your Qlik Sense certificates. For more information, see *Backing up certificates* (page 182).

You must perform this backup procedure on each of the nodes that host the components listed above.



Rim nodes maintain local log files that may be worth backing up in order to identify and investigate issues. It may also be worth backing up any general operating system data that may be required.

Do the following:

1. Stop all Qlik Sense services except the Qlik Sense Repository Database (QRD), on every node in your deployment.
2. Make a backup of the repository database by creating a database dump file:
 - a. Open a Command Prompt in Microsoft Windows.
 - b. Navigate to the location where the PostgreSQL repository database is installed.



If your deployment includes a local database on the central node that was installed using the Qlik Sense setup program, the location will be:

%ProgramFiles%\Qlik\Sense\Repository\PostgreSQL\<database version>\bin.



If you installed PostgreSQL manually, the location will be:

%ProgramFiles%\PostgreSQL\<database version>\bin.

- c. Run the following commands:

```
pg_dump.exe -h localhost -p 4432 -U postgres -b -F t -f "c:\QSR_backup.tar" QSR
pg_dump.exe -h localhost -p 4432 -U postgres -b -F t -f "c:\SenseServices_
backup.tar" SenseServices
```

5 Backup and restore Qlik Sense Enterprise on Windows

```
pg_dump.exe -h localhost -p 4432 -U postgres -b -F t -f "c:\QSMQ_backup.tar" QSMQ
pg_dump.exe -h localhost -p 4432 -U postgres -b -F t -f "c:\Licenses_backup.tar"
Licenses
```

If you are prompted for the PostgreSQL super user password, enter the password that was created during the Qlik Sense setup.



To avoid being prompted for the password (for example, if you want to automate the Qlik Sense backup process), you can use the `pgpass` functionality in PostgreSQL. See the PostgreSQL documentation for more information.

3. Make a backup of all of the content in the file share.
4. Make a backup of any locations where content that supports the Qlik Sense environment may be kept (for example, QVD files created by load scripts).
5. Restart the Qlik Sense services.

Backing up the Qlik Sense Repository Database after uninstalling Qlik Sense



We recommend creating your database dump file before you uninstall Qlik Sense.

If you uninstall Qlik Sense before creating the database dump file, do the following:

1. Copy the PostgreSQL folder from `%ProgramData%\Qlik\Sense\Repository\PostgreSQL` to a temporary location outside of the `%ProgramData%` folder.
2. Download and install PostgreSQL version 12.x from the [PostgreSQL](#) website. See: *Installing and configuring PostgreSQL* (page 132).
3. Open a Command Prompt in Microsoft Windows.



The `pg_ctl.exe` command should not be run as an administrator.

4. Navigate to the location where the PostgreSQL repository database is installed.



*If your deployment includes a local database on the central node that was installed using the Qlik Sense setup program, the location will be:
`%ProgramFiles%\Qlik\Sense\Repository\PostgreSQL\<database version>\bin.`*



*If you installed PostgreSQL manually, the location will be:
`%ProgramFiles%\PostgreSQL\<database version>\bin.`*

5. Run the following commands:
 - a. `pg_ctl.exe start -w -D "C:\SenseDB\12.x"`
 - b. `set PGUSER=postgres`
 - c. `set PGPASSWORD=password`

5 Backup and restore Qlik Sense Enterprise on Windows

- d. `pg_dumpall.exe > [<path to dump file>]`
- e. `pg_ctl.exe stop -w -D "C:\SenseDB\12.x"`

If you are prompted for the PostgreSQL super user password, enter the password that was created during the Qlik Sense setup.



To avoid being prompted for the password (for example, if you want to automate the Qlik Sense backup process), you can use the `pgpass` functionality in PostgreSQL. See the PostgreSQL documentation for more information.

5.7 Restoring a Qlik Sense site

Consider the following when restoring a site:

- Qlik Sense software
- Repository database (QSR): The database contains all configuration data for the site.
- SenseServices database (if you have a linked cloud environment)
- QSMQ database (if you have a linked cloud environment)
- Licenses (optional, license assignments restored from LBS after applying license key - SGK)
- Certificates for the Qlik Sense services: The certificates are used to encrypt the traffic between the services and the users. Make sure to backup the certificates in order not to lose any encrypted data (for example, passwords for data connections).
- Log data
- Application data: The data models in the Qlik Sense apps.
- Any content that supports the apps (for example, QVD files)
- If you want to restore the site to a central node with the same hostname, see *Restoring a Qlik Sense site to a machine with the same hostname* (page 203).
If you want to restore the site to a central node with a new hostname, see *Restoring a Qlik Sense site to a machine with a different hostname* (page 205).

Restoring a Qlik Sense site to a machine with the same hostname

When performing the procedure below you must log in using an account that had the Root Admin role when the site was backed up. If you log in using a local admin account and the machine name is different, your permissions will not follow through.

Do the following:

1. Restore the certificates used to secure the Qlik Sense services.
Restoring certificates (page 191)
2. Install Qlik Sense on the computer where you plan to restore.

5 Backup and restore Qlik Sense Enterprise on Windows



Make sure to deselect **Start the Qlik Sense services when the installation has completed** during the installation setup.

3. Start the Qlik Sense Repository Database (QRD).
4. Restore the repository database:
 - a. Open a Command Prompt with administrator privileges in Microsoft Windows.
 - b. Navigate to: `cd "%ProgramFiles%\Qlik\Sense\Repository\PostgreSQL\<database version>\bin"`
 - c. Run the following command to restore the repository database on a clean server:
`pg_restore.exe -h localhost -p 4432 -U postgres -d QSR "c:\QSR_backup.tar"`
`pg_restore.exe -h localhost -p 4432 -U postgres -d SenseServices "c:\SenseServices_backup.tar"`
`pg_restore.exe -h localhost -p 4432 -U postgres -d QSMQ "c:\QSMQ_backup.tar"`
`pg_restore.exe -h localhost -p 4432 -U postgres -d Licenses "c:\Licenses_backup.tar"`



You may need to adjust the path `"c:\QSR_backup.tar"` depending on where you backed up your database dump file.

If running these commands on a server where a repository database may have been installed previously, you may get the following error messages. Note the name of affected database and adjust corrective commands accordingly. For example, for the "QSR" database you might get:

- `pg_restore: [archiver (db)] connection to database "QSR" failed: FATAL: database "QSR" does not exist`

If you get this error, from the same location run following command:

```
createdb -h localhost -p 4432 -U postgres -T template0 QSR
```

Then run the restore command again.

- `pg_restore: [archiver (db)] Error while PROCESSING TOC`
`pg_restore: [archiver (db)] Error from TOC entry 185; 1259 134513 TABLE Apps`
`qliksenserepository`
`pg_restore: [archiver (db)] could not execute query: ERROR: relation "Apps"`
`already exists`

If you get many errors like above, stop the restoration process and from the same location run following two commands, one after another:

```
dropdb -h localhost -p 4432 -U postgres QSR
```

```
createdb -h localhost -p 4432 -U postgres -T template0 QSR
```

Then run the restore command again.

5. Restore log and application data to the file share used for storage of log and application data.
6. Restore any supporting content to its original location as required.
7. Start the Qlik Sense services. If the services are started manually, start them in the following order:
 - a. Qlik Sense Service Dispatcher (QSD)
 - b. Qlik Sense Repository Service (QRS)
If the user running Qlik Sense services is not local administrator on the machine, you need to start Repository.exe from an elevated command prompt using the `-bootstrap` parameter.
Services (page 30)
 - c. Qlik Sense Proxy Service (QPS), Qlik Sense Engine Service (QES), Qlik Sense Scheduler Service (QSS), and Qlik Sense Printing Service (QPR) in no specific order

5 Backup and restore Qlik Sense Enterprise on Windows

The start-up order is important. During start-up the QRS must be able to contact the Qlik License Service, which is managed by the QSD. The other services are dependent on the QRS. The QSD must therefore be running when the QRS is started.

8. Try to access the QMC or the Hub to verify that the migration has been successful. Also, from the Qlik Management Console reload the monitoring apps to verify that your certificates have been installed correctly.



If restoring a multi-node site with central node on a machine with a different hostname, all rim nodes must be reset, that is, you need to remove them and then add them again.

Restoring a Qlik Sense site to a machine with a different hostname

You can restore a Qlik Sense site to a machine with a host name that is different from the site that you backed up.

Perform the following steps on the target server machine, where you want to restore Qlik Sense.

Do the following:

1. Restore the certificates used to secure the Qlik Sense services.
Restoring certificates (page 191)
2. Install Qlik Sense on the computer where you plan to restore.



*Make sure to deselect **Start the Qlik Sense services when the installation has completed** during the installation setup.*

3. Start the Qlik Sense Repository Database (QRD).
4. Restore the repository database:
 - a. Open a Command Prompt with administrator privileges in Microsoft Windows.
 - b. Navigate to: `cd "%ProgramFiles%\Qlik\Sense\Repository\PostgreSQL\<database version>\bin"`
 - c. Run the following command to restore the repository database on a clean server:
`pg_restore.exe -h localhost -p 4432 -U postgres -d QSR "c:\QSR_backup.tar"`
`pg_restore.exe -h localhost -p 4432 -U postgres -d SenseServices "c:\SenseServices_backup.tar"`
`pg_restore.exe -h localhost -p 4432 -U postgres -d QSMQ "c:\QSMQ_backup.tar"`
`pg_restore.exe -h localhost -p 4432 -U postgres -d Licenses "c:\Licenses_backup.tar"`



You may need to adjust the path "c:\QSR_backup.tar" depending on where you backed up your database dump file.

If running these commands on a server where a repository database may have been installed previously, you may get the following error messages. Note the name of affected database and adjust corrective commands accordingly. For example, for the "QSR" database you might get:

5 Backup and restore Qlik Sense Enterprise on Windows

- `pg_restore: [archiver (db)] connection to database "QSR" failed: FATAL: database "QSR" does not exist`

If you get this error, from the same location run following command:

```
createdb -h localhost -p 4432 -U postgres -T template0 QSR
```

Then run the restore command again.

- `pg_restore: [archiver (db)] Error while PROCESSING TOC`
`pg_restore: [archiver (db)] Error from TOC entry 185; 1259 134513 TABLE Apps`
`qliksenserepository`
`pg_restore: [archiver (db)] could not execute query: ERROR: relation "Apps"`
`already exists`

If you get many errors like above, stop the restoration process and from the same location run following two commands, one after another:

```
dropdb -h localhost -p 4432 -U postgres QSR
```

```
createdb -h localhost -p 4432 -U postgres -T template0 QSR
```

Then run the restore command again.

5. Restore log and application data to the file share used for storage of log and application data.
6. Restore any supporting content to its original location as required.
7. To launch Qlik Sense with the new hostname:
 - a. Open a Command Prompt with administrator privileges in Microsoft Windows.
 - b. Change the directory to the Repository installation path
Default path: `"C:\Program Files\Qlik\Sense\Repository"`
 - c. Execute the following command:
`repository.exe -bootstrap -standalone -restorehostname`




The parameter '-standalone' means that Repository runs as a normal executable process (as opposed to running as a service, and registering in Windows Service Manager).

- d. Start the Qlik Sense Service Dispatcher when the following message is displayed. If the Qlik Sense Service Dispatcher is not running, the hostname update will not complete.
`[INFO] Entering main startup phase...`
When the command has completed successfully check for errors in the logs and the following message is displayed:
`Bootstrap mode has terminated. Press ENTER to exit..`
8. Start the Qlik Sense services. If the services are started manually, start them in the following order:
 - a. Qlik Sense Repository Service (QRS)
If the user running Qlik Sense services is not local administrator on the machine, you need to start `Repository.exe` from an elevated command prompt using the `-bootstrap` parameter.
Services (page 30)
 - b. Qlik Sense Proxy Service (QPS), Qlik Sense Engine Service (QES), Qlik Sense Scheduler Service (QSS), and Qlik Sense Printing Service (QPR) in no specific orderThe start-up order is important. During start-up the QRS must be able to contact the Qlik License Service, which is managed by the QSD. The other services are dependent on the QRS. The QSD must therefore be running when the QRS is started.
 9. Try to access the QMC or the Hub to verify that the migration has been successful. Also, from the Qlik Management Console reload the monitoring apps to verify that your certificates have been installed correctly.

5 Backup and restore Qlik Sense Enterprise on Windows



If restoring a multi-node site with central node on a machine with a different hostname, all rim nodes must be reset, that is, you need to remove them and then add them again.

For additional information on changing the hostname after installation, see  [Qlik Sense: Change hostname \(and certificates\) after an installation](#).

6 Qlik Sense Enterprise on Windows security

Security in Qlik Sense Enterprise on Windows consists of the following:

- **Protection of the platform**
How the Qlik Sense platform itself is protected and how it communicates and operates.
- **Authentication**
Who is the user and how can the user prove it? Qlik Sense uses standard authentication protocols (for example, Integrated Windows Authentication), HTTP headers, and ticketing to authenticate every user requesting access to data.
- **Authorization**
What does the user have access to? Authorization is the procedure of granting or denying users access to resources.
- **Auditing**
The Qlik Sense platform tracks changes in the repository database, provides comprehensive audit and security logging, and monitors applications.
- **Confidentiality**
Qlik Sense protects confidentiality by:
 - encrypting network connections with Transport Layer Security (TLS)
 - leveraging the operating system file system and server access controls to protect content on Qlik Sense nodes
 - protecting memory using operating system controls
 - securing application access at the resource level
 - encrypting sensitive information (e.g. passwords and data connection strings) with AES-256 encryption
 - protecting app data using data reduction and data encryption
- **Integrity**
Operating system controls like the file system are leveraged to provide integrity by protecting data at rest, encrypting sensitive information, and preventing data write back to the source system.
- **Availability**
Qlik Sense deployed in a multi-node environment is designed for resiliency and reliability.

6.1 Certificates

A certificate is a data file that contains keys that are used to encrypt communication between a client and a server in a domain. Certificates also confirm that the domain is known by the organization that issued the certificate. A certificate includes information about the keys, information about the identity of the owner, and the digital signature of an organization that has verified that the content of the certificate is correct. The pair of keys (public and private keys) are used to encrypt communication.

Qlik products use standard TLS certificates when they communicate with each other.

- The Qlik Sense Proxy Service uses front-end certificates to establish HTTPS connections to the Qlik Sense Hub and Management Console. For more information, see *Certificates used by the Qlik Sense Proxy Service (page 209)*.

- Qlik products use internal certificates for secure communication between components installed on separate computers. For more information, see *Certificate trust* (page 217).

The organization that issues the certificate, the certificate authority (CA), is said to "sign" the certificate. You can arrange to get certificates from a certificate authority, to show your domain is known. You can also issue and sign your own certificates (self-signed certificates).

Certificate revocation check

A revocation check is a process used to determine whether a digital certificate that has been previously issued and is in use should continue to be trusted. The status of a certificate can be verified by checking against a Certificate Revocation List (CRL) or an Online Certificate Status Protocol (OCSP) responder.

Front-end certificates

For a self-signed or custom root certificate authority, Qlik Sense installations use CRL and OSCP extensions. These extensions rely on public infrastructure, specifically CRL and OSCP endpoints, which must be embedded in the certificates. Qlik Sense does not provide this infrastructure, but you are free to embed the extensions in certificates and maintain the associated CRL/OSCP infrastructure.

For CA certificates issued by trusted certificate authorities (for example, DigiCert), the extensions are based on infrastructure managed by the respective certificate authority. Modern browsers support Certificate Transparency (CT)—a security measure for SSL/TLS certificates used in web encryption. It involves public logging of all issued certificates, making it possible to detect and prevent unauthorized or fraudulent certificates. For more information, see [What is Certificate Transparency?](#).

Internal certificates

Qlik internal certificates (trust) do not use CRL or OSCP for certificate chain validation.

Certificates used by the Qlik Sense Proxy Service

The Qlik Sense Proxy Service uses certificates to establish HTTPS connections to the Qlik Sense Hub and Management Console.

Common errors

As part of the Certificate Transparency (CT) framework employed by modern web browsers for enhanced security, error messages related to certificates and potential communication blocks are displayed to inform users about the legitimacy of a site. For more information, see [What is Certificate Transparency?](#).

Some common errors are related to the certificate authority. For example, if there is no certificate authority or if the certificate has expired, the default level of security in most browsers will stop communication with a message about "unsigned certificates", "expired certificates", or similar terms. If your security administrators know that the certificate is still good, you can create an exception so the error is ignored for that certificate.

Other common errors are related to how the domain is named. For example, `companyname.com` is a different domain from `www.companyname.com`, and `localhost` is a different domain from a server name. A fully qualified domain name is an unambiguous name for a domain. For example, a server at `companyname.com` might be named `mktg-SGK`, and can be referred to that way, but the fully qualified domain name is `mktg-SGK.companyname.com`.

Encryption and keys

The kind of encryption used in certificates in Qlik products requires a pair of keys (asymmetric encryption). One key, the public key, is shared. The other key, the private key, is used only by the owner.

PEM is an ASCII text format for public certificates. It is portable across platforms.

You can get certificates and key pairs from certificate authorities or you can generate them. To get a certificate signed, you will need to also generate a signing request.

Adding third-party certificates

By default, Qlik Sense uses a self-signed certificate to enable HTTPS access across both the Hub (https://<your_sense_server>/hub) and the Management Console (https://<your_sense_server>/qmc). However, self-signed certificates cannot be validated or trusted by web browsers and tend to prompt a warning message.

To establish a secure HTTPS connection, the browser must trust the SSL/TLS certificate installed on the server. In the case of self-signed certificates, the signing certificate authority (CA) is not trusted, hence no certificates generated by the certificate authority are trusted.


For the installation of a trusted certificate to be used with the Qlik Sense Hub and Management Console, an additional signed certificate is required, along with a brief (5-minute) downtime of the Qlik Sense Proxy. For more details, including a video, see [👉 How to change the certificate used by the Qlik Sense Proxy to a custom third party certificate](#) on Qlik Community.

Choosing a suitable certificate

There are three potential types of certificates:

- A certificate purchased from and signed by a trusted certificate authority like VeriSign, Thawte, or Geotrust.
- A certificate provided and signed by your own enterprise certificate authority.
- A self-signed certificate created by various applications, such as Microsoft IIS. These certificates are typically recommended for testing purposes only. They might unintentionally lead users to disregard browser warnings, which is not desirable.

The certificate must:

- Meet the requirements listed in [👉 Qlik Sense Enterprise on Windows: Compatibility information for third-party SSL certificates to use with HUB/QMC](#).
- Contain a private key.
Verify that the certificate comes bundled with a private key.
 **You have a private key that corresponds to this certificate.**
- Have an up-to-date valid from/valid to date range.
- Be signed by an officially recognized certificate authority that is recognized and pre-configured by the operating system or the web browser.

The certificate authority will have instructions for where to get a certificate and how to do a Certificate Signing Request (CSR). To obtain a self-signed certificate or one from your corporation's certificate authority, a local administrator can provide the certificate to you. In either case, generating a CSR is necessary to submit to your certificate authority. Various tools are available for this task, such as Microsoft's [certreq](#). More information about Certificate Signing Requests can be found in this article: [What is a CSR \(Certificate Signing Request\)?](#)

Installing the certificate

When you have the certificate, the next step is to install it and activate it in Qlik Sense. This includes importing the certificate and retrieving the certificate thumbprint to provide it to the Qlik Sense Proxy.

Do the following:

1. On the Qlik Sense node running the Qlik Sense Proxy, log in with the user running the Sense services.
2. Import the certificate.

If the certificate provided from your CA was saved in .pfx format:

- Double click the certificate file, and then follow the prompts to import the certificate into the Personal store.

To import the certificate manually:

1. Open the Microsoft Management Console (mmc.exe) on the proxy node.
 2. Go to **File > Add/Remove Snap-in**.
 3. Select **Certificates** and click **Add**.
 4. Select **Computer account** and click **Next**.
 5. Select **Local computer**, click **Finish**, and then click **OK**.
 6. Under **Certificates (Local Computer)**, select **Personal**.
 7. Go to **Actions > All Tasks > Import**.
 8. Browse to locate your certificate file, and then follow the prompts to import the certificate including the private key to the Personal store.
3. Verify that the certificate was correctly installed.
 1. Open the Microsoft Management Console (mmc.exe) on the proxy node.
 2. Under **Certificates (Local Computer)**, select **Personal**.
 3. Verify that the new certificate is imported into **Certificates (Local Computer)/Personal > Personal > Certificates** and that it contains a private key.
 4. Double-click the certificate to open the **Certificate** dialog.
 5. On the **Certification Path** tab, confirm that the **Certificate status** says "This certificate is OK".
 4. Copy the certificate thumbprint.
 1. In the Microsoft Management Console, double-click the certificate to open the **Certificate** dialog.
 2. On the **Details** tab, locate **Thumbprint** in the list.
 3. Copy the thumbprint and paste it somewhere, for example, in a notepad, to be used in the next step.
 5. Configure the Qlik Sense Proxy Service.

1. Open the Qlik Management Console (QMC).
2. Go to **Proxies**.
3. Double-click the proxy you want to use.
4. On the **Edit proxy** page, under **Security**, paste the certificate thumbprint in the **SSL browser certificate thumbprint** field.
5. Click **Apply**.
The Qlik Sense Proxy Service will now restart. During the restart, Windows API calls are used to correctly bind the new certificate to its SSL ports.
6. Verify that the certificate was accepted.
When you open the Qlik Sense Hub or Qlik Management Console, the certificate will be shown in the browser.
 1. In Google Chrome, for example, you can verify the certificate by clicking on the padlock icon to the left of the URL. (The appearance may vary depending on the web browser you are using.)
 2. Make sure that the information displayed about the certificate matches the properties of the installed certificate, which you can find in the Microsoft Management Console on the **General** tab in the **Certificate** dialog.

6.2 Protecting the platform

The security in Qlik Sense does not depend only on the Qlik Sense software. It also relies on the security of the environment that Qlik Sense operates in. This means that the security of, for example, the operating system and the cryptographic protocols (such as TLS/SSL) has to be set up and configured to provide the security needed for Qlik Sense.

Network security

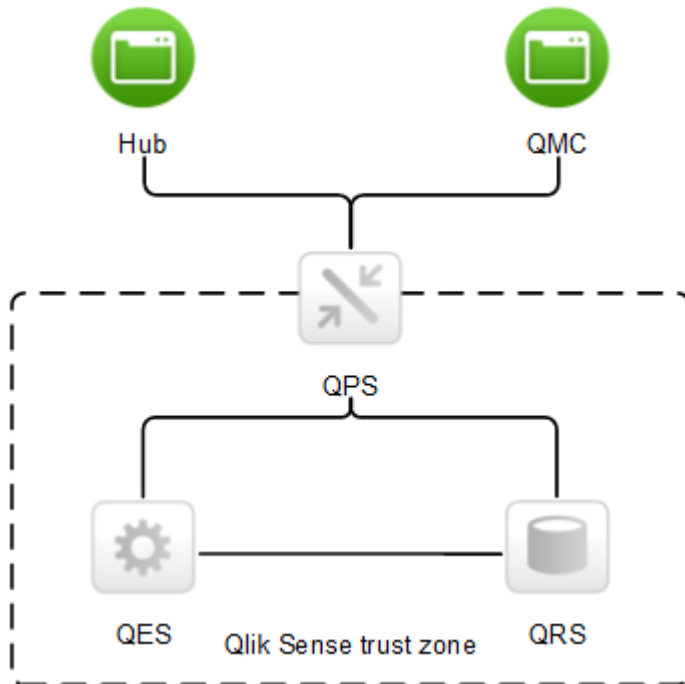
For all Qlik Sense components to communicate with each other in a secure way, they need to build trust.

In Qlik Sense, all communication between the Qlik Sense services and clients is based on web protocols. The web protocols use Transport Layer Security (TLS) for encryption and exchange of information and keys and certificates for authentication of the communicating parties.

TLS provides a way to build encrypted tunnels between identified servers or services. The parties that communicate are identified using certificates. Each tunnel needs two certificates; one to prove to the client that it is communicating with the right server and one to prove to the server that the client is allowed to communicate with the server.

So, how to make sure that the certificates are from the same Qlik Sense trust zone? All certificates that belong to a trust zone are signed with the same signature. If the signature exists in the certificate, it is accepted as proof that the certificate belongs to the trust zone.

When the protected tunnels and the correct certificates are in place, the Qlik Sense services have a trust zone to operate within. Within the trust zone, only services that belong to the specific Qlik Sense site can communicate with each other.

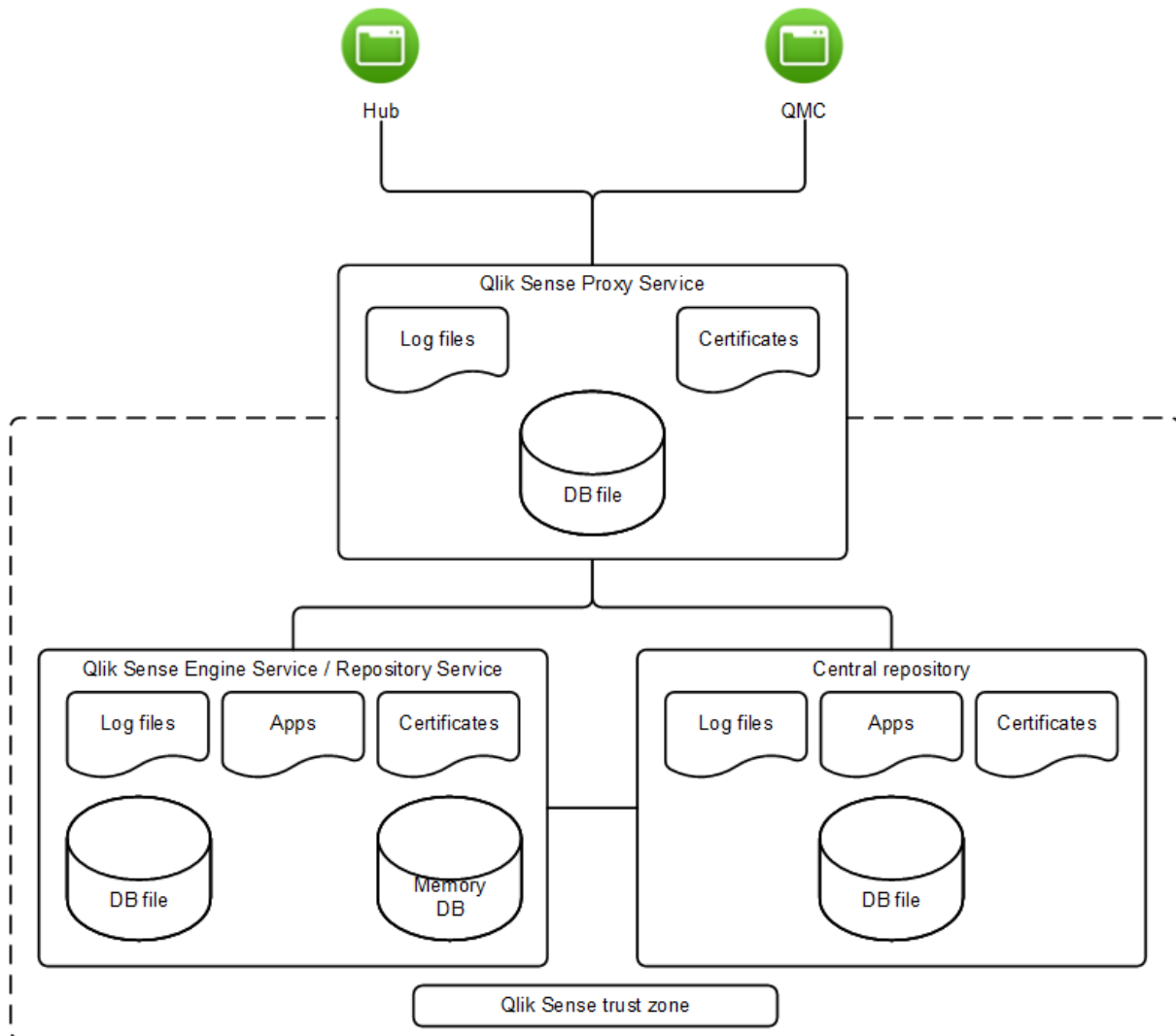


The Qlik Sense clients are considered to be outside of the Qlik Sense trust zone because they often run on less trusted end-user devices. The Qlik Sense Proxy Service (QPS) can bridge the two zones and allow communication between the clients and the Qlik Sense services, if the user is authenticated to the system.

TLS-protected tunnels can be used to secure the communication between the Qlik Sense clients and the QPS. As the clients are outside of the Qlik Sense trust zone, the communication between the clients and the QPS uses a certificate with a different signature than the one used within the trust zone.

Server security

Qlik Sense uses the server operating system to gain access to resources. The operating system provides a security system that controls the use of the server resources (for example, storage, memory, and CPU). Qlik Sense uses the security system controls to protect its resources (for example, files, memory, processes, and certificates) on the server.



Through the use of access control, the security system grants access to Qlik Sense files (for example, log files, database files, certificates, and apps) only to certain users on the server.

The security system also protects the server memory, so that only authorized processes are allowed to write to the Qlik Sense part of the memory.

In addition, the security system is responsible for assigning users to processes. This is used to restrict who is allowed to interact with the Qlik Sense processes on the server. The processes are also restricted in terms of which parts of the operating system they are allowed to access.

So, by using the controls in the security system, a secure and protected environment can be configured for the Qlik Sense processes and files.

Process security

Each process executes in an environment that poses different threats to the process. In this layer of the security model, the focus is on ensuring that the software is robust and thoroughly analyzed from a security perspective.

Rugged software

For software to be considered as rugged, it must cope with all potential threats to the confidentiality, integrity, and availability of the information, and be robust when used in ways not anticipated.

Several mitigating actions have been implemented in the Qlik Sense software in order to make it rugged:

- Authorization of communication using certificates
- Validation of all external data that is sent to the system
- Encoding of content to avoid injection of malicious code
- Use of protected memory
- Encryption of data
- Audit logging
- Use of checksums
- Isolated execution of external components
- Escaping of SQL data

Threat analysis

To ensure that the Qlik Sense software is secure and rugged, threat analysis of the design has been performed as part of the development process. The following threat areas, often abbreviated as STRIDE, have been covered:

- **S**poofing
- **T**ampering
- **R**epudiation
- **I**nformation disclosure
- **D**enial of service
- **E**levation of privilege

In addition to the threat analyses, exploratory security testing has also been performed on the Qlik Sense software.

App security

The major components of the Qlik Sense app security are:

- **Access control system:** The access control system grants users access to the resources in Qlik Sense. See [Access control](#)
- **Data reduction:** The data reduction functionality is based on the concept of section access, which is a way to dynamically change which data a user can view. This makes it possible to build apps that can be used by many users, but with different data sets that are dynamically created based on user information. The reduction of data is performed by the Qlik Sense Engine Service (QES). See [Managing data security with Section Access](#)
- **Data encryption:** Sensitive data in QVF and QVD files is encrypted with customer supplied key pairs which allows you to control who gets access to your data. The encryption keys are managed through certificates, that must be stored in a certificate store for the user running the Qlik Sense Engine Service

(QES).

