Deploying QlikView

QlikView®
September 2026
Copyright © 1993-2021 QlikTech International AB. All rights reserved.

© 2026 QlikTech International AB. All rights reserved. All company and/or product names may be trade names, trademarks and/or registered trademarks of the respective owners with which they are associated.

Contents

1 Introduction ... 6
1.1 Plan and install ... 6
Planning your Deployments ... 6
Installing QlikView Server ... 6
1.2 Backup and upgrade ... 6
Backup and Restore QlikView ... 6
Upgrade and migrate your installation ... 6
1 System Requirements for QlikView Server ... 7
1.3 QlikViewPublisher ... 8
1.4 Qlik NPrinting compatibility ... 8
1.5 Browser Support ... 8
2 Planning QlikView Deployments ... 10
2.1 Architecture ... 10
QlikView Server ... 11
QlikView Web Server ... 13
QlikView Directory Service Connector ... 14
QlikView Management Service ... 14
QlikView Distribution Service ... 14
Reload Engine ... 15
Qlik License Service ... 15
Documents, Data, and Tasks ... 15
Ports ... 19
Services ... 21
Settings and Configuration ... 29
Logs ... 29
2.2 Deployment ... 49
Building a Farm ... 49
Clustering QlikView Servers ... 53
Clustering QlikView Publisher ... 64
QlikView Server Extensions ... 77
Configuring IIS for Custom Users ... 78
QlikView Triggering EDX Enabled Tasks ... 80
Cleaning and converting the shared files ... 82
IPv6 configurations ... 88
2.3 Logs and error codes ... 89
Logging from QlikView Server ... 89
Session Log ... 89
Performance Log ... 91
Server-side Extension Log ... 94
Event Log ... 95
End-user Audit Log ... 96
Manager Audit Log ... 100
Task Performance Summary ... 101
Reload performance log ... 102
QIX performance log ... 103
3 QlikView Installation ... 106
3.1 Installing QlikView Server ... 106

Before Installing QlikView Server ... 106
Setup Procedure ... 106
Logging the Installation ... 108
Obtaining the MSI package ... 108
Completing the Installation ... 108
3.2 Downloading installation files ... 111
3.3 Configuring servers with digital certificates ... 111
Configuring security ... 111
Adding QlikView services ... 113
Updating certificates ... 115
Service failure due to undecryptable data ... 116
3.4 Silent Installation ... 117
Settings ... 118
Dialogs ... 119
Additional Dialogs ... 126
MST ... 127
Silent uninstallation ... 127
3.5 Configuring a proxy for Qlik License Service communication in QlikView Server ... 128
3.6 Configuring preferred cipher suites for QlikView Server ... 129
3.7 Deploying MSI Packages with Group Policies ... 130
General ... 130
Deploying the MSI Package ... 131
Step-by-step Guide ... 132
4 QlikView Upgrades and Updates ... 139
4.1 Maintenance contract on upgrade ... 139
4.2 Upgrading and migrating QlikView Server ... 140
Requirements ... 140
Best practices ... 140
Upgrading on the same machine ... 141
Upgrading to a different machine ... 142
Upgrading and migrating QlikView Server from 11.20 to November 2017 or later ... 145
5 Backup and Restore QlikView ... 149
5.1 Backup and upgrade preparation ... 149
Backing up files ... 149
Backing up custom content ... 153
Multi-server deployments ... 154
5.2 Backing up and restoring certificates ... 155
Backing up certificates ... 155
Restoring certificates ... 156
Removing certificates ... 157
Configuration files ... 158
Using Microsoft Management Console ... 158
6 Security ... 160
6.1 Certificates ... 160
Certificate Trust ... 161
6.2 Protection of the Platform ... 166
Functionality ... 166

Special Accounts ... 166
Communication ... 166
6.3 Authentication ... 167
Authentication when Using QlikView Server in a Windows User Environment ... 168
Authentication with a QlikView Server Using an Existing Single Sign-on Software Package 169
Authentication Using neither IWA nor Single Sign-on Software ... 170
QlikView Server Authentication Using Custom Users ... 171
6.4 Authorization ... 172
Document Level Authorization ... 173
Data Level Authorization ... 173
6.5 QVD Encryption ... 175
Encryption certificates overview ... 175
Using QVD encryption ... 176
Enabling QVD encryption ... 176
Managing encryption certificates ... 177
Creating encryption certificates using Windows PowerShell ... 178
Exporting encryption certificates using Windows PowerShell ... 180
Backing up encryption certificates using Microsoft Management Console ... 181
Importing encryption certificates using Windows PowerShell ... 183
Restoring encryption certificates using Microsoft Management Console ... 184
7 Licensing QlikView ... 186
7.1 Overview ... 186
7.2 Unified license ... 186
7.3 QlikView Server license ... 186
User-based and capacity-based licenses ... 186
Access types ... 187
CALs ... 187
Restrictions ... 187
Professional and Analyzer access dynamic assignment ... 187
QlikView Server signed key ... 188
QlikView Server license key ... 188
7.4 QlikView Publisher license ... 188
7.5 QlikView Desktop ... 188
7.6 Configure Professional and Analyzer access in QlikView Server ... 189
Restrictions ... 189
Activating the Professional and Analyzer Users license ... 189
Allocating Professional and Analyzer access ... 190
Sharing users across deployments ... 192
7.7 OEM ... 192
General ... 192
Detailed Function Description ... 193

1 Introduction

In this guide you will find information on how to plan and deploy QlikView, including installation advice.

QlikView Server is a platform for hosting and sharing QlikView information over an intranet or the Internet. QlikView Server connects users, client types, documents, and objects within a secure environment.

QlikView Publisher manages content, access, and distribution. By reducing data, each user can be presented with tailored information. The QlikView Publisher service and user interface are fully integrated into QlikView Server and QlikView Management Console (QMC).

1.1 Plan and install

Planning your Deployments

Learn how to plan the deployment of a QlikView site by finding out what is required, in terms of architecture, deployment scenarios, security aspects, logging and licensing.

Installing QlikView Server

Follow this section to install a QlikView site to make it operational.

1.2 Backup and upgrade

Backup and Restore QlikView

In this section, you can read about how to create a complete backup of your QlikView Server installation. Here, you also find a dedicated documentation on how to backup and restore certificates.

Upgrade and migrate your installation

In this section, you find information on how to upgrade QlikView Server to the latest release. Here you can also find information on how to migrate a QlikView Server deployment to a different machine or cluster of machines.

1 System Requirements for QlikView Server

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

System requirements

Component	Requirements
Platforms *	• Microsoft Windows Server 2016 • Microsoft Windows Server 2019 • Microsoft Windows Server 2022 • Microsoft Windows Server 2025 For development and testing purposes only: • Microsoft Windows 11
Processors (CPUs)	Multi-core x64 compatible processors
Memory	4 GB minimum. Depending on data volumes more may be required. QlikView is an in-memory analysis technology; memory requirements for QlikView products are directly related to the quantity of data being analyzed.
Disk space	900 MB total required to install
Security	• Microsoft Active Directory (NTLM or Kerberos) • Local Windows user accounts (NTLM) • Third-party security (requires QlikView Server Enterprise Edition)
Web server	• QlikView web server • Microsoft IIS 10
.NET framework	4.8 or higher
Internet protocol	• IPv4 • IPv6 • Dual stack (IPv4 and IPv6)
Qlik Sense compatibility	It is not possible to install QlikView Server on a machine with Qlik Sense Enterprise already installed. It is possible to publish QlikView apps on Qlik Sense Enterprise SaaS.
TLS	TLS 1.2 or higher is required (TLS 1.3 where supported by the operating system)

* Provided there is Standard manufacturer support for the platform.

License activations request access to the Qlik Licensing Service. Open port 443 and allow outbound calls to license.qlikcloud.com.
Use of a proxy is supported. For more information about setting up a proxy service in Windows, see Configuring a proxy for Qlik License Service communication in QlikView Server.

1.3 QlikViewPublisher

QlikViewPublisher is an additional licensed module of QlikView Server. It is installed by applying a license to a QlikView Server.

Publisher requirements

Component	Requirements
Repository database	• Native XML • SQL Server (any version supported by Microsoft)

1.4 Qlik NPrinting compatibility

See the ≤ Release Notes on Qlik Community for QlikView and Qlik NPrinting compatibility details.

QlikView Desktop is required for QlikView connection with Qlik NPrinting and must be installed on each Qlik NPrinting Engine computer.

If using server or cluster connections, QlikView Server and QlikView Desktop must be on the same version.

For more information, see Connecting Qlik NPrinting with QlikView.

1.5 Browser Support

Browsers supported

Browser	AccessPoint QlikView Portal	QlikView Plug-in	QlikView Ajax Client	QlikView Management Console
Microsoft Edge (latest version for Microsoft Windows)	√	√ (running in IE mode)	√	√
Microsoft Edge (latest version for iOS devices and Android devices)	√	X	√	X
Mozilla Firefox (latest version)	√	X	√	√
Apple Safari (the last 3 major versions)	√	X	√	X

Browser	AccessPoint QlikView Portal	QlikView Plug-in	QlikView Ajax Client	QlikView Management Console
Apple Mobile Safari (latest 3 versions)	√	X	√	X
Google Chrome (latest version for Microsoft Windows, Apple Mac, and Android devices)	√	X	√	√

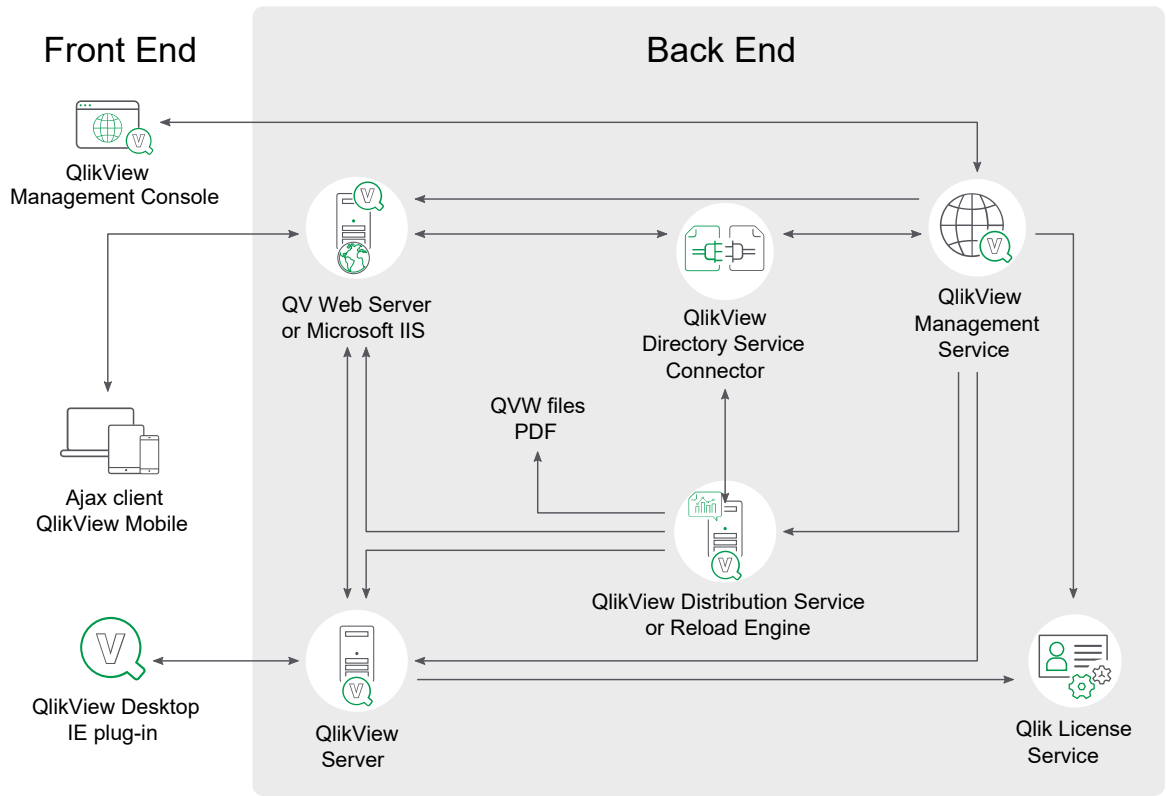
2 Planning QlikView Deployments

This section provides details on the QlikView architecture, deployment, security, logging and licensing.

> *If Microsoft IIS is to be used as web server, it must be installed prior to QlikView Server.*

2.1 Architecture

The overall architecture of a QlikView installation reflects the separation of roles.



QlikView Server architecture with one instance of each service

Front End

The front end is where end users interact with the documents and data that they are authorized to see via QlikView Server. The front end contains the QlikView user documents that typically have been created via QlikView Publisher (QlikView Distribution Service with Publisher license) at the back end. All communications between the client and server take place here and QlikView Server is fully responsible for the client authorization.

The front end relies on infrastructure resources (for example, Windows-based file share for clustering).

> *QlikView Server currently only conforms with Windows file sharing. This means that storage must be owned, governed, and shared by a Windows operating system instance (typically accessed using a path like \\<servername>\<share>).*

> *QlikView does not support Windows Distributed File System (DFS).*

Authentication of end users is (with exception of the built-in Custom Users) handled outside QlikView.

Back End

The back end is where the QlikView source documents, created using QlikView Developer, reside. These source files contain scripts to extract data from various data sources (for example, data warehouses, Microsoft Excel® files, SAP®, and Salesforce.com®). This extraction sometimes involves intermediate files (QVD files). The main QlikView component that performs the loading and distribution at the back end is the QlikView Distribution Service.

The back end uses the infrastructure resources for clustering (for example, Windows-based file share) and may also use resources like SMTP servers and directory catalogs.

> *QlikView Server currently only conforms with Windows file sharing. This means that storage must be owned, governed, and shared by a Windows operating system instance (typically accessed using a path like \\<servername>\<share>).*

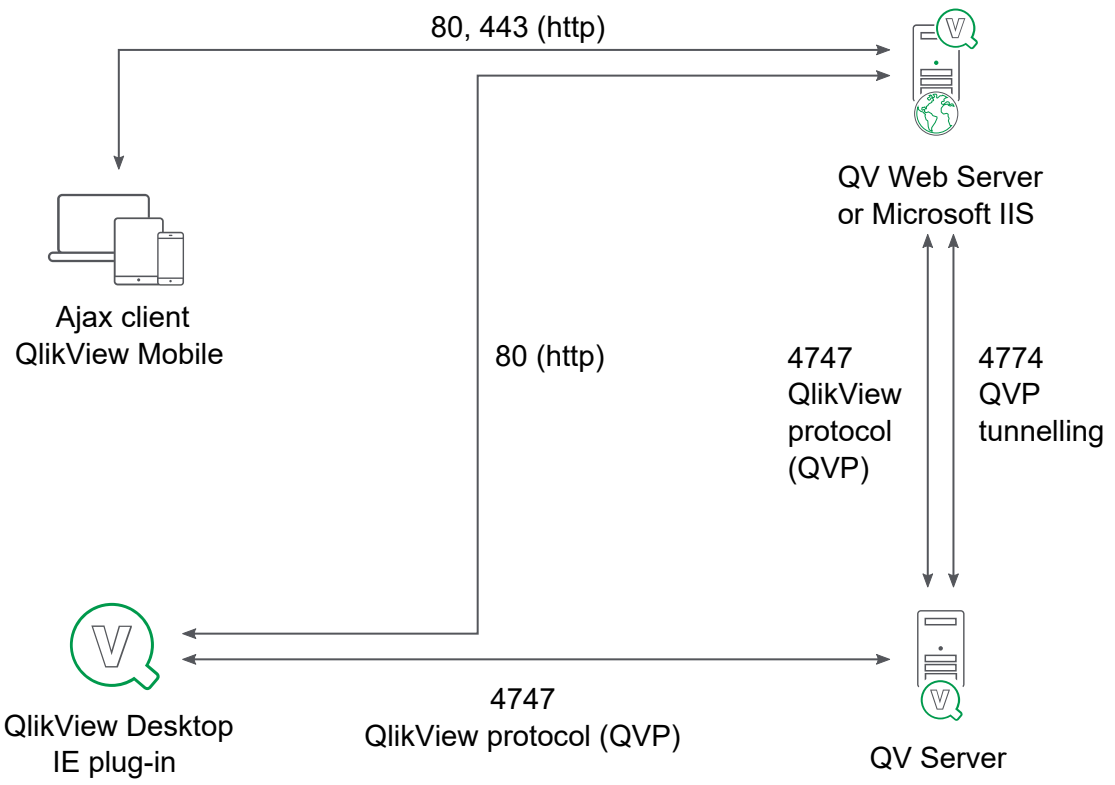
> *QlikView does not support Windows Distributed File System (DFS).*

QlikView Server

The number of servers (clustered or not) within an installation is only limited by the license. It is, however, not feasible to run more than one QlikView Server (QVS) process per server (physical or virtual). QlikView Server is designed to make the most of the resources available to it. Notably the QlikView Server keeps as many calculation results as possible cached in memory to keep the response times to a minimum.

QlikView Server – Client Communication

The QlikView Server – client communication architecture requires three primary processes, which must be able to communicate with each other in a consistent and secure manner. This interaction can potentially involve multiple machines and multiple network connections, as well as other subordinate processes.



QlikView Server - client communication

The three primary processes are described below.

Client communication processes

Process	Description
QlikView Server (QVS)	Provides QlikView functionality to the client. The machine that hosts this service must be running a Microsoft Windows operating system.
Client	Runs in a web browser or an application shell that provides a container for the client code. The client communicates with QVS either directly or through the web server to provide the QlikView interface and functionality to the end user.
Web server	Runs an http server, which can be used to serve html web pages to the client, assists with authentication of the user, and enables communication between the client and QVS.

With the exception of Custom Users, the authentication of client users is done outside QlikView using, for example, Windows authentication.

The protocols defined for client communication with QlikView Server are listed below.

Client communication protocols

Protocol	Description
QlikView Protocol (QVP)	Encrypted, binary, and TCP-based; communicates directly with QVS on port 4747.
QVPX	XML-based; communicates with the QVS using http/https through a web server.

Windows clients (*.exe*/*.ocx*) communicate directly with QlikView Server using QVP on port 4747. These clients do not require a web server to establish and maintain a connection with QlikView Server.

The AJAX client and mobile clients do not communicate directly with QlikView Server. They establish and maintain a connection using the QVPX protocol through a web server, QlikView Web Server (QVWS) or Microsoft IIS. This is normally done using port 80 (http). The web server, in turn, communicates with QVS using the QVPX2 protocol on port 4747.

The default installation settings for QVS use QVWS, not Microsoft IIS. QVWS shares port 80 with IIS on Windows 7 (and later) and Windows Server 2008 (and later).

QlikView Server access to User Document

For a user to open a document, it is required that:

- There is a Client Access License (CAL) for the user
- The user has access to the document

The user documents are always read by QlikView Server (QVS) and thus technically only need to be readable by the account running QVS. The access rights are either stored in the ACL list of the document (when QVS runs in NTFS mode) or in the *.META* file (when QVS runs in Document Metadata Service – that is, DMS – mode). These settings are part of the distribution from the back end.

Items (for example, layout, reports, bookmarks, annotations, and input field values) created by end users are stored in shared files. Shared files are not replaced by the distribution from the back end.

QlikView Web Server

QlikView Web Server (QVWS) is included as part of the QlikView Server installation. The QlikView Web Server can act as a standalone service to fulfill the need of many QlikView Server installations.

As an alternative, a Microsoft IIS solution that provides more flexibility, additional authentication schemes, and web services for applications other than QlikView Server can be deployed. When Microsoft IIS is used, a special service, QlikView Settings Service, that handles management calls is installed.

Other web servers can be used in a QVS environment, but at some point the traffic targeting QVS has to go through either QVWS or the dedicated ASPX pages on IIS.

The QlikView Web Server component (either QVWS or IIS-based) performs several tasks:

- Handles the AccessPoint back end
- Transforms/routes traffic between stateless http and to/from the session-based communication with QVS
- Handles load balancing of QVS clusters
- Serves static content (optional)
- Handles authorization of Windows-authenticated users
- Handles authentication of Custom Users (optional)
- Handles group resolution through Windows or Directory Service Connector (DSC) (optional)

QlikView Server Tunnel

If the QVS communication port (4747) is blocked in the network firewall, Windows clients attempt to re-route their connection through port 80 (http). This connection path must then include the QVWS, or be installed on Microsoft IIS, so that QVS tunnel communication can be established.

QlikView Directory Service Connector

The Directory Service Connector (DSC) is responsible for retrieving user information related to end users from a variety of sources, including (but not limited to) Active Directory, LDAP, ODBC, and Custom Users.

The web server uses DSC for group resolution, the QlikView Distribution Service uses it to look up e-mail addresses or UIDs during distribution, and the Management Service uses it to help the administrator find users and groups.

QlikView Management Service

The QlikView Management Service is the entry point for all management, both through QlikView Management Console and the QlikView APIs.

The QlikView Management Service (QMS) keeps settings in a database of its own, the QVPR. The QVPR is by default stored as XML files – an alternative is storing the settings in an SQL database.

> *All QlikView servers must have the same regional settings. Different regional setting may cause errors when loading QlikView XML reference files.*

An installation can only have a single instance of QMS active. Active/passive failover should be used for redundancy. Note that no other service needs QMS to be running.

QlikView Distribution Service

In a QlikView installation using QlikView Distribution Service, both the back end and the front end are suitable for development, testing, and deployment.

The QlikView Distribution Service works with the source documents to produce:

- User documents
- *.qvw* files for distribution to a folder or via e-mail
- *.pdf* documents for distribution to a folder or via e-mail

The chain of events up to the final distribution involves one or many of the following tasks:

- Data is loaded from one or more data sources (including QVD) into one or more *.qvw* or *.qvd* files.
- A document is reduced into one or more smaller documents.
- Attributes and usage rules are added (applicable only when distributed to a QVS).

The QlikView Distribution Service performs the tasks according to defined schedules and/or as responses to events.

QlikView architecture without QlikView Distribution Service

Without QlikView Distribution Service, the QlikView architecture becomes more restrictive. All distribution and reduction facilities are removed and replaced by a reload directly on the user documents. Without QlikView Distribution Service, developers need to manually deploy the *.qvw* file behind the server.

Reload Engine

In the absence of a Publisher license connected to the QlikView Distribution Service, the Reload Engine provides a subset of the Publisher distribution services. The Reload Engine only reloads user documents and the settings are defined directly in the user documents.

> *All QlikView services must be running on the same machine for the Reload Engine to work. If you install the services on different machines (for example, the QMC, DSC, and QDS on one machine and the QVS and QVWS on another machine), the Reload Engine will not work.*

Qlik License Service

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

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

Documents, Data, and Tasks

User Documents

A user document is the document that an end user sees when accessing a document on QlikView Server (QVS). To fully identify a user document, both the QVS server/cluster and the path relative to the server have to be known. Technically, a user document consists of three files:

- QlikView document file (*.qvf* or *.qvw*) that contains the data and layout.
- *.META* file that contains:

  - AccessPoint attributes
  - Pre-load options
  - Authorization (Document Metadata Service – that is, DMS – mode only)
- Shared file (*.Shared* or *.TShared*, see below)

> *If the user document is distributed by the QlikView Distribution Service, both the QlikView document file and the data in the .META file are overwritten.*

The access to user documents is controlled by QlikView Server.

Shared Files

There are multiple objects available for user collaboration and sharing through QlikView Server:

- Bookmarks
- Sheet objects, including charts
- Reports
- Annotations

Each of these objects may be defined as a user object, available to authenticated users, regardless of access method or location, or a shared object, available to all users of the document through QVS.

The objects are configured and managed using QlikView Management Console (QMC).

Once QVS is enabled for server objects, any of the QVS object settings are checked, and the document is opened in QVS, a special database file is created and maintained in the same location as the QlikView document. The file has the same name as the QlikView document, but a shared file extension (*.Shared* or *.TShared*).

Example:

- QlikView document: *Presidents.qvw*
- QVS share file: *Presidents.qvw.TShared*

If the name of the QlikView document is changed, the shared file has to be manually renamed to match before opening the renamed QlikView document in QVS. This preserves the shared objects attached to the document.

When updating a Server object, report, bookmark, or input field data, the file is exclusively locked. Making a selection or simply activating the object does not lock the file and any number of servers can read the file at the same time. A partial lock is implemented so that different sections of the file may be updated simultaneously by different servers in a cluster.

The file is read once when the server opens the document, but it is not read again unless there are changes. All sessions share the same internal copy of the shared file (that is, opening a session generally does not require the file to be read from disk).

The server objects can be managed (for example, change of ownership or delete) on the **Documents**>**User Documents**>**Server**>**Server Objects** tab in QMC.

Source Data

Source data is any external data used to populate the data within a QlikView document file. The source data is loaded to the QlikView document at reload time, which can be done:

1. Through the QlikView Distribution Service
2. Through the Reload Engine
3. Manually by the developer

Access to source data is not required for end users to use the QlikView document through QVS once the QlikView document file is populated.

Source Documents

Source documents are only applicable when a Publisher license is applied. Most source documents originate from a developer, others are created by the QlikView Distribution Service as part of the distribution process. QlikView Data files (QVD) can also be created as part of the distribution process as an intermediate step. A QVD file is a table of data stored in format that is optimized for speed when read by QlikView.

The access to source documents is governed by NTFS.

Tasks

Tasks can be used to perform a wide variety of operations and be chained together in any arbitrary pattern. The starting point when describing tasks is the transformation of a source document into a user document.

Transforming Source Document into User Document

The transformation starts with a source document and ends in one or many user documents.

Source

A task is always tied to a source document, so the source is given.

Layout

The source document contains the layout, which is copied unchanged all the way to the user documents. The server side layout is associated with the user document and is also unchanged.

Reload

The data can be:

- Used as stored in the document (that is, no reload)
- Partly reloaded from the source (that is, require script preparation)
- Fully reloaded from the source, discarding any old data
- Reloaded in parts by use of “Script Parameters” (which require script preparation)

Reduce

The document can be reduced after reload. The reduction can either reduce the input into a smaller document (simple reduce) or split it up into several smaller documents (loop and reduce).

The reduction is based on a selection, either done directly in QMC or using bookmarks.

Distribution

Distribution requires a QlikView Publisher license.

The destination is defined as:

- A list of users and a folder on a QlikView Server
- A list of users and a folder in the file system
- A list of users (assuming their e-mail addresses are known)

> *"Loop and distribute" must be used, if different content is to be distributed to different users. If not, the same document (or documents) is distributed to all.*

Information

Information can be associated with the document as part of the distribution to a server. The information is not moved with the document, if it is distributed to another location. The information is used in QlikView AccessPoint.

