



Plan Qlik Sense deployments

Qlik Sense®

June 2017

Copyright © 1993-2017 QlikTech International AB. All rights reserved.



Copyright © 1993-2017 QlikTech International AB. All rights reserved.

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

1 About this document	10
2 Planning your deployment	11
System requirements for Qlik Sense	11
Ports	11
Supported browsers	11
Architecture	11
Persistence	11
Services	11
User accounts	11
File share	11
Security	11
Licensing Qlik Sense	11
Qlik Sense installation	12
2.1 System requirements for Qlik Sense	12
2.2 Supported browsers	14
Qlik Management Console (QMC)	14
Microsoft Windows 7, 8.1	14
Microsoft Windows Server 2008 R2, 2012, 2012 R2, 2016	14
Microsoft Windows 10	14
Apple Mac OS X 10.11 and 10.12	14
Qlik Sense (the hub)	15
Microsoft Windows 7	15
Microsoft Windows 8.1	15
Microsoft Windows 10	15
Apple Mac OS X 10.11 and 10.12	15
Microsoft Windows Server 2008 R2	15
Microsoft Windows Server 2012	15
Microsoft Windows Server 2012 R2, 2016	16
iOS	16
Android	16
Windows Phone 8.1	16
Windows 10 phone	16
2.3 Architecture	16
Sites	16
Nodes	17
Storage	18
Clients	18
Services	19
Qlik Sense Repository Service (QRS)	19
Qlik Sense Repository Database (QRD)	21
Qlik Sense Proxy Service (QPS)	21
Qlik Sense Scheduler Service (QSS)	22
Qlik Sense Engine Service (QES)	25
Qlik Sense Printing Service (QPS)	25

Contents

Qlik Sense Service Dispatcher (QSD)	25
Deployment examples of nodes running Qlik Sense services	26
Service dependencies	27
Start and restart of services	27
Selecting the metrics to display	28
Ports	28
Ports overview	28
Ports used internally within a node	31
Ports used between user web browsers and proxies	32
Ports used between nodes and Qlik Sense services	32
Minimum ports used for communication in multi-node sites	33
Ports used between master and slave schedulers	34
Ports used between a proxy node and an engine node	34
Ports used between a proxy node and the printing service	35
Ports examples	35
Single node site	35
Multi-node site	36
Proxy node in demilitarized zone	36
Separate proxy and engine node	37
High availability proxy and engine nodes	38
Separate scheduler node and high availability proxy and engine nodes	38
Separate proxy and scheduler nodes and high availability engine nodes	39
Generic scale out	40
Persistence	41
File share	41
Repository database	41
.....	42
Basic deployment	42
Basic single-node deployment example	42
Enterprise deployment	43
Enterprise deployment examples	43
Single-node (small)	44
Multi-node (medium)	45
Multi-node (large)	46
2.4 Licensing	48
2.5 Performance	48
Geographical deployments	49
Capacity and performance	49
DMZ deployments	49
Central node dependencies	49
2.6 User accounts	49
Windows Qlik Sense services administrator	50
Windows Qlik Sense services user that is not an administrator	50
PostgreSQL database superuser	50

Qlik Sense Repository Database administrator	51
2.7 Security	51
Security and availability in a shared persistence deployment	51
Maintaining database password integrity	51
Database traffic encryption	52
Forcing the database connection to use TLS 1.2 only	52
Database replication and failover	53
Setting up replication to standby nodes for failover	53
Configure the primary database server	53
Configure the standby database server	53
Manual database failover	54
3 Qlik Sense installation	55
3.1 Installing Qlik Sense on a single node	55
3.2 Installing Qlik Sense in a multi-node site	59
Installing Qlik Sense	60
Adding a Qlik Sense node	66
3.3 Creating a file share	68
3.4 Failover	69
Automatic failover	69
Manually migrating the central node	69
3.5 Manually installing a repository database in PostgreSQL	70
3.6 Changing the user account to run Qlik Sense services	73
3.7 Performing a silent installation	75
Syntax	75
Commands	76
Arguments	76
Shared persistence configuration file syntax	78
Configuration file syntax	78
Deprecated command line arguments	79
3.8 Setting up Qlik Sense after installation	81
Connecting Qlik Sense to your user directory	81
Assigning licenses to users	81
Configuring the monitoring apps	81
How Qlik Sense uses HTTPS and certificates	82
Creating and opening apps	82
Disabling search indexing (only multi-node)	82
Working with streams, apps and publishing	82
4 Qlik Sense upgrades and updates	84
4.1 Upgrading	84
Upgrading from Qlik Sense 3.1 SR2 or later to Qlik Sense June 2017	86
Upgrading to Qlik Sense June 2017 after uninstalling Qlik Sense 3.1 SR2 or later	86
Upgrading to Qlik Sense June 2017 from Qlik Sense versions earlier than 3.1 SR2	87
4.2 Upgrading and migrating from synchronized to shared persistence	90

4.3 Performing a silent upgrade	93
Syntax	93
Commands	93
Arguments	93
Deprecated command line arguments	94
4.4 Repairing an installation	94
4.5 Performing a silent repair	95
Syntax	95
Commands	95
4.6 Patching Qlik Sense	96
4.7 Uninstalling Qlik Sense	97
5 Backing up and restoring	99
5.1 Backing up and restoring a site	99
Backing up a site	99
Backing up a site with shared persistence	100
Creating a PostgreSQL database dump file after uninstalling Qlik Sense	101
Restoring a site	102
Restoring a site with shared persistence	102
Restoring a central node to a machine with a different hostname	103
Known issues	104
5.2 Backing up and restoring certificates	104
Backing up certificates	104
Restoring certificates	113
Moving a certificate	123
5.3 Switching central node	132
Prerequisites	132
Procedure	132
6 Security	134
6.1 Protecting the platform	134
Network security	134
Server security	136
Process security	137
Rugged software	137
Threat analysis	137
App security	137
6.2 Authentication	138
Default authentication module	139
Certificate trust	139
Certificate architecture	139
Certificate trust requirements	140
General	140
Communication ports	140
Unlocking distributed certificates	141

Confirming certificates using Microsoft Management Console	141
Handling of certificates when a service starts	141
Client certificate	141
Server certificate	142
Root certificate	142
Definition of invalid certificate	143
Maximum number of trusted root certificates	143
Authentication solutions	144
Ticket solution	144
Session solution	145
Header solution	146
SAML	147
How SAML works	147
SAML in Qlik Sense	147
JWT	147
How JWT works	147
Anonymous users	148
6.3 Authorization	148
Access control	148
Resource access control	149
Resource access control rules	149
Resource access control streams	150
Administrator access control	151
Data reduction	151
6.4 Security summary	152
Authentication	152
Authorization	152
Auditing	152
Confidentiality	152
Integrity	153
Availability	153
Security example: Opening an app	153
7 Logging	155
7.1 New logging framework	155
7.2 Legacy logging framework	155
7.3 Reading and analyzing log files in Qlik Sense	155
7.4 Requirements	155
Securing the file system	155
Synchronizing time	156
Setting time zone	156
7.5 Storage	156
Log folder	156
Archived log files	159
7.6 Naming	159

Contents

7.7 Rows	160
7.8 Fields	160
Audit activity log	160
Audit security log	164
Server log	167
Qlik Sense engine service log fields	171
7.9 Trace logs	171
Storage	172
Naming	172
Rows	173
Fields	173
Common fields	173
Additional fields	176
Application log	176
Audit log	177
License log	178
Performance log	178
QIX performance log	180
Qlik Management Consolelog	181
Session log	181
System log	182
Task execution log	183
Traffic log	183
7.10 Configuring the logging	183
Appenders	184
QSRollingFileAppender	184
Configuring the appender	184
Converters	185
Built-in log4net appenders	186
Example: EventLogAppender	186
Example: SmtppAppender	187
Local log configuration file	187
Requirements	187
XML schema	187
8 Licensing	190
8.1 License Enabler File	190
Increase in tokens	190
Decrease in tokens	190
8.2 Access passes	190
Allocation of access passes	191
Login and logout	192
Login	192
Logout	192
Removing access passes	193

Contents

User access pass	193
Login access pass	193
Disconnected node	193
Multi-deployment sites	193
Development site	193
Test site	193
Anonymous users	194
8.3 Licensing metrics	194
8 Troubleshooting	195
8.4 Cannot access the hub or the QMC directly after installation	195
8.5 One or more Qlik Sense services did not start after installation	195
8.6 Anti-virus software scanning affects the performance of Qlik Sense	196
8.7 Exit codes	196
8.8 Rim node loses connection to the central node	198
8.9 Repository cannot connect to database after installation	198
8.10 Unable to upgrade, reinstall, or add a rim node due to password validation failure	198

1 About this document

This guide will introduce you to planning and installing Qlik Sense.

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

You find the online help, additional guides and much more at help.qlik.com/sense.

2 Planning your deployment

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

System requirements for Qlik Sense

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 architecture, and the different node types.

Persistence

Understand the persistence model used by Qlik Sense.

Services

Understand the Qlik Sense services.

User accounts

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

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

File share

Create a file share to store your Qlik Sense application data.

Security

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

Licensing Qlik Sense

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

Understand how Qlik Sense uses tokens for user access allocation.


Ensure that you have your Qlik Sense license key available.


Qlik Sense installation

Once you have reviewed and completed these items, you are ready to install Qlik Sense.

2.1 System requirements for Qlik Sense

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

Platforms	<ul style="list-style-type: none"> • Microsoft Windows Server 2012 • Microsoft Windows Server 2012 R2 • Microsoft Windows Server 2016 <p>For development and testing purposes only:</p> <ul style="list-style-type: none"> • Microsoft Windows 7 (64-bit version only) • Microsoft Windows 8.1 (64-bit version only) • Microsoft Windows 10 (64-bit version only) <div style="border: 1px solid gray; 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>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	<p>1.5 GB total required to install</p>
Storage	<ul style="list-style-type: none"> • Local disks • SAN storage mounted to Windows as a logical local drive
Security	<ul style="list-style-type: none"> • Microsoft Active Directory • Microsoft Windows Integrated Authentication • Third-party security
WebSockets	<p>Web browsers and infrastructure components (such as proxies and routers) must support WebSockets.</p>
.NET framework	<p>4.5.2</p>

<p>Repository database</p>	<p>PostgreSQL 9.6 (included)</p> <div style="border: 1px solid gray; padding: 5px; margin: 10px 0;">  <p><i>The version of PostgreSQL 9.6 installed with Qlik Sense does not include pgAdmin tools. You can download and install them manually if required.</i></p> </div> <p>PostgreSQL, often referred to as Postgres, is an open source object-relational database management system. It is released under the PostgreSQL License, which is a free/open source software license.</p>
<p>Internet protocol</p>	<ul style="list-style-type: none"> • IPv4 • IPv6 • Dual stack (IPv4 and IPv6)
<p>Network</p>	<p>The configured hostname must resolve to an IP address on the host machine.</p>
<p>Qlik Management Console (QMC), supported browsers</p>	<p>Microsoft Windows 7, Windows 8.1:</p> <ul style="list-style-type: none"> • Microsoft Internet Explorer 11 • Google Chrome • Mozilla Firefox (requires hardware acceleration, not supported in virtual environments) <p>Microsoft Windows Server 2008 R2, Windows Server 2012, Windows Server 2012 R2, 2016:</p> <ul style="list-style-type: none"> • Microsoft Internet Explorer 11 (not supported on Windows Server 2012) • Google Chrome • Mozilla Firefox (requires hardware acceleration, not supported in virtual environments) <p>Microsoft Windows 10:</p> <ul style="list-style-type: none"> • Microsoft Internet Explorer 11 • Microsoft Edge • Google Chrome • Mozilla Firefox (requires hardware acceleration, not supported in virtual environments) <p>Apple Mac OS X 10.11 and 10.12:</p> <ul style="list-style-type: none"> • Apple Safari • Google Chrome • Mozilla Firefox (requires hardware acceleration, not supported in virtual environments)

QMC, minimum screen resolution	Desktops, laptops, and Apple Mac: 1024x768 No mobile or small screen support.
QlikView compatibility	It is not possible to install Qlik Sense on a machine with QlikView Server already installed.



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.

2.2 Supported browsers

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

Qlik Sense Cloud is designed to work on web browsers listed in this section.

Qlik Management Console (QMC)

Microsoft Windows 7, 8.1

- Microsoft Internet Explorer 11
- Google Chrome
- Mozilla Firefox (requires hardware acceleration, not supported in virtual environments)

Microsoft Windows Server 2008 R2, 2012, 2012 R2, 2016

- Microsoft Internet Explorer 11 (not supported on Windows Server 2012)
- Google Chrome
- Mozilla Firefox (requires hardware acceleration, not supported in virtual environments)

Microsoft Windows 10

- Microsoft Internet Explorer 11
- Microsoft Edge
- Google Chrome
- Mozilla Firefox (requires hardware acceleration, not supported in virtual environments)

Apple Mac OS X 10.11 and 10.12

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



Minimum screen resolution for desktops, laptops, and Apple Mac is 1024x768. The QMC does not support tablets or iPads.

Qlik Sense (the hub)



You cannot open a second app in a new tab in Microsoft Internet Explorer 11. You have to open the app in a new browser window.

Microsoft Windows 7

- Microsoft Internet Explorer 11
- Google Chrome
- Mozilla Firefox (requires hardware acceleration, not supported in virtual environments)

Microsoft Windows 8.1

- Microsoft Internet Explorer 11
- Google Chrome
- Mozilla Firefox (requires hardware acceleration, not supported in virtual environments)

Microsoft Windows 10

- Microsoft Edge
- Microsoft Internet Explorer 11
- Google Chrome
- Mozilla Firefox (requires hardware acceleration, not supported in virtual environments)

Apple Mac OS X 10.11 and 10.12

- Apple Safari
- Google Chrome
- Mozilla Firefox (requires hardware acceleration, not supported in virtual environments).

Microsoft Windows Server 2008 R2

- Microsoft Internet Explorer 11
- Google Chrome
- Mozilla Firefox (requires hardware acceleration, not supported in virtual environments)

Microsoft Windows Server 2012

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

Microsoft Windows Server 2012 R2, 2016

- Microsoft Internet Explorer 11
- Google Chrome
- Mozilla Firefox (requires hardware acceleration, not supported in virtual environments)

iOS

iPad 3rd Generation and above or iPhone 5 and above with latest iOS (script editing is not supported on tablet devices):

- Apple Safari

Android

Version 4.4.4, 5.1.1, 6.0 and 7.0 (script editing is not supported on tablet devices):

- Google Chrome

Windows Phone 8.1

- Microsoft Internet Explorer 11

Windows 10 phone

- Microsoft Edge



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

2.3 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. For more information on how to configure fail over, see [Creating a node](#) and [Service cluster](#).

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

- 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

For more information on node purpose, see: [Creating a node](#).

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 applications, the app structure, 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, and 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.

See: *Creating a file share (page 68)*



For an app to run in Qlik Sense, it must be stored in the repository database.

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.

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).

Executable	<i>%ProgramFiles%\Qlik\Sense\Repository\Repository.exe</i>
Data	<i>%ProgramData%\Qlik\Sense\Repository</i>
Logs	<i>%ProgramData%\Qlik\Sense\Log\Repository</i> See: <i>Logging (page 155)</i>
Repository database	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. All files related to the repository database in a default Qlik Sense installation are stored in the following folder: <i>%ProgramData%\Qlik\Sense\Repository\PostgreSQL</i>

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 73)*

- `-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.
3. Start Qlik Sense services.

Metrics

This section lists the metrics related to the Qlik Sense Repository Service (QRS).

See: *Selecting the metrics to display (page 28)*

REST API metrics

The following metrics are available in the Performance Monitor in Microsoft Windows:

- Number of DELETE calls
- Number of GET calls
- Number of POST calls
- Number of PUT calls
- Number of HTTP status 200 (OK)
- Number of HTTP status 201 (Created)
- Number of HTTP status 400 (Bad request)
- Number of HTTP status 401 (Unauthorized)
- Number of HTTP status 403 (Forbidden)
- Number of HTTP status 406 (Not acceptable)
- Number of HTTP status 409 (Conflict)
- Number of HTTP status 415 (Unsupported media type)

- Number of HTTP status 500 (Internal server error)
- Number of HTTP status 503 (Service unavailable)

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.

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).

Executable	<i>%ProgramFiles%\Qlik\Sense\Proxy\Proxy.exe</i>
Data	<i>%ProgramData%\Qlik\Sense\Proxy</i>
Logs	<i>%ProgramData%\Qlik\Sense\Log\Proxy</i> See: <i>Logging (page 155)</i>

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 73)*

- `-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.
- `-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

This section lists the metrics related to the Qlik Sense Proxy Service (QPS). The following metrics are available in the Performance Monitor in Microsoft Windows:

See: *Performance log (page 178)*

See: *Selecting the metrics to display (page 28)*

- **ActiveConnections:** The number of active connections from the client.
A connection is a stream (or 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 (or 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 metrics are also available as entries in the Performance log for the QPS.

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 master, slave, or both on a node.

Master

There is only one master Qlik Sense Scheduler Service within a site and it is always located on the central node, where the master 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 master 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 master QSS sends the task ID to a slave QSS within the site. The load balancing operation performed by the master QSS determines which slave QSS to distribute the task ID to.

When a slave QSS completes a task, it returns the task state (successful or fail) to the master QSS. The master 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 slave QSS fails to perform the task, the master QSS repeatedly requests the same or another slave QSS to perform the task until it has been completed or until the maximum number of attempts has been reached.

Slave

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

Executable	<i>%ProgramFiles%\Qlik\Sense\Scheduler\Scheduler.exe</i>
Data	-
Logs	<i>%ProgramData%\Qlik\Sense\Log\Scheduler</i>
	<i>See: Logging (page 155)</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.

See: *Changing the user account to run Qlik Sense services (page 73)*

- `-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

This section lists the metrics related to the Qlik Sense Scheduler Service (QSS). The following metrics are available in the Performance Monitor in Microsoft Windows:

See: *Selecting the metrics to display (page 28)*

- Number of connected slaves
- Number of Qlik Sense Engine Service (QES) instances that are running on a slave (this metric is only available on the node where the QES instances run)
- Number of running processes
- Number of running tasks as understood by the master
- Number of running tasks on the slave
- Number of task messages that have been dispatched by the slave
- Number of task messages that have been received by the master
- Number of task retries
- Number of tasks that have completed successfully when executed by the slave
- Number of tasks that have failed when executed by the slave
- Number of tasks that the master has acknowledged as completed
- Number of tasks that the master has acknowledged as failed
- Number of times that the settings have been updated
- Number of tasks that have attempted to start
- Number of tasks that have attempted to stop

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).

Executable	<i>%ProgramFiles%\Qlik\Sense\Engine\Engine.exe</i>
Data	<i>%ProgramData%\Qlik\Sense\Engine</i>
Logs	<i>%ProgramData%\Qlik\Sense\Log\Engine</i> <i>See: Logging (page 155)</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 (QPS)

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).

Executable	<i>%ProgramFiles%\Qlik\Sense\Printing\Printing.exe</i>
Data	<i>%ProgramData%\Qlik\Sense\Printing</i>
Logs	<i>%ProgramData%\Qlik\Sense\Log\Printing</i> <i>See: Logging (page 155)</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.
- 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.
- Migration Service: ensures that your apps can be used in the currently installed version of Qlik Sense. This service only runs on the central node in a site. 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.

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.

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

See: *Logging (page 155)*

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.

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:

- a. Qlik Sense Repository Database (QRD)
- b. Qlik Sense Service Dispatcher (QSD)
- c. Qlik Sense Repository Service (QRS)
- d. 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 order is important because the QRS is dependent on the QRD and the rest of the services are dependent on the QRS.

Selecting the metrics to display

To select which metrics to display for the Qlik Sense services in the Microsoft Windows, Performance Monitor:

1. Select **Start>Run**.
2. Enter *perfmon* and click **OK**.
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 - REST API
 - Qlik Sense Repository Service
 - 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**.

Ports

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

Ports overview

The following table is an overview of the ports used in a Qlik Sense deployment.

Component	Inbound	Outbound	Internal only
-----------	---------	----------	---------------

2 Planning your deployment



Qlik Sense Proxy Service (QPS)	80 (HTTP)	4239 (QRS websocket)	
	443 (HTTPS)	4242 (QRS REST API)	
	4243 (REST API)	4747 (Engine)	
	4244 (HTTPS Windows authentication)	4899 (Printing)	
		4900 (Broker)	
	4248 (HTTP Windows authentication)	4949 (Data profiling/prep)	

No additional ports.



Qlik Sense Engine Service (QES)	4747 (QES listen port)		
	4748 (notifications from QRS)		

4242 (QRS REST API)



Qlik Sense Repository Service (QRS)	4242 (REST API)	4242 (REST API)	4545 (Migration Service)
	4239 (from QPS - websocket)	4243 (Proxy REST API)	4570 (Certificate unlock)
	4444 (Setup API - inbound on rim nodes)	4444 (Setup API – outbound on central node)	
	4899 (from QPR)	4747 (Engine)	
		4748 (Engine notification API)	
		5050 (Scheduler master API)	



Qlik Sense Scheduler Service (QSS)	5050 (from slave QSS)	4242 (QRS REST API)	No additional ports.
	5151 (from master QSS)	5151 (QSS Slave REST API)	
	5151 (Slave REST API)		
	5252 (Monitoring API - optional)		



Qlik Sense Repository Database (QRD)	4432 (default listen port for database connections)		No additional ports.
--------------------------------------	---	--	----------------------



Qlik Sense Printing service (QPR)	4899 (QPR listen port)		
-----------------------------------	------------------------	--	--

443 (Sense web server - proxy)

4242 (QRS REST API)

4244 (Sense authentication -



Qlik Sense Service Dispatcher (QSD)

Starts up the following services:

Broker service 4900

Data profiling service 4949 (listen port for REST API and websocket)

proxy)

8088 (CEF debugging)

3003 (Converter Service)

4545 (App migration)

4555 (Chart sharing)

4949 (Data profiling)

9028 (Hub Service)

9031 (Capability Service)

9079 (Depgraph Service)

9090 (DownloadPrep)

9098 (On-demand App Service)

9080 (Web extension Service)

4242 (QRS REST API)

4747 (QES)



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.

Service	Port	Direction	Purpose
Converter Service	3003	Internal	This port is used by the Converter Service which is utilized by QlikView converter.
QPS	4243	Inbound	Qlik Sense Proxy Service (QPS) REST API listen port. 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.
QRD	4432	Internal	Default listen port for the Qlik Sense Repository Database (QRD). With shared persistence, this port is used to listen for connections from the Qlik Sense Repository Service (QRS).
Migration Service	4545	Internal	This port is used by the Migration Service for app migration purposes. The service is launched and managed by the Qlik Sense Service Dispatcher (QSD) when required. The Migration Service only runs on the central node.
Chart Sharing Service	4555	Internal	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. This port uses HTTPS for communication.
QRS	4570	Internal	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.
QES	4748	Internal	This callback port is used by the Qlik Sense Repository Service (QRS) for sending HTTP events to the Qlik Sense Engine Service (QES).
Data Profiling Service	4949	Internal	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.
Broker Service	4900	Internal	Default listen port for the Broker Service.
Hub Service	9028	Internal	Default listen port for the Hub Service.

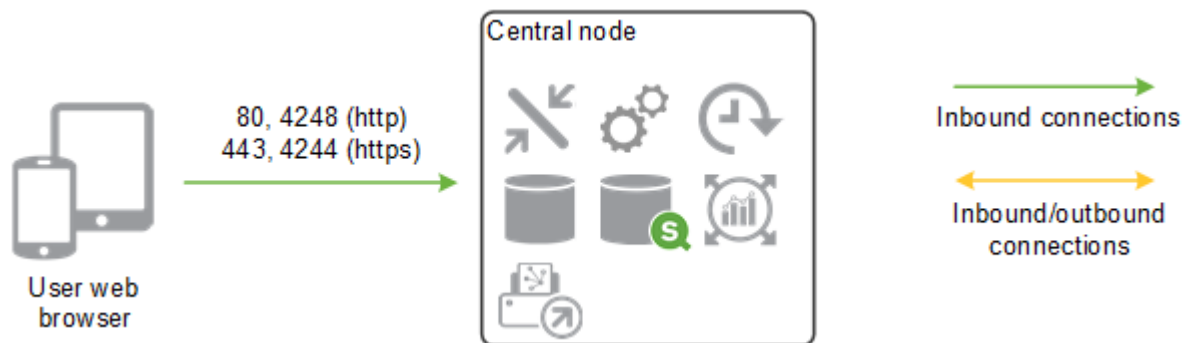
2 Planning your deployment

Capability Service	9031	Internal	This port is used by the Capability Service to handle Qlik Sense system feature configuration.
Depgraph Service	9079	Internal	This port is used by the Service Dispatcher microservices.
Web Extension Service	9080	Internal	Default listen port for the Web Extension Service.
DownloadPrep	9090	Internal	This port is used by the Service Dispatcher microservices.
On-demand App Service	9098	Internal	Default listen port for the On-demand App Service.

Ports used between user web browsers and proxies

The default ports are exposed to the Qlik Sense users and need to be open through any firewalls to each Qlik Sense Proxy Service (QPS) in the site.

Service	Port	Direction	Purpose
QPS	443	Inbound	Inbound user web traffic when using HTTPS.
QPS	4244	Inbound	Authentication port when using Windows authentication over HTTPS.
QPS	80	Inbound	Inbound user web traffic when using HTTP (optional).
QPS	4248	Inbound	Authentication port when using Windows authentication over HTTP (optional).



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.

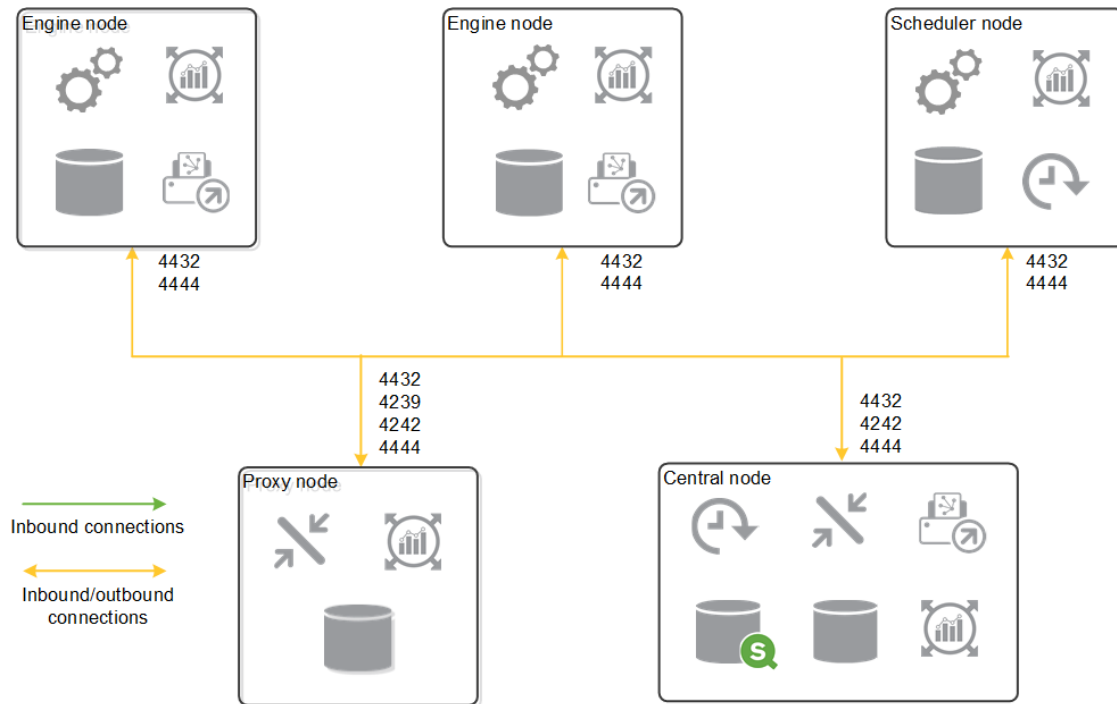
In a multi-node site, the ports in use vary depending on the services installed and running on each node. The ports need to be open in any firewalls between the nodes, but do not have to be open to the Qlik Sense users.

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.

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.
QRD	4432	Inbound from rim 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 master 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.

2 Planning your deployment



Ports used between master and slave schedulers

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

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

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.

2 Planning your deployment

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 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.

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.

Ports examples

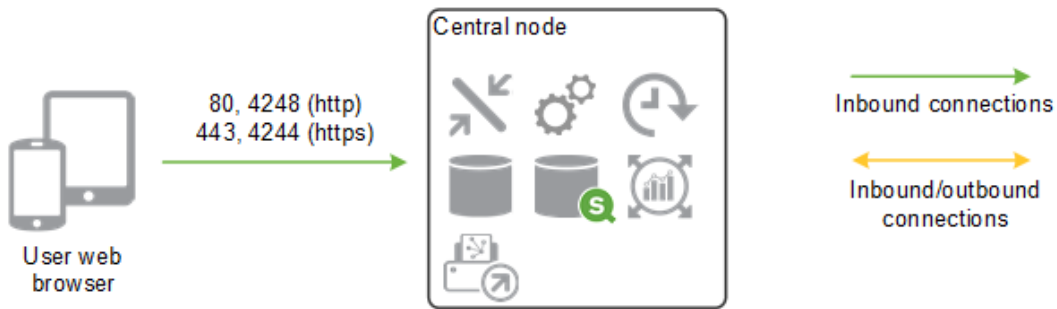
This section provides examples of the ports that are used in different Qlik Sense deployments.



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 28) table.

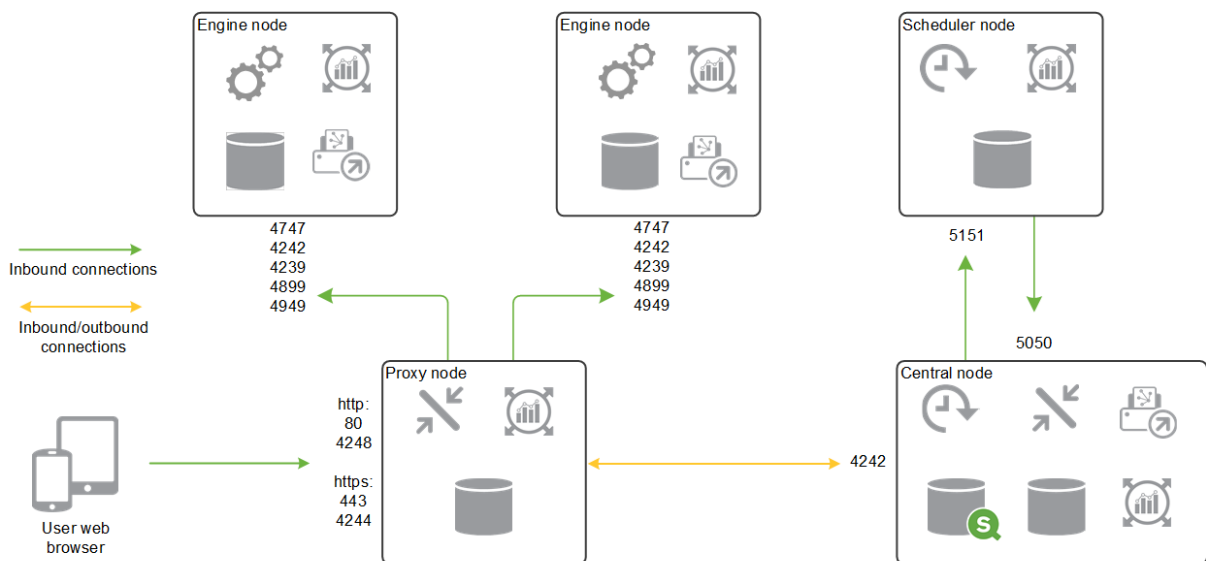
Single node site

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



Multi-node site

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

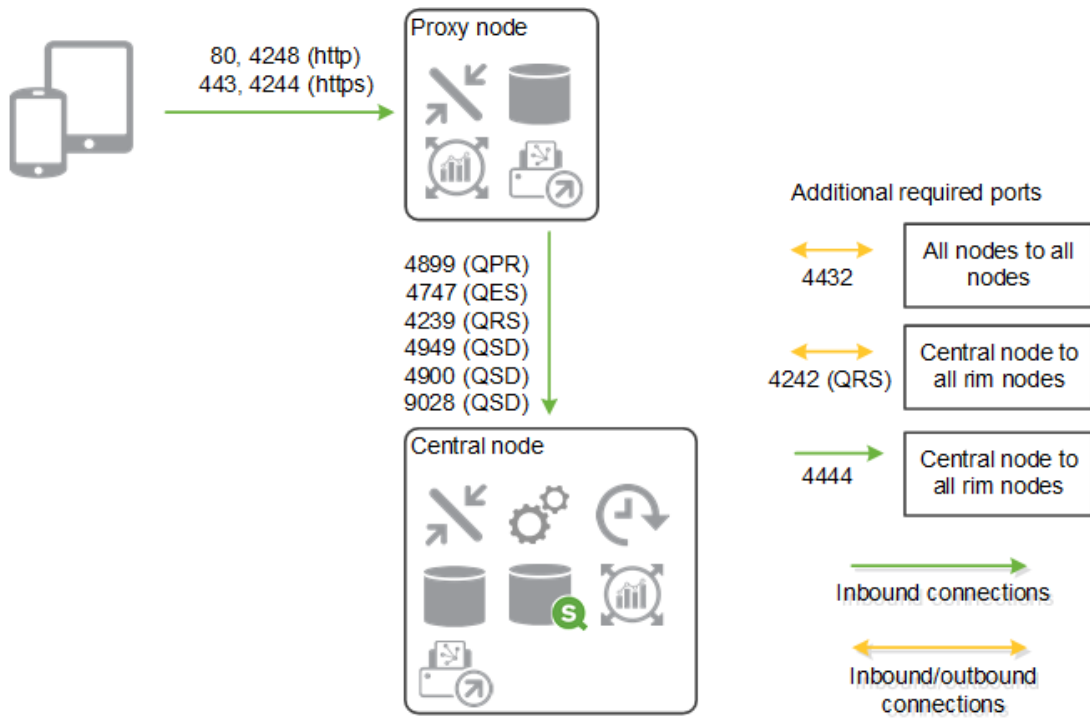


This deployment also requires port 4444 between all nodes

Proxy node in demilitarized zone

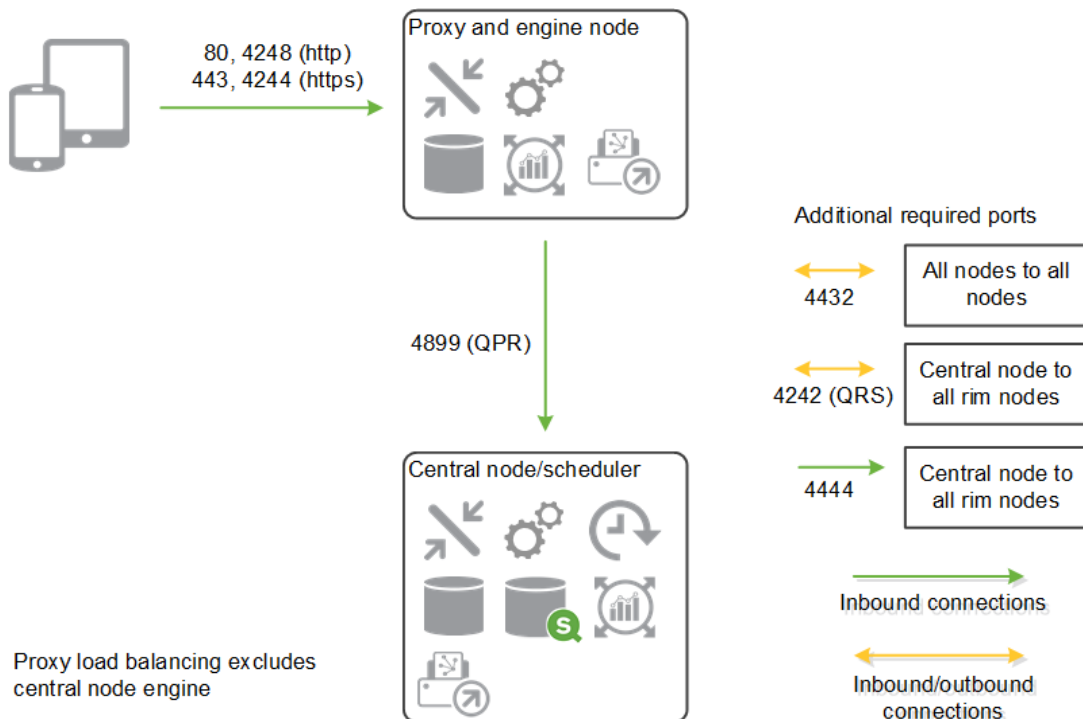
This example shows the ports that are used in a multi-node site when deploying a proxy node in a demilitarized zone.