See [Data encryption](#)

Using these components, the resources and data (that is, the content) consumed by the Qlik Sense users can be secured.

6.3 Authentication

All authentication in Qlik Sense is managed by the Qlik Sense Proxy Service (QPS). The QPS authenticates all users regardless of Qlik Sense client type. This means that the QPS also authenticates users of the Qlik Management Console (QMC).



In Qlik Sense, authentication and authorization are two distinct, unconnected actions. In addition, the sources of information used for authentication do not have to be the same as for authorization, and the other way around.

Qlik Sense always asks an external system to verify who the user is and if the user can prove it. The interaction between Qlik Sense and the external identity provider is handled by authentication modules.

For a module to communicate with Qlik Sense, it has to be trusted. Transport Layer Security (TLS) and certificate authentication are used to authorize external components for communication with Qlik Sense.

In Qlik Sense, the authentication of a user consists of three distinct steps:

1. Authentication module: Get the user identity and credentials.
2. Authentication module: Request an external system to verify the user identity using the credentials.
3. Transfer the user to Qlik Sense using the Ticket API, the Session API, headers, SAML, JWT, or OIDC.

The first two steps are always handled by the authentication module. It is up to the authentication module to verify the user in an appropriate way.


The third step can be performed in the following ways:

- Using the Ticket API, which transfers the user and the user's properties using a one-time ticket.
- Using the Session API, whereby an external module can transfer web sessions that identify the user and the user's properties to Qlik Sense.
- Using headers, with which a trusted system can transfer the user using HTTP headers. This is a common solution for integrating with single sign-on (SSO) systems.
- Qlik Sense can be configured to allow anonymous users (using, for example, SAML).



Logging in to Qlik Sense Enterprise on Windows is not possible using the default internal accounts identified by names starting with 'sa_' and having user directory INTERNAL (for example, sa_converter, sa_engine, and sa_proxy). These accounts are reserved for specific system functions and are not intended for direct user access or interactive logins.

See also:

 [Protecting the platform \(page 212\)](#)

Default authentication module

After a default installation of Qlik Sense, the Qlik Sense Proxy Service (QPS) includes a module that handles authentication of Microsoft Windows users. The module supports the use of Kerberos and NTLM.

If you want to use Kerberos authentication, you need to make sure that browsers that are used to access Qlik Sense are configured to support Kerberos.



The default authentication module requires that the proxy that handles the authentication is part of the Microsoft Windows domain.

Certificate trust

Qlik Sense uses certificates for authentication. A certificate provides trust between nodes within a site.

Certificate trust requirements

The requirements described in this section must be fulfilled for the certificate trust to function properly.

When using Transport Layer Security (TLS) in Microsoft Windows environments, the private key must be stored together with the certificate in the Windows certificate store. In addition, the account that is used to run the Qlik Sense services must have permission to access the certificate private key.

If you want to use TLS 1.2 authentication, you need to enable TLS 1.2 support in the Windows registry of the server machine. You should consider the impact of enabling TLS 1.2, as this is a global system setting.

Communication ports

To set up certificate trust, the Qlik Sense Repository Services (QRSs) require that the ports listed in the following table can be opened and used for communication. If any communication passes through a network firewall, the ports in the firewall must be opened and configured for the services.

Required ports

Port	Description
4570	Certificate password verification port, only used within multi-node sites by Qlik Sense Repository Services (QRSs) on rim nodes to receive the password that unlocks a distributed certificate. The port can only be accessed from localhost and it is closed immediately after the certificate has been unlocked. The communication is always unencrypted. This port uses HTTP for communication.

Port	Description
4444	Security distribution port, only used by Qlik Sense Repository Services (QRSs) on rim nodes to receive a certificate from the primary QRS on the central node. The communication is always unencrypted, but the transferred certificate package is password-protected. This port uses HTTP for communication.

Ports (page 42)

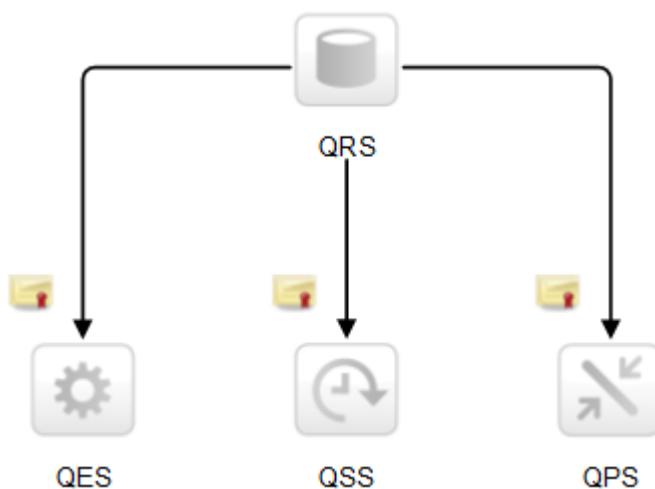
Unlocking distributed certificates

When adding a new rim node to a site, the distributed certificate needs to be unlocked.

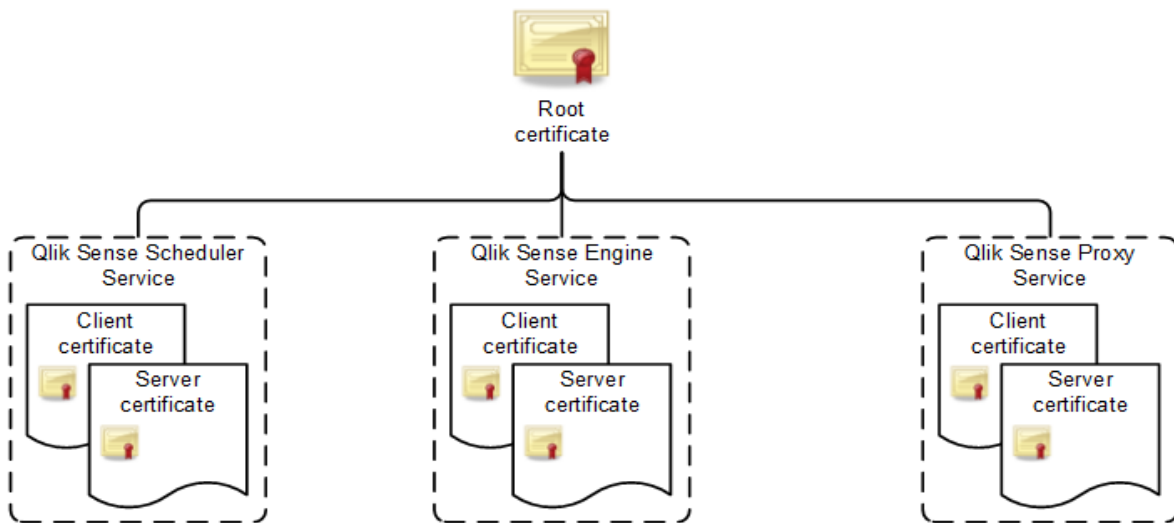
Certificate trust architecture

Certificates are used within a Qlik Sense site to authenticate communication between services that reside on different nodes. In addition, certificates can be used to build a trust domain between services that are located in different domains or areas (for example, internal networks, extranets, and Internet) without having to share a Microsoft Active Directory (AD) or other user directories.

The architecture is based on the primary Qlik Sense Repository Service (QRS) on the central node acting as the certificate manager or Certificate Authority (CA). The primary QRS creates and distributes certificates to all nodes within a site. The primary QRS is therefore an important part of the security solution and has to be managed from a secure location to keep the certificate solution secure.



The root certificate for the installation is stored on the central node in the site, where the primary QRS runs. All nodes with Qlik Sense services that are to be used within the site receive certificates signed with the root certificate when added to the primary QRS. The primary QRS (that is, the CA) issues digital certificates that contain keys and the identity of the owner. The private key is not made publicly available – it is kept secret by the nodes. The certificate enables the services in a Qlik Sense deployment to validate the authenticity of the other services. This means that the primary QRS is responsible for making sure that a service that is deployed on a node is a service within the site.



After the nodes have received certificates, the communication between the Qlik Sense services is encrypted using Transport Layer Security (TLS) encryption.

Confirming certificates using Microsoft Management Console

Certificates can be visually confirmed in the Microsoft Management Console (MMC) with the certificate snap-in added.

If the certificates have been properly deployed, they are available in the locations listed in the table.

Types of certificates

Certificate	Location
QlikClient	Certificates - Current User>Personal>Certificates
<full computer name>-CA	Certificates - Current User>Trusted Root Certification Authorities>Certificates
<full computer name>-CA	Certificates (Local Computer)>Trusted Root Certification Authorities>Certificates
<computer name>	Certificates (Local Computer)>Personal>Certificates

Certificate handling

This section describes how the certificates are handled when a Qlik Sense service starts.

Client certificate

This section describes how the primary Qlik Sense Repository Service (QRS) on the central node in a site handles the client certificate when a Qlik Sense service starts.

The client certificate is located in the following place in the Microsoft Windows certificate store:

Current User>Personal>Certificates

When a Qlik Sense service starts, the QRS searches the certificate store to see if there are any Qlik Sense certificates. Depending on the results of the search, the QRS does the following:

- If no client certificate is found, the QRS logs that no certificate was found.
- If only one client certificate is found, the QRS checks if it is valid. If the certificate is not valid, the QRS logs that an invalid certificate was found.
- If more than one client certificate is found, the QRS deletes all certificates. Duplicates are not allowed. In addition, the QRS logs the number of valid and invalid certificates that were found and deleted.

If certificates are found to be missing or invalid, you must run the QRS in bootstrap mode to recreate the certificates. For more information, see *Services (page 30)*.

Server certificate

This section describes how the primary Qlik Sense Repository Service (QRS) on the central node in a site handles the server certificate when a Qlik Sense service starts.

The server certificate is located in the following place in the Microsoft Windows certificate store:

Local Computer>Personal>Certificates

When a Qlik Sense service starts, the QRS searches the certificate store to see if there are any Qlik Sense certificates. Depending on the results of the search, the QRS does the following:

- If no server certificate is found, the QRS logs that no certificate was found.
- If only one server certificate is found, the QRS checks if it is valid. If the certificate is not valid, the QRS logs that an invalid certificate was found.
- If more than one server certificate is found, the QRS deletes all certificates. Duplicates are not allowed. In addition, the QRS logs the number of valid and invalid certificates that were found and deleted.

If certificates are found to be missing or invalid, you must run the QRS in bootstrap mode to recreate the certificates. For more information, see *Services (page 30)*.

Root certificate

This section describes how the primary Qlik Sense Repository Service (QRS) on the central node in a site handles the root certificate when a Qlik Sense service starts.

The root certificate is located in the following places in the Microsoft Windows certificate store:

Current User>Trusted Root Certification Authorities>Certificates

Local Computer>Trusted Root Certification Authorities>Certificates

When a Qlik Sense service starts, the QRS searches the certificate store to see if there are any Qlik Sense certificates. Depending on the results of the search, the QRS does the following:

- If no root certificate is found, the QRS logs that no certificate was found.
- If only one root certificate is found, the QRS checks if it is valid. If it is not valid, the QRS logs a fatal error that an invalid root certificate was found, which means that the service is shut down, and that

the administrator must manually delete any unwanted certificates. In addition, the QRS logs information about the certificates that are affected by this.

- If more than one root certificate is found, the QRS logs a fatal error that an invalid root certificate was found, which means that the service is shut down and that the administrator manually has to delete any unwanted certificates. In addition, the QRS logs information on the certificates that are affected by this.

If certificates are found to be missing or invalid, you must run the QRS in bootstrap mode to recreate the certificates. For more information, see *Services (page 30)*.



In order not to break any certificate trust between machines, the QRS does not remove any root certificates. It is up to the administrator to decide on what to do with invalid root certificates.

Invalid certificate

The definition of an invalid certificate is as follows:

- The operating system considers the certificate to be too old or the certificate chain is incorrect or incomplete.
- The Qlik Sense certificate extension (OID “1.3.6.1.5.5.7.13.3”) is missing or does not reflect the location of the certificate:
 - Current User/Personal certificate location: Client
 - Local Machine/Personal certificate location: Server
 - Local Machine/Trusted Root certificate location: Root
 - Current User/Trusted Root certificate location: Root
- The server, client, and root certificates on the central node do not have a private key that the operating system allows them to access.
- The server and client certificates are not signed by the root certificate on the machine.

Maximum number of trusted root certificates

When a Qlik Sense service starts, it checks the number of trusted root certificates on the machine where it is running. If there are more than 300 certificates on the machine, warning messages containing the following information are logged:

- There are too many root certificates for the service to trust.
- The Microsoft Windows operating system will truncate the list of certificates during the Transport Layer Security (TLS) handshake.

If the Qlik Sense root certificate (*<host-machine>-CA*) that the Qlik Sense client certificate belongs to is deleted from the list of certificates because of the truncation, the service cannot be authenticated.

To manually view the root certificates on a machine, open the Microsoft Management Console (MMC) and go to **Certificates (Local Computer)>Trusted Root Certification Authorities**.

Authentication solutions

Qlik Sense authentication can be managed with any of the following solutions:

- Ticket solution
- Session solution
- Header solution
- SAML
- JWT
- OIDC
- Anonymous users
- Configuring single sign-on (SSO) from Microsoft SQL (MSSQL) server

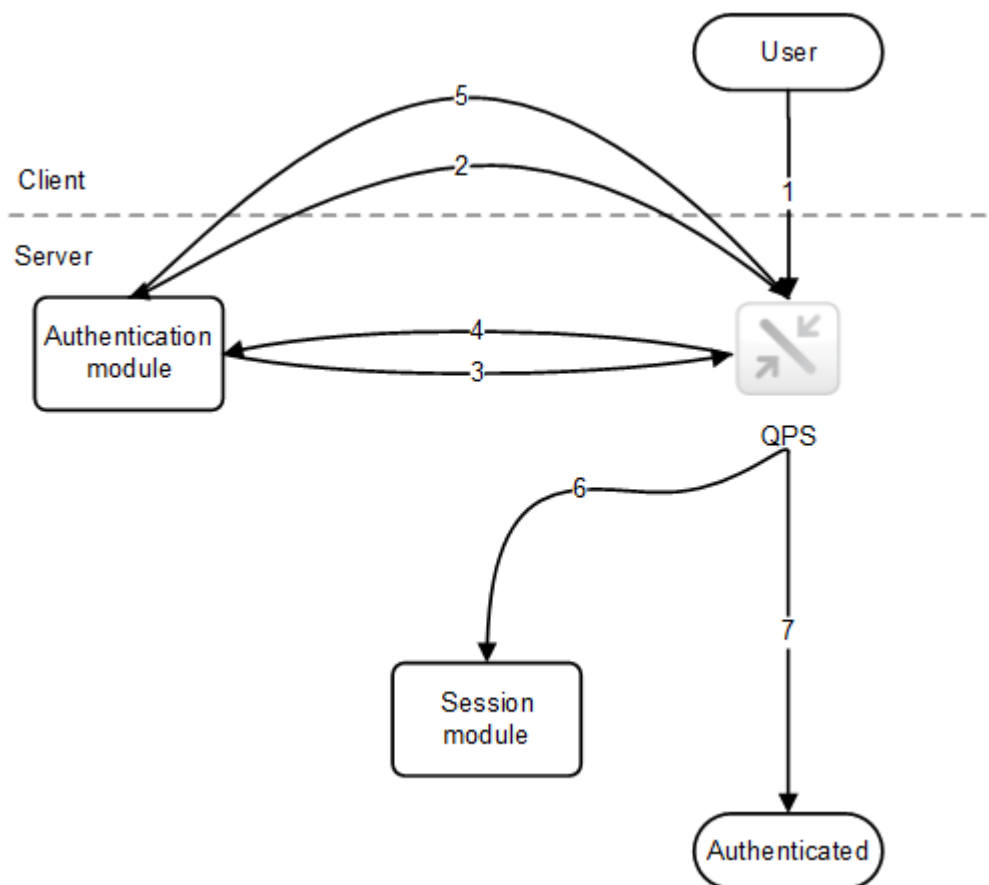
Ticket solution

The ticket solution is similar to a normal ticket. The user receives a ticket after having been verified. The user then brings the ticket to Qlik Sense and, if the ticket is valid, is authenticated. In order to keep the tickets secure, the following restrictions apply:

- A ticket is only valid for a short period of time.
- A ticket is only valid once.
- A ticket is random and therefore hard to guess.

All communication between the authentication module and the Qlik Sense Proxy Service (QPS) uses Transport Layer Security (TLS) and must be authorized using certificates.

The figure below shows a typical flow for authenticating a user with tickets.



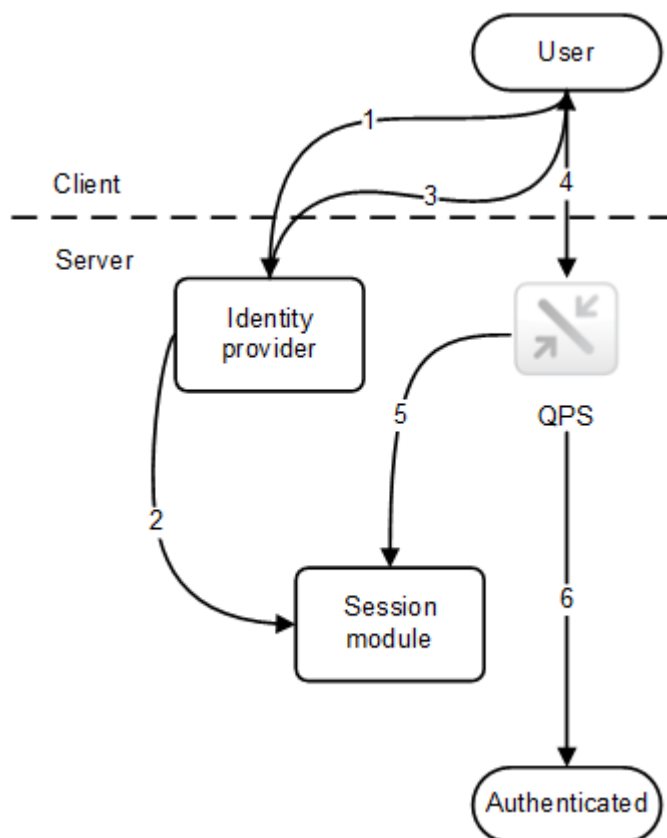
1. The user accesses Qlik Sense.
2. Qlik Sense redirects the user to the authentication module. The authentication module verifies the user identity and credentials with an identity provider.
3. Once the credentials have been verified, a ticket is requested from the QPS. Additional properties may be supplied in the request.
4. The authentication module receives a ticket.
5. The user is redirected back to the QPS with the ticket. The QPS checks that the ticket is valid and has not timed out.
6. A proxy session is created for the user.
7. The user is now authenticated.

Session solution

The session solution allows the Qlik Sense Proxy Service (QPS) to use a session from an external system to validate who the user is.

All communication between the authentication module and the QPS uses Transport Layer Security (TLS) and must be authorized using certificates.

The figure below shows a typical flow for authenticating a user using a session from an external system.



1. The user accesses the identity provider, which, for example, can be integrated into a portal. The identity provider gets the user identity and credentials and then verifies them. After that, the identity provider creates a new session.

2. The identity provider registers the session token with the Qlik Sense session module.
3. The identity provider sets the session token as a session cookie.
4. The user accesses the QPS to get content (for example, through an iframe in the portal).
5. The QPS validates the session to the session module.
6. If the session is valid and has not yet timed out, the user is authenticated.

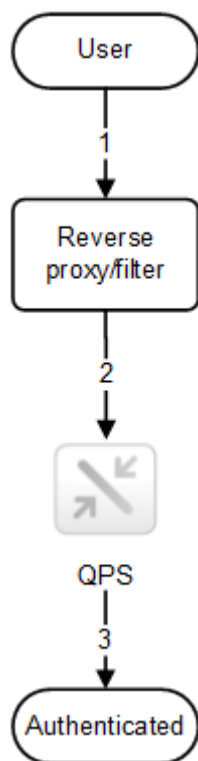


The name of the session cookie used by the authentication module can be configured in the Qlik Management Console (QMC).

Header solution

Header authentication is often used in conjunction with a Single Sign-On (SSO) system that supplies a reverse proxy or filter for authenticating the user.

The figure below shows a typical flow for authenticating a user using header authentication.



1. The user accesses the system and authenticates to the reverse proxy.
2. The reverse proxy injects the username into a defined HTTP header. The header must be included in every request to the Qlik Sense Proxy Service (QPS).
3. The user is authenticated.



For this solution to be secure, the end-user must not be able to communicate directly with the QPS but instead be forced to go through the reverse proxy/filter.



The reverse proxy/filter must be configured to preserve the host name, that is, the host header from the client must not be modified by the reverse proxy/filter.



The name of the HTTP header used for the user can be configured in the Qlik Management Console (QMC).

There are some restrictions on the values you can use for header name. For more information, see [Virtual proxies](#).

SAML

Security Assertion Markup Language (SAML) is an XML-based, open-standard data format for exchanging authentication and authorization data between parties (for example, between an identity provider and a service provider). SAML is typically used for web browser single sign-on (SSO).

How SAML works

The SAML specification defines three roles:

- Principal: Typically a user
- IdP: The identity provider
- SP: The service provider

The principal requests a service from the SP, which requests and obtains an identity assertion from the IdP. Based on the assertion, the SP decides whether or not to perform the service requested by the principal.

SAML in Qlik Sense

Qlik Sense supports SAML V2.0 by:

- Implementing an SP that can integrate with external IdPs
- Supporting HTTP Redirect Binding for SAML requests
- Supporting HTTP Redirect Binding and HTTP POST Binding for SAML responses
- Supporting SAML properties for access control of resources and data

Limitations:

- Qlik Sense does not support SAML message signature validation.

JSON Web Token (JWT)

JSON Web Token (JWT) is an open standard for secure transmission of information between two parties as a JavaScript Object Notation (JSON) object. JWT is used for authentication and authorization. Because JWT enables single sign-on (SSO), it minimizes the number of times a user has to log on to cloud applications and websites.

How JWT works

A JWT consists of three parts: a header, a payload, and a signature.

- The header usually consists of two parts: `type` (`typ`) and `algorithm` (`alg`). The algorithm is used to generate the signature.
- The payload is a JSON object that consists of the claims that you want to make. Claims are statements about an entity (usually the user) and additional metadata.
- The signature is used to verify the identity of the JWT sender and to ensure that the message has not been tampered with.

Authentication is performed by verifying the signature. If the signature is valid, access is granted to Qlik Sense.

Limitations

The following limitations exist:

- Encrypted JWTs are not supported.



When using HTTPS, all traffic, including JWTs, are encrypted during transport.

- Only the following signing algorithms are supported:
 - RS256 - RSA signature with SHA256
 - RS384 - RSA signature with SHA384
 - RS512 - RSA signature with SHA512

OpenID Connect

OpenID Connect (OIDC) is an authentication layer on top of OAuth 2.0, an authorization framework. OIDC enables single sign-on (SSO) to reduce the number of times a user has to log on to access websites and applications. OIDC can be configured for authentication with third-party products.

How OIDC works

OIDC is an open standard that uses JSON format authentication. OIDC uses the same components and architecture as OAuth, but to authenticate. Applications that use OIDC work with any identity provider that supports the authentication protocol. The protocol supports identity verification when a user tries to access a protected endpoint.

Workflow

1. A user accesses an application and is redirected to the OIDC identity provider for authentication and authorization.
2. The user logs in with the OIDC identity provider.
3. The OIDC identity provider sends a response to the application that a user has been authenticated and authorized.
4. The application requests user information from the OIDC identity provider.
5. The OIDC identity provider responds with the required user information.

Anonymous users

If anonymous use of Qlik Sense is allowed, users who are not authenticated are not automatically redirected to an authentication module. Instead, the user first gets anonymous access and is then, if the user chooses to sign in, redirected to the authentication module to supply user identity and credentials.

Configuring single sign-on (SSO) for Microsoft SQL (MS SQL) Server

If your database files access data from MS SQL Server, you can configure the host server to enable SSO. ODBC data source single sign-on permits clients to use one Windows authenticated login to access data in shared files.

To configure SSO for MS SQL Server, a Windows domain administrator must do the following:

- Create service principal names (SPN) in Active Directory
- Configure delegation for the Qlik Sense services administrator account
- Configure the Qlik Sense server for SSO
- Configure the MS SQL Server for SSO



The Microsoft SQL Server Connector in the Qlik ODBC Connector Package also supports SSO. If you are using the connector in the ODBC Connector Package, use the following configuration instructions: ODBC Connector: Configuring SSO for the Microsoft SQL Server connector.



The same Qlik Sense services administrator account used during the Qlik Sense (central node) installation must be used. If a different account is used, the Qlik Sense services administrator account must own the HTTP service principal. For more information, see User accounts (page 63).

Creating service principal names (SPN) in Active Directory

A service principal name (SPN) is a unique identifier of a service instance. SPNs are used during authentication to associate a service instance with a service logon account. This allows a client application to request that a service authenticate an account even if the client does not have the account name. A SPN always includes the name of the host computer on which the service instance is running, so a service instance might register a SPN for each name or alias of its host.

Before the authentication service can use a SPN to authenticate a service, the SPN must be registered on the account object that the service instance uses to log on. A given SPN can be registered on only one account. For Win32 services, a service installer specifies the logon account when an instance of the service is installed. The installer then composes the SPNs and writes them as a property of the account object in Active Directory Domain Services. If the account of a service instance changes, the SPNs must be re-registered under the new account.

When a client connects to a service, it locates an instance of the service, composes an SPN for that instance, connects to the service, and presents the SPN for the service to authenticate.

To set up SSO for MS SQL server, you must create SPNs for the Qlik Sense services administrator account.

Do the following:

1. Log on as a domain administrator.
2. Open an elevated command prompt.

3. Enter the following to create a SPN for the Qlik Sense services administrator:
setspn -A HTTP/<Qlik_Sense_server>:<port> <domain>\<Qlik_Sense_services_administrator>



The <Qlik_Sense_server> must be entered as the fully qualified domain name of the server.



The <Qlik_Sense_server> is the central node where the Qlik Sense is running.

4. Enter the following to create a SPN for the MS SQL Server services administrator:
setspn -A MSSQLSvc/<server_name>:<port> <domain>\<services_administrator>



The <server_name> must be entered as the fully qualified domain name of the server.

5. Enter the following commands to verify the result of your SPN setup:
 - a. setspn -L <domain>\<Qlik_Sense_services_administrator> to verify the Qlik Sense services administrator.
 - b. setspn -L <domain>\<MS_Sql_server_services_administrator> to verify the MS SQL Server services administrator.

Configuring delegation for the Qlik Sense services administrator account

Delegation allows a front-end service to forward client requests to a back-end service so that the back-end service can also impersonate the client. Impersonation is used to check whether a client is authorized to perform a particular action, while delegation is a way of flowing impersonation capabilities, along with the client's identity, to a back-end service.

To configure SSO for MS SQL Server, you must set up delegation rights to the MS SQL Server service for the Qlik Sense services administrator.

A Windows domain administrator can change the delegation tab on the Qlik Sense services administrator account properties page.

Do the following:

1. Log on as a Windows domain administrator.
2. Right click on your Qlik Sense services administrator account and click **Properties**.
3. Go to the **Delegation** tab, and select **Trust this user for delegation to specified services only**, then select **Use any authentication protocol**.
4. Click **Add...**
5. On the **Add Services** window, click **Users or Computers...**
6. On the **Select Users or Computers** window, enter the domain and user name of the Microsoft SQL Server services administrator and click **OK**.
7. On the **Add Services** window, select the MS SQL Server service and click **OK**.

You can verify your delegation configuration on the **Delegation** tab. The MS SQL Server service should now be set as the service to which the Qlik Sense services administrator can present delegation credentials.

Configuring the Qlik Sense server for SSO

To configure the Qlik Sense server for SSO with MS SQL Server, you must:

- Add the Qlik Sense services administrator to the **Administrator** group on the Qlik Sense server if it's not already part of that group.
- Add Qlik Sense services administrator as part of the **Act as part of the operating system** role in the **Local Security Policy**.

Do the following:

1. Log on to the Qlik Sense server as an administrator.
2. Open **Local Security Policy**, and go to **Security Settings > Local Policies > User Rights Assignment**.
3. Under **Policy**, right click on **Act as part of the operating system** and select **Properties**.
4. On the **Local Security Setting** tab, click **Add User or Group...**
5. Add the Qlik Sense services administrator account, and click **OK**.

Configuring MS SQL Server

To configure the MS SQL Server for SSO, you must ensure that the MS SQL Server service runs as the MS SQL Server services administrator.

Do the following:

1. Log on to the MS SQL Server as an administrator.
2. Open the **Sql Server Configuration Manager**.
3. Select **SQL Server Services**.
4. Select **SQL Server** in the right pane and verify that the **Log On As** column is populated with your MS SQL Server services administrator account.



You must reboot after making changes to remove the SQL self registration of the SPN under machine account and register the SPN manually on the domain account.

6.4 Authorization

Authorization is the procedure of granting or denying users access to resources.



In Qlik Sense, authentication and authorization are two distinct, unconnected actions. In addition, the sources of information used for authentication do not have to be the same as for authorization, and the other way around.

In Qlik Sense, there are two authorization systems:

- **Access control:** The access control system grants users access to the resources in Qlik Sense. The access control system is implemented in the Qlik Sense Repository Service (QRS) and independent of

the operating system.

- **Data reduction:** The data reduction functionality is based on the concept of section access, which is a way to dynamically change which data a user can view. This makes it possible to build apps that can be used by many users, but with different data sets that are dynamically created based on user information. The reduction of data is performed by the Qlik Sense Engine Service (QES). See [Managing data security with Section Access](#) for more information.

The two authorization systems are unconnected and configured separately.

Access control

This section describes the different types of access control:

- **Resource access control:** Is the user allowed to access the app? Which functions in the app is the user allowed to use (for example, printing, exporting, and snapshots)?
- **Administrator access control:** Which access rights are needed for the different roles and responsibilities of the administrators?

Resource access control

The resource access control system in Qlik Sense is based on properties. This means that the access is based on rules that refer to properties connected to resources and users in Qlik Sense.

All authorization to resources is enforced by the Qlik Sense Repository Service (QRS). The QRS only gives other Qlik Sense services access to resources that the current user is allowed to access.