The following information can be associated with the document:

- Description
- Category
- Arbitrary name value pairs

Server Settings

The settings for the document are distributed to a server. The settings are not moved with the document, if it is distributed to another location. The settings are enforced by QlikView Server.

Authorization enforced by the server (equal to all servers):

- The users authorized to create server objects
- The users authorized to download the document
- The users authorized to print and export the document to Microsoft Excel

Preferences applied by QlikView AccessPoint (equal to all servers):

- QlikView plugin is recommended
- Mobile client is recommended
- AJAX client is recommended

Performance enforced by the server (equal to all servers):

- Audit logging
- Maximum open sessions
- Document timeout
- Session timeout

Availability (per server):

- Never
- On-demand
- Pre-loaded

Ports

QlikView uses ports to communicate between web browsers (users) and servers, and between different services in single or multi-node deployments.

The following table provides an overview of the ports used in a QlikView deployment.

QlikViewports

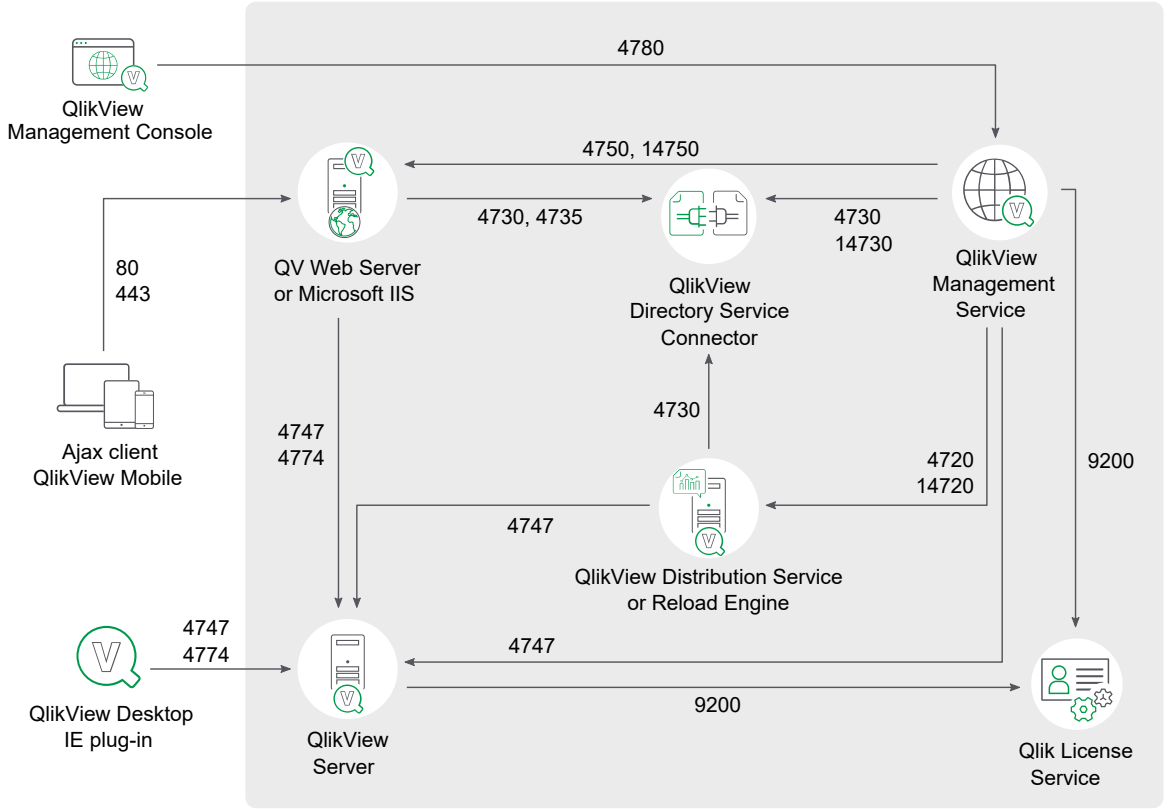
-	Component	Inbound	Outbound
	QlikView Server (QVS)	4747 (QVP: QlikView Protocol) 4774 (QVP Tunnelling) 14747 (Cluster users SOAP API) 4749 (SSL)	14747 (Cluster broadcast)
	QlikView Web Server (QVWS)	80 (HTTP) 443 (HTTPS) 4750 (SOAP API) 14750 (Certificate)	4730 (DSC SOAP API) 4735 (DSC custom users SOAP API) 4747 (QVS QVP) 4774 (QVS QVP tunnelling)
	QlikView Distribution Service (QDS)	4720 (SOAP API) 14720 (Certificate)	4730 (DSC SOAP API) 4747 (QVS QVP)
	QlikView Directory Service Connector (DSC)	4730 (SOAP API) 14730 (Certificate) 4735 (Custom users SOAP API)	-

-	Component	Inbound	Outbound
	QlikView Management Service (QMS)	4780 (HTTP) 4799 (SOAP API)	4747 (QVS QVP) 4750 (QVWS SOAP API) 14750 (QVWS Certificate) 4720 (QDS SOAP API) 14720 (QDS Certificate) 4730 (DSC SOAP API) 14730 (DSC Certificate) 4735 (DSC custom users) 4799 (Remote QMS SOAP API)
	Qlik License Service	9200	443 (HTTPS)

The following example shows the ports used to connect the different QlikView services. In a QlikView deployment, all these services can be installed on the same server (single-node deployment). Alternatively, you can decide to set up a multi-node deployment, and install different services on different servers. For further information about QlikView architecture and deployments, see *Architecture (page 10)* and *Deployment (page 49)* pages.

The following diagram shows the ports used for connecting the different services in a QlikView Server deployment (QV: QlikView).

QlikView Server Ports Communication



Services

This chapter describes the QlikView Server/Publisher components in detail.

> *The account that is used to run the QlikView services must have local administrator privileges.*

To learn how to set up your QlikView services, see Setup.

QlikView Server Load Sharing (Clustering)

Overview

Clustering overview

Executable	*%ProgramFiles%\QlikView\Server\QVS.exe*
Data	*%ProgramData%\QlikTech\QlikViewServer*
Listens to	QVP: 4747; QVP (tunneling): 4774; Broadcast: 14747; SNMP: 161
Uses/Controls	-
Used by	QDS, QMS, QVWS, QlikView Desktop/QlikView plugin/OCX

Files

Settings and Configuration

Configuration files

File	Description
Settings.ini	Stores the QlikView Server (QVS) settings. Manual changes in this file require restart of QVS. This file is always stored in the “Data” folder.

Cluster

QVS uses *.pgo* files to coordinate a cluster. The files are stored in the “Data” folder.

Cluster files

File	Description
BorrowedCalData.pgo	Keeps track of borrowed Client Access Licenses (CALs).
CalData.pgo	Keeps track of CALs.
IniData.pgo	Coordinated version of *Settings.ini*.
ServerCounters.pgo	Keeps track of statistics.
TicketData.pgo	Keeps track of tickets.

Logs

The logs are kept one per node in the cluster. The log files are stored in the “Data” folder by default.

Log files

File	Description
Events_<computer_name>.log	Event log.
Performance_<computer_name>.log	Performance log.
Sessions_<computer_name>.log	Session log.

Special Folders

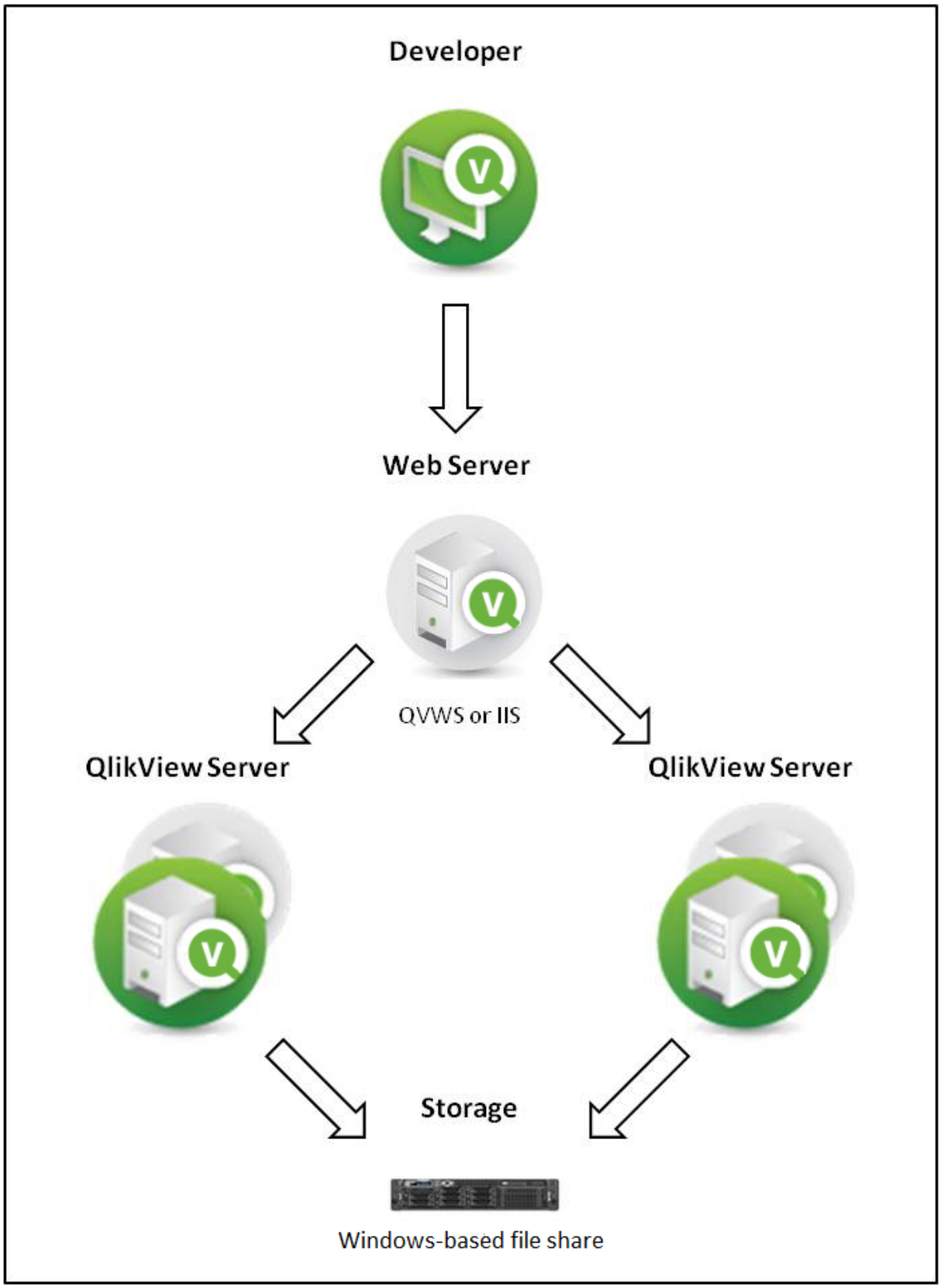
The special folders are stored in the “Data” folder.

Special folders

Folder	Description
Extensions	*The Extensions folder has to be created manually.* By default, QVS looks for extensions in this folder. Extension objects are located in *Extensions\Objects* and document extensions are located in *Extensions\Document*. Use QlikView Management Console (QMC) to manage all extensions in one place in case of a cluster.
Temp	By default, QVS puts temporary files in this folder (for example, when exporting using the AJAX client, a temporary file is created in the folder).

Load Sharing (Clustering)

All clustering requires a cluster-enabled QlikView Server license. QlikView Server supports load sharing of documents across multiple machines. This sharing includes the ability to share in real time, information about server objects, automated document loading, and user license CALs. Special licensing is available to enable multiple server instances share the same license number.



Load sharing using QlikView Web Server

To use load sharing between multiple QVSs, all document and support files must be shared between the servers. In other words, all servers should point to the same physical location for the files. QVS creates and maintains additional files to store load sharing data. These files have a Persistent Group Object (*.pgo*) file type extension and are located in the "Data" folder. These files are locked when QVS is running. The different *.pgo* files contain information on borrowed CALs, CALs in use, server settings, and ticket data.

Operating system load balance or failover configurations are external to the QVS load sharing configuration, and QVS has no control over those systems.

Server configuration settings are shared between all clustered QVSs and can be maintained through QMC connected to any of the clustered QVSs. Performance of a particular QVS system can be monitored through QMC by connecting to that system. The load balancing settings, that is, which QVS the client should be directed to, are stored in QlikView Web Server (QVWS).

Document-related meta data is shared via *.meta* files (one per document). This data is often referred to as Document Metadata Service (DMS) data. Since DMS data is shared among the QVSs, any automated document load procedures are performed on all servers. DMS authorization is also shared among all clustered QVSs.

QlikView Distribution Service

Overview

QlikView Distribution Service overview

Executable	*%ProgramFiles%\QlikView\Distribution Service\QVDistributionService.exe*
Data	*%ProgramData%\QlikTech\DistributionService*
Listens to	HTTP: 4720; SNMP: 4721
Uses/Controls	DSC, QVS, QVB
Used by	QMS

After restarting the machine, the Windows event log may contain a message that the QlikView Distribution Service (QDS) failed to start in a timely manner, even though it started successfully. This is because the QDS initialization phase is longer than the Windows timeout period (30 seconds by default). To avoid the event log message, either change the Windows timeout period or configure QDS to depend on another late starting service to make QDS start up during a less busy period.

Files

The QlikView Distribution Service (QDS) files can be divided into three groups based on main purpose. All files, with the exception of *DistributionGroupDefinition.xml*, are stored in the QDS "Data" folder.

In a clustered setup, all QDSs must share the same program folder. This is solved by the file *config_<server_name>.xml*, which contains the program data path to use.

You might see other files in your QDS Data folder that are not listed below. Any file not listed below is required for system use only. They should not be altered or removed.

Settings and Configuration

The files listed below are local copies of the information stored in QVPR.

Settings and configuration files

File	Description
service_key.txt	The service key that is used to make calls to the QMS API.
Config_servername.xml	This file lists all changes made to the default configuration of the *QVDistributionService.exe.config* file. In a QDS clustered environment, this file exists on each node.
Configuration.xml	Configuration file for the service.
Tasks\Task_<GUID>.xml	The actual tasks. Note that deleted tasks are not automatically removed (due to support issue analysis).
Triggers\Triggers_<GUID>.xml	The actual triggers. Note that deleted triggers are not automatically removed (due to support issue analysis).
MasterConfigurationNotification.xml	A list of configuration notification files. Used to keep QDS in sync and to notify QDS nodes of configuration changes.
MasterTaskNotification.xml	A list of task notification files. Used to keep QDS in sync and to notify QDS nodes of task changes.
MasterTaskExecutionNotification.xml	A list of task execution notification files. Used to keep QDS in sync and to notify QDS nodes of task execution changes.
MasterTriggerNotification.xml	A list of trigger notification files. Used to keep QDS in sync and to notify QDS nodes of changes to triggering events.
TaskDetails.xml	A list of the available tasks in the *Tasks* folder. Used to synchronize the files in that folder with QVPR.

TriggerDetails.xml	A list of the available triggers in the *Triggers* folder. Used to synchronize the files in that folder with QVPR.
DistributionGroupDefinition.xml	Configuration file for Distribution Groups Location: *%ProgramData%\QlikTech\ManagementService\DistributionGroups*

Cluster

Cluster files

File	Description
LoadBalancer.xml	Used by QDS to decide which node (in a cluster) should run a task.
NodeInformation.xml	Contains all other QDS node data that is not used by the load balancer.

Logs

Log files

File	Description
TaskResults\TaskResult_<GUID>.xml	Latest result of the task identified by the GUID.
TaskLogIndex\TaskLogIndex_<GUID>.xml	This is just for lookup (one file per task), pointing to the actual log.
EdxResults\EdxResult_<GUID>.xml	Until the task is completed, this file contains the current status of the EDX task. When the execution is finished, it contains the result (success/fail) and the task started as a result (if any).
<node-nr>\Log\<Date>.txt	General QDS event and error log.
<node-nr>\Log\Cluster_<Date>.txt	Synchronization log.
<node-nr>\Log\LoadBalancer_<Date>.txt	Load balancing log.
<node-nr>\Log\Root_<Date>.txt	QDS event log.
<node-nr>\Log\WebService_<Date>.txt	QDS event log.
<node-nr>\Log\Workorder_<Date>.txt	QDS event log.
<node-nr>\Log\<date>\<time> - <task name>\Tasklog.txt	QDS task event log.
<node-nr>\Log\<date>\<time> - <task name>\DistributionReport.xml	The distribution related to the task (only exists for distribution tasks).

Changing the storage time of log files

By default, log files are stored for 30 days and then are deleted from the Application Data Folder, by default *C:\ProgramData\QlikTech\DistributionService*.

You can change the storage time of log files within the *QVDistributionService.exe.config* file.

> *It is recommended that you make a copy of QVDistributionService.exe.config as a backup before you edit QVDistributionService.exe.config.*

Do the following:

1. Open Windows Services.
2. Stop the QlikView Distribution Service by right-clicking the service and clicking **Stop**.
3. Close Windows Services.
4. Browse to *C:\Program Files\QlikView\DistributionService* and open QVDistributionService.exe.config with a text editing program.
5. Locate `<add key="NbrOfDaysToKeepQDSLogs" value="30" />` and add the number of days reports are to be stored as the value.
6. Save and close the file.
7. Open Windows Services.
8. Restart the QlikView Distribution Service by right-clicking the service and clicking Start.
9. Close Windows Services.

QlikView Batch

Overview

QlikView Batch Overview

Executable	*%ProgramFiles%\QlikView\Distribution Service\qvb.exe*
Data	-
Listens to	COM
Uses/Controls	-
Used by	QDS

> *QlikView Batch (QVB) does not support graphical or user input objects. This means that QVB cannot reload documents that, for example, contain scripts that require user input.*

Files

Settings and Configuration

Settings and configuration files

File	Description
Settings.ini	Used to store settings.

Logs

Log files

File	Description
<document_name>.log	Reload log that is placed together with the reloaded document.

QlikView Publisher Repository

Overview

Publisher Repository overview

Executable	-
Data	*%ProgramData%\QlikTech\ManagementService\QVPR*
Listens to	-
Uses/Controls	-
Used by	QMS

Files

By default, QlikView Publisher Repository (QVPR) is a set of XML files. These files are backed up as *.zip* files in *%ProgramData%\QlikTech\ManagementService\QVPR\Backups*.

Security Groups

When installing QlikView Server/Publisher, a couple of security groups are created.

The QlikView Server/Publisher services must run under an account that is member of the security group QlikView Administrators. Users connecting to QMC must be part of this group. Anyone connecting to a remote service must also be member of QlikView Administrators.

The users connecting through the API must be members of the QlikView Management API security group. The group is not created during the installation and has to be added (and populated, for example, with the members of the QlikView Administrators group) manually. A membership in this group is required to import tasks from another QlikView Server/Publisher.

The QlikView EDX security group is not created during the installation and has to be added (and populated) manually in order for users to run EDX tasks.

Document Administrators

To delegate the responsibility of creating tasks to people not part of the QlikView Administrators group, users can be appointed document administrators. The document administrators are only allowed to access the tabs in QMC that are related to either user documents or source documents.

> *The use of document administrators requires a QlikView Publisher license.*

For more information on how to appoint document administrators, see the QMC online help.

Configuration Files

> *Use QMC to set the parameters described in this section, since modifying the configuration files directly may cause problems.*

QlikView Management Service - QVManagementService.exe.config

In a default installation, this file is located in *%ProgramFiles%\QlikView\Management Service*. The file has a number of automatically generated tags that should not be modified, but the settings listed below can be modified.

QlikView Management Service settings

Setting	Description
ApplicationDataFolder	Folder where the log folder and all other files/folders are created. The default value is *%ProgramData%\QlikTech\ManagementService*. This folder is where the XML version of QVPR and the LEF information are stored.
UseHTTPS	`True` = Communication runs over https. A certificate for the web site is needed to enable this setting.
Trace	Used for debug logging.
QMSBackendListenPort	Port that the back end management service listens to. The default value is 4799.
QMSFrontendWebServicePort	Port that the front end management service listens to. The default value is 4780.
MaxLogRecords	Maximum number of log records that should be retrieved for a task.
EnableAuditLogging	`True` = Track a) changes on tasks and settings made in the system, b) who made the changes, and c) when the changes were made.
AuditLogFolder	Path to the folder where the audit logs are saved.
AuditLogKeepMaxDays	Maximum number of days each log is saved.

Setting	Description
ServiceFailureAlertEmailAddresses	List of semicolon-separated email addresses. An email will be sent to the listed receivers in case of service failure. Adding email-addresses to the ServiceFailureAlertEmailAddresses tag activates the feature. The Management Service must be restarted for the change to take effect. This feature requires a mail server to be configured in the QMC. See Mail server setup for details. The email subject and body can be customized by using ServiceFailureAlertEmailSubject and ServiceFailureAlertEmailBody.
ServiceFailureAlertEmailBody	The plain-text body of the email that will be sent in case of service failure.
ServiceFailureAlertEmailSubject	The subject of the email that will be sent in case of service failure.

QlikView Distribution Service - QVDistributionService.exe.config

In a default installation, this file is located in *%ProgramFiles%\QlikView\Distribution Service*. The app settings tag is the part that can be modified. Some of the settings in the configuration file are described below.

QlikView Distribution Service settings

Setting	Description
ApplicationDataFolder	Folder where the log folder and all other files/folders are created. The default value is *%ProgramData%\QlikTech\DistributionService*. This folder is where the XML version of QVPR and the LEF information are stored.
WebservicePort	Port that the QlikView Distribution Service uses to communicate with. The default value is 4720.
UseHTTPS	`True` = Communication runs over https.
DSCAddress	Port that the Directory Service Connector service uses to communicate with. The default value is 4730. If the value is modified, the tag “DSCAddress” in the *QVDirectoryServiceConnector.exe.config* file has to be modified too.
DSCTimeoutSeconds	Timeout for calls to the Directory Service Connector.
DSCCacheSeconds	How long the service caches the responses from the Directory Service Connector.

Setting	Description
QlikViewEngineQuarantineTimeInms	How often a QlikView engine is allowed to start (in milliseconds).
OpenDocumentAttempts	How many tries that can be made to open a document before it is logged as an error during distribution.
DebugLog	`True` = Enable logging of memory usage and stack trace on “Error” logging.
Trace	`True` = Enable debug logging.
EnableBatchMode	Enable this setting to make batch calls to the QlikView Distribution Service.
ServiceStopGracetimeInSeconds	The time in seconds allowed for tasks running in the QlikView Distribution Service (QDS) to complete, when a request is made from the QMC to shut down the QDS. The default value is 1800.

Directory Service Connector - QVDirectorServiceConnector.exe.config

This file is by default located in *%ProgramFiles%\QlikView\Directory Service Connector\QVDirectoryServiceConnector.exe.config*. The settings most commonly modified are listed below.

Directory Service Connector settings

Setting	Description
ApplicationDataFolder	Folder where the log folder and all other files/folders are created. The default value is *%ProgramData%\QlikTech\DirectoryServiceConnector*.
WebservicePort	Port that the Directory Service Connector service uses to communicate with. The default value is 4730. If the value is modified, the tag “DSCAddress” in the *QVDistributionService.exe.config* file has to be modified too.
UseHTTPS	`True` = Communication runs over SSL/TSL instead of http. A certificate for the web site is needed to enable this setting.
PluginPath	Path where the Directory Service Connector looks for available DSP plugins. The default value is *%ProgramFiles%\QlikView\Directory Service Connector\DSPlugins*.
Trace	`True` = Enable debug logging.
DisableCompress	Enable this setting to disable compression of the http communication.

QlikView Web Server

The web server can be the built-in QlikView Web Server (QVWS) or Microsoft IIS. QVWS is installed as a Windows service during a default, complete installation of QlikView Server. When IIS is used, the same functionality is provided by a set of ASPX pages and a special support service, QlikView Settings Service

(QSS). QSS acts as the management interface for settings used by the ASPX pages.

Overview

QlikView Web Server

QlikView Web Server properties

Property	
Executable	*%ProgramFiles%\QlikView\Server\Web Server\QVWebServer.exe*
Data	*%ProgramData%\QlikTech\WebServer*
Listens to	HTTP: 80; HTTP: 4750; SNMP: 4751
Uses/Controls	DSC
Used by	Web browser clients and mobile clients

QlikView Settings Service

QlikView Settings Service properties

Property	
Executable	*%ProgramFiles%\QlikView\Server\Web Server Settings\QVWebServerSettingsService.exe*
Data	*%ProgramData%\QlikTech\WebServer*
Listens to	HTTP: 4750
Used by	QMS

Files

Settings and Configuration

Configuration files

File	Description
Config.xml	Configuration file for the service.

Logs

Log files

File	Description
Log\<date>.txt	Event and error log.