2 Planning your deployment



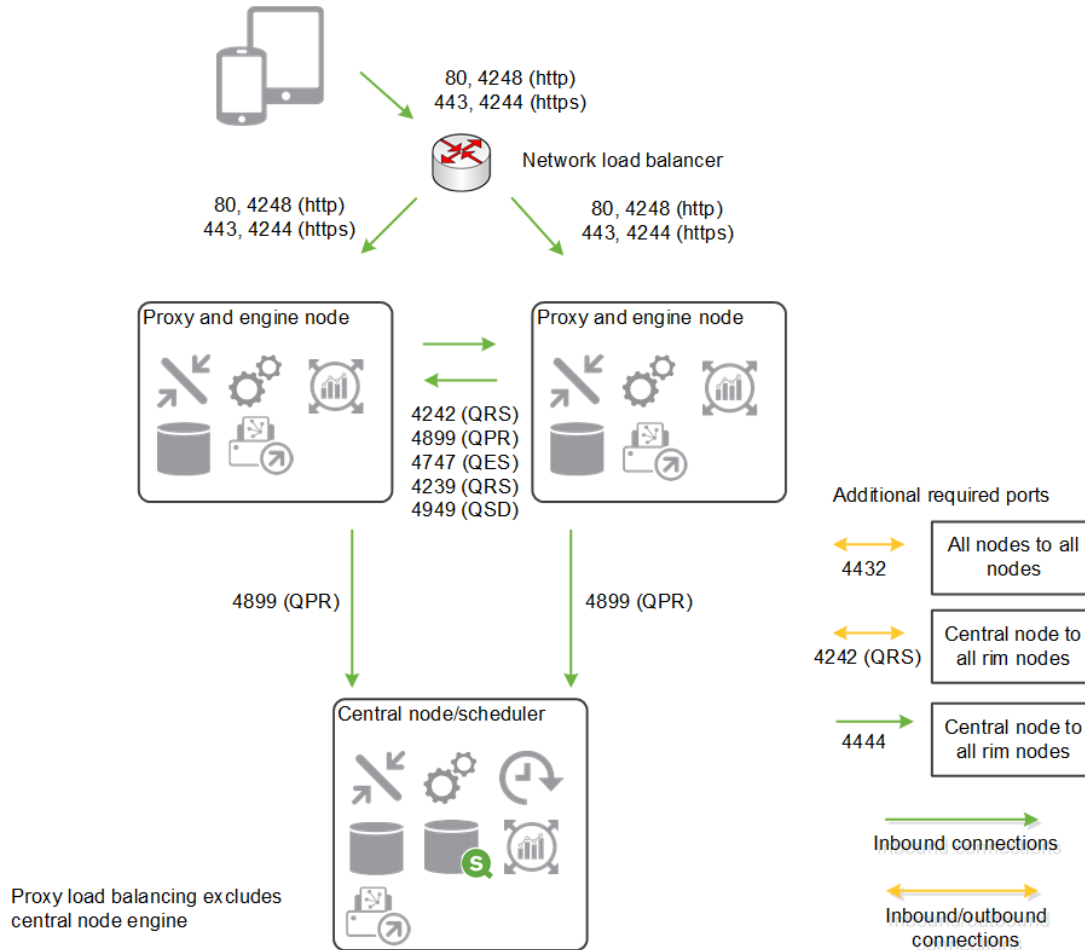
Separate proxy and engine node

This example shows the ports that are used in a multi-node site when deploying a separate proxy and engine node.



High availability proxy and engine nodes

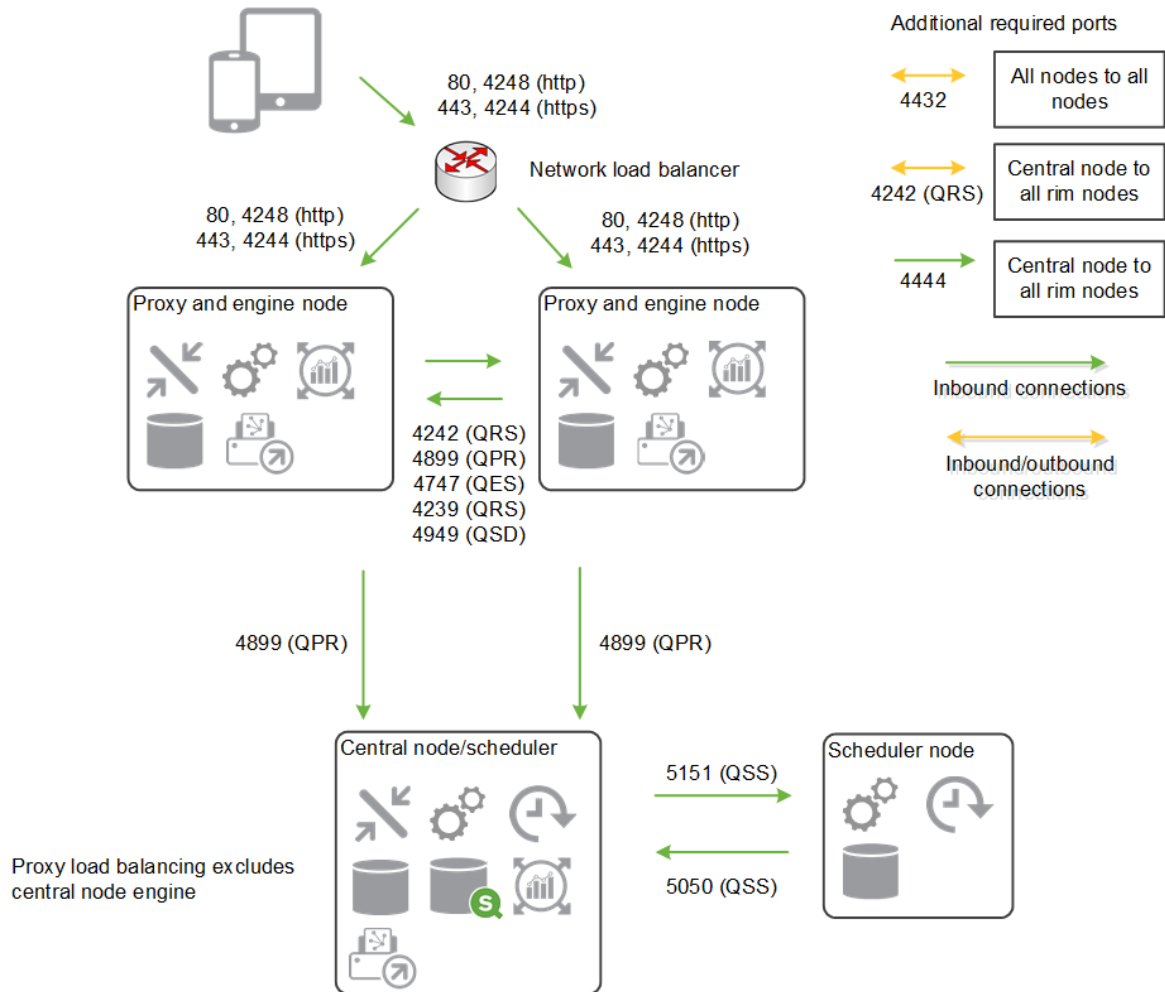
This example shows the ports that are used in a multi-node site when deploying more than one proxy and engine node.



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.

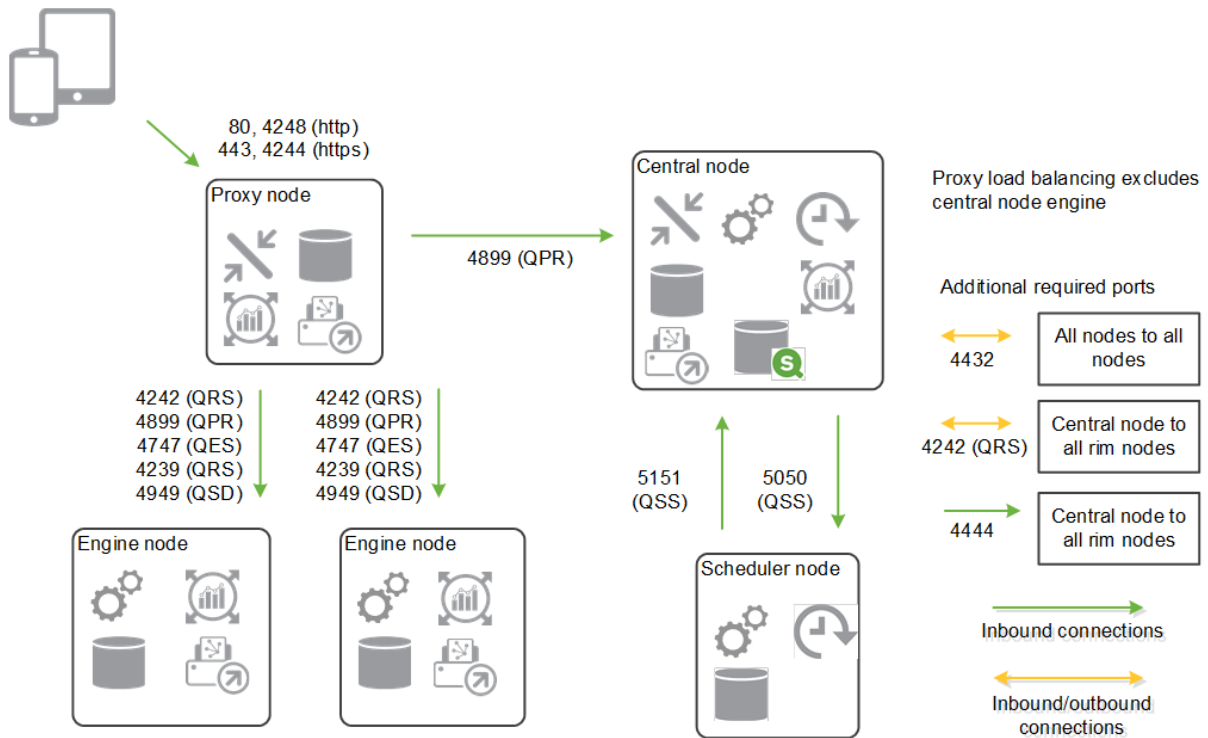
2 Planning your deployment



Separate proxy and scheduler nodes and high availability engine nodes

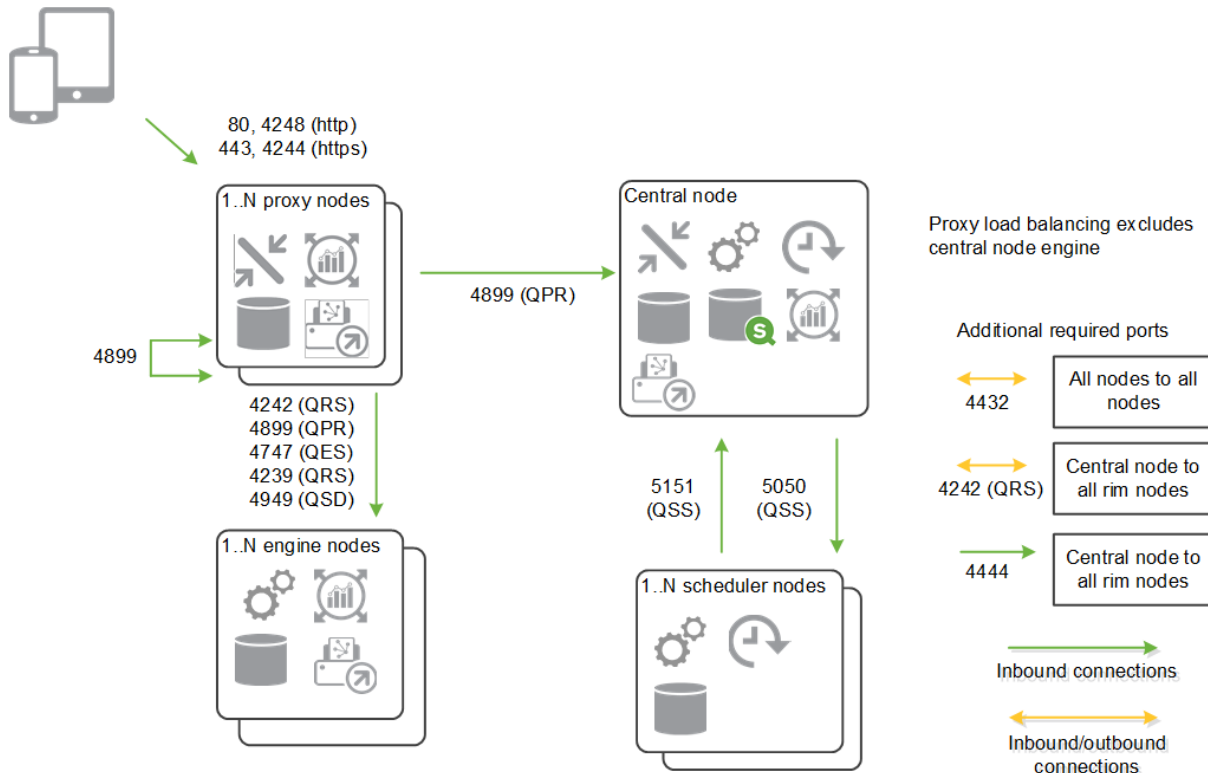
This example shows the ports that are used in a multi-node site when deploying separate proxy and scheduler nodes and more than one engine node.

2 Planning your deployment



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.



Persistence

The nodes in a Qlik Sense site form a cluster that share a single repository database and file share for the application data. This means that any changes made to the nodes are saved directly to the central repository database without delay, resulting in improved performance, availability, and stability. All content in a Qlik Sense site is created and stored in the repository database and in the file share. In a multi-node deployment, one site has a single shared database and a single file share regardless of how many nodes there are in the site. The repository database stores all data relating to the app structure, including the paths to the binary files in the file share.



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

The file share is a shared folder usually on the C:\ drive of your central node that is used to store all the Qlik Sense application data and must be accessible to all nodes in your site. It can also be located on a dedicated server for better resilience and performance. You create this folder before you install Qlik Sense.

See: *Creating a file share (page 68)*

Repository database

In a Qlik Sense site, a single PostgreSQL repository database and file share contain all the data for all nodes.

You have two options when installing the repository database:

- Install as a local database on a central node - This is an option in both single-node and multi-node sites.
- 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 site.

The repository database contains two types of data:

- Entity data: The repository database contains the system configuration and all meta data about apps. This data is referred to as entity data and is usually small in size. The repository database is controlled by the Qlik Sense Repository Service (QRS).
- Binary data: The app data files contain the data models and app definitions. These files are referred to as binary data and the data model element of the files can be large in size. The app data files are controlled by the Qlik Sense Engine Service (QES).

Every time you install a node in your Qlik Sense site you must configure the node to connect to the repository database. All data and applications are available to all server nodes, and any changes that occur are immediately visible to all other nodes in the site. You can use load balancing rules to route users or reloads on particular apps to a specific node in the site.

Basic deployment

In a basic single-node deployment, all services are deployed to a single server. This type of deployment is best suited to a small organization operating within a single time zone.

For larger organizations, an enterprise deployment is recommended, see *Enterprise deployment (page 43)*.

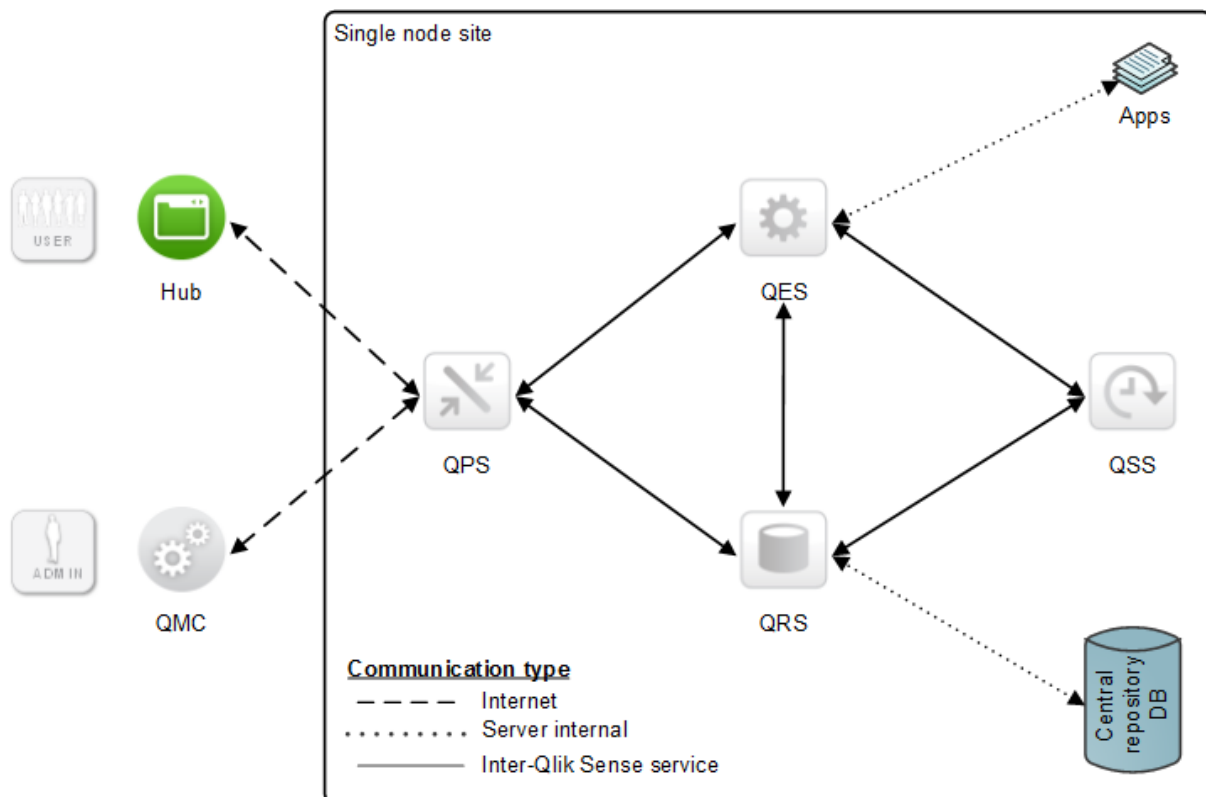
Services

In a single-node deployment, the Qlik Sense services behave as follows:

- Qlik Sense Repository Service
Within a single node site, 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
When deployed in a single node site, the Qlik Sense Scheduler Service (QSS) acts as both master and slave.

Basic single-node deployment example

In this deployment scenario, all Qlik Sense services run on a single node. This kind of deployment works best in a single time zone, where reloads of data can be done during the night.



See: *Architecture (page 16)*

Enterprise deployment

You can configure a Qlik Sense enterprise deployment in a variety of different ways to suit the needs of your organization. For example, you can install Qlik Sense services to run on a single node or on multiple nodes for better performance and scalability. In a small single-node deployment, you deploy all services to a single server, which we do not recommend for larger organizations.

This section provides three examples of Qlik Sense deployments.

The following terms are used in the deployment scenarios:

- Central node: the central point for managing all nodes in a site.
- 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.

Enterprise deployment examples

The scenarios described here are examples of a small, medium and large Qlik Sense enterprise deployments. Every deployment of Qlik Sense is different and these examples only aim to provide a rough indication of what resources would be appropriate for a given workload. The figures included here are flexible, allowing extra capacity for growth and for handling peaks in demand. They are not intended to set a maximum limit on your deployment.

If you have an attribute significantly 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 48)*.

The following table provides some basic performance information for each type of deployment example:

	Single-node (small)	Multi-node (medium)	Multi-node (large)
Apps	50	100	250
Active apps per day	25	50	125
Total users (from UDC)	500	1000	20000
Concurrent users (equals active users within the same)	50	100	400

hour)

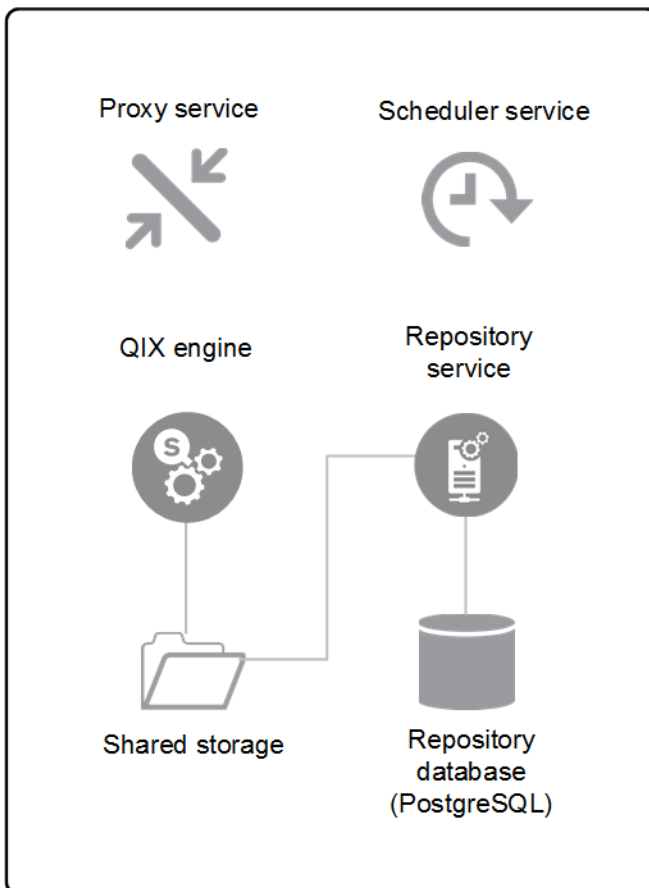
Average app size (in gigabytes)	0.1	0.1	0.1
Maximum app size (in gigabytes)	1	2	5
Content creation (objects per hour)	20	40	50
Reloads per hour	10	20	375



These figures are examples that you can use for guidance but may vary depending on how you have configured your Qlik Sense deployment.

Single-node (small)

This example illustrates a small, single-node Qlik Sense production deployment where all services are configured to run on the same server.



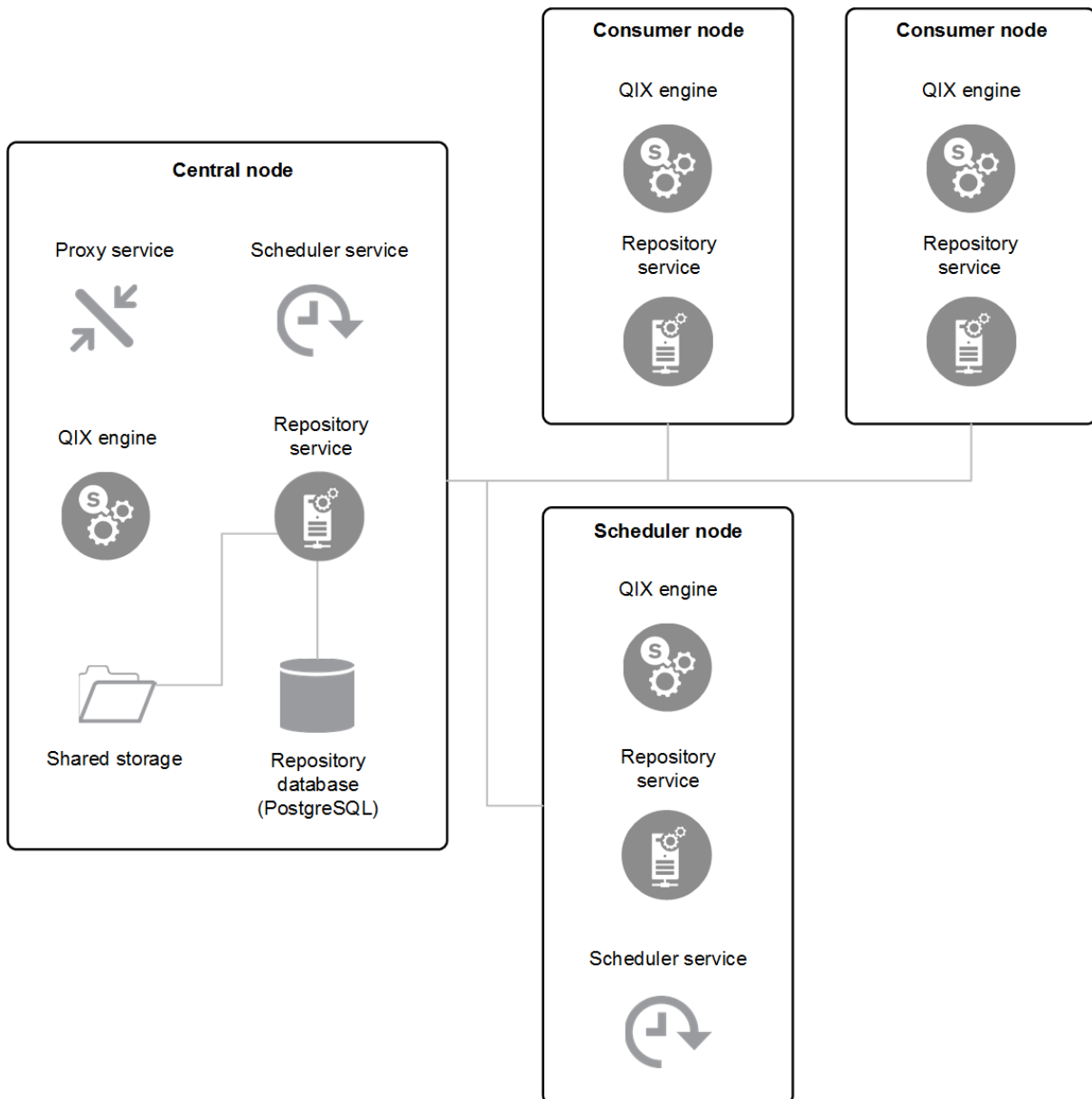
Single-node site

Multi-node (medium)

This example illustrates a typical medium-size, multi-node Qlik Sense production deployment consisting of three nodes:

- Central node
- Consumer node
- Scheduler node

In this configuration, the repository database (PostgreSQL), and the file share are installed together with other Qlik Sense services on the central node. It has two dedicated Consumer nodes and a Scheduler node.



Multi-node site

Multi-node (large)

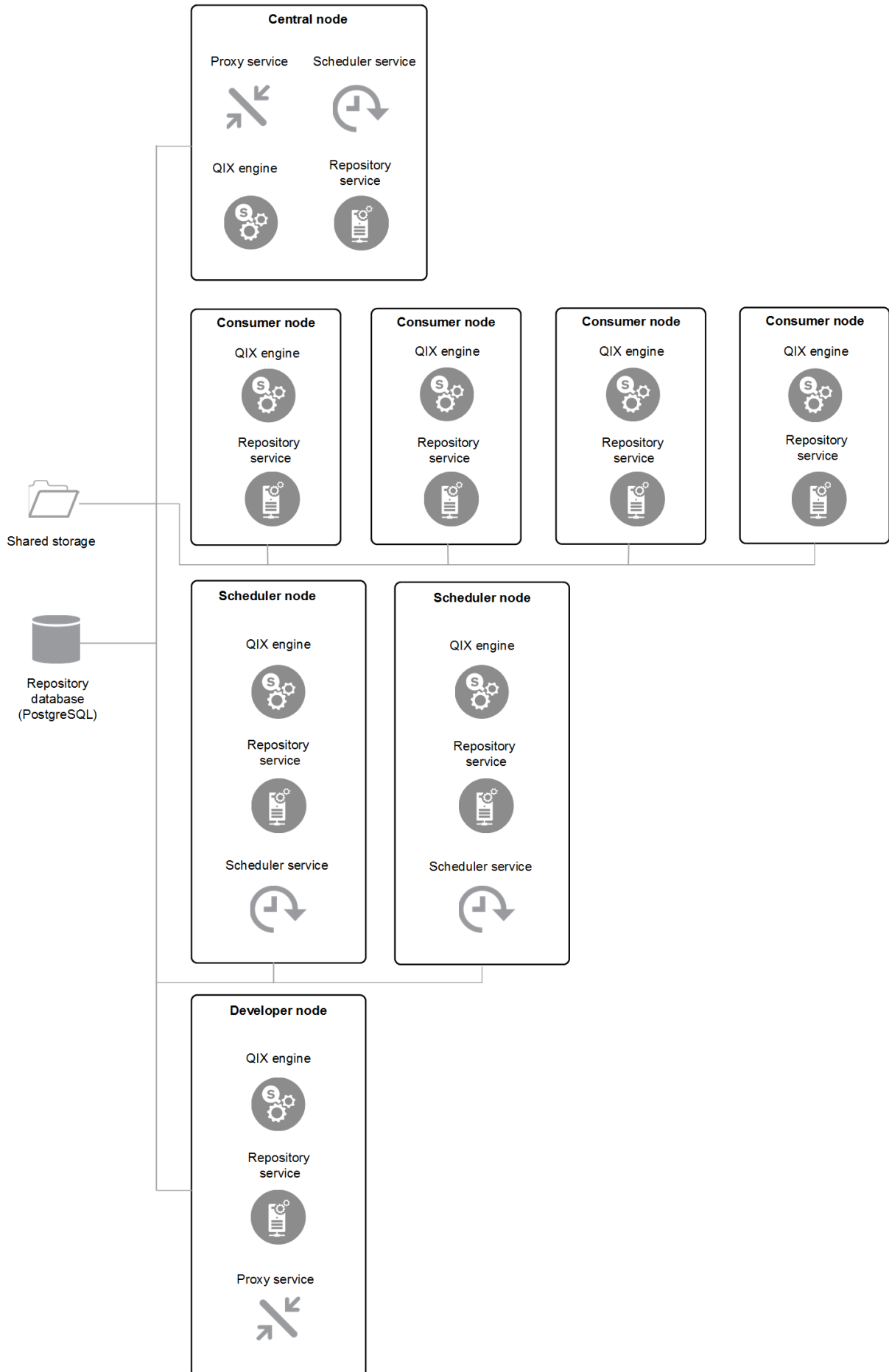
This example illustrates a typical large, multi-node Qlik Sense production deployment consisting of eight nodes, providing the ability to scale up both reloads and user load. This deployment consists of the following nodes:

- Central node
- Four consumer nodes
- Two scheduler nodes
- Developer node



In this configuration example, the repository database (PostgreSQL) and the file share are installed on separate, dedicated servers.

2 Planning your deployment



2.4 Licensing

Licensing allows you to manage the usage of the Qlik Sense software in your organization. The licensing in Qlik Sense is based on tokens, which you use to allocate access passes that give users access to Qlik Sense. There are different types of access passes to choose from and each type corresponds to a specific consumption model:

- User access pass - assigned to unique and identified users allowing them limited use of Qlik Sense apps.
- Login access pass - allocates a block of passes to a group for infrequent or anonymous access. Allows full access for a limited period.

For more information on types of access passes and the consumption model, see [Managing license and tokens](#).



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

Every Qlik Sense site needs at least one License Enabler File (LEF). This file defines the number of tokens available in your site, which you can manage from the central node. You can download the LEF after you have entered the correct serial number and control number in the Qlik Management Console (QMC). If you do not have a network connection, you can paste the LEF directly into the QMC.

If you purchase more tokens, they are added to the pool of unallocated tokens that you can use to allocate access passes in Qlik Sense. As you use up your tokens, any unallocated tokens are removed. However, any tokens that are freed up by the removal of access passes cannot be used for new allocations until the number of allocated tokens drops below the number set in the LEF.

For more information about the LEF, see *License Enabler File (page 190)*.

2.5 Performance

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

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.

Capacity and performance

Qlik Sense supports up to a maximum of 8 machines. In addition to the number of nodes, there are other factors that will contribute to total capacity:

- Workload
- Hardware speed
- 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.

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

DMZ deployments

All machines in a site, including machines 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.

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:

- Master 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 *Failover (page 69)*.

2.6 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 73).

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,

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

- 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.
- Only used for Qlik Sense Windows 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, .

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

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, . 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.

2.7 Security

Security and availability in a shared persistence deployment

In shared persistence deployments the network traffic between the servers, the database and the file share is 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} PASSWORD 'newPassword'
```

Do not change password in the PostgreSQL user interface for the same reasons as above.
- Use md5 hashing.

- Do not set your password to `PASSWORD ''`, that is, an empty string, since this is not handled well in PostgreSQL.

Database traffic encryption

Qlik Sense supports database traffic encryption using SSL, but you need to perform some manual configuration to setup SSL and MD5 password protection in a shared persistence deployment:

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 the following lines in `pg_hba.conf`

```
hostssl  all  all  all  md5
```
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\9.3`.
5. Use the ConnectionString editor to make the following setting in `repository.exe.config` on the central node and all rim nodes that belong to the cluster. `ConnectionStringEditor.exe` is located in
`%ProgramFiles%\Qlik\Sense\Repository\Util\ConnectionStringEditor`. You need to run the executable as Administrator.
Add `'ssl Mode=Require;'` to the connection string:

```
<add name="QSR" connectionString="User ID=qlikenserepository;Password='randompass'; 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;"providerName="Devart.Data.PostgreSql"/>
```
6. Start all Qlik Sense services and verify that everything works.
7. 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'
```

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\9.3\data\postgresql.conf`
Locate and set the following settings
`wal_level = hot_standby`
`max_wal_senders = 3`
`checkpoint_segments = 8`
`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\9.3\bin\psql.exe" -h <machinename> -p 4432 -w -c "CREATE USER replicator REPLICATION LOGIN ENCRYPTED PASSWORD 'secretpassword';"`
3. Open the file `%ProgramFiles%\PostgreSQL\9.3\data\pg_hba.conf`.
At the bottom of the file add:
`host replication replicator 0.0.0.0/0md5`
You can restrict the subnet access further, if required.
4. Restart the PostgreSQL service.