The resource access control system determines the access based on the following parameters:

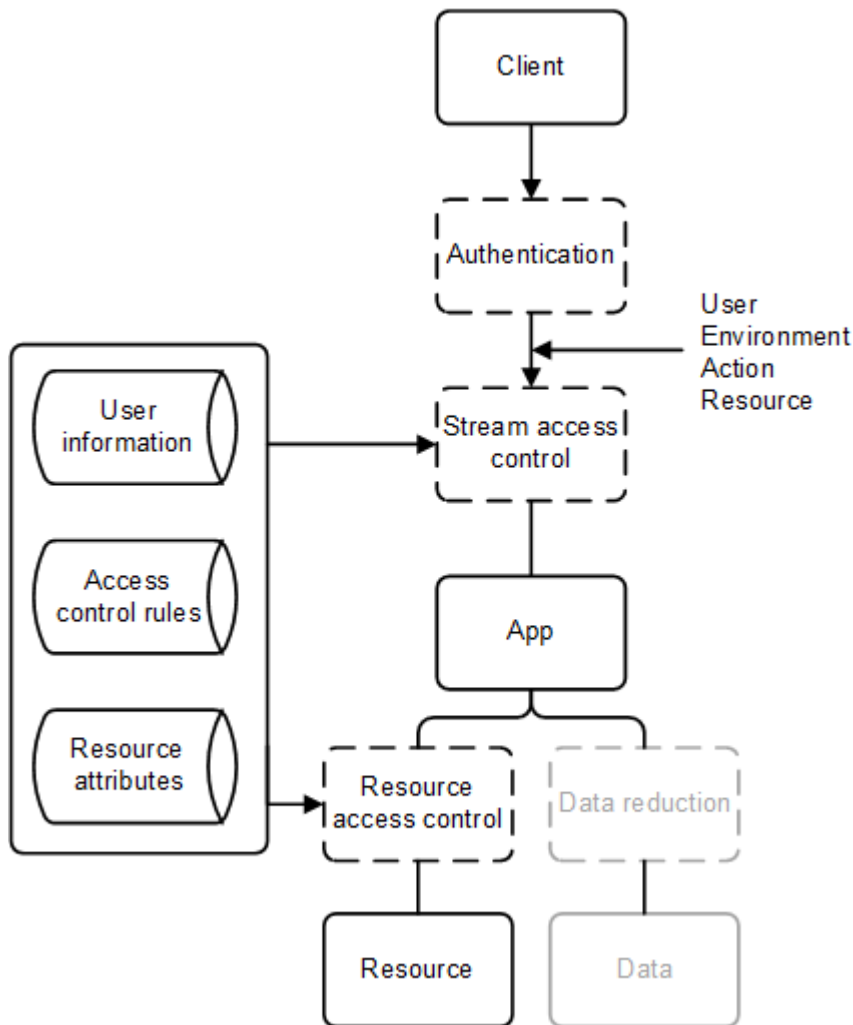
- **User name and user properties:** The user name and user properties are supplied by the Qlik Sense Proxy Service (QPS) that authenticated the user.
- **Action:** The method that the user is trying to perform on a resource (for example, create, read, or print).
- **Resource:** The entity that the user is trying to perform an action on (for example, app, sheet, or object).
- **Environment:** The environment is supplied by the QPS and describes, for example, time, location, protection, and the type of Qlik Sense client used.

Resource access control rules

The system administrator can set up rules for the resources access control. The rules are divided into three parts:

- **Resource filter:** The resources that the rule applies to.
- **Condition:** A logical condition that, if evaluated as true, grants access.
- **Action:** The action that the user is allowed to perform, if the condition is true.

Properties connected to resources or users may be used in the rules. Examples of properties include the name of user or resource, type of resource, and Active Directory groups for users or custom-defined properties.



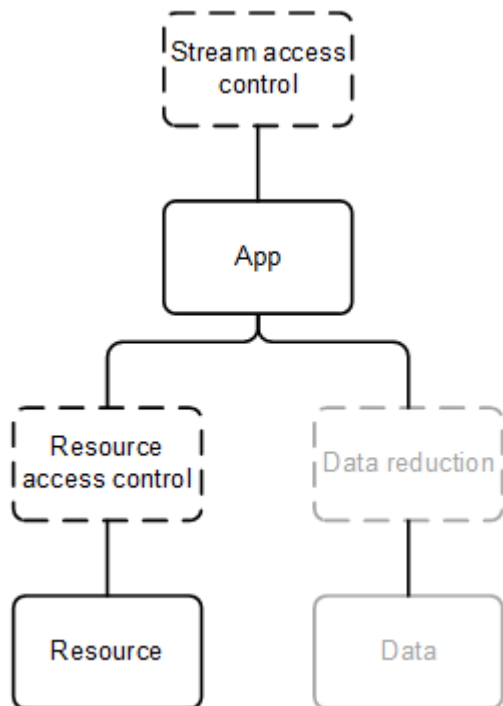
Resource access control streams

To make the management of the Qlik Sense authorization systems efficient, apps can be grouped into streams. From an authorization perspective, a stream is a grouping of apps that a group of users has read (often referred to as “subscription”) or publish access to.

By default, Qlik Sense includes the following streams:

- Everyone: All users have read and publish rights to this stream.
- Monitoring apps: Contains a number of apps for monitoring of Qlik Sense.

Streams are created and managed in the Qlik Management Console (QMC).



Administrator access control

In addition to setting up the access control for the users, it is important to configure the access control for the administrators so that they get access rights in the Qlik Management Console (QMC) that correspond to their roles and responsibilities.

Common administrator roles include the following:

- RootAdmin
- AuditAdmin
- ContentAdmin
- DeploymentAdmin
- HubAdmin (access to hub only)
- SecurityAdmin

For a presentation of the access rights for the respective administrator roles, see the topic *Default administration roles* in the document *Manage Qlik Sense sites*.

6.5 Auditing

Governance is critical in enterprise business intelligence. Qlik Sense delivers auditing, monitoring and logging using the QMC, applications, and log files to inform administrators and mitigate risks in deployments.

Qlik Sense supports auditing in the following ways:

- The repository database stores information about when the database was last changed and who made the change.
- The logging framework provides audit and security logs.

- The logs are centrally stored.
- The log format is resistant to injection from the Qlik Sense clients.
- The license logs are signed with a signature to protect them from tampering.

6.6 Confidentiality

Qlik Sense provides confidentiality by encrypting network connections with TLS, leveraging the operating system file system and server access controls to protect content on Qlik Sense nodes, protecting memory using operating system controls, securing application access at the resource level, encrypting sensitive information (e.g. passwords and data connection strings), and protecting app data using data reduction.

Qlik Sense supports confidentiality in the following ways:

- The network uses Transport Layer Security (TLS) for encryption and certificates for authentication.
- The information stored in the file share and the repository database, including Qlik Sense content, is protected by the operating system using server access control and file system controls.
- The process memory and loaded data for Qlik Sense are protected by the physical server and the operating system controls.
- The apps are secured using access control on the resource level.
- Sensitive information (for example, passwords and connection strings) that is used to access external data sources is stored with AES-256 encryption.
- The app data is protected using data reduction and data encryption.

6.7 Integrity

Qlik Sense provides integrity through operating system controls like the file system to protect data at rest, encrypt sensitive information, and prevent data write back to the source system.

Qlik Sense supports integrity in the following ways:

- Stored data is protected using the operating system controls (for example, the file system).
- Sensitive information (for example, passwords and connection strings) that is used to access external data sources is stored with AES-256 encryption.
- Qlik Sense does not support write back to the source system (that is, the Qlik Sense clients cannot edit the data sources).

Database security

In shared persistence deployments the network traffic between the servers, the database and the file share are not encrypted by default after an installation. You may also need to consider setting up replication of the database to handle cases where the central database fails.

Maintaining database password integrity

Here are some guidelines to maintain password integrity in a Qlik Sense shared persistence deployment.

- It is important that you disable the **Store password option** for your user in PostgreSQL. If this option is enabled, the password is stored in a file, and incoming connections without a password will be able to connect to the database.
- Change password by executing this query in the PostgreSQL database:

```
ALTER USER <user> WITH PASSWORD '<newpassword>';
```

```
ALTER ROLE
```

 is displayed after successfully changing the password.
Do not change password in the PostgreSQL user interface for the same reasons as above.
- Use MD5 (default) password encryption.
- Do not set your password to `PASSWORD ''`, that is, an empty string, since this is not handled well in PostgreSQL.

Changing from MD5 to SCRAM encryption

The default password encryption for the PostgreSQL database is MD5. You can change this to SCRAM after installation from the pgAdmin desktop app. The encryption method is always appended to the database user password and PostgreSQL superuser password, for example, `md5password` or `SCRAM-SHA-256password`.

Run the following commands as superuser in pgAdmin desktop:

1. Check that encryption is currently set to MD5.

```
show password_encryption;
```
2. Verify that the database user password is encrypted with MD5.

```
select passwd from pg_shadow where username = 'qliksenserepository';
```
3. Verify that the PostgreSQL superuser password is encrypted with MD5.

```
select passwd from pg_shadow where username = 'postgres';
```
4. Change the encryption to SCRAM.

```
alter system set password_encryption = 'scram-sha-256';
```
5. Reload the config file to show encryption change.

```
select pg_reload_conf();
```
6. Verify password encryption has changed to SCRAM.

```
show password_encryption;
```
7. Reset the database user password so it's saved using SCRAM encryption.

```
alter user qliksenserepository with password '*****';
```
8. Reset the PostgreSQL superuser password so it's saved using SCRAM encryption.

```
alter user postgres with password '*****';
```
9. Verify the encryption method for the user.

```
select passwd from pg_shadow where username = 'qliksenserepository';
```
10. Verify the encryption method for the superuser.

```
select passwd from pg_shadow where username = 'postgres';
```

Manually update the `pg_hba.conf` file.

1. Go to `%ProgramData%\Qlik\Sense\Repository\PostgreSQL\14`.
2. Open `pg_hba.conf`.
3. Change `md5` to `scram-sha-256` and save the file.
4. Restart the Qlik Sense Repository Database service.

Database traffic encryption

Qlik Sense supports database traffic encryption using SSL, but you need to perform some manual configuration to setup SSL and password protection, for example, MD5 or SCRAM-SHA-256, in a shared persistence deployment.



The Qlik Sense installer cannot use SSL encryption for establishing connection to PostgreSQL. When SSL encryption is enabled, the installer does not recognize any already installed PostgreSQL databases, and as a consequence, installation cannot be completed. Workaround: temporarily disable SSL during installation or upgrade.

Do the following:

1. Edit the following values in *postgresql.conf*:

```
listen_addresses = '*'
port = 4432
ssl = on
ssl_cert_file = 'server.pem'
ssl_key_file = 'server_key.pem'
#ssl_ca_file = ''
#ssl_cr1_file = ''
```
2. Add one of the following lines in *pg_hba.conf* depending on the authentication method used.

```
hostssl    all             all             all             md5
hostssl    all             all             all             scram-sha-256
```
3. Remove any other lines starting with *hostssl* or *host* in *pg_hba.conf*.
4. Copy *server.pem*, and *server_key.pem* from *%PROGRAMDATA%\Qlik\Sense\Repository\Exported Certificates\Local Certificates* to *%PROGRAMDATA%\Qlik\Sense\Repository\PostgreSQL\12.x*.
5. Use the **Connection String Editor** to add the following setting to the *repository.exe.config* on the central node, and all rim nodes that belong to the cluster. To open the **Connection String Editor**, navigate to *C:\Program Files\Qlik\Sense\Repository\Util\QlikSenseUtil* and open the *QlikSenseUtil.exe* file as an administrator.
6. In the **Connection String Editor** tab, click **Read** to open the *Repository.exe* file connection string.
7. Add 'ssl Mode=Require;' to the connection string:

```
<add name="QSR" connectionString="User ID=qlikenserepository;ssl
Mode=Require;Host='fullhostname.com';Port='4432';Database=QSR;Pooling=true;Min Pool
Size=0;Max Pool Size=90;Connection Lifetime=3600;Unicode=true;Password='randompass';"
providerName="Devart.Data.PostgreSql" />
<add name="QSMQ" connectionString="User ID=qlikenserepository;ssl
Mode=Require;Host='fullhostname.com';Port='4432';Database=QSMQ;Pooling=true;Min Pool
Size=0;Max Pool Size=90;Connection Lifetime=3600;Unicode=true;Password='randompass';"
providerName="Devart.Data.PostgreSql" />
```
8. Click **Save value in config file encrypted** to save your changes.
9. Start all Qlik Sense services and verify that everything works.
10. Verify the authentication using the pgAdmin tool in PostgreSQL:
Users *postgres* and *qlikenserepository* must enter a valid password to connect.

Forcing the database connection to use TLS 1.2 only

You can configure the database connection to support TLS 1.2 only, and block connections using TLS 1.1 or lower.

Do the following:

- Add the following parameter to the connection string: "SSL TLS Protocol=1.2"

We recommend these additional configuration changes to maintain database integrity:

- Configure the database to only accept connections from servers where the repository is running.
- Configure SSL to reject weak cipher suites by adding this line to the file *postgresql.conf*:
ssl_ciphers = 'DEFAULT:!LOW:!EXP:!eNULL:!aNULL:!MD5:!RC2:!RC4:!DES:@STRENGTH'

Encrypting database connection for services controlled by the Qlik Sense Service Dispatcher

The following code snippets can be used to enable database encryption for the following services controlled by the Qlik Sense Service Dispatcher.

Licenses service

In *C:\Program Files\Qlik\Sense\Licenses\appsettings.json*:

```
{
  "licenses": {
    "host": "localhost",
    "port": 4432,
    "dbName": "Licenses",
    "user": "qliksenserepository",
    "password":
"AQAAANCMnd8BFdERjHoAwE/Cl+sBAAAABuVYPntQ2k+cr8K7Frd+MQQAAAACAAAAAAQZGAAAAEAACAAAAD8/TGvNzOD0
PC1eEynZCIfw+q/cpFaHRLCSRuR2CXjSgAAAAAOGAAAAIAACAAAABSZavuu/1RwW2s92wdDbOeUW2sHSZP8sXI0PFPyAT
7ZSAAAAD4GqzdVQacn/SzaN03617zNlfzglowMethVPGop2bv2UAAAADSfbcNkIOY4CEBJ/jh2djgfvEUu0L2Q8nipfwxy
Mg3N05XLEGxUTpZ0riJ+J9LRX9wyw84tkAToP4pexntagZ+",
    "sslMode": "require"
  },
  "messageQueue": {
    "host": "localhost",
    "port": 4432,
    "dbName": "QSMQ",
    "user": "qliksenserepository",
    "password":
"AQAAANCMnd8BFdERjHoAwE/Cl+sBAAAABuVYPntQ2k+cr8K7Frd+MQQAAAACAAAAAAQZGAAAAEAACAAAAA78d6YdDM+L
1OGg0C/d1irzf3M14/cskYQxB4A/DvyfwAAAAAOGAAAAIAACAAAACTPvVY32teeFMJbZNSsSC/4xqaOF5j5BT7T1CA/RW
kgjAAAADa00tbEjL6DpP1sPh8optOF+diHum2gpxFzmmfDtubF0AAAAD9ujXzsYyw53yvVUQUMtJNfoZnz6y40wdU0LcSo
MACuCst4w5vryetKdRAQF7jn1P1b5Rnt4+xONi17d4bPJs1",
    "sslMode": "require"
  }
}
```

App distribution service

In *C:\Program Files\Qlik\Sense\AppDistributionService\appsettings.json*:

```
"Postgres": {
  "Host": "localhost",
  "Port": 4432,
  "Database": "SenseServices",
  "Username": "qliksenserepository",
  "Password":
"AQAAANCMnd8BFdERjHoAwE/Cl+sBAAAABuvYPntQ2k+cr8K7frd+MQQAAAACAAAAAAQZgAAAAEAACAAACEws1dk+PEB
5TNRkrMpmMguUuMYKQx/StRpCT08T4mSgAAAAAOGAAAAIAACAAAD9CE26tQn2no6qttNjzyqeBZQkgIY1491w98Fvy6T
yriAAAAA2LiBpizUuEgfs1XKZHgrD4bdy12ErkG3zD3afabBmBkAAAAAZGqqheCccu1CnhEMiMjCbIEcyPFLQKmtJ5cXHN
HSN2S9kTdAJjnZi5N9DiQi+0PhxgHFFPawpsqvSvJbDrgXs",
  "ConnectionRetryPolicy": {
    "MaxRetries": 10,
    "RetryTimeMs": 100
  },
  "Security": {
    "Enable": true,
    "ServerCertificate": {
      "Path": "C:\\ProgramData\\Qlik\\Sense\\Repository\\Exported Certificates\\.Local
Certificates\\server.pem",
      "PrivateKeyPath": "C:\\ProgramData\\Qlik\\Sense\\Repository\\Exported
Certificates\\.Local Certificates\\server_key.pem"
    },
    "RootCertificate": {
      "Path": "C:\\ProgramData\\Qlik\\Sense\\Repository\\Exported Certificates\\.Local
Certificates\\root.pem"
    }
  }
},
```

Hybrid deployment service

In *C:\Program Files\Qlik\Sense\HybridDeploymentService\appsettings.json*:

```
"Postgres": {
  "Host": "localhost",
  "Port": 4432,
  "Database": "SenseServices",
  "Username": "qliksenserepository",
  "Password":
"AQAAANCMnd8BFdERjHoAwE/Cl+sBAAAABuvYPntQ2k+cr8K7frd+MQQAAAACAAAAAAQZgAAAAEAACAAADKcv4roLbsa
B0Vw9XBLAyHp+d/+C7m31sSQg0vhBIKdAAAAAOGAAAAIAACAAADce1T09aFSv0NGUHYt5fjvkd/w+vTEnsfXT4uXACK
puIAAAAAHZVoGx2tMg/zUVqykZVtAVngR2BtNcrk1z0zG2z90QUAAACQUSC0gv71htu90HA51n1VVXSTUB1GfVto0nc/z
qoIujyAcMi8svRQHJLZ1ae90hQM+SnKUT1Yvs7JkQ4FquSg",
  "Security": {
    "Enable": true,
    "ServerCertificate": {
      "Path": "C:\\ProgramData\\Qlik\\Sense\\Repository\\Exported Certificates\\.Local
Certificates\\server.pem",
      "PrivateKeyPath": "C:\\ProgramData\\Qlik\\Sense\\Repository\\Exported
Certificates\\.Local Certificates\\server_key.pem"
    },
    "RootCertificate": {
      "Path": "C:\\ProgramData\\Qlik\\Sense\\Repository\\Exported Certificates\\.Local
Certificates\\root.pem"
    }
  }
},
```

Notifier service

In C:\Program Files\Qlik\Sense\NotifierService\appsettings.json:

```
{
  "qsmq": {
    "host": "localhost",
    "port": 4432,
    "database": "QSMQ",
    "user": "qliksenserepository",
    "password":
"AQAAANCMnd8BFdERjHoAwE/Cl+sBAAAAR/UQ7Qw2UkKeUZc0tkzpuQAAAACAAAAAAQZgAAAAEAACAAAACH6Y8cTrKGn
DeaCwnDdIG5GVZyVs8FwozTBMJdysKTzQAAAAOgAAAAIAACAAAADTJstqSpIU9o6n3xzLXRqJFHgx3chZqxnsSHJV7b
kdRAAAABP7QcqZrgE9F4K5AoAGBZQAAAP+8Sewi+n1B6TOBS+ps1xMKYTKJD1vqa8TzcOdep54sBJfiEjLu2q1q0YKN4
DnI/KCMMLMVHdaMm1qzk9w1o1M=",
    "ssl": "true"
  },
  "senseServices": {
    "host": "localhost",
    "port": 4432,
    "database": "SenseServices",
    "user": "qliksenserepository",
    "password":
"AQAAANCMnd8BFdERjHoAwE/Cl+sBAAAAR/UQ7Qw2UkKeUZc0tkzpuQAAAACAAAAAAQZgAAAAEAACAAAABHh7YwG9F99
6GbE1Jbry6B7Jiytn8432DsQ0VmgIXKQGAAAAOgAAAAIAACAAAACyXIKEvBO7axFgGINUuWLD76jSkNNK6DbiBaVnRU
kGBAAAABqzh9FMFbJDXwd532nEukBQAAAAHjGKDYS+/BN1FhMqBd77G0tXN/i5LAC96mwZahRZ4hE/Ve7aa2Uqx2/SwdwM
UIr6g8xhu9CJ56QwRkukj7pRXc=",
    "ssl": "true"
  },
}
```

Mobility registrar service

In C:\Program Files\Qlik\Sense\MobilityRegistrarService\appsettings.json:

```
{
  "Postgres": {
    "Host": "localhost",
    "Port": 4432,
    "Database": "SenseServices",
    "Schema": "qlik_mobility_registrar_service",
    "Username": "qliksenserepository",
    "Password":
"AQAAANCMnd8BFdERjHoAwE/Cl+sBAAAAR/UQ7Qw2UkKeUZc0tkzpuQAAAACAAAAAAQZgAAAAEAACAAA6L9dGr9oeI
aqpdxz9W4BP2QmUHTxaFzGfzx051sUrnQAAAAOgAAAAIAACAAAAXU1esPxGwBi+xs4eH3qB3WXUDPm4QbwbiAWbn1f
w9hAAAAADamUq8qBtA6qhQUzmcP12MQAAAAHYky7wdQgBw20cXPN6wK00xnp+Iizw+MeMhqDQPH0iUnnkCLQo40jCF1ijHw
XeDcxVEGircje1xCVBV/Itf94k=",
    "SSL": 0
  }
}
```

NL broker service

In C:\Program Files\Qlik\Sense\NLBroker\appsettings.json:

6 Qlik Sense Enterprise on Windows security

You must use the common name (CN) for the host value. To get the CN, run the following commands in an elevated PowerShell.

```
$CertPath = [Environment]::GetFolderPath('CommonApplicationData') +
"\Qlik\Sense\Repository\PostgreSQL\12.x\server.pem"
$Cert = New-Object System.Security.Cryptography.X509Certificates.X509Certificate2 -
ArgumentList $CertPath
$Cert | Select-Object -Property Subject
```

Take the output from the commands above and replace "localhost" in the snippet below.

```
{
  "Postgres": {
    "Host": "localhost",
    "Port": 4432,
    "Database": "SenseServices",
    "Schema": "n1_broker",
    "Username": "qliksenserepository",
    "Password":
"AQAAANCMnd8BFdERjHoAwE/Cl+sAAAAAnq+F3zxLakeZ96CexCiJJwQAAAACAAAAAAQZgAAAAEAACAAAAAnIrak2CdHui
vvpChAMP8d0053ZGmaG3WFnuNnqSxaFAAAAAAOGAAAAIAACAAACQSam6rRrQrLsgSq+IKUeEZBzppyDNhKC+ss2uAA4I
8CAAAAIIj29QiQoCXEKfZHPxzOhas2M0tC/tYm+QJbUtSC7SY0AAAACdXaXWr688zj1DciHYx/h79vXX611+G0U5AigLwr
SwsCNTJTh1c1p0gGr2YSyS54ESdkqqD+fi+vEKHO00+2wH",
    "Security": {
      "comment": "See sslmode descriptions at https://github.com/brianc/node-postgres/tree/master/packages/pg-connection-string#tcp-connections and
https://www.postgresql.org/docs/12/libpq-ssl.html#LIBPQ-SSL-PROTECTION",
      "Enable": true,
      "Mode": "require",
      "Certificate": "C:\\ProgramData\\Qlik\\Sense\\Repository\\Exported Certificates\\.Local
Certificates\\client.pem",
      "Key": "C:\\ProgramData\\Qlik\\Sense\\Repository\\Exported Certificates\\.Local
Certificates\\client_key.pem",
      "RootCertificate": "C:\\ProgramData\\Qlik\\Sense\\Repository\\Exported
Certificates\\.Local Certificates\\root.pem"
    }
  }
}
```

NL app search service

In *C:\Program Files\Qlik\Sense\NLAppSearch\appsettings.json*

You must use the common name (CN) for the host value. To get the CN, run the following commands in an elevated PowerShell.

```
$CertPath = [Environment]::GetFolderPath('CommonApplicationData') +
"\Qlik\Sense\Repository\PostgreSQL\12.x\server.pem"
$Cert = New-Object System.Security.Cryptography.X509Certificates.X509Certificate2 -
ArgumentList $CertPath
$Cert | Select-Object -Property Subject
```

Take the output from the commands above and replace "localhost" in the snippet below.

```
{
  "Postgres": {
    "Host": "localhost",
    "Port": 4432,
    "Database": "QSMQ",
```

```
"Password":
"AQAAANCMnd8BFdERjHoAwE/Cl+sBAAAAnq+F3zxlakeZ96CexCiJwQAAAAACAAAAAAQZgAAAAEAACAAAC7N047wjSCf
F7zII62y1DMRh6ijHaj57BNW+asBRisqgAAAAAogAAAAIAACAAAC/FuJ1IavkkFodWnIoeRLO8RpiysHPop4Dyqz8PFy
aoSAAACxdHYmcrnNTYM73q0FccSeQVWW3dZ1y/gz3Q4PRGOEAAAABK5mNZCMFF68nReI7oFhLJXw7oZ13u2PoD804hmqN
JHicsOvzufdDDzM8YNCcrq/YVYakhiOceReEbiehxm0Jh",
  "Username": "qliksenserepository",
  "Security": {
    "comment": "See sslmode descriptions at https://github.com/brianc/node-
postgres/tree/master/packages/pg-connection-string#tcp-connections and
https://www.postgresql.org/docs/12/libpq-ssl.html#LIBPQ-SSL-PROTECTION",
    "Enable": true,
    "Mode": "require",
    "Certificate": "C:\\ProgramData\\Qlik\\Sense\\Repository\\Exported Certificates\\.Local
Certificates\\client.pem",
    "Key": "C:\\ProgramData\\Qlik\\Sense\\Repository\\Exported Certificates\\.Local
Certificates\\client_key.pem",
    "RootCertificate": "C:\\ProgramData\\Qlik\\Sense\\Repository\\Exported
Certificates\\.Local Certificates\\root.pem"
  }
},
"QRS": {
  "Url": "https://localhost:4242/qrs",
  "ClientCertificate": {
    "Path": "C:\\ProgramData\\Qlik\\Sense\\Repository\\Exported Certificates\\.Local
Certificates\\client.pem",
    "PrivateKeyPath": "C:\\ProgramData\\Qlik\\Sense\\Repository\\Exported
Certificates\\.Local Certificates\\client_key.pem"
  },
  "RootCertificate": {
    "Path": "C:\\ProgramData\\Qlik\\Sense\\Repository\\Exported Certificates\\.Local
Certificates\\root.pem"
  }
}
}
```

DataPrep service

In *C:\Program Files\Qlik\Sense\DataPrepService\appsettings.json*

You must use the common name (CN) for the host value. To get the CN, run the following commands in an elevated PowerShell.

```
$CertPath = [Environment]::GetFolderPath('CommonApplicationData') +
"\Qlik\Sense\Repository\PostgreSQL\12.x\server.pem"
$Cert = New-Object System.Security.Cryptography.X509Certificates.X509Certificate2 -
ArgumentList $CertPath
$Cert | Select-Object -Property Subject
```

Take the output from the commands above and replace "localhost" in the snippet below.

```
{
  "Postgres": {
    "Host": "localhost",
    "Port": 4432,
    "Database": "SenseServices",
    "Schema": "dataprep_service",
    "Username": "qliksenserepository",
    "Password":
```



```
"AQAAANCMnd8BFdERjHoAwE/C1+sAAAAAnq+F3zxLakeZ96CexCiJJwQAAAAACAAAAAAQZgAAAAEACAAAAAnIrak2CdHui
vVPChAMP8d0053ZGmaG3WFnuNnqSxaFAAAAAAogAAAAIAACAAACQSam6rRrQrLsgSq+IKUeEZBzppYDNhKC+ss2uAA4I
8CAAAAAIj29QiqoCXEKfZHPxzOhas2MOTC/tYm+QJbUtSC7SY0AAAAACdXaWr688zj1DciHYx/h79vXX611+G0U5AigLwr
SwsCNTJTh1c1p0gGr2YSys54ESdkqqD+fi+vEKHo00+2wH",
  "Security": {
    "comment": "See sslmode descriptions at https://github.com/brianc/node-
postgres/tree/master/packages/pg-connection-string#tcp-connections and
https://www.postgresql.org/docs/12/libpq-ssl.html#LIBPQ-SSL-PROTECTION",
    "Enable": true,
    "Mode": "require",
    "Certificate": "C:\\ProgramData\\Qlik\\Sense\\Repository\\Exported Certificates\\.Local
Certificates\\client.pem",
    "key": "C:\\ProgramData\\Qlik\\Sense\\Repository\\Exported Certificates\\.Local
Certificates\\client_key.pem",
    "RootCertificate": "C:\\ProgramData\\Qlik\\Sense\\Repository\\Exported
Certificates\\.Local Certificates\\root.pem"
  }
}
```

Precedents service

In *C:\Program Files\Qlik\Sense\PrecedentsService\appsettings.json*

You must use the common name (CN) for the host value. To get the CN, run the following commands in an elevated PowerShell.

```
$CertPath = [Environment]::GetFolderPath('CommonApplicationData') +
"\Qlik\Sense\Repository\PostgreSQL\12.x\server.pem"
$Cert = New-Object System.Security.Cryptography.X509Certificates.X509Certificate2 -
ArgumentList $CertPath
$Cert | Select-Object -Property Subject
```

Take the output from the commands above and replace "localhost" in the snippet below.

```
{
  "Postgres": {
    "Host": "localhost",
    "Port": 4432,
    "Database": "SenseServices",
    "Schema": "precedents_service",
    "Username": "qliksenserepository",
    "Password":
"AQAAANCMnd8BFdERjHoAwE/C1+sBAAAAAnq+F3zxLakeZ96CexCiJJwQAAAAACAAAAAAQZgAAAAEACAAAAACI2Gn1PHqAG
8iKXI+Nv92uE5DeGKaDpoMLSLjSX3M3BgAAAAAogAAAAIAACAAADo8y30KfZ517PHi1kr+9SQA9uYGnnZjKLM8eBfrzw
2UCAAAAB+Mrbs74uJo41Mr+Jr8gSbxxa2ZkrNEKqI1WRDo5PpwkAAADw2QwbQktqRT23B9N5jBJtqw/7BqVKVylw1VCzy
W0FqHHMBxyD9Gn6ajMfmeKMRjxjwB2eREt5HXLM7EUE0s1w",
    "Security": {
      "comment": "See sslmode descriptions at https://github.com/brianc/node-
postgres/tree/master/packages/pg-connection-string#tcp-connections and
https://www.postgresql.org/docs/12/libpq-ssl.html#LIBPQ-SSL-PROTECTION",
      "Enable": true,
      "Mode": "require",
      "Certificate": "C:\\ProgramData\\Qlik\\Sense\\Repository\\Exported Certificates\\.Local
Certificates\\client.pem",
      "key": "C:\\ProgramData\\Qlik\\Sense\\Repository\\Exported Certificates\\.Local
Certificates\\client_key.pem",
      "RootCertificate": "C:\\ProgramData\\Qlik\\Sense\\Repository\\Exported
```

```

Certificates\\.Local Certificates\\root.pem"
  }
},
"QRS": {
  "Url": "https://localhost:4242/qrs",
  "RetryPolicy": {
    "MaxRetries": 3,
    "RetryTimeMs": 100
  },
  "Headers": {
    "X-Qlik-User": "UserDirectory=INTERNAL; UserId=sa_api"
  },
  "ClientCertificate": {
    "Path": "C:\\ProgramData\\Qlik\\Sense\\Repository\\Exported Certificates\\.Local
Certificates\\client.pem",
    "PrivateKeyPath": "C:\\ProgramData\\Qlik\\Sense\\Repository\\Exported
Certificates\\.Local Certificates\\client_key.pem"
  },
  "RootCertificate": {
    "Path": "C:\\ProgramData\\Qlik\\Sense\\Repository\\Exported Certificates\\.Local
Certificates\\root.pem"
  }
}
}
}

```

Additional configuration

For each node.js service, it is also possible to configure https communication with the service with additional `--ssl` parameter in the `C:\Program Files\Qlik\Sense\ServiceDispatcher\services.conf`. In the following example, TLS 1.2 is configured for Resource distribution service:

```

[resource-distribution]
Identity=Qlik.resource-distribution
DisplayName=Resource Distribution
ExePath=Node\node.exe
Script=..\ResourceDistributionService\server.js

```

```

[resource-distribution.parameters]
--secure
--wes-port=${WESPort}
--mode=server
--log-path=${LogPath}
--log-level=info
--ssl=369098752

```

The following node.js code generates the number 369098752 to be used to configure OpenSSL according to [OpenSSL Options](#):

```

const crypto = require("crypto");
console.log(crypto.constants.SSL_OP_NO_SSLV2 |
crypto.constants.SSL_OP_NO_SSLV3 | crypto.constants.SSL_OP_NO_TLSV1 |
crypto.constants.SSL_OP_NO_TLSV1_1);

```



*The generated number might differ depending on the node.js version shipped with Qlik Sense:
C:\Program Files\Qlik\Sense\ServiceDispatcher\Node\node.exe.*

Database replication and failover

This section describes how to set up database replication and failover in a shared persistence environment. Additionally, the file storage content will also need to be replicated. To fail over to a standby node in case the central database or node is lost, one or more standby databases can be configured for streaming replication from the database on the primary node.