Configuring the QlikView Web Service

You may configure the web server either through the QlikView Management Console. Additional configuration can be done by editing the *config.xml* file, found in the following location:

C:\ProgramData\QlikTech\WebServer

The *config.xml* file contains the following section that is commented out to simplify the usage of common but non-default options.


```
<Config>
<DefaultUrl>http://_/</DefaultUrl>
<DefaultQvs>local</DefaultQvs>
<ConfigUrl>http://_:4750/qvws.asmx</ConfigUrl>
<TunnelUrl>/scripts/QVSTunnel.dll</TunnelUrl>
<QvsStatusUrl>/QvAjaxZfc/QvsStatus.aspx</QvsStatusUrl>
<LogLevel>Information</LogLevel>
<UseCompression>True</UseCompression>
<InstallationPath>C:\Program Files\QlikView\Server\Web Server</InstallationPath>
<QvsAuthenticationProt>Negotiate</QvsAuthenticationProt>
<QvpPort>-1</QvpPort>
<AddCluster>
<Name>local</Name>
<LoadBalancing>Random</LoadBalancing>
<AlwaysTunnel>False</AlwaysTunnel>
<AddQvs>
<Machine>localhost</Machine>
<Port>4747</Port>
<LinkMachineName>RD-CENTEST1</LinkMachineName>
<Weight>1</Weight>
</AddQvs>
</AddCluster>
<AddDSCCluster>
<CustomUserPort>-1</CustomUserPort>
<DirectoryServiceConnectorSettings>
<ID>17da91ee-c4a6-4cdb-a2fb-ab472ece659f</ID>
<Url>http://rd-centest1:4730/qtds.asmx</Url>
<Name>Default DSC</Name>
<Username>DxdCGMWfOwU=</Username>
<Password>DxdCGMWfOwU=</Password>
<LogLevel>Normal</LogLevel>
</DirectoryServiceConnectorSettings>
</AddDSCCluster>
<Authentication>
<AuthenticationLevel>Always</AuthenticationLevel>
<LoginAddress>/qlikview/login.htm</LoginAddress>
<LogoutAddress>logout.htm</LogoutAddress>
<GetTicket url="/QvAjaxZfc/GetTicket.aspx" />
<HttpAuthentication url="https://_/scripts/GetTicket.asp" scheme="Basic" />
<HttpAuthentication url="/QvAJAXZfc/Authenticate.aspx" scheme="Ntlm" />
</Authentication>
<AccessPoint>
<Path>/QvAJAXZfc/AccessPoint.aspx</Path>
<AjaxClientPath>/QvAJAXZfc/opendoc.htm</AjaxClientPath>
<PluginClientPath>/QvPlugin/opendoc.htm</PluginClientPath>
<DefaultPreferedClient>Ajax</DefaultPreferedClient>
<DefaultView>Thumbnails</DefaultView>
<DefaultPagesizeDetails>40</DefaultPagesizeDetails>
<DefaultPagesizeThumbnails>4</DefaultPagesizeThumbnails>
<HighlightNotExecutedJobs>False</HighlightNotExecutedJobs>
<HighlightThresholdMinutes>60</HighlightThresholdMinutes>
<AllowCmdUrl>False</AllowCmdUrl>
<Target />
<RespectBrowsable>True</RespectBrowsable>
</AccessPoint>
<Ajax>
```

```
<Path>/QvAJAXZfc/QvsViewClient.aspx</Path>
<Path>/QvAJAXZfc/QvsViewClient.asp</Path>
<NoCrypto>False</NoCrypto>
<ProhibitMachineId>False</ProhibitMachineId>
<Recording>False</Recording>
<AllowCmdUrl>True</AllowCmdUrl>
</Ajax>
<Web>
<Folders>
<Folder>
<Name>QlikView</Name>
<Path>C:\Program Files\QlikView\Web</Path>
</Folder>
<Folder>
<Name>QvClients</Name>
<Path>C:\ProgramFiles\QlikView\Server\QvClients</Path>
</Folder>
<Folder>
<Name>QvAJAXZfc</Name>
<Path>C:\ProgramFiles\QlikView\Server\QvClients\QvAjaxZfc</Path>
</Folder>
<Folder>
<Name>QvDesktop</Name>
<Path>C:\Program Files\QlikView\Server\QlikViewClients\QlikViewDesktop</Path>
<Folder>
<Folder>
<Name>QvPlugin</Name>
<Path>C:\Program Files\QlikView\Server\QvClients\QvPlugin</Path>
</Folder>
</Folders>
<Types>
<Type>
<Extension>.css</Extension>
<Content>text/css</Content>
</Type>
<Type>
<Extension>.htm</Extension>
<Content>text/html</Content>
</Type>
<Type>
<Extension>.html</Extension>
<Content>text/html</Content>
</Type>
<Type>
<Extension>.jpg</Extension>
<Content>image/jpg</Content>
</Type>
<Type>
<Extension>.gif</Extension>
<Content>image/gif</Content>
</Type>
<Type>
<Extension>.jar</Extension>
<Content>application/octet-stream</Content>
</Type>
<Type>
<Extension>.png</Extension>
<Content>image/png</Content>
</Type>
```

```
<Type>
<Extension>.exe</Extension>
<Content>application/octet-stream</Content>
</Type>
<Type>
<Extension>.msi</Extension>
</Type>
<Type>
<Extension>.htc</Extension>
<Content>text/xml</Content>
</Type>
<Type>
<Extension>.js</Extension>
<Content>text/javascript</Content>
</Type>
<Type>
<Extension>.xslt</Extension>
<Content>text/xml</Content>
</Type>
<Type>
<Extension>.xml</Extension>
<Content>text/xml</Content>
</Type>
<Type>
<Extension>.xls</Extension>
<Content>application/vnd.ms-excel</Content>
</Type>
<Type>
<Extension>.csv</Extension>
<Content>application/octet-stream</Content>
</Type>
<Type>
<Extension>.pdf</Extension>
<Content>application/pdf</Content>
</Type>
</Types>
</Web>
</Config>
```

The following table describes the tags listed in the example.

Example tags

Tag	Description
`DefaultURL`	The URL of the QlikView Server.
`ConfigUrl`	This is the URL the QMC uses to communicate with the QlikView Web Server.
`TunnelUrl`	The URL used for tunneling.
`QvsStatusUrl`	The URL to the status page for the QlikView Server.

Tag	Description
`LogLevel`	Sets the level of logging. Possible settings are `Information` (High), `Warning` (Medium) and `Error` (Low).
`UseCompression`	Set whether the information sent should be compressed.
`InstallationPath`	The installation path of the QlikView Web Server.
`QvsAuthenticationProt`	How the QlikView Server Authenticates. Set to `Negotiate`, `Kerberos` or `NTLM`.
`AddCluster - Name`	The name of the cluster.
`AddCluster - LoadBalancing`	How the load balance should be calculated. Possible values are `Random`, where the client is directed to a QVS at random, `CpuUsage` where the QVS reporting the least average CPU will be selected or `LoadedDocument`, where the client is directed to the QVS where the document the client requests is already loaded.
`AddCluster - AddQvs - AlwaysTunnel`	Set to `true` to always tunnel the communication to the QlikView Server.
`AddCluster - AddQvs - Machine`	The name of the computer where the QlikView Server is running.
`AddCluster - AddQvs - Port`	The port the QlikView Server listens to.
`AddCluster - AddQvs - LinkMachineName`	The external name of the QlikView Server, used by the QlikView Plug-in clients.
`AddCluster - AddQvs - Weight`	Set a higher value if you wish that the QlikView Server be selected more frequently when using random load balancing.

Tag	Description
`AddDSCCluster - CustomUserPort`	The port for the custom user Directory Service Connector.
`AddDSCCluster - DirectoryServiceConnectorSettings - Url`	The location of the Directory Service Connector.
`AddDSCCluster - DirectoryServiceConnectorSettings - Name`	The cluster name.
`AddDSCCluster - DirectoryServiceConnectorSettings - Username`	Enter a user name if needed to connect to the Directory Service Connector.
`AddDSCCluster - DirectoryServiceConnectorSettings - Password`	Enter a password if needed to connect to the Directory Service Connector.
`Authentication - AuthenticationLevel`	Sets how the client should access the AccessPoint. Possible values are `Always`, `Login` and `Never`.
`Authentication - LoginAddress`	The path to an alternative login page used for custom users.
`Authentication - LogoutAddress`	The path to an alternative logout page used for custom users.
`Authentication - GetTicket`	The URL and authentication used to get a ticket from the Server for a client.
`Authentication - HttpAuthentication`	The URL and authentication used to get a ticket from the Server for a client if using SSL/TSL.
`AccessPoint - Path`	The path where the Access Point is installed.
`AccessPoint - AjaxClientPath`	The relative path to the Ajax client.
`AccessPoint - PluginClientPath`	The relative path to the QlikView plug-in client.
`AccessPoint - DefaultPreferredClient`	Sets which client should be set as the preferred client for a user's first visit to the AccessPoint for clients.

Tag	Description
`AccessPoint - DefaultView`	The default view of documents on the AccessPoint, details or thumbnails.
`AccessPoint - DefaultPagesizeDetails`	The number of rows on the AccessPoint when using the Details view.
`AccessPoint - DefaultPagesizeThumbnails`	The number of rows on the AccessPoint when using the Thumbnails view.
`AccessPoint - RespectBrowsable`	When set to `True` only mounts that are set as `Browsable` in the QVS are displayed on the AccessPoint.
`Ajax - Path`	The path to *QvsViewClient.aspx*. The path may be changed, but the file name must remain unchanged for the installation to work.
`Ajax - NoCrypto`	Prohibit the use of encryption between the QlikView Web Server and the QlikView Server.
`Ajax - ProhibitMachineID`	Prohibit sending the machine ID. This will effectively exclude the usage of anonymous bookmarks.
`Ajax - Recording`	When set to `True`, the qvpx calls for the AJAX zero footprint client are logged.
`SafeForwardList`	When a redirect is requested through *Authenticate.aspx*, a DNS lookup is done to retrieve the IP addresses of the path provided in this tag. If the IP addresses matches that of the redirect request, the redirect is allowed.

Tag	Description
`StrictSafeForwardList`	When a redirect is requested through *Authenticate.aspx*, the host name of the path provided in this tag is compared with the host name of the incoming redirect path. If they match (not case sensitive), the redirect is allowed.
`Web - Folders`	The path to the different virtual folders in the QlikView Web Server. Change the name and path if the files are installed to folders other than the default.
`Web - Types`	Specify what file extensions the clients are allowed to download from the Access Point/QlikView Web Server.

Load Balancing

QVWS hosts web pages, prepares the file list for AccessPoint, and manages the load balancing of QlikView Servers (QVSs).

AccessPoint is a web portal for documents hosted on QVWS. The pages for AccessPoint are by default located in the folder *%ProgramFiles%\QlikView\Web*. QVWS also acts as web server for any AJAX pages accessed by the end users.

The load balancing performed by QVWS is different from load balancing a web server, since the additional work and resource consumption is almost similar for each user, so it does not matter on which server the user ends up.

The load balancing schemes are listed below.

Load balancing schemes

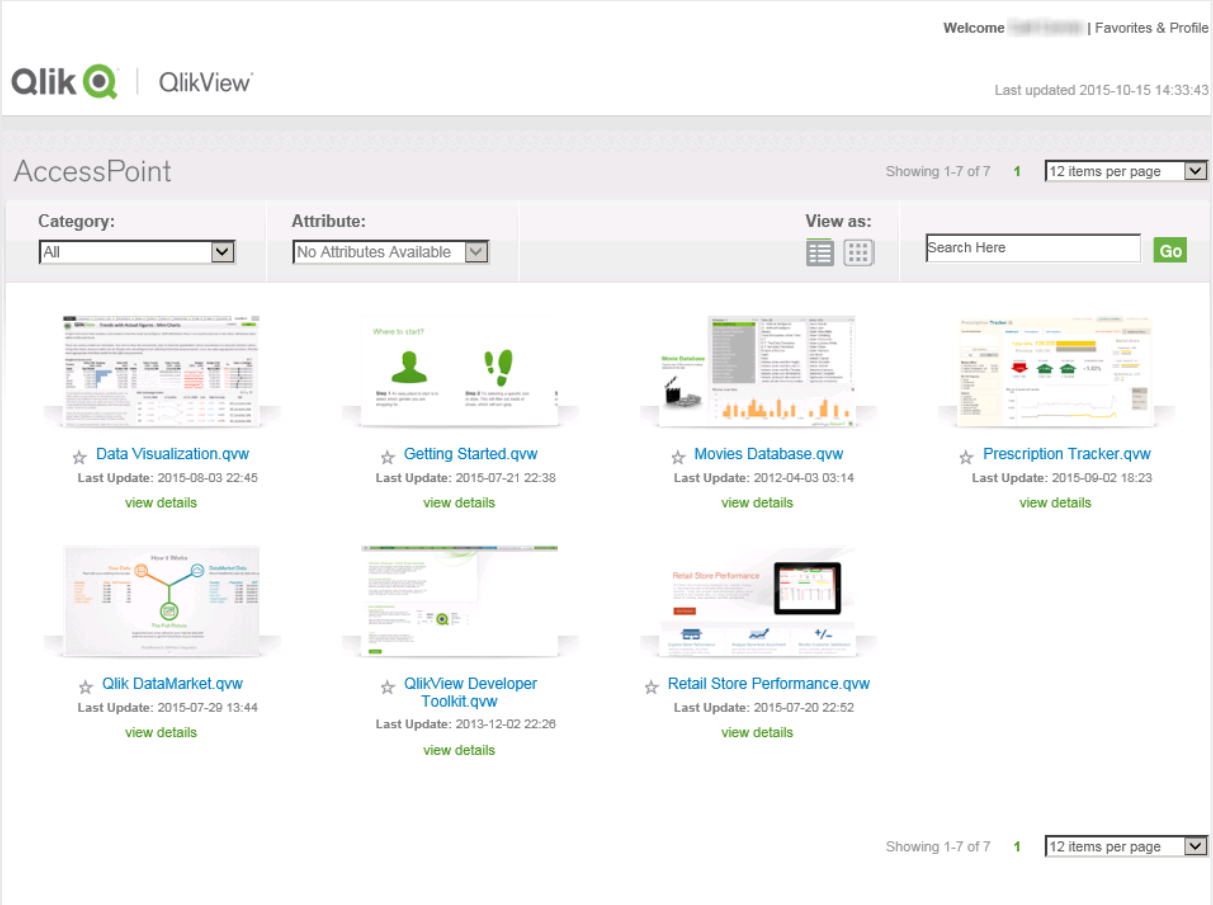
Scheme	Description
Random	The default load balancing scheme. The user is sent to a random server, no matter if the document the user is looking for is loaded or not.
Loaded Document	If only one QVS has the particular document loaded, the user is sent to that QVS. If more than one QVS or none of the QVSs has the document loaded, the user is sent to the QVS with the largest amount of free RAM.
CPU with RAM Overload	The user is sent to the least busy QVS.

The settings for load balancing are configured in QMC.

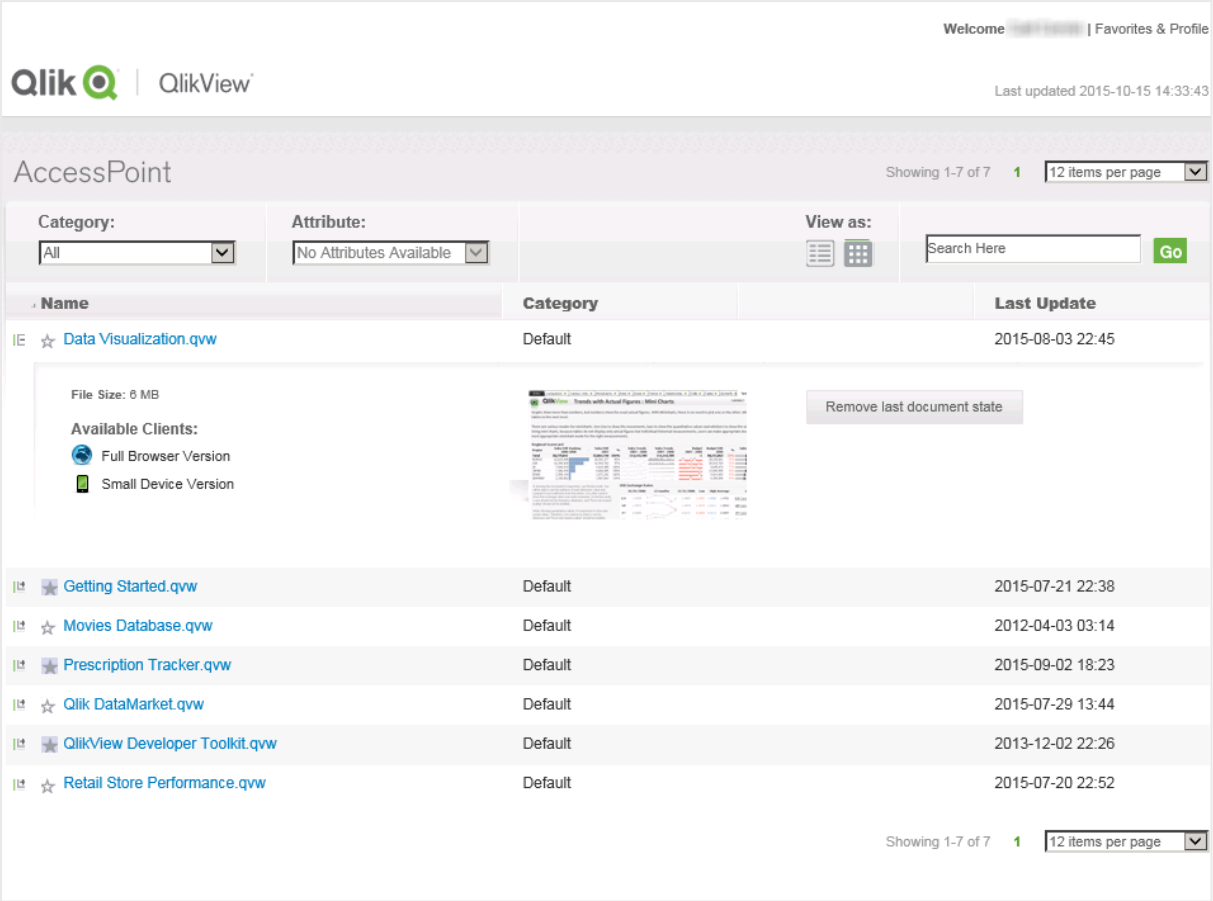
QlikView AccessPoint

QlikView AccessPoint is a web portal that lists the documents each user has access to. AccessPoint only links to each document - it does not host the documents. The hosting is done by QlikView Server.

The documents can be displayed as thumbnails or in a detailed list.



Thumbnails view in AccessPoint



Detailed view in AccessPoint

The settings available in AccessPoint are listed below.

AccessPoint settings

Setting	Description
Category	Category grouping for the document. Categories are managed in QMC under **Documents**>**User Documents**>**Document Information**.
Attribute	Attribute grouping for the document. Attributes are managed in QMC under **Documents**>**User Documents**>**Document Information**.
View as	Document display type, **Detailed** view or **Thumbnails** view. In the Detailed view, the documents can be sorted by Name, Category, and Last Update.

Click a **view details** link in the Thumbnails view or a plus sign (**+**) to the left of a document name in the Detailed view to display additional information on a document (see below).

Additional document information

Field/Button	Description
Last Update	When the document was last updated. *This is only displayed in the Thumbnails view.*
Next Update	When the document will be updated next time. *This is only displayed if the document is part of a task that has a schema.*
File Size	Size of the document.
Available Clients	Click a client to open the document with that client.
Remove last document state	Click this button to remove the last document state.

*If the QlikView admin enables the **Display warning if last update time is older than (minutes)** setting in the QMC, you will see an icon beside the last update field. A ✔ indicates that the document has been updated within the defined period. A ⚠ indicates that the document has not been updated within the defined period.*

Click a star icon next to a document name in the Thumbnails or Detailed view to set the preferences for the document.

Document preferences

Setting	Description
Open with	Select a client to make it the default client to open the document with.
Add to favorites	Click this link to add the document to the favorite documents. Select **Category**>**Favorites** in AccessPoint to display the favorites.

Modifying the modal dialogs in the Ajax client

The modal dialogs, such as **Print**, **Export**, and **Server Connection Lost**, can be modified in the file *customTranslations*.

Navigate to *C:\Program Files\QlikView\Server\QlikViewClients\QlikViewAjax\htc\customFiles*. The files *customConfig* and *customTranslations* are empty, but the files *customConfigExample* and *customTranslationsExample* present examples on how to edit.

In the file *customConfig*, it is a prerequisite that `TranslationEvents` is set to `true` in order for the edits in *customTranslations* to be valid.

For the changes to take effect, the server has to be stopped and restarted.

Directory Service Connector

Overview

Directory Service Connector overview

Executable	*%ProgramFiles%\QlikView\Directory Service Connector\QVDirectoryServiceConnector.exe*
Data	*%ProgramData%\QlikTech\DirectoryServiceConnector*
Listens to	HTTP: 4730; SNMP: 4731
Uses/Controls	-
Used by	QDS, QMS, QVWS

Files

Settings and Configuration

These settings originate from QVPR.

Settings and configuration files

File	Description
Config.xml	Configuration file for the service.
Resources/<id>.xml	DSP configurations.

Logs

Log files

File	Description
Log\<date>.txt	Event and error log.

DSP Interface

The reason for developing a proprietary Directory Service Provider (DSP) is to have QlikView distribute documents to users in a directory service not supported by default, and to provide group resolution to the web server.

DirectoryServiceProvider

DirectoryServiceProvider is the interface of the class that plugs into the framework. The members of the interface are listed below.

DirectoryServiceProvidor interface members

Member	Description
`LogMessage LogMessageEvent { set; get; }`	Directly after construction, this field is instantiated with a delegate that provides crude logging facilities.

Member	Description
`string ProviderName { get; }`	A free-form, preferably descriptive, name of the component that is suitable for the end user.
`string ProviderType { get; }`	An installation-unique identifier used internally by the framework and related components. The identifiers used by the supplied providers are `AD`, `NT`, `Local`, and `Custom`.
`void SetupPath (string _path, string _username, string _ password);`	Creates a node that represents the corresponding directory service node on the specified path. Upon failure, an exception is thrown.
`IList<string>GetKnownRootPaths ();`	The returned list should contain one or more viable paths for the methods listed here.
`void ClearCache ();`	Clears the cache (if any).
`string DomainName { get; }`	A “domain name” associated with the path that is set up. It is used as a qualifier to separate nodes from different providers (for example, the shipped Active Directory provider uses `NetBIOSName` as domain name).
`IDictionary<string, string> GetSettings ();`	The dictionary of supported settings has the name of the setting as `key` and the name of the type as `value`.
`void SetSetting (string _name, string _value);`	The parsing responsibility is obviously put on the provider.
`IList<IDSObject> Search (string [] _pattern, eSearchType _type, string _otherattribute);`	Searches for nodes with attributes matching any of the patterns provided. The attributes are specified with the `type` parameter, which can be one or more values from the enumeration. If `type` is “other”, the last parameter specifies the name of the attribute. The search type “legacyid” is used for backwards compatibility. Search should support patterns containing the wildcard sign “*”, which matches zero or more characters of any kind.
`void Dispose ();`	Called whenever a provider object is released.
`IDSObject`	A simple interface for any type of node within the directory service.
`string ID { get; }`	Node ID, unique within the instantiated path and consistent over all executions.
`string DisplayName { get; }`	Common name of the node in the directory service.
`string AccountName { get; }`	Account name associated with the node (if present).
`eDSObjectType ObjectType { get; }`	Basic type of the object.
`IList<IContainer> MemberOf ();`	A list of all groups that the node is member of.

Member	Description
`string GetCustomProperty (string _name);`	Any other property not natively supported by the interface. If not present, null is returned.
`string Email { get; }`	The primary e-mail address associated with the node (if any).

QlikView Management Service

Overview

QlikView Management Service overview

Executable	*%ProgramFiles%\QlikView\Management Service\QVManagementService.exe*
Data	*%ProgramData%\QlikTech\ManagementService*
Listens to	HTTP: 4780 (Web); HTTP: 4799 (API); SNMP: 4781
Uses/Controls	DSC, QDS, QVS, QVWS
Used by	Web browser/API client

Files

Settings and Configuration

QlikView Management Service (QMS) keeps a global view of the settings in QVPR.

Settings and configuration files

File	Description
Config.xml	Configuration file for the service.

Logs

Settings and configuration files

File	Description
Log\<date>.txt	Event and error log.

SNMP

QlikView provides SNMP agents for all services.

> *QlikView supports the iReasoning MIB browser for pulling data from the SNMP agents.*

The SNMP setting is off by default, since the implementation is in its initial stages and subject to change. At the time of writing, reading operations from the agents are enabled. The following messages are supported:

- `GetRequest`
- `GetResponse`

- `GetNextRequest`

All services answer the standard SNMP queries (see below).

Standard SNMP queries

Identifier	Query	Description
1.3.6.1.2.1.1.1	sysDescr	Description of service/product. Example: `sysDescr.0:Qlikview Publisher Commandcenterservice version 8.50.600`
1.3.6.1.2.1.1.2	sysObjectID	Unit type. Example: `sysObjectID.0:iso.org.dod.internet.private.enterprises.qliktech.products.publisher.Distributionservice`
1.3.6.1.2.1.1.3	sysUpTime	System uptime. Example: `sysUpTime.0:0 hours, 12 minutes, 15 seconds`
1.3.6.1.2.1.1.4	sysContact	Can be set in the configuration file. Example: `sysContact.0:Unspecified System contact`
1.3.6.1.2.1.1.5	sysName	Can be set in the configuration file. Example: `sysName.0:Unspecified name`
1.3.6.1.2.1.1.6	sysLocation	Can be set in the configuration file. Example: `sysLocation.0:Unspecified location`
1.3.6.1.2.1.1.7	sysService	Constant, 72 means application server. Example: `sysServices.0:72`

The QlikView Distribution Service can answer additional queries. These are specified in the MIB file.

Each service has a configuration file, which is stored in the subfolder for the service in the installation folder. For example, the configuration file for the QlikView Distribution Service is *QlikViewdistributionService.exe.config*.

The SNMP settings can be adjusted in the `SNMP SETTINGS` part of the configuration file. SNMP has to be enabled for all services (the default is off).

SNMP settings

Setting	Description
EnableSNMP	Enables the SNMP listener. The default value is `false`.
SNMPPort	Sets the port to use for the particular Publisher service. See the default settings for each service below.
SNMPsysContact	Contact information for the person responsible for the managed node. The default value is `Unspecified System contact`.
SNMPsysName	An administratively assigned name for the managed node. By convention, this is the fully qualified domain name of the node. If the name is unknown, the value is a zero-length string. If left empty, it defaults to the current machine name. The default value is `Unspecified name`.
SNMPsysLocation	Physical location of the node (for example, “telephone closet, third floor”). The default value is `Unspecified location`.
DebugSNMP	Enables the extended debug log for the SNMP listener. The default value is `false`.

The default port settings for the services are listed below.

Default port settings

Service	Default Port Setting
QlikView Management Service	4781
Directory Service Connector	4731
QlikView Distribution Service	4721 (default SNMP port)
QlikView Server	161
QlikView Web Server	4751

All ports can be configured. If the services are installed on different machines, they can all run on the same port. The ports change as the implementation moves away from the experimental SNMP range and into the range allotted by Qlik.

MIB File

A MIB file is included in the QlikView delivery, so that all SNMP managers can interpret the additional responses from the QlikView Distribution Service. Note, however, that the MIB file is subject to change. The file is installed in *\QlikView\Support Tools*. The support tools require a customized installation.

The QlikView Distribution Service can answer the queries listed below, in addition to the ones previously mentioned.