Configure the standby database server

On the standby PostgreSQL database server, do the following:

1. Stop the Postgres service.
2. Delete all content from `%ProgramFiles%\PostgreSQL\9.3\data`.
3. From the command line run the following command adjusted to use the name of the primary server:
`"C:\Program Files\PostgreSQL\9.3\bin\pg_basebackup" -h <primaryServer> -D "C:\Program Files\PostgreSQL\9.3\data" -U replicator -v -P -p 4432 -w`
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\9.3\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\9.3\data`



The failover file should have no file extension.

The file triggers PostgreSQL to cease recovery and enter read/write mode. PostgreSQL also changes the name of the file `recovery.conf` to `recovery.done` to reflect the transition.

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 ConnectionString editor, supplied by Qlik, to decrypt the string, edit it, and write back an encrypted string.
 - a. Open the ConnectionString editor, `ConnectionStringEditor.exe` which is located in `%ProgramFiles%\Qlik\Sense\Repository\Util\ConnectionStringEditor`. You need to run the executable as Administrator.
 - b. Open the file `%Program Files%\Qlik\Sense\Repository\Repository.exe` in the ConnectionString editor.
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. Save the changes to the file.

3 Qlik Sense installation

When you install Qlik Sense for 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.

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

For more information on architecture options and considerations before you install see the following:

- *Architecture (page 16)*
- *Planning your deployment (page 11)*
- *Enterprise deployment examples (page 43)*

When you are ready to proceed with the installation, choose whether to install on a single computer or not:

3.1 Installing Qlik Sense on a single node

A basic installation of Qlik Sense can be done by installing all of the Qlik Sense services on a single node. This kind of deployment works best in a single time zone, where reloads of data can be done during the night. To determine if a single-node installation is the right choice for you, see *Planning your deployment (page 11)*.

For information about multi-node deployments of Qlik Sense, see *Installing Qlik Sense in a multi-node site (page 59)*.

Before you install:

- Check that your environment meets the system requirements.
See: *System requirements for Qlik Sense (page 12)*
- Check that the required ports are available.
See: *Ports (page 28)*
- Check that your browser is supported.
See: *Supported browsers (page 14)*
- Prepare the user accounts required to run the Qlik Sense services.
See: *User accounts (page 49)*
- Understand how Qlik Sense uses LEF for site licensing and tokens for user access allocation, and have your license key available.
See: *User accounts (page 49)*

Do the following:

1. Log in to the computer where you plan to install Qlik Sense as a local Windows administrator.
See: *User accounts (page 49)*.
2. Create a file share before you run the Qlik Sense setup. The file share is a shared folder that stores all the Qlik Sense application data.
 - a. Create a new folder.
 - b. Right click on the folder, and click **Properties**.
 - c. On the **Sharing** tab, and click **Share**.
 - d. Enter the names of Windows users that you want to share the folder with, and click **Add**.
Share this folder with your Windows Qlik Sense administrator and your Windows Qlik Sense services user. For more information, see *User accounts (page 49)*.
 - e. In the **Permission level** column, select **Read/Write**, and click **Share**.



Make note of the network path displayed on the confirmation screen. You will enter this information during the Qlik Sense setup. The network path will be in the following format: \\server-name\QlikShare

3. Download the *Qlik_Sense_setup.exe* file from www.qlik.com, and launch the setup.
4. Do the following:
 - a. Accept the license agreement, and click **Next**.
 - b. On the **Create or join a cluster screen**, click **Create cluster**.
 - c. On the **Shared persistence database connections settings** screen, leave the **Install local database** check box selected if you want to install a local repository database, or clear the check box if you want to connect to an existing repository database.
See: *Manually installing a repository database in PostgreSQL (page 70)*

If you want to connect to an existing repository database, then enter the following values:

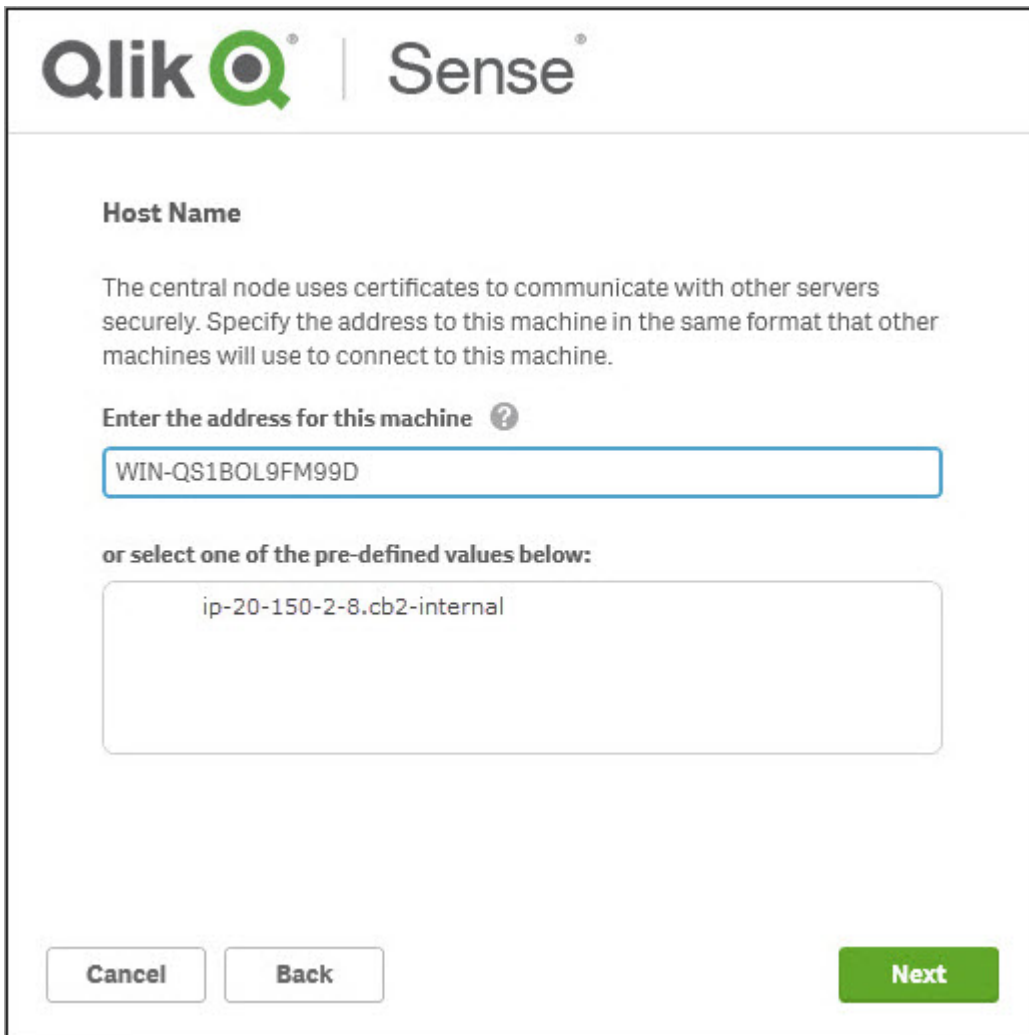
Field name	Value
Database host name	Enter the full URL to your repository database.
Database port	4432
Database user	Enter the username that will be used to access the database.



Do not enter the username postgres.

Database user password	Create your own database user password to access your repository database in the PostgreSQL database.
------------------------	---

- d. On the **Database service listener** screen, click **Next**.
There is no need to configure the database service to allow connections from other nodes in a single-node installation.
 - e. On the **Shared persistence storage screen**, enter the path or URL to your file share, for example \\<domain>\QlikShare, and click **Next**. Your file share can either be a local folder or a remote folder.
 - f. On the **Installation location** screen, choose your own installation location or install Qlik Sense to the default location on the C:\ drive, and click **Next**.
 - g. On the **Repository Database Superuser Password** screen, enter a password for the PostgreSQL repository database superuser. Confirm the password and click **Next**.
It is possible, but not recommended, to proceed without creating a password. See, *User accounts (page 49)*.
 - h. On the **Service Credentials** screen, enter the domain, user name and password for the account that you want use to run the Qlik Sense services, and click **Next**.
5. On the **Host Name** screen, enter the name of the computer that you are installing Qlik Sense on and click **Next**.



Qlik | **Sense**

Host Name

The central node uses certificates to communicate with other servers securely. Specify the address to this machine in the same format that other machines will use to connect to this machine.

Enter the address for this machine ?

or select one of the pre-defined values below:

Cancel **Back** **Next**

6. On the **Ready to install** screen, select the appropriate check boxes if you want the setup to create desktop shortcuts and automatically start the Qlik Sense services when the setup is complete.



*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, clear the **Start the Qlik Sense services when the setup is complete** checkbox.*

7. Click **Install**.
You will see a message indicating that Qlik Sense has been installed successfully.
8. Click **Finish**.
9. 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 *Changing the user account to run Qlik Sense services (page 73)*

You are 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 QMC.
When you open the QMC for the first time the **Site license properties** screen is displayed.
2. Enter the license information from the License Enabling File (LEF).
The property group **Site license** contains properties related to the license for the Qlik Sense system. All fields are mandatory and must not be empty.

Property name	Description
Owner name	The user name of the Qlik Sense product owner.
Owner organization	The name of the organization that the Qlik Sense product owner is a member of.
Serial number	The serial number assigned to the Qlik Sense software.
Control number	The control number assigned to the Qlik Sense software.

3. Expand **LEF access** and click **Get LEF and preview the license**. If you received your LEF via email, you can copy and paste the information into the text field.



***Failed to get LEF from server** is displayed if the serial number or control number is incorrect.*

4. Click **Apply**.

Successfully licensed is displayed.

5. Click **Close**.

You have activated your Qlik Sense site license.

You are ready to connect to a user directory (optional), allocate user access, and set up permissions. See: (ref to connect user directory topic & set up permissions topic).

Allocating user access

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 and tokens**.
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 install Qlik Sense, you may want to:

- Create load balancing rules in the QMC to improve resilience and performance in a multi-node site. For more information, see [Load balancing](#).
- Configure the virtual proxy advanced settings to add your own hosts names to the white list. For more information, see [Host white list](#).
- Configure the user directory connector to retrieve users from a user directory. For more information, see [User imports \(UDC\)](#).

You are now ready to start using Qlik Sense. See: [Get started](#).

3.2 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.

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

For more information about single-node deployments of Qlik Sense, see *Installing Qlik Sense on a single node* (page 55).

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

- *Planning your deployment* (page 11)
- *Architecture* (page 16)
- *Security* (page 51)
- *Performance* (page 48)

Before you install:

- Check that your planned environment meets the system requirements.
See: *System requirements for Qlik Sense* (page 12)
- Prepare the user accounts required to run the Qlik Sense services on the computer where you plan to install Qlik Sense.
See: *User accounts* (page 49)
- Ensure that your firewall is enabled and you have created the appropriate rules to allow rim nodes to communicate with the central node:
 - Central node - port 4432
 - Consumer node - ports 4242, 4747, 4748
 - Scheduler node - ports 4242, 5050, 5151
 - Certificates - port 4444 (required when registering a rim node)

See: *Ports* (page 28) for a full list of ports.

- Repository database - if you already have a Qlik Sense repository database on another server from a previous installation, you can continue to use this in your new deployment. If you do not intend to use this database then remove it before you start.
- Create a local file share to store your Qlik Sense application data.
See: *Creating a file share* (page 68)
- Understand how Qlik Sense uses tokens for licensing, and have your license key available.
See: *Licensing* (page 48)

This topic includes the following sections:

- *Installing Qlik Sense* (page 60)
- *Adding a Qlik Sense node* (page 66)

Installing Qlik Sense

You can install a Qlik Sense server as either a central node or as a rim node. In a multi-server site, rim nodes must be connected to a central node. See: *Architecture* (page 16). If you are 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. For more information on how to configure a failover candidate, see [Creating a node](#) and [Service cluster](#).

To install a node:

1. Create a file share before you run the Qlik Sense setup. The file share is a shared folder that stores all the Qlik application data and must be accessible to all nodes in your Qlik Sense site. You can create a file share either on the same server computer as the central node or on another server.
See: *Creating a file share (page 68)*
2. Log in to the computer where you plan to install Qlik Sense as a domain or local Windows administrator. You must have full administrator rights to run the Qlik Sense setup. You can start the Qlik Sense services as either an administrator or a local user without administrator privileges.
See: *User accounts (page 49)*.
3. Download the *Qlik_Sense_setup.exe* file from www.qlik.com.
4. Run the installation program as an administrator, and on the first screen click **Install**.
5. Read the **License agreement** screen. If you agree, select the check box and click **Next**.
6. On the **Create or join a cluster** screen, you have two options:
 - **Create cluster** - To install a central node. All the other nodes in your site will connect to this node.
 - **Join cluster** - To install a rim node that connects to a central node (if you choose this option, fewer screens are displayed in the setup).
7. On the **Shared persistence database connections** settings screen, leave the **Install local database** check box selected if you want to install a local repository database, or clear the check box if you want to connect to an existing repository database hosted on another server.
See: *Manually installing a repository database in PostgreSQL (page 70)*

If you want to install a local repository database, then enter the following values:

Field name	Value
Database host name	<i>localhost</i>
Database port	<i>4432</i>
Database user	<i>qliksenserepository</i>




Do not enter the username postgres.

Database user password Create a password to access the local repository database.

If you want to connect to an existing repository database on another server, then enter the following values:

Field name	Value
Database host name	Enter the full URL to your repository database.
Database port	4432
Database user	<i>qliksenserepository</i>
	This is the login role you created in the PostgreSQL database (QSR)
Database user password	Enter the password you created in PostgreSQL.


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.

- On the **Database service listener** screen, configure the listen addresses and IP range to allow connections from other nodes, and 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: *Manually installing a repository database in PostgreSQL (page 70)*


Enter the following values:

Field name	Value	Description
Listen addresses	*	The IP address(es) to listen on. Use the value * to allow access for all IP addresses.
IP Range	0.0.0.0/0	To allow all servers to access the repository database, use the value 0.0.0.0/0 (for all IPv4 addresses) or :: (for all IPv6 addresses).

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

- On the **Shared persistence storage** screen, enter the path or URL to your file share, for example `\\<domain>\QlikShare` and click **Next**. Your file share can either be a local folder or a remote folder on another server.

See: *Creating a file share (page 68)*

 This screen does not appear if you are installing a rim node (**Join cluster** option).

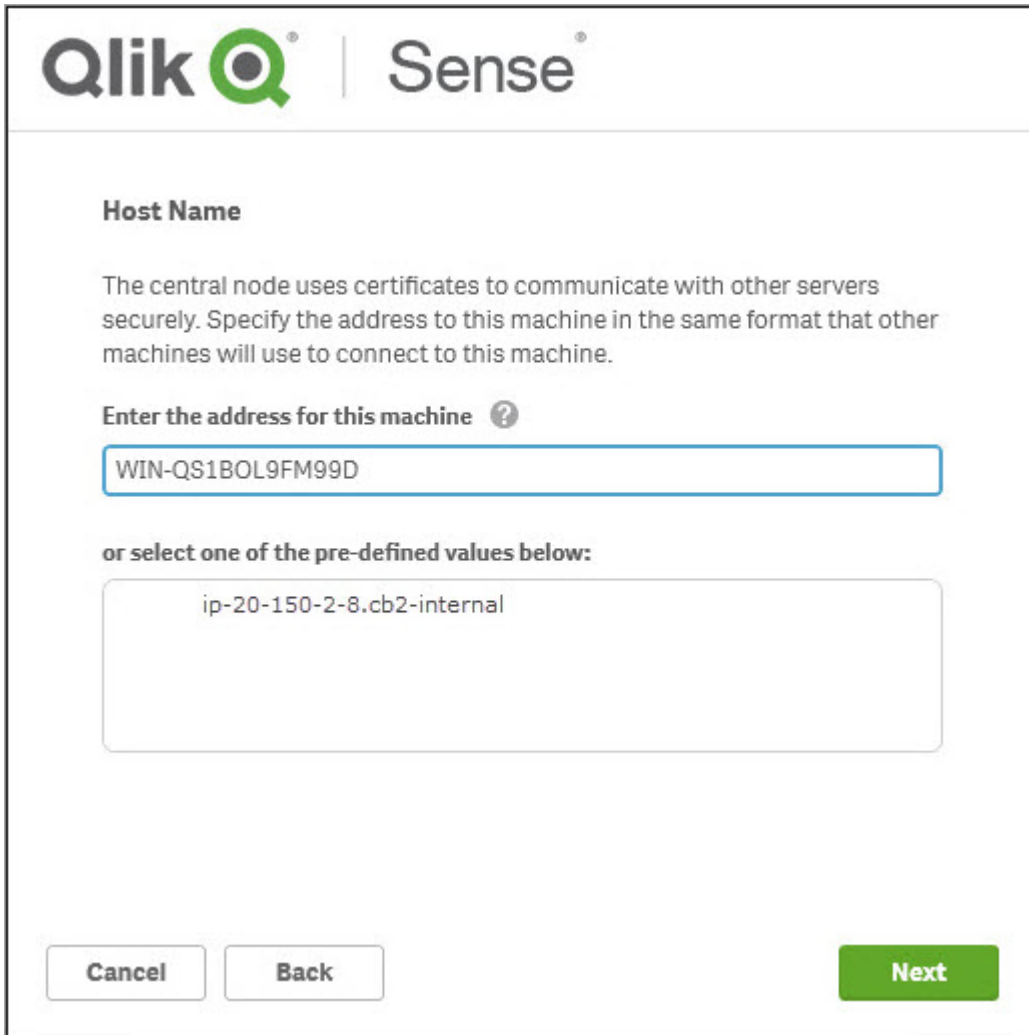
- On the **Installation location** screen, choose a location to install Qlik Sense or use the default location on the C:\ drive, and click **Next**.

11. On the **Repository Database Superuser Password** screen, create a superuser password for the PostgreSQL database, and click **Next**. Enter the same **Database user** password that you created earlier in the **Shared persistence database connections settings** screen.



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

12. On the **Service Credentials** page, enter the domain, user name and password for the account that you want use to run the Qlik Sense services, and click **Next**.
See: *User accounts (page 49)*
13. On the **Host Name** screen, enter the address for the Qlik Sense node that you are installing, and click **Next**. The address must be in a format that other nodes can use when connecting to this node, otherwise the connection will fail.
For example:
 - IP address: 10.1.123.234
 - Machine name: WIN-QS1BOL9FM99D
 - Fully qualified machine name: WIN-QS1BOL9FM99D.CUSTOMER.COM



The screenshot shows the 'Host Name' configuration screen in the Qlik Sense installation wizard. At the top, the Qlik Sense logo is displayed. Below the logo, the title 'Host Name' is centered. A paragraph explains that the central node uses certificates for secure communication and that the user must specify the machine's address in the same format as other machines. There are two input options: a text field labeled 'Enter the address for this machine' with a question mark icon, containing the text 'WIN-QS1BOL9FM99D', and a list box below it labeled 'or select one of the pre-defined values below:' containing the value 'ip-20-150-2-8.cb2-internal'. At the bottom, there are three buttons: 'Cancel', 'Back', and 'Next'.



Ensure that the recommended server node name displayed in the **Enter the address for this machine** field matches the one you will use to access this node, otherwise enter an appropriate address or fully qualified domain name. Only use the fully qualified name if you understand the full implications.

14. On the **Ready to install** screen, select the appropriate check boxes if you want the setup to create desktop shortcuts and automatically start the Qlik Sense services when the setup is complete.



If you wish to use a dedicated service account to run the Qlik Sense services, clear the **Start the Qlik Sense services when the setup is complete** checkbox.

15. Click **Install**.
You will see a message indicating that Qlik Sense has been installed successfully.
16. Click **Finish**.

17. 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 49)*

You are ready to license your Qlik Sense installation.

Licensing Qlik Sense

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

To activate your license:

1. Open the QMC.
2. When you open the QMC for the first time the **Site license properties** page is displayed.
3. Enter the license information from the *License Enabler File (page 190) (LEF)*.
The property group **Site license** contains properties related to the license for the Qlik Sense system. All fields are mandatory and must not be empty.

Property name	Description
Owner name	The user name of the Qlik Sense product owner.
Owner organization	The name of the organization that the Qlik Sense product owner is a member of.
Serial number	The serial number assigned to the Qlik Sense software.
Control number	The control number assigned to the Qlik Sense software.

4. Expand **LEF access** and click **Get LEF and preview the license**. If you received your LEF via email, you can copy and paste the information into the text field.



Failed to get LEF from server is displayed if the serial number or control number is incorrect.

5. Click **Apply**.
Successfully licensed is displayed.
6. Click **Close**.

You have activated your Qlik Sense site license.

You are ready to connect to a user directory (optional), allocate user access, and set up permissions. See: (ref to connect user directory topic & set up permissions topic).

Allocating user access

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 and tokens**.
The **License usage summary** page is displayed.
2. Click the **User access allocations** tab.
3. Click the **+ Allocate** button.
The **Users** page 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.

If you have created a rim node, you are now ready to register the rim node with the central node.

Adding a Qlik Sense node

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:

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>
Node purpose	Choose a suitable purpose for the node: <ul style="list-style-type: none">• Production• Development• Both	For example, choose Production for a scheduler node or Development for a

Node configuration	Select this node as a failover candidate.	developer node used for creating apps. For more information on types of nodes, see: Creating a node Check that your license supports the node purpose that you have chosen. 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>Failover (page 69)</i>
Service activation	Select the services you want to run on this server node: <ul style="list-style-type: none">• Repository• Engine• Printing• Proxy• Scheduler	For example, if you are installing a consumer node, select the Repository and Engine services. For more information on which services to run on different types of nodes, see: <i>Architecture (page 16)</i> and <i>Services (page 19)</i>

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.
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 white list](#) - configure the virtual proxy advanced settings to add your own hosts names to the white list.
- [User imports \(UDC\)](#) - configure the user directory connector to retrieve users from a user directory .

You are now ready to start using Qlik Sense.

See: [Get started](#).

3.3 Creating a file share

Creating a file share or shared folder is a necessary prerequisite before you install Qlik Sense. The file share is used to store all the Qlik application data and 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.

To do this:

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

3.4 Failover

To avoid having a single point of failure in a multi-node site, when you add a new node to your deployment you can assign it the role of failover candidate. This means that any server or node in your Qlik Sense site can perform the same role as the central node. The role of the central node can now be swapped, for example if the central node has been offline for more than 10 minutes.

Automatic failover

After you have configured a node to become a failover candidate, each node in your site will regularly check the primary node (central node) for a heartbeat. If there is no communication between the primary node and the other nodes in the site after 10 minutes then the primary node will be replaced by the next available node. If more than one node is set as a failover candidate each node will compete to get a lock on a database field and the winner becomes the central node. There is an additional field in the QMC to show which node is currently the central node.

Manually migrating the central node

If you decide that you want to move the central node to another node in your site, you can manually migrate it using the the following REST API calls:

- Get `/qrs/serverNodeConfiguration` to get a list of server GUIDs.

- Do an empty POST to `/qrs/failover/tonode/{serverNodeConfigurationID}` to retrieve the ID of the node you want to migrate to.

3.5 Manually installing a repository database in PostgreSQL

When installing a Qlik Sense multi-node enterprise site, this is an optional step that you can perform if you want to install your repository database on a dedicated remote server. If instead, you want to install a local repository database then select the **Install local database** check box during setup. In Qlik Sense, the repository database is a PostgreSQL database.



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 9.6. 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.

To install a dedicated PostgreSQL repository database there are several steps that you need to follow:

- Run the PostgreSQL setup
- Create a PostgreSQL database and **Login Role**
- Verify that the database has installed and is running correctly
- Configure the following files:
 - `postgresql.conf` file
 - `pg_hba.conf` file

Running the PostgreSQL setup

1. Log in to the server where you want to install PostgreSQL as an administrator.
See: *User accounts (page 49)*
2. Download PostgreSQL version 9.6 from the [PostgreSQL](#) website.
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 with the central node.

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.

Creating a repository database and a Login Role

1. Open the *pgAdmin* tool.
2. In the **Object browser**, under **Servers**, right click the PostgreSQL node and then click **Connect**.
3. Enter your PostgreSQL superuser password to make a connection.
4. Right click the **Databases** node and then click **New Database**.
5. Enter the name **QSR** for the repository database that you are creating.
6. Right click the **Login Roles** node, and then click **New Login Role**, to create a new database user.
7. In the **New Login Role** window, in the **Properties** tab, enter the name *qliksenserepository*.
8. In the **Role privileges** tab, assign full privileges to this **Login Role** by selecting all check boxes.
9. In the **Definition** tab, enter a password of your choice.
When you run the Qlik Sense setup, in the **Shared persistence database connections settings** screen, you are asked to enter the **Database user** password that you created here so that Qlik Sense can connect to the repository database.
10. Make *qliksenserepository* the owner of the repository database. To do this, right click the **QSR** database you created earlier and then click **Properties**.
11. In the **Properties** tab, in the **Owner** drop-down, select *qliksenserepository* as **Owner** of the **QSR** database.

You have created a PostgreSQL repository database, now verify that it has installed correctly.

Verifying that database has installed and is running correctly

1. Open the *pgAdmin* tool.
2. Under **Databases**, select the **QSR** database you created.
3. In the **Properties** window, you can see a list of properties. If the **Connected?** property has the value **Yes** then your database is running correctly.

Editing the configuration files

Edit the following configuration files so the repository database can listen for connections from Qlik Sense nodes. If you choose the option **Install local database** in the Qlik Sense setup, you can also specify these settings as part of the setup.

To edit the *postgresql.conf* file:

1. Navigate to the *postgresql.conf* file in *C:\Program Files\PostgreSQL\<version>\data* of your PostgreSQL installation.

- To edit this file, you must first copy it to another location, such as to the Windows Desktop, before making your changes.
- Make the following configuration changes :

Setting	Value	Description
listen_addresses	*	Listen for connections from all IP addresses.
max_connections	600	To calculate this value multiply x 100 the number of servers in your deployment.



If you set the listen address to '' the database service will listen on all IP addresses. For more information on how to set a more restrictive IP address, see the PostgreSQL documentation.*

- Save your changes and copy the file back to its original location overwriting the contents of the file.

To edit the `pg_hba.conf` file:

- Navigate to the `pg_hba.conf` file in `C:\Program Files\PostgreSQL\<version>\data` of your PostgreSQL installation.
- Copy this file to another location and open it in a text editor.
- 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 setting, 127.0.0.1/32, only allows local host to access the database.
- Replace 127.0.0.1/32 with a sub net specification that covers all the IP addresses of the nodes in your site.

When doing this you have the following options:

- 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.
 - Use 0.0.0.0/0 to allow all IPv4 addresses to access the repository database.
 - Use :: to allow all IPv6 addresses to access the repository database.
- Save your changes and copy the file back to its original location overwriting the contents of the file.

You have installed and configured a PostgreSQL repository database on a separate server. You are now ready to resume your installation of Qlik Sense.

Verifying that all the nodes in your site are connected to the repository database

When you have configured one or more nodes in your Qlik Sense site, you can verify the connection to the PostgreSQL database by viewing the *Server statistics report* in PostgreSQL.

To generate the **Server statistics report**:

1. Open the *pgAdmin* tool.
2. In the **Object browser**, right click the **PostgreSQL localhost** database server node.
3. Click **Reports**, and then click **Statistics Report**.
4. In **Report Title**, enter a suitable name for your report, such as *postgresql statistics report.html*.
5. In **Output file**, browse to a suitable location to save your report, and click **OK**.
6. Open the *Server statistics* report in a web browser and view the rows for **qliksenserepository**, and the **QSR** database. In the **Client** column, you can see the IP addresses of any Qlik Sense servers that are connected to PostgreSQL.

3.6 Changing the user account to run Qlik Sense services

If your company policy requires you to run the Qlik Sense services as a user without administrator privileges you can either switch user accounts during the installation process or after you have installed your Qlik Sense site.

Changing the user account to run the Qlik Sense services during installation

If you are installing a central node you can follow the same procedure as a regular administrator installation. However, if you want 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.

To install a node:

1. Log in to the computer where you plan to install Qlik Sense as an administrator.
See: *User accounts (page 49)*.
2. Download the *Qlik_Sense_setup.exe* file from www.qlik.com
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 URL and password to the repository database that you want to connect to.
See: *Installing Qlik Sense (page 60)*
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*.

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. While the bootstrap is running, in the QMC on the central node, register the rim node with a certificate that is generated. For more information, in *Services (page 19)*, see the [Repository service](#).
3. On the central node, register the rim node in the QMC, see: *Adding a Qlik Sense node (page 66)*. 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.

Changing the user account type to run the Qlik Sense services on an existing site

If you use 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, you must choose not to start the service during the installation.

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
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 -iscentral`.
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**.

The services are as follows:

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



Depending on your setup some of the services may not be available.

12. Start the Qlik Sense Service Dispatcher, and then the Qlik Sense Repository Service (QRS).
13. Start the rest of the Qlik Sense services.

3.7 Performing a silent installation

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.



Note that elevation will take place if run from an unelevated process and the UAC is on.

Syntax

```
Qlik_Sense_setup.exe [-silent] {-log "path\filename"} {layout="path"}  
{rimnode=1|0} {rimnodetype="NodeTypename"} {desktopshortcut=1|0}  
{skipstartservices=1|0} {installdir="path"}  
{userwithdomain="domain\user"} {userpassword="password"}  
{dbpassword="password"} {hostname="www.hosturlofyourmachine.com"}  
{sharedpersistenceconfig="configfilepath"} {skipvalidation=1|0}
```


```
Qlik_Sense_setup.exe -? or -h
```

Brings up the on-screen silent setup help.

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.
-layout	[destination directory]	Extracts files (including .msi files) to the destination directory.

 *The user must have access to this directory.*

 *This argument should not be combined with other command line arguments.*

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.

rimnode	1 0	Determines the Repository role.
rimnodetype	Complete Proxy Engine ProxyEngine Scheduler	Installs all the features required for the rim node type selected.
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 (<i>%ProgramFiles%\Qlik\Sense</i>).
userwithdomain	[domain\username]	The username used to run the Qlik Sense services.

3 Qlik Sense installation

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.
hostname	[address of the central node]	The central node uses certificates to communicate securely with other servers. Leave blank to use default.
sharedpersistencenfig (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.



This is a parameter must be configured to install successfully.