When editing text files related to the Qlik Sense installation, do the following:

1. Copy the file to another location on the server.
2. Edit the file and save the changes.
3. Copy the updated file back to its original location.

Setting up replication to standby nodes for failover

The instructions in this section describe how to set up asynchronous streaming replication to one or more standby nodes. Before starting, ensure that the environment is configured and running, and install PostgreSQL on a standby machine.



The paths in the instructions are adapted to a default PostgreSQL installation used as database on a dedicated machine. If you are using a PostgreSQL database installed by Qlik Sense you need to adapt the paths used, as the database is installed in %ProgramData%\Qlik\Sense\Repository\PostgreSQL\<version>.

Configure the primary database server

On the primary database server, do the following:

1. Open the file %ProgramFiles%\PostgreSQL\<version>\data
Locate and set the following settings
wal_level = replica
max_wal_senders = 3
wal_keep_segments = 8
hot_standby = on
2. Create a user account that can be used for replication. To do so from a command prompt, run the following command. Adjust the hostname as needed, and specify a suitable password. You may be prompted for a password, this is the password that was specified during installation.

```
"C:\Program Files\PostgreSQL\<version>\bin\psql.exe" -h <machinename> -p 4432 -w -c "CREATE USER replicator REPLICATION LOGIN ENCRYPTED PASSWORD 'secretpassword';"
```
3. Open the file %ProgramFiles%\PostgreSQL\<version>\data\pg_hba.conf.
At the bottom of the file add:
host replication replicator 0.0.0.0/0 md5
You can restrict the subnet access further, if required.
4. Restart the PostgreSQL service.

Configure the standby database server (PostgreSQL 12.x)

On the standby PostgreSQL database server, do the following:

1. Stop the PostgreSQL service.
2. Delete all content from `%ProgramFiles%\PostgreSQL\<version>\data`.
3. From the command line run the following command adjusted to use the name of the primary server:

```
"C:\Program Files\PostgreSQL\<version>\bin\pg_basebackup.exe" -h <primaryServer> -D  
"C:\Program Files\PostgreSQL\<version>\data" -U replicator -v -P -p 4432
```

You can ignore any warnings about copying files manually.
4. In a text editor, create a file called `standby.signal` (to indicate the server should start up as a hot standby) or `recovery.signal` (to indicate the server should start up in targeted recovery mode) and place it in `%ProgramFiles%\PostgreSQL\<version>\data`.



If standby.signal and recovery.signal are both present, standby.signal takes precedence.

5. Start the PostgreSQL service.

Configure the standby database server (PostgreSQL 11.x)

On the standby PostgreSQL database server, do the following:

1. Stop the PostgreSQL service.
2. Delete all content from `%ProgramFiles%\PostgreSQL\<version>\data`.
3. From the command line run the following command adjusted to use the name of the primary server:

```
"C:\Program Files\PostgreSQL\<version>\bin\pg_basebackup.exe" -h <primaryServer> -D  
"C:\Program Files\PostgreSQL\<version>\data" -U replicator -v -P -p 4432
```

You can ignore any warnings about copying files manually.
4. In a text editor, create a file called `recovery.conf` and place it in `%ProgramFiles%\PostgreSQL\<version>\data`.
5. Open `recovery.conf` and add the following text, adjusting the hostname and port:

```
standby_mode = 'on'  
primary_conninfo = 'host=< primaryServer > port=4432 user=replicator  
password=secretpassword'  
trigger_file = 'failover'  
recovery_target_timeline = 'latest'
```
6. Start the PostgreSQL service.

You should now be able to connect to the database and view the data being streamed over from the primary node in read only mode.

Manual database failover

If the database on primary node is lost, a standby node needs to take over.

Do the following:

1. On the standby node that is to become the new primary node, create a file called `failover` in the folder `%ProgramFiles%\PostgreSQL\<version>\data`



The failover file should have no file extension.

The file triggers PostgreSQL to cease recovery and enter read/write mode.

2. On each node, change the repository database connection string to point to the hostname or IP address of the new database node. As the connection string is encrypted in the config file, you need to use the **Connection String Editor** to decrypt the string, edit it, and write back an encrypted string.
 - a. To open the **Connection String Editor**, navigate to `C:\Program Files\Qlik\Sense\Repository\Util\QlikSenseUtil` and open the `QlikSenseUtil.exe` file as an administrator.
 - b. In the **Connection String Editor** tab, click **Read** to open the `Repository.exe` file connection string.
The decrypted database connection string is displayed.
 - c. Replace the value for **Host** with the hostname or IP address of the new database node.
 - d. Click **Save value in config file encrypted** to save your changes.

Data encryption

You can encrypt sensitive data in QVF and QVD files with customer supplied key pairs which allows you to control who gets access to your data. The encryption keys are managed through certificates, that must be stored in a certificate store for the user running the Engine service.

The encryption is configured in the Qlik Management Console (QMC), where encryption is enabled and the certificate thumbprint is added. Data encryption is not enabled by default.

The engine reads and then uses the thumbprint to get the key from the Windows CNG key store. The engine then generates a new data encryption key (DEK) which is used to encrypt the data.



A DEK is never reused which ensures that if one file is compromised, the encryption is still valid for all other files.

QVF encryption

The following is encrypted:

- data (tables and fields)
- bookmarks

The following is not encrypted:

- objects, for example sheets and stories
- static content, such as images



You must reload an existing QVF for it to be encrypted after QVF encryption has been enabled in the QMC.

QVD encryption

The following is encrypted:

- Data (tables and fields)

The QVD header is not encrypted. Encryption parameters are stored in the QVD header as extra meta-data.



You must reload an existing QVD for it to be encrypted after QVD encryption has been enabled in the QMC.

Older versions of Qlik Sense and QlikView returns an error when reading encrypted QVDs files.

Encryption certificates

Encryption keys are best managed through certificates. The certificates must be stored in a certificate store for the user running the Engine service.

The encryption certificate functions as a shell around the encryption key. The key can be fetched even if the certificate has expired, and therefore there is no need to renew an expired encryption certificate.



Make sure to back up the certificate. You may not be able to open your encrypted app if the certificate is lost. It is your responsibility to keep safe the certificate backup for as long as it is needed.



*If your organization has a key rotation policy, you may need to update the thumbprint definition when the key is changed.
Remember to keep the certificate containing the old key on the server until all QVFs and QVDs have been saved with the new key.*

Encryption keys

The encryption solution uses two types of keys:

- Data encryption keys
- Key encryption keys

Data encryption keys

Data encryption keys (DEK) are auto-generated keys for AES-256 encryption of the data. A new key is generated for each object that is encrypted.

Key encryption keys

Key encryption keys (KEK) are private and public key pair for secure, asymmetric encryption of the data encryption keys. The public key is used to encrypt the data and the private key is used to decrypt the data encrypted by the public key.



Only keys using the RSA algorithm are supported.

The key used for key encryption is specified in the Qlik Management Console (QMC) *Data encryption* section of the *Service cluster* resource.

It is stored in a Microsoft Cryptography Next Generation (CNG) Key Storage Provider and it is contained in a certificate stored in a Windows Certificate Store.

For details of how to enable and manage encryption certificates, see *Encryption certificates*.

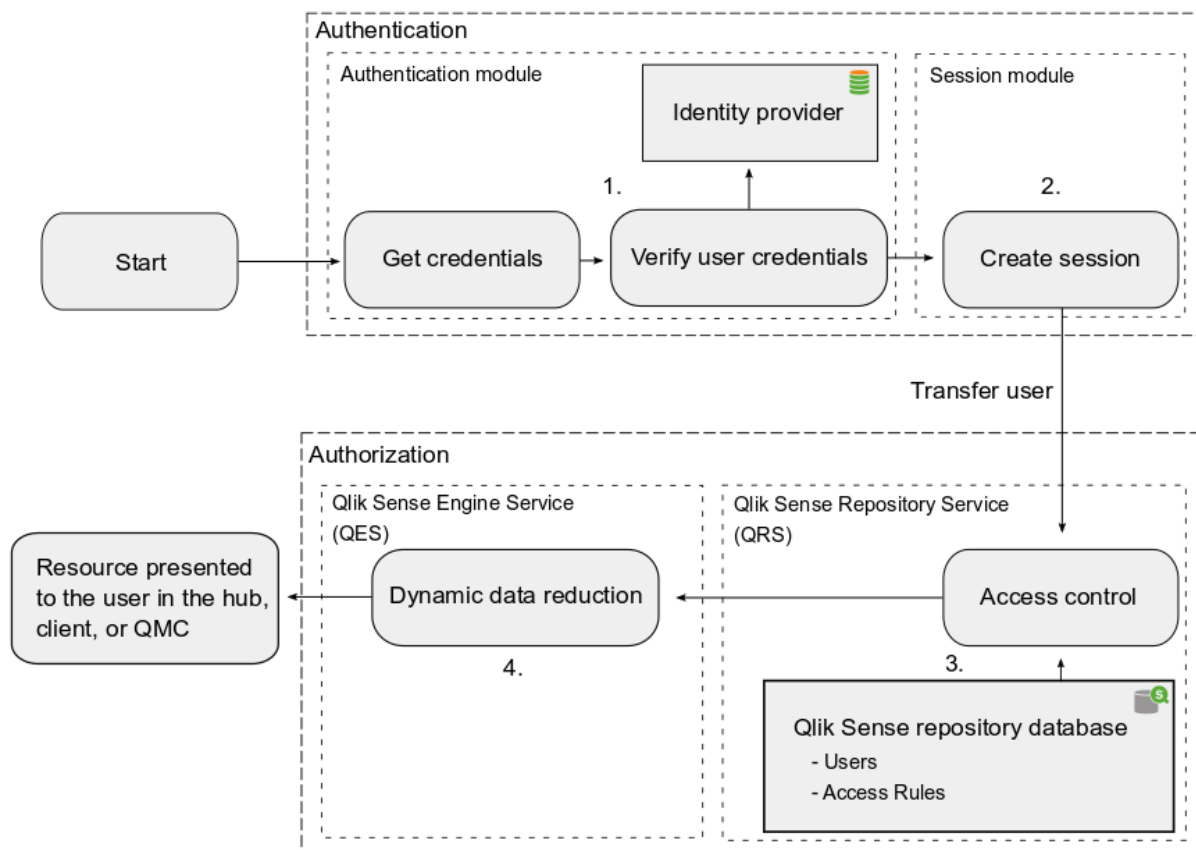
6.8 Availability

Qlik Sense supports availability in the following ways:

- The nodes in a multi-node site are resilient by design. Each node connects to a central node to access the data it needs to fulfill its role.
- The Qlik Sense protocols are designed to be fault tolerant.

6.9 Security example: Opening an app

The figure below shows the flow in the Qlik Sense security system when a user logs in and opens an app.



1. Authentication: The authentication module in the Qlik Sense Proxy Service (QPS) handles the authentication. The credentials provided by the user are verified against information from the identity provider (for example, a directory service such as Microsoft Active Directory).
2. Session creation: When the user credentials have been successfully verified by the authentication module, a session is created for the user by the session module in the QPS.

3. Access control system: When the user tries to open an app, the Qlik Sense Engine Service (QES) requests the Qlik Sense Repository Service (QRS) to check if the user is authorized to perform the action. The QRS then checks the repository database, where, among other things, all users and access rules are stored.



A user is imported into the repository database from a User Directory (UD) (for example, Microsoft Active Directory) using Qlik Sense User Directory Connectors (UDCs). The import is triggered by the Qlik Sense Scheduler Service (QSS) and the intervals in-between imports can be scheduled.

4. Dynamic data reduction: When the user has been successfully authorized by the QRS, the app is opened. Before the data is displayed to the user, the QES performs a dynamic data reduction, where the data that the user is allowed to see is prepared.

6.10 AWS and Azure security

Before you deploy Qlik Sense on AWS or Azure you need to get an overview of the basic security implications. In AWS and Azure there are specific tools that you use during setup to configure permissions and to set security options. Once you have deployed Qlik Sense to your chosen cloud environment, you use the Qlik Management Console to configure security in the same way as you would in an on-premise Qlik Sense deployment.

Qlik Sense

An overview of your Qlik Sense security considerations:

- In Qlik Sense, you manage all security and authentication settings from the Qlik Management Console.
- A module in the Qlik Sense Proxy Service handles authentication of Microsoft Windows users.
- Authentication is often used in conjunction with a single sign-on (SSO) system that supplies a reverse proxy or filter for authentication of the user.
- Other authentication methods are available, and it is possible to implement your own customized solutions for different authentication scenarios.

Resources managed directly from the QMC:

- Admin roles to grant QMC users administrator level access to various sections
- Proxy certificate for communication between the web browser and the proxy component
- Virtual proxies to allow different modules based on the URI to be used to access the Qlik Sense environment
- Custom properties enabling you to use your own values in security rules
- Access control and security rules to grant users access to Qlik Sense resources

Authentication methods used by Qlik Sense:

- NTLM/Kerberos
- Security Assertion Markup Language (SAML)

- Anonymous authentication
- Session/Ticket API

For more information about Qlik Sense security, see *Qlik Sense Enterprise on Windows security (page 208)*

AWS

To configure security in an AWS deployment you need a basic understanding of how to set up AWS security groups, key pairs, and Qlik Sense security groups. Use the Amazon Management Console to configure AWS security, and the Qlik Management Console to configure all security and authentication settings in Qlik Sense. A module in the Proxy Service (QPS) handles the authentication of Microsoft Windows users. If required, it is also possible to implement your own custom authentication solutions.

Use the Amazon Management Console to configure:

- AWS security groups - configure access rules for an initial Qlik Sense security group for your EC2 instance.
- Key pair - In the AWS console, create a Qlik Sense key pair. Save the `qlik_sense.pem` keypair file locally, as you will need it later to access your instance.

You can use AWS Directory Services to set up security and authentication on the Qlik Sense server side. This service makes it easier to set up and run Microsoft Active Directory (AD) in the AWS cloud, or connect your AWS resources to an existing on-premises Microsoft Active Directory.

AWS Directory Service provides you with the following three directory types:

- AWS Directory Service for Microsoft Active Directory (Enterprise Edition), also referred to as Microsoft AD
- Simple AD
- AD Connector

AWS Directory Services makes it possible to connect AWS resources to an on-premises directory using the same corporate credentials. This option uses the Microsoft Security Support Provider Interface (SSPI) to read the Windows user name and password working in a similar way to single sign-on. If you have multiple nodes in the Qlik Sense Server environment, all nodes need to be part of the same domain.

For more information, see [AWS security](#).

Azure

Use the Resource Manager to configure Azure security and the QMC to configure all security groups and authentication settings in Qlik Sense. In Azure, to configure security you first set up a subnet, a virtual network, an IP address for an instance, and network security rules. This is similar to configuring ports in a firewall. You then set up a network interface that your instance can use, and bind it to the previously set up network and subnet. A module in the Qlik Sense Proxy Service (QPS) handles the authentication of Microsoft Windows users. If required, it is also possible to implement your own custom authentication solutions.

Use the Azure Resource Manager to configure:

- Azure security groups
- Azure Active Directory and Identity Management

Azure Active Directory (Azure AD) is Microsoft's multi-tenant cloud based directory and identity management service. For IT administrators, Azure AD provides an easy to use solution to give users single sign-on (SSO) access to other cloud SaaS Applications, such as Office365, Salesforce.com, and Concur. Azure AD also includes a full suite of identity management capabilities including multi-factor authentication, device registration, self-service password management, self-service group management, privileged account management, role based access control, application usage monitoring, rich auditing, and security monitoring and alerting.

For more information, see [Azure security](#).

7 Logging

The log messages produced by Qlik Sense provide important information about the general well-being of the deployment.

The logging is based on the log4net component in Apache Logging Services. This means that Qlik Sense uses a standardized logging framework and conforms to standard logging procedures.

7.1 Updated logging framework

An updated logging framework was introduced in Qlik Sense version 2.0. Unless otherwise stated, the documentation describes the updated logging framework.

7.2 Legacy logging framework

The legacy logging framework is still available in Qlik Sense, but the logs are as of Qlik Sense version 2.0 referred to as trace logs. The log files remain the same, in the old logging format, but they are stored in a new location.

See: *Trace logs (page 269)*

7.3 Requirements

The requirements described in this section must be fulfilled for the Qlik Sense logging to function properly.

Securing the file system

The system administrator must secure the file system so that the log files cannot be tampered with.



By default, the account used for the Qlik Sense installation gets full permissions for the log folder, %ProgramData%\Qlik\Sense\Log, whereas the Users group only gets read permission. No other accounts or users get any permissions for the log folder.

Synchronizing time

The nodes within a Qlik Sense site must have synchronized time.

For the date and time stamps to be correct, all nodes within a site must be configured to synchronize their system clocks with either an internal or an external Network Time Protocol (NTP) service to ensure that all log entries are time-stamped accurately. NTP is a networking protocol for synchronizing the clocks of computer systems over packet-switched, variable-latency data networks.

Setting time zone

It is recommended that every node within a Qlik Sense site is set to the correct time zone so that the time zone corresponds to the geographical location of the node.

7.4 Storage

The default log files are stored in folders under `%ProgramData%\Qlik\Sense\Log`. The local log configuration file can be used to set up the logging so that the log files are also stored in another location.

Log folder

The following table describes the contents of the `%ProgramData%\Qlik\Sense\Log` folder.

Log table contents

Folder	Sub-folder	Files	Description
<code>\AboutService</code>	-	-	This folder contains log files related to the About Service.
<code>\AppDistributionService</code>	<code>\Trace</code>	-	This folder contains log files related to the App Distribution Service. You can modify some of the settings via the <code>appsettings.json</code> file.
<code>\BrokerService</code>	-	-	This folder contains log files related to the Broker Service.
<code>\CapabilityService</code>	-	-	This folder contains log files related to the Capability Service.
<code>\ConnectorRegistryProxy</code>	-	-	This folder contains log files related to the Connector Registry Proxy.
<code>\ConverterService</code>	-	-	This folder contains log files related to the Converter Service.
<code>\DataProfiling</code>	-	-	This folder contains log files related to the Data Profiling Service.
<code>\DepGraphService</code>	-	-	This folder contains log files related to the Dependency Graph Service.

Folder	Sub-folder	Files	Description
\DeploymentBasedWarningsService	-	-	This folder contains log files related to the Deployment Based Warnings Service.
\DownloadPrepService	-	-	This folder contains log files related to the Download Prep Service.
\Engine	-	<MachineName>_Exit_Engine_<Date>.txt	NewSet. is a temporary log file that is used by the the Qlik Sense Engine Service (QES) until the log pipe to the Qlik Sense Repository Service (QRS) is up and running. NewSet. log file is not archived.
-	-	<MachineName>_Start_Engine_<Date>.txt	NewSet. is a temporary log file that is used by the the Qlik Sense Engine Service (QES) until the log pipe to the Qlik Sense Repository Service (QRS) is up and running. NewSet. log file is not archived.
-	\Audit	<MachineName>_AuditActivity_<Service>.txt	This log tracks user-related actions.
-	-	<MachineName>_AuditSecurity_<Service>.txt	This log contains information on security-related actions.
-	\System	<MachineName>_Service_<Service>.txt	This log contains information on service and system operations, including all errors.
-	\Trace	<MachineName>_<Facility>_<Service>.txt	The trace log files are stored in this folder. See: <i>Trace logs (page 269)</i>

Folder	Sub-folder	Files	Description
\ExtensionBundles	-	-	This folder contains log files related to the Extension Bundles Service.
\HubService	-	-	This folder contains log files related to the Hub Service.
\HybridDeploymentService	\Trace	-	This folder contains log files related to the Hybrid Deployment Service. You can modify some of the settings via the appsettings.json file.
\HybridSetupConsoleBff	-	-	This folder contains log files related to the Hybrid Setup Console Service.
\Licenses	-	-	This folder contains log files related to the License Service.
\OdagService	-	-	This folder contains log files related to the ODAG Service.
\PrecedentsService	-	-	This folder contains log files related to the Precedents Service.
\Printing	-	-	This folder contains log files related to the Qlik Sense Printing Service (QPR).
-	\Audit	<MachineName>_AuditActivity_<Service>.txt	This log tracks user-related actions.
-	-	<MachineName>_AuditSecurity_<Service>.txt	This log contains information on security-related actions.
-	\System	<MachineName>_Service_<Service>.txt	This log contains information on service and system operations, including all errors.
-	\Trace	<MachineName>_<Facility>_<Service>.txt	The trace log files are stored in this folder. See: <i>Trace logs</i> (page 269)

Folder	Sub-folder	Files	Description
\Proxy	\Audit	<MachineName>_AuditActivity_<Service>.txt	This log tracks user-related actions.
-	-	<MachineName>_AuditSecurity_<Service>.txt	This log contains information on security-related actions.
-	\System	<MachineName>_Service_<Service>.txt	This log contains information on service and system operations, including all errors.
-	\Trace	<MachineName>_<Facility>_<Service>.txt	The trace log files are stored in this folder. See: <i>Trace logs (page 269)</i>
\QdcCatalogService	\Trace	-	This folder contains log files related to the Qdc Catalog Service.
\QlikMobilityRegistrar	-	-	This folder contains log files related to the Qlik Mobility Registrar.
\QlikNotifierService	-	-	This folder contains log files related to the Qlik Notifier Service.
\Repository	\Audit	<MachineName>_AuditActivity_<Service>.txt	This log tracks user-related actions.
-	-	<MachineName>_AuditSecurity_<Service>.txt	This log contains information on security-related actions.
-	\System	<MachineName>_Service_<Service>.txt	This log contains information on service and system operations, including all errors.
-	\Trace	<MachineName>_<Facility>_<Service>.txt	The trace log files are stored in this folder. See: <i>Trace logs (page 269)</i>
\ResourceDistributionService	-	-	This folder contains log files related to the Resource Distribution Service.

Folder	Sub-folder	Files	Description
\Scheduler	\Audit	<MachineName>_AuditActivity_<Service>.txt	This log tracks user-related actions.
-	-	<MachineName>_AuditSecurity_<Service>.txt	This log contains information on security-related actions.
-	\System	<MachineName>_Service_<Service>.txt	This log contains information on service and system operations, including all errors.
-	\Trace	<MachineName>_<Facility>_<Service>.txt	The trace log files are stored in this folder. See: <i>Trace logs (page 269)</i>
\Script	-	-	This folder contains log files related to app reloads. These logs are treated differently depending on where the reload was triggered from, the QMC or the hub. By default, four logs are saved. For more information, see: How to find the Script (Reload) logs in Qlik Sense... Qlik Sense Script logs are deleted from ArchivedLogs location
\WebExtensionService	-	-	This folder contains log files related to the Web Extension Service.

Archived log files

Archived log files are by default stored in the `\\<server>\<share>\ArchivedLogs` folder. You define the location of the file share folder during installation. Archived log files have the file extension `.log`, whereas active log files have the extension `.txt`.

See also:

 [Local log configuration file \(page 285\)](#)

7.5 Naming

The Qlik Sense log files are named in accordance to the following file rollover procedure:

1. The log is stored in a file named `<MachineName>_<LogType>_<Service>.txt`.
2. When the file is full or a pre-defined amount of time has passed, the file extension is automatically changed to `.log` and a time stamp is appended to the file name for uniqueness and archiving. This means that the new file name becomes `<MachineName>_<LogType>_<Service>_<YYYYMMDDTHHmmss>Z.log`. The file is then moved to the repository database on the central node by the Qlik Sense Repository Service (QRS) and archived.
3. A new log file, named `<MachineName>_<LogType>_<Service>.txt`, is created.



If the .log file is deleted before being copied to the repository database on the central node, the file is lost and cannot be recreated.

The format of the file name is as follows:

- `<MachineName>` = Name of the server where the log was created.
- `<LogType>` = The type of events that are covered by the log.
- `<Service>` = The service that the log originates from (for example, Proxy or Repository).
- `<YYYYMMDDTHHmmss>Z` = Time stamp for when the log file was closed for new entries, where:
 - `YYYY`: Year
 - `MM`: Month
 - `DD`: Day in month
 - `T`: Delimiter, time designator
 - `HH`: Hour
 - `mm`: Minutes
 - `ss`: Seconds
 - `Z`: UTC designator, indicates that the time stamp is in UTC format

7.6 Rows

The first row of each log file contains a header that, in turn, contains the names of all fields, separated by tabs.

Each log entry is one row and the characters listed in the following table are replaced with Unicode characters.

Unicode character replacements

Character	Unicode replacement	Description
<code>\t</code>	<code>\u21d4</code>	Symbol for horizontal tabulation, HT.
<code>\n</code>	<code>\u2193</code>	Symbol for line feed, LF.

Character	Unicode replacement	Description
\f	\u2192	Symbol for form feed, FF.
\r	\u21b5	Symbol for carriage return, CR.

7.7 Fields

This section describes the fields in the Qlik Sense log files.

Audit activity log


The following table lists the fields in the audit activity log, `<MachineName>_AuditActivity_<Service>.txt`.



The Audit activity log does not include a Severity field. This is because all rows in the log have the same log level.

Audit activity log fields

Field	Format	Description
Sequence#	Int	1 - 2147483647 by default, but can be configured using custom logging as described in <i>Appendix (page 281)</i> . Each row in the log starts with a sequence number that is used to ensure that the log is not tampered with (that is, that no rows are inserted or deleted). The sequence number wraps a) when the last sequence number is reached, or b) when the logging, for some reason, is restarted without the last sequence number being reached.
ProductVersion	String	The version number of the Qlik Sense service (for example, 1.2.1.3).
Timestamp	ISO 8601	Timestamp in ISO 8601 format, <code>YYYYMMDDThhmmss.fffk</code> , where: <ul style="list-style-type: none"> • YYYY: Year • MM: Month • DD: Day in month • T: Delimiter • hh: Hour • mm: Minutes • ss: Seconds • fff: Milliseconds • k: Time zone offset For example, <code>20110805T145657.000+0200</code> means year 2011, month 8, day 5 at 14:56:57 GMT+2.
Hostname	String	The name of the server.
Id	String	A unique identifier of the log entry (added by Log4net).

Field	Format	Description
Description	String	<p>A human-readable message that summarizes the action in the system.</p> <p>Format:</p> <p>Command=<CommandName>;Result=<ReturnCode (Int)>;ResultText=<Description, Success, or Error message></p>
ProxySessionId	String	<p>The ID of the proxy session.</p> <p>0 = Internal system command or a command that does not go through the QPS</p> <div style="border: 1px solid gray; padding: 5px; margin-top: 10px;">  <i>The proxy session ID is logged as a salted-hash ID.</i> </div>
ProxyPackageld	String	<p>A unique ID of each HTTP(S) package that passes through the Qlik Sense Proxy Service (QPS).</p> <p>0 = Internal system command or a command that does not go through the QPS</p>
RequestSequenceld	String	<p>The combination of RequestSequenceld and ProxyPackageld is unique for every row in a log file and creates the timeline for the proxy session. The combination also forms a primary key in the log file.</p> <p>The initial RequestSequenceld is an integer. Subrequests are linked to the initial request by adding a dot and an ID for the subrequest:</p> <ul style="list-style-type: none"> • Initial request: RequestSequenceld = 1 <ul style="list-style-type: none"> • Subrequest 1 based on the initial request: RequestSequenceld = 1.0 • Subrequest 2 based on the initial request: RequestSequenceld = 1.1 <p>0 = Internal system command or a command that does not go through the Qlik Sense Engine Service (QES)</p>
UserDirectory	String	The user directory linked to the logged in Qlik Sense user.
UserId	String	<p>The Qlik Sense user that initiated the command.</p> <p>System = Internal system command</p>

Field	Format	Description
ObjectId	String	<p>The internal ID of the object. Used to link system actions to user actions.</p> <p>0 = Cannot get the ID of the object</p> <p>In some cases the ObjectId field contains multiple IDs, separated by the " " (pipe) sign.</p> <p>Example: ObjectId field containing multiple IDs</p> <p>Log event: Start reload task</p> <p>Contents of the ObjectId field: ed5715cd-2d7f-44ec-825f-44084efb3443 d63c7e4e-6089-4314-b60f-ed47ba6c35cc</p> <ul style="list-style-type: none"> • First ID: The ID of the task. • Second ID: The ID of the app.
ObjectName	String	<p>The human-readable name of the object. The ObjectName is linked to the ObjectId.</p> <p>Not available = Cannot link the ObjectName to the ObjectId or the ObjectId is missing</p> <p>In some cases the ObjectName field contains multiple names.</p> <p>Example: ObjectName field containing multiple names</p> <p>Log event: Start reload task</p> <p>Contents of the ObjectName field: MyReload MyApp</p> <ul style="list-style-type: none"> • First identifier (MyReload): The name of the task. • Second identifier (MyApp): The name of the app. <p>The list of ObjectNames always matches the list of ObjectIds, meaning that the ObjectName in the first position is identified by the ID in the corresponding position of the ObjectId field. In this example the following IDs apply (see also the description of the ObjectId field):</p> <ul style="list-style-type: none"> • MyReload = ed5715cd-2d7f-44ec-825f-44084efb3443 • MyApp = d63c7e4e-6089-4314-b60f-ed47ba6c35cc
Service	String	The Qlik Sense service on the server that hosts the process.