Other queries

Identifier	Query
1.3.6.1.4.1.30764.1.2.2.1	QDSTaskExecuteStatusTable
1.3.6.1.4.1.30764.1.2.2.1.1	QDSTaskExecuteStatusEntry
1.3.6.1.4.1.30764.1.2.2.1.1.1	QDSTaskID (task ID number)
1.3.6.1.4.1.30764.1.2.2.1.1.2	QDSTaskName (task name)
1.3.6.1.4.1.30764.1.2.2.1.1.3	QDSTaskExecuteStatus (task status): • Waiting • Running • Aborting • Failed • Warning
1.3.6.1.4.1.30764.1.2.2.1.1.4	QDSTaskNextExecutionAt (when the task will be executed next)
1.3.6.1.4.1.30764.1.2.2.1.1.5	QDSTaskLastExecutedAt (when the task was executed last)
1.3.6.1.4.1.30764.1.2.2.1.1.6	QDSTaskCurrentWork (what the task is currently doing)
1.3.6.1.4.1.30764.1.2.2.1.1.7	

See also:

≤ A Simple Network Management Protocol
≤ Simple Network Management Protocol (Wikipedia)

2.2 Deployment

The QlikView architecture is based on the concept of sites. A QlikView site is a collection of one or more nodes (that is, server machines) connected to a common logical repository or central node.

QlikView can be deployed in many ways. This section describes different deployment scenarios.

Building a Farm

Server farms can be used to provide additional performance, redundancy, and security in place of a single server solution.

Planning

Before starting the actual installation, planning is needed. The following items have to be considered:

- Trust mechanism
- Web server (QlikView Web Server or Microsoft IIS)
- Redundancy level
- Account to run the services under

- QVPR format (XML or SQL)
- User directory
- User authentication
- Firewalls

Trust Mechanism

Trust mechanisms are provided with Windows groups or certificates.

Windows groups can easily be deployed, if all services reside in a single Active Directory (AD). If encrypted communication is needed, it can be added manually.

Certificates provide for trust mechanisms in cross-domain environments and can also provide SSL/TSL encryption.

Web Server

QlikView Web Server is intended for use when the web server is not needed for other purposes. It is lightweight and easy to manage, but at the same time limited to support the tasks needed by a QlikView installation.

A Microsoft IIS-hosted web server is recommended, if:

- More flexibility or more advanced tuning is required
- The web server is to be used for other tasks than QlikView
- An authorization scheme not available out-of-the-box is required

Redundancy Level

The redundancy level is mainly a question of clustering and/or having multiple machines running the same service. All services except QlikView Management Service (QMS) can be installed on multiple machines. In addition, QlikView Server (QVS), QlikView Distribution Service (QDS), and Directory Service Connector (DSC) can be clustered.

Account to Run the Services Under

A dedicated account should be created to manage the QlikView services. The account should be assigned with proper privileges during the installation.

It is recommended that the same account is used for all services.

QVPR Format

The choice of QVPR format is based on reasons outside the QlikView product (for example, backup and availability). The installation always starts in XML mode.

User Directory

QlikView defaults to Windows users (that is, NTFS mode). If non-Windows users are to be given access (other than anonymously), QlikView Server must run in Document Metadata Service (DMS) mode.

User Authentication

QlikView supports multiple authentication schemes. Additional schemes may require ASPX development and the possible use of Microsoft IIS for web services.

Firewalls

Make sure that the services are able to communicate (for example, by opening the appropriate ports in the firewalls).

Services (page 21)

Root/First Install

Before starting, make sure that the appropriate service account (or accounts) is set up and available on the machines where the services are to be installed.

In all installations, there must exist exactly one QMS, which must be installed first. Note that the QMS must be able to communicate with all the subsequently installed services.

If more services are to run on the same server, they can be installed at the same time.

Adding Services on Other Machines

The next step is to install the other services on the other servers. If more services are to run on the same server, they can be installed at the same time. The order in which the services are added is not important.

When the services have been installed, it is time to return to QlikView Management Console (QMC) and configure the services. This is done on the System tab. The first step is to add the services. Make sure to note the differences between building out a cluster and creating a brand new cluster.

Clustering

This section provides an overview of how create a QlikView Server cluster.

QlikView Server

For the QlikView Server cluster to work properly, it is important to set **System**>**Setup**>**QVS resource**>**Folders**>**Root Folder** to a common shared folder. In addition, **Alternate Temporary Files Folder Path** must be set to a common shared folder (separate from the root folder).

If extensions are used, it simplifies management if **Alternate Extension Path** is set to a common shared folder.

It is also common practice to set **System**>**Setup**>**QVS resource**>**Logging**>**Log Folder** to a common place, but this is not strictly necessary.

> *The root folder must not be used for anything else than cluster files (that is, .pgo files) and user documents.*

QlikView Distribution Service

For a cluster of QDSs, **System**>**Setup**>**General**>**Application Data Folder** must be set to a common shared folder. In addition, **Source Folders** must be common shared folders.

Directory Service Connector

A cluster of DSCs does not need any specific settings. The difference between clustered and non-clustered DSCs is whether the settings are shared or not.

QlikView Web Server

Multiple web servers can be set up, but they are always configured independently (that is, they are never clustered). Note that it is uncommon, but from a technical perspective possible, to have some web servers running QlikView Web Server (QVWS) and some Microsoft IIS.

Tunneling Using Microsoft IIS

Tunneling is used by Windows native clients (QlikView Desktop, the OEM OCX, and the QlikView plugin) and needed when the clients cannot communicate with QlikView Server on port 4747 (most likely due to a firewall blocking the traffic):

- QVWS: No extra settings are required.
- Microsoft IIS: The *QVSTunnel.dll* file must be added as an ISAPI filter.

Proceed as follows to set up tunneling for Microsoft IIS 7:

1. Open the Internet Information Services Manager.
2. Select the IIS top node.
3. Open the ISAPI and CGI Restrictions dialog.
4. Select **Add** in the Actions pane and browse to the location of *QVSTunnel.dll*.
5. Provide a description of the instance and check the **Allow extension path to execute** box.
6. Open the site that is to host the QlikView Server and Publisher pages and click **Scripts**.
7. Open the Handler Mappings dialog.
8. Locate ISAPI dll and select **Edit Features Permission** in the Actions pane.
9. Click **Execute** in the dialog that opens.