See: *Shared persistence configuration file syntax (page 78)*

skipvalidation	1 0	To skip validation of the password provided for service user.
-----------------------	-----	---

Example 1:

```
Qlik_Sense_setup.exe -s rimnode=1 rimnodetype=Scheduler
```

Example 2:

```
Qlik_Sense_setup.exe -s userwithdomain=mydomain\myUser  
password=myPassword
```

Example 3:

```
Qlik_Sense_setup.exe -s -l "c:\mylogpath" installdir="c:\mycustompath"
dbpassword=mydbpassword rimnode=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
```

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/hostname</DbHost>
  <DbPort>4432</DbPort>
  <RootDir>\\server\share</RootDir>
  <StaticContentRootDir>\\server\share\StaticContent</StaticContentRootDir>
  <CustomDataRootDir>\\server\share\CustomData</CustomDataRootDir>
  <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</IpRange>
  <!--<JoinCluster>true</JoinCluster>-->
</SharedPersistenceConfiguration>
```

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 <code>StaticContentRootDir</code> , <code>CustomDataRootDir</code> and <code>ArchivedLogsDir</code> settings.





Setting	Description
AppsDir	Directory to store apps in.
StaticContentRootDir	Root directory for all static content of the site.
CustomDataRootDir	Root directory for all custom data of the site, for example, custom connectors.
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 <code><!--<JoinCluster>true</JoinCluster>--></code> .
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 <code>0.0.0.0</code> (for all IPv4 addresses), <code>::</code> (for all IPv6 addresses) or <code>*</code> (for all addresses).
IpRange	Subnet specification that covers the IP addresses of all nodes in your site. Either add one row for each node, using <code>/32</code> as suffix for each address, or add a subnet that covers all addresses using, for example, <code>/24</code> as suffix. To allow all servers to access the repository database, use <code>0.0.0.0/0</code> .

Deprecated command line arguments

The use of the following command line arguments is no longer recommended.

-rimnodetype (or -rnt) Installs all the features required for the rim node type selected. The node type can be any one of:

Complete, Proxy, Engine, Scheduler.

-addfeatures (or -a)	Install the named features. Can be all or any of: Engine, Proxy, Scheduler, PostgreSQL
	 <i>The silent feature names are case sensitive. For example this command will not start the installation: <code>-s -a engine</code>. The correct command is: <code>-s -a Engine</code>.</i>
	 <i><code>-a Engine Scheduler</code> will also install mandatory features, such as repository.</i>
-removefeatures (or -r)	Removes the named features
-desktopshortcut	To add a desktop shortcut.
-skipstartservices	To skip starting services after the installation has finished.
-installdir (or -i)	Path to custom install directory. Need only be defined if the default install directory will not be used (<code>%ProgramFiles%\Qlik\Sense</code>).
-dbpassword (or -dbp)	Password for the Repository database superuser that creates the user that runs the database.
	 <i>Use carefully! The actual password will be visible in the log file. To mitigate this limitation please remove the installation log files after installation.</i>
-hostname	Host name of the central node.
-userwithdomain (or -u)	Dictates the account that runs the services: <code>Domain\User</code>
-password (or -p)	Password for the account that runs the services.
	 <i>Use carefully! The actual password will be visible in the log file. To mitigate this limitation please remove the installation log files after installation.</i>
-cleanup	When the uninstall has completed, the Qlik Sense certificates and any file in the <code>ProgramData\Qlik\Sense</code> folder will be deleted

3.8 Setting up Qlik Sense after installation

This section guides you through the process of setting up your Qlik Sense site after installing. You can configure the server to fit with your organization's particular needs. Below are the common task most deployments will require.

Connecting Qlik Sense to your user directory

Qlik Sense has a range of methods for authenticating users. Windows authentication is the default method.

When a user connects to Qlik Sense for the first time a user record is created to identify that user. Once this record is created, the administrator can track the user's activity and assign her a license and permissions.

Administrators can also connect to a user directory (for example, Active Directory or LDAP) to obtain further information about that user (such as user groups). The user information can be fetched in advance and then kept in sync with the user directory. This is optional but recommended since it will provide you with the best management experience.

See the following sections in the *Manage Qlik Sense sites* guide:

- Setting up a user directory connector and schedule by task
- Managing users

Assigning licenses to users

Users need a license to open an app. Qlik Sense offers a token-based system that allows the administrator to assign the most suitable license type to each user. Licenses can be allocated on an individual basis or automatically by using rules to define who is allowed to obtain a license, for example, all users in a specific department.

See the following sections in the *Manage Qlik Sense sites* guide:

- Managing license and tokens
- Allocating user access
- Creating new login access

Configuring the monitoring apps

All installations of Qlik Sense (single node and multi-node) require configuration of the monitoring apps for them to work properly.

See the following sections in the *Monitor Qlik Sense sites* guide:

- Configuring the monitoring apps

How Qlik Sense uses HTTPS and certificates

Qlik Sense deploys securely by default when it is installed. It uses self-sign certificates to ensure that data is transferred to users in a secure way. When users access the system they by default receive a warning that the certificate used by the site is not trusted. The user can then accept the certificate and proceed to use Qlik Sense securely.

There are two options to prevent the warning; one is to use a trusted certificate from either a trusted provider or an internal corporate source. The other option is to run the site using HTTP only. Both options are available as settings on the Qlik Sense Proxy Service (QPS).

Regardless of which option you choose, the services in Qlik Sense always use encryption when communicating.

See the following sections in the *Manage Qlik Sense sites* guide:

- Changing proxy certificate
- Editing proxies

Creating and opening apps

To create and open apps on the server users must browse to the Qlik Sense hub using their web browser. The hub lists two areas: **Work** contains the apps belonging to the user who has logged in, and **Streams** contains the other apps the user has access to. After the installation the administrators see two built-in monitoring apps, while all other users do not see any apps. Click on an existing app to open it. To create a new app, click **Create new app**.

Disabling search indexing (only multi-node)

In a multi-node environment, you need to disable the QMC engine setting **Create search index during reload** (in the **Advanced** properties section), if the reload node is not the node on which the user performs searches. If you do not disable this option, an index is created when you reload, which consumes time and disk space to no advantage.

See: [Manage Qlik Sense sites: Editing an engine](#)

See: [Script syntax and chart functions: CreateSearchIndexOnReload](#)

See the following section in the *Manage Qlik Sense sites* guide: Editing an engine

Working with streams, apps and publishing

A stream is a way to group together apps that have similar permissions. Once an app has been created, an administrator can publish it to a stream. The app then becomes visible to users who have access to that stream. Apps, streams and publishing are managed in the Qlik Management Console (QMC).

See the following sections in the *Manage Qlik Sense sites* guide:

- Managing streams
- Managing apps
- Publishing app

See also:

 The following section in the *Manage Qlik Sense sites* guide: Managing nodes and services

4 Qlik Sense upgrades and updates

You can upgrade from Qlik Sense 3.1 SR2 to Qlik Sense June 2017 using the Qlik Sense setup program. Upgrading from any version of Qlik Sense earlier than 3.1 SR2 to Qlik Sense June 2017 cannot be done using the setup program. To upgrade to Qlik Sense June 2017, see *Upgrading* (page 84).

You can update your Qlik Sense deployment by applying patches. A patch primarily includes software updates and fixes that are applied to the existing Qlik Sense version. For more information, see *Patching Qlik Sense* (page 96).

Upgrades and migrating persistence models

Qlik Sense June 2017 only supports the shared persistence model. It does not support the synchronized persistence model. When you upgrade your Qlik Sense 3.1 SR2 deployment to Qlik Sense June 2017 you will be migrated to a shared persistence model.

You can upgrade from synchronized persistence to shared persistence if the existing deployment is running Qlik Sense version 3.1 SR2. See, *Upgrading and migrating from synchronized to shared persistence* (page 90).

When upgrading a central node from synchronized persistence to shared persistence, the existing repository database is shared with all nodes. If you want to setup a dedicated repository database on a separate machine, you must perform a new installation. For more information, see *Manually installing a repository database in PostgreSQL* (page 70), and *Installing Qlik Sense in a multi-node site* (page 59).

4.1 Upgrading

You can upgrade from Qlik Sense 3.1 SR2 or later to Qlik Sense June 2017 using the Qlik Sense setup program. When upgrading, the previous version is completely replaced by the new version. To upgrade from Qlik Sense 3.1 SR2 or later with a shared persistence model to Qlik Sense June 2017, see *Upgrading from Qlik Sense 3.1 SR2 or later to Qlik Sense June 2017* (page 86).



*Do not uninstall Qlik Sense before upgrading to Qlik Sense June 2017. If you are upgrading to Qlik Sense June 2017, and you have uninstalled Qlik Sense, see *Upgrading to Qlik Sense June 2017 after uninstalling Qlik Sense 3.1 SR2 or later* (page 86).*

Qlik Sense June 2017 does not support the synchronized persistence model. To upgrade from Qlik Sense 3.1 SR2 or later to Qlik Sense June 2017 and migrate from a synchronized persistence model to a shared persistence model, see *Upgrading and migrating from synchronized to shared persistence* (page 90).

Upgrading from any version of Qlik Sense earlier than 3.1 SR2 to Qlik Sense June 2017 cannot be done using the setup program. To upgrade from earlier versions of Qlik Sense with a synchronized persistence model to Qlik Sense June 2017, see *Upgrading to Qlik Sense June 2017 from Qlik Sense versions earlier than 3.1 SR2* (page 87).

Qlik Sense apps

When you upgrade Qlik Sense all existing apps need to be migrated to ensure compatibility between the versions. This happens automatically when the system starts the first time after the upgrade. If the migration fails for one or more apps, these apps will not be available in the **Hub** after the upgrade. Apps that are not migrated are indicated in the **Apps** section of Qlik Management Console, where you can also perform a manual migration.

Multi-node deployments

In a multi-node deployment, all nodes must run the same version of Qlik Sense to be able to communicate with each other. It is recommended to upgrade with all nodes offline, and to start with the central node.



When upgrading 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.



*When upgrading a multi-node site, you must disable the QMC engine setting **Create search index during reload** (in the **Advanced** properties section), if the reload node is not the node on which the user performs searches. If you do not disable this option, an index is created when you reload, which consumes time and disk space to no advantage.*

Qlik Sense Repository Database

Qlik Sense June 2017 uses PostgreSQL version 9.6 for the Qlik Sense Repository Database. If you upgrade in place without uninstalling Qlik Sense the Qlik Sense Repository Database is upgraded to PostgreSQL version 9.6 and your data, and standard settings are carried forward. If you have made custom configurations to your PostgreSQL installation, those must be recreated in the PostgreSQL after upgrade.

The PostgreSQL installation included in the Qlik Sense June 2017 setup does not include pgAdmin tools. For information about manually installing the PostgreSQL database, see *Manually installing a repository database in PostgreSQL (page 70)*.

Before you upgrade Qlik Sense, do the following:

- Review *System requirements for Qlik Sense (page 12)*.
- Download the *Qlik_Sense_setup.exe* file.
- Make sure you have logged on as an administrator using an account that has an actual password defined, that is, not a blank password.
- Create a backup of your Qlik Sense deployment before upgrading.

Upgrading from Qlik Sense 3.1 SR2 or later to Qlik Sense June 2017

To upgrade from Qlik Sense 3.1 SR2 or later with a shared persistence model to Qlik Sense June 2017, do the following:

1. Stop your Qlik Sense services.
2. Upgrade your central node by launching the Qlik Sense setup file (Qlik_Sense_setup.exe).
3. Select **Upgrade** to upgrade your existing shared persistence deployment.
4. Accept the license agreement and click **Next**.
5. On the **Service Credentials** page, enter the **Username** and **Password** for your Windows Qlik Sense service user account.
If the user is member of a domain, enter the service account as <domain>\<username>. For more information, see *User accounts (page 49)*.
6. On the **Ready to upgrade** page, select the appropriate check boxes if you want the setup to create desktop shortcuts and automatically start the Qlik Sense services when the setup is complete, and click **Upgrade**.
7. Check that all of the Qlik Sense services have started successfully.
8. Check that all apps have been migrated successfully on the central node. If migration has failed for one or more apps, resolve the issues before continuing.
9. Deploy the Qlik Sense upgrade with shared persistence on the remaining nodes.



Any custom manual configurations that you make to the PostgreSQL database must be manually reproduced after the upgrade.

Upgrading to Qlik Sense June 2017 after uninstalling Qlik Sense 3.1 SR2 or later

If you have uninstalled Qlik Sense but maintained your PostgreSQL database, and you want to upgrade to Qlik Sense June 2017 or later, you must create a database dump file and restore the PostgreSQL database manually. You will also need to manually reconfigure any custom parameters.

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 9.3 from the [PostgreSQL](#) website. For more information, see *Manually installing a repository database in PostgreSQL (page 70)*.
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, %ProgramFiles%\PostgreSQL<database version>\bin, and run the following commands:

- a. `pg_ctl.exe start -w -D "C:\SenseDB\9.3"`
 - b. `set PGUSER=postgres`
 - c. `set PGPASSWORD=password`
 - d. `pg_dumpall.exe > [<path to dump file>]`
 - e. `pg_ctl.exe stop -w -D "C:\SenseDB\9.3"`
5. In the **Command Prompt** window, navigate to the folder containing the `Qlik_Sense_setup.exe` file.
 6. 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, using a relative path will result in an installation failure.

7. Follow the setup to complete the installation. For more information, see *Qlik Sense installation (page 55)*.

Upgrading to Qlik Sense June 2017 from Qlik Sense versions earlier than 3.1 SR2

Qlik Sense June 2017 does not support the synchronized persistence model. To upgrade to Qlik Sense June 2017 from any version of Qlik Sense earlier than 3.1 SR2 and migrate from a synchronized to shared persistence model, do the following:



You cannot upgrade from Qlik Sense versions earlier than 3.1 SR2 to Qlik Sense June 2017 using the Qlik Sense setup program. If you attempt to upgrade using the setup program you will receive an error.

1. Create a backup of your existing Qlik Sense deployment. For more information, see the help for the version of Qlik Sense that you are currently running.
2. Change the PostgreSQL authentication mode in the configuration settings to allow the password to be changed.
 - a. Stop the Qlik Sense Repository Database service.
 - b. Open the Client Authentication file located in `ProgramData\Qlik\Sense\Repository\PostgreSQL\<database version>\pg_hba.conf`.
 - c. Change the **ADDRESS** to `127.0.0.1/32`, and change the **METHOD** to `trust` for **IPv4 local connections** and local host replication.
 - d. Change the **ADDRESS** to `:::1/128`, and change the **METHOD** to `trust` for **IPv6 local connections** and local host replication.
 - e. Start the Qlik Sense Repository Database service.
3. Change the Qlik Sense Repository Database password.
To change the password using PostgreSQL command line:

4 Qlik Sense upgrades and updates

- a. Open a command prompt and navigate to
`ProgramFiles\Qlik\Sense\Repository\PostgreSQL\<database version>\bin.`
- b. Connect to the database by entering the following command:
`psql.exe -p 4432 -U postgres.`
- c. Enter the following command to set the new user password:
`ALTER USER qlikenserepository WITH PASSWORD <newpassword>.`

To change the password using the *pgAdmin* tool:

- a. Launch **pgAdmin** and connect to the Qlik Sense Repository Database.
 - b. Expand the tree in the left pane and click **Login Roles > qlikenserepository**.
 - c. Right-click on **qlikenserepository** and select **Properties**.
 - d. Click the **Definition** tab, and enter a **Password**.
4. Reset the PostgreSQL authentication mode in the configuration settings to require authentication.
 - a. Stop the Qlik Sense Repository Database service.
 - b. Open the Client Authentication file located in
`ProgramData\Qlik\Sense\Repository\PostgreSQL\<database version>\pg_hba.conf`.
 - c. Change the **METHOD** back to `md5`.
 - d. Start the Qlik Sense Repository Database service.

5. Create a database dump file.


If Qlik Sense is installed:

- a. Stop all Qlik Sense services except Qlik Sense Repository Service. Ensure the Qlik Sense Repository Service is running.
- b. Open a command prompt and navigate to the location where the PostgreSQL database is installed, and enter the following commands:
 - `set PGUSER=postgres`
 - `set PGPASSWORD=[superuserpassword]`
 - `pg_dumpall.exe -p 4432 > [path to dump file]`

If Qlik Sense has been uninstalled:

- a. Copy the PostgreSQL folder from `%ProgramData%\Qlik\Sense\Repository\PostgreSQL\9.3` to a temporary location outside of the `%ProgramData%\Qlik` folder.
- b. Download and install PostgreSQL version 9.3 from the [PostgreSQL](#) website. For more information, see *Manually installing a repository database in PostgreSQL (page 70)*.
- c. Open a Command Prompt in Microsoft Windows.
- d. Navigate to the location where the PostgreSQL repository database is installed, `cd "%ProgramFiles%\PostgreSQL\9.3\data\bin"`, and run the following commands:
 - `pg_ctl.exe start -w -D "C:\SenseDB\9.3"`
 - `set PGUSER=postgres`
 - `set PGPASSWORD=password`
 - `pg_dumpall.exe > [path to dump file]`
 - `pg_ctl.exe stop -w -D "C:\SenseDB\9.3"`

6. Make a backup of log and application data in the following folders:
 - *%ProgramData%\Qlik\Sense\Log*
 - *%ProgramData%\Qlik\Sense\Apps*
 - *%ProgramData%\Qlik\Sense\Repository\Content*
 - *%ProgramData%\Qlik\Sense\Repository\Extensions*
 - *%ProgramData%\Qlik\Sense\Repository\AppContent* (if available)
7. 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).
8. Create a file share, see *Creating a file share* (page 68).
9. Create the following sub-folders in the file share:
 - *Apps*
 - *ArchivedLogs*
 - *CustomData*
 - *StaticContent*
10. Copy following content from your synchronized persistence deployment to the file share:

Content	Copy from	To subfolder
Apps	<i>..\ProgramData\Qlik\Sense\Apps</i>	<i>Apps</i>
Logs (optional)	<i>..\ProgramData\Qlik\Sense\Repository\Archived Logs</i>	<i>ArchivedLogs</i>
Custom data	<i>..\ProgramData\Qlik\Custom Data</i>	<i>CustomData</i>
Connectors	<i>..\ProgramData\QlikTech\Custom Data</i>	<i>CustomData</i>
Static content	<i>..\ProgramData\Qlik\Sense\Repository\AppContent</i> <i>..\ProgramData\Qlik\Sense\Repository\Content</i> <i>..\ProgramData\Qlik\Sense\Repository\DefaultContent</i> <i>..\ProgramData\Qlik\Sense\Repository\Extensions</i> <i>..\ProgramData\Qlik\Sense\Repository\DefaultApps</i>	<i>StaticContent</i>
 <i>Each of these folders must be added as a sub-folder of the StaticContent folder.</i>		

11. Ensure that all Qlik Sense nodes are synchronized, and take all nodes offline by stopping the Qlik Sense services in Windows.
12. Uninstall Qlik Sense. Accept the defaults when uninstalling to preserve the certificates settings.
13. In the **Command Prompt** window, navigate to the folder containing the *Qlik_Sense_setup.exe* file.
14. 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, using a relative path will result in an installation failure.

15. Uninstall Qlik Sense on each of your rim nodes in multi-node deployment. Select the option to completely uninstall Qlik Sense when you uninstall on the rim nodes.
16. Install Qlik Sense June 2017 on each of the rim nodes.
17. Connect the rim nodes in the QMC, select each node, and click the **Redistribute** button.

4.2 Upgrading and migrating from synchronized to shared persistence

You can upgrade and migrate from synchronized persistence to shared persistence if the existing deployment is running Qlik Sense version 3.1 SR2 or later. For more information about persistence models, see *Persistence (page 41)*.

The files that are persisted in a Qlik Sense deployment must be available to all nodes via the file share. They can be stored on any of the nodes in the cluster, or on another server. If you are migrating from a synchronized persistence deployment to a shared persistence deployment, you must first create the file share to use as shared storage, and copy your data from the synchronized persistence deployment into the file share folders. For instructions on how to create a file share, see *Creating a file share (page 68)*.

Before you upgrade Qlik Sense, do the following:

- Review *System requirements for Qlik Sense (page 12)*.
- Download the *Qlik_Sense_setup.exe* file.
- Create a backup of your Qlik Sense deployment before upgrading.

Backing up a site with synchronized persistence

Proceed as follows to backup a Qlik Sense site deployed with the synchronized persistence model:

1. Make a backup of the certificates used to secure the Qlik Sense services. This only needs to be done once.

See: *Backing up certificates (page 104)*

2. Stop all Qlik Sense services except the Qlik Sense Repository Database (QRD).
3. Make a backup of the repository database.
 - a. Open a Command Prompt with administrator privileges in Microsoft Windows.
 - b. Produce a dumpfile for the repository database (that is, a single file for the entire database):
 - i. Navigate to the installation location.
`%ProgramFiles%\Qlik\Sense\Repository\PostgreSQL\<database version>\bin`
 - ii. `pg_dump.exe -h localhost -p 4432 -U postgres -b -F t -f "c:\QSR_backup.tar" QSR`

4 Qlik Sense upgrades and updates

If you are prompted for the PostgreSQL super user password, enter the password that was given during the installation of Qlik Sense.




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.

- c. Make a backup of the dumpfile for the repository database.
4. Make a backup of log and application data in the following folders:
 - `%ProgramData%\Qlik\Sense\Log`
 - `%ProgramData%\Qlik\Sense\Apps`
 - `%ProgramData%\Qlik\Sense\Repository\Content`
 - `%ProgramData%\Qlik\Sense\Repository\Extensions`
 - `%ProgramData%\Qlik\Sense\Repository\AppContent` (if available)
5. 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).
6. 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.
See: *Services (page 19)*
 - 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 order is important because the QRS is dependent on the QRD and the rest of the services are dependent on the QRS.

Upgrading and migrating from synchronized persistence to shared persistence

Do the following:

1. Create a file share, see *Creating a file share (page 68)*.
2. Create the following sub-folders in the file share:
 - `Apps`
 - `ArchivedLogs`
 - `CustomData`
 - `StaticContent`
3. Copy following content from your synchronized persistence deployment to the file share:

Content	Copy from	To subfolder
Apps	..\ProgramData\Qlik\Sense\Apps	Apps
Logs (optional)	..\ProgramData\Qlik\Sense\Repository\Archived Logs	ArchivedLogs
Custom data	..\ProgramData\Qlik\Custom Data	CustomData
Connectors	..\ProgramData\QlikTech\Custom Data	CustomData
Static content	..\ProgramData\Qlik\Sense\Repository\AppContent ..\ProgramData\Qlik\Sense\Repository\Content ..\ProgramData\Qlik\Sense\Repository\DefaultContent ..\ProgramData\Qlik\Sense\Repository\Extensions ..\ProgramData\Qlik\Sense\Repository\DefaultApps <div style="border: 1px solid gray; padding: 5px; margin-top: 10px;">  Each of these folders must be added as a sub-folder of the StaticContent folder. </div>	StaticContent

4. Ensure that all Qlik Sense nodes are synchronized, and take all nodes offline by stopping the Qlik Sense services in Windows.
5. Upgrade your central node by launching the Qlik Sense setup file (Qlik_Sense_setup.exe).
6. Select **Upgrade with Shared Persistence** to upgrade Qlik Sense and migrate to the shared persistence model.
7. Accept the license agreement and click **Next**.
8. On the **Shared persistence storage** page, enter the path or URL to your file share folders that you prepared and click **Next**.
9. On the **Database service listener** page, enter the following:
 - **Listen addresses** - add the addresses that the database service should listen to. You can enter a comma separated list of IPv4 or IPv6 addresses, or *0.0.0.0* (for all IPv4 addresses), *::* (for all IPv6 addresses) or *** (for all addresses).
 - **IP range** - add a 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*.
10. On the **Service Credentials** page, enter the **Username** , and **Password** for your WindowsQlik Sense service user account.
If the user is member of a domain, enter the service account as *<domain>\<username>*. See: *User accounts (page 49)*.
11. On the **Repository Database Superuser Password** page, enter the password for your repository database superuser. See: *User accounts (page 49)*.

12. On the **Ready to upgrade** page, select the appropriate check boxes if you want the setup to create desktop shortcuts and automatically start the Qlik Sense services when the setup is complete, and click **Upgrade**.
13. Check that all of the Qlik Sense services have started successfully.
14. Check that all apps have been migrated successfully on the central node. If migration has failed for one or more apps, resolve the issues before continuing.
15. Install Qlik Sense with shared persistence on the remaining nodes, and join the existing cluster created when you upgraded the central node.

4.3 Performing a silent upgrade

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.



Note that elevation will take place if run from an unelevated process and the UAC is on.

Syntax

```
Qlik_Sense_setup.exe [-silent] {-log "path\filename"}  
{desktopshortcut=1|0} {skipstartservices=1|0} {installdir="path"}  
{userpassword="password"} {dbpassword="password"}
```

```
Qlik_Sense_setup.exe -? or -h
```

Brings up the on-screen silent setup help.

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.

4 Qlik Sense upgrades and updates

The default values are the same as those used in the setup user interface.

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 (<i>%ProgramFiles%\Qlik\Sense</i>).
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.

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
```

Deprecated command line arguments

For a list of the command line arguments that are no longer recommended, see [Installing silently](#).

4.4 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.

4.5 Performing a silent repair

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

```
Qlik_Sense_setup.exe [-silent] [-repair] {-log "path\filename"}
```

```
Qlik_Sense_setup.exe -? or -h
```

Brings up the on-screen silent setup help.

Commands

-silent (or -s) Trigger the silent mode (mandatory).

- repair** Repair the product silently.
- log (or -l)** Log file directory and log file name.



The user must have access to this directory.

If this option is not defined, the log file will be stored with the default name in the default location.

Example:

This example shows how to silently repair the Qlik Sense installation.

```
Qlik_Sense_setup.exe -s -repair
```

4.6 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.

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 (page 12)*.
- Download the *Qlik_Sense_setup.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 site (page 99)*.



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.
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.*


4.7 Uninstalling Qlik Sense

 *If any updates have been applied to Qlik Sense since installation, the Uninstall option will also remove all the updates.*

Do the following:

1. To start uninstalling, open the **Control Panel** and select **Uninstall a program**. Then select **Qlik Sense** from the list of programs and click **Uninstall**.

A confirmation screen is displayed asking if you are sure that you want to uninstall Qlik Sense from your computer. Select the **Remove Qlik Sense certificates and data folders** checkbox if you want to remove all files from the machine ready for a new configuration.

 *If the machine is a central node in a Qlik Sense site, there may be rim nodes on other machines that require access to the central node to function properly.*



You can also uninstall Qlik Sense by double-clicking the Qlik_Sense_setup.exe file and then selecting Uninstall from the Maintenance screen. In that case, you must use the correct version of the setup file when uninstalling your Qlik Sense installation, that is, the same version used when installing Qlik Sense.

2. Click **Uninstall** to start uninstalling Qlik Sense.
If User Account Control (UAC) is disabled, the uninstall starts.
If UAC is enabled, the **User Account Control** dialog is displayed.
Click **Yes** to start the uninstall.

The progress of the uninstall process is displayed. When finished the uninstall dialog confirms that Qlik Sense has been uninstalled successfully.

3. Click **Finish**.

You have now uninstalled Qlik Sense.

5 Backing up and restoring

This section describes how to back up and restore Qlik Sense sites and certificates and how to move a node to a new machine.

5.1 Backing up and restoring a site

This section describes how to backup and restore a Qlik Sense site.

- See *Backing up a site (page 99)* and *Restoring a site (page 102)*



These instructions define the minimum steps required. The use of specific backup software may further extend the options for backup and restore.

In a single node site, the single node is referred to as the central node.

In a multi-node site, the central node is the master record that contains all data about the site. The rim nodes in the multi-node site contain either a full copy or a limited subset of the data, which is maintained by the synchronization mechanism. This means that the central node is the only node that needs to be backed up in order to keep the data and configuration safe. The rim nodes can be restored by simply re-adding them as new nodes after a clean installation, since they will have their data restored by the synchronization mechanism.

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.

See also:

 [Architecture \(page 16\)](#)

Backing up a site

This section describes how to backup a Qlik Sense site in a default installation where a PostgreSQL database is used as the repository database.

PostgreSQL version 9.6 is installed with the latest version of Qlik Sense. If you have uninstalled Qlik Sense but maintained your PostgreSQL database, and you want to upgrade your Qlik Sense deployment, you must create a database dump file and restore the PostgreSQL database manually. You will also need to manually reconfigure any custom parameters.



Create the PostgreSQL database dump file before you uninstall Qlik Sense.



The instructions in this section are only applicable to the central node of a Qlik Sense site.

The following items need to be considered when backing up a site:

- Repository database: The database contains all configuration data for the site.
- 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)

See: *Backing up and restoring certificates (page 104)*

Backing up a site with shared persistence

Proceed as follows to backup a Qlik Sense site with shared persistence:

1. Make a backup of the certificates used to secure the Qlik Sense services. This only needs to be done once.

See: *Backing up certificates (page 104)*

2. On all nodes of the site, stop all Qlik Sense services except the Qlik Sense Repository Database (QRD).
3. Make a backup of the repository database.



In a shared persistence deployment with the database on a dedicated machine, this needs to be performed on the database server, using the installation location that you selected.

- a. Open a Command Prompt with administrator privileges in Microsoft Windows.
- b. Produce a dumpfile for the repository database (that is, a single file for the entire database):
 - i. Navigate to the installation location.

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 your deployment includes a dedicated database server that was manually installed with the default PostgreSQL install path, the location will be:
`%ProgramFiles%\PostgreSQL\<database version>\bin`
 - ii. `pg_dump.exe -h localhost -p 4432 -U postgres -b -F t -f "c:\QSR_backup.tar" QSR`

If you are prompted for the PostgreSQL super user password, enter the password that was given during the installation of Qlik Sense.



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.

- c. Make a backup of the dumpfile for the repository database.
4. Make a backup of log and application data in the file share used for storage of log and application data.
5. 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).
 - 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.
See: *Services (page 19)*

- 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 order

The order is important because the QRS is dependent on the QRD and the rest of the services are dependent on the QRS.

Creating a PostgreSQL database dump file 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 9.6 from the [PostgreSQL](#) website. See: *Manually installing a repository database in PostgreSQL (page 70)*.
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, `%ProgramFiles%\PostgreSQL\<database version>\bin`, and run the following commands
 - a. `pg_ctl.exe start -w -D "C:\SenseDB\<database version>"`
 - b. `set PGUSER=postgres`
 - c. `set PGPASSWORD=password`
 - d. `pg_dumpall.exe > [<path to dump file>]`
 - e. `pg_ctl.exe stop -w -D "C:\SenseDB\<database version>"`

Restoring a site

This section describes how to restore a Qlik Sense site in a default installation where a PostgreSQL database is used as the repository database.

PostgreSQL version 9.6 is installed with the latest version of Qlik Sense. To migrate from an earlier version of PostgreSQL you must create a dump file. You will also need to manually configure any custom parameters.



The instructions in this section are only applicable to the central node of a Qlik Sense site.

The following items need to be considered when restoring a site:

- Qlik Sense software
- If you want to restore the site to a central node with a new hostname there are additional considerations.
See: Restoring a central node to a machine with a different hostname (page 103)
- Repository database: The database contains all configuration data for the site.
- 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)

Restoring a site with shared persistence

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.

Proceed as follows to restore a Qlik Sense site:

1. 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. If the services are started, new certificates and a new repository database are created and they must be removed before proceeding with the restore procedure.*

2. Restore the certificates used to secure the Qlik Sense services.

See: Restoring certificates (page 113)



Do not start the Qlik Sense services at the end of the Restoring certificates (page 113) procedure.

3. Start the Qlik Sense Repository Database (QRD).
4. Restore the repository database:
 - a. Place the backed up repository database on the machine targeted for the restore.
 - b. Open a Command Prompt with administrator privileges in Microsoft Windows.
 - c. Run the following commands to restore the repository database (adjust the paths as needed):
 - i. `cd "%ProgramFiles%\Qlik\Sense\Repository\PostgreSQL<database version>\bin"`
 - ii. `createdb -h localhost -p 4432 -U postgres -T template0 QSR`
If the command fails because a database already exists, run the following command and then repeat the `createdb` command:
`dropdb -h localhost -p 4432 -U postgres QSR`
 - iii. `pg_restore.exe -h localhost -p 4432 -U postgres -d QSR "c:\QSR_backup.tar"`
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 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.
See: *Services (page 19)*
 - 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 order is important because the QRS is dependent on the QRD and the rest of the services are dependent on the QRS.

Restoring a central node to a machine with a different hostname

A restore can occur onto a machine with a different name than the one from which the data was backed up. However, if the machine is a central node in a multi-node site, you need to adapt the procedure.

- All rim nodes need to be reset, that is, you need to remove them and then add them again.
- Before starting the database, you need to update the hostname in the repository by executing the following command in PostgreSQL:

```
UPDATE "LocalConfigs"  
SET "HostName" = '@newHostName'  
WHERE "HostName" = '@oldHostName';
```

replacing `@newHostName` and `@oldHostName` with the actual new and old hostnames.
- Do not import certificates from the old system, as they will be invalid on the new node. You need to authorize certificates on the new node.
You also need to update password information for all data connectors, as the passwords were encrypted using the certificate on the old system.

Known issues

If Qlik Sense stops responding or does not respond properly after the restore operation, try restarting the Qlik Sense Repository Service (QRS) and then wait for the rest of the services to start up.

See also:

 [Backing up and restoring certificates \(page 104\)](#)

5.2 Backing up and restoring certificates

It is recommended that you back up the certificates on the central node in a Qlik Sense site so that they can be restored, if needed.

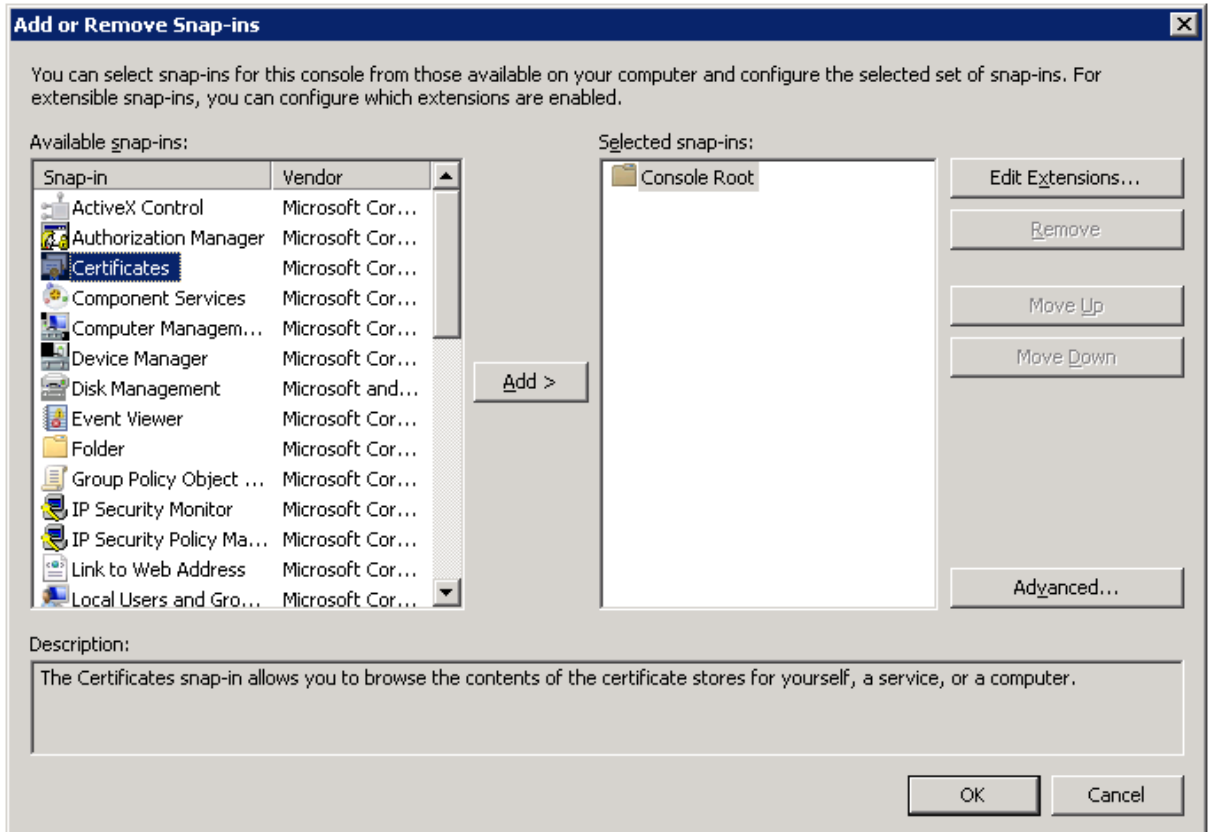
The backed up certificates can be used for different purposes:

- Restore the certificates on the **same** node as they were exported from.
- Move a node to **another** node in the site. This means that the repository database and its associated crypto key are reused on another node, but with new certificates for communication.