Field	Format	Description
Origin	String	The origin of the request: <ul style="list-style-type: none"> • AppAccess • ManagementAccess • Not available
Context	String	The context of the command. The context can be a URL that is linked to the command or a short version of the module path linked to the command.
Command	String	The core name of the use case or system command.
Result	String	Return code: <ul style="list-style-type: none"> • 0, 200 - 226: Success • Any other number: Error
Message	String	Text that describes the log entry. If the request is successful, this field contains "success".
Id2	String	A unique row identifier (the checksum is added by Log4Net).

Audit security log

The following table lists the fields in the audit security log, `<MachineName>_AuditSecurity_<Service>.txt`.




This log is not available for the Qlik Sense Engine Service (QES).



The Audit security log does not include a Severity field. This is because all rows in the log have the same log level.

Audit security log fields

Field	Format	Description
Sequence#	Int	1 - 2147483647 by default, but can be configured using custom logging as described in <i>Appenders (page 281)</i> . Each row in the log starts with a sequence number that is used to ensure that the log is not tampered with (that is, that no rows are inserted or deleted). The sequence number wraps a) when the last sequence number is reached, or b) when the logging, for some reason, is restarted without the last sequence number being reached.
ProductVersion	String	The version number of the Qlik Sense service (for example, 1.2.1.3).

Field	Format	Description
Timestamp	ISO 8601	<p>Timestamp in ISO 8601 format, <code>YYYYMMDDThhmmss.fffk</code>, where:</p> <ul style="list-style-type: none"> • YYYY: Year • MM: Month • DD: Day in month • T: Delimiter • hh: Hour • mm: Minutes • ss: Seconds • fff: Milliseconds • k: Time zone offset <p>For example, <code>20110805T145657.000+0200</code> means year 2011, month 8, day 5 at 14:56:57 GMT+2.</p>
HostName	String	The name of the server.
Id	GUID	A unique identifier of the log entry (added by Log4net).
Description	String	<p>A human-readable message that summarizes the action in the system.</p> <p>Format:</p> <p>Command=<CommandName>;Result=<ReturnCode (Int)>;ResultText=<Description, Success, or Error message></p>
ProxySessionId	String	<p>The ID of the proxy session.</p> <p>0 = Internal system command or a command that does not go through the QPS</p> <div style="border: 1px solid gray; padding: 5px; margin-top: 10px;">  <i>The proxy session ID is logged as a salted-hash ID.</i> </div>
ProxyPackageId	String	<p>A unique ID of each HTTP(S) package that passes through the Qlik Sense Proxy Service (QPS).</p> <p>0 = Internal system command or a command that does not go through the QPS</p>

Field	Format	Description
RequestSequenceId	String	<p>The combination of RequestSequenceId and ProxyPackageId is unique for every row in a log file and creates the timeline for the proxy session. The combination also forms a primary key in the log file.</p> <p>The initial RequestSequenceId is an integer. Subrequests are linked to the initial request by adding a dot and an ID for the subrequest:</p> <ul style="list-style-type: none"> Initial request: RequestSequenceId = 1 <ul style="list-style-type: none"> Subrequest 1 based on the initial request: RequestSequenceId = 1.0 Subrequest 2 based on the initial request: RequestSequenceId = 1.1 <p>0 = Internal system command or a command that does not go through the Qlik Sense Engine Service (QES)</p>
UserDirectory	String	<p>The user directory linked to the logged in Qlik Sense user.</p> <p>System = Internal system command</p>
UserId	String	<p>The Qlik Sense user that initiated the command.</p> <p>System = Internal system command</p>
ObjectId	String	<p>The internal ID of the object. Used to link system actions to user actions.</p> <p>0 = Cannot get the ID of the object</p> <p>In some cases the ObjectId field contains multiple IDs, separated by the " " (pipe) sign.</p> <p>Example: ObjectId field containing multiple IDs</p> <p>Log event: Start reload task</p> <p>Contents of the ObjectId field: ed5715cd-2d7f-44ec-825f-44084efb3443 d63c7e4e-6089-4314-b60f-ed47ba6c35cc</p> <ul style="list-style-type: none"> First ID: The ID of the task. Second ID: The ID of the app.

Field	Format	Description
ObjectName	String	<p>The human-readable name of the object. The ObjectName is linked to the ObjectId.</p> <p>Not available = Cannot link the ObjectName to the ObjectId or the ObjectId is missing</p> <p>In some cases the ObjectName field contains multiple names.</p> <p>Example: ObjectName field containing multiple names</p> <p>Log event: Start reload task</p> <p>Contents of the ObjectName field: MyReload MyApp</p> <ul style="list-style-type: none"> • First identifier (MyReload): The name of the task. • Second identifier (MyApp): The name of the app. <p>The list of ObjectNames always matches the list of ObjectIds, meaning that the ObjectName in the first position is identified by the ID in the corresponding position of the ObjectId field. In this example the following IDs apply (see also the description of the ObjectId field):</p> <ul style="list-style-type: none"> • MyReload = ed5715cd-2d7f-44ec-825f-44084efb3443 • MyApp = d63c7e4e-6089-4314-b60f-ed47ba6c35cc
SecurityClass	String	<p>A categorization of the security-related information:</p> <ul style="list-style-type: none"> • Security: Access to resources, authentication, authorization • License: License access, license usage, license allocation • Certificate: Certificate-related information
ClientHostAddress	String	The hostname/IP address of the client.
Service	String	The Qlik Sense service on the server that hosts the process.
Origin	String	<p>The origin of the request:</p> <ul style="list-style-type: none"> • AppAccess • ManagementAccess • Not available
Context	String	<p>The context of the command.</p> <p>The context can be a URL that is linked to the command or a short version of the module path linked to the command.</p>
Command	String	The core name of the use case or system command.


Field	Format	Description
Result	String	Return code: <ul style="list-style-type: none"> • 0, 200 - 226: Success • Any other number: Error
Message	String	Text that describes the log entry. If the request is successful, this field contains "success".
Checksum	ID	Each row has a checksum. The security log file also includes a file signature.

Server log

The following table lists the fields in the service log, `<MachineName>_Service_<Service>.txt`.

Service log fields

Field	Format	Description
Sequence#	Int	1 - 2147483647 by default, but can be configured using custom logging as described in <i>Appendix (page 281)</i> . Each row in the log starts with a sequence number that is used to ensure that the log is not tampered with (that is, that no rows are inserted or deleted). The sequence number wraps a) when the last sequence number is reached, or b) when the logging, for some reason, is restarted without the last sequence number being reached.
ProductVersion	String	The version number of the Qlik Sense service (for example, 1.2.1.3).
Timestamp	ISO 8601	Timestamp in ISO 8601 format, <code>YYYYMMDDThhmmss.fffk</code> , where: <ul style="list-style-type: none"> • YYYY: Year • MM: Month • DD: Day in month • T: Delimiter • hh: Hour • mm: Minutes • ss: Seconds • fff: Milliseconds • k: Time zone offset <p>For example, <code>20110805T145657.000+0200</code> means year 2011, month 8, day 5 at 14:56:57 GMT+2.</p>

Field	Format	Description
Severity	String	<p>Row log level, can be configured using custom logging as described in <i>Appenders (page 281)</i>:</p> <ul style="list-style-type: none"> • Debug: Information useful to developers for debugging purposes. This level is not useful during normal operation as it generates vast amounts of logging information. • Info: Normal operational messages that may be harvested for reporting, measuring throughput, and so on. No action is required. • Warn: Not an error message, but an indication that an error will occur if no action is taken (for example, the file system is 85% full). • Error: Messages regarding unexpected situations and errors that prevent the server from operating normally. • Fatal: Messages that the Qlik Sense service or application has to shut down in order to prevent data loss.
HostName	String	The hostname of the server that runs the process or executes the task.
Id	GUID	A unique identifier of the log entry (added by Log4net).
Description	String	<p>A human-readable message that summarizes the action in the system.</p> <p>Format:</p> <p>Command=<CommandName>;Result=<ReturnCode (Int)>;ResultText=<Description, Success, or Error message></p>
ProxySessionId	String	<p>The ID of the proxy session.</p> <p>0 = Internal system command or a command that does not go through the QPS</p> <div style="border: 1px solid gray; padding: 5px; margin-top: 10px;">  <i>The proxy session ID is logged as a salted-hash ID.</i> </div>
ProxyPackageld	String	<p>A unique ID of each HTTP(S) package that passes through the Qlik Sense Proxy Service (QPS).</p> <p>0 = Internal system command or a command that does not go through the QPS</p>

Field	Format	Description
RequestSequenceId	String	<p>The combination of RequestSequenceId and ProxyPackageId is unique for every row in a log file and creates the timeline for the proxy session. The combination also forms a primary key in the log file.</p> <p>The initial RequestSequenceId is an integer. Subrequests are linked to the initial request by adding a dot and an ID for the subrequest:</p> <ul style="list-style-type: none"> Initial request: RequestSequenceId = 1 <ul style="list-style-type: none"> Subrequest 1 based on the initial request: RequestSequenceId = 1.0 Subrequest 2 based on the initial request: RequestSequenceId = 1.1 <p>0 = Internal system command or a command that does not go through the Qlik Sense Engine Service (QES)</p>
UserDirectory	String	<p>The user directory linked to the logged in Qlik Sense user.</p> <p>System = Internal system command</p>
UserId	String	<p>The Qlik Sense user that initiated the command.</p> <p>System = Internal system command</p>
ObjectId	String	<p>The internal ID of the object. Used to link system actions to user actions.</p> <p>0 = Cannot get the ID of the object</p> <p>In some cases the ObjectId field contains multiple IDs, separated by the " " (pipe) sign.</p> <p>Example: ObjectId field containing multiple IDs</p> <p>Log event: Start reload task</p> <p>Contents of the ObjectId field: ed5715cd-2d7f-44ec-825f-44084efb3443 d63c7e4e-6089-4314-b60f-ed47ba6c35cc</p> <ul style="list-style-type: none"> First ID: The ID of the task. Second ID: The ID of the app.

Field	Format	Description
ObjectName	String	<p>The human-readable name of the object. The ObjectName is linked to the ObjectId.</p> <p>Not available = Cannot link the ObjectName to the ObjectId or the ObjectId is missing</p> <p>In some cases the ObjectName field contains multiple names.</p> <p>Example: ObjectName field containing multiple names</p> <p>Log event: Start reload task</p> <p>Contents of the ObjectName field: MyReload MyApp</p> <ul style="list-style-type: none"> • First identifier (MyReload): The name of the task. • Second identifier (MyApp): The name of the app. <p>The list of ObjectNames always matches the list of ObjectIds, meaning that the ObjectName in the first position is identified by the ID in the corresponding position of the ObjectId field. In this example the following IDs apply (see also the description of the ObjectId field):</p> <ul style="list-style-type: none"> • MyReload = ed5715cd-2d7f-44ec-825f-44084efb3443 • MyApp = d63c7e4e-6089-4314-b60f-ed47ba6c35cc
Service	String	The Qlik Sense service on the server that hosts the process.
Origin	String	<p>The origin of the request:</p> <ul style="list-style-type: none"> • AppAccess • ManagementAccess • Not available
Context	String	<p>The context of the command.</p> <p>The context can be Internal System command or User Activity command (based on URL for the command).</p>
Command	String	The core name of the use case or system command.
Result	Int	<p>Return code:</p> <ul style="list-style-type: none"> • 0, 200 - 226: Success • Any other number: Error
Message	String	Text that describes the log entry. If the request is successful, this field contains "success".
Id2	String	A unique row identifier (the checksum is added by Log4Net).

Qlik Sense engine service log fields

The following table lists the fields that are unique for the Qlik Sense Engine Service (QES) logs.

QES log fields

Field	Format	Description
EngineTimestamp	ISO 8601	<p>The date and time when the QES wrote the log message to file.</p> <p>Timestamp in ISO 8601 format, <code>YYYYMMDDThhmmss.fffk</code>, where:</p> <ul style="list-style-type: none"> • <code>YYYY</code>: Year • <code>MM</code>: Month • <code>DD</code>: Day in month • <code>T</code>: Delimiter • <code>hh</code>: Hour • <code>mm</code>: Minutes • <code>ss</code>: Seconds • <code>fff</code>: Milliseconds • <code>k</code>: Time zone offset <p>For example, <code>20110805T145657.000+0200</code> means year 2011, month 8, day 5 at 14:56:57 GMT+2.</p>
EngineVersion	String	The version number of the QES that executed the request.

7.8 Trace logs

The legacy logging framework is still available in Qlik Sense, but the logs are as of Qlik Sense version 2.0 referred to as trace logs. The log files remain the same, in the old logging format, but they are stored in a new location.

Storage

The trace log files are stored in the `%ProgramData%\Qlik\Sense\Log\<Service>\Trace` folder.

Naming

The trace log files are named in accordance to the following file rollover procedure:

1. The log is stored in a file named `<MachineName>_<Facility>_<Service>.txt`.
2. When the file is full or a pre-defined amount of time has passed, the file extension is automatically changed to `.log` and a time stamp is appended to the file name for uniqueness and archiving. This means that the new file name becomes `<MachineName>_<Facility>_<Service>_<YYYY-MM-DDTHH.mm.ss>Z.log`. The file is then moved to the repository database on the central node by the Qlik Sense Repository Service (QRS) and archived.
3. A new log file, named `<MachineName>_<Facility>_<Service>.txt`, is created.



If the `.log` file is deleted before being copied to the repository database on the central node, the file is lost and cannot be recreated.

The format of the file name is as follows:

- `<Machine>` = Name of the server where the log was created.
- `<Facility>` = The type of events that are covered by the log.
Logger (page 273)
- `<Service>` = The service that the log originates from (for example, Proxy or Repository).
- `<YYYY-MM-DDTHH.mm.ss>Z` = Time stamp for when the log file was closed for new entries, where:
 - `YYYY`: Year
 - `MM`: Month
 - `DD`: Day in month
 - `T`: Delimiter, time designator
 - `HH`: Hour
 - `mm`: Minutes
 - `ss`: Seconds
 - `Z`: UTC designator, indicates that the time stamp is in UTC format

See also:

 *Logger (page 273)*

Rows

The first row of each log file contains a header that, in turn, contains the names of all fields, separated by tabs.

Each log entry is one row and the characters listed in the following table are replaced with Unicode characters.

Unicode replacements for characters

Character	Unicode replacement	Description
<code>\t</code>	<code>\u21d4</code>	Symbol for horizontal tabulation, HT.
<code>\n</code>	<code>\u2193</code>	Symbol for line feed, LF.
<code>\f</code>	<code>\u2192</code>	Symbol for form feed, FF.
<code>\r</code>	<code>\u21b5</code>	Symbol for carriage return, CR.

Fields


This section describes the fields in the trace log files.

Common fields




The following table lists the fields (in order of appearance) included in all trace log files.

Common trace log file fields

Field	Description
Sequence#	1 - 2147483647 by default, but can be configured using custom logging as described in <i>Qlik Sense Appenders (page 281)</i> . Each row in the log starts with a sequence number that is used to ensure that the log is not tampered with (that is, that no rows are inserted or deleted). The sequence number wraps either when the last sequence number is reached or when the logging, for some reason, is restarted without the last sequence number being reached.
Timestamp	Timestamp in ISO 8601 format, <code>YYYYMMDDThhmmss.fffk</code> , where: <ul style="list-style-type: none">• <code>YYYY</code>: Year• <code>MM</code>: Month• <code>DD</code>: Day in month• <code>T</code>: Delimiter• <code>hh</code>: Hour• <code>mm</code>: Minutes• <code>ss</code>: Seconds• <code>fff</code>: Milliseconds• <code>k</code>: Time zone offset For example, <code>20110805T145657.000+0200</code> means year 2011, month 8, day 5 at 14:56:57 GMT+2.

Field	Description
Level	<p>Row log level, can be configured using custom logging as described in Qlik Sense <i>Appenders (page 281)</i>:</p> <ul style="list-style-type: none"> • Debug: Information useful to developers for debugging purposes. This level is not useful during normal operation since it generates vast amounts of logging information. • Info: Normal operational messages that may be harvested for reporting, measuring throughput, and so on. No action required. • Warn: Not an error message, but an indication that an error may occur, if no action is taken (for example, the file system is 85% full). Each item must be resolved within a given time. • Error: Non-urgent failures that are relayed to developers or administrators. Each item must be resolved within a given time. • Fatal: Indicates a failure in a primary system (for example, loss of primary ISP connection) and must be corrected immediately. • Off: No logs, except for license logs, are produced. <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;">  <i>Starting with the Qlik Sense May 2021 release all execution completion log entries are logged as Info instead of Debug.</i> </div>
Hostname	Server name.

Field	Description
Logger	<p>Logger in <Facility>.<Service>.<Fully qualified name of class> format, where:</p> <ul style="list-style-type: none"> • <Facility>: <ul style="list-style-type: none"> • Application: Log events that are related to the app running in Qlik Sense. • Audit: Log events that provide an audit trail of a user's activities and administration of the Qlik Sense platform. • Exit: Log events that are related to the shutdown of the Qlik Sense Engine Service (QES). • License: Log events that are related to the Qlik Sense license. • ManagementConsole: Log events that are related to the Qlik Management Console (QMC). • Performance: Log events that are related to the performance of the Qlik Sense platform or app. • QixPerformance: Log events that are related to the performance of the QIX protocol in the QES. • Security: Log events that are related to security issues. • Session: Log events that are related to the termination of a proxy session. • SSE: Log events that are related to server-side extensions. • Synchronization: Log events that are related to the synchronization of the Qlik Sense Repository Service (QRS) instances in a multi-node site. • System: Log events that are related to the Qlik Sense platform and not to the app running on the platform (for example, log messages related to the QMC, QRS, Qlik Sense Proxy Service (QPS), and so on). • TaskExecution: Log events that are related to the execution of tasks by the Qlik Sense Scheduler Service (QSS). • Traffic: Log events that are related to debugging. • UserManagement: Log events that are related to the management of the users. • <Service>: The Qlik Sense service that the log originates from (for example, QRS or QPS). • <Fully qualified name of class>: Indicates the part of the service that generated the log message.
Thread	Thread name or Managed Thread ID (if available).
Id	Globally Unique Identifier (GUID) for the log message.
ServiceUser	Name of the user or account used by the Qlik Sense service.
Message	Log message.

Field	Description
Exception	Exception message.  <i>This field is only present when there is an exception message.</i>
StackTrace	A trace to the place in Qlik Sense where the exception occurred.  <i>This field is only present when the Exception field is present.</i>
ProxySessionId	The ID of the proxy session for the user.  <i>This field is not present in all log files.</i>
Id2 or Checksum	The last field in a log entry either contains an Id2 or a Checksum: <ul style="list-style-type: none"> • Id2: Log message GUID (same as Id described earlier). This is the normal value. • Checksum: To protect logs that contain sensitive information (for example, audit, security, and license logs) from tampering, the last field in such log entries contains a cryptographic hash of the entire row up to the hash itself.

Additional fields

The common fields are present in all trace log files. Some trace logs contain additional fields, which are listed in this section. In addition, optional fields can be defined.

Application log

Qlik Sense Repository Service (QRS)


The following fields are specific to the Application log for the QRS:

- Application: The name of the application (if there is a name to associate with the log entry).

Qlik Sense Scheduler Service (QSS)

The following fields are specific to the Application log for the QSS:

- Application: The name of the application (if there is a name to associate with the log entry).

 [Common fields \(page 271\)](#)

Audit log

Qlik Sense Repository Service (QRS)

The following fields are specific to the Audit log for the QRS:

- Action: The action that the user performed (add, update, delete, export).
- ActiveUserDirectory: The user directory for the user.

- **ActiveUserId:** The ID of the user.
- **ResourceId:** The ID of the resource on which the user performed the action.

Qlik Sense Proxy Service (QPS)


The following fields are specific to the Audit log for the QPS:

- **ConnectionId:** The ID of the connection.
ActiveConnections field in *Performance log* (page 276)
- **ActiveUserDirectory:** The user directory for the user.
- **ActiveUserId:** The ID of the user.
- **TicketId:** The ID of the login ticket that was issued for the user. The ticket ID exists until it is consumed by the QPS.
- **IpAddress:** The IP address of the client.
- **AppId:** The ID of the app (empty if no app is loaded).
- **TargetHost:** The call from the client is forwarded to a Qlik Sense Engine Service (QES) or QRS. This field contains the name of the machine on which the service is running.
- **VirtualProxy:** The virtual proxy prefix in {prefix} format.

Qlik Sense Engine Service (QES)

The following fields are specific to the Audit log for the QES:

- **ActiveUserDirectory:** The user directory for the user.
- **ActiveUserId:** The ID of the user.
- **EngineTimestamp:** The time when the QES wrote the log message to file.
- **EngineThread:** The ID of the thread that was used when the QES wrote the log message to file.
- **ProcessId:** The ID of the QES process from which the log message originates.
- **ServerStatus:** The time when the QES started.
- **AppId:** The ID of the app.
- **Type:** The type of operation that the user performed to generate the audit message.
- **Qlik Sense User:** The user who generated the audit message.


 *Common fields* (page 271)

License log

Qlik Sense Repository Service (QRS)

The following fields are specific to the License log for the QRS:

- **AccessTypeId:** The ID of the access type entity.
- **AccessType:** The name of the access type (LoginAccess or UserAccess).
- **Operation:** The operation that was performed (Add, Update, Delete, UsageGranted, UsageDenied, Available, Timedout, or Unquarantined).
- **UserName:** The name of the user (who, for example, uses an access pass).
- **UserId:** The ID of the user in Qlik Sense.

 *Common fields* (page 271)

Performance log

Qlik Sense Repository Service (QRS)

The following fields are specific to the Performance log for the QRS:

- Tracenumber: A unique ID for the call to the QRS REST API.
- Httpmethod: The HTTP method that was used (Get, Put, Post, or Delete).
- Url: The URL that was used.
- Resourcetype: The type of resource.
- Method: The backend code that was called.
- Elapsedmilliseconds: The time (in milliseconds) to complete the call to the QRS REST API.

Example: Get `http://mytest/cars/4`

- Httpmethod: Get
- Url: `http://mytest/cars/4`
- Resourcetype: cars
- Method: `get/cars/{0}`

Qlik Sense Proxy Service (QPS)

The following fields are specific to the Performance log for the QPS:

- ActiveConnections: The number of active connections (in any form or shape) from the client.
A connection is a stream (that is, a socket) between a Qlik Sense client and the Qlik Sense Proxy Service (QPS). This stream is often connected to another stream, which runs from the QPS to the Qlik Sense Repository Service (QRS) or the Qlik Sense Engine Service (QES). The two streams allow the client to communicate with the QRS or the QES.
- ActiveStreams: The number of active data streams (that is, sockets), either from the browser to the QPS or from the QPS to the QRS or the QES.
- ActiveSessions: The number of active sessions in the QPS.
A Qlik Sense user gets a proxy session when the user has been authenticated. The session is terminated after a certain period of inactivity.
- LoadBalancingDecisions: The number of users who currently have at least one engine session.
- PrintingLoadBalancingDecisions: The number of users who have been load balanced to the Qlik Sense Printing Service (QPR).
- Tickets: The number of issued login tickets that have not yet been consumed.
- ActiveClientWebsockets: The number of active WebSockets between the client and the QPS.
- ActiveEngineWebsockets: The number of active WebSockets between the QPS and the target Qlik Sense service.



The logging entries are also available as metrics; see [Proxy service](#).

Qlik Sense Engine Service (QES)

Each entry (that is, row) in the Performance log corresponds to a snapshot (that is, a number of measurements) of the performance of the QES at the given point in time.

The following fields are specific to the Performance log for the QES:

- **ActiveUserDirectory:** The user directory for the user.
- **ActiveUserId:** The ID of the user.
- **EngineTimestamp:** The time when the QES wrote the log message to file.
- **EngineThread:** The ID of the thread that was used when the QES wrote the log message to file.
- **ProcessId:** The ID of the QES process from which the log message originates.
- **Exe Type:** The configuration type (release or debug version) of the QES process.
- **Exe Version:** The version number of the QES process.
- **Server Started:** The time when the QES started.
- **Entry Type:** The reason (Server Starting, Normal, or Server Shutting Down) for the log entry in the Performance log.
- **ActiveDocSessions:** The number of active engine sessions at the given point in time.
- **DocSessions:** The number of engine sessions at the given point in time.
- **ActiveAnonymousDocSessions:** The number of active anonymous engine sessions at the given point in time.
- **AnonymousDocSessions:** The number of anonymous engine sessions at the given point in time.
- **ActiveTunneledDocSessions:** The number of active tunneled engine sessions at the given point in time.
- **TunneledDocSessions:** The number of tunneled engine sessions at the given point in time.
- **DocSessionStarts:** The number of started engine sessions since the previous snapshot.
- **ActiveDocs:** The number of active apps in the QES at the given point in time.
- **RefDocs:** The number of apps in the QES at the given point in time.
- **LoadedDocs:** The number of loaded apps in the QES at the given point in time.
- **DocLoads:** The number of app loads in the QES since the previous snapshot.
- **DocLoadFails:** The number of failed app loads in the QES since the previous snapshot.
- **Calls:** The number of calls to the QES since the previous snapshot.
- **Selections:** The number of selections in the QES since the previous snapshot.
- **ActiveIpAdrrs:** The number of IP addresses of active connected clients in the QES at the given point in time.
- **IpAdrrs:** The number of IP addresses of all connected clients in the QES at the given point in time.
- **ActiveUsers:** The number of active users in the QES at the given point in time.
- **Users:** The total number of users in the QES at the given point in time.
- **CPUload:** A measurement of the load on the CPU on which the QES runs at the given point in time.
- **VMCommitted(MB):** The committed Virtual Memory (in megabytes) at the given point in time.
- **VMAllocated(MB):** The allocated Virtual Memory (in megabytes) at the given point in time.
- **VMFree(MB):** The freed Virtual Memory (in megabytes) at the given point in time.

- `VMLargestFreeBlock(MB)`: The largest freed Virtual Memory block (in megabytes) at the given point in time.


 *Common fields (page 271)*

QIX performance log

Qlik Sense Engine Service (QES)

The following fields are specific to the QIX performance log for the QES:

- `ActiveUserDirectory`: The user directory for the user.
- `ActiveUserId`: The ID of the user.
- `EngineTimestamp`: The time when the QES wrote the log message to file.
- `EngineThread`: The ID of the thread that was used when the QES wrote the log message to file.
- `ProcessId`: The ID of the QES process from which the log message originates.
- `SessionId`: The ID of the engine session for which the QIX method call was made.
- `CServerId`: The ID of the server instance that handled the request.
- `Server Started`: The time when the QES started.
- `Method`: The name of the QIX method that was called.
- `RequestId`: The ID of the request in which the QIX method call was handled.
- `Target`: The memory address of the target for the QIX method call.
- `RequestException`: The ID of an exception (if any) that occurred as a result of the QIX method call.
- `AnyException`: The exception code if the request fails.
- `ProcessTime`: The amount of time that was needed to process the request.
- `WorkTime`: The amount of time that the request did actual work.
- `LockTime`: The amount of time that the request had to wait for an internal lock.
- `ValidateTime`: The amount of time that the request used for validation.
- `TraverseTime`: The amount of time the request uses for the traverse part of the calculation.
- `Handle`: The ID of the interface that handled the request. The interface can be Global, a certain sheet, a certain object, or similar.
- `ApplId`: The ID of the application.
- `ObjectId`: The ID of the object.
- `ObjectType`: The object type.
- `NetRam`: The amount of memory used for the calculation.
- `PeakRam`: The peak amount of memory used for the calculation.

 *Common fields (page 271)*

Qlik Management Consolelog



The Qlik Management Console log is not created until there is an event (for example, an error message) for the Qlik Management Console (QMC) to write in the log.

Qlik Sense Repository Service (QRS)

The following fields are specific to the Qlik Management Console log for the QRS:

- Browser: The web browser that is used to run the QMC.

 *Common fields (page 271)*

Server-side extension log

Qlik Sense Engine Service (QES)

The following fields are specific to the server-side extension (SSE) log for the QES:

- **ActiveUserDirectory:** The user directory for the user.
- **ActiveUserId:** The ID of the user.
- **EngineTimestamp:** The time when the QES wrote the log message to file.
- **EngineThread:** The ID of the thread that was used when the QES wrote the log message to file.
- **ProcessId:** The ID of the QES process from which the log message originates.
- **QixRequestId:** The ID established by the initiator of the request. If this member is not present, the RPC call is assumed to be a notification.
- **AppId:** The ID of the app that includes the call to the server-side extension (SSE) plugin through an analytic connection.
- **App Title:** The title of the app that includes the call to the SSE plugin through an analytic connection.
- **SSEPlugin:** If the log message was created during a call to the SSE plugin, the mapping/alias of that plugin, for example, SSEPython for a Python plugin. If the log message was created without a call to the SSE plugin, for example, while initializing the SSE, the value is a dash (-).
- **SSEPluginAddress:** Two elements separated by a colon that define the analytic connection to the SSE plugin.
 - **<Host>:** DNS name (or IP-address) of the plugin.
 - **<Port>:** Port on which the plugin listens, typically 50051.

For example, localhost:50051.

 *Common fields (page 271)*


Session log

Qlik Sense Engine Service (QES)

The following fields are specific to the Session log for the QES:

- **ActiveUserDirectory:** The user directory for the user.
- **ActiveUserId:** The ID of the user.
- **EngineTimestamp:** The time when the QES wrote the log message to file.
- **EngineThread:** The ID of the thread that was used when the QES wrote the log message to file.
- **ProcessId:** The ID of the QES process from which the log message originates.
- **Exe Version:** The version number of the QES process.
- **Server Started:** The time when the QES started.
- **AppId:** The ID of the app that was loaded by the finished engine session.
- **App Title:** The title of the loaded app that was used during the finished engine session.
- **Doc Timestamp:** The last modified timestamp of the app that was loaded by the finished engine session.
- **Exit Reason:** The reason for the engine session to finish.

- **Session Start:** The time when the engine session started.
- **Session Duration:** The duration (in days) of the finished engine session.
- **CPU Spent (s):** The CPU time (in seconds) that was spent handling requests during the finished engine session.
- **Bytes Received:** The number of bytes of data that were received during the engine session.
- **Bytes Sent:** The number of bytes of data that were sent during the engine session.
- **Calls:** The number of calls that were made during the engine session.
- **Selections:** The number of selections that were made during the engine session.
- **Authenticated User:** The authenticated user that used the engine session.
- **Secure Protocol:** An on/off flag that indicates whether the protocol was run over a secure connection.

 *Common fields (page 271)*

System log

Qlik Sense Scheduler Service (QSS)


The following fields are specific to the System log for the QSS:

- **TaskName:** The name of the task that was executed.
- **TaskId:** The ID of the task that was executed.
- **User:** The name of the user who executed the task. When the QSS starts a scheduled execution of a task, the QSS is listed as user.
- **ExecutionId:** A unique ID that identifies the execution of the task. A task gets a new ExecutionId every time it is executed.
- **AppName:** The name of the app that executed the task (if any).
- **AppId:** The ID of the app that executed the task (if any).