The following entries are required in the registry when the QVS and Microsoft IIS are located on different machines:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\QlikTech\QlikTunnel]
```

- “QVSPort”=dword:000012a6
- “QVSServer”=“QvsHost”

If the entries do not already exist in the registry, they have to be added manually.

Test the QlikView Server tunnel by entering the following URL in a client browser window:

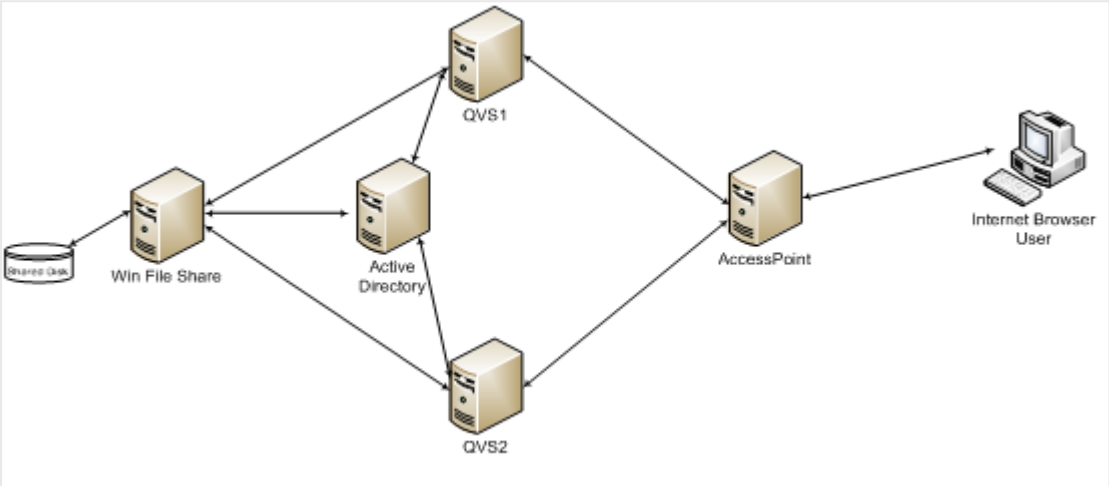
http://<Servername>/scripts/qvstunnel.dll?test

Servername is the web server. If the tunnel is correctly set up, the web page returns a message (that tunneling is available) and the QlikView Server version number.

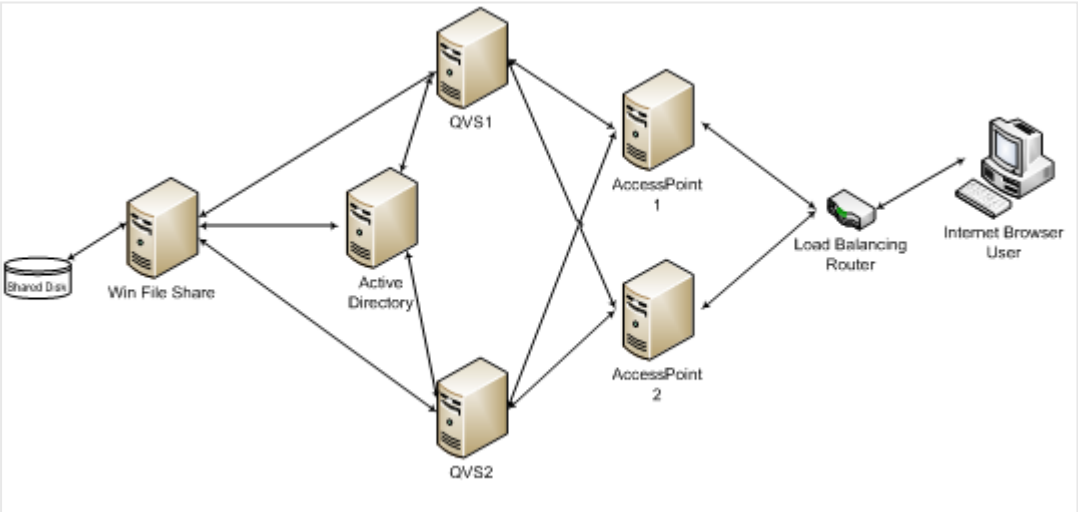
Clustering QlikView Servers

This chapter discusses the architectural and installation requirements and options for building a clustered and resilient QlikView Server deployment.

The following figure shows a clustered, load balanced QlikView Server deployment.



The following figure shows a resilient, clustered, load balanced QlikView Server deployment that uses AccessPoint load balancing.



The QlikView Server load balancing capabilities are included in the QlikView web portal, AccessPoint. This chapter also discusses how to make this component resilient using network load balancing (if needed).

Why Cluster QlikView Servers?

By clustering QlikView Servers, the objectives described below can be achieved.

Horizontal User Scalability

If more resources than can be provided by a single QlikView Server are needed, an additional server can be added. For example, if the server can support 100 concurrent users, but 200 concurrent users have to be supported, an additional server can be added. In this scenario, the first 100 users could be allocated to server A and the second 100 users to server B. Alternatively, the servers could be clustered so that you set resilience.

Resilience

When the number of users increases, so does the users' reliance on QlikView. By clustering the QlikView Servers, resilience can be built into the deployment. In the case above, where a single server can support 100 users, three servers could be used to build resilience into the deployment. This would allow one server to be lost (for example, because of hardware failure) with the system still capable of supporting 200 users. Having all three servers as active nodes helps reduce the response times by not running all servers at 100% of their capacity. This also limits the number of users affected if a node is lost.

QlikView does not provide any session recovery. In practice, this means that if a node in the QlikView cluster is lost, the users lose the analysis they are currently performing. They will have to reconnect to the cluster to resume their work. This does not mean that the data within the QlikView application is lost and needs to be reloaded, because the data is stored in the QlikView document file on the file system. Only the selections made in the application are lost.

Load balancing

A QlikView deployment uses a load-balancing algorithm to take advantage of the full capacity of all the machines in a QVS clusters. The web server running the AccessPoint determines which QVS to use. There are three options for how to load balance your QVS. See: *QVS Load Balancing Options (page 56)*.

Requirements for Clustered QlikView Deployment

There are three high-level requirements for building a clustered QlikView deployment:

- Clustered QlikView Server license key
- Shared storage area for Root folder
- Same build number

Clustered QlikView Server License Key

In a clustered environment, the QlikView Server machines are installed with the same license key, which must be enabled for clustering. This can be checked confirmed by examining the following entry in the License Enabler File (LEF):

`NUMBER_OF_CLUSTER_NODES; 2` (number of nodes in the cluster)

Clustered QlikView Servers share configuration and license information between themselves via the shared storage, so that configuration and license management only needs to be performed once from the QlikView Management Console (QMC) for all nodes.

The servers must be installed on the same network subnet and have a shared root document directory; hence the requirement for a shared network storage. The configuration information is stored in Persistent Global Objects (*.pgo*) files.

If the servers fail to start or reset after ten minutes, check for the LEF entry above. This is usually an indication that QlikView Server is installed on more machines than allowed.

Shared Network Storage

In QlikView shared network storage is required for storing QlikView documents that need to be accessed in a QlikView Server cluster. A shared network storage is also used for storing *.pgo* (Persistent Global Objects) files, *.meta* files, and shared files (*.Shared* or *.TShared*). Configuring a shared network storage enables collaborative objects to be shared across the nodes in the cluster (using shared files).

The requirements for a shared network storage in QlikView Server are the following:

- The network storage must be hosted on a Windows-based file share.
- QlikView Server (QVS) supports the use of a SAN (NetApp, EMC, etc.) mounted to a Windows Server 2008 R2 (or later) and then shared from that server.
- The QlikView Server nodes in the cluster must have network latency below 4 milliseconds to connect to the file share server. Performance can degrade if this is not the case.
- The bandwidth to the file share must be appropriate for the amount of traffic on the site. The frequency and size of the documents being saved after reloading, and opened into memory, drives this requirement. 1 Gigabit networking is suggested.
- The following shared storage options are not supported:
    - Shared storage systems based on Linux OS are not supported. This includes systems supporting SMB file sharing protocol or NTFS disk drive format .
    - Windows-based shared storage systems that rely on CIFS file sharing protocol are not supported.
    - QlikView does not support Windows Distributed File System (DFS).

> *Hosting files on any type of unsupported system may create an unstable QVS cluster where CALs disappear and QVSs stall.*

> *When you upgrade from QlikView Server 11.20 to QlikView Server 12.10 or later, your installation might encounter a variety of issues due to backend file system. QlikView Server 12.10 and later versions are more disk intense and require bigger file server than QlikView 11.20. When planning your QlikView deployment, it is important to keep in mind the type of storage and resource capacity. For more information, read the following Qlik Support article:* QlikView and its backend File Share System.

> *QlikView does not support Windows Distributed File System (DFS).*

Although not strictly required, it is also good practice to :

- Set an alternate path for temporary files to a common shared folder reachable by all QlikView Servers in the cluster.

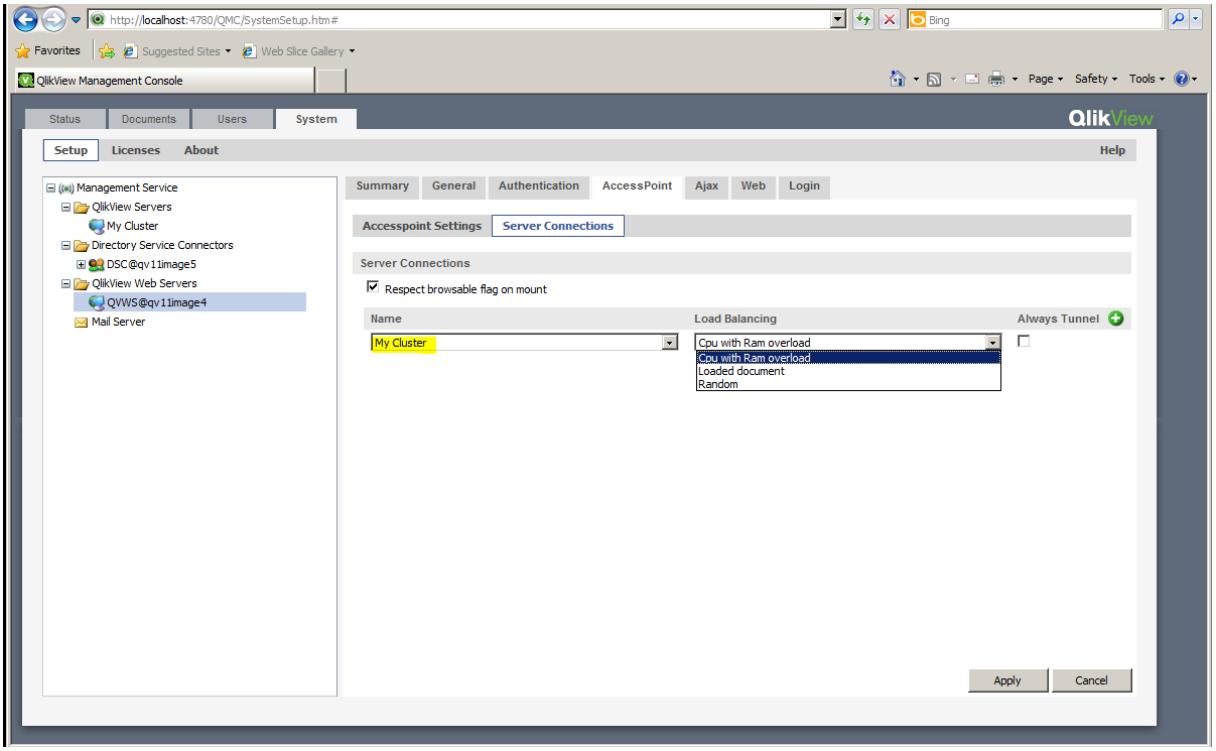
- If extensions are used, set the alternate extension path to a common shared folder.
- Set the log folder to a common shared folder.

QVS Load Balancing Options

QVS supports three load balancing strategies:

- Random (default setting): A round robin type strategy ideal for most users, since the session is distributed across all nodes in the cluster.
- Loaded document: Used when sessions for the same document are to be routed to the same server. This strategy is designed for deployments where there are more documents than a single node in the cluster can handle. AccessPoint makes the decision based on if the document is already loaded and on the amount of RAM available on the server.
- CPU with RAM overload allows QlikView Web Server (QVWS) to route traffic based on two factors, (1) RAM and (2) CPU use. The node is chosen using the following criteria:
  - If RAM is readily available (low) on all available nodes, choose the node with the lowest CPU use.
  - If RAM is moderately used on all available nodes, choose the node with the most RAM available.

The QVS load balancing strategy can be set in the QMC under **System>Setup>QlikView Web Servers**. Select the web server on the **AccessPoint** tab:



Location of the Load Balancing options.

Load Balancing the Web Server

The network load balancer provides the resilience for AccessPoint, routing the sessions to an available AccessPoint server. This is done by third-party software and hardware.

There are several requirements on the load balancer:

- Support for session persistence / sticky sessions: This ensures a user's session persists on the same node within the cluster, usually by using a cookie.
- Availability: The load balancer checks the availability of the AccessPoint web server and the QlikView servers.
- Some form of load balancing algorithm to determine which server is the least loaded.

Session Persistence

The requirement is for the user's session to be routed consistently to the same server. Methods for doing this vary from device to device - refer to the load balancer documentation for information on the options available.

Availability Checking

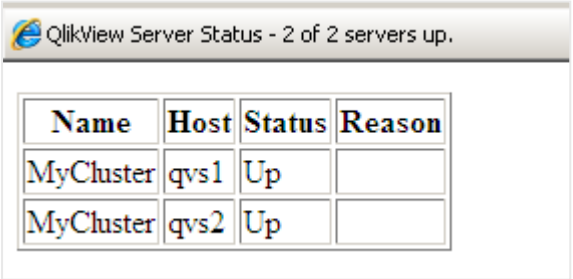
A special web page on the AccessPoint provides automated checking of the system status:

http://myAccessPoint/QvAjaxZfc/QvsStatus.aspx

This page returns an http status code of 200, if the AccessPoint and at least one QlikView Server in the cluster respond. Any other status code returned by this page should be considered an error. Common errors from this page include:

- 404: The AccessPoint is unable to respond. Check the web server.
- 503: No QlikView Servers responded to the AccessPoint and therefore it cannot service user requests.

The status of the QlikView Server cluster is also displayed on the web page:

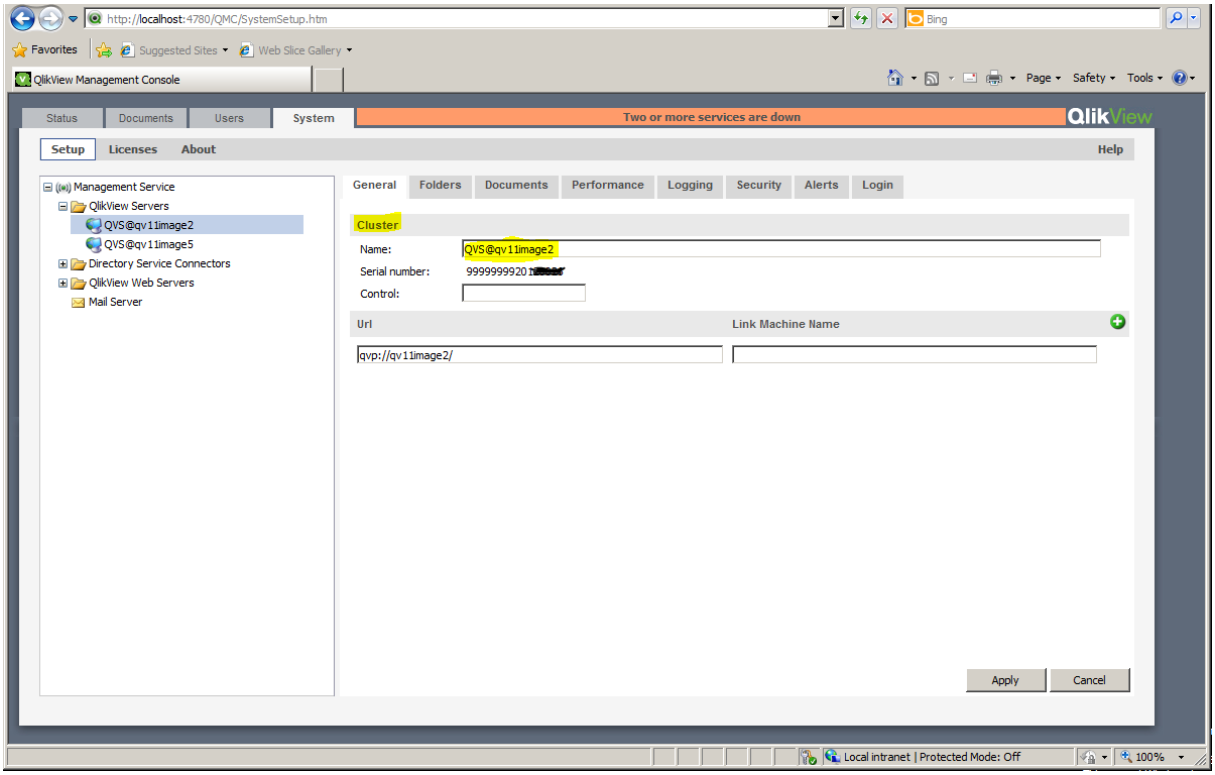


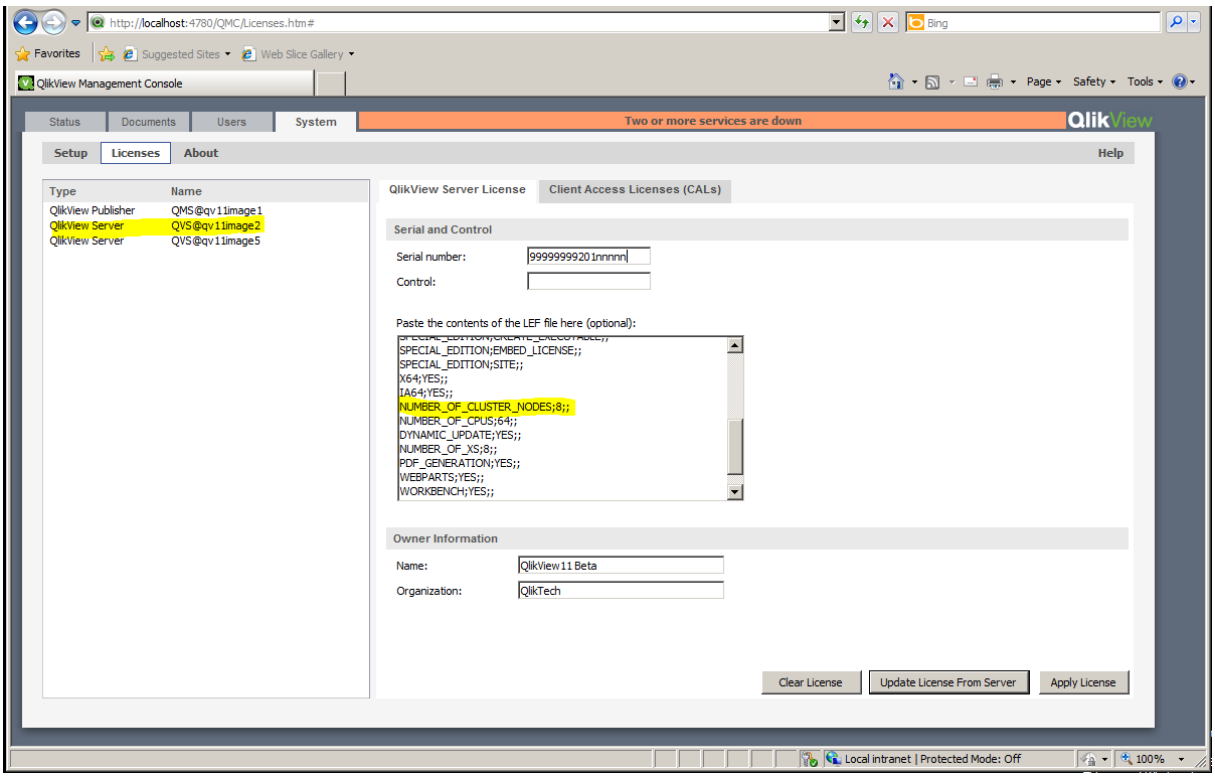
The QlikView Server Status screen.

Building and Installing a QlikView Cluster

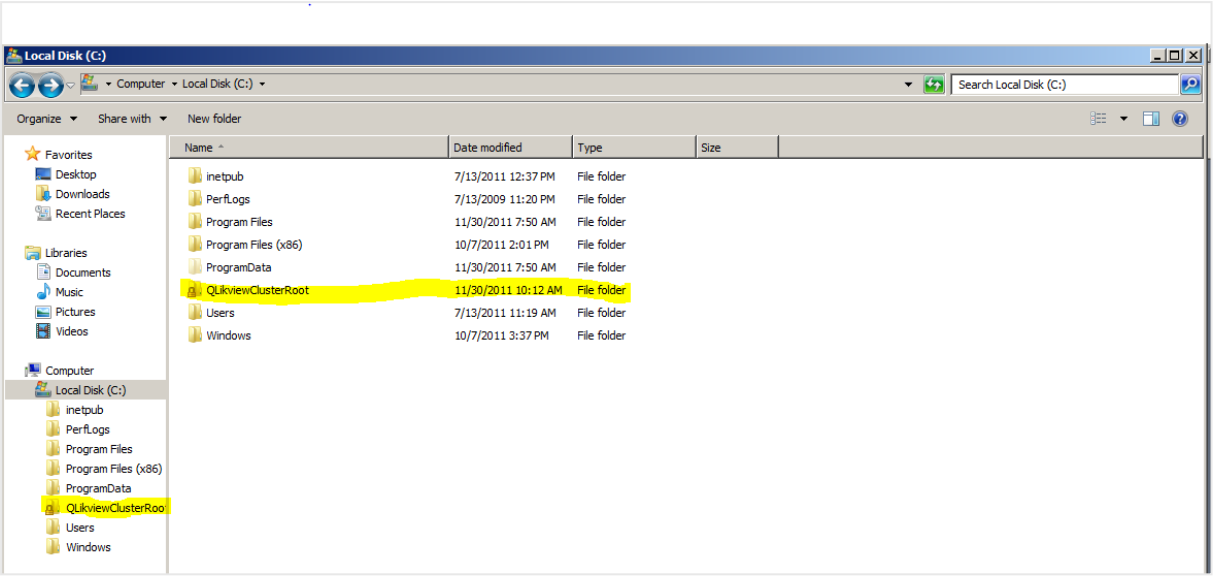
Proceed as follows to configure and activate a QlikView Server cluster using the QMC:

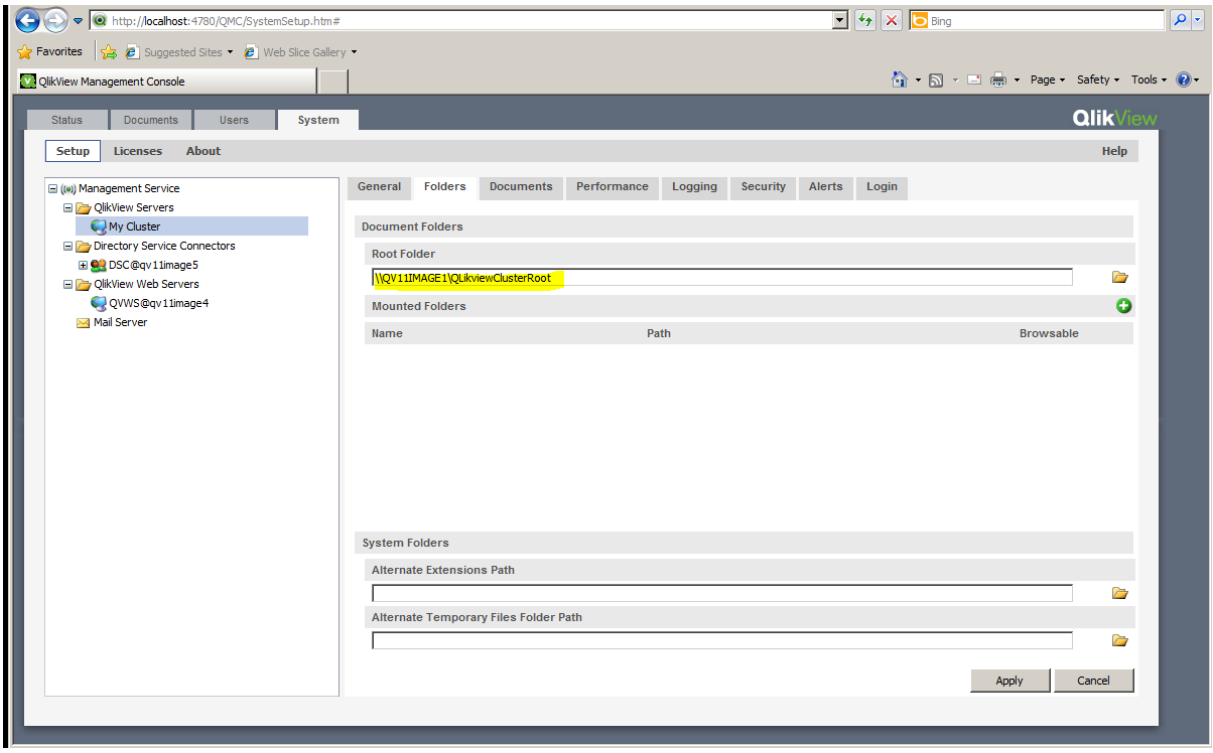
1. Install and license the first QlikView Server in the cluster. This will be the first copy of QlikView Server.



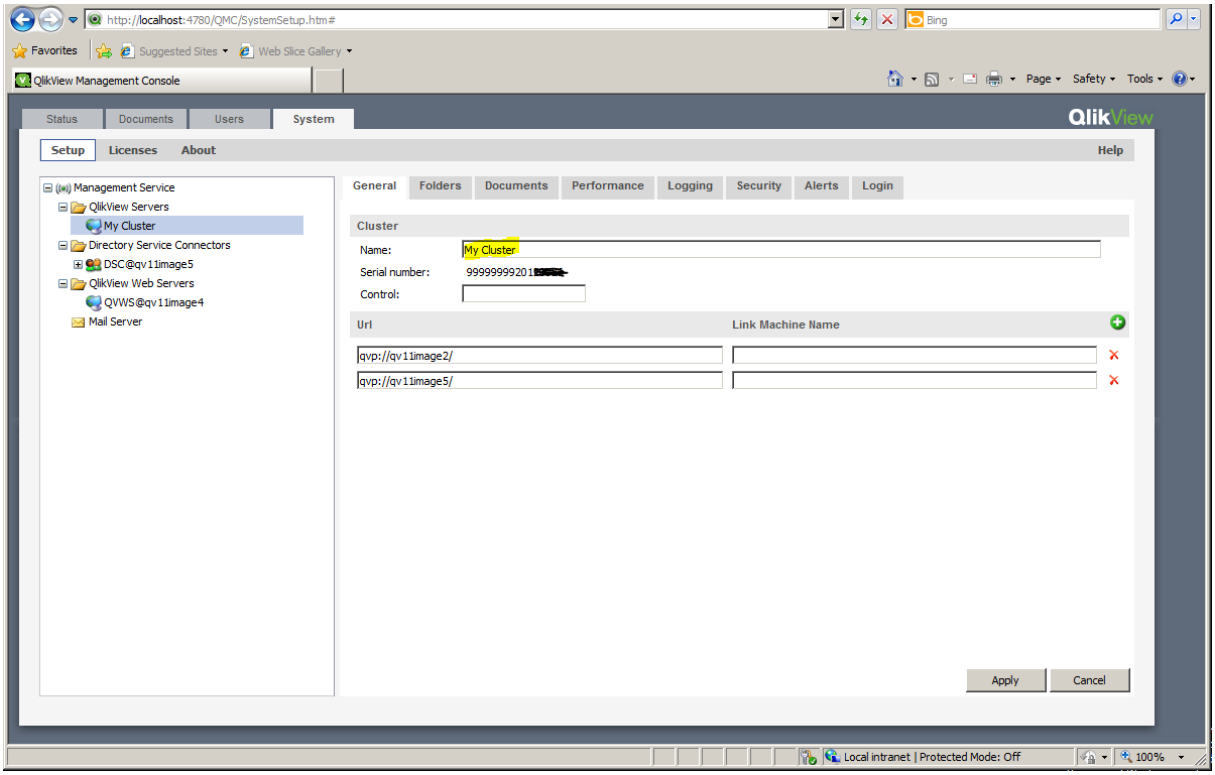


2. Configure the document folder to point to a folder on the file system that all QlikView Servers in the cluster can access.

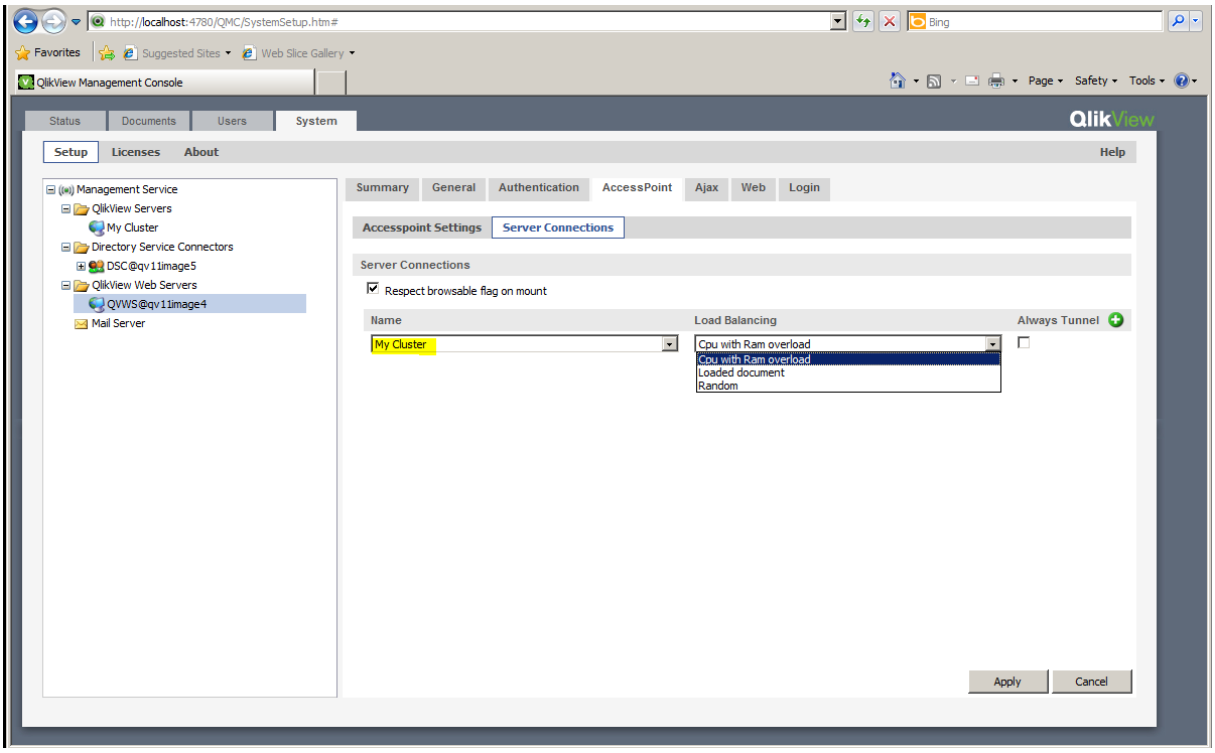




3. Install the next QlikView Server in the cluster.
4. Ensure that all QlikView services are running as local administrators and that they are members of the “QlikView Administrators” local group.
5. Open **System**>**Setup** in the QMC and select the server. Then go to the **General** tab and enter the control number for your license and the address to the second QlikView Server in the cluster.
6. Rename your cluster to an appropriate name.
7. Repeat steps 3 - 5 for the QlikView Server nodes in the cluster.



8. Make sure that the cluster is selected in **Server Connections** in the settings for the AccessPoint.



9. The cluster is now configured and ready to use.

Removing a node from a QlikView Server cluster

Do the following:

1. Navigate to the **System** tab in QMC and select **QlikView Server**.
2. Highlighting the QVS Cluster, identify the node that you want to remove and select .
3. Enter the license control key and select **Apply**.
4. Restart the QlikView Server Service (QVS).

Unbalanced QVS Clustering

By default, a QlikView Server (QVS) cluster requires that all nodes are equal regarding CPU, cores, and RAM. You can, however, use nodes with different hardware specifications. Setting up an unbalanced cluster can be helpful in case you need to cluster machines with different performance capabilities, or if you need to handle documents of different sizes.

You enable QVS unbalanced clustering by modifying the *ActivateUnbalancedCluster* configuration. By enabling *ActivateUnbalancedCluster*, it is no longer possible to set CPU affinity and Working Set limits in the QlikView Management Console (QMC). Instead, you manage your cluster's settings by editing the QlikView Server (QVS) *Settings.ini* file in each individual node of your cluster.

Only full **CPU Affinity** is supported when using this feature (100% of cores).

The load balancing algorithm should be activated when setting up an unbalanced cluster. This can be configured to specific needs, see: *Customizing the load balancing algorithm (page 62)*.

> *If you decide to set up an unbalanced QVS cluster, we recommend that you complete the following initial configurations before adding unbalanced nodes to your cluster.*

Setting up a QVS unbalanced cluster

The procedure for setting up a QVS unbalanced cluster varies depending on whether your cluster uses QlikView Web Service (QVWS) or Microsoft IIS (QV Settings Service).

Unbalanced QVS clustering using QVWS

Do the following :

1. Set the *ActivateUnbalancedCluster* configuration to *true* in the *QVManagementService.exe.config* file. By default, the file is located in *C:\Program Files\QlikView\Management Service*.
2. Set the *UnbalancedClusterLoadBalancer* configuration parameter to *true* in the *QVWebServer.exe.config* file. By default, this file is located in *C:\Program Files\QlikView\Server\Web Server*.
3. In the QlikView Management Console, navigate to **System** menu, select **Setup**, select **QlikView Web Servers** from the list of services, go to the **AccessPoint** tab, **Server Connections** and select **Cpu with Ram overload** option in the **Load Balancing** field to take advantage of the algorithm for the unbalanced clustered environment.
   If you want to customize the load balancing algorithm to grant a higher weight to previous loaded documents, set *UnbalancedClusterLoadBalancerLoadedDocWeight* to a higher value than *UnbalancedClusterLoadBalancerCpuWeight* and *UnbalancedClusterLoadBalancerRamWeight*. See *Customizing the load balancing algorithm (page 62)* for how to set the value for *UnbalancedClusterLoadBalancerLoadedDocWeight*.

Unbalanced QVS clustering using Microsoft IIS

If you are using Microsoft IIS, you need to add the following parameters to the IIS settings by using the Internet Information Services (IIS) Manager.

Do the following :

1. Set the *ActivateUnbalancedCluster* configuration to *true* in the *QVManagementService.exe.config* file. By default, the file is located in *C:\Program Files\QlikView\Management Service*.
2. Launch the Internet Information Services (IIS) Manager.
3. On the left navigation menu, click on the site to which the QlikView services are installed. Depending on the settings of your installation, this is either **Default Web Site** or another custom site.
4. In the central pane, in the **ASP.NET** section, double click on **Application Settings**.
5. In the right **Actions** pane, click **Add...** and an **Add Application Setting** window opens. Under **Name:** enter *UnbalancedClusterLoadBalancer* and under **Value:** enter *true*. Click **OK** to confirm the action.
6. In the right **Actions** pane, click **Add...** again. In the **Add Application Setting** window, under **Name:** enter *UnbalancedClusterLoadBalancerCpuWeight* and under **Value:** enter *5*. Click **OK** to confirm the action.
7. In the right **Actions** pane, click **Add...** again. In the **Add Application Setting** window, under **Name:** enter *UnbalancedClusterLoadBalancerRamWeight* and under **Value:** enter *3*. Click **OK** to confirm the action.
8. In the right **Actions** pane, click **Add...** again. In the **Add Application Setting** window, under **Name:** enter *UnbalancedClusterLoadBalancerLoadedDocWeight* and under **Value:** enter *3*. Click **OK** to confirm the action.
9. In the QlikView Management Console, navigate to **System** menu, select **Setup**, select **QlikView Web Servers** from the list of services, go to the **AccessPoint** tab, **Server Connections** and select **Cpu with Ram overload** option in the **Load Balancing** field to take advantage of the algorithm for the unbalanced clustered environment.
   If you want to customize the load balancing algorithm to grant a higher weight to previous loaded documents, set *UnbalancedClusterLoadBalancerLoadedDocWeight* to a higher value than *UnbalancedClusterLoadBalancerCpuWeight* and *UnbalancedClusterLoadBalancerRamWeight*. See *Customizing the load balancing algorithm (page 62)* for how to set the value for *UnbalancedClusterLoadBalancerLoadedDocWeight*.

Customizing the load balancing algorithm

You can customize the weights of the load balancing algorithm if needed. The procedure varies depending on whether your cluster uses QlikView Web Service (QVWS) or Microsoft IIS (QV Settings Service).

Customize the load balancing algorithm for QVWS

If your installation uses QVWS, edit the following settings in the *QVWebServer.exe.config* file, located in *C:\Program Files\QlikView\Server\Web Server*. The procedure requires the QVS *Settings.ini* file to be modified as well.

1. Set the *UnbalancedClusterLoadBalancerCpuWeight* to a value between 0 and 10. A higher value indicates the processing power should be given more weight when the load-balancing algorithm

determines which QVS cluster node is used to open documents.

2. Set the *UnbalancedClusterLoadBalancerRamWeight* to a value between 0 and 10. A higher value indicates the RAM performance should be given more weight when the load-balancing algorithm determines which QVS cluster node is used to open documents.
3. Set the *UnbalancedClusterLoadBalancerLoadedDocWeight* to a value between 0 and 10. A higher value indicates the number of previous loaded documents on a QVS cluster node should be given more weight when the load-balancing algorithm determines which cluster is used to open documents.
4. Make sure that no CPU Affinity settings are present in the local node's QVS *Settings.ini* files. By default, this file is located in *C:\ProgramData\QlikTech\QlikViewServer*.
   Remove the following if present:
   ```
   MaxCoreMask
   MaxCoreMaskHi
   MaxCoreMaskGrp1
   MaxCoreMaskGrp1Hi
   MaxCoreMaskGrp2
   MaxCoreMaskGrp2Hi
   MaxCoreMaskGrp3
   MaxCoreMaskGrp3Hi
   ```
5. Working Set limit Low and High will by default be set to 70 and 90 respectively (usage of RAM in percent). Remove old settings if necessary or change to customized levels in the local node QVS *Settings.ini* files:
   ```
   WorkingSetSizeLoPct=nn
   WorkingSetSizeHiPct=nn
   ```
6. Restart all systems involved.

Customize the load balancing algorithm for Microsoft IIS

To customize the load balancing algorithm for an installation that uses Microsoft IIS as web server, edit the following settings by using the Internet Information Services (IIS) Manager.

1. Launch the Internet Information Services (IIS) Manager.
2. On the left navigation menu, click on the site to which the QlikView services are installed. Depending on the settings of your installation, this is either **Default Web Site** or another custom site.
3. In the central pane, in the **ASP.NET** section, double click on **Application Settings**.
4. Select the *UnbalancedClusterLoadBalancerCpuWeight* setting. In the right **Actions** pane, click **Edit...** and an **Edit Application Setting** window opens. Under **Value:** enter a value between 0 and 10. A higher value indicates the processing power should be given more weight when the load-balancing algorithm determines which QVS cluster node is used to open documents. Click **OK** to confirm the action.
5. Select the *UnbalancedClusterLoadBalancerRamWeight* setting. In the right **Actions** pane, click **Edit...** and an **Edit Application Setting** window opens. Under **Value:** enter a value between 0 and 10. A higher value indicates the RAM performance should be given more weight when the load-balancing algorithm determines which QVS cluster node is used to open documents. Click **OK** to confirm the action.

6. Select the *UnbalancedClusterLoadBalancerLoadedDocWeight* setting. In the right **Actions** pane, click **Edit...** and an **Edit Application Setting** window opens. Under **Value:** enter a value between 0 and 10. A higher value indicates the number of previous loaded documents on a QVS cluster node should be given more weight when the load-balancing algorithm determines which cluster is used to open documents. Click **OK** to confirm the action.
7. Restart all systems involved.

Clustering QlikView Publisher

This chapter provides an overview of QlikView Publisher and how to use it in a clustered deployment for scalability, resilience, or both. This chapter also addresses the architectural and installation requirements and the options for building a clustered and resilient QlikView Publisher deployment.

Introduction

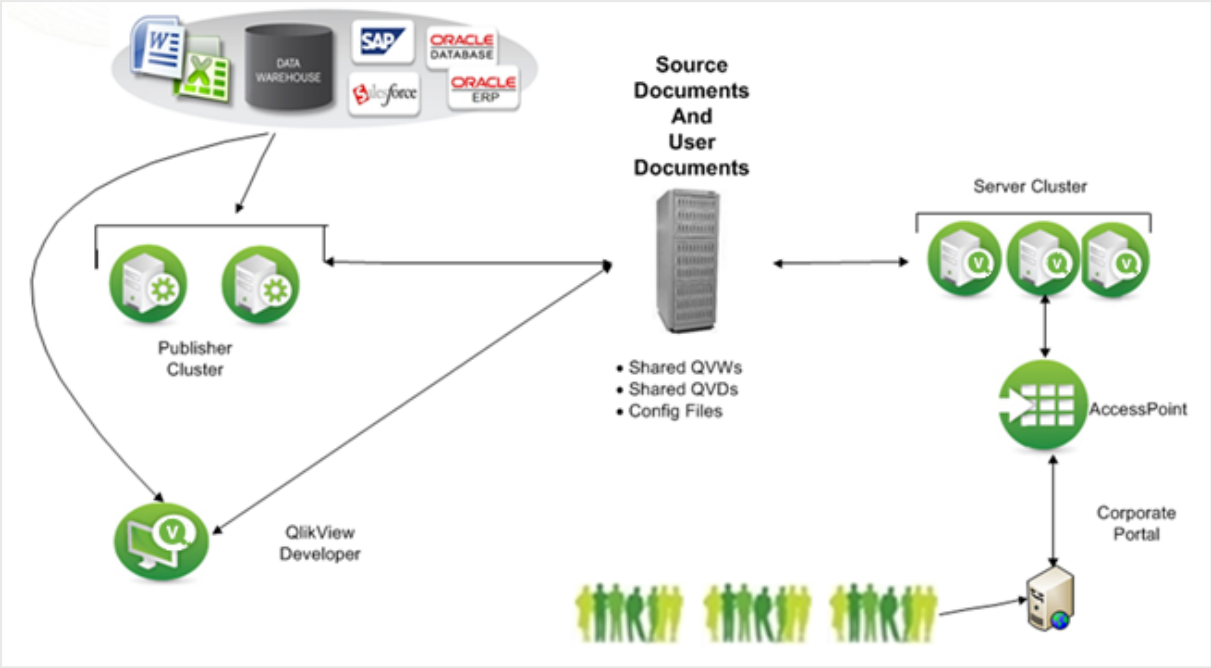
QlikView Publisher is an optional module for QlikView Server that enables scheduling, administration, and management tools that provide a single point of control for QlikView analytics applications and reports. Administrators can schedule, distribute, and manage security and access for QlikView applications and reports across the enterprise.

QlikView Publisher performs the following main functions:

- It loads data directly from data sources defined in connection strings in the source QlikView document files.
- It is used as a distribution service to “reduce” data and applications from source QlikView document files based on various rules (for example, user authorization or data access) and distribute these newly-created documents to the appropriate QlikView Servers or as static reports via email.
- When using QlikView Publisher, only Publisher has access to the source documents folder and the data sources for data load and distribution. The source documents and data are not accessible by QlikView users.

By deploying a clustered architecture, QlikView Publisher achieves scalability and/or resilience using web services technology. Administrators can cluster services together to provide load balancing. Native support for SNMP enables integration with enterprise system monitoring tools. External enterprise scheduling tools can trigger Publisher tasks using web service calls. Tasks can also be scheduled and executed on demand by QlikView administrators.

The figure below shows a two-server, clustered QlikView Publisher where each server is configured for processing different tasks and load balancing. The figure also includes a three-server, clustered QlikView Server that uses QlikView AccessPoint for load balancing. Documents created by QlikView Developer are stored in the source documents folder. QlikView Publisher tasks are used to retrieve data and store the result in the user documents folder.



To see how to set up an unbalanced distribution service cluster, see *Unbalanced QlikView Publisher Clustering (page 75)*

Source Documents

The source documents contain a) scripts within QlikView document files to extract data from various data sources (for example, data warehouses, Microsoft Excel files, SAP, and Salesforce.com), b) the actual binary data extracts themselves within *.qvd* files, or c) a binary load from another QlikView document file, inheriting its data model in one line of code.

The QlikView source documents, created using QlikView Developer, reside in the following folder:

- Windows Server 2008 and later: *\ProgramData\QlikTech\SourceDocuments*. This is the default QlikView location for Windows Server 2008 and later.

User Documents

The user documents folder is the repository used by QlikView Server. The folder is located at:

- Windows Server 2008 and later: *\ProgramData\QlikTech\Documents*. This is the default QlikView location for Windows Server 2008 and later.

Tasks

Tasks are created by administrators for data distribution and data reloads. Tasks are stored in the QlikView Publisher repository as a collection of XML files or in an SQL Server database. When a task is executed, QlikView Publisher invokes QlikView Batch (QVB), which is comparable to QlikView Desktop without the user interface.

> *QlikView Batch (QVB) does not support graphical or user input objects. This means that QVB cannot reload documents that, for example, contain scripts that require user input.*

QVB reloads the documents, which are stored in the source documents folder(s) and creates an associative QlikView database, which is stored within each document. The QVB performs the reload by retrieving the data described by the load script from the data sources. QlikView Publisher distributes the documents to the user documents folder for QlikView Server using the encrypted QVP protocol, to a cloud environment, a mail server, and/or a file folder. QlikView Publisher can use the Directory Service Connector (DSC) to determine where and to whom the documents are to be distributed.

Why Cluster QlikView Publisher?

The role of Publisher in the QlikView solution is to distribute and refresh data by criteria set by the QlikView administrator. To accomplish this, Publisher executes many tasks, either scheduled or on demand. A Publisher task is the smallest entity that can be distributed in a cluster; a single task cannot be divided and executed in parallel on multiple cluster nodes. Clustering the Publisher service on more than one server enables the administrator to distribute multiple tasks to multiple servers operating in parallel using the Publisher load balancing algorithm. This means Publisher clusters can be used to increase the scalability, availability, and serviceability of data distribution and reloading.

In addition, a Publisher cluster license enables the configuration of Publisher services in clusters and standalone Publisher services. For example, a Publisher cluster can be used in a corporate office to handle large volumes of data and tasks, whereas a single Publisher service can be used in an associated manufacturing plant where the Publisher only needs to distribute documents using the manufacturing data source.

By clustering QlikView Publisher, the following objectives can be met:

- Horizontal scalability
- Resilience

Horizontal Scalability

Horizontal scaling of hardware provides the ability to increase the resources of the QlikView deployment. By adding additional hardware servers, the workload of QlikView Publisher can be increased. The clustered Publisher servers can then be configured to load balance the QlikView tasks.

For example, on a certain hardware server, QlikView Publisher can process eight concurrent tasks. When the resource needs increase, the QlikView Publisher service can grow as needed. By adding an additional QlikView Publisher service on a new hardware server, the deployment can handle up to sixteen concurrent tasks by configuring the additional server in a Publisher cluster deployment. In this scenario, the first eight tasks are allocated to Server A and the second eight tasks to Server B. Alternatively, if the servers are clustered, the tasks can be load balanced over the two servers.

Resilience

When the number of tasks in the deployment increases, the window for completing the tasks in time becomes increasingly important. Clustering the QlikView distribution services provides for resilience in the deployment. In the case above, where a single server can support 100 concurrent tasks, an additional server can be deployed (for a total of three servers) in order to build resilience into the deployment. If a server is lost (for example, due to a hardware failure or network connection issues), the resilient cluster still

supports up to 200 tasks. Having all three servers as active nodes helps reduce response times by not running all servers at 100% of their capacity. It also limits the number of tasks and task chains affected if a node is lost.

Requirements for a Clustered QlikView Publisher Deployment

The following high-level requirements must be fulfilled for a clustered QlikView Publisher deployment:

- Clustered QlikView Publisher license key
- Shared network storage
- Load balancing strategies

Clustered QlikView Publisher License Key

In a clustered environment, the QlikView Publisher servers are installed with the same license key. This can be verified by examining the following entry in the License Enabler File (LEF):

`PRODUCTLEVEL;30;;` (where `30` is the code for QlikView Publisher)

`NUMBER_OF_XS;N;;` (where `N` is the number of allowed QlikView Distribution Services)

The servers in a clustered QlikView Publisher deployment share configuration and license information among themselves via the shared storage, so configuration and license management only needs to be performed once in the QMC for all nodes.

Shared Network Storage

In QlikView shared network storage can be used for storing source (.qvf or .qvw) and cluster files (notification, tasks, triggers, logs etc) that need to be accessed in QlikView Publisher cluster.

The requirements for a shared network storage in a QlikView Publisher cluster are the following:

- The network storage must be hosted on a Windows-based file share.
- QlikView Publisher supports the use of a SAN (NetApp, EMC, etc.) mounted to a Windows Server 2008 R2 (or later) and then shared from that server. Storage presented to a server via a SAN must appear as locally attached storage. If SAN storage is used for Publisher, any distributed data that is accessed by QlikView Server should not reside on the SAN storage.
- The QlikView Publisher nodes in the cluster must have network latency below 4 milliseconds to connect to the file share server. Performance can degrade if this is not the case.
- A maximum of two nodes in a QlikView Publisher cluster can share the same shared storage. If more than two QlikView Publisher nodes are required, it is recommended to deploy the additional publisher nodes in an additional cluster. The QlikView Management Console can manage multiple publisher clusters.
- The bandwidth to the file share must be appropriate for the amount of traffic on the site. The frequency and size of the documents being saved after reloading, and opened into memory, drives this requirement. 1 Gigabit networking is suggested.
- The following shared storage options are not supported:
    - Shared storage systems based on Linux OS are not supported. This includes systems supporting SMB file sharing protocol or NTFS disk drive format .

- Windows-based shared storage systems that rely on CIFS file sharing protocol are not supported.
- QlikView does not support Windows Distributed File System (DFS).

Load Balancing Strategies

Load Balancing

The load balancing is determined by an internal ranking system based on the amount of memory available and the CPU use. Qlik recommends using the default settings, since they have been extensively tested.

To change the default settings, edit the configuation file, *QlikViewDistributionService.exe.config*. The key is written in JavaScript:

```
<add key="LoadBalancingFormule" value="(AverageCPULoad*400) + ((MemoryUsage / TotalMemory) *
300) + ((NumberOfQlikViewEngines / MaxQlikViewEngines)*200) + (NumberOfRunningTasks*100)"/>
```

where:

- `AverageCPULoad`: Average CPU load for all running QVBs.
- `MemoryUsage`: Total memory use for the entire application.
- `TotalMemory`: Total amount of memory on the server.
- `NumberOfQlikViewEngines`: Number of QlikView engines currently used.
- `MaxQlikViewEngines`: Configured value for the maximum number of QlikView engines.
- `NumberOfRunningTasks`: Number of tasks currently running.

Simultaneous Tasks

By default, four QlikView tasks can execute simultaneously on a node. The recommended maximum is eight simultaneous tasks per node. If more than ten tasks have to be executed simultaneously on a node, modifications are necessary in the Windows registry to change the desktop heap size to allow for more simultaneous tasks.

> *A large-scale server is required for executing ten or more simultaneous tasks. Alternatively, add additional servers for Publisher tasks.*

Proceed as follows to change the number of tasks allowed to execute simultaneously:

1. Backup the Windows Server registry.
2. Locate the following Windows Server registry setting:

   ```
   HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Session\Manager\SubSystems\Windows
   %SystemRoot%\system32\csrss.exe ObjectDirectory=\Windows
   SharedSection=1024,3072,512 Windows=On SubSystemType=Windows
   ServerDll=basesrv,1 ServerDll=winsrv:UserServerDllInitialization,3
   ServerDll=winsrv:ConServerDllInitialization,2 ProfileControl=Off
   MaxRequestThreads=16
   ```

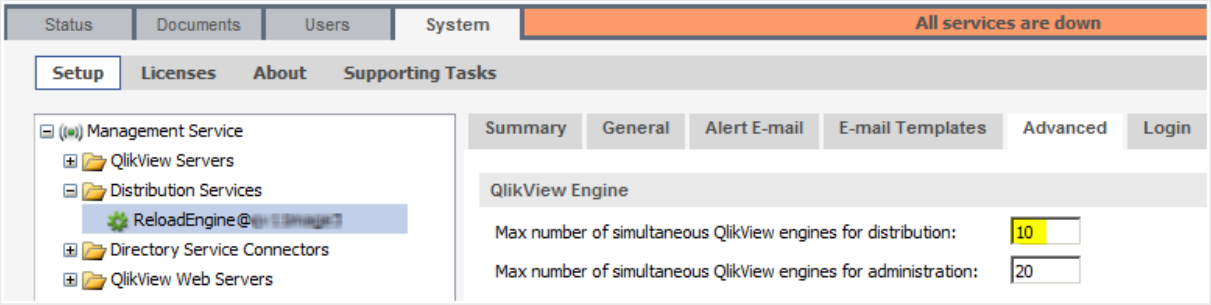
   The default value for `SharedSection` is 1024,20480,768 for 64-bit (x64).

3. Change the desktop heap size by setting `SharedSection` to 1024,20480,2048:

   ```
   HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Session\Manager\SubSystems\Windows
   %SystemRoot%\system32\csrss.exe ObjectDirectory=\Windows
   SharedSection=1024,20480,2048 Windows=On SubSystemType=Windows
   ```

```
ServerDll=basesrv,1 ServerDll=winsrv:UserServerDllInitialization,3
ServerDll=winsrv:ConServerDllInitialization,2 ProfileControl=Off
MaxRequestThreads=16
```

4. Save the registry changes and restart the machine.
5. Change the **Max number of simultaneous QlikView engines for distribution** setting in QMC to the number of engines needed.

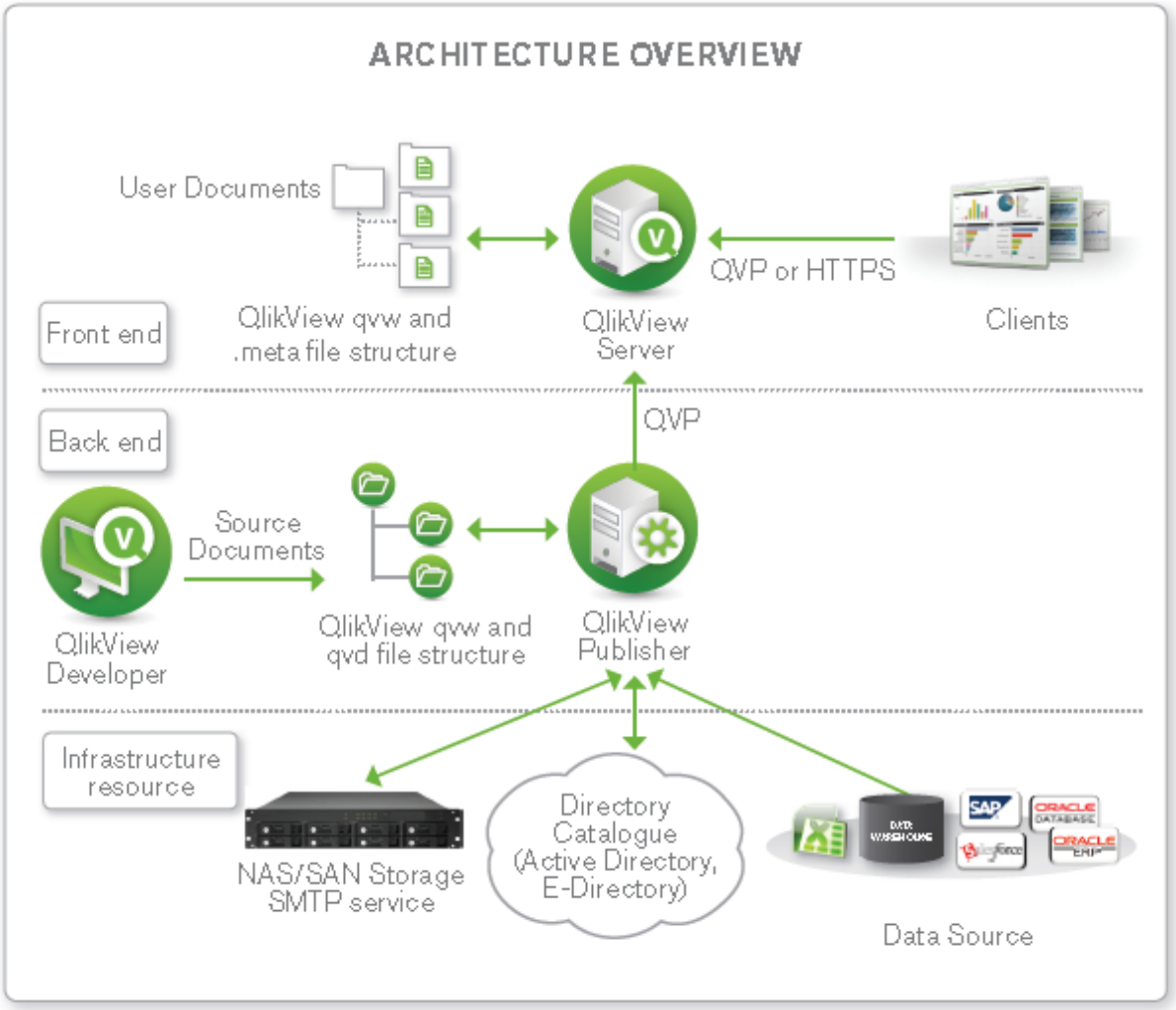


Security

QlikView Publisher provides access to QlikView applications and data. It is therefore important to integrate QlikView Publisher with the enterprise security solutions in addition to the standard security features of QlikView Server.

QlikView Publisher is viewed as a backend process within the QlikView solution. From a security perspective, it is important to understand that the frontend does not have any open ports to the backend. The frontend does not send any queries to data sources on the backend, nor do any of the user documents (`.qvf` or `.qvw` files) contain any connection strings to data sources located on the backend. End users can only access QlikView documents that exist on the frontend. Within the backend, the Windows file system is always in charge of authorization; QlikView is not responsible for access privileges.

The figure below shows a simplified view of a standard QlikView deployment containing the location of the QlikView products and the data and applications.



Directory Services

To provide security for QlikView documents, QlikView Publisher can connect to an external directory service (for example, Active Directory, LDAP, a database, or other sign-on solutions). The external directory service is an authentication source with which QlikView has a trust relationship.

QlikView provides a built-in Directory Service Provider (DSP) for Active Directory that allows QlikView administrators to assign Active Directory user privileges to QlikView documents or portions thereof. QlikView Publisher leverages this built-in provider to provide direct integration with, and support for, Active Directory.

QlikView also provides a means of creating a Configurable LDAP for other directory services. A Configurable LDAP enables QlikView administrators to grant privileges to users authenticated by any authentication system other than Active Directory.

QlikView Server Authorization Modes

QlikView Server provides two mutually exclusive options for authorizing access to QlikView documents. Depending on the authorization mode of QlikView Server (NTFS or DMS), Publisher populates the appropriate Access Control List (ACL) when assigning rights to a document. In case of NTFS

authorization, Publisher populates a standard NTFS ACL when sending documents to QlikView Server. In case of DMS authorization, Publisher populates an ACL contained within a *.meta* file associated with the application.

Static Data Reduction

Data reduction is a security mechanism that allows application data to be purged from a QlikView application in accordance with row-level security settings. QlikView Publisher can automate data reduction independently of the applicable security scenario. However, Publisher allows an administrator to configure data reduction based on users or groups defined within any external authentication source available through a custom or Active Directory DSP. Publisher performs the data reduction using the “loop and reduce” functionality in QlikView. The Publisher data reduction should not be confused with the dynamic data reduction associated with Section Access.

Configuring QlikView Publisher Clustering

The instructions in this section are valid for Windows Server 2008 R2 and later.

Requirements

The following requirements must be fulfilled before starting the QDS cluster configuration:

- A QlikView Publisher license that supports more than one QDS. The Publisher LEF must contain the entry `NUMBER_OF_XS;N;;`, where `N` is 2 or higher.
- QlikView AccessPoint (based on QlikView Web Server or Microsoft IIS), QlikView Management Service (QMS), QlikView Server (QVS), and DSC are already installed in the QlikView system in the network.
- A domain user to run the QlikView services on every machine is available.
- A shared storage device; Qlik recommends a shared device mounted as a Windows-based file share.

  All QDS cluster nodes need read and write access to the following, centrally stored data:

  - QlikView Publisher status, configuration, and log files
  - QlikView source documents

Step-by-step Instructions

Prepare the Shared Storage Device

Create folders for the files accessed by every Publisher cluster node:

- *<server1>**ProgramData**QlikTech**DistributionService* (application folder)
- *<server1>**ProgramData**QlikTech**SourceDocuments* (source documents folder)

Prepare the Cluster Nodes

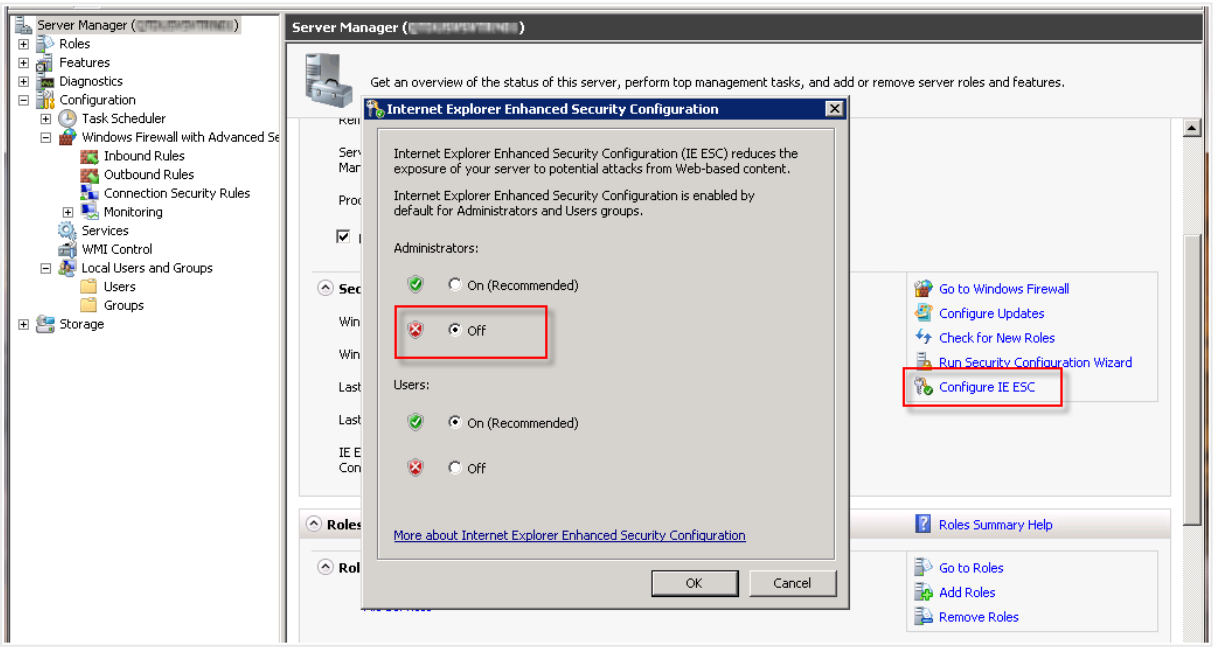
Proceed as follows on each planned QDS cluster node:

1. Login as administrator.
2. Configure the firewall to secure the QlikView solution. The QlikView services require the ports listed in the table below to be “opened”.

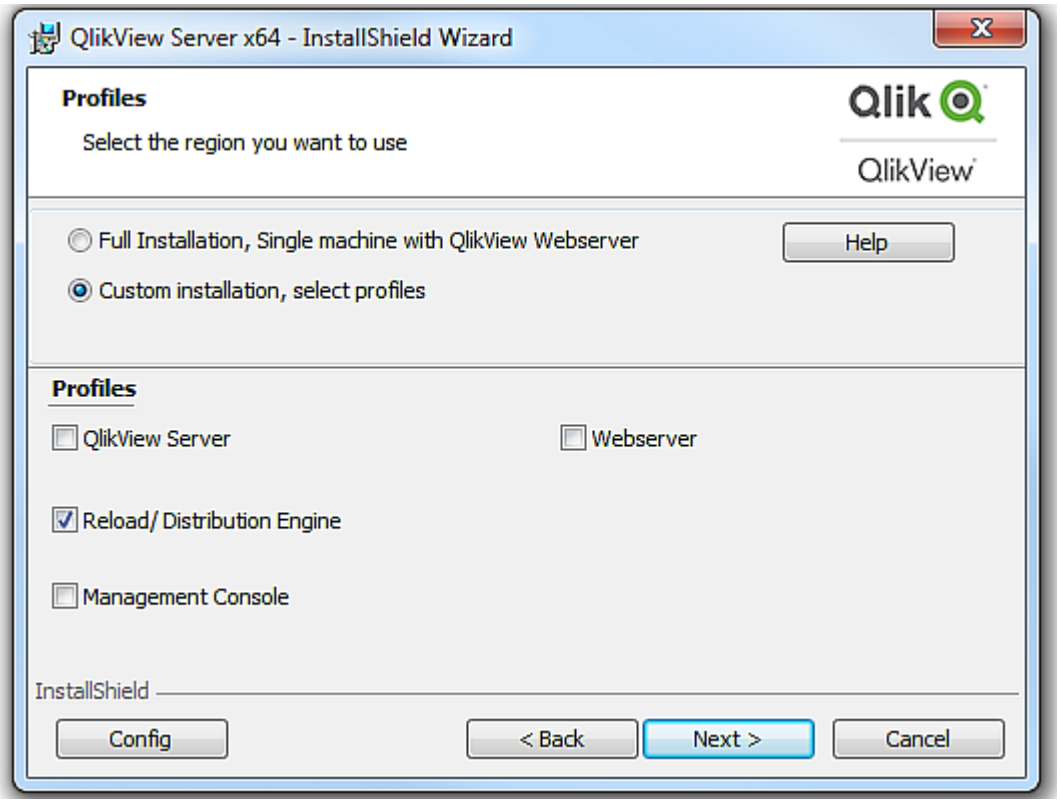
Required open ports

Service	Port
QDS (Publisher) (required for Publisher)	4720/TCP
DSC (required for Publisher)	4730/TCP
QMS (required for Publisher)	4780/TCP
QlikView Web Server/Microsoft IIS configuration	4750/TCP
QVS configuration	4749/TCP
QVP communication	4747/TCP
QMS (EDX calls) (required for Publisher)	4799/TCP

3. Deactivate the Internet Explorer Enhanced Security Configuration for administrators. By default, Windows Server 2008 and later ship with this configuration enabled, which is basically a locked down version that adds a bit of extra security to the servers for web browsing. When the configuration is enabled, it may cause problems in viewing the QMC and service content. The Internet Explorer Enhanced Security Configuration can be left turned on, but if any issues arise, turn off the feature for the Administrators group.



4. Add the domain user that is used to run the QlikView services to the Local Administrators Group.
5. Start the QlikView 64-bit (x64) server setup and select **Custom installation, select profiles**. Then select the **Reload/Distribution Engine** feature and install it on each node where Publisher is to reside.

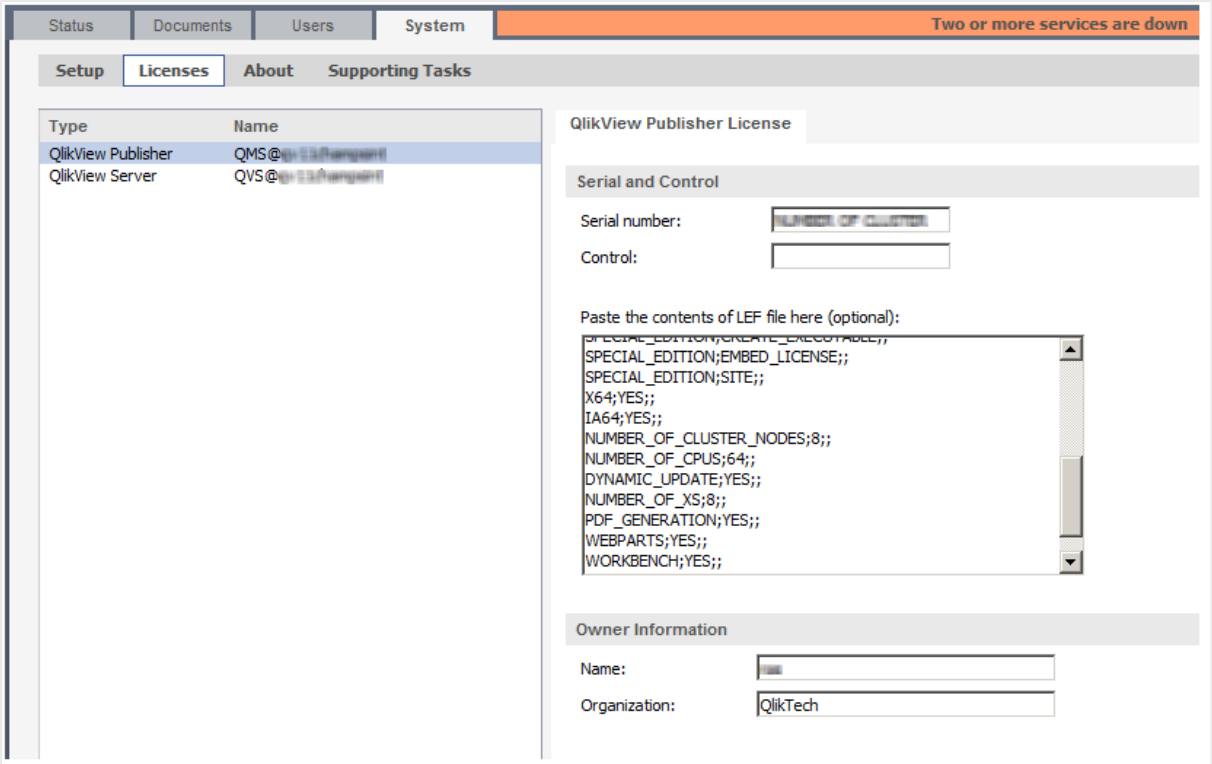


6. Enter the QlikView service account credentials.
7. Finish the setup and restart the system immediately.

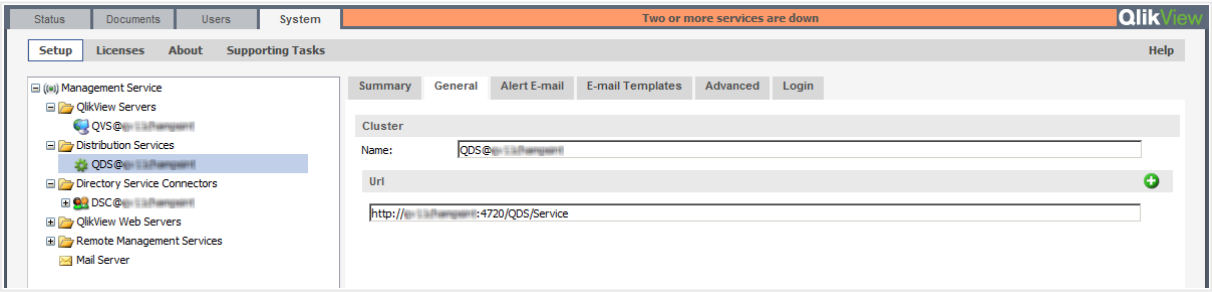
Configuring QDS Cluster in the QMC

Proceed as follows to configure a QDS cluster in the QMC:

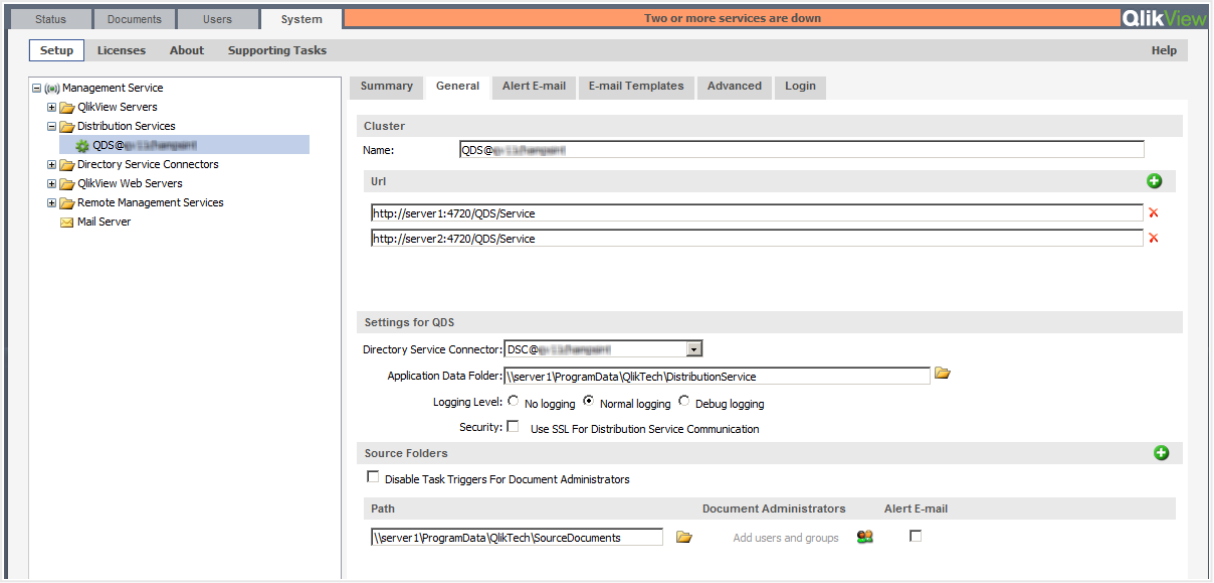
1. Open QMC and register the QlikView Publisher license with the activated cluster nodes.



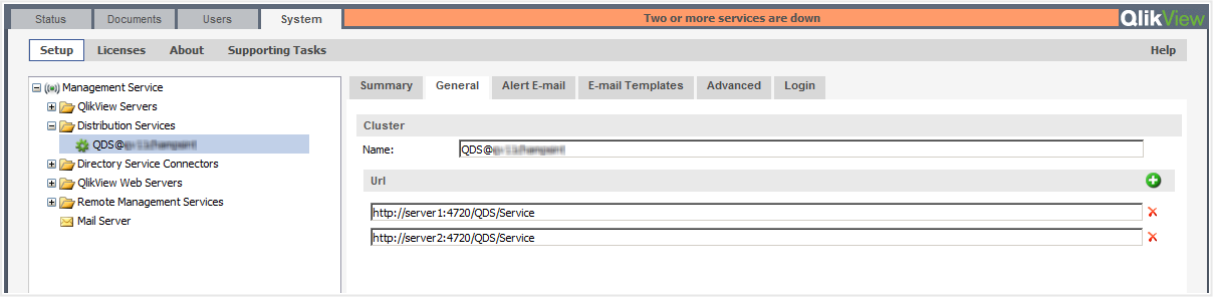
2. On the **System**>**Setup** tab, add the first QDS cluster node under **Distribution Services**.



3. Switch the **Application Data Folder** and the **Source Folders** to the shared device folder paths using UNC syntax.



4. Click **Apply** and restart the QDS manually.
5. Add each additional QDS cluster node in URL format.



6. Click **Apply** and restart the QDS on all nodes manually.

Unbalanced QlikView Publisher Clustering

This chapter discusses the requirements and options for building a clustered and unbalanced QlikView Distribution Service (QDS) deployment. By default, a QlikView Distribution Service cluster requires that all nodes are equal regarding CPU, cores, and RAM.

A QlikView Publisher license is necessary in order to set up clusters. For more information on QlikView Publisher, see the *Clustering QlikView Publisher (page 64)* page.

The QlikView load balancing capabilities are included in the QlikView Management Console. This chapter also discusses how to make this component efficient using distribution groups.

What is a QDS Publisher Group?

A publisher group is a subset of a QDS cluster. Each publisher group is given a unique name, and the set of QDS nodes (one or more) that are included in this group. A node may exist in any number of publisher groups (zero or more).

Each task is assigned to none or one of these publisher groups. A task assigned to a publisher group is called a Dedicated Task and may only be executed by one of the QDS nodes included in this group. A task not assigned to any publisher group is called a Regular Task and may be executed by any of the QDS nodes (but may be prevented to run on a QDS in a publisher group under certain circumstances).

The QDS cluster must be setup and functional prior to activating this feature.

To activate this feature, make a copy of *DistributionGroupDefinition.Template* in *C:\ProgramData\QlikTech\ManagementService\DistributionGroups* and name it *DistributionGroupDefinition.xml*. Restart the QMS service manually on the QDS cluster node.

QDS publisher group configuration

You can configure the distribution group using the following settings in the *DistributionGroupDefinition.xml* file.