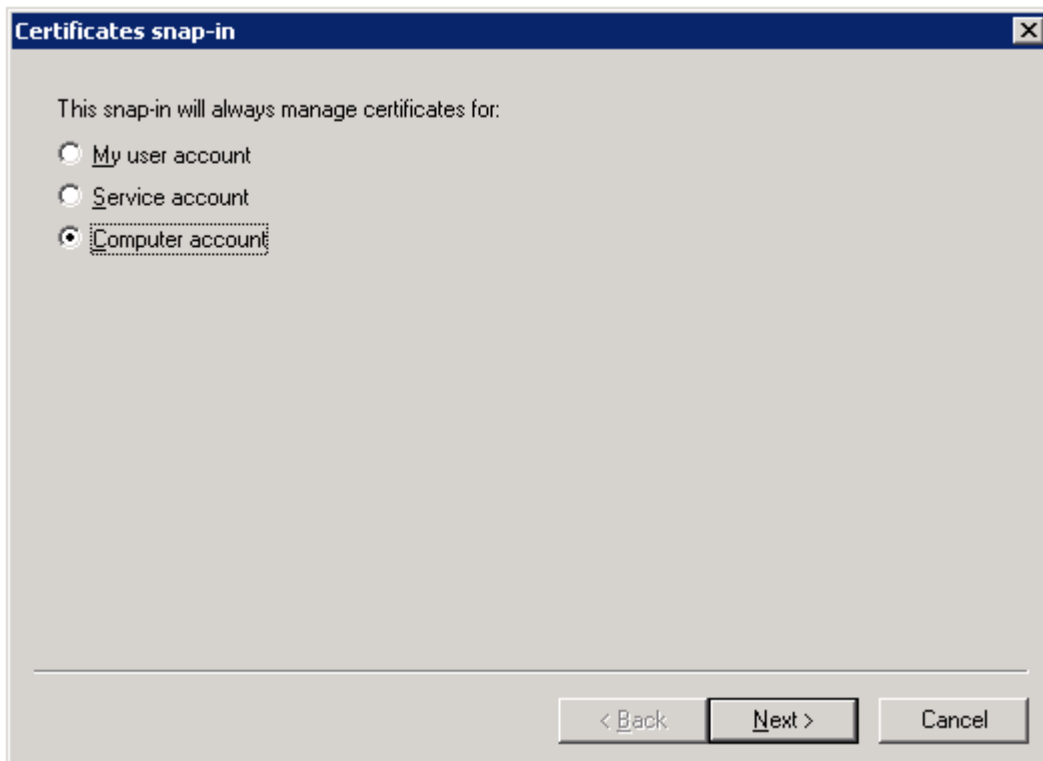
Backing up certificates

Proceed as follows to make a backup of the certificates on the central node in a Qlik Sense site:

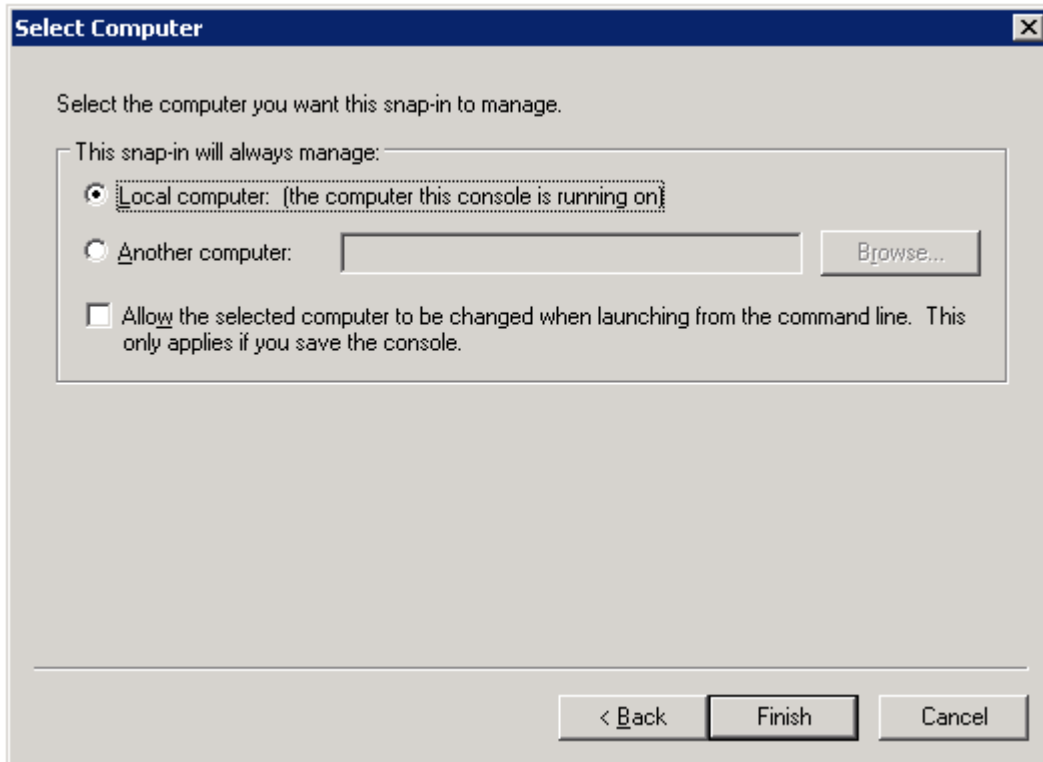
1. Run *mmc* as the user that is used to run Qlik Sense services.
2. Select **File>Add/Remove Snap-in**.
3. Double-click **Certificates**.



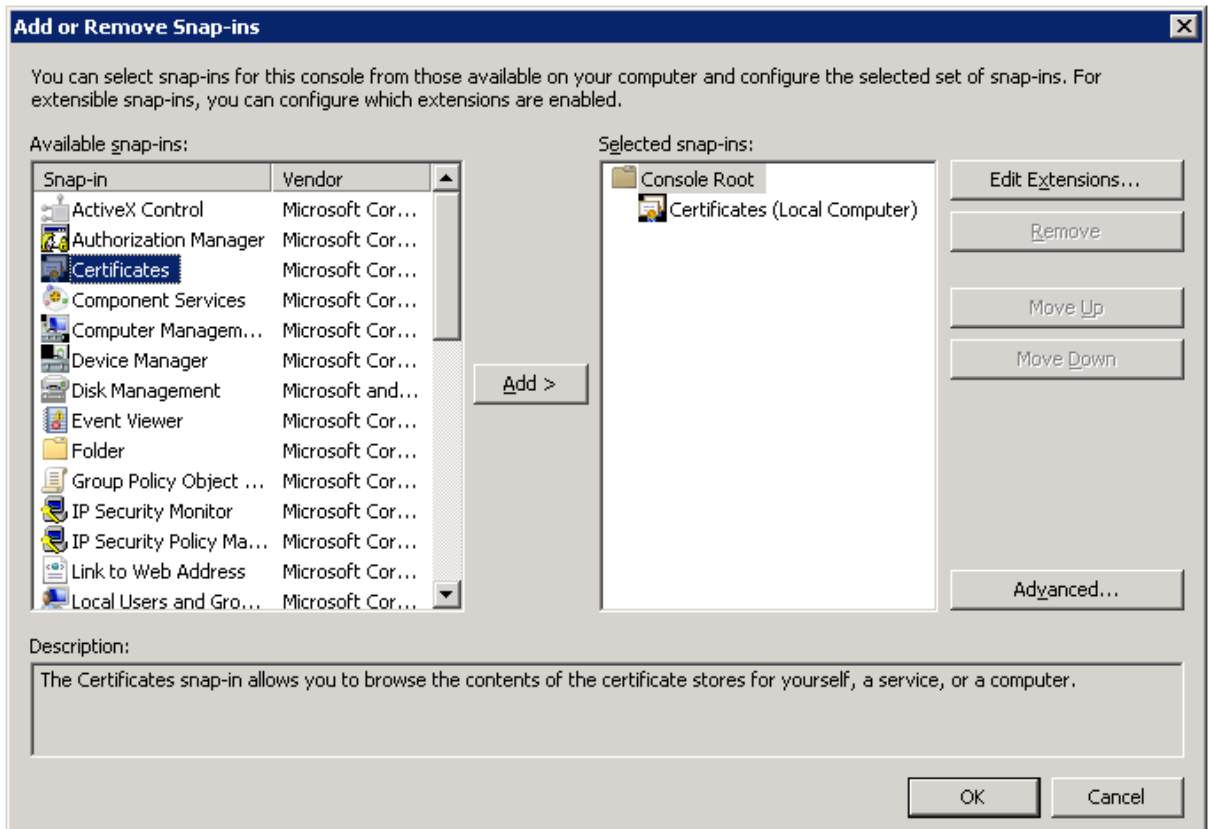
4. Select **Computer account** and click **Next**.



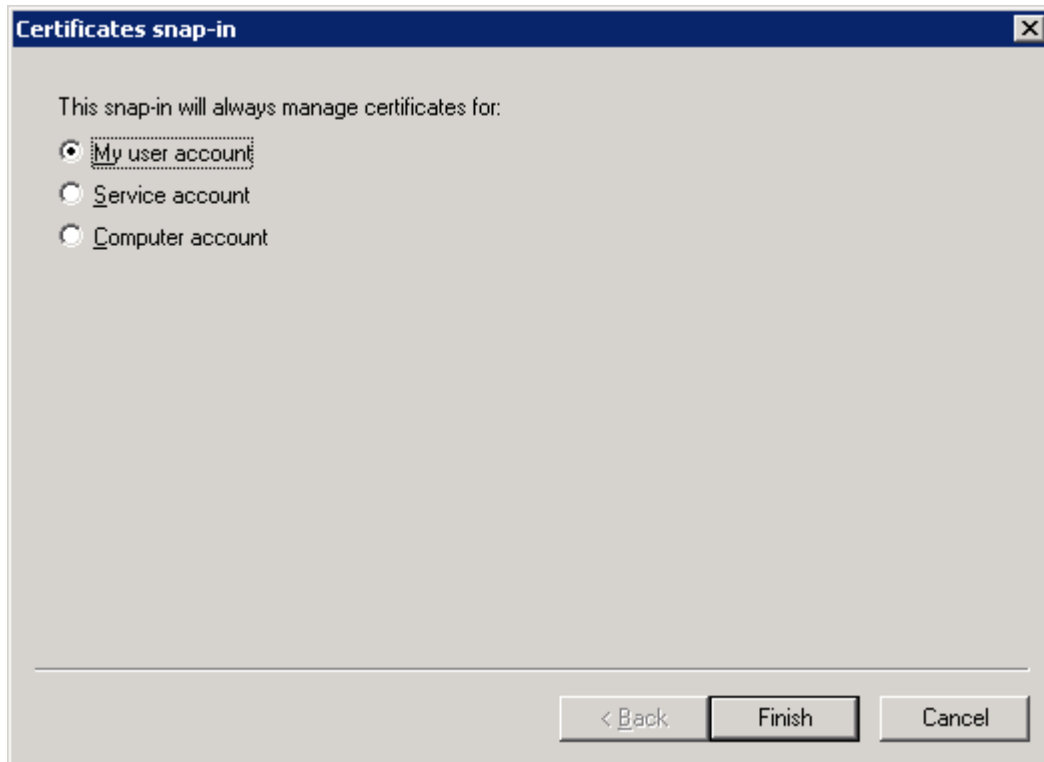
5. Select **Local computer** and click **Finish**.



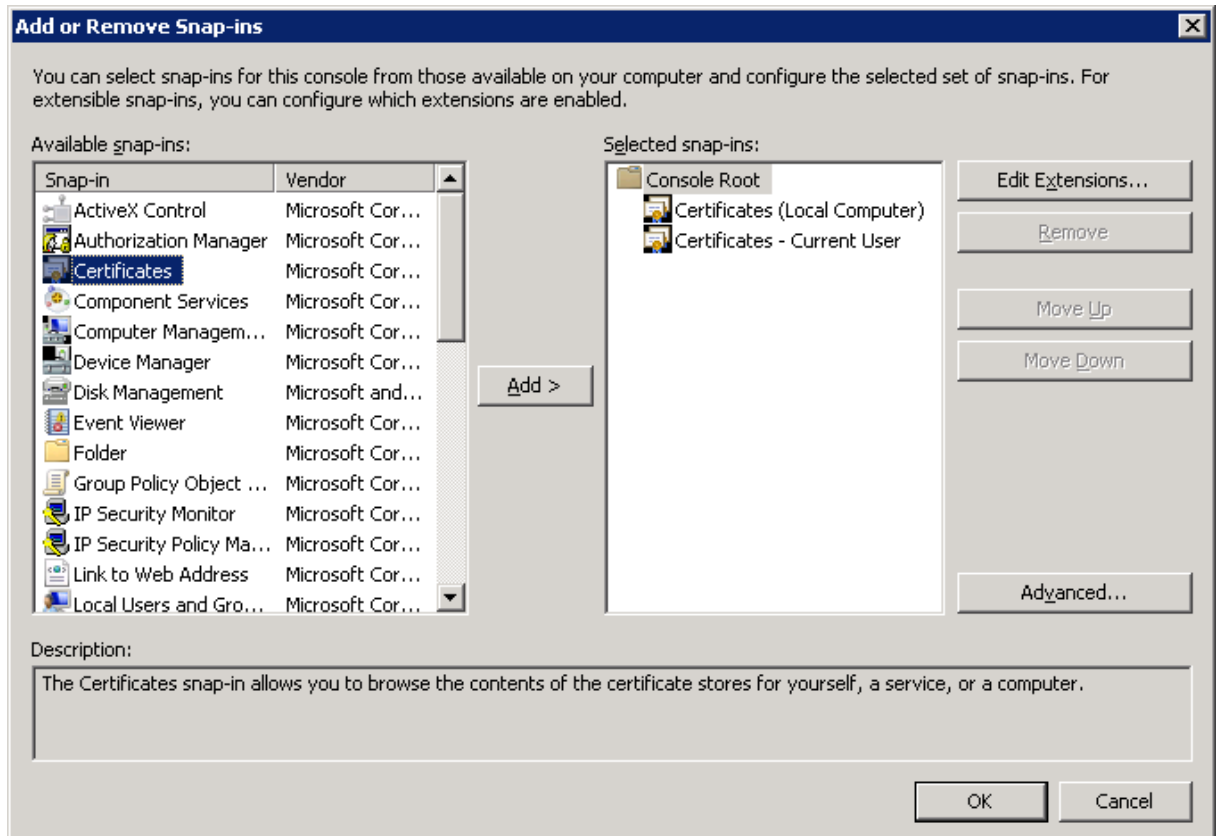
6. Double-click **Certificates**.



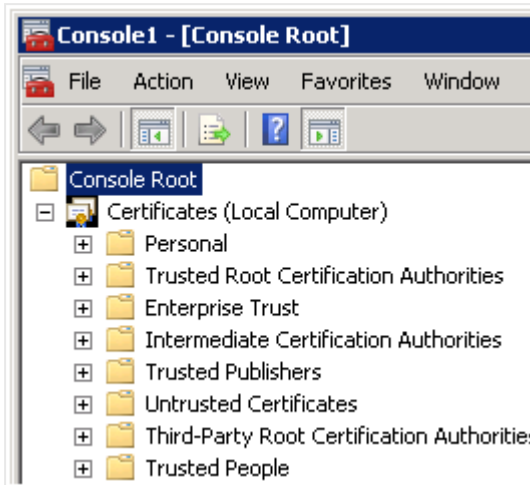
7. Select **My user account** and click **Finish**.



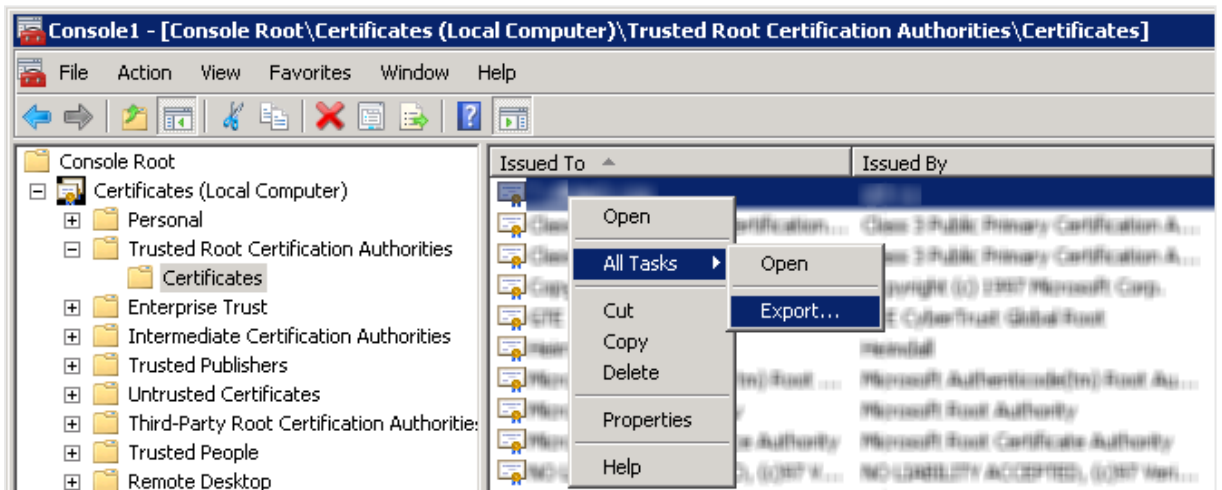
8. Click **OK**.



- Expand **Certificates (Local Computer)** in the left panel.



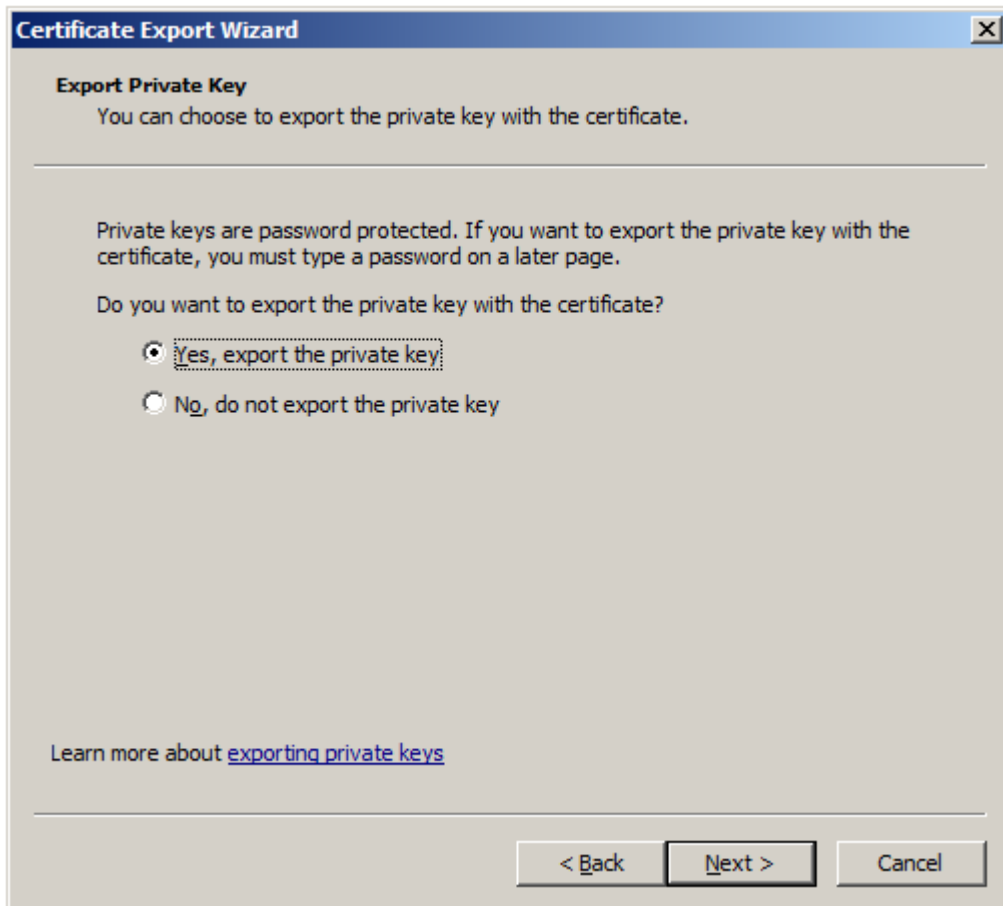
- Expand the **Trusted Root Certification Authorities** folder and select the **Certificates** folder.
- Right-click the certificate that is Certificate Authority (CA) for all nodes in the Qlik Sense site and select **All Tasks>Export**. The CA is named *<machine_that_issued_the_certificate>-CA* by default.



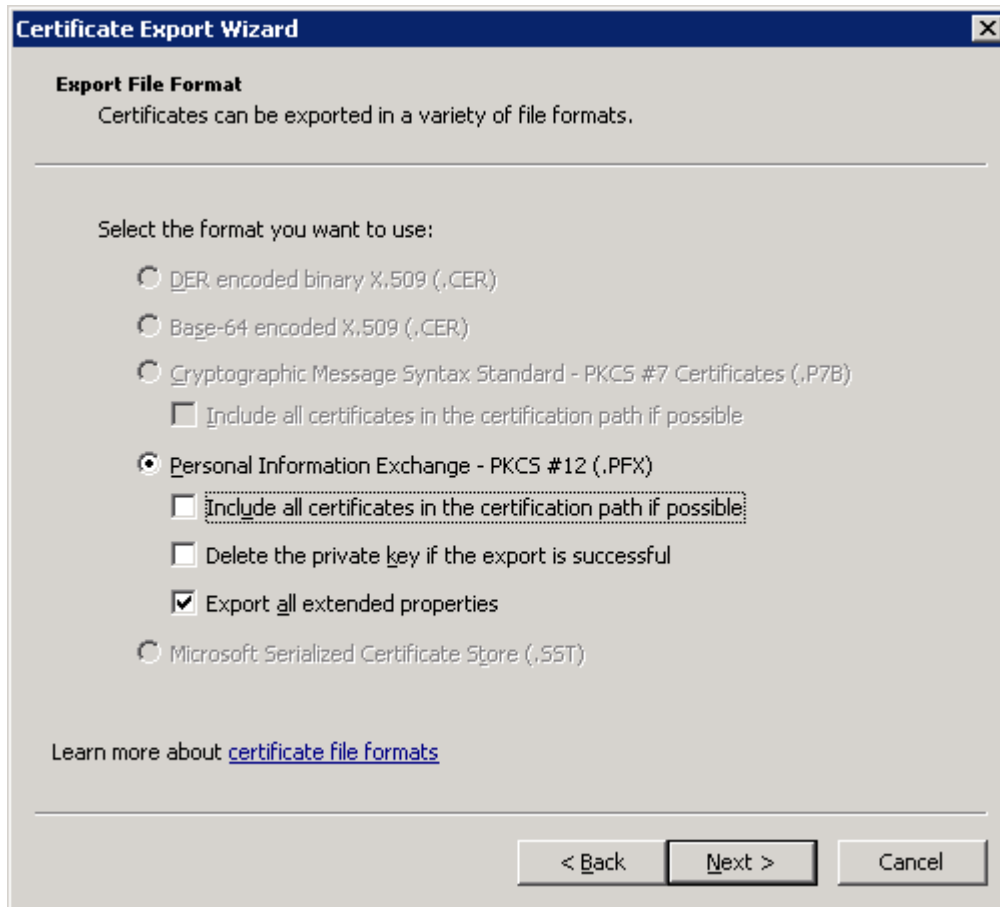
- Click **Next**.



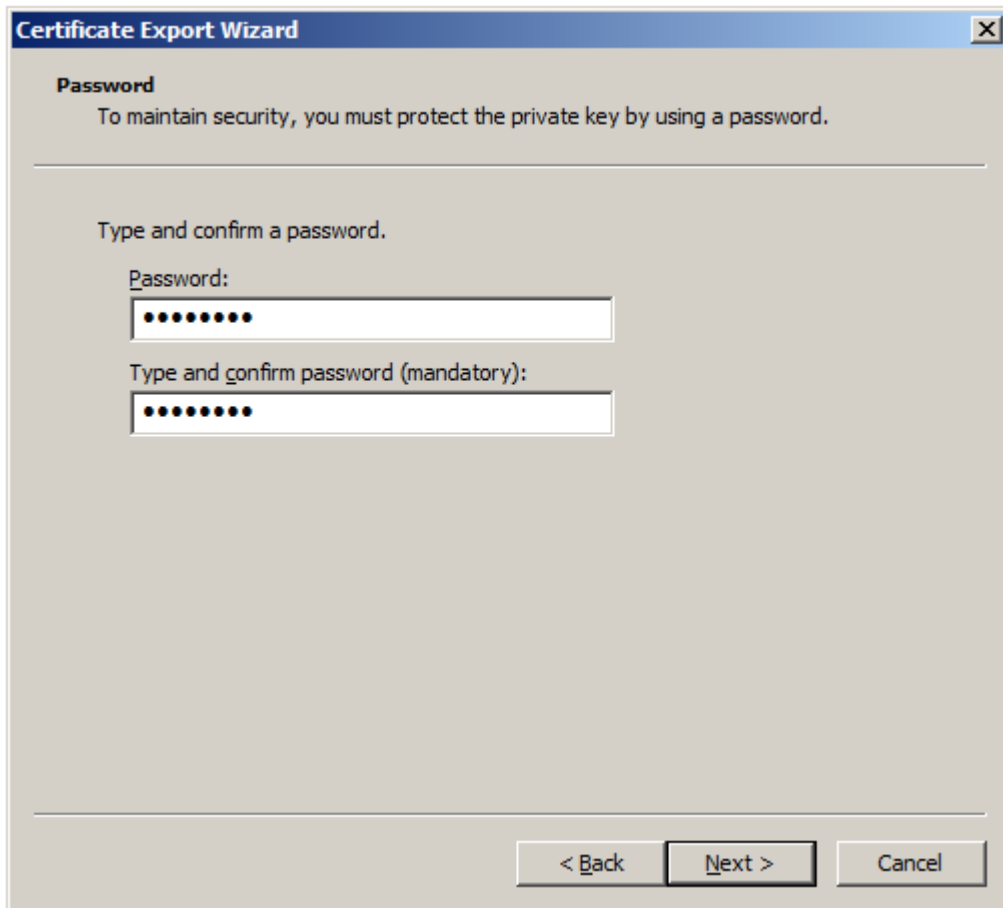
13. Select **Yes, export the private key** and click **Next**.



14. Select **Personal Information Exchange**.
15. Tick the **Export all extended properties** box and then click **Next**.



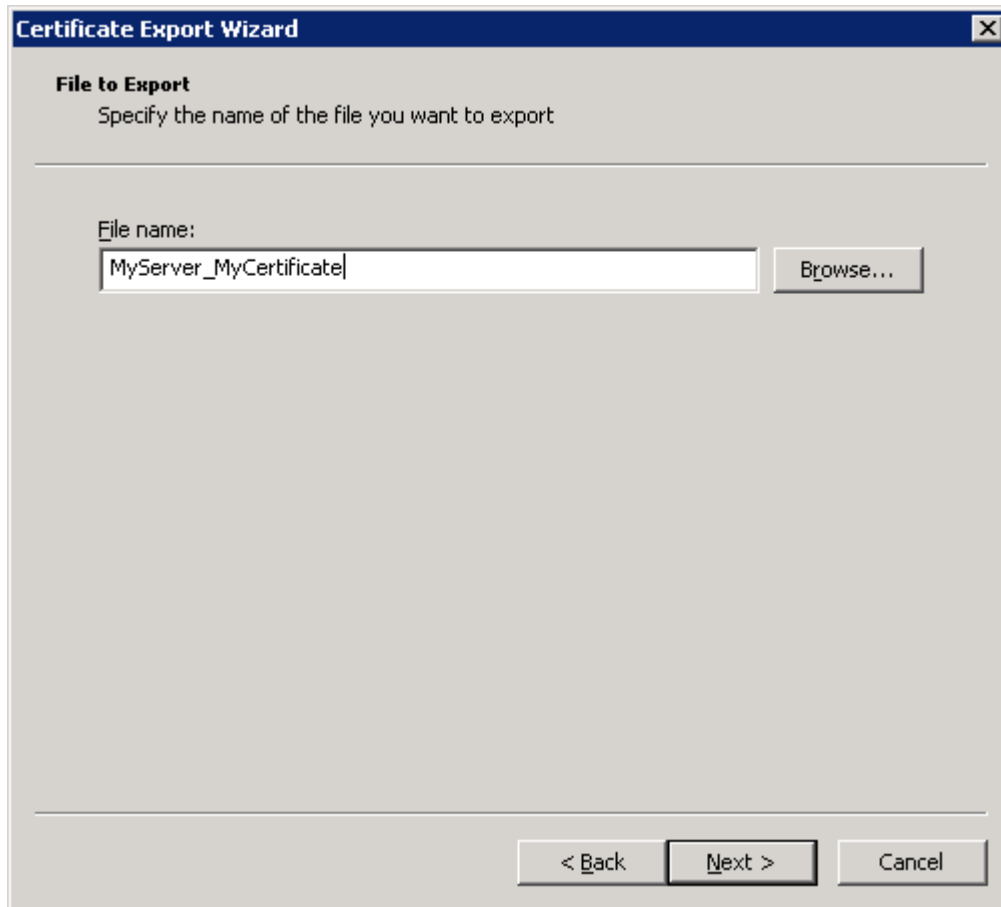
16. Enter and confirm a password. Then click **Next**.
The password is needed when importing the certificate.



17. Enter a file name for the .pfx file and click **Next**.



It is recommended to include the server name in the file name to avoid confusion with other certificate files.



18. Click **Finish**.
The *.pfx* file that contains the CA for all nodes in the Qlik Sense site is stored in the selected location.
19. Starting at step 11, repeat the procedure and export the server certificate (that is, the SSL certificate), which is located under **Certificates (Local Computer)>Personal>Certificates**. The server certificate a) has the same name as the Domain Name System (DNS) name of the machine, and b) is signed by the CA for all nodes in the site.
20. Starting at step 11, repeat the procedure and export the client certificate (that is, the ID of the client), which is located under **Certificates - Current User>Personal>Certificates**. The client certificate is named *QlikClient* and is signed by the CA for all nodes in the site.
21. Close the MMC console.
No changes have to be saved.

Restoring certificates

In case of a system crash, the certificates may have to be restored on the central node in the Qlik Sense site.

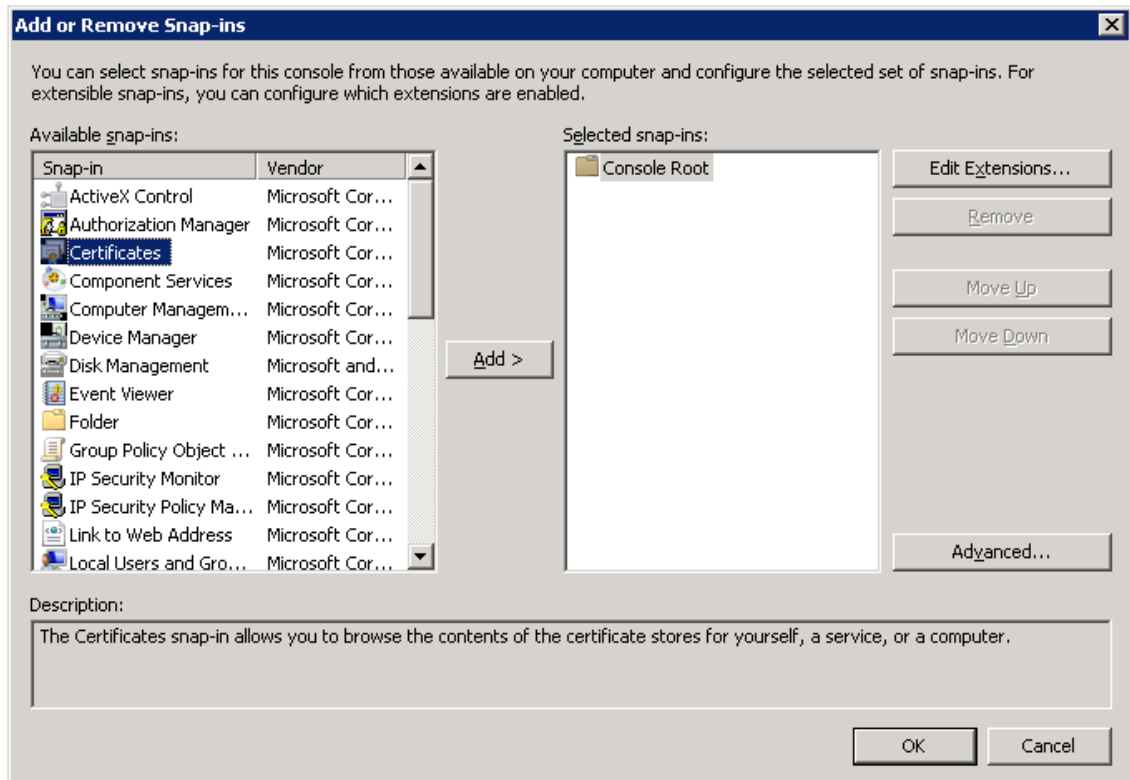
Proceed as follows to restore the certificates on the central node in a site:

1. Open the Task Manager in Microsoft Windows and stop all Qlik Sense services except the Qlik Sense Repository Database (QRD) service.

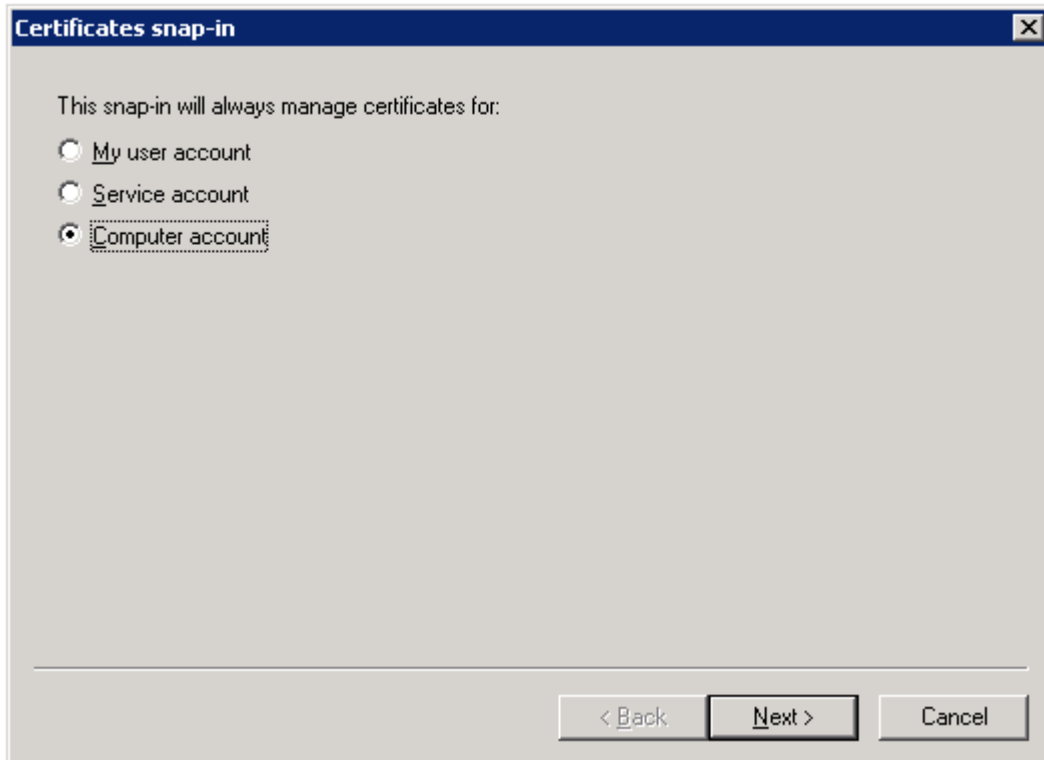


If you are restoring the certificates as part of the Restoring a site (page 102) procedure, skip this step.

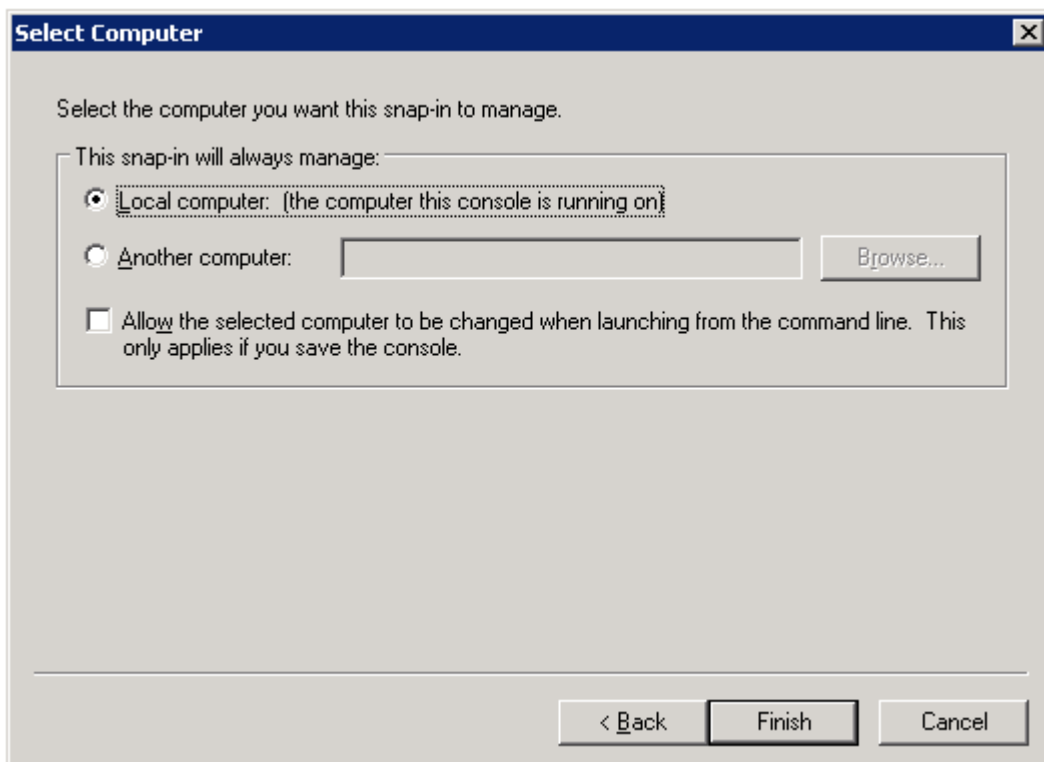
2. Run *mmc* as the user that is used to run Qlik Sense services.
3. Select **File>Add/Remove Snap-in**.
4. Double-click **Certificates**.