Qlik Sense Engine Service (QES)

The following fields are specific to the System log for the QES:

- **ActiveUserDirectory:** The user directory for the active user who was logged in when the log message was generated in the QES.
- **ActiveUserId:** The user ID for the active user who was logged in when the log message was generated in the QES.
- **EngineTimestamp:** The time when the QES wrote the log message to file.
- **EngineThread:** The ID of the thread that was used when the QES wrote the log message to file.
- **ProcessId:** The ID of the QES process from which the log message originates.
- **Server Started:** The time when the QES started.

 *Common fields (page 271)*


Task execution log

Qlik Sense Scheduler Service (QSS)

The following fields are specific to the Task execution log for the QSS:

- **TaskId:** A unique ID of the task that was executed.
- **TaskName:** The name of the task that was executed.

- **AppId:** A unique ID of the app that executed the task (if any).
- **AppName:** The name of the app that executed the task (if any).
- **ExecutionId:** A unique ID that identifies the execution of a task. A task gets a new ExecutionId every time it is executed.
- **ExecutionNodeId:** A unique ID that identifies the node in the site on which the specific execution of the task was performed.
- **Status:** The result of the execution of the task (successful, failed, aborted, skipped, or retry).
- **StartTime:** The time when the execution of the task started.
- **StopTime:** The time when the execution of the task stopped.
- **Duration:** The time (in milliseconds) for the execution of the task to be completed.
- **FailureReason:** Empty, unless an error occurred during the execution of the task.


 *Common fields (page 271)*

Traffic log

Qlik Sense Engine Service (QES)

The following fields are specific to the traffic log for the QES:

- **ActiveUserDirectory:** The user directory for the user.
- **ActiveUserId:** The ID of the user.
- **EngineTimestamp:** The time when the QES wrote the log message to file.
- **EngineThread:** The ID of the thread that was used when the QES wrote the log message to file.
- **ProcessId:** The ID of the QES process from which the log message originates.

 *Common fields (page 271)*

7.9 Configuring the logging

The standard logging in Qlik Sense is managed from the Qlik Management Console (QMC) for each Qlik Sense service.

Services (page 30)

Customized logging is setup using appenders and the local log configuration file, *LocalLogConfig.xml*.

Appenders (page 281)

Appenders

The logging in Qlik Sense implements a custom appender, *QSRollingFileAppender*, which is based on the *log4net* component. The custom appender is used internally by the Qlik Sense logging system.

QSRollingFileAppender and some of the built-in, predefined appenders in the *log4net* framework can be used to configure customized logging, which is specified in the local log configuration file, *LocalLogConfig.xml*.

QSRollingFileAppender can also log events in the local log file (for example, the Microsoft Windows event log) or send log information to a remote log server.

QSRollingFileAppender

QSRollingFileAppender inherits from `log4net.Appenders.FileAppender` and all parameters, except for `AppendToFile`, are also available to QSRollingFileAppender. QSRollingFileAppender stores the log files in accordance to the `MaxFileSize` and `MaxFileTime` parameters.

Configuring the appender

The QSRollingFileAppender configuration is as follows:

```
<appender name="MyQSRollingFileAppender"
type="Qlik.Sense.Logging.log4net.Appender.QSRollingFileAppender">
<param name="threshold" value="info" />
<param name="encoding" value="utf-8" />
<param name="file" value="C:/ProgramData/Qlik/Sense/Log/output.log"/>
<param name="maximumfiletime" value="720" />
<param name="maximumfilesize" value="512KB" />
<layout type="log4net.Layout.PatternLayout">
<converter>
  <param name="name" value="rownum" />
  <param name="type" value="Qlik.Sense.Logging.log4net.Layout.Pattern.CounterPatternConverter"
/>
</converter>
<converter>
  <param name="name" value="longIso8601date" />
  <param name="type"
value="Qlik.Sense.Logging.log4net.Layout.Pattern.Iso8601TimeOffsetPatternConverter" />
</converter>
<converter>
  <param name="name" value="hostname" />
  <param name="type"
value="Qlik.Sense.Logging.log4net.Layout.Pattern.HostNamePatternConverter" />
</converter>
<converter>
  <param name="name" value="guid" />
  <param name="type" value="Qlik.Sense.Logging.log4net.Layout.Pattern.GuidPatternConverter" />
</converter>
<converter>
  <param name="name" value="user" />
  <param name="type"
value="Qlik.Sense.Logging.log4net.Layout.Pattern.ServiceUserNameCachedPatternConverter" />
</converter>
<converter>
  <param name="name" value="encodedmessage" />
  <param name="type"
value="Qlik.Sense.Logging.log4net.Layout.Pattern.EncodedMessagePatternConverter" />
</converter>
<converter>
  <param name="name" value="encodedexception" />
  <param name="type"
value="Qlik.Sense.Logging.log4net.Layout.Pattern.EncodedExceptionPatternConverter" />
</converter>
<param name="ignoresexception" value="false" />
<param name="header"
```

```

value="Sequence##x9;Timestamp##x9;Level##x9;Hostname##x9;Logger##x9;Thread##x9;Id##x9;User##x
9;
Message##x9;Exception##x9;Id2##xD;##xA;" />
<param name="conversionpattern" value="%rownum
{9999}##x9;%longIso8601date##x9;%level##x9;%hostname##x9;%logger##x9;%thread##x9;
%guid##x9;%user##x9;%encodedmessage##x9;%encodedexception{innermostmessage}##x9;%guid%newline"
/>
</layout>
</appender>

```

Converters

log4net.Layout.PatternLayout and a couple of custom converters are used to format rows in logs based on QSRollingFileAppender:

- Qlik.Sense.Logging.log4net.Layout.Pattern.CounterPatternConverter: Add a sequence number to the log row. Parameter:
 - Integer: The last number of the sequence before it is reset.
- Qlik.Sense.Logging.log4net.Layout.Pattern.Iso8601TimeOffsetPatternConverter: Add a time stamp (local time with time offset in ISO 8601 format) to the log row.
- Qlik.Sense.Logging.log4net.Layout.Pattern.HostNamePatternConverter: Add the host name to the log row.
- Qlik.Sense.Logging.log4net.Layout.Pattern.GuidPatternConverter: Add a GUID to the log row.
- Qlik.Sense.Logging.log4net.Layout.Pattern.ServiceUserNameCachedPatternConverter: Add the username to the log row.
- Qlik.Sense.Logging.log4net.Layout.Pattern.EncodedMessagePatternConverter: Add the encoded message to the log row.
- Qlik.Sense.Logging.log4net.Layout.Pattern.EncodedExceptionPatternConverter: Add information on the logged exception to the log row. Parameter (one of the following):
 - MESSAGE: The message in the logged exception.
 - INNERMOSTMESSAGE: The message in the innermost exception of the logged exception.
 - SOURCE: The source of the exception (that is, the name of the app or the object that caused the error).
 - STACKTRACE: The stack trace for the exception.
 - TARGETSITE: The target site for the exception (that is, the method that threw the current exception).
 - HELPLINK: Link to the help file associated with the exception.

Built-in log4net appenders

In addition to the Qlik Sense custom appender, QSRollingFileAppender, the log4net framework comes with a set of built-in predefined appenders that also can be used in the local log configuration file,


LocalLogConfig.xml:

- AdoNetAppender
- AnsiColorTerminalAppender
- AspNetTraceAppender
- ColoredConsoleAppender

- ConsoleAppender
- EventLogAppender
- FileAppender
- NetSendAppender
- RemoteSyslogAppender
- RemotingAppender
- RollingFileAppender
- SmtpAppender
- SmtpPickupDirAppender
- TelnetAppender
- UdpAppender

Each appender has its own set of parameters to control the output.

See also:

 [Apache Logging Services](#)

Example: EventLogAppender

The following example shows how to use the EventLogAppender in the local log configuration file, *LocalLogConfig.xml*, for the Qlik Sense Proxy Service (QPS). In the example, all QPS audit log entries at warning level are sent to the Microsoft Windows event log.

```
<appender name="EventLogAppender" type="log4net.Appender.EventLogAppender" >
  <param name="threshold" value="warn" />
  <param name="applicationName" value="Qlik Sense Proxy Service" />
  <layout type="log4net.Layout.PatternLayout">
    <param name="conversionPattern" value="%message" />
  </layout>
</appender>
<logger name="Audit.Proxy">
  <appender-ref ref="EventLogAppender" />
</logger>
```

Example: SmtpAppender

The following example shows how to use the SmtpAppender in the local log configuration file, *LocalLogConfig.xml*, for the Qlik Sense Proxy Service (QPS). In the example, all QPS audit log entries at warning level are sent to an email address (to@domain.com).

```
<appender name="MyMailAppender" type="log4net.Appender.SmtpAppender">
<param name="threshold" value="warn" />
<param name="to" value="to@domain.com" />
<param name="from" value="from@domain.com" />
<param name="subject" value="test logging message" />
<param name="smtpHost" value="SMTPServer.domain.com" />
<param name="port" value="25" />
<param name="bufferSize" value="512" />
<param name="lossy" value="true" />
<layout type="log4net.Layout.PatternLayout">
<param name="conversionPattern" value="%newline%date %-5level
```

```

%message%newline%newline%newline" />
</layout>
</appender>
    <logger name="Audit.Proxy">
        <appender-ref ref="MyMailAppender" />
    </logger>

```

Local log configuration file

The logging in Qlik Sense can be set up to produce customized logging as an addition to the default logging.

To set up customized logging, create a local log configuration file named *LocalLogConfig.xml* in the *%ProgramData%\Qlik\Sense\<Service>* folder.



The logging defined by the local log configuration file does not affect the default logging.

Requirements

The requirements described in this section must be fulfilled for the customized logging to function properly.

Conforming to the XML schema

The local log configuration file must conform to the XML schema because the Qlik Sense Repository Service (QRS), Qlik Sense Proxy Service (QPS), and Qlik Sense Scheduler Service (QSS) have built-in schema validation.

If the local log configuration file is not accepted by the services, an error is logged in the System log.

Maximum file size

The size of the local log configuration file must not exceed 1 MB.

XML schema

The XML schema for the local log configuration file, *LocalLogConfig.xml*, is as follows:

```

<?xml version="1.0" encoding="utf-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

    <xs:complexType name="ParamType">
        <xs:attribute name="name" type="xs:string" use="required"/>
        <xs:attribute name="value" type="xs:string" use="required"/>
    </xs:complexType>

    <xs:simpleType name="AppenderNameType">
        <xs:restriction base="xs:string">
            <xs:pattern value="^[^$].*" /> <!-- '$' is not allowed as prefix-->
        </xs:restriction>
    </xs:simpleType>

    <xs:complexType name="ConverterType">
        <xs:sequence>
            <xs:element name="param" minOccurs="0" maxOccurs="unbounded" type="ParamType" />
        </xs:sequence>
    </xs:complexType>

    <xs:complexType name="FilterType">
        <xs:sequence>

```

```

        <xs:element name="param" minOccurs="0" maxOccurs="unbounded" type="ParamType" />
    </xs:sequence>
    <xs:attribute name="class" type="xs:string" use="optional"/> <!-- log4cxx-->
    <xs:attribute name="type" type="xs:string" use="optional"/> <!-- log4net-->
</xs:complexType>

<xs:complexType name="EvaluatorType">
    <xs:sequence>
        <xs:element name="param" minOccurs="0" maxOccurs="unbounded" type="ParamType" />
    </xs:sequence>
    <xs:attribute name="class" type="xs:string" use="optional"/> <!-- log4cxx-->
    <xs:attribute name="type" type="xs:string" use="optional"/> <!-- log4net-->
</xs:complexType>

<xs:complexType name="LayoutType">
    <xs:sequence>
        <xs:element name="converter" minOccurs="0" maxOccurs="unbounded" type="ConverterType" />
        <xs:element name="param" minOccurs="0" maxOccurs="unbounded" type="ParamType" />
    </xs:sequence>
    <xs:attribute name="class" type="xs:string" use="optional"/> <!-- log4cxx-->
    <xs:attribute name="type" type="xs:string" use="optional"/> <!-- log4net-->
</xs:complexType>

<xs:complexType name="AppenderType">
    <xs:sequence>
        <xs:element name="filter" minOccurs="0" maxOccurs="unbounded" type="FilterType" />
        <xs:element name="evaluator" minOccurs="0" type="EvaluatorType" />
        <xs:element name="lossyevaluator" minOccurs="0" type="EvaluatorType" /> <!-- log4net-->
        <xs:element name="param" minOccurs="0" maxOccurs="unbounded" type="ParamType" />
        <xs:element name="layout" minOccurs="1" type="LayoutType" />
    </xs:sequence>
    <xs:attribute name="name" type="AppenderNameType" use="required"/>
    <xs:attribute name="class" type="xs:string" use="optional"/> <!-- log4cxx-->
    <xs:attribute name="type" type="xs:string" use="optional"/> <!-- log4net-->
</xs:complexType>

<xs:complexType name="AppenderRefType">
    <xs:attribute name="ref" type="AppenderNameType" use="required"/>
</xs:complexType>

<xs:complexType name="RootType">
    <xs:sequence>
        <xs:element name="appender-ref" type="AppenderRefType" minOccurs="0" maxOccurs="unbound
/>
    </xs:sequence>
</xs:complexType>

<xs:complexType name="LoggerType">
    <xs:sequence>
        <xs:element name="appender-ref" type="AppenderRefType" minOccurs="0" maxOccurs="unbound
/>
    </xs:sequence>
    <xs:attribute name="name" type="AppenderNameType" use="required"/>
</xs:complexType>

<xs:element name="configuration">

```

```

        <xs:complexType>
            <xs:sequence>
                <xs:element name="appender" type="AppenderType" minOccurs="0" maxOccurs="unbounded" />
                <xs:element name="root" type="RootType" minOccurs="0" />
                <xs:element name="logger" type="LoggerType" minOccurs="0" maxOccurs="unbounded" />
            </xs:sequence>
        </xs:complexType>
    </xs:element>
</xs:schema>

```

Example:

In this example, the local log configuration file is configured to write the system logs at debug level in `%ProgramData%\Qlik\Sense\Log\Proxy\Debug_System_Proxy.txt`.

```


<?xml version="1.0"?>
<configuration>
    <appender name="LocalApp_AppenderSystem"
type="Qlik.Sense.Logging.Log4net.Appender.QSRollingFileAppender">
        <param name="threshold" value="debug" />
        <param name="encoding" value="utf-8" />
        <param name="file" value="C:\ProgramData\Qlik\Sense\Log\Proxy\Debug_System_Proxy.txt" />
        <param name="maximumfiletime" value="720" />
        <param name="maximumfilesize" value="512KB" />
        <layout type="log4net.Layout.PatternLayout">
            <converter>
                <param name="name" value="rownum" />
                <param name="type"
value="Qlik.Sense.Logging.Log4net.Layout.Pattern.CounterPatternConverter" />
            </converter>
            <converter>
                <param name="name" value="longIso8601date" />
                <param name="type"
value="Qlik.Sense.Logging.Log4net.Layout.Pattern.Iso8601TimeOffsetPatternConverter" />
            </converter>
            <converter>
                <param name="name" value="hostname" />
                <param name="type"
value="Qlik.Sense.Logging.Log4net.Layout.Pattern.HostNamePatternConverter" />
            </converter>
            <converter>
                <param name="name" value="guid" />
                <param name="type"
value="Qlik.Sense.Logging.Log4net.Layout.Pattern.GuidPatternConverter" />
            </converter>
            <converter>
                <param name="name" value="serviceuser" />
                <param name="type"
value="Qlik.Sense.Logging.Log4net.Layout.Pattern.ServiceUserNameCachedPatternConverter" />
            </converter>
            <converter>
                <param name="name" value="encodedmessage" />
                <param name="type"
value="Qlik.Sense.Logging.Log4net.Layout.Pattern.EncodedMessagePatternConverter" />
            </converter>

```

```

    <converter>
      <param name="name" value="encodedexception" />
      <param name="type"
value="Qlik.Sense.Logging.log4net.Layout.Pattern.EncodedExceptionPatternConverter" />
    </converter>
    <param name="ignoresexception" value="false" />
    <param name="header" value="Sequence##x9;Timestamp##x9;Level##x9;Hostname##x9;
Logger##x9;Thread##x9;Id##x9;ServiceUser##x9;Message##x9;Exception##x9;
ActiveUserDirectory##x9;ActiveUserId##x9;ProxyTimestamp##x9;ProxyThread##x9;
Id2##xD;##xA;" />
    <param name="conversionpattern" value="%rownum{9999}##x9;%longIso8601date##x9;
%level##x9;%hostname##x9;%logger##x9;%thread##x9;%guid##x9;%serviceuser##x9;
%encodedmessage{1000000}##x9;%encodedexception{innermostmessage:1000000}##x9;
%property{ActiveUserDirectory}##x9;%property{ActiveUserId}##x9;
%property{ProxyTimestamp}##x9;%property{ProxyThread}##x9;%guid%newline" />
  </layout>
</appender>
<logger name="System.Proxy">
  <appender-ref ref="LocalApp_AppenderSystem" />
</logger>

```

 [Conversion \(page 283\)](#)

7.10 Telemetry logging

With the February 2018 release of Qlik Sense, you can capture granular usage metrics from the Qlik in-memory engine based on configurable thresholds. This provides, among other things, the ability to capture CPU and RAM usage of individual chart objects and CPU and RAM usage of reload tasks.

Enabling telemetry logging

Do the following:

1. Open the Qlik Management Console (QMC) by entering the QMC address in your browser.
By default, the QMC address is `https://<QPS server name>/qmc`.
2. In the QMC, navigate to **Engines**, select an engine and go the setting **QIX performance log level** under **Logging**.
3. Choose one of the following values:
 - **Off**: No logging will occur.
 - **Error**: Activity meeting the *error* threshold is logged. Recommended option.
 - **Warning**: Activity meeting the *error* and *warning* thresholds is logged. Recommended option.
 - **Info**: All activity is logged. Only recommended during troubleshooting because the log files may become very big.



The options **Fatal** and **Debug** are not applicable in this scenario.

4. Repeat steps 1 and 2 for each engine for which you want to enable telemetry.
5. Set threshold parameters. Edit `C:\ProgramData\Qlik\Sense\Engine\Settings.ini`. If the file does not exist, create it. You may need to open the file as an administrator to make changes.

Set the values according to the following list. We recommend that you start with these threshold values and increase or decrease them as you become more aware of how your particular environment performs. Too low values will create very large log files.

[Settings 7]

ErrorPeakMemory=2147483648

warningPeakMemory=1073741824

ErrorProcessTimeMs=60000

WarningProcessTimeMs=30000

6. Save and close the file.
7. Restart the Qlik Sense Engine Service.
8. Repeat steps 4-6 for each engine for which you want to enable telemetry.

Parameter descriptions

ErrorPeakMemory: Default 2147483648 bytes (2 Gb). If an engine operation requires more than this value of **Peak Memory**, a record is logged with log level *error*. **Peak Memory** is the maximum, transient amount of RAM an operation uses.

warningPeakMemory: Default 1073741824 bytes (1 Gb). If an engine operation requires more than this value of **Peak Memory**, a record is logged with log level *warning*. **Peak Memory** is the maximum, transient amount of RAM an operation uses.

ErrorProcessTimeMs: Default 60000 millisecond (60 seconds). If an engine operation requires more than this value of **Process Time**, a record is logged with log level *error*. **Process Time** is the end-to-end clock time of a request.

warningProcessTimeMs: Default 30000 millisecond (30 seconds). If an engine operation requires more than this value of **Process Time**, a record is logged with log level *warning*. **Process Time** is the end-to-end clock time of a request.



*You can only track either process time or peak memory. It is not required to track both metrics. However, if you set **ErrorPeakMemory**, you must set **warningPeakMemory**. If you set **ErrorProcessTimeMs**, you must set **warningProcessTimeMs**.*

Reading the logs

Telemetry data is logged to `C:\ProgramData\Qlik\Sense\Log\Engine\Trace\<hostname>_QixPerformance_Engine.txt` and rolls to the `ArchiveLog` folder in your `ServiceCluster` share.

In addition to the common fields found described, fields relevant to telemetry are the following:

Level: The logging level threshold the engine operation met.

ActiveUserId: The User ID of the user performing the operation.

Method: The engine operation itself, see *Important Engine Operations* (page 290).

DocId: The ID of the Qlik application.

objectId: For chart objects, the Object ID of chart object.

PeakRAM: The maximum RAM an engine operation used.

NetRAM: The net RAM an engine operation used. For hypercubes that support a chart object, the net RAM is often lower than Peak RAM as temporary RAM can be used to perform set analysis, intermediate aggregations, and other calculations.

ProcessTime: The end-to-end clock time for a request including internal engine operations to return the result.

workTime: Effectively the same as **ProcessTime**, excluding internal engine operations to return the result. Will report very slightly shorter time than **ProcessTime**.

TraverseTime: Time spent running the inference engine (that is, the green, white, and gray).

Important Engine Operations

The **Method** column details each engine operation and are too numerous to completely detail. The most relevant methods to investigate are as follows and will be the most common methods that show up in the logs if a *Warning* or *Error* log entry is written.

Method Description

Global::openApp - Opening an application

Doc::DoReload, **Doc::DoReloadEx** - Reloading an application

Doc::DoSave - Saving an application

GenericObject::GetLayout - Calculating a hypercube (that is, a chart object)

Comments

For best overall representation of the time it takes for an operation to complete, use **ProcessTime**.

About **Error** and **warning** log level designations: These designations were used because they conveniently fit into the existing logging and QMC frameworks. A row of telemetry information written out as an error or warning does not at all mean the engine had a warning or error condition that should require investigation or remedy unless you are interested in optimizing performance. It is simply a means of reporting on the thresholds set within the engine *Settings.ini* file and it provides a means to log relevant information without generating overly verbose log files.

In addition to the above information, once the logs mentioned above are created, the *Telemetry Dashboard* for Qlik Sense can be downloaded and installed to read the log files and analyze the information.

The *Telemetry Dashboard* provides the ability to capture CPU and RAM usage of individual chart objects, CPU and RAM usage of reload tasks, and more.

The dashboard can be downloaded at: [Telemetry Dashboard for Qlik Sense](#).

Do the following:

1. Run the installer. The files are installed at `C:\Program Files\Qlik\Sense`.
2. Once installed, you will see two new tasks, two data connections, and one new app in the QMC.

3. In the QMC, change the ownership of the application to yourself, or the user you want to open the app with.
4. Open the **Tasks** section in the QMC, select *TelemetryDashboard-1-Generate-Metadata*, and click **Start** at the bottom. This task will run and automatically reload the app upon completion.
5. Use the application from the hub to browse the information by sheets.

8 Troubleshooting - Deployment

There are several things you can do to troubleshoot and resolve problems before logging a case with product support. The general guidance here is designed to help you to understand the problem and know where to look for possible errors and potential solutions.

Before you call support:

- *Understand the problem (page 292)*
- *Use the log files (page 293)*
- Study the Qlik Sense Help.
- Read the troubleshooting topics in this section.

If you cannot find a solution in the product help, then follow the general guidance in this topic.

8.1 Understand the problem

Understanding the problem may help you to find a solution, and will enable you to provide Qlik support with the information needed to process your case more effectively. Ensure that you understand the problem and can describe it as fully as possible before seeking further support:

Questions and answers for support

Questions to ask	Answers - that may lead to a better understanding of the problem
Who experienced the problem?	What type of users were affected, and how many? This can help you to determine if it is a global issue, a configuration problem, a component problem, or due to user configuration.
What happened after an action was performed?	Pay attention to any symptoms, behavior, and error messages. This can help you to identify which component is causing the error, and which log files to use.
When did the problem first occur?	When is it triggered, and what user action or system action causes it? For example, is it due to a trigger reload, or if a user clicks on an object in an app?
Have you experienced this issue before?	If yes, how often has the problem occurred?

Where did this issue occur first?	Describe where in the system or environment the problem occurs. Is it on the client side, server backend, or in a specific application? For example, does the end user have a direct connection to the Qlik Sense hub, or are they passing through a third-party reverse proxy before reaching the hub?
Why do you think it happened?	Gather the relevant log files. Compare log files that include the problem with those that do not. For example, compare a successful reload with an unsuccessful reload of the same app. For log file locations, see the individual product help pages.

8.2 Use the log files

To troubleshoot and resolve issues effectively you need to understand how to use the log files. You also need to know when to use the default logs, and when to use the archived log files.

When you get an error message, the following steps can help you to identify which component has failed:

1. Read the error message carefully, as it can tell you which component has failed.
2. Navigate to the default log files, or the archived logs folder for the failed component.
3. When you have navigated to the correct folder, search for errors in the log file to identify the issue.

Default log files

In Qlik Sense, the log files are by default stored in `C:\ProgramData\Qlik\Sense\log`. After 12 hours they are moved to the archived logs folder.

There is one **Log** folder per machine, and the following sub-folders for each component (engine, repository, proxy, and scheduler):

- Audit - High level user action logs. For example, open app, reload app, get ticket, and login success.
- System - Service logs including all errors, and system or service operations.
- Trace - Debug diagnostics. For example, user selections, https redirects, method work times, and session information.

If you are running a multi-node environment, ensure you are connected to the correct node.

Criteria for moving the default logs to the archived logs folder:

- On service restart or crash
- If the file is larger than a predefined size
- If the file is more than 12 hours old

Archived log files

The archived logs folder is located in the Qlik Sense file share that you created as part of the Qlik Sense installation. Use the archived log files if the problem occurred more than 12 hours ago.

To find the archived logs, open the QMC and go to **Service Cluster, Cluster Settings**, and you can see the path that you specified during installation.

Unlike the standard logs, the archived log files are stored in a central, shared location, so if you are running a multi-node environment, you will find one sub-folder per node. Like the default log files, the archived log files also contain Audit, System, and Trace sub-folders for each main component.

For more information on the location of the log files for each component, see: *Storage (page 252)*

8.3 Qlik Sense client or application problems

If you get an error in a Qlik Sense application, the following questions can help you to narrow down the issue:

- Was the application working before the error occurred?
- Is the issue present in the Qlik Sense Desktop client?
- Is the issue specific to a browser, or is it present in all browsers?
- Does this issue affect a specific user, user group, or all users?
- Does this issue occur in one application, or every application?

8.4 Other resources

Once you have gathered all the information you need, use the following links to research other possible solutions:


- [Knowledge base](#)
- [Community website](#)
- [Log a case](#) with product support.

8.5 Cannot find the password for the PostgreSQL database superuser

Possible cause

In your Qlik Sense installation, you cannot find your repository database superuser password.

Proposed action

It is not possible to retrieve the password for the repository database superuser. To reset the password, see  [How-to reset forgotten PostgreSQL password in Qlik Sense](#) on Qlik Community.

If you used the same password for the database superuser as for the *qliksenserepository* user, see *Cannot find the password for the qliksenserepository database user (page 295)*.

8.6 Cannot find the password for the *qliksenserepository* database user

Possible cause

In your Qlik Sense installation, you cannot find the password for the *qliksenserepository* database user.

Proposed action

You can find the *qliksenserepository* database user password using the **Connection String Editor** which is included in the Qlik Sense diagnostic tools.

To open the **Connection String Editor**:

1. Navigate to *C:\Program Files\Qlik\Sense\Repository\Util\QlikSenseUtil* and double-click the *QlikSenseUtil.exe* file.
2. In the **LogOnForm** screen, enter the database user and password that you used during the Qlik Sense installation.
3. In the **Diagnostics Tool**, click the **Connection String Editor** tab.
4. In the **Connection String Editor** click **Read** to see the encrypted connection string.

8.7 Cannot access the hub or the QMC after installation

Possible cause

After you have installed Qlik Sense, the services are started automatically, which can take a few minutes. You cannot access the Hub or the Qlik Management Console until the services have started up correctly.

Proposed action

Check that the services have started and that the appropriate ports are available.

Do the following:

1. In Windows, open the Task Manager and check that all Qlik Sense services have started.
2. Check that the ports needed by Qlik Sense are available.
Plan and deploy Qlik Sense

8.8 Error message "No access path" after upgrade

You have set up an environment which requires proxy settings to connect to the Internet, as described in *Configuring a proxy for Qlik License Service communication in Qlik Sense Enterprise on Windows (page 151)*. After a version upgrade you get the error message "No access path" when you open the hub.

Possible cause

After the upgrade, the proxy settings in the *services.conf* file are reset to default values.



As of the November 2022 release of Qlik Sense, the existing services are by default retained when upgrading. See: Considerations about custom configurations (page 161)

Edit the configuration file by adding back the settings they had before the upgrade.

Path: `C:\Program Files\Qlik\Sense\ServiceDispatcher\services.conf`

This has to be done on every node in the cluster, followed by a restart of the Qlik Sense Service Dispatcher.

8.9 One or more Qlik Sense services did not start after installation

Possible cause

The Qlik Sense Repository Service (QRS) cannot start if there is no repository database, and if the QRS is not running, none of the other Qlik Sense services can start.



After installation the services are started automatically, which can take a few minutes. This means there may be a short delay before all services have started correctly.

Proposed action

Restart the service, check the user account, restart the server, or check the logs.

Do the following:

1. Stop the service and start it again.
You can also try changing the **Start Type** of the failing service from **Automatic** to **Automatic (Delayed Start)** in the Task Manager in Windows.
2. Check that the user that runs the Qlik Sense services is member of the Local Administrators group.
If you are using a domain administrator account, check that there is no problem related to the User Account Control (UAC).
3. Restart the server on which Qlik Sense is running.
4. Check the log files for the service to see if there is any information regarding why the service has not started.

The log files are available in the `%ProgramData%\Qlik\Sense\Log\<Service>` folder.

Set the `ServicesPipeTimeout` setting in the Registry Editor in Windows to 120000 milliseconds (that is, two minutes). This is needed to give the Qlik Sense Repository Service (QRS) enough time to start.



Serious problems might occur if you modify the registry incorrectly by using the Registry Editor or by using another method. Make sure that you can recover if the changes lead to problems.

- If the steps in this topic do not solve the problem, uninstall and reinstall Qlik Sense.

8.10 Anti-virus software scanning affects performance

Possible cause

Anti-virus software scanning can have an effect on the performance of Qlik Sense.

Proposed action

Configure the anti-virus software scanning so that it does not interfere with Qlik Sense. Make sure that regular scans and live/real-time scans are turned off for the following locations:

- `%ProgramData%\Qlik`
- All executables under `%ProgramFiles%\Qlik\Sense`
- All executables under `%ProgramFiles%\Common Files\Qlik\Custom Data`
- The file share location
- Any additional folder path configured for storing QVF files, except for the Service cluster Apps folder (`Share\Apps`) which should be included in scans as it could contain user uploads.



For additional support with anti-virus software exclusions, see [Antivirus exceptions for Qlik Sense McAfee, Symantec & Other Anti-Virus exclusions required](#).

8.11 Exit codes

Exit codes can be particularly useful when using the silent mode operations. The exit code can be viewed in the command prompt window by using the following command:

```
Echo %errorlevel%
```

The following table contains a complete list of the exit codes.