```
<DistributionGroupDefinition>
  <QDSSettings>
    <QDS QDIdentifier = "d033930c-0000-e6ec-1519-f3c628a443ae">
      <MaxSimultaneousQvbs>4</MaxSimultaneousQvbs>
      <MaxSimultaneousReaderQvbs>2</MaxSimultaneousReaderQvbs>
      <DedicatedQvbs>1</DedicatedQvbs>
      <RunDedicatedTaskAlone>True</RunDedicatedTaskAlone>
      <GraceTimeMinutes>30</GraceTimeMinutes>
      <DistributionGroups>
        <Group>Group A</Group>
        <Group>Group B</Group>
      </DistributionGroups>
    </QDS>
  </QDSSettings>
</DistributionGroupDefinition>
```

For each QDS in a publisher group, the following should be configured:

- *MaxSimultaneousQvbs* - The maximum number of simultaneous QlikView Batch instances (default 4).
- *MaxSimultaneousReaderQvbs* - The maximum number of simultaneous QlikView Batch readers (default 20).
- *DedicatedDistributionQvbs* - The number of dedicated QlikView Batch instances (default 0).
- *RunDedicatedTaskAlone* - Whether to run dedicated tasks alone or not (default false).
- *GraceTimeMinutes* - If *RunDedicatedTaskAlone* is set to *True*and this setting means that no regular task may be started by this QDS within number of minutes or less until the nearest dedicated task is scheduled (default 0).

The following table provides an example of the number of regular and dedicated tasks that may be started based number of dedicated task currently running if *MaxSimultaneousQvbs* is set to *4* and *DedicatedQvbs* is set to *2*.

Numbers of tasks

Number of dedicated tasks running	Number of new dedicated tasks that may be started	Number of regular tasks that may be started
0	4	2
1	3	2
2	2	2
3	1	1
4	0	0

A QVB should always be available for dedicated tasks if the *RunDedicatedTaskAlone* option is set to *True*. The following table provides an example of the number regular and dedicated tasks that may be started based number of dedicated task currently running if *MaxSimultaneousQvbs* is set to *4*, *DedicatedQvbs* is set to *2* and *RunDedicatedTaskAlone* is set to *True*.

Numbers of tasks

Number of dedicated tasks running	Number of new dedicated tasks that may be started	Number of regular tasks that may be started
0	4	2
1	3	0
2	2	0
3	1	0
4	0	0

Task Configuration

Once you have created a publisher group, the feature is active and each existing task is considered to be a regular task. When creating a new or editing an existing task, a **Publisher Group** dropdown is available on the Source Document's General tab.

This drop-down contains the names of all publisher groups. If a publisher group is assigned to a document, all task associated with this document dedicated. Select **<any>** from the publisher groups dropdown to make tasks associated with a document regular. A regular task may be executed on any node.

QlikView Server Extensions

Adding Extensions to QlikView Server

To run QlikView Extensions on a QlikView Server, the contents of the *Extensions* folder have to be copied from *%UserProfile%\AppData\Local\QlikTech\QlikView\Extensions\Objects* to the *%ProgramData%\QlikTech\QlikViewServer\Extensions\Objects* folder on the server.

If the path to the extensions is changed (for example, to a common place for all servers in a cluster), that path must be used instead. Note that the path set corresponds to *%UserProfile%\AppData\Local\QlikTech\QlikView\Extensions* (that is, it does not include *\Objects*).

Configuring IIS for Custom Users

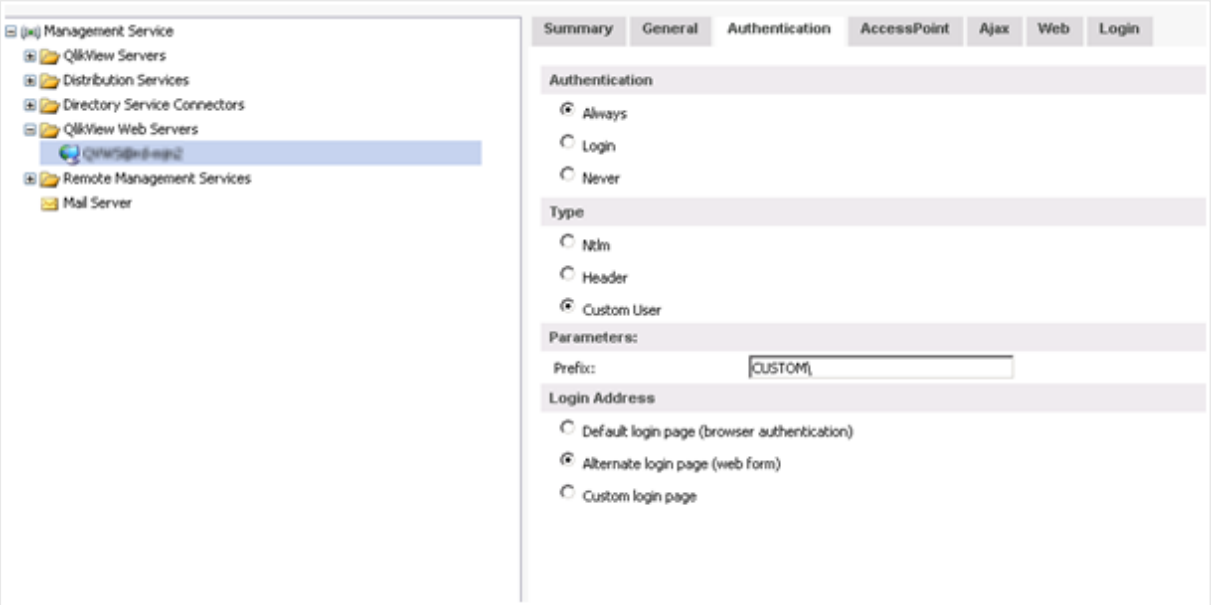
When using Microsoft IIS as web server for Custom Users, configuration is needed.

Proceed as follows to configure IIS for Custom Users:

1. In QlikView Management Console, change the parameters on the **System**>**Setup**>**Authentication** tab in accordance to the following:

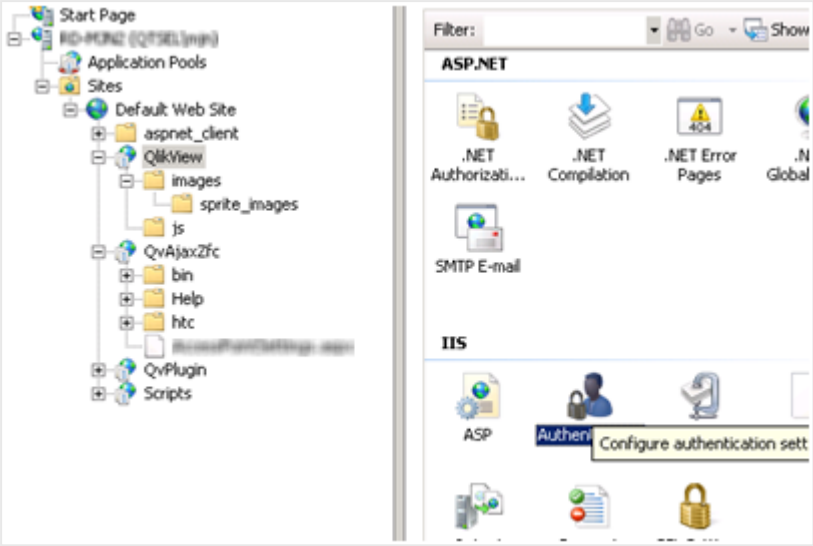
QlikView Management Console parameters

Parameter	Changes
Authentication	Always
Type	Custom User
Parameters	CUSTOM\
Login Address	Alternate login page (web form)



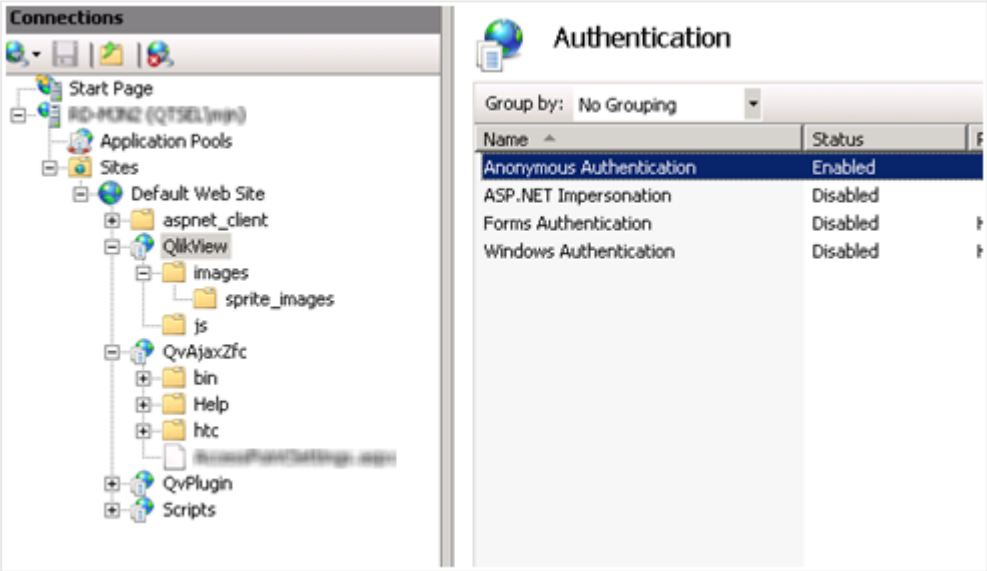
Authentication tab

2. Select the `QlikView` virtual folder and then **Authentication**.



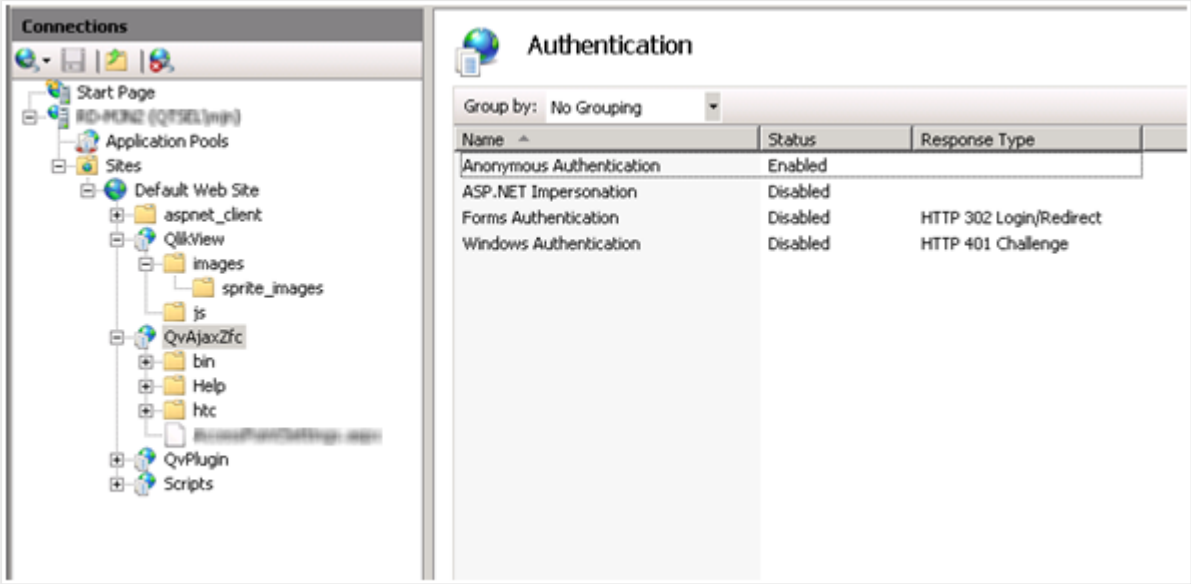
Selecting Authentication

3. Disable **Windows Authentication** and enable **Anonymous Authentication**.



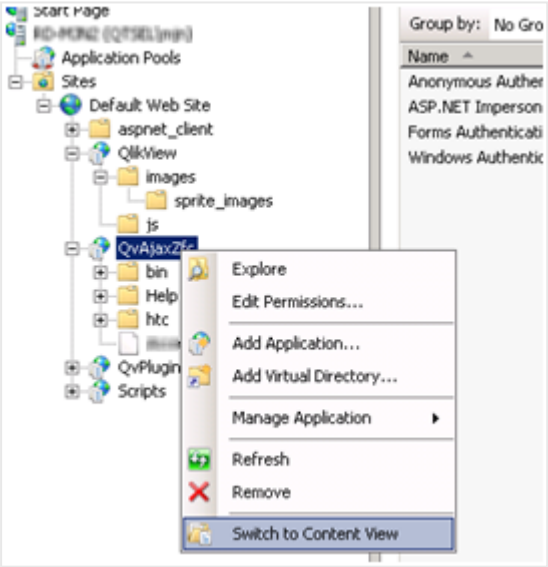
Enabling Anonymous Authentication for the QlikView virtual folder

4. Select the `QvAjaxZfc` folder and then **Authentication**.
5. Disable **Windows Authentication** and enable **Anonymous Authentication**.



Enabling Anonymous Authentication for the QvAjaxZfc folder

6. Right-click `QvAjaxZfc` and select **Switch to Content View**.



Selecting Switch to Content View

7. The configuration of IIS for the Custom User is complete.

QlikView Triggering EDX Enabled Tasks

QlikView Event Driven Execution (EDX) allows you to start tasks in the QlikView Publisher using an external event as the trigger.

To set up an EDX task, you must use the QlikView Management Service API (QMS API). The user making the request calls must be a member of the QlikView Administrators local group or of the QlikView EDX local group. The QlikView Administrators group is created during the installation of QlikView Server, but the QlikView EDX group must be created manually in **Computer Management**. Members of the QlikView EDX group can only trigger EDX-enabled tasks.

Creating the QlikView EDX group

Do the following:

1. Open **Local Users and Groups** from **Computer Management**.
2. Expand the group section and in the toolbar select **Action** > **New Group...**
3. Enter "QlikView EDX" as the group name and select **Create**.

Creating an EDX task

To create an EDX task, use the following signature:

```
TriggerEDXTaskResult TriggerEDXTask(Guid guid, string taskNameOrId,
                                    string password, string variableName,
                                    List<string> variableValues)
```

EDX task parameters

Parameter	Purpose
`guid`	ID of the QlikView Distribution Service (QDS) where the task is defined.
`taskNameOrId`	Task name or ID of the task in string format.
`password`	Password (if required by the task).
`variableName`	Variable name (if required by the task).
`variableValues`	List of values for the variable.

The returned result contains information on whether the task was successfully started or not.

The example below shows how to trigger a task and wait until it has finished or until a certain amount of time has passed.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Threading;
using QMSAPI;

class Program
{
        static void Main(string[] args)
        {
                try
                {
                        // create a QMS API client
                        IQMS apiClient = new QMSClient();
```

```
                                // retrieve a time limited service key
                                ServiceKeyClientMessageInspector.ServiceKey =
                                apiClient.GetTimeLimitedServiceKey();

                                if (qdsService != null)
                                {
                                        //Trigger the task
                                        TriggerEDXTaskResult result =
                                        apiClient.TriggerEDXTask(qdsService.ID, "PauseEDX", "edx", "", new Lis
());

                                        EDXStatus executionStatus = null;

                                        //wait until the task is completed or 60 seconds has passed.
                                        SpinWait.SpinUntil(() =>
                                                {
                                                        System.Threading.Thread.Sleep(1000);
                                                        Console.WriteLine("Checking the task...");

                                                        //Get the current state of the task.
                                                        executionStatus =
                                                        apieClient.GetEDXTaskStatus(qdsService.ID, result.Exec

                                                        //Return true if the task has completed.
                                                        return executionStatus !=
                                                        null && executionStatus..TaskStatus == TaskStatusValue
                                                }, 60 * 1000);

                                                //write the result
                                                if (executionStatus != null)
                                                        Console.WriteLine(executionStatus.TaskStatus);
                                }
                        }
                        catch (Exception ex)
                        {
                                Console.WriteLine("An exception occurred: " + ex.Message);
                        }
                        //wait for user to press any key
                        Console.ReadLine();
                }
        }
```

The example comes from the QMS API documentation, which is installed as part of the QlikView Management Console (QMC). It contains detailed information on the available methods and how to get started with the QMS API.

Cleaning and converting the shared files

The QlikView shared file cleaning tool is a command line tool that allows system administrators to verify (analyze) and purge (repair) shared files. This tool can also be used to convert between different shared file formats, see Converting the shared files. You can invoke it by running the QlikView Server executable (QVS.exe) with special parameters.

There are two modes available with the cleaning tool, each is specified by a different command-line parameter.

Verify mode

Use the `-v` parameter to analyze the shared file specified in the command-line. During analysis, the cleaning tool detects if there is one or more invalid or corrupted object entries. The QVS then logs as much information as possible about the invalid entries.

Purge mode

Use the `-p` parameter to verify the shared file and then create a new shared file with the corrupt entries removed. This clean version is placed into the same folder as the original. The new file uses the suffix *_clean* after the format (*.Shared* or *.Tshared*), and the original shared file is not overwritten. You can then decide to replace the original shared file with the clean version.

Converting the shared files

When you create shared files, you can save them in original or transactional format. The original format is recognizable by the *.Shared* ending, while the transactional shared file format ends with *.TShared*. A shared file using the transactional *.TShared* format is more reliable in case of failures, such as network issues, power outages, or insufficient storage space on disk. We recommend to use the *.TShared* format for files larger than 2 GB, because this format can handle file size up to 16 EB (exabyte).

You can use the two different formats, original and transactional, simultaneously for different applications on the same server. However, only one format (either *.Shared* or *.TShared*) should be used in a given application. You can decide which format to use when creating a new shared file by configuring the *Settings.ini* file. For QlikView Server, the *Settings.ini* file is located in *C:\ProgramData\QlikTech\QlikViewServer*.

Set the file format:

`DefaultBlobDbType=0`
With this setting, the *.Shared* format is used when creating new shared files.

`DefaultBlobDbType=1`
With this setting, the *.TShared* format is used when creating new shared files.

You can also convert the shared files using the QlikView shared files cleaning command as shown in the tab below, and in the example n.4 in the Examples section at the bottom of the page.

Setting and changing ownership of shared file content

You can change the owner of server objects with QMC, but for some object types (“DocumentContent”, “InputFieldValues” and “ObjectContent”) ownership cannot be changed this way. In this case you need to use the cleaning tool to change ownership, using the `-so` (set ownership) or `-ro` (replace ownership) parameters. These parameters should be used in purge mode.

Cleaning tool command format

The cleaning tool command format is as follows:

```
"<QVS_executable_path>" -x "<Shared_file_path>" <Cleaning_tool_mode> <Output format>
<Ownership> <Delete_user_entries> [-l "<Log_folder_path>"] [-rBM <BM_size>]   [-o "<Shared_
file_save_path>"]
```
The following table describes each command parameter.

Cleaning tool command parameters

Parameter	Description
`QVS_executable_path`	The full path to the system folder containing the QVS executable (QVS.exe).
`-x`	The `-x` parameter tells the QVS to only run the cleaning tool.
`Shared_file_path`	The path to the shared files to clean. It accepts a path to a directory or a path to a file. • If invoked with a path to a folder, the operation applies to all shared files in the folder. • If a single file is specified, the operation is applied to this item only.
`Cleaning_tool_mode`	• `-p` for purge mode • `-v` for verify mode
`Output format`	[Optional] The `-f` (specify output format) parameter allows to use the cleaning tool to convert between shared file formats. The format can be specified as `same`, `orig` or `tx` (e.g. `-f tx`). • `same` the file format of the input file will be used • `orig` the original *.Shared* format is used as output format • `tx` the *.TShared* (transactional file) format is used as output format When the format (`-f` parameter) is not specified, the default option is `same`.
`Ownership`	• `-so user` to set ownership • `-ro from_user to_user` to replace ownership

Parameter	Description
`Delete_user_entries`	• `-du0` user deletes non-shared entries from the user • `-du1` user deletes all entries from the user This field accepts a path to a file if more than one user needs to be removed • `-df0` file.txt deletes non-shared entries from the users listed in the file file.txt • `-df1` file.txt deletes all entries from the users listed in the file file.txt To obtain a list of users that have accessed the QlikView servers, the Governance Dashboard application can be used. It is available for free on ≤ Qlik Community (see associated documentation here). The list of users can be easily extracted by exporting to 'csv' format the ListBox '*Athenticated User*' in the Operations/Session sub-tab of the Governance Dashboard. This list can then be edited (keep only the users to be removed from the shared file) and passed on to the Cleaning Tool as an input.
`-l Log_folder_path`	[Optional] If you want to change the location of the generated log file, use `-l` and provide a log folder path.
`-rBM BM_size`	[Optional] The `-rBM` parameter is used to remove large bookmarks from the shared file. All bookmarks larger than <BM_size> (in bytes) will be removed.
`-o Shared_file_save_path`	[Optional] The `-o` parameter is used to change the path to where shared files are saved.

Using the shared file cleaning tool

The share file cleaning tool is run by using the Windows Command Prompt in Administrator mode. Do the following:

It is recommended to run the cleaning tool with a copy of the QVS.exe and the shared file in a (temporary) folder different from %ProgramData%Qliktech\Documents. The user running the cleaning tool on the %ProgramData%Qliktech\Documents folder must have administrator rights over it.

The cleaning process completely regenerates the shared file. Issues regarding fragmentation of the file will disappear and file size and access time may be reduced.

> *You can run the cleaning tool for a folder by using the option* `-subF`*. It is very important to take into account that the list of users to be removed will be common to all shared files within the folder.*

> *Backup your shared files before using the cleaning tool.*

1. Create a copy of the QVS executable. By default the QVS.exe is installed in *C:\Program Files\QlikView\Server* .
2. Navigate to the folder where the copy of the QVS.exe is located and run the cleaning tool in verify mode. For example:
   ```
   "C:\<Temporary_path>\QVS.exe –x
   “C:\ProgramData\QlikTech\Documents\FinanceAnalysis.qvw.Shared” –v
   ```
3. Locate the *CleaningTool_MACHINENAME.log* verify file log. If not specified in your command, the log is stored by default in *C:\ProgramData\QlikTech\QlikViewServer*.
   The log lists each type of corrupted shared file object if there is corruption. If the corrupt entry can be identified, it will list the object ID.
4. If there are corrupt entries, run the cleaning tool again in purge mode.
   The purge process will create a new shared file with the corrupt objects removed or corrected. The new file identified by the suffix *_clean* (for example: *MYFILENAME.QVW.TShared_clean*) is placed in the same folder as the source shared file.

   > *The new file may be larger than the source file.*

5. Replace the old corrupt shared file with the new file. This must be done when no QlikView Server services are running.