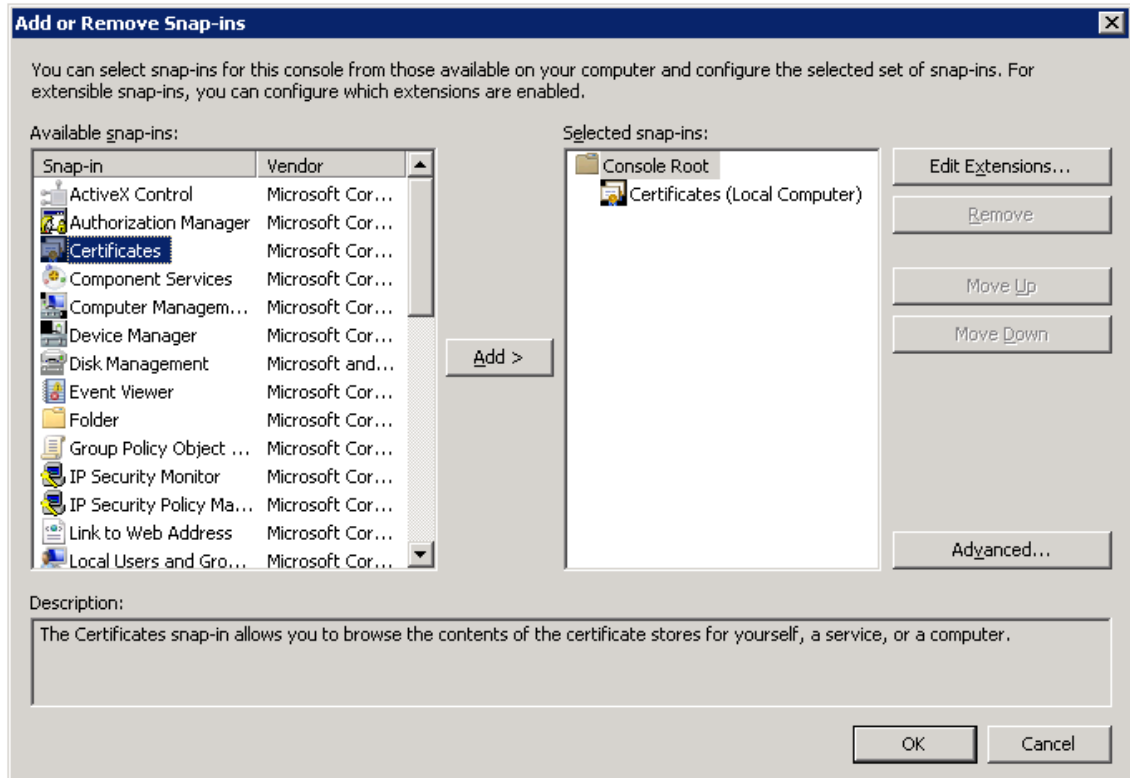
5. Select **Computer account** and click **Next**.



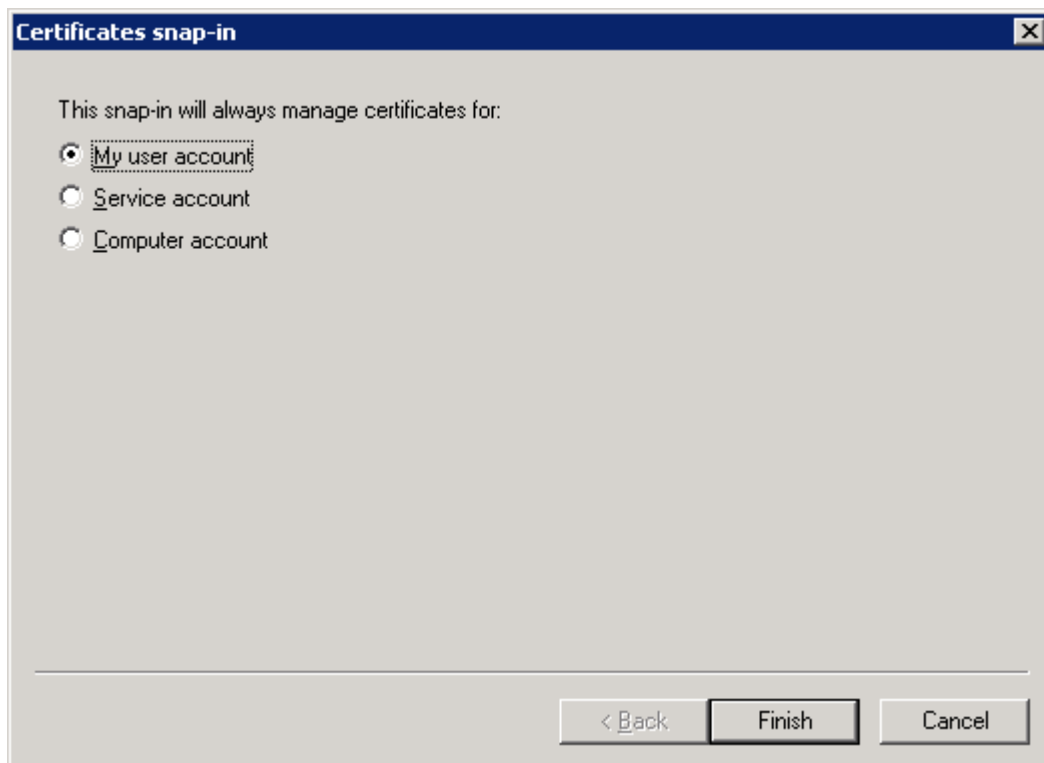
6. Select **Local computer** and click **Finish**.



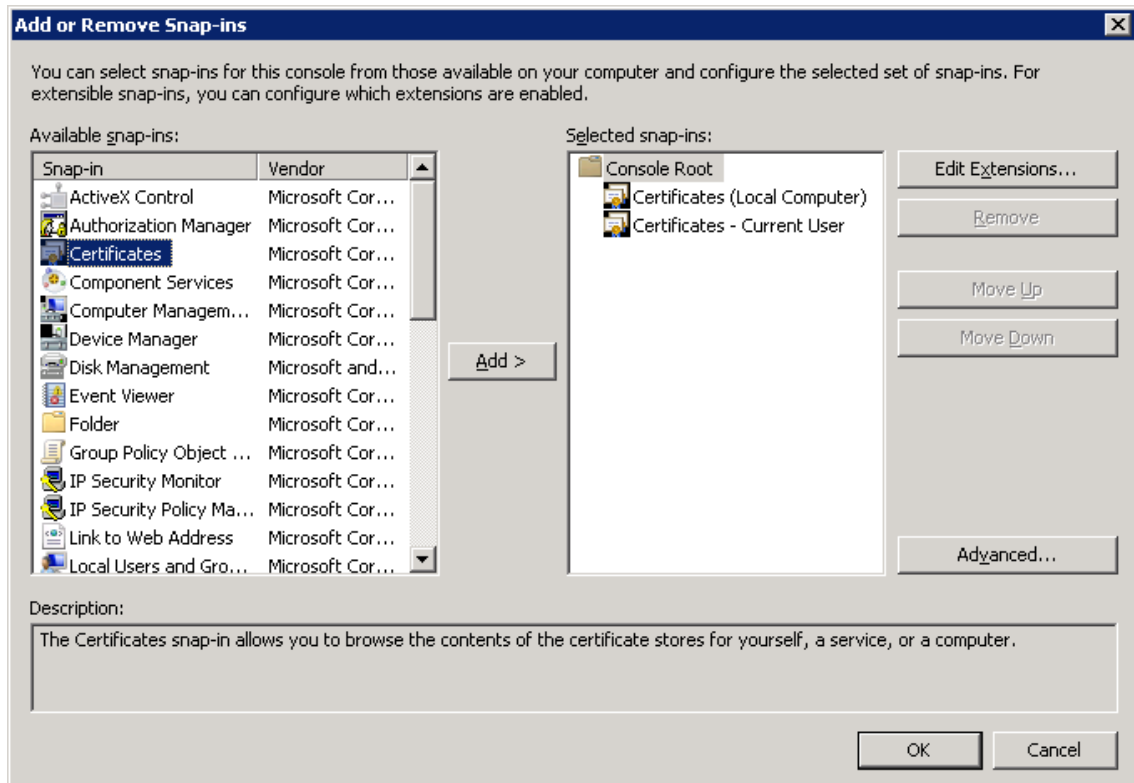
7. Double-click **Certificates**.



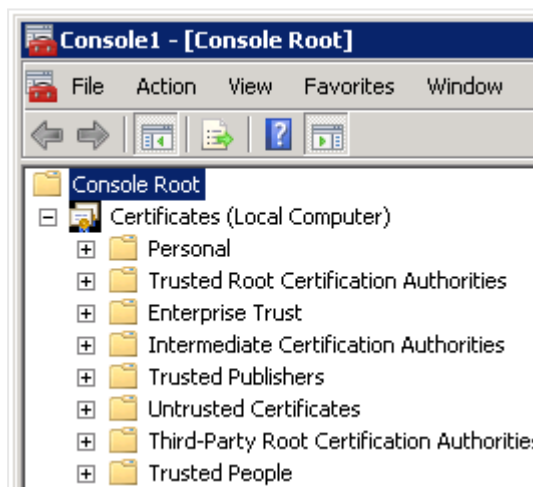
8. Select **My user account** and click **Finish**.



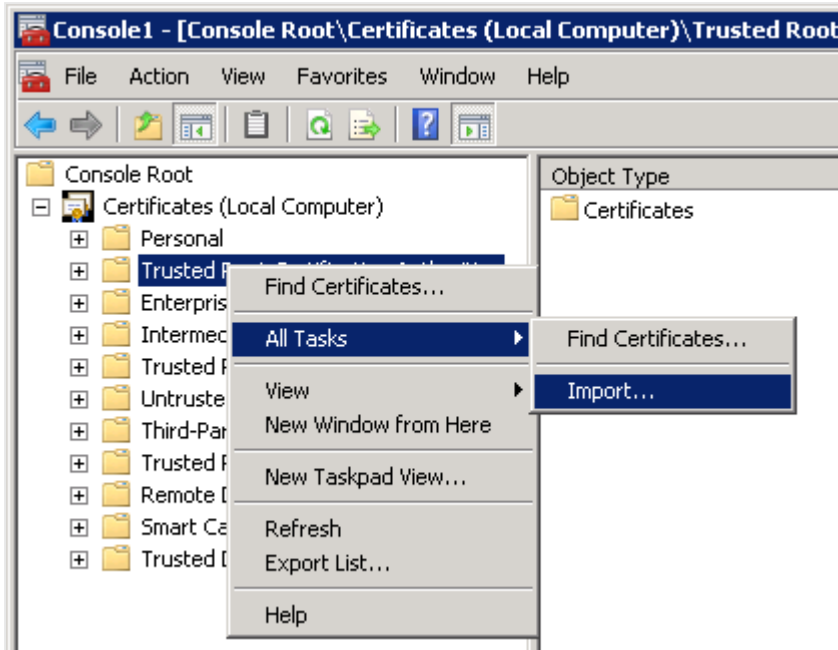
9. Click **OK**.



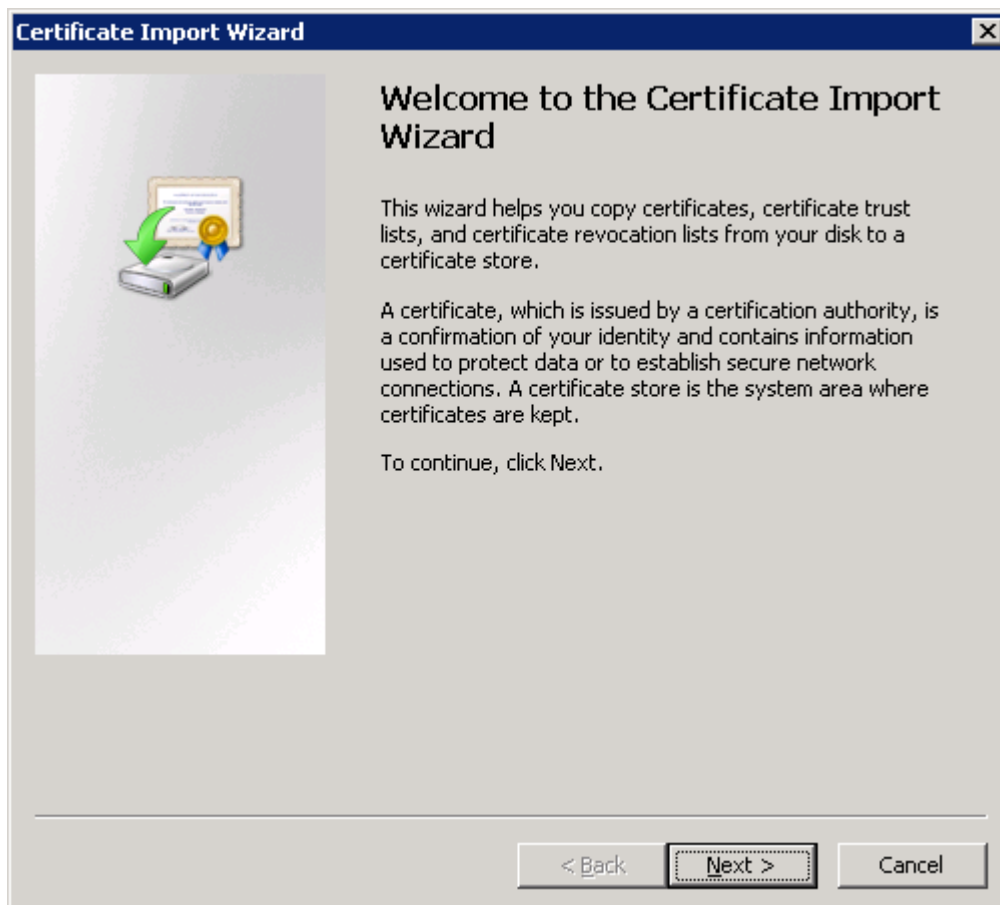
- Expand **Certificates (Local Computer)** in the left panel.



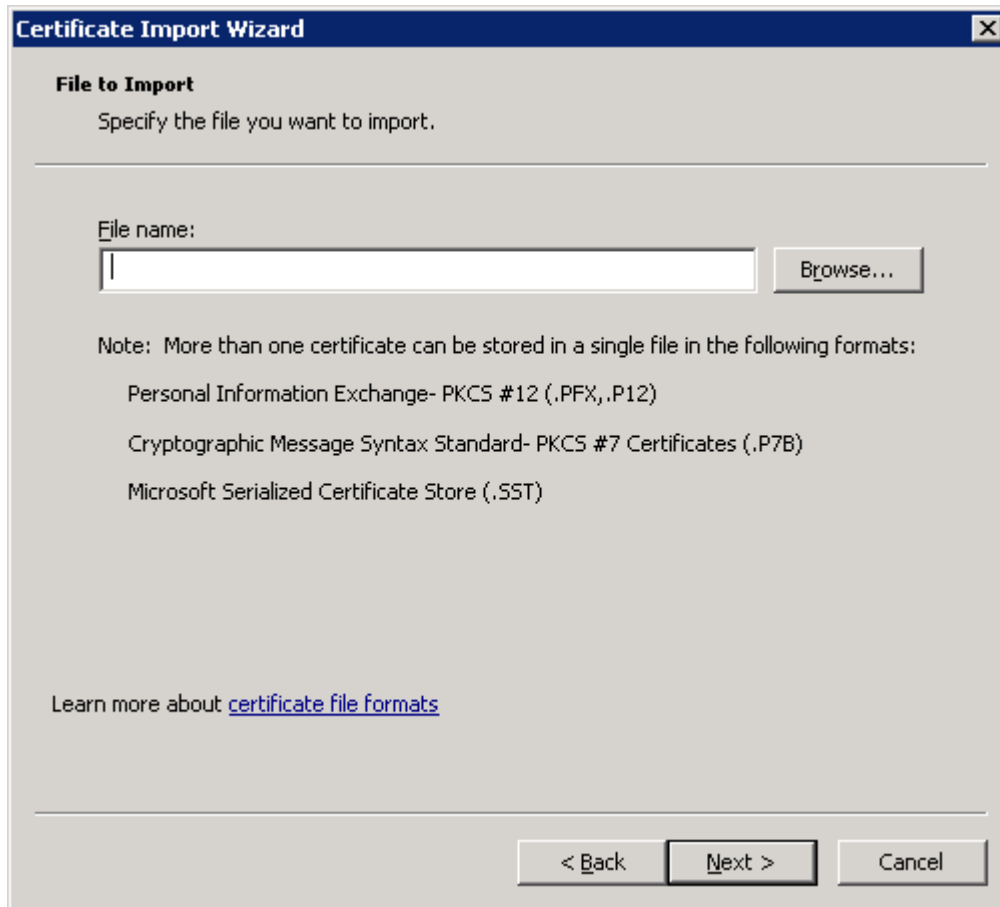
- Right-click the **Trusted Root Certification Authorities** folder and select **All Tasks>Import**.



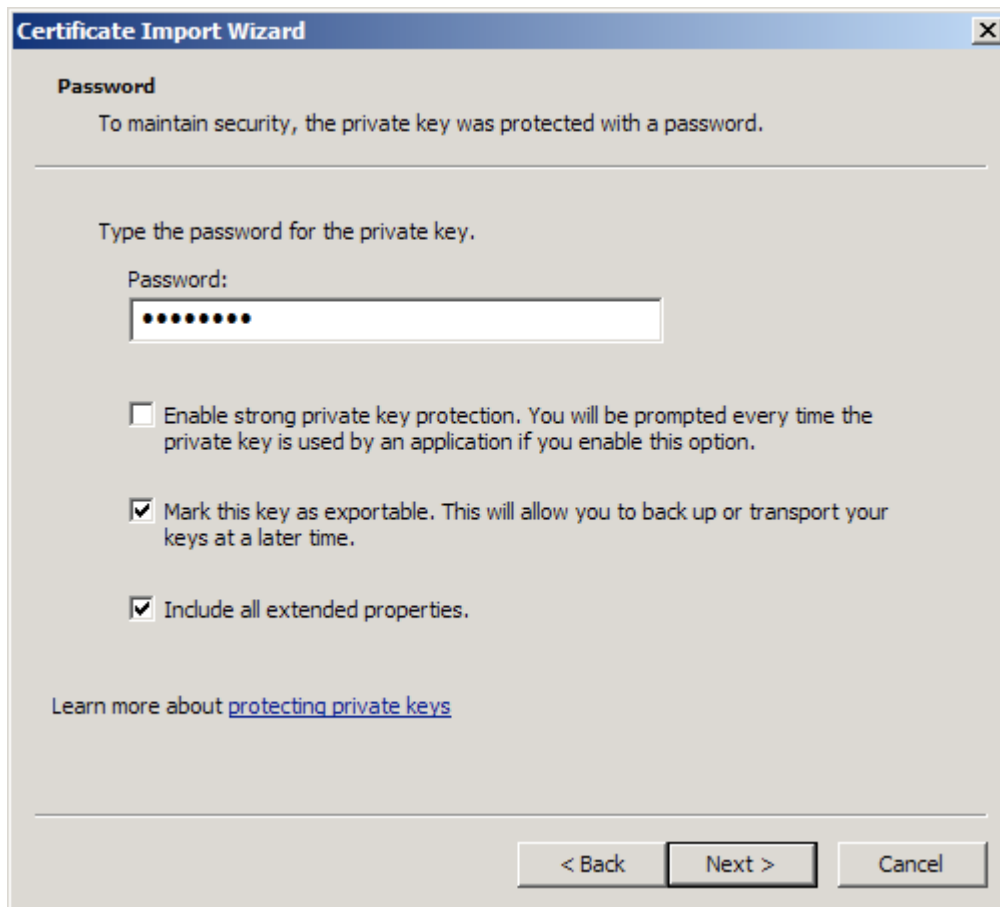
12. Click **Next**.



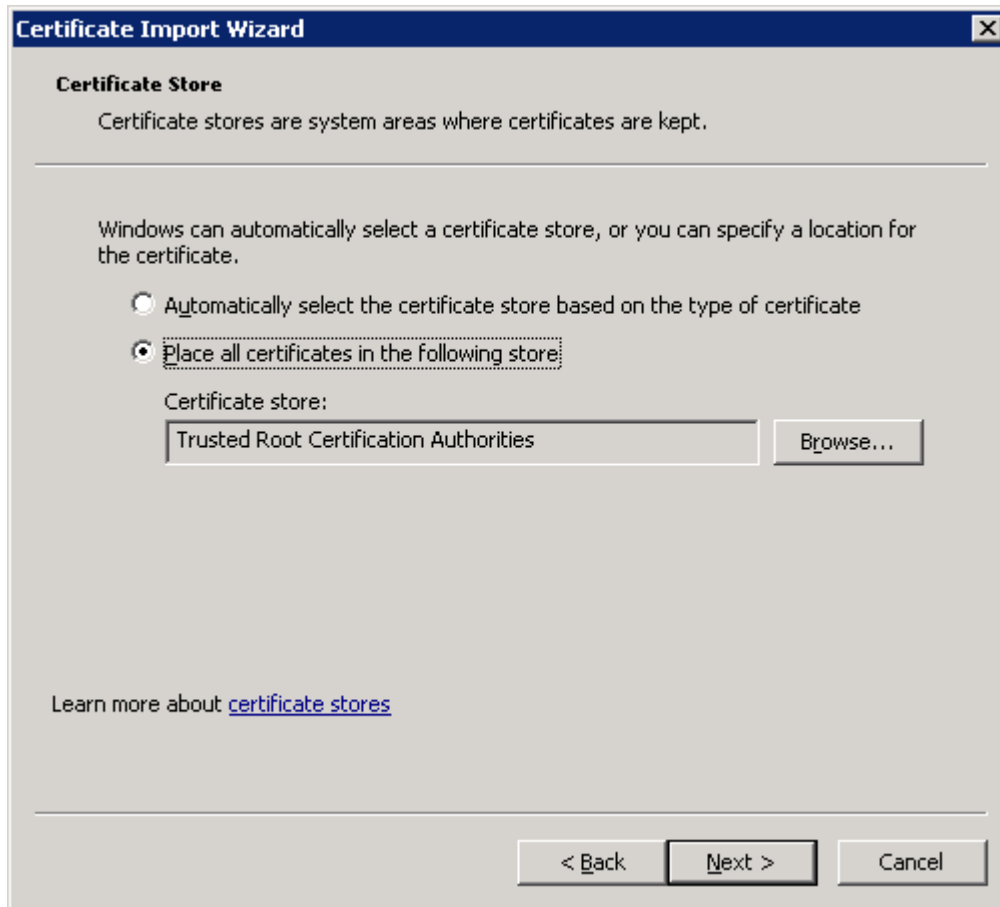
13. Browse to the file that contains the backed up Certificate Authority (CA) for all nodes in the site and then click **Next**. The CA is named *<machine_that_issued_the_certificate>-CA* by default.



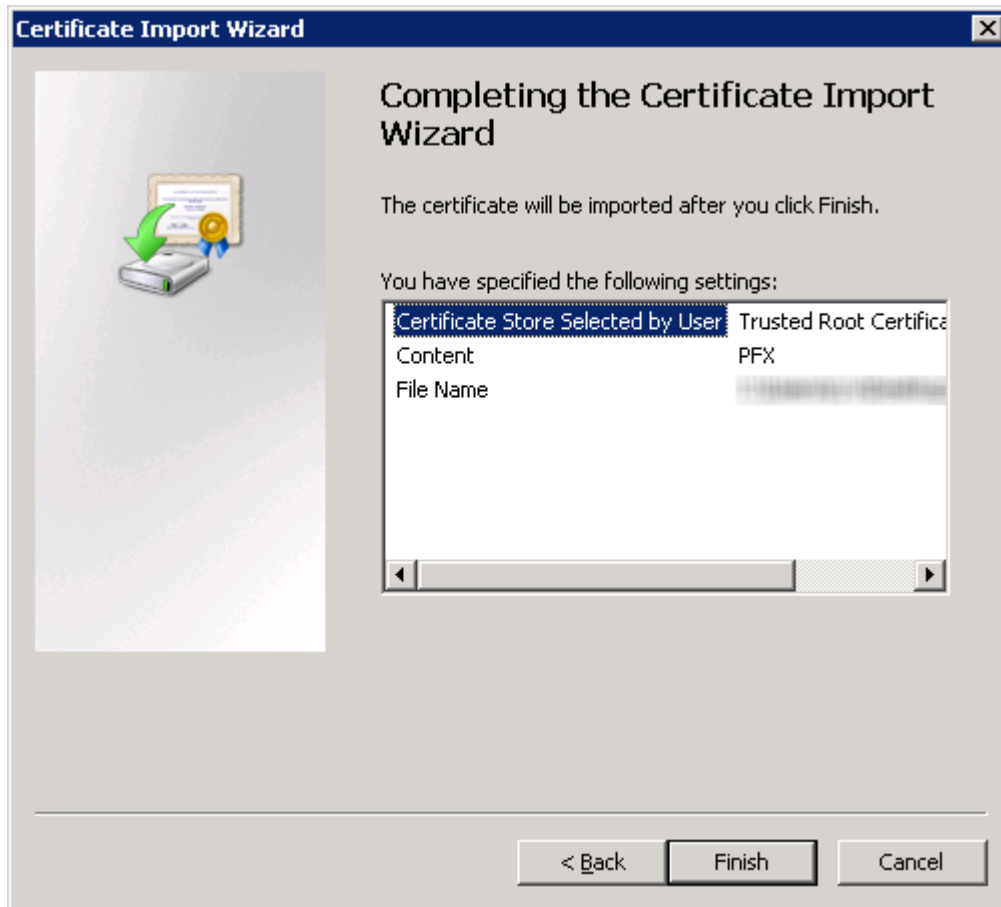
14. Enter the password for the *.pfx* file (that is, the password that was given when the file was exported).
15. Select **Mark this key as exportable** and **Include all extended properties**. Then click **Next**.



16. Select **Place all certificates in the following store** and click **Next**.



17. Click **Finish**.



18. Click the **Refresh** button (🔄) and check that the imported CA is available in the **Trusted Root Certification Authorities** folder.
19. Starting at step 11, repeat the procedure and import the server certificate to **Certificates (Local Computer)>Personal>Certificates**. The server certificate a) has the same name as the Domain Name System (DNS) name of the machine, and b) is signed by the CA for all nodes in the site.
20. Starting at step 11, repeat the procedure and import the client certificate (that is, the ID of the client) to **Certificates - Current User>Personal>Certificates**. The client certificate is named *QlikClient* and is signed by the CA for all nodes in the site.
21. Close the MMC console.
No changes have to be saved.
22. 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 site (page 102) procedure, do not start the Qlik Sense services.

- 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.

See: *Services (page 19)*

- 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 order

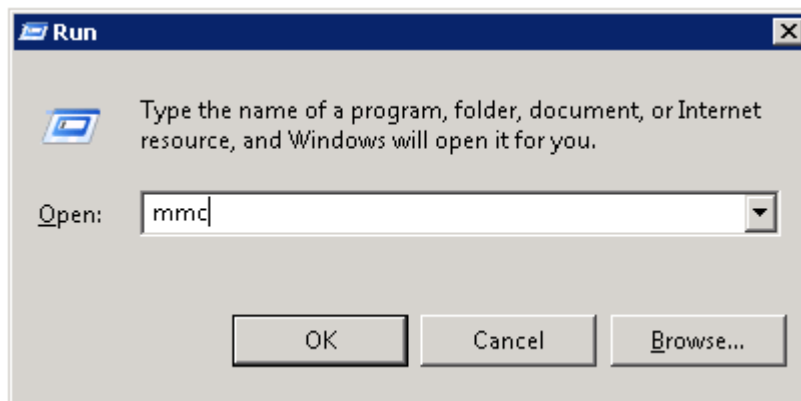
The order is important because the QRS is dependent on the QRD and the rest of the services are dependent on the QRS.

Moving a certificate

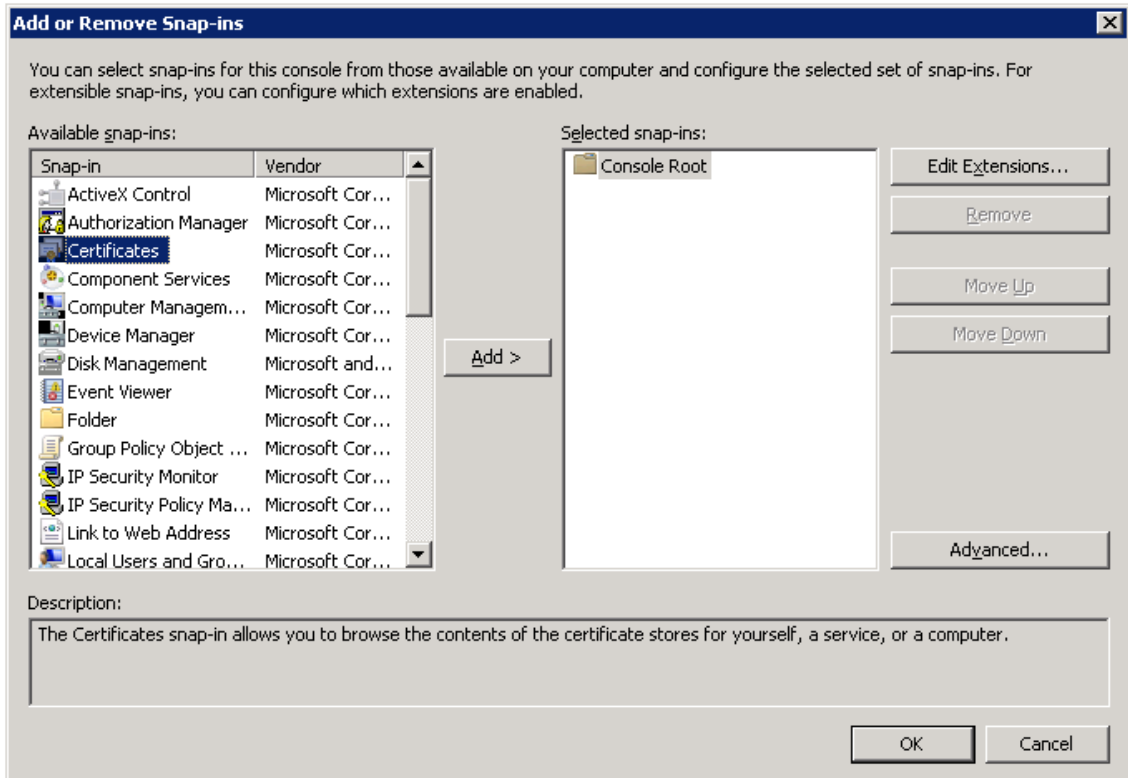
A backed up server certificate can be moved from a node in a Qlik Sense site to another node in the same site.

Proceed as follows to install the crypto key for the repository database on a new node in the site:

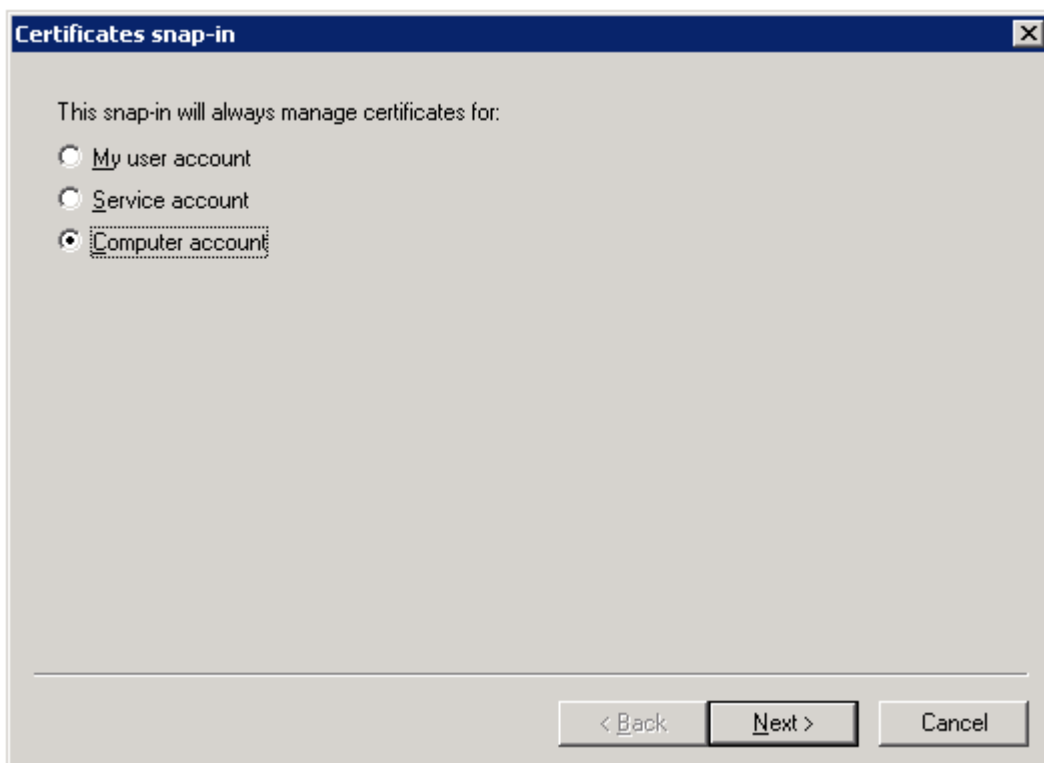
1. Open the Task Manager in Microsoft Windows and stop all Qlik Sense services except the Qlik Sense Repository Database (QRD) service.
2. Select **Start>Run**.
3. Enter *mmc* and click **OK**.