List of exit codes

Code	Description
0	Success
-1	General fatal error
-2	Command line parsing error
-3	Not implemented error

-4	Downgrade
-5	Malformed bundle XML
-6	Install condition not met
-7	Unknown upgrade scenario
-8	Pending reboot must be applied first
-9	Patch run with no baseline installed
-10	Disallowed setup process running
-11	Unsupported minor upgrade error
-12	Invalid policy
-13	User validation failed
-14	Database superuser password validation error
-15	Not supported error
-16	Host name from certificate retrieval error
-17	Inconsistent upgrade
-18	General silent workflow error
-19	OS bitness not supported
-20	OS too old
-21	OS type not supported
-22	Patch is superseded
-23	General MSI Error
-24	Disabled services exist
-1335	CAB is corrupt
-1601	Disk space
-1602	User exit
-1923	Cannot install service
-7777	Unknown dark process exception

8.12 Rim node loses connection to the central node

Possible cause

The Windows setting "**System cryptography: Force strong key protection for user keys stored on the computer**" is enabled. This setting is not supported by Qlik Sense.

Proposed action

Disable "**System cryptography: Force strong key protection for user keys stored on the computer**".

8.13 Repository cannot connect to database after installation

The installation was successful, but when the repository service is started it fails to connect to the database.

Possible cause

You used a database username and/or password that contains characters from mixed character sets.

Proposed action

1. Uninstall Qlik Sense and select **Remove Qlik Sense certificates and data folders** at the end of the installation.
2. Reinstall using a database username and password with characters from the same character set.

8.14 Unable to upgrade Qlik Sense, missing database

Possible cause

During upgrade, Secure Sockets Layer (SSL) encryption is enabled, and then the installer does not recognize any already installed PostgreSQL databases. Databases for SenseServices, QSMQ, and Licenses are reported missing.

Proposed action

During upgrade, temporarily disable SSL. See *Database security* (page 233).

8.15 Troubleshooting - database not configured for IP address or range

If you find the following error message in the installation logs: "psql: FATAL: no pg_hba.conf entry for host [ipv4 or ipv6]", it means the database needs to be configured.

Possible cause

The database is not configured to allow connection from that IP address or range.

Proposed action

Add an IP address or range in the Shared Persistence configuration file, see *Shared persistence configuration file syntax* (page 174), or in the installation UI, see *Installing Qlik Sense in a multi-node site* (page 108).

8.16 Troubleshooting app distribution in multi-cloud

There is more than one possible cause when app distribution fails in a multi-cloud environment. You could encounter problems on the QSEoW side (with custom properties), at the IdP (with names and groups), during the actual distribution, and after distribution (with apps not being displayed).

Publishing is a little slow

Possible cause

You have published an app, and when checking the collection, the app is not present.

Proposed action

Allow some time to pass before troubleshooting why an app does not appear in a collection. Publishing is not instantaneous.

A temporary error occurred

Possible cause

A temporary error occurred.

Proposed action

Restart the Qlik Sense Service Dispatcher.

Do the following:

1. In Windows, open **Services**.
2. Scroll down and right-click the Qlik Sense Service Dispatcher. Select **Restart**.

An unknown error occurred

Possible cause

An error occurred and you do not know why.

Proposed action

Investigate the log files for multi-cloud services, for example, the App Distribution Service and Hybrid Deployment Service.

8.17 Cannot read or write to the logging database

You have installed Qlik Sense successfully, but you cannot connect to the logging database.

Possible cause

You used a password that contains characters from mixed character sets. The log writer and log reader password cannot handle all mixed characters.

Proposed action

1. Uninstall Qlik Sense and select **Remove Qlik Sense certificates and data folders** at the end of the installation.
2. Reinstall using a password with characters from the same character set .

8.18 Failed to remove soft deleted records

When upgrading Qlik Sense to November 2017 or later, the installation fails and you get the following error message: "Failed to remove soft deleted records. An exception was thrown while invoking the constructor 'Void .ctor()' on type 'DatabaseContext'".

Possible cause

The database contains soft deleted records that generate an error when upgrading to a version of Qlik Sense without soft deletes, that is, November 2017 or later.

Proposed action

Run a script to delete the soft deleted records.



VERY IMPORTANT! Back up the whole QRS database before executing the script. If an error occurs, restore the backup, find out the data discrepancy, fix the issue and execute again, see *Backup and restore Qlik Sense Enterprise on Windows* (page 181).

Do the following:

1. Stop all the services, except the Qlik Sense Repository Database.
2. Save the script below to a file as `recurse_cleanup.sql`.
3. Move the file `recurse_cleanup.sql` to `%ProgramFiles%\Qlik\Sense\Repository\PostgreSQL<database version>\bin`.
4. Open a command prompt with elevated privileges.
5. Navigate to `%ProgramFiles%\Qlik\Sense\Repository\PostgreSQL<database version>\bin`, for example: `cd C:\Program Files\Qlik\Sense\Repository\PostgreSQL\9.6\bin`.



If you installed PostgreSQL manually, the location where to place and run the script from will be: `%ProgramFiles%\PostgreSQL<database version>\bin`.

6. Run `.\psql.exe -h localhost -d QSR -U postgres -p 4432 -a -f recurse_cleanup.sql`
7. If prompted, enter database superuser password.
8. Restart Qlik Sense Service Dispatcher, then start the Qlik Sense Repository Service in the given order.



When running the script on a non-English OS, you may encounter errors during the script execution. The errors can be caused by the character set conversion between server (PostgreSQL) and client (Powershell). To enable automatic character set conversion, run the following command from the command prompt before opening **Powershell** and executing the script: `SET PGCLIENTENCODING=UTF-8`. The variable is lost the moment the command prompt is closed. For more information refer to [Character Set Support](#).

Script for deleting soft deleted records in the Qlik Sense Repository Database

```
/*
#####
#####
    Script Name: Recurse cleanup
    Description: the script is intended to delete all entities marked as soft deleted in the
    QRS database
    Caution: BACKUP the whole QRS database before executing the script!

#####
#####
*/

/* Step 1. Update records according to QRS special logics

#####
#####
*/
-- Step 1.1 Update Owner to sa_repository if Owner is deleted

-- Step 1.1.1 Get all Qlik Sense Tables
CREATE OR REPLACE FUNCTION get_all_sense_tables() RETURNS SETOF information_schema.tables AS
$BODY$
BEGIN
    RETURN QUERY SELECT *
        FROM information_schema.tables
        WHERE table_schema='public'
            AND table_type='BASE TABLE'
            AND table_catalog='QSR'
            AND table_name <> '__MigrationHistory';
    RETURN;
END
$BODY$
LANGUAGE plpgsql;

-- Step 1.1.2 Filter Qlik Sense Tables with name of column
CREATE OR REPLACE FUNCTION get_tables(columnname varchar)
RETURNS SETOF information_schema.columns AS $$
BEGIN
    RETURN QUERY SELECT DISTINCT * FROM information_schema.columns as isc WHERE isc.column_
name = columnname And isc.table_name IN (SELECT ts.table_name FROM get_all_sense_tables() as
ts);
    RETURN;
END
$$
LANGUAGE plpgsql;

-- Step 1.1.3 Change ownership of soft deleted users to sa_repository
CREATE OR REPLACE FUNCTION fix_orphan_owners() RETURNS void AS
$BODY$
DECLARE username character varying;
DECLARE
    tables CURSOR FOR
```

```

SELECT * FROM get_tables('Owner_ID');

BEGIN
    SELECT E'\sa_repository\' INTO username;
    FOR table_record IN tables LOOP
        EXECUTE 'UPDATE ' || table_record.table_name || ' SET "Owner_ID" = (SELECT "ID" FROM
"Users" WHERE "UserId" = ' || username || ') WHERE "Owner_ID" IN (SELECT "ID" FROM "Users"
WHERE "Deleted" = true)';
    END LOOP;
END
$BODY$
LANGUAGE 'plpgsql';

SELECT * FROM fix_orphan_owners();

-- Step 1.1.4 Remove created DB functions for fixing ownership relations
DROP FUNCTION fix_orphan_owners();
DROP FUNCTION get_tables(columnname varchar);
DROP FUNCTION get_all_sense_tables();

-- Step 1.2 Unpublish App if Steam is deleted
UPDATE "Apps"
    SET "Stream_ID" = null, "Published" = false
WHERE "Stream_ID" IN (SELECT "ID" FROM "Streams" where "Deleted" = true);

UPDATE "AppObjects"
    SET "Approved" = false, "Published" = false
WHERE "App_ID" IN (SELECT "ID" FROM "Apps" where "Published" = false);

/* Step 2. Prepare for deletion: Alter foreign keys to Casacade Delete

#####
#####
*/

CREATE TABLE temp_foreign_key (
    constraint_name VARCHAR,
    table_name VARCHAR,
    column_name VARCHAR,
    ref_table_name VARCHAR,
    ref_column_name VARCHAR
);

INSERT INTO temp_foreign_key (constraint_name, table_name, column_name, ref_table_name, ref_
column_name)
SELECT fk.constraint_name, child.table_name, child.column_name, parent.table_name,
parent.column_name
FROM information_schema.referential_constraints fk
JOIN information_schema.key_column_usage AS child ON fk.constraint_name =
child.constraint_name
JOIN information_schema.key_column_usage AS parent ON fk.unique_constraint_name =
parent.constraint_name
WHERE fk.constraint_schema = 'public'
AND child.position_in_unique_constraint = parent.ordinal_position
AND fk.delete_rule = 'NO ACTION';

```

```

-- Step 2.2 Create a function the replace foreign keys with new on DELETE option
CREATE OR REPLACE FUNCTION replace_foreign_key (new_option VARCHAR) RETURNS void AS
$BODY$
DECLARE
    fks CURSOR FOR
        SELECT * FROM temp_foreign_key;
BEGIN
    FOR rec IN fks LOOP
        EXECUTE 'alter table "' || rec.table_name || '" '
            || 'drop constraint "' || rec.constraint_name || '" ,'
            || 'add constraint "' || rec.constraint_name || '" FOREIGN KEY ("' || rec.column_name ||
''') REFERENCES "' || rec.ref_table_name || '" ("' || rec.ref_column_name || '" ) ' || new_
option || ';' ;
    END LOOP;
END;
$BODY$
LANGUAGE plpgsql;

-- Step 2.3 execute the function to replace all foreign keys with CASCADE on Delete
SELECT *
    FROM replace_foreign_key('on delete cascade');

/* Step 3. Delete entities marked as Soft Deleted

#####
#####
*/
-- 3.1 Create a function to delete all softDeleted records
CREATE OR REPLACE FUNCTION delete_softdeleted_records(keep_for_days int) RETURNS void AS
$BODY$
DECLARE
    entity_tables CURSOR FOR
        SELECT table_name
            FROM information_schema.columns
            WHERE table_schema='public'
            AND column_name='Deleted';
BEGIN
    FOR tbl IN entity_tables LOOP
        EXECUTE 'delete from "' || tbl.table_name || '" where "Deleted" = true and
"ModifiedDate" <= CURRENT_DATE - ' || keep_for_days || ' ';
    END LOOP;
END;
$BODY$
LANGUAGE plpgsql;

-- Step 3.2 execute the function to delete entities
SELECT *
    FROM delete_softdeleted_records(3);

/* Step 4. Resume foreign keys to No Action on Delete

#####
#####
*/
SELECT *
    FROM replace_foreign_key('');

```



```
/* Step 5. Drop temp objects
```

```
#####  
#####  
*/  
DROP FUNCTION delete_softdeleted_records(keep_for_days int);  
DROP FUNCTION replace_foreign_key(new_option varchar);  
DROP TABLE temp_foreign_key;
```

8.19 Issues with Qlik Sense Enterprise when not connected to the internet

Internet access is not a requirement when working with Qlik Sense Enterprise, but the following issues can occur when a Qlik Sense Enterprise server cannot connect to the internet:

- Apps are not opened, or open very slowly.
- Data reloads run infinitely.

Possible cause

No internet connection.

Proposed action

- Connect to the internet whenever possible to avoid potential issues.
- Remove the Qlik ODBC Connector Package folders.

8.20 The Qlik Sense Client-Managed Mobile app encounters a network error and must close

Possible cause

If your Qlik Sense Client-Managed Mobile app was deployed using the VMware Tunnel for per-app VPN security, and the per-app VPN is later disabled in the iOS **Settings**, the following error will appear the next time the Qlik Sense Client-Managed Mobile app is launched:

The Qlik Sense Mobile app has encountered a network error and must stop. Restart the mobile app.

Proposed action

Ensure that the VMware Tunnel is enabled on your device.

Do the following:

1. On your iOS device, go to **Settings > VPN > VMware Tunnel > Connect On Demand** and toggle it on.

9 Deploying Qlik Sense Client-Managed Mobile

The Qlik Sense Client-Managed Mobile app allows you to securely connect to your Qlik Sense Enterprise on Windows deployment from a supported mobile device. The Qlik Sense Client-Managed Mobile app can be deployed and managed using Enterprise Mobile Management (EMM) software.

For more information about deploying and managing Qlik Sense Client-Managed Mobile, see *Installing Qlik Sense Client-Managed Mobile* (page 308).

9.1 The Qlik Sense Client-Managed Mobile app

The Qlik Sense Client-Managed Mobile app can be installed on supported devices running compatible versions of iOS or Android OS, and connected to a Qlik Sense Enterprise on Windows deployment.

For a detailed list of devices, OS versions, and Qlik Sense versions supported, see *System requirements for Qlik Sense Enterprise* (page 18)

The Qlik Sense Client-Managed Mobile app connects to a Qlik Sense hub. When connected, you can view and consume Qlik Sense apps and mashups available on the Qlik Sense Enterprise installation. Qlik Sense Client-Managed Mobile supports offline access to Qlik Sense apps. You can download the Qlik Sense apps for use offline when no internet connection is available. The Qlik Sense administrator controls which apps are available to download for offline use, using the QMC.



Developing Qlik Sense apps offline using the Qlik Sense Client-Managed Mobile app is not currently supported.

When you log into the Qlik Sense Client-Managed Mobile app for the first time, you must authenticate your credentials against the Qlik Sense Enterprise on Windows server. For more information, see *Connecting to Qlik Sense from the Qlik Sense Client-Managed Mobile app* (page 325). Once you have authenticated your credentials and logged in to the app, you may choose to have the Qlik Sense Client-Managed Mobile app remember your credentials. To protect your data, ensure that the device is protected by a password and locked when not in use. For more information, see *Qlik Sense Client-Managed Mobile security* (page 307).



To connect to a Qlik Sense Enterprise on Windows deployment from Qlik Sense Client-Managed Mobile, users must be allocated the appropriate access type. Users who have been allocated User access, or a Professional or Analyzer access license, can connect. Users with Analyzer capacity licenses or login access cannot. Anonymous access is not allowed.

9.2 Enterprise Mobile Management (EMM) and Qlik Sense Client-Managed Mobile

Using a supported EMM, you can remotely deploy and manage the Qlik Sense Client-Managed Mobile app on all of your organization's supported mobile devices. Using an EMM console, you can:

- Distribute the Qlik Sense Client-Managed Mobile app to mobile devices.
- Configure the Qlik Sense hub list in the Qlik Sense Client-Managed Mobile app.
- Configure the certificate validation policy.

For more information about configuring the certificate validation policy, see *Configuring the certificate validation policy for the Qlik Sense Client-Managed Mobile app (page 308)*.

For more information about deploying and managing Qlik Sense Client-Managed Mobile with AirWatch, see *Deploying the Qlik Sense Client-Managed Mobile app using AirWatch (page 309)*.

9.3 Qlik Sense Client-Managed Mobile security

Qlik Sense Client-Managed Mobile connects to a Qlik Sense Enterprise on Windows hub. When you are connected, you can view Qlik Sense apps and mashups, and download Qlik Sense apps using the Qlik Sense Client-Managed Mobile app.

Authentication

When you log into the Qlik Sense Client-Managed Mobile app for the first time, you must authenticate your credentials against the Qlik Sense Enterprise on Windows server. Once you have authenticated your credentials, and logged in to the Qlik Sense Client-Managed Mobile app, you may choose to have the Qlik Sense Client-Managed Mobile app remember your credentials. To protect your data, ensure that the device is protected by a password and locked when not in use. This can be configured through your Enterprise Mobile Management (EMM) console.



To connect to a Qlik Sense Enterprise on Windows deployment from Qlik Sense Client-Managed Mobile, users must be allocated the appropriate access type. Users who have been allocated User access, or a Professional or Analyzer access license, can connect. Users with Analyzer capacity licenses or login access cannot. Anonymous access is not allowed.

The Qlik Sense Client-Managed Mobile app can be used offline for up to 10 days (240 hours). This time period starts when the Qlik Sense Client-Managed Mobile app is first launched following the last log in to the Qlik Sense Enterprise on Windows server. When the 10 day period expires, you must to log back into the Qlik Sense Enterprise on Windows server to continue using the Qlik Sense Client-Managed Mobile app.

Section access in the data load script can also be used for security. A single file can be used to hold the data for a number of users or user groups. Qlik Sense then uses the information in the section access for authentication and authorization on the Qlik Sense Enterprise on Windows server, and dynamically reduces the data, so that users only see their own data. The security is built into the file itself, which means downloaded files are also protected.

Certificates

When Qlik Sense is deployed over SSL, the Qlik Sense Client-Managed Mobile app obtains a certificate from the Qlik Sense Enterprise on Windows server and verifies that it is valid. This allows the Qlik Sense Client-

Managed Mobile app to trust that the server it is talking to is a legitimate Qlik Sense Enterprise on Windows server. The Qlik Sense Client-Managed Mobile app will always reject the certificate if it is not valid. Every Qlik Sense hub that you add to the hub list must therefore have a valid certificate.

To ensure that a certificate is valid, you need to check that the certificate:

- Is signed by a certificate authority, such as VeriSign, or signed by a certificate authority that has been added to the list of trusted certificate authority for the device (either manually added to the device or pushed to the device from an EMM console).
- Is not expired.
- Has a common name or a name that matches the domain name of the Qlik Sense hub.

Configuring the certificate validation policy for the Qlik Sense Client-Managed Mobile app

The certificate validation policy applies when Qlik Sense is deployed over SSL, and the Qlik Sense Client-Managed Mobile app encounters invalid certificates from a Qlik Sense Enterprise on Windows server that has been added to the hub list by the device user.




You can configure the certificate settings from your EMM console.

Do the following:

1. Make sure the Qlik Sense Client-Managed Mobile app has been installed on the device.
 2. If the Qlik Sense server has a certificate that is not signed by a trusted certificate authority, make sure that the certificate that was used to sign the server certificate is added to the list of trusted certificate authorities for the device either manually or using your EMM.
 - a. Configure the certificate from your EMM console.

If your EMM console does not have this functionality, you can use this software to make the edits and then upload the changes to your EMM console:

 [Apple Configurator](#)
The available settings are:
- **Ask user**
 - **Always allow**
 - **Always reject**
- b. Push the changes to the device.

9.4 Installing Qlik Sense Client-Managed Mobile

The Qlik Sense Client-Managed Mobile app can be downloaded and installed directly from the Apple App Store or Google Play Store. The Qlik Sense Client-Managed Mobile app includes a Qlik Sense demo server that is hosted by Qlik, and allows you to view and download apps. You can connect to the Qlik Sense demo server without Qlik Sense Enterprise on Windows account credentials. To connect the Qlik Sense Client-Managed Mobile app to your Qlik Sense Enterprise on Windows deployment, your Qlik Sense administrator must configure the connection and deploy to users.

9 Deploying Qlik Sense Client-Managed Mobile

Qlik Sense Client-Managed Mobile can be deployed and managed using either Enterprise Mobile Management (EMM) software, or Apple Developer Enterprise Program tools.



To deploy using Apple Developer Enterprise Program tools, you must be a member of the Apple Developer Enterprise Program. For more information about deploying using Apple Developer Enterprise Program tools, see the Apple Developer Enterprise Program documentation.

Using a supported EMM, you can remotely deploy and manage the Qlik Sense Client-Managed Mobile app on all of your organization's mobile devices. From an EMM console you can:

- Distribute the Qlik Sense Client-Managed Mobile app to mobile devices.
- Configure the Qlik Sense hub list.
- Configure the certificate validation policy.

For more information about configuring the certificate validation policy, see *Configuring the certificate validation policy for the Qlik Sense Client-Managed Mobile app (page 308)*.

Qlik Sense Client-Managed Mobile and VPP

Qlik Sense Client-Managed Mobile can be deployed using the Apple Volume Purchase Program (VPP).

The Apple Volume Purchase Program (VPP) is a service that allows registered organizations to purchase iOS apps in bulk. After making a bulk purchase, the organization receives redemption codes for each app bought. Those app codes can then be distributed to individual users, who use the codes to download public apps from the Apple App Store. Codes can be distributed to users through email, a web portal, or Enterprise Mobile Management (EMM) software. Instead of pushing software and profiles out to devices, organizations can push licenses to devices while downloading apps directly from the Apple App Store.

Volume-purchased software and licenses can be distributed to users in the following ways:

- Email redemption URLs directly to users, which allows them to download and install the app.
- Post redemption URLs on an enterprise-hosted web page that is accessible only to authorized users.
- Use the Apple Configurator utility to push redemption codes to local connected devices.



Note that this method is only recommended for small work groups.

- Push redemption codes to remotely managed devices using EMM software to push redemption codes to remotely managed devices.

The Apple Volume Purchase Program allows businesses and schools to retain ownership of purchased apps, so apps can be reclaimed and redistributed as needed.

Deploying the Qlik Sense Client-Managed Mobile app using AirWatch

The Qlik Sense Client-Managed Mobile app can be deployed using AirWatch. To deploy using AirWatch, add the app to your AirWatch Catalog. Once the app is added to your AirWatch Catalog, you can choose to either push the app directly to your users' devices, or allow them to install the app manually.

To deploy the app using AirWatch:

1. Open your AirWatch Management Console.
2. Go to **Apps & Books > Applications > List View > Public** and select **Add Application**.
3. Select the **Platform**.
4. Select **Enter URL** and enter the URL to download the Qlik Sense Client-Managed Mobile app.
5. Click **Next**.
6. Configure options on the **Details** tab.
7. Assign the application to smart groups on the **Assignment** tab.
8. Configure the **App Delivery Method**:
 - On Demand - Deploys the app to a catalog and lets the user decide if and when to install it.
 - Automatic - Deploys the app to a catalog on a device upon enrollment. After the device enrolls, the user is prompted to install.
9. Select **Send Application Configuration** if you want to populate the Qlik Sense Client-Managed Mobile app with links to your Qlik Sense hub.
10. Assign a **Required Terms of Use** for the application on the **Terms of Use** tab.
11. Select **Save & Publish** to view the device assignment page that lists the assigned devices.
12. Select **Publish** to deploy the application.

For details about how users download and install the app manually using AirWatch, see *Connecting to Qlik Sense using AirWatch (page 312)*.

Qlik Sense Client-Managed Mobile and per-app VPN support

The Qlik Sense Client-Managed Mobile app supports per-app VPN tunneling when deployed using AirWatch.

Per-app VPN functionality, provides endpoint security by limiting connections at the application level, instead of at a device level. The VMware Tunnel restricts app access to allow-listed domains, and specific databases that allow-listed domains can access.

The following are the current minimum requirements for AirWatch support:

- AirWatch Agent version 5.5.1
- VMware Tunnel version 2.0.4

To enable per-app VPN tunneling support for Qlik Sense Client-Managed Mobile in AirWatch you will need to customize your VMware Tunnel configuration. For more information, see *Configuring AirWatch for per-app VPN (page 310)*.

Configuring AirWatch for per-app VPN

The Qlik Sense Client-Managed Mobile app supports per-app VPN tunneling when deployed using AirWatch. To enable per-app VPN tunneling, you must add network traffic rules so that app-internal TCP traffic within Qlik Sense Client-Managed Mobile bypasses the VMware Tunnel and remains on the device.

Add VMware Tunnel rules

Do the following:

1. Open your AirWatch Management Console.
2. From the **Settings** menu, go to **System > Enterprise Integration > VMware Tunnel > Network Traffic Rules**.
3. On the **Device Traffic Rules** tab, add the following rules:

Device traffic rules to add

Rank	Application	Action	Destination Hostname
1	Qlik Sense Client-Managed Mobile-iOS	Bypass	127.0.0.1
2	Qlik Sense Client-Managed Mobile-iOS	Tunnel	*

Once you have configured AirWatch for per-app VPN support of the Qlik Sense Client-Managed Mobile app, you can proceed with your deployment. For more information, see *Deploying the Qlik Sense Client-Managed Mobile app using AirWatch* (page 309).

Configuring the Qlik Sense Client-Managed Mobile app hub list using AirWatch

When you deploy the Qlik Sense Client-Managed Mobile app using AirWatch, you can choose to push the link to the Qlik Sense Enterprise hub directly to your users using AirWatch.

Do the following:

1. Open your AirWatch Management Console.
2. Go to **Apps & Books > Applications**.
3. Select the **Qlik Sense Mobile** app.
4. Click **Assign**.
5. Select the radio button for the group that you want to deploy the application configuration file to and click **Edit**.
6. On the **Add Assignment** page, expand the **ADVANCED** section, then expand the **APPLICATION CONFIGURATION** section.
7. In the **Configuration Key** field, enter *mdm*.
8. Ensure that the **Value Type** is set to **String**.
9. In the **Configuration Value** field, enter name and URL for each Qlik Sense Enterprise hub in the following format:

```
{ "Accounts" : [ { "name": "Account 1", "url": "http://www.ahub.com" },  
{ "name": "Account 2", "url": "http://www.asecondhub.com" } ] }
```
10. Click **Add**.
11. Click **Save & Publish**.
12. Click **Publish**.
13. Go to the **More** menu and select **Send Application Configuration**.

9 Deploying Qlik Sense Client-Managed Mobile

The Qlik Sense Enterprise hubs that you added will appear in the your users' Qlik Sense Client-Managed Mobile app list under **Select an account** the next time that they open the app.

Connecting to Qlik Sense using AirWatch

To connect to Qlik Sense from a mobile device using AirWatch per-app VPN, you must:

- Download the AirWatch Agent app
- Register the device
- Install a supported app or browser

To connect to Qlik Sense using AirWatch on iOS:

1. Download the AirWatch Agent app.
2. Open AirWatch Agent and enroll using one of the available options:
 - Email address
 - Server details
 - QR code
3. On the **Authenticate** screen, enter your **user name** and **password** and select **Next**.
4. On the **Secure** screen, select **Redirect & Enable** to enable management of your device by installing the Device Manager configuration profile. You are redirected and asked for permission to open Settings. Select **Allow**.
5. In Settings, select **Install** to install the Device Manager configuration profile, and then select **Trust** to confirm that you allow your device to be enrolled into remote management.
6. Once the installation of the Device Manager configuration profile is complete, select **Done**. You will be redirected to AirWatch Agent where a **Configure** screen confirms that the authentication procedure is complete. Select **Done**.



*If a pop-up appears asking to install VMware Tunnel, select **Install** to allow the installation of the VMware Tunnel app. If the pop-up does not appear, you can install VMware Tunnel from AirWatch Catalog. See step 9.*

7. In AirWatch Agent you can now manage your enrolled devices in the **My Device** portal.



You may be asked to create a device passcode to access AirWatch Agent. The passcode will be required every time you access the app. If you already have a passcode configured on your device you can enter it here to maintain the same passcode. If you enter a new passcode here it will overwrite your existing device passcode.

8. Close AirWatch Agent.
9. If you haven't installed VMware Tunnel already, open AirWatch Catalog, which has now been added to your device, and install it.
10. Open VMware Tunnel app and select **Continue** to enable it.

9 Deploying Qlik Sense Client-Managed Mobile

11. Open AirWatch Catalog and install the Qlik Sense Client-Managed Mobile app or one of the supported browsers.

For a list of mobile browsers that support the connection to Qlik Sense Enterprise on Windows through AirWatch per-app VPN, see [System requirements for Qlik Sense](#).



The Qlik Sense Client-Managed Mobile app allows you to download Qlik Sense apps for offline use.



Your AirWatch Agent administrator may have already populated the hub list with your Qlik Sense server connection.

12. To connect to Qlik Sense for the first time using the Qlik Sense Client-Managed Mobile app, see [Connecting to Qlik Sense from the Qlik Sense Client-Managed Mobile app \(page 325\)](#).

To connect to Qlik Sense using AirWatch on Android, follow the instruction for your specific device found here: [AirWatch Agent for Android](#).

Deploying the Qlik Sense Client-Managed Mobile app using MobileIron

You can deploy the Qlik Sense Client-Managed Mobile app using MobileIron Cloud or MobileIron Core by adding the app to your MobileIron App Catalog. Once you have added the app to your MobileIron App Catalog, you can choose to either push the app directly to your users' devices, or allow them to install the app manually. You can also configure the Qlik Sense Hub URLs that the user sees.

MobileIron Cloud

Follow the steps below to deploy the Qlik Sense Client-Managed Mobile app using MobileIron Cloud.

Do the following:

1. Open your MobileIron Management Console.
2. Go to **Apps > App Catalog** and click **Add**.
3. Select the **Platform** then select **Search App Store** and type *Qlik Sense Mobile*.
4. Select **Qlik Sense Mobile** from the apps that match this search term, then click **Next**.
5. Adjust the **Category**, then click **Next**.
6. Choose whether to **Delegate this app to all spaces**, then click **Next**.
7. Choose one of the **Distribution Options**, then click **Next**.
8. Do the following in **App Configurations**:
 - a. Configure the **Install on device**:
 - **Off**: the user decides if and when to install it from the App Catalog
 - **On**: after the device enrolls, the user is prompted to install immediately.
 - Enable **Convert to Managed App**, then click **Next**.
 - b. Configure the **iOS App Settings** that control whether to prevent backups, and remove apps on un-enrollment of the device, then click **Next**.

9 Deploying Qlik Sense Client-Managed Mobile

- c. Select **iOS Managed App Configuration** if you want to populate the Qlik Sense Client-Managed Mobile app with links to your Qlik Sense hub.
 - d. Configure the **Per App VPN** if you want to enable connectivity to any Qlik Sense hub via MobileIron Tunnel, then click **Next**.
9. Click **Done**.

MobileIron Core

Follow the steps below to deploy the Qlik Sense Client-Managed Mobile app using MobileIron Cloud.

Do the following:

1. Open your MobileIron Management Console.
2. Go to **Apps > App Catalog** and click **Add**.
3. Select the **Platform** then select **Application name**, then type *Qlik Sense Mobile* and click **Search**.
4. Select **Qlik Sense Mobile** from the apps that match this search term, then click **Next**.
5. Adjust the **Category**, then click **Next**.
6. Choose whether to hide or feature this app in the **Apps@Work** catalog, and conversion of app from unmanaged to managed, then click **Next**.
7. Do the following in the **App Configurations** section:
 - Configure the **Per App VPN** if you want to enable connectivity to any Qlik Sense hub via MobileIron Tunnel, then click **Next**.
 - Configure the **Managed App Settings** that control whether to prevent backups, and remove the app when the MDM profile is removed.
8. Click **Finish**.