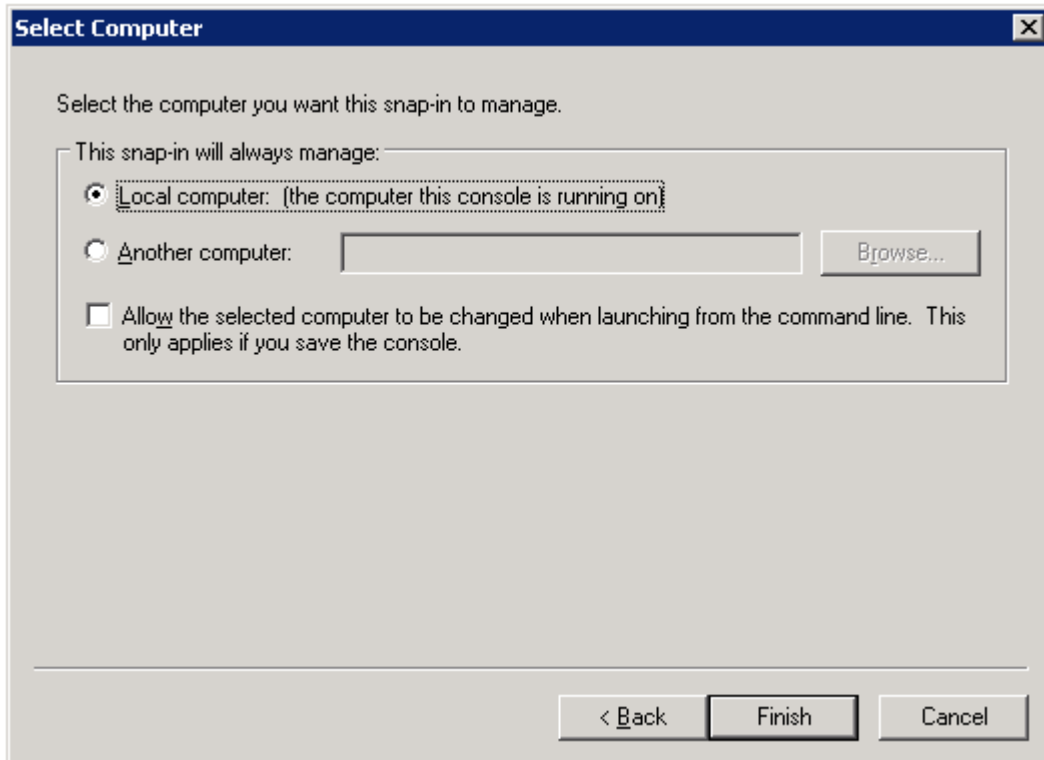
4. Select **File>Add/Remove Snap-in**.
5. Double-click **Certificates**.



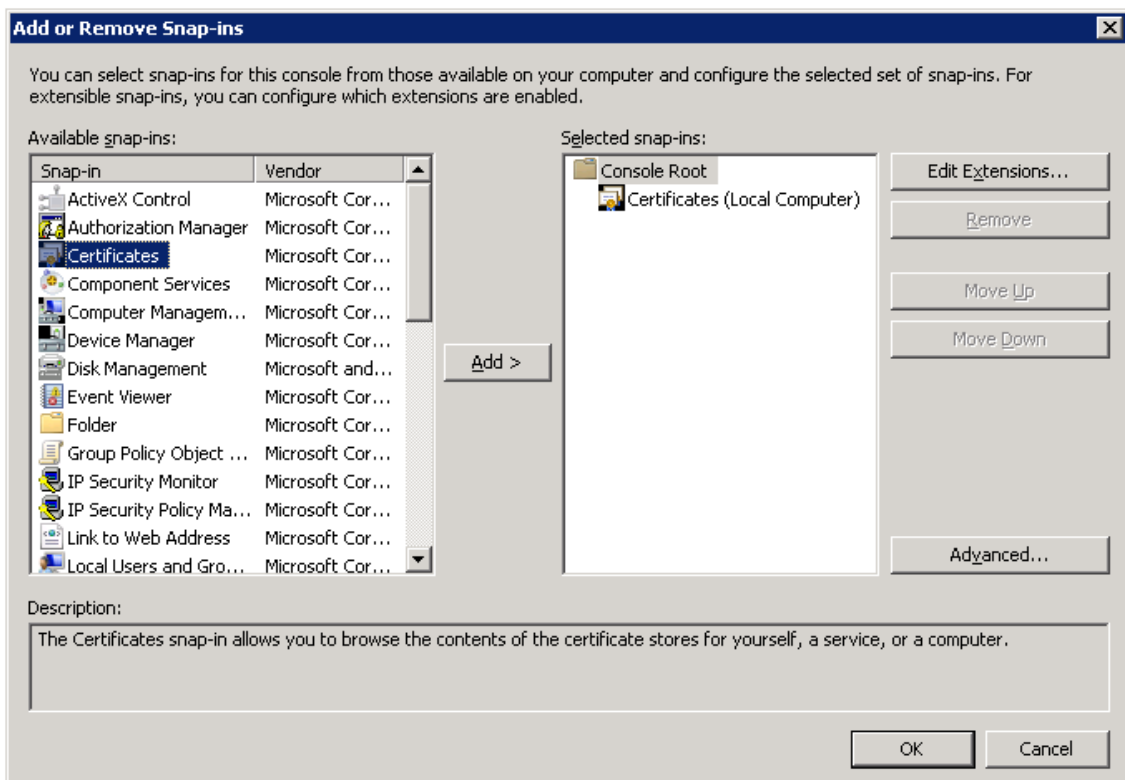
6. Select **Computer account** and click **Next**.



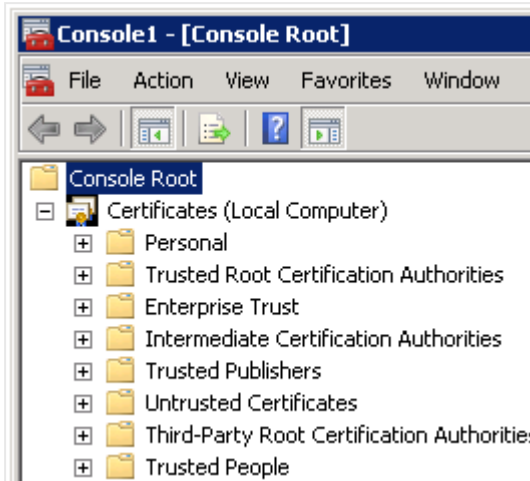
7. Select **Local computer** and click **Finish**.



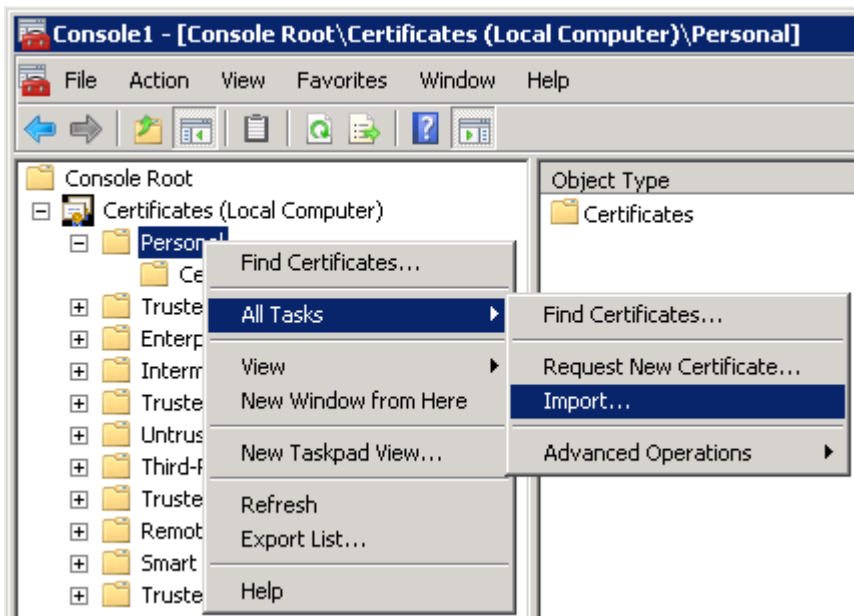
8. Click **OK**.



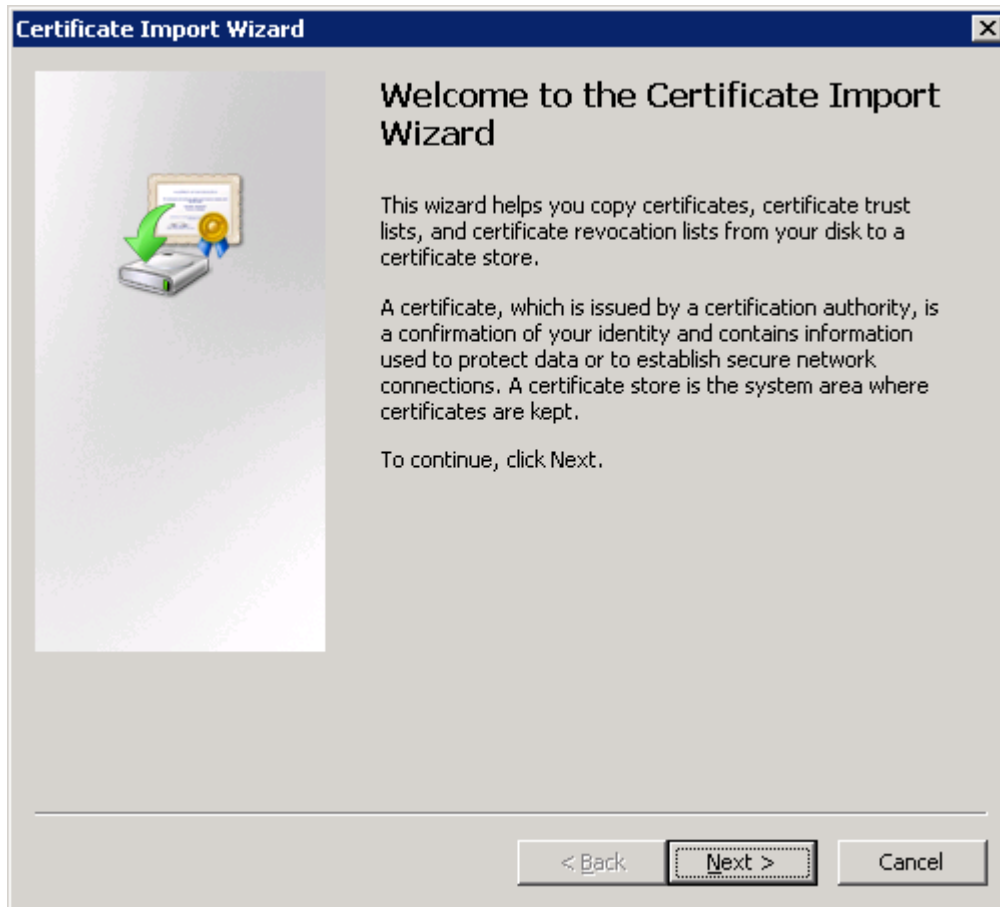
9. Expand **Certificates (Local Computer)** in the left panel.



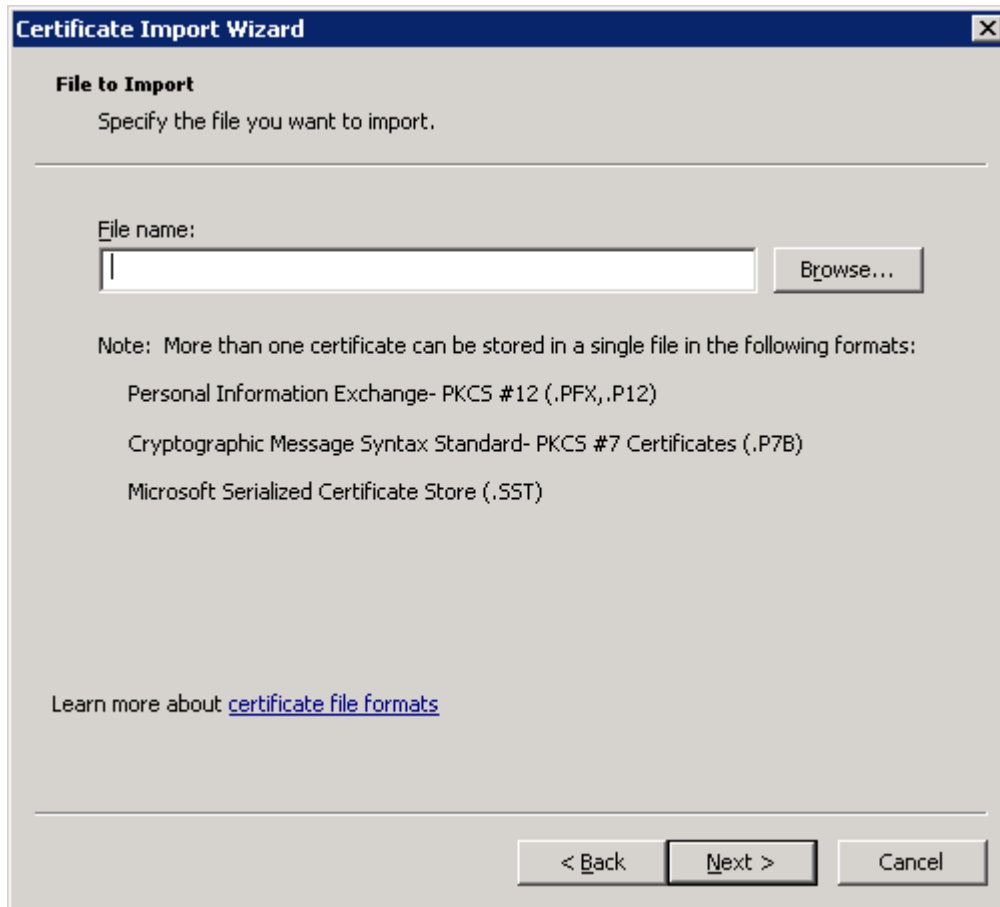
10. Right-click the **Personal** folder and select **All Tasks>Import**.



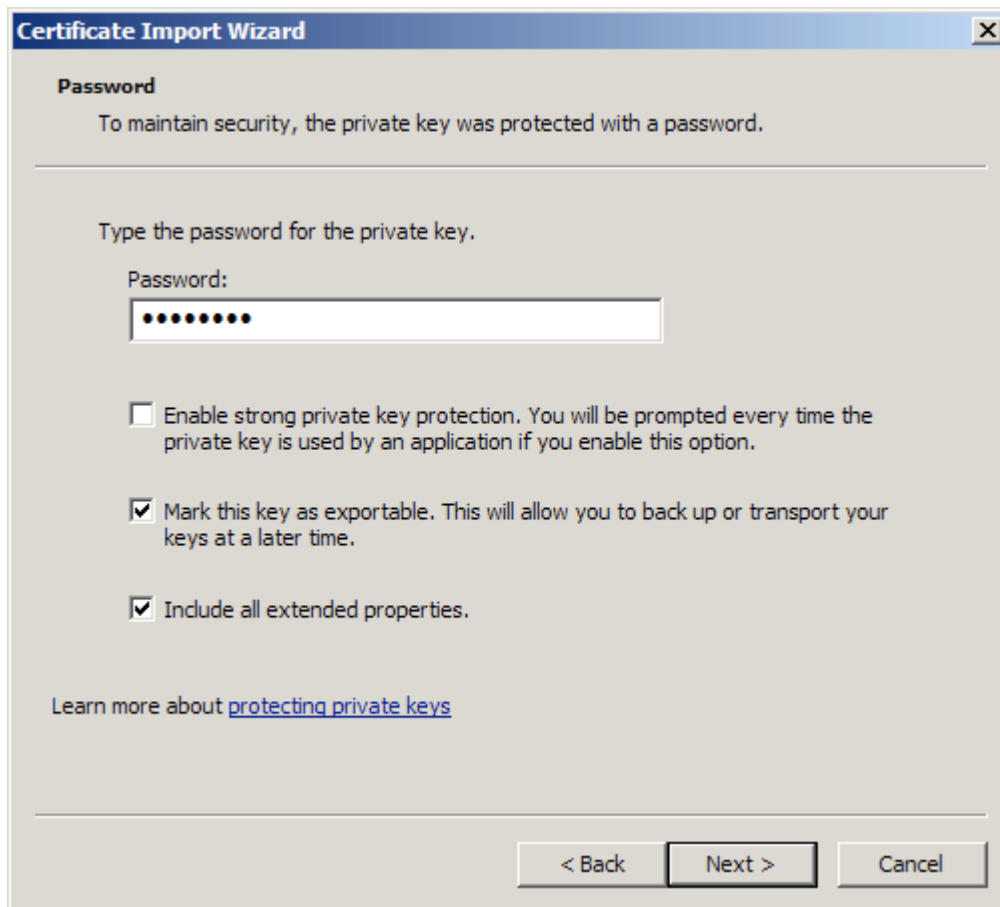
11. Click **Next**.



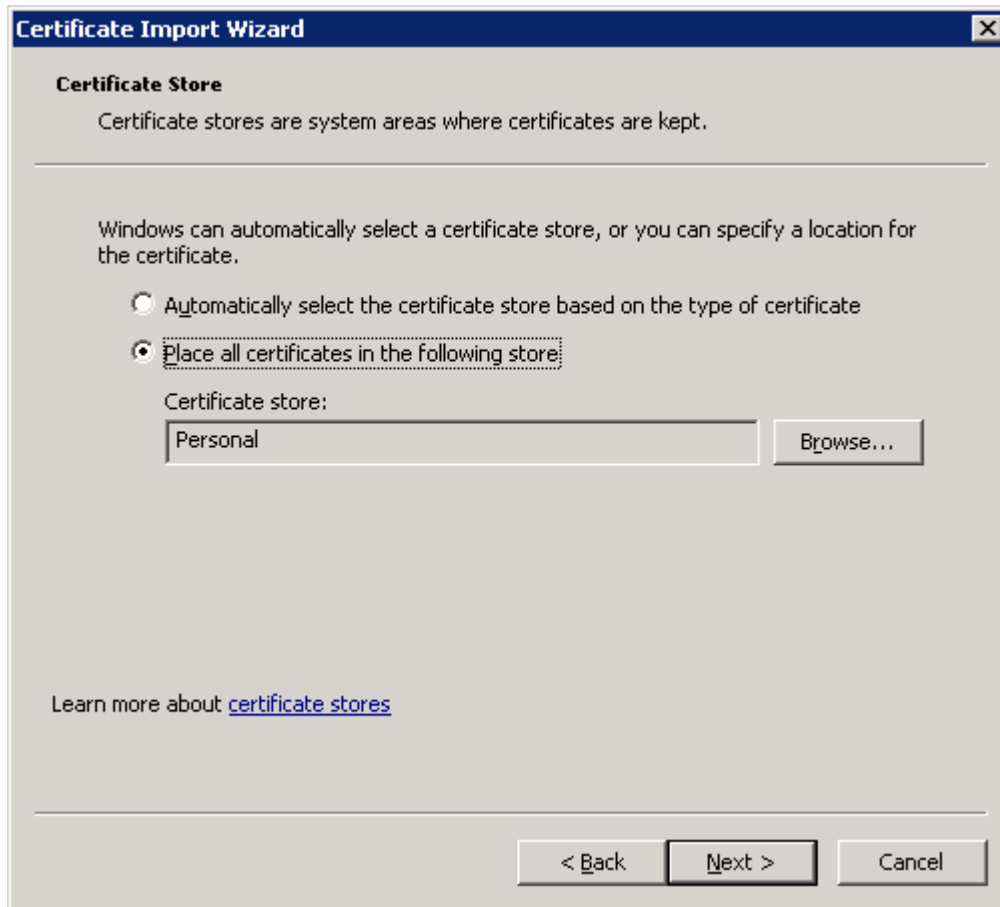
12. Browse to the file that contains the backed up server certificate. The server certificate a) has the same name as the Domain Name System (DNS) name of the machine, and b) is signed by the CA for all nodes in the site. Then click **Next**.



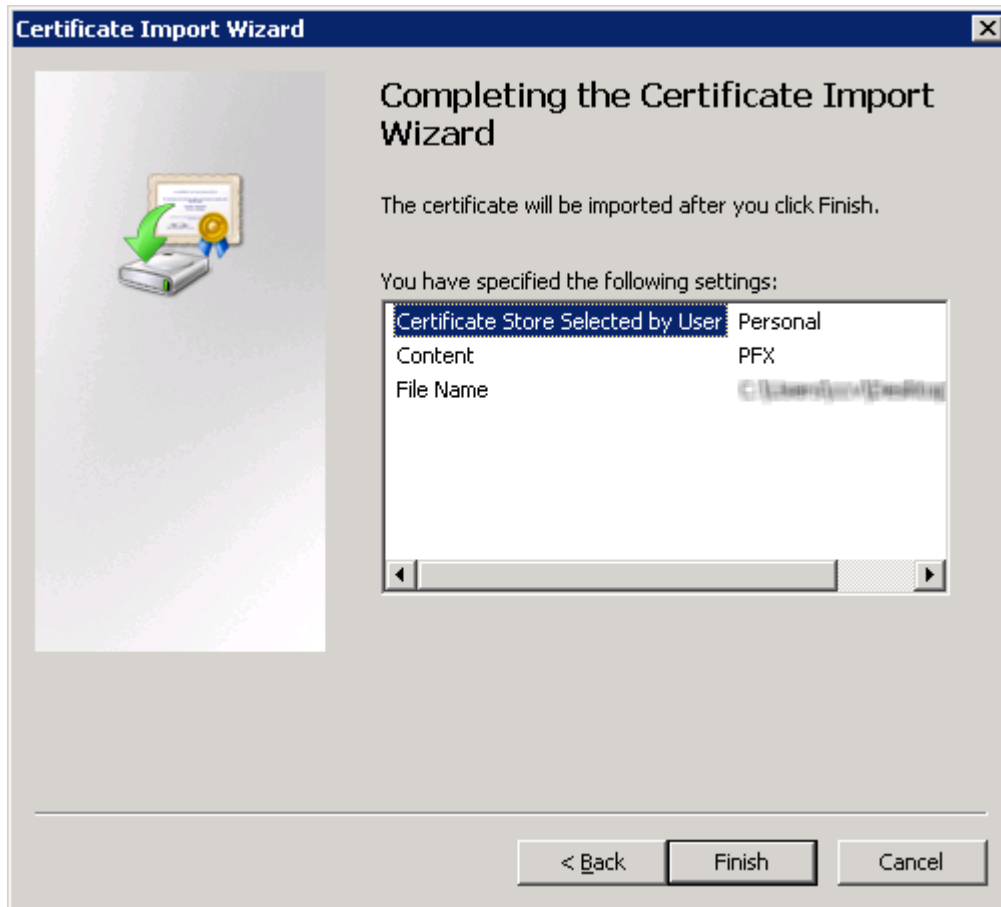
13. Enter the password for the *.pfx* file (that is, the password that was given when the file was exported).
14. Select **Mark this key as exportable** and **Include all extended properties**. Then click **Next**.



15. Select **Place all certificates in the following store** and click **Next**.



16. Click **Finish**.



17. Click the **Refresh** button (🔄) and check that the imported server certificate is available in the **Personal** folder.
18. Close the MMC console.
No changes have to be saved.
19. 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.
See: *Services (page 19)*
 - 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 order

The order is important because the QRS is dependent on the QRD and the rest of the services are dependent on the QRS.

See also:

- 📄 *Backing up and restoring certificates (page 104)*

5.3 Switching central node

In a multi-node site, one node is configured to be the central node, which is responsible for controlling the site. You can change which node is the central node. This may be needed if the central node fails, but can be used for any operational reason.



There can only be one central node in a Qlik Sense site, so when promoting a rim node to central node, the current central node must be demoted to rim node.

Prerequisites

A rim node that is to become central node must fulfill the following requirements:

- All Qlik Sense services are installed on the node.
- The private key for the root certificate is available on the node.

If not already available on the node, export the root certificate and private key from the central node and then import them to the rim node using the Certificate Store in Windows.

See: *Backing up and restoring certificates (page 104)*

Procedure

Do the following:

1. Stop all Qlik Sense services, except the Qlik Sense Storage Service (QST), on the central node. Then stop all Qlik Sense services, except the QST, on all rim nodes.
2. Run the following script on the rim node that is to be promoted to central node:

```
%ProgramFiles%\Qlik\Sense\Repository\Set-Role.ps1 -role central
```

The script adds the *-iscentral* flag to the Qlik Sense Repository Service (QRS) on the node.

3. Run the following script on the current central node that is to be demoted to rim node:

```
%ProgramFiles%\Qlik\Sense\Repository\Set-Role.ps1 -role rim
```

The script removes the *-iscentral* flag from the QRS on the node.

4. Start the Qlik Sense services in the following order on the rim node that is to become central node:
 - a. Qlik Sense Service Dispatcher (QSD), which launches the Content Service
 - b. Qlik Sense Repository Service (QRS)
 - 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. Start the Qlik Sense services on the rim nodes (including the former central node) in the same order as described in step 4.



Both the new and the former central node retain their original settings (for example, virtual proxy configuration). For the new central node to be an exact replica of the former, any settings have to be manually replicated across the two.

6 Security

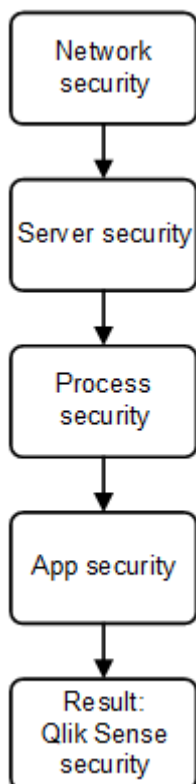
The security in Qlik Sense consists of the following parts:

- 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.

6.1 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.

The figure below shows the components that have to be considered in order to maximize the security.



Network security

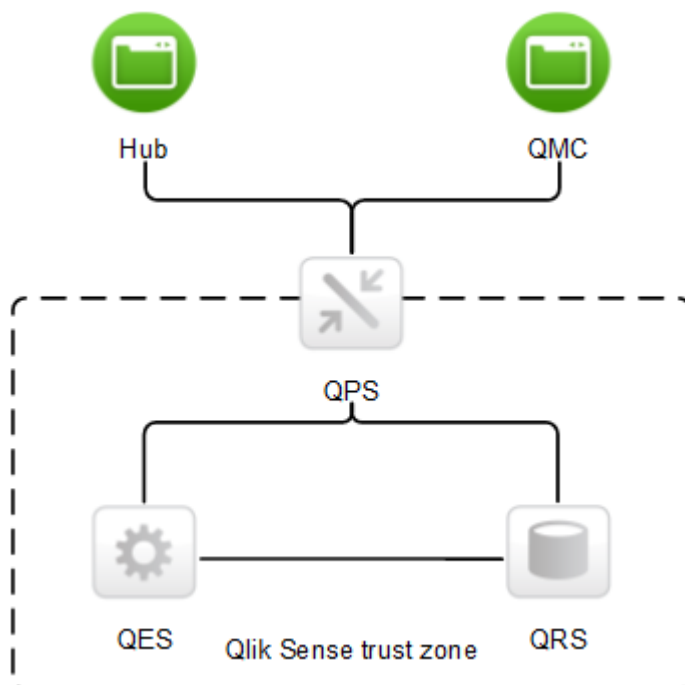
For all Qlik Sense components to communicate with each other in a secure way, they need to build trust.

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.

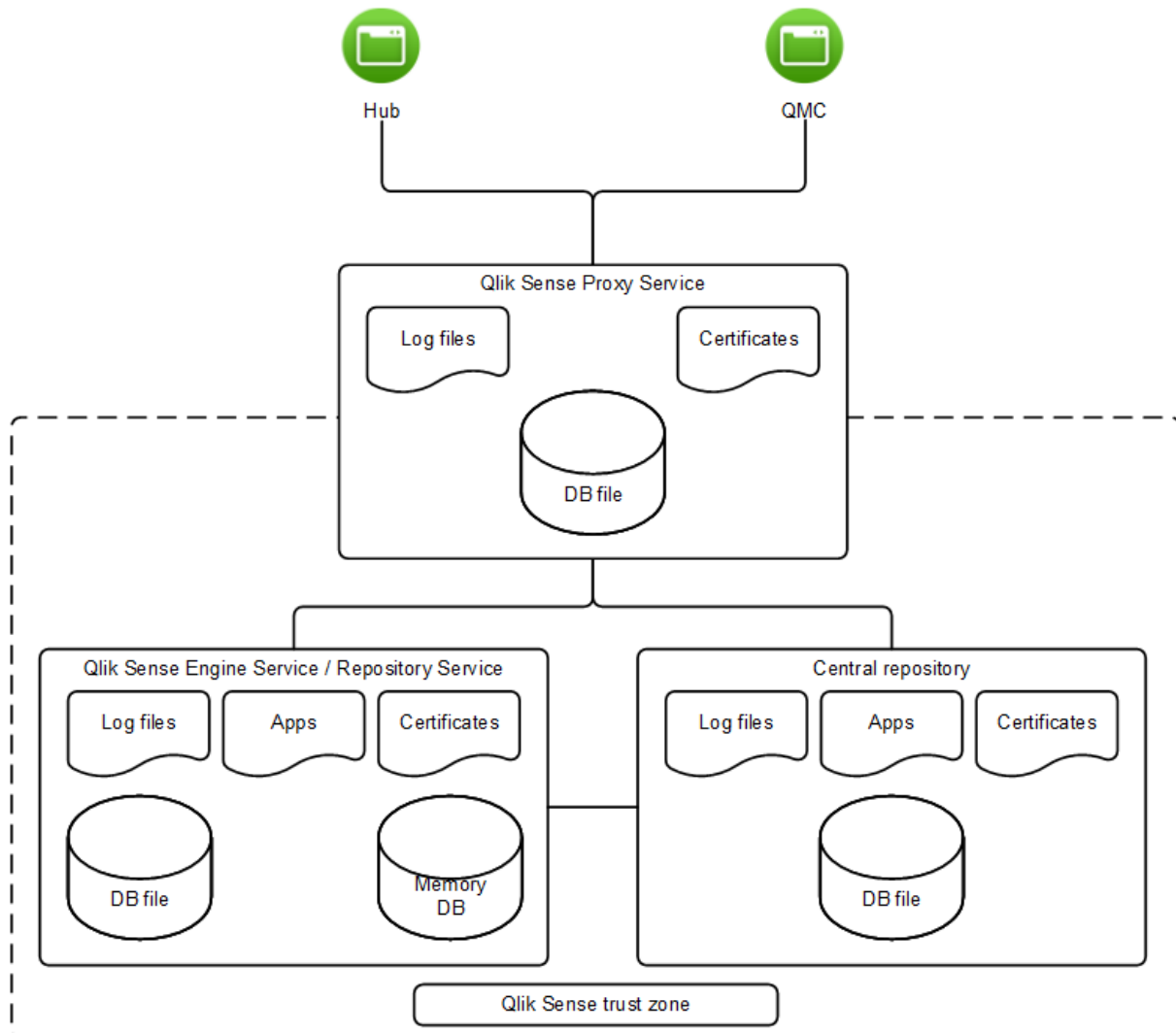
TLS-protected tunnels can be used to secure the communication between the Qlik Sense clients and the QPS. As the clients are outside of the Qlik Sense trust zone, the communication between the clients and the QPS uses a certificate with a different signature than the one used within the trust zone.

See also:

[Certificate trust \(page 139\)](#)

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.
- Data reduction: The data reduction functionality is a way to dynamically change which data a user can view. This makes it possible to build apps that can be consumed 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).

Using these components, the resources and data (that is, the content) consumed by the Qlik Sense users can be secured.

6.2 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, or SAML.

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).

See also:

[Network security \(page 134\)](#)

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 the 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

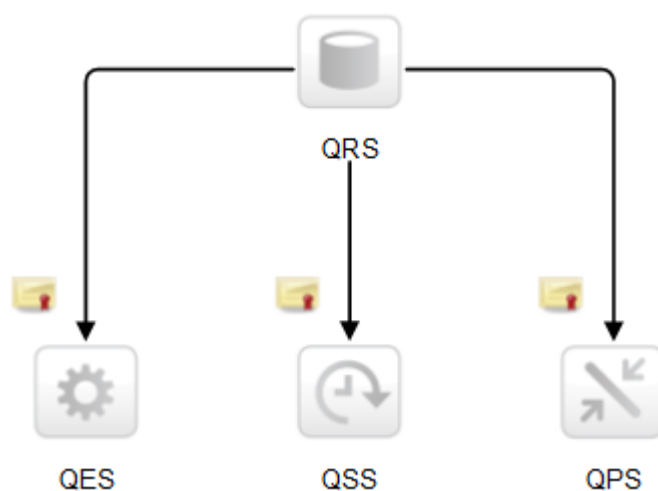
Qlik Sense uses certificates for authentication. A certificate provides trust between nodes within a site.

This section describes how to deploy certificates for use in Qlik Sense.

Certificate architecture

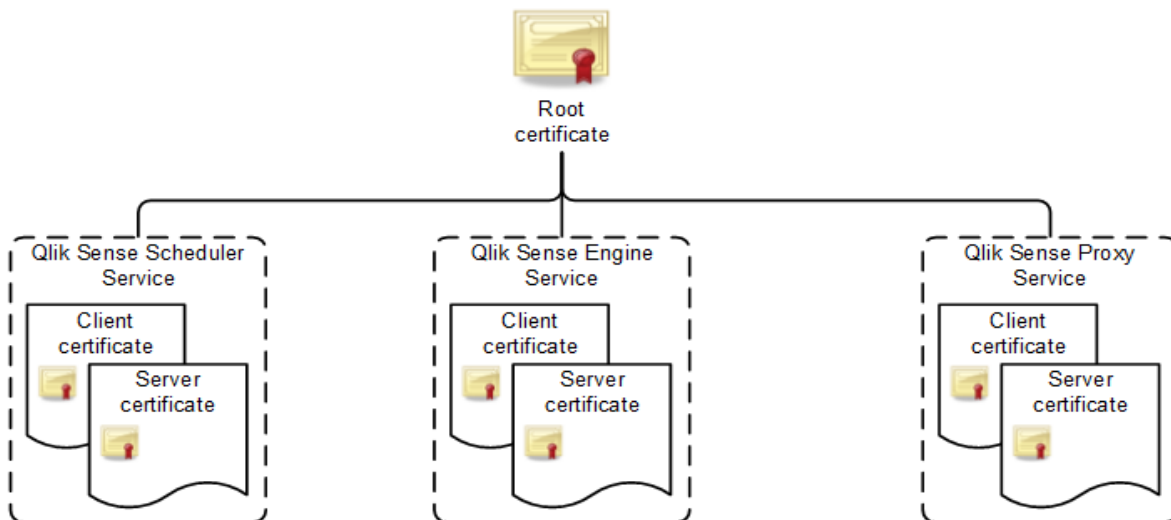
Certificates are used within a Qlik Sense site to authenticate communication between services that reside on different nodes. In addition, certificates can be used to build a trust domain between services that are located in different domains or areas (for example, internal networks, extranets, and Internet) without having to share a Microsoft Active Directory (AD) or other user directories.

The architecture is based on the master Qlik Sense Repository Service (QRS) on the central node acting as the certificate manager or Certificate Authority (CA). The master QRS creates and distributes certificates to all nodes within a site. The master 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 master 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 master QRS. The master 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 master 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.

Certificate trust requirements

The requirements described in this section must be fulfilled for the certificate trust to function properly.