Qlik Sense Client-Managed Mobile and per-app VPN support for MobileIron

The Qlik Sense Client-Managed Mobile app supports per-app VPN tunneling when deployed using MobileIron Core or MobileIron Cloud.

Together with MobileIron Sentry, the MobileIron Tunnel delivers per-app VPN functionality which provides endpoint security by limiting connections at the application level, instead of at a device level.

The following are the current minimum requirements for MobileIron support:

- MobileIron Tunnel version 4.0
- One of:
 - iOS version 13.4, 64bit
 - Android version 9, 64bit

Starting with MobileIron Tunnel 4.0, applications using localhost or the loopback IP 127.0.0.1 are now supported for Per App VPN if one of the following conditions are true:

- The ProviderType in the VPN config is set to use the Layer-3 **packet-tunnel**.
- The ProviderType in the VPN config is set to use the Layer-4 **app-proxy** and a new key-value pair **DirectLocalhost = True** is added to the Tunnel config to prevent the VPN client from routing app-internal TCP traffic to the VPN.

9 Deploying Qlik Sense Client-Managed Mobile

Idle connections from the mobile device to Qlik Sense may be prematurely terminated, interrupting the Qlik user experience, unless **TcpIdleTmoMs** = 300000 is added to the Custom Data key-value pairs. Note that this must be explicitly configured, and is different from the Disconnection Timeout that is also visible.

Differences between provider types

Provider Type	Sentry Service Type	Custom Data	iOS	Android
packet-tunnel (recommended)	IP_ANY	TcpIdleTmoMs=300000	Supported	Supported
app-proxy	TCP_ANY	DirectLocalhost=True TcpIdleTmoMs=300000	Supported	Not Applicable

Customizing the MobileIron Sentry configuration

The Sentry Profile must include a MobileIron Tunnel service configured with the **Service Type** above, corresponding with the **Provider Type** that will be used by MobileIron Tunnel.

Customizing the MobileIron Tunnel configuration

Follow the steps below to customize the MobileIron Tunnel configuration.

Do the following:

1. Create a MobileIron Tunnel Per App VPN configuration.
2. Select the **Provider Type**.
3. Select the **Sentry Profile**.
4. Select the **Sentry Service** that corresponds with the **Provider Type**:
 - *IP_ANY* for **packet-tunnel**
 - *TCP_ANY* for **app-proxy**
5. Select the **SCEP Identity** that is used by the MobileIron Tunnel client to authenticate to the MobileIron Sentry.
6. Identify your internal DNS Servers in the **DNS Resolver IP**, for example *172.16.0.100;172.16.0.101*
7. Record your Domain Names in **Match Domains**, for example *example.com;example.local*.
8. Add **Custom Data** key-pairs:
 - *TcpIdleTmoMs=300000*
 - *DirectLocalhost=true*
9. Add **Safari Domains** that will be routed through VPN, for example:
 - **.example.com*
 - **.example.local*
10. Click **Next**.
11. In Distribution rules, select the devices this configuration is distributed to.
12. Click **Done**.

Configuring the Qlik Sense Client-Managed Mobile app hub list using MobileIron

When you deploy the Qlik Sense Client-Managed Mobile app using MobileIron, you can choose to push one or several Qlik Sense Enterprise on Windows hub links directly to your users using Managed App Configuration. This can be performed via the **App Catalog**, or as a **PLIST** (MobileIron Core only).

MobileIron Cloud

Follow the steps below to configure the Qlik Sense Client-Managed Mobile app hub list using the **App Catalog** mechanism for MobileIron Cloud.

Do the following:

1. Open your MobileIron Cloud Console.
2. Go to **Apps > App Catalog**.
3. Select the **Qlik Sense Client-Managed Mobile** app.
4. Click **App Configurations**.
5. Select the **iOS Managed App Configuration** and do the following:
 - a. Navigate to the **Configuration** section.
 - b. Enter a JSON string into the **mdm** variable that identifies an array of named Hub URLs in the following format:

```
{
  "Accounts" : [
    { "name": "United Kingdom", "url": "https://sense.uk.example.com" },
    { "name": "Brazil", "url": "https://sense.br.example.com" }
  ]
}
```

Ensure that the JSON content is properly quoted and well-structured by validating it at <https://jsonlint.com/>.
 - c. Choose a **Distribution option**.
 - d. Click **Update**.

The Qlik Sense Enterprise on Windows hubs that you added now appear in the users' Qlik Sense Client-Managed Mobile app list under **Select an account** the next time that they open the app

MobileIron Core

In MobileIron Core you can configure the Qlik Sense Client-Managed Mobile app hub list via the **App Catalog**, or as a **PLIST**

App Catalog

Follow the steps below to configure the Qlik Sense Client-Managed Mobile app hub list using the **App Catalog** mechanism for MobileIron Core.

Do the following:

1. Open your MobileIron Cloud Console.
2. Go to **Apps > App Catalog**.
3. Select the **Qlik Sense Client-Managed Mobile** app and click **Edit**.

4. Do the following:
 - a. Navigate to the **Managed App Configurations** section and select **Default Configuration for Qlik Sense Mobile**.
 - b. Enter a JSON string into the **mdm** variable that identifies an array of named Hub URLs in the following format:

```
{
  "Accounts" : [
    { "name": "United Kingdom", "url": "https://sense.uk.example.com" },
    { "name": "Brazil", "url": "https://sense.br.example.com" }
  ]
}
```

Ensure that the JSON content is properly quoted and well-structured by validating it at <https://jsonlint.com/>.
Click **Save**.

The Qlik Sense Enterprise on Windows hubs that you added now appear in the users' Qlik Sense Client-Managed Mobile app list under **Select an account** the next time that they open the app

PLIST

Follow the steps below to configure the Qlik Sense Client-Managed Mobile app hub list using the **PLIST** mechanism for MobileIron Core.

Do the following:

1. Create a **PLIST** file in the following format that provides a JSON array of named Hub URLs in the **mdm** string variable:

```
<?xml version="1.0" encoding="utf-8" ?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLISR 1.0//EN"
"http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
  <key>mdm</key>
  <string>{ "Accounts" : [ { "name": "United Kingdom", "url":
"https://sense.uk.example.com" }, { "name": "Brazil", "url":
"https://sense.br.example.com" } ] }</string>
</dict>
</plist>
```

Ensure that the JSON content is properly quoted and well-structured by validating it at <https://jsonlint.com/> however the string value **MUST** be supplied on a single line without any embedded newline characters.
2. Open your MobileIron Core Console.
3. Go to **Policies & Configs**.
4. Select **Add New > Apple > iOS / tvOS > Managed App Config** and then do the following:
 - a. Enter a **name** for this Managed App Config that uniquely identifies it.
 - b. Enter *com.qlik.qliksense.mobile* in the **BundleId** field.
 - c. Select the PLIST file you created in the first step above.
 - d. Click **Save**.
5. Click **Actions > Apply to Label** to deliver the configuration to applicable registered devices.

9 Deploying Qlik Sense Client-Managed Mobile

The Qlik Sense Enterprise on Windows hubs that you added now appear in the users' Qlik Sense Client-Managed Mobile app list under **Select an account** the next time that they open the app

Connecting to Qlik Sense Client-Managed Mobile using MobileIron

To connect to Qlik Sense from a mobile device using MobileIron per-app VPN, you must:

- Download the MobileIron MDM Agent app.
- Register the device.
- Install the MobileIron Tunnel VPN Client app
- Install a supported app or browser.
- Review the list of mobile browsers that support the connection to Qlik Sense Enterprise through MobileIron per-app VPN: System requirements for Qlik Sense Enterprise.

Do the following:

1. Download the MobileIron MDM Agent app:
 - If using MobileIron Cloud, download MobileIron Go from the Apple App Store or Google Play store.
 - If using MobileIron Core, download MobileIron Mobile@Work from the Apple App Store or Google Play store.
2. Start the Device Registration procedure:
 - If using MobileIron Cloud, browse to <https://mobileiron.com/go>.
 - If using MobileIron Core, start MobileIron Mobile@Work.

Follow the prompts, supply the authentication and server details that your MobileIron administrator has supplied to you

3. Download the configuration profile when prompted.
If you are using iOS, you must then navigate to **Settings > General > Profiles** to proceed.
4. In Settings, select the Downloaded Profile and click **Install** to install the Device Manager configuration profile. Accept the prompts and warnings to **Install** and **Trust** to enroll your device into remote management.
5. Click **Done**.
6. Once the installation of the Device Manager configuration profile is complete, the Apps@Work catalog and several other applications are automatically installed.



*If a pop-up appears asking to install MobileIron Tunnel, select **Install** to allow the installation of the VPN client app. If the pop-up does not appear, you can install MobileIron Tunnel from the Apps@Work catalog.*

7. Open MobileIron Tunnel app and **Enable** the VPN.
8. Open Apps@Work and install the Qlik Sense Client-Managed Mobile app or one of the supported browsers.



The Qlik Sense Client-Managed Mobile app allows you to download Qlik Sense apps for offline use.



Your MobileIron administrator may have already populated the hub list with your Qlik Sense server connection

9. To connect to Qlik Sense for the first time using the Qlik Sense Client-Managed Mobile app, see *Connecting to Qlik Sense from the Qlik Sense Client-Managed Mobile app (page 325)*.

Connecting to Qlik Sense Client-Managed Mobile using MobileIron

To connect to Qlik Sense from a mobile device using MobileIron per-app VPN, you must:

- Download the MobileIron MDM Agent app.
- Register the device.
- Install the MobileIron Tunnel VPN Client app
- Install a supported app or browser.
- Review the list of mobile browsers that support the connection to Qlik Sense Enterprise through MobileIron per-app VPN: System requirements for Qlik Sense Enterprise.

Do the following:

1. Download the MobileIron MDM Agent app:
 - If using MobileIron Cloud, download MobileIron Go from the Apple App Store or Google Play store.
 - If using MobileIron Core, download MobileIron Mobile@Work from the Apple App Store or Google Play store.
2. Start the Device Registration procedure:
 - If using MobileIron Cloud, browse to <https://mobileiron.com/go>.
 - If using MobileIron Core, start MobileIron Mobile@Work.

Follow the prompts, supply the authentication and server details that your MobileIron administrator has supplied to you

3. Download the configuration profile when prompted.
If you are using iOS, you must then navigate to **Settings > General > Profiles** to proceed.
4. In Settings, select the Downloaded Profile and click **Install** to install the Device Manager configuration profile. Accept the prompts and warnings to **Install** and **Trust** to enroll your device into remote management.
5. Click **Done**.
6. Once the installation of the Device Manager configuration profile is complete, the Apps@Work catalog and several other applications are automatically installed.



*If a pop-up appears asking to install MobileIron Tunnel, select **Install** to allow the installation of the VPN client app. If the pop-up does not appear, you can install MobileIron Tunnel from the Apps@Work catalog.*

7. Open MobileIron Tunnel app and **Enable** the VPN.
8. Open Apps@Work and install the Qlik Sense Client-Managed Mobile app or one of the supported browsers.



The Qlik Sense Client-Managed Mobile app allows you to download Qlik Sense apps for offline use.



Your MobileIron administrator may have already populated the hub list with your Qlik Sense server connection

9. To connect to Qlik Sense for the first time using the Qlik Sense Client-Managed Mobile app, see *Connecting to Qlik Sense from the Qlik Sense Client-Managed Mobile app (page 325)*.

Deploying Qlik Sense Client-Managed Mobile with Microsoft Azure and Intune

The Qlik Sense Client-Managed Mobile app can be deployed using Microsoft Azure and Intune. Some configuration changes are required in the Microsoft Azure portal to enable Single Sign On (SSO) and Intune management of Qlik Sense Client-Managed Mobile.

Before you begin:

- Azure AD Connect must be configured to replicate your primary domain (Active Directory) and the Azure Portal (Azure Active Directory).
- Azure AD Application Proxy Connector must be installed and configured.

To deploy the app using Microsoft Azure and Intune:

- Set up a Qlik Sense Enterprise on Windows virtual proxy
- Set up Kerberos constrained delegation in Active Directory
- Add an Azure enterprise application for Qlik Sense Enterprise on Windows virtual proxy
- Add an Azure app registration for Qlik Sense Client-Managed Mobile
- Add the Qlik Sense Client-Managed Mobile app to the Intune **Company Portal**
- Define a Qlik Sense Client-Managed Mobile app protection policy
- Define a Qlik Sense Client-Managed Mobile configuration policy
- Deploy the Qlik Sense Client-Managed Mobile app

Set up a Qlik Sense Enterprise virtual proxy

1. Open the Qlik Management Console (QMC) on the Qlik Sense Enterprise on Windows server by entering the QMC address in your browser.
By default, the QMC address is `https://<QPS server name>/qmc`.
2. Go to **Proxies > Central Proxy**.
3. Enable **Kerberos Authentication**.
4. From the QMC home page, go to **Virtual Proxies**.
5. Click **Create new Virtual Proxy**.
6. Enter the following information:
 - Identification
 - Authentication
 - Load balancing
 - Host allowlist sections



Note the prefix used, it will be used later in the Azure Portal configuration (`https://sense_server_fqdn/prefix`).



The Windows Authentication pattern must be set to Mozilla.

7. Click **Save**.

Set up Kerberos constrained delegation in Active Directory

1. Log in to a server that has access to Active Directory in your primary domain.
2. Open a Windows Power Shell as an administrator.
3. Create a Service Principal Name (SPN) for the Qlik Sense Enterprise on Windows installation using the following command:
setspn.exe -U -S HTTP/sense_server_fqdn domain\sense_server_service_account
4. Open **Active Directory Users and Computer**.
5. Find the computer that hosts the Azure AD App Proxy, to modify the machine properties.
6. Go to the **Delegation** tab and choose **Trust the computer for delegation to specified services only**.
7. Select **Use any authentication protocol** and add the SPN created.
8. Open ADSI, confirm that the Azure AD app proxy host is set to delegate to the Qlik Sense server.

Add an Azure enterprise application for Qlik Sense Enterprise on Windows virtual proxy

1. Log in to the Azure portal and select **Azure Active Directory Service**.
2. Select **Application Proxy** and confirm there is at least one active application proxy.
3. Select **Enterprise Applications**.
4. Click **New application**.
5. Select **On-premises application**.

6. Enter a name for the new application.
7. Enter the URL for the server where Qlik Sense Enterprise on Windows is installed.



*Include the QSE virtual proxy prefix in the URL path.
For example: `https://sense_server_fqdn/prefix`*

8. Set up the **External URL**.



*This will be used later for the App Registration for Microsoft Intune. For example,
`https://sensekcd-qlikemnet.msappproxy.net/prefix`.
Note: The URL consists of a prefix (sensekcd-) followed by your tenant name followed by
`msappproxy.net` followed by the QSE virtual proxy prefix.*

9. Ensure that the application is using **Azure Active Directory** for its **Pre-Authentication** method.
10. Ensure that a valid **Connector Group** is selected to direct traffic to the application proxy.
11. Select **Single sign-on properties** for the **Enterprise Application**.
12. Choose **Integrated Windows Authentication** for **Single Sign-on Mode**.
13. Enter the SPN you created earlier.
14. Choose **On-premises user principal name** for **Delegated Login Identity**.
15. Click **Save**.
16. Select the enterprise application you added and click **Properties**.
17. Set **User assignment required** to **Yes**, and click **Save**.

Add an Azure app registration for Qlik Sense Client-Managed Mobile

1. Log in to the Azure portal and select **Azure Active Directory Service**.
2. Select **Apps Registrations**.
3. Click **New Application Registration**.
4. Enter a **Name**.
5. Enter a **Redirect URI** type Public client/native (mobile & desktop) with a URI of `qliksense-intune://com.qlik.qliksense.mobile`.
6. Click **Register** to continue.
7. Take note of this app registration's **Application ID**.
8. On the left hand panel, click **Authentication**.
9. Click **Add a platform** and add an Android platform.
Enter **Package Name** `com.qlik.qliksense.mobile`.
Enter Signature hash `17PV4mdIRAc/3SeFXILsSWg1aDU=`.
Click **Configure** and then click **Done**.
10. Add and grant the following delegated permissions:
 - Microsoft Mobile Application Management – Read and Write the User's App Management data.
This permission is found under the APIs my organization sers tab
 - The Web app / API defined above – Access <Web App / API name>

9 Deploying Qlik Sense Client-Managed Mobile

- Microsoft Graph – Read Directory Data
- Windows Azure Active Directory – Sign in and read user profile



Some of these permissions require Admin consent.

Add the Qlik Sense Client-Managed Mobile app to the Intune **Company Portal**

1. Log in to the Microsoft Endpoint Manager Admin Center.
2. Select **Apps**.
3. Select **All Apps**.
4. Click **Add**.
5. Select an **App type** of **Android Store App** for Android, or **iOS Store App** for iOS.
6. Click **Select** and then **Search the App Store**.
Search for and select **Qlik Sense Client-Managed Mobile**.
7. Click **Next** and review/change Assignments ensuring that the appropriate users and devices are assigned to the app.
8. Click **Next** and then click **Create**.




Perform these same steps for both Android and iOS versions of Qlik Sense Client-Managed Mobile.


Define a Qlik Sense Client-Managed Mobile app protection policy

1. Log in to the Microsoft Endpoint Manager Admin Center.
2. Select **Apps**.
3. Select **App protection policies**.
4. Click **Create Policy** and select **iOS/iPadOS** or **Android**.
5. Enter a **Name** and **Description**.
6. Click **Next**.
7. Enter a value of **Yes** for **target to all app types**.
8. Add a public app of Qlik Sense Client-Managed Mobile for Android or iOS as defined above and click **Next**.
9. Click on **Select Required Apps** and select the Qlik Sense Client-Managed Mobile for Android or iOS app added above.
10. If applicable, configure the data protection, access requirements and conditional launch values.
11. Click **Create**.
12. If the protection policy is configured to limit data transfer from Qlik Sense Client-Managed Mobile, then the limitation should be set to **policy managed apps** so that Qlik Sense Client-Managed Mobile can send diagnostics emails.



*For Android, use a browser to display help and use a PDF viewer to display the Qlik Sense Client-Managed Mobile **Terms and Conditions** document.*


 For iOS protection policy, a similar setting is required to allow Qlik Sense Client-Managed Mobile to send diagnostic emails. Help and terms and conditions are displayed within the iOS Qlik Sense Client-Managed Mobile app itself.

 Perform these same steps for both Android and iOS versions of Qlik Sense Client-Managed Mobile.

Define a Qlik Sense Client-Managed Mobile configuration policy

1. Log in to the Microsoft Endpoint Manager Admin Center.
2. Select **Apps**.
3. Select **App configuration policies**.
4. Click **Add**.
5. Select an enrollment type of **Managed Apps** for Android or **Managed Devices** for iOS.
6. Enter a **Name** and **Description**.
7. Add a public app of the Qlik Sense Client-Managed Mobile app previously added to the **Company Portal**. Click **Next**.
8. Under the **General configuration settings**, enter a name of **mdm**, and for value enter the JSON document:

```
{ "Accounts" : [ {  
  "name": "Your server name",  
  "url": "<external URL>",  
  "config": {  
    "AADAppId" : "<the Application Id noted above>"  
  } } ] }
```
9. Click **Next** and assign the appropriate users or user groups.
10. Click **Next** and then click **Create**.
11. Ensure that the app configuration shows as assigned with an enrollment type of **Managed apps** for Android, or **Managed devices** for iOS.

 Perform these same steps for both Android and iOS versions of Qlik Sense Client-Managed Mobile.

Deploy the Qlik Sense Client-Managed Mobile app to Android devices

1. On an Intune enrolled Android device open the **Company Portal** and install Qlik Sense Client-Managed Mobile.
2. Launch Qlik Sense Client-Managed Mobile.
3. You should be prompted to indicate that the app is being managed. If you aren't then there is likely a configuration issue with the App protection policy.
4. You should see your Qlik Sense Client-Managed Mobile deployment in the Qlik Sense Client-Managed Mobile server list. If you don't then there is likely a configuration or a user assignment issue.
5. Logging in to Qlik Sense Client-Managed Mobile deployment should follow the Azure SSO login flow.

Deploy the Qlik Sense Client-Managed Mobile app to iOS devices

1. On an Intune enrolled iOS device open the **Company Portal** and install Qlik Sense Client-Managed Mobile.
Intune will present a dialog asking to manage Qlik Sense Client-Managed Mobile.
2. Click **Yes** or **Manage**.
3. Launch Qlik Sense Client-Managed Mobile.
You should see the Qlik Sense Client-Managed Mobile server you defined above. If you don't then there is likely a configuration or a user assignment issue.
4. Click on the server and log in using SSO if required.
5. You will see an Intune dialog indicating that the App data is managed. Click **OK**. Qlik Sense Client-Managed Mobile will exit.
6. Logging in to Qlik Sense Client-Managed Mobile deployment should follow the Azure SSO login flow.

Connecting to Qlik Sense from the Qlik Sense Client-Managed Mobile app

When you install and launch the Qlik Sense Client-Managed Mobile app for the first time you are prompted to select either the Qlik Sense demo server or a Qlik Sense Enterprise on Windows server to connect to.

The Qlik Sense demo server is hosted by Qlik, and allows you to view Qlik Sense apps and mashups, and download apps. You can connect to the Qlik Sense demo server without Qlik Sense Enterprise on Windows account credentials.



You must connect to the Qlik Sense demo server at least once while online before you can access content while offline.

To connect to a Qlik Sense Enterprise on Windows server, you must log in with your Qlik Sense Enterprise on Windows account credentials. Before you can connect to a Qlik Sense server and log in with your Qlik Sense Enterprise on Windows account credentials from the Qlik Sense Client-Managed Mobile app you will need to authenticate your credentials against the Qlik Sense Enterprise on Windows server.

The Qlik Sense Enterprise on Windows authentication link must be generated by your administrator in the Qlik Management Console. Your Qlik Sense administrator will provide you with information about how you can receive the link using one of the following methods:

- Retrieving the authentication link from your Qlik Sense Enterprise on Windows hub
- Receiving the authentication link from your administrator



If your Qlik Sense Client-Managed Mobile app is deployed and managed through an EMM, the hub list may already be populated for you, in which case you do not need to complete this procedure.



To connect to a Qlik Sense Enterprise on Windows deployment from Qlik Sense Client-Managed Mobile, users must be allocated the appropriate access type. Users who have been allocated User access, or a Professional or Analyzer access license, can connect. Users with Analyzer capacity licenses or login access cannot. Anonymous access is not allowed.

Retrieving an authentication link from the Qlik Sense Enterprise on Windows hub

Do the following:

1. Open your mobile browser and enter the URL for your Qlik Sense Enterprise on Windows hub.
2. Click ... in the top toolbar of the hub, and then click **Client authentication**.
3. A dialog box opens asking you to confirm that you want to open the authentication link using the Qlik Sense. Click **Open** to confirm.
4. The Qlik Sense Client-Managed Mobile app opens and the server is added to the welcome page.
5. Click the server name to log in. You may be asked to enter your Qlik Sense Enterprise on Windows credentials.

After this, when you launch the Qlik Sense Client-Managed Mobile app, you can click the server name and log in using your Qlik Sense Enterprise on Windows credentials without authenticating against the Qlik Sense Enterprise on Windows hub each time.

Receiving the authentication link from your administrator

Do the following:

1. Click the authentication link provided by your Qlik Sense administrator. If you cannot click the link, copy the link into your mobile browser.
2. A dialog box opens asking you to confirm that you want to open the authentication link using the Qlik Sense. Click **Open** to confirm.
3. The Qlik Sense Client-Managed Mobile app opens and the server is added to the welcome page.
4. Click the server name to log in. You may be asked to enter your Qlik Sense Enterprise on Windows credentials.

After this, when you launch the Qlik Sense Client-Managed Mobile app, you can click the server name and log in using your Qlik Sense Enterprise on Windows credentials without authenticating against the Qlik Sense Enterprise on Windows hub each time.

9.5 Deploying mashups to the Qlik Sense Client-Managed Mobile app

Qlik Sense mashups are webpages that contain Qlik Sense app objects, such as charts and data. When a mashup is published in the Qlik Sense Enterprise on Windows hub, it can be also accessed from the Qlik Sense Client-Managed Mobile app.

Why use mashups in the Qlik Sense Client-Managed Mobile app

Using mashups in the Qlik Sense Client-Managed Mobile app enables faster loading and reduced data consumption for the mobile device. Mashups are generally less resource intense than Qlik Sense apps. This means that less data has to be retrieved from the Qlik Sense Enterprise on Windows server when loading a mashup in the Qlik Sense Client-Managed Mobile app.

A mashup retrieves the necessary data from the Qlik Sense server every time it is opened. This ensures that the mashup is always up to date with the Qlik Sense Enterprise installation.



Qlik Sense November 2018 or later is required to access mashups from the Qlik Sense Client-Managed Mobile app.



The use of Qlik Sense mashups is not supported in Qlik Sense Mobile for BlackBerry app.

Only mashups published in Qlik Sense can be accessed from the Qlik Sense Client-Managed Mobile. In the Qlik Sense Client-Managed Mobile app, mashups are listed in a dedicated **Mashups** stream. All public mashups in a Qlik Sense Enterprise on Windows installation are visible in the Qlik Sense Client-Managed Mobile app. An admin can restrict access to specific users by creating a security rule in the Qlik Management Console. See: *Restricting access to mashups in the Qlik Sense Client-Managed Mobile app (page 327)*.

Restricting access to mashups in the Qlik Sense Client-Managed Mobile app

To restrict access to mashups in the Qlik Sense Client-Managed Mobile app to specific users, the Qlik Sense Enterprise on Windows administrator must setup a security rule in the Qlik Management Console (QMC).

Do the following:

1. Open the QMC: `https://<QPS server name>/qmc`
2. In the QMC, create a custom property by doing the following:
 - Set a name for the new custom property, for example, "StreamAccess".
 - In the **Resource Types** section, select the **Extension** and **Users** check boxes to apply the custom property to these resource types.
 - In the **Value** section, create a new custom property value, for example, "MyMashup".
See: "Creating a custom property" in the Manage Qlik Sense sites guide.
3. To allow access to mashups to specific users, apply the custom property created in step 1 to the selected users. In the QMC, go in the **Users** section and edit users by adding "MyMashup" in the **StreamAccess** field.
4. To allow access to extensions to specific users, apply the custom property created in step 1 to the selected users. In the QMC, go in the **Extension** section and edit extensions by adding "MyMashup" in the **StreamAccess** field.

5. Create a new stream. Add to the stream the Qlik Sense apps that contain the data used in the mashups.
6. To prevent users from accessing a mashup, change the extension security rule as follows:
 - a. Create a copy of the default extension security rule.
 - b. Edit the copy you created by adding the condition `((resource.name!="MyMashup"))`, where "MyMashup" is the custom property you created in step 1.
 - c. Disable the default extension security rule to make the new one effective.See: "Security rules installed in Qlik Sense" in the Manage Qlik Sense sites guide.
7. Create the following security rule for extensions: `((user.@StreamAccess="MyMashup"))` to allow specific users to access all extensions.
See: "Creating security rules" in the Manage Qlik Sense sites guide.
8. Apply the same security rule `((user.@StreamAccess="MyMashup"))` to the stream you created in step 4 to allow specific users to access the stream.
See: "Editing streams" in the Manage Qlik Sense sites guide.

9.6 Customizing Qlik Sense Client-Managed Mobile with AppConfig

When administering Qlik Sense Client-Managed Mobile in an Enterprise Mobile Management (EMM) environment, you can customize the Qlik Sense Client-Managed Mobile experience for your users by editing the AppConfig file.

The AppConfig is a .json or .xml configuration file that can be edited using a Mobile Device Manager system. By editing the AppConfig file, you can for example change the default stream shown when Qlik Sense Client-Managed Mobile is launched, hide the demo server, or set a mashup as landing page. The way you modify the AppConfig file may vary depending on which Mobile Device Manager you use.

Configurable settings in AppConfig

The following are the configurable settings in the AppConfig file.

Settings

Type: Object

The settings object has the following properties:

hideDemoServer

Type: Boolean

If set to true, the demo server is hid from the account list.

hideAnalytics

Type: Boolean

If set to true, analytics are not displayed nor sent to Qlik.

If set to false, the end user can choose to send analytics to Qlik.

useBundledResources

Type: Boolean

If set to `true`, this setting enables to use the visualization client included in the Qlik Sense Client-Managed Mobile app when consuming apps online, making the online consumption of apps more efficient. The visualization client is used by default when consuming apps offline.

By default, this setting is absent and disabled. To be enabled, it needs to be manually added in the `AppConfig` and set to `true`.

For compatibility reasons, make sure to use the same version of Qlik Sense Enterprise and Qlik Sense Client-Managed Mobile when enabling this setting.

Accounts

Type: Object

The `Accounts` object is a JSON formatted list of accounts. Each item has a name that is shown to the user and a url used to authenticate the user. The value is formatted as follows:

```
{"name": "Account 1", "url": "http://www.hub-A.com"}, {"name": "Account 2", "url": "http://www.hub-B.com"}
```

The `Accounts` object has the following properties:

name

Type: string

The name of the account for which these settings are to be applied.

url

Type: string

The URL to the Qlik Sense hub.

config

Type: Object

The `config` object has the following properties:

- **DefaultStream**
Type: string
Changes the default stream that is selected when the Qlik Sense hub is loaded.
- **LandingPage**
Type: string
The path to a resource, such as a mashup, that should be loaded in place of the hub when a user successfully accesses Qlik Sense.
- **AADAppId**
Type: string

Used for Microsoft Azure Single Sign On. The value for this key is a string equal to the QSM Azure Active Directory App registration Application/Client ID.

AppConfig example

```
{
  "Settings":{
    "hideDemoServer": true,
    "hideAnalytics": true,
    "useBundledResources": true
  },
  "Accounts":[
    {
      "name":"Everyone account",
      "url":"https://acme.com/vprefix",
      "config": {
        "DefaultStream": "Everyone",
        "AADAppId": "95c232bc-5ab2-4954-8640-2a865eeb8597"
      }
    },
    {
      "name":"Mashup account",
      "url":"https://acme.com/vprefix",
      "config": {
        "DefaultStream": "mashups",
        "LandingPage": "/extensions/LandingPageMashup/LandingPageMashup.html"
      }
    }
  ]
}
```

Setting a mashup as landing page

By editing the AppConfig file, you can set a mashup or a mashup stream as the landing page for users accessing Qlik Sense.

In the **Configuration Value** field, enter the following:

```
{
  "name":"Mashup account",
  "url":"https://acme.com/vprefix",
  "config": {
    "DefaultStream": "mashups",
    "LandingPage": "/extensions/LandingPageMashup/LandingPageMashup.html"
  }
}
```

Where:

- "Mashup account" and "https://acme.com/vprefix" are the account and Sense hub to which these settings will be applied.
- "/extensions/LandingPageMashup/LandingPageMashup.html" is the path to the mashup to be used as landing page.
- "mashups" is the ID for a default stream that is loaded when accessing Qlik Sense.