General

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.

You should also make sure that TLS 1.0 is enabled in Windows.

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.

Port no.	Description
4570	<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> <p>This port uses HTTP for communication.</p>
4444	<p>Security distribution port, only used by Qlik Sense Repository Services (QRSs) on rim nodes to receive a certificate from the master QRS on the central node. The communication is always unencrypted, but the transferred certificate package is password-protected.</p> <p>This port uses HTTP for communication.</p>

See: *Ports (page 28)*

Unlocking distributed certificates

When adding a new rim node to a site, the distributed certificate needs to be unlocked.

See: *Manage Qlik Sense sites*

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.

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

Handling of certificates when a service starts

This section describes how the certificates are handled when a Qlik Sense service starts.

Client certificate

This section describes how the master 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 creates a new certificate.
- If only one client certificate is found, the QRS checks if it is valid. If the certificate is not valid, the QRS deletes the certificate and creates a new one. In addition, the QRS logs that an invalid certificate was found and deleted.
- If more than one client certificate is found, the QRS deletes all certificates and creates a new one. Duplicates are not allowed. In addition, the QRS logs the number of valid and invalid certificates that were found and deleted.

Server certificate

This section describes how the master 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 creates a new certificate.
- If only one server certificate is found, the QRS checks if it is valid. If the certificate is not valid, the QRS deletes the certificate and creates a new one. In addition, the QRS logs that an invalid certificate was found and deleted.
- If more than one server certificate is found, the QRS deletes all certificates and creates a new one. Duplicates are not allowed. In addition, the QRS logs the number of valid and invalid certificates that were found and deleted.

Root certificate

This section describes how the master 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 creates a new certificate.
- 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 manually has to delete any unwanted certificates. In addition, the QRS logs information on 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.



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.

See also: *Services (page 19)*

Definition of 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

This section describes various authentication solutions for Qlik Sense.

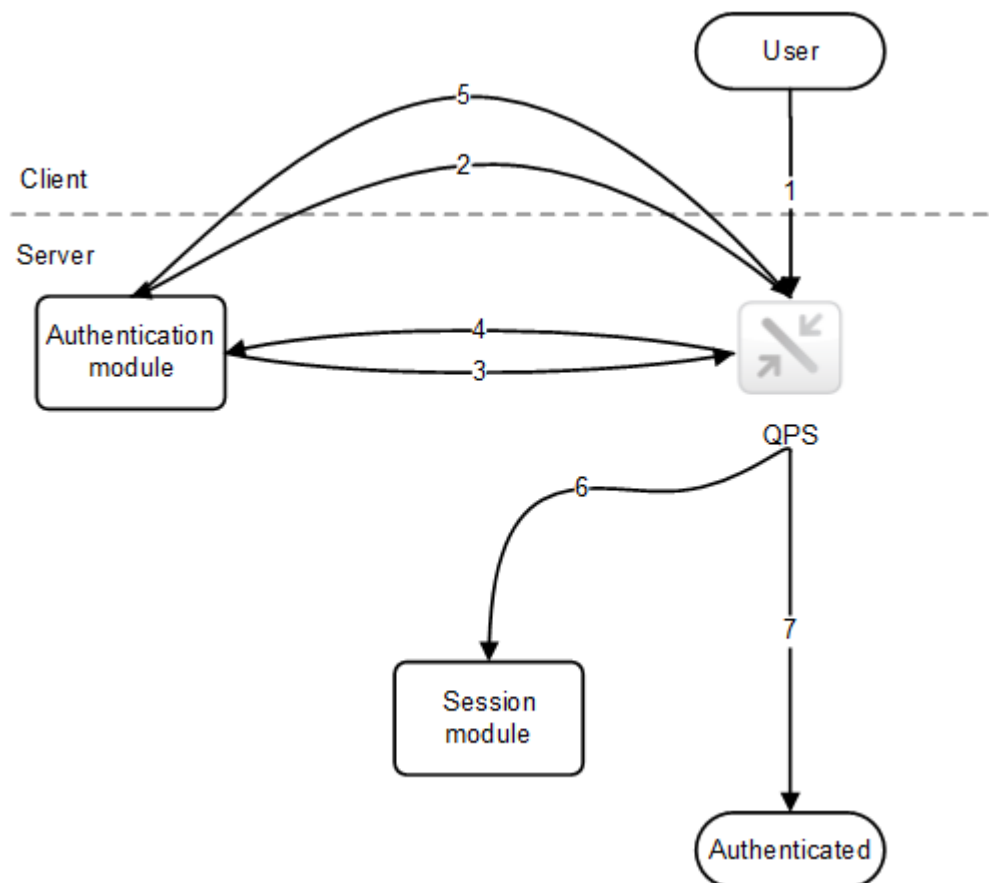
Ticket solution

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.

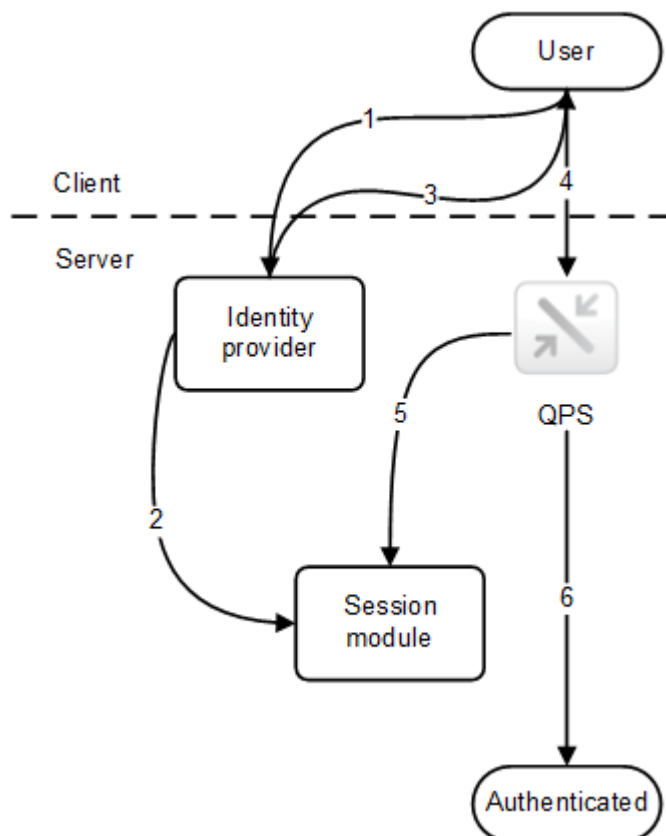
3. Once the credentials have been verified, a ticket is requested from the QPS. Additional properties may be supplied in the request.
4. The authentication module receives a ticket.
5. The user is redirected back to the QPS with the ticket. The QPS checks that the ticket is valid and has not timed out.
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.

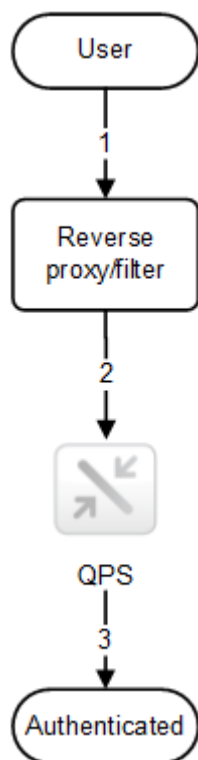


The name of the session cookie used by the authentication module can be configured in the Qlik Management Console (QMC).

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).

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.

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.
- Only the following signing algorithms are supported:
 - RS256 - RSA signature with SHA256
 - RS384 - RSA signature with SHA384
 - RS512 - RSA signature with SHA512

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.

6.3 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 a way to dynamically change which data a user can view. This makes it possible to build apps that can be consumed 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).

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:

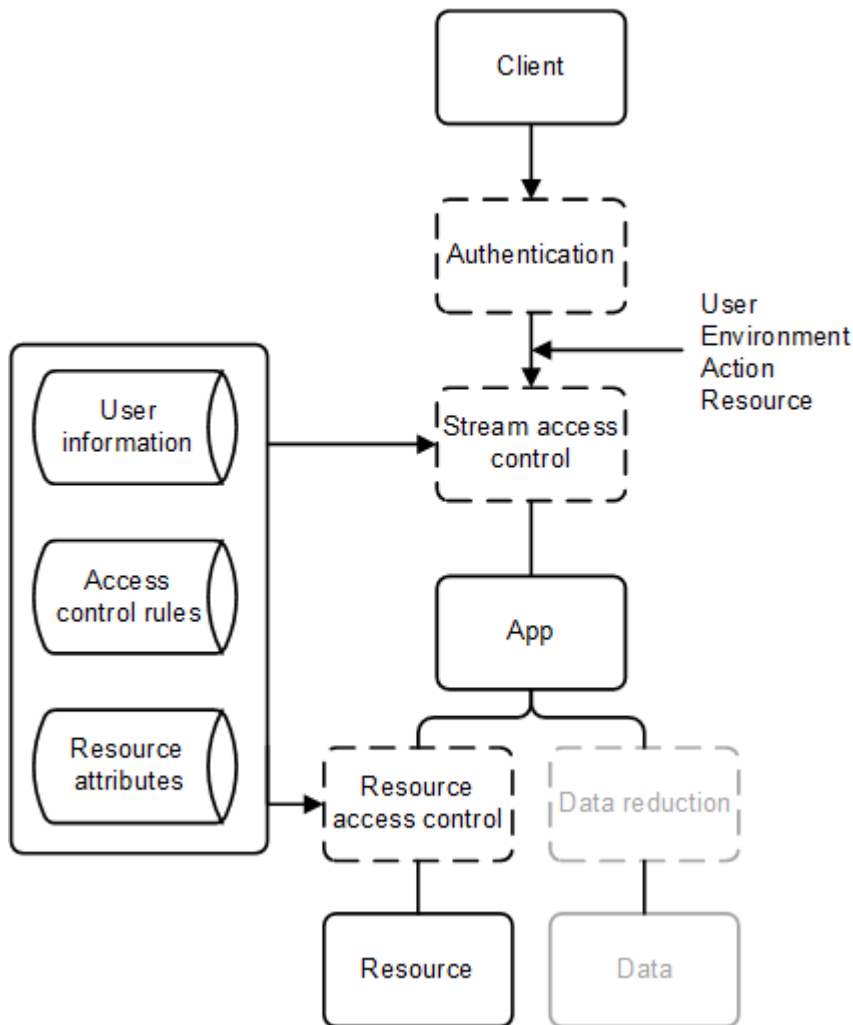
- User name and user properties: The user name and user properties are supplied by the Qlik Sense Proxy Service (QPS) that authenticated the user.
- Action: The method that the user is trying to perform on a resource (for example, create, read, or print).
- Resource: The entity that the user is trying 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.



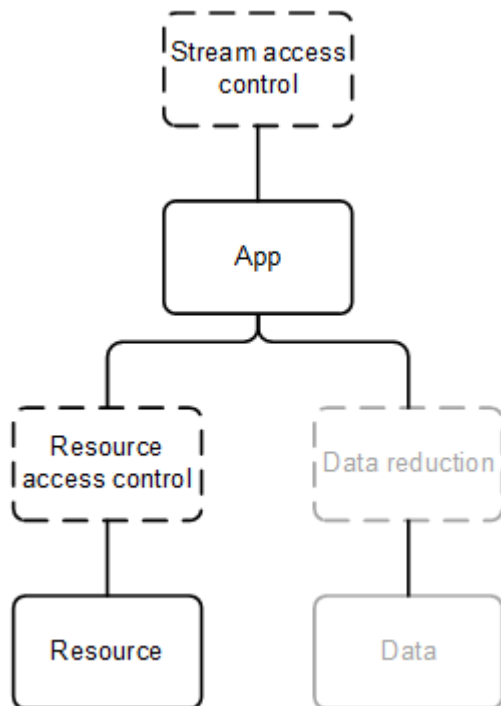
Resource access control streams

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:

- RootAdmin: Full access to all Qlik Sense resources.
- AuditAdmin: Read access to all resources.
- ContentAdmin: Full access to all resources except nodes, engines, repositories, schedulers, and syncs.
- DeploymentAdmin: Full access to apps, tasks, licenses, nodes, repositories, schedulers, proxies, virtual proxies, and engines.
- SecurityAdmin: Same as ContentAdmin, but with full access to proxies and virtual proxies and no access to tasks.

Data reduction

Data reduction is used to determine which data a user is allowed to see: all of it or just parts of it?

The data reduction functionality is a way to dynamically change which data a user can view. This makes it possible to build apps that can be consumed 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).

The definition of access rights for section access is maintained in the apps and configured through the load script.

6.4 Security summary

This section provides a summary of the Qlik Sense security system.

Authentication

Qlik Sense supports authentication in the following ways:

- The users are authenticated by the Qlik Sense Proxy Service (QPS).
- The QPS supports the use of multiple proxies and each proxy can use multiple authentication methods over a network protected by Transport Layer Security (TLS).

Authorization

Qlik Sense supports authorization in the following ways:

- Inter-server communication is authorized through Transport Layer Security (TLS) using certificates for authentication.
- The Qlik Sense Repository Service (QRS) provides property-based access control of user content.
- Authorization to data is managed using section access.

Auditing

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.

Confidentiality

Qlik Sense supports confidentiality in the following ways:

- The network uses Transport Layer Security (TLS) for encryption and certificates for authentication.
- The locally stored information on a node, 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 encryption.
- The app data is protected using data reduction.

Integrity

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 encryption.
- Qlik Sense does not support write back to the source system (that is, the Qlik Sense clients cannot edit the data sources).

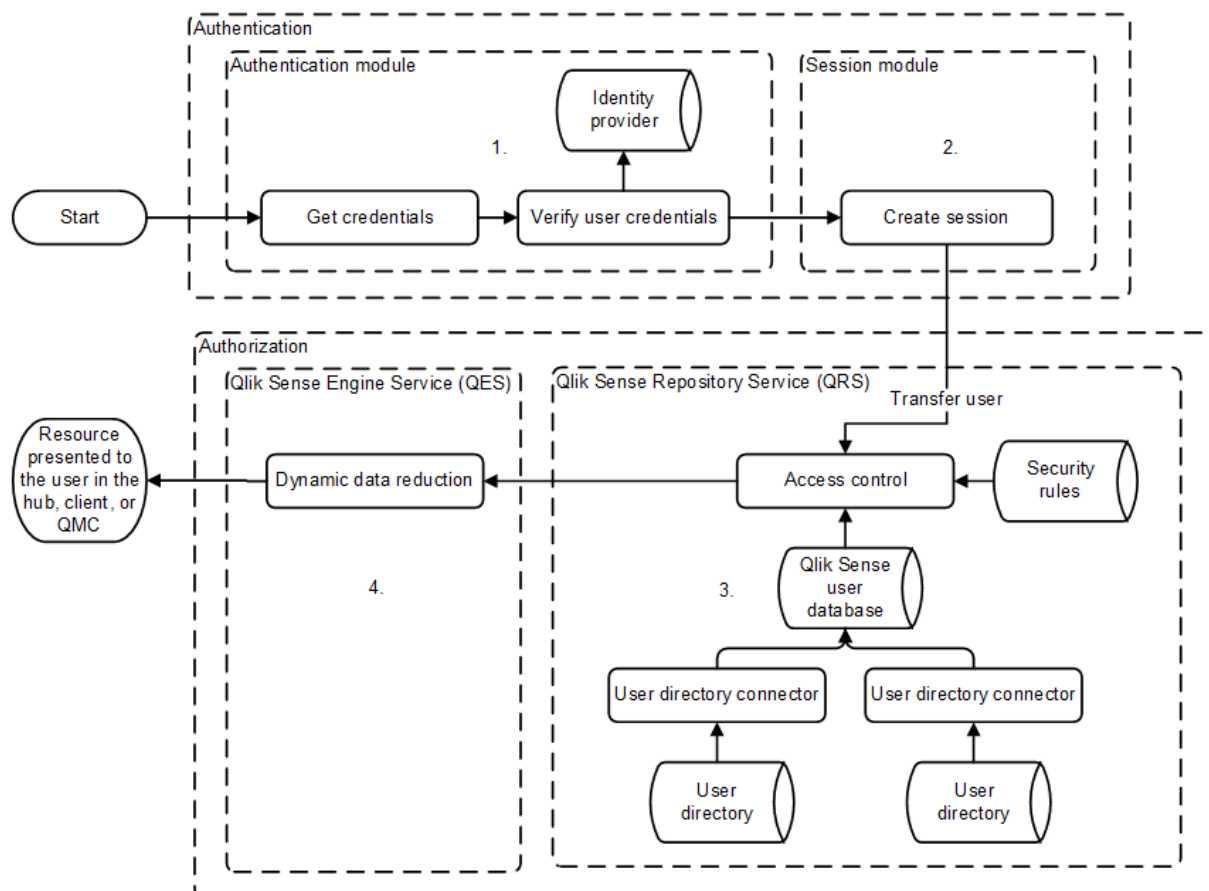
Availability

Qlik Sense supports availability in the following ways:

- The nodes in a multi-node site are resilient by design. Each node has a local copy of the data that it needs to fulfill its role.
- The Qlik Sense protocols are designed to be fault tolerant.

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 rights are stored.

The users are imported into the repository database from one or more User Directories (UDs) (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.

See also:

 [App security \(page 137\)](#)

7 Logging

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

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 New logging framework

A new logging framework was introduced in Qlik Sense version 2.0. Unless otherwise stated, the documentation describes the new 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, but they are stored in a new location.

See: *Trace logs (page 171)*

7.3 Reading and analyzing log files in Qlik Sense

The log files can be read and analyzed using Qlik Sense, which includes the following pre-defined, log-related data connections after installation:

- ServerLogFolder: Links to the active log files.
- ArchivedLogsFolder: Links to the archived log files.

The data connections can be edited in the Qlik Management Console (QMC).

In addition, users with root, security, content, or deployment administrator rights can use the Qlik Sense log data in apps by selecting one of the data connections listed above in the data load editor.

See also:

 [Apache Logging Services](#)

7.4 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.

7.5 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.

Folder	Sub-folder	Files	Description
<code>\AppMigration</code>			This folder contains log files related to the Migration Service.
<code>\BrokerService</code>			This folder contains log files related to the Broker Service.
<code>\DataProfiling</code>			This folder contains log files related to the Data Profiling Service.
<code>\Engine</code>		<code><MachineName>_Exit_Engine_<Date>.txt</code>	This 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. This log file is not archived.

Folder	Sub-folder	Files	Description
		<MachineName>_Start_Engine_<Date>.txt	This 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. This 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 171)</i>
\HubService			This folder contains log files related to the Hub Service.
\Printing	\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 171)</i>
\Proxy	\Audit	<MachineName>_AuditActivity_<Service>.txt	This log tracks user-related actions.

Folder	Sub-folder	Files	Description
		<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 171)</i>
\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 171)</i>
\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 171)</i>
\Script			This folder contains log files related to app reloads.
\WebExtensionService			This folder contains log files related to the Web Extension Service.

Archived log files

Archived log files are by default stored in `%ProgramData%\Qlik\Sense\Repository\Archived Logs` on the central node in the Qlik Sense site. Archived log files have the file extension `.log`, whereas active log files have the extension `.txt`.

See also:

 [Local log configuration file \(page 187\)](#)

7.6 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>_<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>_<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).
- `<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

7.7 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.

Character	Unicode replacement	Description
\t	\u21d4	Symbol for horizontal tabulation, HT.
\n	\u2193	Symbol for line feed, LF.
\f	\u2192	Symbol for form feed, FF.
\r	\u21b5	Symbol for carriage return, CR.

7.8 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.

Field	Format	Description
Sequence#	Int	1 - 2147483647 by default, but can be configured using custom logging as described in <i>Appendix (page 184)</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"> • <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>
Hostname	String	The name of the server.
Id	String	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><code>Command=<CommandName>;Result=<ReturnCode (Int)>;ResultText=<Description, Success, or Error message></code></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>
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	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>
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 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.
Result	String	<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).

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.

Field	Format	Description
Sequence#	Int	1 - 2147483647 by default, but can be configured using custom logging as described in <i>Appendix (page 184)</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"> • <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>
HostName	String	The name of the server.
Id	GUID	A unique identifier of the log entry (added by Log4net).
Description	String	A human-readable message that summarizes the action in the system. Format: <code>Command=<CommandName>;Result=<ReturnCode (Int)>;ResultText=<Description, Success, or Error message></code>

Field	Format	Description
ProxySessionId	String	The ID of the proxy session. 0 = Internal system command or a command that does not go through the QPS
ProxyPackageld	String	A unique ID of each HTTP(S) package that passes through the Qlik Sense Proxy Service (QPS). 0 = Internal system command or a command that does not go through the QPS
RequestSequenceld	String	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. The initial RequestSequenceld is an integer. Subrequests are linked to the initial request by adding a dot and an ID for the subrequest: <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 0 = Internal system command or a command that does not go through the Qlik Sense Engine Service (QES)
UserDirectory	String	The user directory linked to the logged in Qlik Sense user. System = Internal system command
UserId	String	The Qlik Sense user that initiated the command. System = Internal system command

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
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.

Field	Format	Description
Service	String	The Qlik Sense service on the server that hosts the process.
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".
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`.

Field	Format	Description
Sequence#	Int	1 - 2147483647 by default, but can be configured using custom logging as described in <i>Appendix (page 184)</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"> • <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>
Severity	String	<p>Row log level, can be configured using custom logging as described in <i>Appenders (page 184)</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>

Field	Format	Description
ProxySessionId	String	The ID of the proxy session. 0 = Internal system command or a command that does not go through the QPS
ProxyPackageId	String	A unique ID of each HTTP(S) package that passes through the Qlik Sense Proxy Service (QPS). 0 = Internal system command or a command that does not go through the QPS
RequestSequenceId	String	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. The initial RequestSequenceId is an integer. Subrequests are linked to the initial request by adding a dot and an ID for the subrequest: <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 0 = Internal system command or a command that does not go through the Qlik Sense Engine Service (QES)
UserDirectory	String	The user directory linked to the logged in Qlik Sense user. System = Internal system command
UserId	String	The Qlik Sense user that initiated the command. System = Internal system command

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.
Origin	String	<p>The origin of the request:</p> <ul style="list-style-type: none"> • AppAccess • ManagementAccess • Not available

Field	Format	Description
Context	String	The context of the command. The context can be Internal System command or User Activity command (based on URL for the command).
Command	String	The core name of the use case or system command.
Result	Int	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).

Qlik Sense engine service log fields

The following table lists the fields that are unique for the Qlik Sense Engine Service (QES) logs.

Field	Format	Description
EngineTimestamp	ISO 8601	The date and time when the QES wrote the log message to file. 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.
EngineVersion	String	The version number of the QES that executed the request.

7.9 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, 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.
See: *Logger (page 175)*
- `<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 175)*

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.

Character	Unicode replacement	Description
\t	\u21d4	Symbol for horizontal tabulation, HT.
\n	\u2193	Symbol for line feed, LF.
\f	\u2192	Symbol for form feed, FF.
\r	\u21b5	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.

Field	Description
Sequence#	1 - 2147483647 by default, but can be configured using custom logging as described in <i>Qlik Sense Appenders (page 184)</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 184)</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.
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. • 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.

See also:

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).

See also:

 [Common fields \(page 173\)](#)

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.
See: ActiveConnections field in *Performance log (page 178)*
- 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.

See also:

 [Common fields \(page 173\)](#)

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.

See also:

 [Common fields \(page 173\)](#)

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.

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.

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.
- CServerId: The ID of the server instance that handled the request.
- SessionId: The ID of the engine session for which the QIX method call was made.
- 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.
- 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.
- Handle: The ID of the interface that handled the request. The interface can be Global, a certain sheet, a certain object, or similar.

See also:

 [Common fields \(page 173\)](#)

Qlik Management Console log



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.

See also:

 [Common fields \(page 173\)](#)

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 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.
- 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.
- Qlik Sense User: The user that started 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 milliseconds) 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.
- Client Machine Identification: The ID of the client machine that opened the engine session.
- Serial Number: The serial number that was used during the engine session.
- Client Type: The type of client that was used for the engine session.
- Client Build Version: The build version of the client.
- Secure Protocol: An on/off flag that indicates whether the protocol was run over a secure connection.

See also:

 [Common fields \(page 173\)](#)

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.

See also:

 [Common fields \(page 173\)](#)

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.

See also:

 [Common fields \(page 173\)](#)

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.

See also:

 [Common fields \(page 173\)](#)

7.10 Configuring the logging

The standard logging in Qlik Sense is configured using the Qlik Management Console (QMC).

Customized logging is setup using appenders and the local log configuration file, *LocalLogConfig.xml*.

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##x9;
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`
- `SmtppickerDirAppender`
- `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:



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>
</configuration>
```

See also:

 [Converters \(page 185\)](#)

8 Licensing

The licensing in Qlik Sense is based on tokens, which are used to allocate access passes that allow users to access Qlik Sense. There are different types of access passes to choose from and each type corresponds to a specific consumption model for accessing Qlik Sense.



The tokens used in Qlik Sense are not compatible with the Client Access Licenses (CALs) used in QlikView. In addition, QlikView licenses cannot be used in Qlik Sense.

8.1 License Enabler File

The Qlik Sense licensing is administered using a License Enabler File (LEF), which 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.

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.

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.

See: Removing access passes (page 193)

8.2 Access passes

The licensing in Qlik Sense is based on tokens, which are used to allocate access passes that allow users to access Qlik Sense. There are different types of access passes to choose from and each type corresponds to a specific consumption model for accessing Qlik Sense.

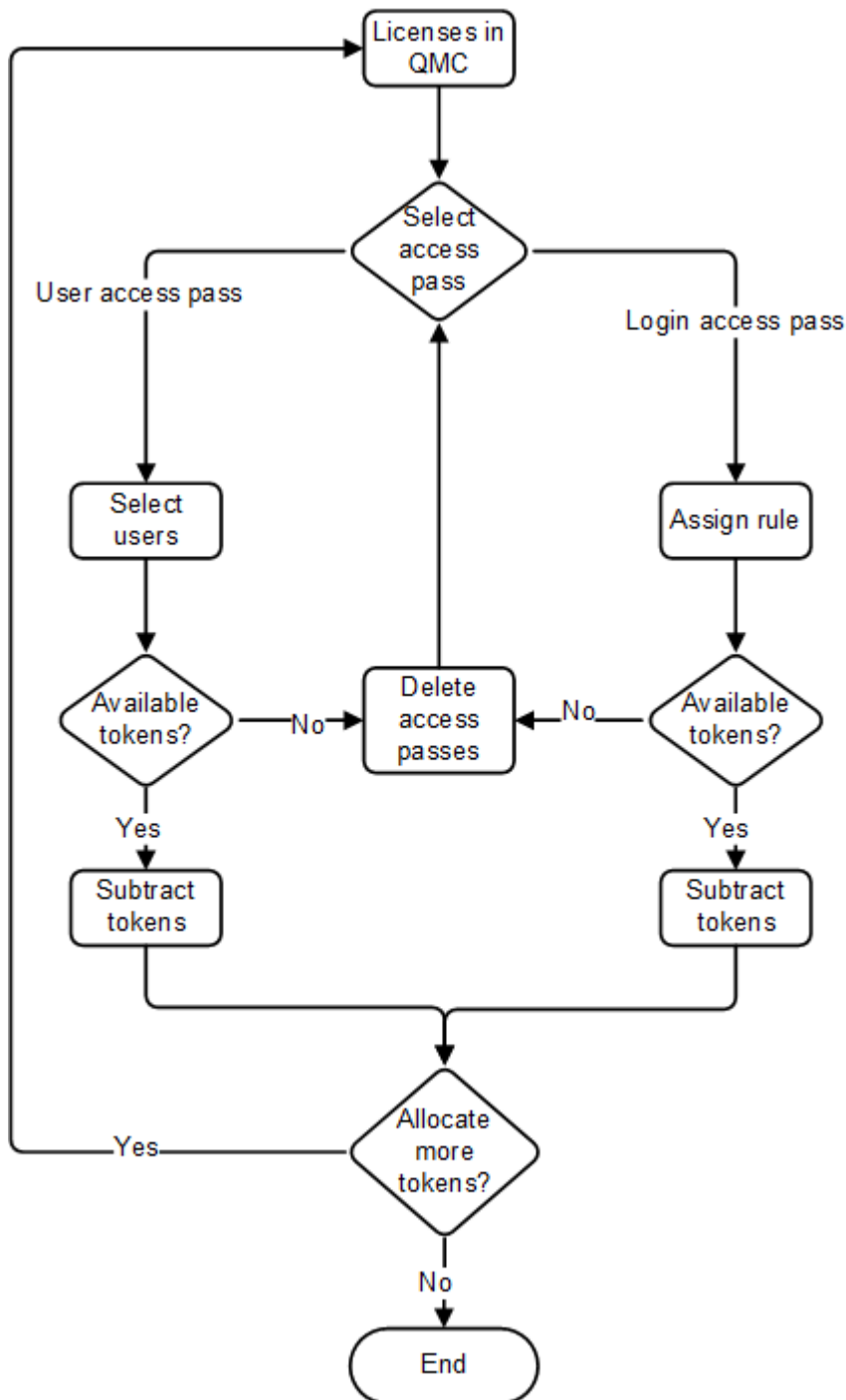
A user connection is the combination of device and browser that is used by a single user to connect to Qlik Sense. If a user who already has a user connection connects to Qlik Sense from another browser or device, an additional user connection is established.

The following table lists the types of access passes that are available in Qlik Sense.

Access type	Description
User access pass	<p>This type of access pass allows a unique and identified user to access the hub.</p> <p>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).</p> <p>One (1) token corresponds to one (1) access pass. The access passes are allocated using the Qlik Management Console (QMC).</p>
Login access pass	<p>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.</p> <ul style="list-style-type: none"> • 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. <p>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.</p> <p>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.</p> <p>One (1) token corresponds to ten (10) access passes. The access passes are allocated using login access groups in the QMC.</p>

Allocation of access passes

The following figure shows how the Qlik Management Console (QMC) is used to manage the allocation of access passes.



Login and logout

Login

When a user logs in to Qlik Sense, an access pass of the applicable type is used to provide the user with access to Qlik Sense.

Logout

When a user logs out of Qlik Sense, the following happens depending on the type of access pass used:

- User access pass: The access pass is not affected when the user logs out.
- Login access pass: The access pass that was used to access Qlik Sense is considered to be used and will not be available for a new login until the period specified in *Login access pass (page 191)* has passed.

Removing access passes

This section describes how to free up tokens for new allocations of access passes by removing existing access passes in the Qlik Management Console (QMC).

User access pass

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

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 *Login access pass (page 191)* has passed, one (1) token is freed up.

Disconnected node

A disconnected node is a rim node that fails to synchronize with the central node in a Qlik Sense site. A disconnected node continues to serve users to the best of its ability while waiting for a synchronization with the central node to take place.

Multi-deployment sites

This section describes how the Qlik Sense licensing is handled within multi-deployment sites, where apps are promoted from a development site to a test site and finally to a production site.

Development site

In a Qlik Sense deployment that includes a development site and a production site, two (2) License Enabler Files (LEF) are needed (that is, one per site).

Each node within the development site is licensed with one (1) access pass type (for example, user access passes), if only disconnected users are expected.


Test site

The LEF for a test site mirrors the LEF for a development site.

Anonymous users

Anonymous users only use login access passes.

See also:

 *Login access pass (page 191)*

8.3 Licensing metrics

License metrics for the software are available at www.qlik.com/license-terms.

8 Troubleshooting

This section describes problems that can occur related to installation of Qlik Sense. The possible causes are described and you are presented with actions to solve the problems.

8.4 Cannot access the hub or the QMC directly after installation

Possible cause

The Qlik Sense services are started automatically delayed. This means it can take a while for them all to start after the installation.

Proposed action

Check that the services have started and that the ports are available.

Do the following:

1. Open the Task Manager in Windows and check that all Qlik Sense services have started.
2. Check that the ports needed by Qlik Sense are available.

See: Plan and deploy Qlik Sense

8.5 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.



The services are started automatically delayed. This means it can take a while for them all to start after the installation.

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.
5. 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. [Microsoft Knowledge Base: 884495](#)



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.

6. If the actions above cannot remedy the problem, you need to uninstall and reinstall Qlik Sense.

8.6 Anti-virus software scanning affects the performance of Qlik Sense

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`
- Any additional folder path configured for storing QVF files
- All executables under `%ProgramFiles%\Qlik\Sense`

8.7 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.

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

8.8 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**".

8.9 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.

8.10 Unable to upgrade, reinstall, or add a rim node due to password validation failure

Possible cause

When you install Qlik Sense with the setup program and choose to install a local database, you also create a database user (*qliksenserepository*) and a password. If you previously installed Qlik Sense with synchronized persistence then the database user will have a randomly generated password.

When you upgrade, reinstall, or add a rim node to your installation you need to use this password again. If you did not create a super user password when you installed PostgreSQL or cannot remember the database user password, then you cannot continue upgrade, reinstall, or add a rim node unless you change this password.

Proposed action

Change the database user password either using the command line or the *pgAdmin* tool.

Do the following:

1. Stop the Qlik Sense Repository Database service if it is running.
2. In PostgreSQL, change the authentication mode in the configuration settings to allow the password to be changed.
To do this, edit *ProgramData\Qlik\Sense\Repository\PostgreSQL\<database version>\pg_hba.conf* client authentication and change the address to 127.0.0.1/32 or ::1/128 to TRUST.
3. Start the Qlik Sense Repository Database.
4. Change the password.
To change the password using PostgreSQL command line:
 - a. Open a command prompt window.
 - b. Connect to the database using `psql: psql.exe -p 4432 -U postgres`.
 - c. Use the following sql command to set the new user password: `ALTER USER qliksenserepository WITH PASSWORD '<newpassword>'`. This is either *qliksenserepository* or the user you set manually during the first installation of PostgreSQL.
 - d. Stop the Qlik Sense Repository Database service.
 - e. Revert the `pg_hba.conf` localhost method back to md5.
 - f. In **Services**, start the **Qlik Sense Repository Database** service.
To change the password using the *pgAdmin* tool:
 - a. Run *pgAdmin* and connect to the database. Note you will not be prompted for a password since you are in trust mode.
 - b. In **Login Roles**, open the **Properties** window, and scroll down to the *qliksenserepository* database user.
 - c. Click the **Definition** tab, and enter a password.
5. Update the connection string for the Qlik Sense Repository Service using the **Connection String editor**.
To do this:
 - a. On your Qlik Sense installation, to open the **Connection String editor**, navigate to `C:\Program Files\Qlik\Sense\Repository\Util\ConnectionStringEditor` and double-click the executable file.
 - b. In the **ConnectionString editor**, click **Read** to see the encrypted connection string.
 - c. Update the connection string credentials with `name="QSR"` with your new repository database password.
 - d. Click **Save value above in config file encrypted** to save your changes.
6. Start the services.



In PostgreSQL 9.6 the pgAdmin tool is not included in the installation. If you want to change the password using the pgAdmin tool download it from the PostgreSQL website.

You can now continue to upgrade, reinstall, or add a rim node to your Qlik Sense installation.