Examples

Example 1: Analyzing a shared file

Running the following command in the windows command prompt analyzes the shared file and creates a log file in the *C:\logs* folder:

```
QVS.exe -x "C:\ProgramData\QlikTech\Documents\TESTFILE.QVW.TShared" -v -l "C:\logs"
```

Example 2: Setting file ownership

Running the following command in the windows command prompt sets ownership of the server objects in the shared file to user UserX:

```
QVS.exe -x "C:\ProgramData\QlikTech\Documents\TESTFILE.QVW.TShared" -p -so UserX
```

Example 3: Replacing file ownership

Running the following command in the windows command prompt replaces ownership of the server objects in the shared file from UserX to UserY:

```
QVS.exe -x "C:\ProgramData\QlikTech\Documents\TESTFILE.QVW.Shared" -p -ro UserX UserY
```

Example 4: Changing output format

Running the following command in the windows command prompt allows to convert a file in the original shared file format to the new format:

```
QVS.exe -x "C:\Temp\1.QVW.Shared" -p -f tx
```

Example 5: Removing non-shared entries from a specific user

Running the following command in the windows command prompt removes all non-shared entries associated to a specified user UserX:

```
QVS.exe -x "C:\ProgramData\QlikTech\Documents\TESTFILE.QVW.TShared" -p -du0 UserX
```

Example 6: Removing all entries from a set of users specified in a text file

Running the following command in the windows command prompt removes all entries (including the ones that are shared) associated to a list of specified users in the Users.txt column text file.:

```
QVS.exe -x "C:\ProgramData\QlikTech\Documents\TESTFILE.QVW.Shared" -p -df1 "C:\temp\Users.txt"
```

Example of the Users.txt file:

DOMAIN\User1

DOMAIN\User2

DOMAIN\User3

...

DOMAIN\UserX

Example 7: Removing all entries from a set of users specified in a text file for a whole folder

There is also the possibility of processing a whole set of shared files contained within a folder with a common list of users to be removed from them.

Running the following command in the windows command prompt removes all (including the ones that are shared) entries associated to a list of specified users in the Users.txt column text file. For all shared files within the folder 'Documents':

```
QVS.exe -x "C:\ProgramData\QlikTech\Documents" -p -subF -df1 "C:\temp\Users.txt"
```

Example of the Users.txt file:

DOMAIN\User1

DOMAIN\User2

DOMAIN\User3

...

DOMAIN\UserX

IPv6 configurations

QlikView supports the Internet Protocol IPv6, as well as dual stack IPv6-IPv4 configurations.

IPv6 configuration for QlikView Server service (QVS)

You can customize the IPv6 settings to adapt your QlikView Server deployment to different network configurations. To customize the IPv6 configurations for QlikView Server service (QVS), open the *Settings.ini* file and add the following parameters.

Description of IPv6 configurations and default values

Name	Description	Default value
ClusterMulticastIpV6Addr	Link-Local Scope IPv6 multicast address. By default: All Nodes Address	`FF02::1`
ClusterMulticastIpV6Loop	Enable or disable loopback of outgoing multicast datagrams	`true`
ClusterMulticastIpV6Hops	Limit the lifetime of the packet. When set to `1`, multicast is available only to the local subnet.	`1`

The default location of QVS *Settings.ini* file is *%ProgramData%\QlikTech\QlikViewServer*.

All machines running QVS must have the same IPv6 settings.

Clustering QlikView services using IPv6 format

To cluster services in QlikView Management Console (QMC), you can use either a machine name or its IPv6 address. If you decide to use the machine's IPv6 address, the address must be included in square brackets. For example: `[fe80::dd3d:36bb:e284:af99]`

IPv6 configuration when using certificates

If your QlikView Server deployment uses certificates for authentication, and if you configured your deployment to use IPv6 protocol only, you must enable the `UseCertificatesIpvSix` setting for QlikView Management Service (QMS) and QlikView Server service.

For the QMS, open the QMS *exe.config* file, which by default is located in *%Program Files%\QlikView\Management Service* Add the following setting:

```
<add key=" UseCertificatesIpvSix " value="true"/>
```

For the QVS, open the *Settings.ini* file, which by default is located in *%ProgramData%\QlikTech\QlikViewServer*. Add the following setting:

```
UseCertificatesIpvSix=1
```

> *All machines running QVS must have the same `UseCertificatesIpvSix` settings.*

2.3 Logs and error codes

All alerts from the QlikView services appear in the Windows event log.

Logging from QlikView Server

Detailed session logs are found in the logging directory, which is specified on the **System**>**Setup**>**Logging** tab in QlikView Management Console (QMC). The default location is *%ProgramData%\QlikTech\QlikViewServer*.

Log files can be set to split (that is, create new) daily, weekly, monthly, yearly, or never. Performance log intervals can be set from one minute and higher.

> *Setting the interval to be very small, for example, only one minute, may negatively impact the performance.*

Session Log

A session is defined as a single user connected to a single document.

> *The session log is updated each time a session ends. This means no log entry is created when a session starts.*

The file name of the session log is *Sessions*.log*, where * reflects the server name and the split interval. Each entry of the session log contains the fields listed below.

List of entries for the session log

Entry	Description
Exe Type	Type of QVS build. Example: “RLS64” = 64-bit release build
Exe Version	Full version number of QVS. Example: “11.00.11076.0409.10”
Server Started	Date and time when QVS was started.

Entry	Description
Timestamp	Date and time when the log entry was created.
Document	QlikView document that was accessed.
Document Timestamp	File timestamp of the document that was accessed.
QlikView User	QlikView section access user ID (if used).
Exit Reason	Reason for session termination: • “Socket closed” = Client-induced termination • “LRU” = Terminated as Least Recently Used in favor of new user • “Shutdown” = Server-induced termination for other reasons *This is not a complete list, as the exit value in some cases comes from the operating system.*
Session ID	The ID of the session.
Session Start	Time when the session was started.
Session Duration	Duration of session in hours:minutes:seconds.
CPU Spent (s)	CPU seconds spent by the session.
Bytes Received	Bytes received by the server during the session.
Bytes Sent	Bytes sent by the server during the session.
Calls	Number of QlikView calls during the session (bidirectional).
Selections	Number of QlikView selections made during the session.
Authenticated User	Authenticated Windows NT® user ID (if any).
Identifying User	Client user identification.
Client Machine Identification	The client machine identification. By default, this is the universally unique identifier (UUID) receiver from the call to the Windows Management Instrumentation (WMI). If the UUID is unavailable, one of the following IDs may display instead: • MAC address of the computer • Computer name • Unique machine ID (if the browser used in the session was in a private mode)

Entry	Description
Serial Number	Serial number of the QlikView client (installed clients only, that is, QlikView Desktop and QlikView plugin).
Client Type	Client type used: • “Windows Exe” = QlikView Desktop and QlikView plugin • “Ajax” = all clients that use the QVPX protocol • “Unknown”
Client Build Version	Build version of the QlikView client.
Secure Protocol	Secure protocol used: • “On” when encrypted communication is used (typically Windows clients). • “Off” when non-encrypted communication is used.
Tunnel Protocol	“Tunnel” when QVS tunnel communication is used.
Server Port	Port used by the server.
Client Address	Client IP number for the client that is connected to the server (through the port specified in the Server Port field above).
Client Port	Client port.
CAL Type	Client Access License (CAL) type: • “User” = Named User CAL • “Session” = Session CAL • “Usage” = Usage CAL • “Document” = Document CAL
CAL Usage Count	Number of Usage CALs.

Performance Log

The performance log is updated at the interval specified on the **System**>**Setup**>**Logging** tab in QMC. The default interval is five minutes. Additional entries are added whenever the server is started or stopped. The file name of the session log is *Performance*.log*, where * reflects the server name and the split interval.

Each entry of the log contains the fields listed below.

List of entries for the performance log

Entry	Description
Exe Type	Type of QVS build. Example: “RLS64” = 64-bit release build
Exe Version	Full version number of QVS. Example: “11.00.11076.0409.10”
Server Started	Date and time when QVS was started.
Timestamp	Date and time when the log entry was created.
EntryType	Entry type: • “Server starting” = Startup • “Normal” = Normal interval log entry • “Server shutting down” = Shutdown
ActiveDocSessions	Number of document sessions* that has shown activity during the interval and still exists at the end of the interval.
DocSessions	Total number of document sessions* that exists at the end of the interval.
ActiveAnonymousDocSessions	Number of document sessions* with anonymous user that has shown activity during the interval and still exists at the end of the interval.
AnonymousDocSessions	Total number of document sessions* with anonymous user that exists at the end of the interval.
ActiveTunneledDocSessions	Number of document sessions* with tunneled connection that has shown activity during the interval and still exists at the end of the interval.
TunneledDocSessions	Total number of document sessions* with tunneled connection that exists at the end of the interval.
DocSessionStarts	Number of document sessions* that has been initiated during the interval.
ActiveDocs	Number of documents loaded at the end of the interval in which there has been user activity during the interval.
RefDocs	Number of documents loaded at the end of the interval for which there is a session at the end of the interval.
LoadedDocs	Total number of documents loaded at the end of the interval.
DocLoads	Number of new documents loaded during the interval.
DocLoadFails	Number of documents that has failed to load during the interval.

Entry	Description
Calls	Total number of calls to QVS during the interval.
Selections	Number of selection calls during the interval.
ActiveIpAddrs	Number of distinct IP addresses that has been active during the interval and still exists at the end of the interval. *Tunneled sessions and multiple users originating from the same IP cannot be distinguished.*
IpAddrs	Total number of distinct IP addresses connected at the end of the interval. *Tunneled sessions and multiple users originating from the same IP cannot be distinguished.*
ActiveUsers	Number of distinct NT users that has been active during the interval and still exists at the end of the interval. *Anonymous users cannot be distinguished.*
Users	Total number of distinct NT users connected at the end of the interval. *Anonymous users cannot be distinguished.*
CPULoad	Average CPU load from QVS during the interval.
VMAllocated(MB)	Size in MB of the virtual memory allocated by QVS at the end of the interval**.
VMCommitted(MB)	Size in MB of the virtual memory actually used by QVS at the end of the interval. This number is part of VMAllocated(MB) and should not exceed the size of the physical memory in order to avoid unacceptable response times.
VMFree(MB)	Size in MB of the unallocated virtual memory available to QVS**.
VMLargestFreeBlock(MB)	Size in MB of the largest contiguous block of unallocated virtual memory available to QVS. This number is part of VMFree(MB).
UsageCalBalance	“-1.00” = There are no Usage CALs.
CacheHits	Number of generic cache hits
CacheLookups	Number of generic cache lookups

Entry	Description
CacheObjectAdded	Number of objects added to the generic cache
CacheBytesAdded	Number of bytes added to the generic cache
CacheTimeAdded	Time spent adding new objects to the generic cache
CacheReplaced	Number of objects replaced in the generic cache

*One user + one document = One document session.

**VMAllocated(MB) + VMFree(MB) = Total maximum virtual memory space available to the QVS process.

Server-side Extension Log

The file name of the server-side extension (SSE) log is *SSE*.log*, where * reflects the server name and the split interval. Each entry of the SSE log contains the fields listed below.

List of entries for server-side extension log

Entry	Description
Severity	• 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 and must be corrected immediately. • Off: No logs, except for license logs, are produced.
Timestamp	Date and time when the log entry was created.
ProcessId	The ID of the process from which the log message originates.
ThreadId	The ID of the thread that was used when the log message was written to file.
UserId	The ID of the user.
QixRequestId	The ID established by the initiator of the request. If this member is not present, the RPC call is assumed to be a notification.
AppId	The ID of the app that includes the call to the server-side extension (SSE) plugin through an analytic connection.
App Title	The title of the app that includes the call to the SSE plugin through an analytic connection.

Entry	Description
SSEPlugin	If the log message was created during a call to the SSE plugin, the mapping/alias of that plugin, for example, SSEPython for a Python plugin. If the log message was created without a call to the SSE plugin, for example, while initializing the SSE, the value is a dash (-).
SSEPluginAddress	Two elements separated by a colon that define the analytic connection to the SSE plugin. • <Host>: DNS name (or IP-address) of the plugin. • <Port>: Port on which the plugin listens, typically 50051. For example, localhost:50051.
Message	Log message.

Event Log

The event log is updated each time a log entry is made in the Windows event log by QVS. The stored information is a mirror of the information written to the Windows event log. The file name of the event log is *Events*.log*, where * reflects the server name and the split interval.

Use the **Event Log Verbosity** radio buttons on the **System**>**Setup**>**QlikView Servers**>**Logging** tab in the QMC to set the verbosity level. Depending on the verbosity level selected, the following entries are written to the Event log:

- **Low**: Error messages
- **Medium**: Error and warning messages
- **High**: Error, warning, and information messages

Each entry of the log contains the fields listed below.

List of entries for event log

Entry	Description
`Server Started`	Date and time when QVS was started.
`Timestamp`	Date and time when the log entry was created.
`SeverityID`	ID for the severity level: • 1 = Error • 2 = Warning • 4 = Information
`EventID`	Unique ID for the event type.

Entry	Description
`Severity`	Event severity level: • Error • Information • Warning
`Message`	Event description.

End-user Audit Log

The end-user audit log contains information on user selections, including cleared selections, activated sheets, application of bookmarks, accessed reports, and maximized objects.

A log file called *AUDIT_<machinename>* is saved to *%ProgramData%\QlikTech\QlikViewServer*.

> *Tick the* **Enable Extensive Audit Logging** *check box on the* **System** *>* **Setup** *>* **QlikView Servers** *>* **Logging** *tab in the QMC to enable detailed audit logging (for example, logging of all selections that come with a bookmark). However, the logging of user selections in QVS is based on how the current selections object works and therefore larger selections may not be logged in detail.*

List of entries for end-user audit log

Entry	Description
`Session ID`	Session ID
`Server started`	Date and time when QVS was started.
`Timestamp`	Date and time when the log entry was created.
`Document`	Path and name of the document that was accessed.
`Type`	Type of selection made (for example, “Selection” or “Bookmark”). For an overview of the types available, see the table below.
`User`	User name.
`Message`	Information on the type of selection or application of bookmark that was made in the document (for example, “Apply Server\Bookmark15”). For an overview of the messages that can be posted in this field, see the table below.
`Id`	ID of the object that is connected to the operation (for example, "Document\SH03"). If there is no object connected to the operation, this field is empty.
`Session`	Session number

Types and Messages in the end-user audit log

The types and messages that can be posted in the Type and Message fields in the end-user audit log are listed below.

In the end-user audit log, "XXX" and "YYY" are replaced with values from the QlikView document.

Types and messages found in the end-user audit log

Type	Message	Description
Action	action (#) [XXX]	Action # was executed with XXX. The numeric value corresponds to one of the following actions: • Info = 0 • Lock All = 2 • Unlock All = 3 • Clear All = 4 • Clear All Including Locked = 5 • Back = 6 • Forward = 7 • File Close = 8 • Next Tab = 9 • Previous Tab = 10 • Export = 11 • Launch = 12 • Macro = 13 • Recall Bookmark = 14 • Replace Bookmark = 15 • Create Bookmark = 16 • Print Report = 17 • Activate Sheet = 18 • Print Sheet = 19 • Print Object = 20 • Restore Object = 21 • Minimize Object = 22 • Maximize Object = 23 • Activate Object = 24 • Select Excluded = 25 • Clear Other Fields = 26 • Select Possible = 27 • Lock = 28 • Unlock = 29 • Pareto Select = 30

Type	Message	Description
Action	action (#) [XXX]	• Set Value = 31 • Field Select = 32 • Field Toggle Select = 33 • Open URL = 34 • Document Chain = 35 • Clear Field = 36 • Reload = 37 • Set state = 38 • Transfer state = 39 • Swap state = 40 • Dynamic update = 41
Bookmark	Apply XXX	Bookmark XXX was applied.
Bookmark Selection	XXX	Selection XXX was made because a bookmark was selected. Entries of this type are only logged when detailed audit logging is selected.
Document	Document XXX	Document XXX was opened or closed.
Export	Sheet Object XXX	Sheet object XXX was exported.
Maximize	Sheet Object XXX	Sheet object XXX was maximized.
Minimize	Sheet Object XXX	Sheet object XXX was minimized.
Print	Sheet Object XXX	Sheet object XXX was printed.
Report	Accessed report XXX	Report XXX was accessed.
Restore	Sheet Object XXX	Sheet object XXX was restored.
Selection	Clear All	All selections were cleared.
Selection	XXX	Selection XXX was made.
SendToExcel	Sheet Object XXX	Sheet object XXX was sent to Microsoft Excel.
Sheet Object	Sheet Object XXX	Various activities that can apply to Sheet object XXX.
Session Collaboration	Session Collaboration Initiated, ID:XXX	A session collaboration with ID XXX was initiated.
Session Collaboration	Session Collaboration user XXX joined session, ID:YYY	User XXX joined the session collaboration with ID YYY.
Session Collaboration	Session Collaboration user XXX left session, ID:YYY	User XXX left the session collaboration with ID YYY.

The following example shows the resulting log entry when a bookmark (“Bookmark01”) is selected. The log has been put in a table for better overview.

Example of the end-user audit log when (“Bookmark01”) is selected

Entry	Value
`Session ID`	b5134c4f-7f3d-4107-a37b-d842e9452d93
`Server started`	20130506T101733.000+0900
`Timestamp`	20130506T102328.000+0900
`Document`	C:\ProgramData\QlikTech\Documents\Test.qvw
`Type`	Bookmark
`User`	QlikTech\jsmith
`Message`	Apply Server\Bookmark01
`Id`	Document\SH03
`Session`	3667

If detailed audit logging is selected, the log entry above may be followed by one or more log entries that detail the selections that were made because the bookmark was selected. In these log entries, the Type field is set to “Bookmark Selection”.

Manager Audit Log

The audit logging provides the possibility to track changes to tasks and settings in the system in order to see who made the changes and when they were made.

The audit logs are stored in *%ProgramData%\QlikTech\ManagementService\AuditLog*. One folder per table is created. The number of folders created varies depending on the settings of your installation. Each folder contains one file per day with the changes made to the tasks. The logs are tab separated files.

The tab below lists the entries common to all audit log files. Each audit log files contains further entries specific for each type of log file.

List of entries common to all audit log files

Entry	Description
`TransactionID`	Transaction ID, which is useful for keeping track of changes made simultaneously.
`ChangeType`	Type of operation, `Update` (new or changed entries) or `Delete` (entries have been deleted).
`ModifiedTime`	Time and date (in UTC) when the changes were made.
`ModifiedByUser`	The user that made the changes in the user interface. `System` means that the change was initiated by the system and not by any user.
`ID`	ID of the row (that was updated or deleted) in the table that was changed.

The following example comes from the `AlertEmail` table. The log has been put in a table for better overview. Not all entries are listed in this example.

Manager Audit log example from `AlertEmail`

Entry	Value
`TransactionID`	455a241d-8428-4dc7-ba67-4ae7cb21cf3d
`ChangeType`	Update
`ModifiedTime`	20100202T151254.000+0900
`ModifiedByUser`	MyDomain\mjn
`ID`	b3745325-cee7-4fe7-b681-9c9efe22fc5c
`DistributionServiceID`	8846d7dd-bb3f-4289-9c9b-b0ca71b7c3b2
`EmailAddress`	mjn

The following example comes from the `QDSCluster` table. Note that `TransactionID` is the same for both examples. This means that the changes were made simultaneously. Not all entries are listed in this example.

Manager Audit log example from `QDSCluster`

Entry	Value
`TransactionID`	455a241d-8428-4dc7-ba67-4ae7cb21cf3d
`ChangeType`	Update
`ModifiedTime`	20100202T151254.000+0900
`ModifiedByUser`	MyDomain\mjn
`ID`	a37f242c-6d80-42da-a10c-1742d2ec927f
`DistributionServiceID`	8846d7dd-bb3f-4289-9c9b-b0ca71b7c3b2
`QDSWebAdress`	http://computer-mjn:4720/qtxs.asmx
`CurrentWorkorderID`	96bff2dc-f1ea-84d2-b6c4-ea58bf5c98e5

Task Performance Summary

The task performance summary is used to log task performance information.

Proceed as follows to activate the task performance summary:

1. Open the *Settings.ini* file in a text editor. The default location of the file is: *C:\Windows\system32\config\systemprofile\AppData\Roaming\QlikTech\QlikViewBatch*
2. Locate the following section in the *Settings.ini* file:

```
[Settings 7]

InterfaceLanguage=English
```

```
InstalledLIBID110={4D121C39-415E-11D1-934D-0040333C91CC}
```

3. Add `EnableQVBProcessSummary=1` at the end of the section to activate the task performance summary.

   The last row in the Settings.ini file must be empty.

4. Save the *Settings.ini* file.
5. Restart theQlikView Distribution Service (QDS).

   Once the QDS has restarted, the task log is updated.

Example of task performance summary output

Entry	Value
`Name`	qvb.exe
`PID`	1360
`Peak CPU`	50,0%
`Peak Physical RAM`	26.00 Mb
`Peak Virtual RAM`	21.69 Mb
`Average CPU`	CPU: 1,0%
`Average Physical RAM`	24.47 Mb
`Average Virtual RAM`	20.37 Mb
`Peak Total CPU`	58,3%
`Peak Total Physical RAM`	6143.49 Mb
`Peak Total Virtual RAM`	12285.17 Mb
`Elapsed Time`	00:00:36.4692722

Reload performance log

You can enable the creation of a dedicated reload performance log *.xml* file for each task. This log file gathers the task reload performance metadada and process summary.

Do the following:

1. Open the QVB *Settings.ini* file for which by default is located in *%System32%\config\systemprofile\AppData\Roaming\QlikTech\QlikViewBatch*.
2. Add the following line below `[Settings 7]`:

   ```
   EnableQVBReloadMetadata=1
   ```

3. Save the *Settings.ini* file.

A task reload performance log *.xml* file is created at each execution of a reload, and saved by default in *%ProgramData%\QlikTech\DistributionService\TaskResults*. The name format of the *.xml* file is:

ReloadMetaData_machine-name_20180904T104446_Document-name.xml

Where:

- <machine-name> is the execution machine name
- <20180904T104446> is the date and time of execution
- <Document-name> is the name of the document reloaded

The task reload performance log file has the following `reload_meta` performance fields:

List of `reload_meta` and `static_byte_size` entries in the reload performance log

Entry	Description
`cpu_time_spent_in_ms`	Time spent by the CPU to perform the reload, displayed in milliseconds.
`logical_cores`	Number of cores in the CPU.
`total_memory`	Total physical RAM available on the machine.
`static_byte_size`	Static memory usage for the document.

The reload performance log file has the following `ProcessSummary` performance fields . These entries are similar to those listed in the Task Performance Summary log file. See: Task Performance Summary.

List of `ProcessSummary` entries in the reload performance log

Entry	Description
`App`	Name of the document reloaded.
`Date`	Date, displayed as: 20180904T104446.
`CurrentProcessCpu`	Current CPU usage in percentage. For example: 99.991681.
`PeakPhysMemUsedByProc`	Peak of physical RAM used by the processor, displayed in byte.
`PeakVirtualMemUsedByProc`	Peak of virtual RAM used by the processor, displayed in byte.
`AvgCurrentProcessCpu`	Average CPU usage in percentage. For example: 80.81025.
`AvgPhysMemUsedByProc`	Average physical RAM used by the processor, displayed in byte.
`AvgVirtualMemUsedByProc`	Average virtual RAM used by the processor, displayed in byte.
`TotalCpu`	Total CPU available on the machine.
`TotalPhysMem`	Total physical RAM available on the machine.
`TotalVirtualMem`	Total virtual RAM available on the machine.

The reload performance log can also list `FieldMetadata` and `TableMetadata` entries, which describe the fields and tables visible in the Table Viewer in QlikView.

QIX performance log

The QIX performance log provides detailed information on the QIX Engine performance. By default, the QIX Performance log file is disabled.

Enabling the QIX performance log

To enable the QIX performance log, you must add the line `QixPerformanceLogVerbosity` to the QlikView Server Service (QVS) *Settings.ini* file, followed by the desired level of verbosity. For example:

```
QixPerformanceLogVerbosity=3
```

The levels of verbosity for the QIX performance log are as follows:

- 0 = Off
- 1 = Fatal
- 2 = Error
- 3 = Warning
- 4 = Info
- 5 = Debug

If you enable the QIX performance log, and set `QixPerformanceLogVerbosity` to level 3 or 2, the following four levels must also be added to the *Settings.ini* file:

```
WarningProcessTimeMs
ErrorProcessTimeMs
WarningPeakMemory
ErrorPeakMemory
```

These four levels are necessary to determine when to trigger a warning or error event.

Do the following:

1. Open the QlikView Server Service (QVS) *Settings.ini* file, which by default is located in: *%ProgramData%\QlikTech\QlikViewServer*.
2. Add the following line:
   ```
   QixPerformanceLogVerbosity=3
   ```
3. Save the *Settings.ini* file.
4. Restart the QlikView Server Service (QVS).

> *When enabled, the QIX performance log file produces a considerably large amount of data. It is therefore recommended to enable this log file only for limited periods of time.*

The following table lists the entries included in the QIX performance log.

List of entries in the QIX performance log

Entry	Description
`Timestamp`	Time when the engine wrote the log message to file.
`ProcessId`	ID of the engine process from which the log message originates.
`ThreadId`	ID of the thread that was used when the engine wrote the log message to file.
`SessionId`	ID of the engine session for which the QIX method call was made.

Entry	Description
`CServerId`	ID of the server instance that handled the request.
`Server Started`	Time when the engine started.
`Method`	Name of the QIX method that was called.
`RequestId`	ID of the request in which the QIX method call was handled.
`Target`	Memory address of the target for the QIX method call.
`RequestException`	ID of an exception (if any) that occurred as a result of the QIX method call.
`AnyException`	Returned error code
`ProcessTime`	Amount of time that was needed to process the request.
`WorkTime`	Amount of time that the request did actual work.
`LockTime`	Amount of time that the request had to wait for an internal lock.
`ValidateTime`	Amount of time that the request used for validation.
`TraverseTime`	Time in milliseconds spent by the thread or fiber for traversing within the Hypercube
`Handle`	ID of the interface that handled the request. The interface can be Global, a certain sheet, a certain object, or similar.
`DocId`	Path and name of the QlikView document.
`ObjectId`	ID of the object included in the QlikView document.
`NetRAM`	Current RAM allocation count in bytes.
`PeakRAM`	Peak RAM allocation count in bytes.
`ObjectType`	Type of object included in the QlikView document.

3 QlikView Installation

This section gives information on how to install QlikView. It also describes some maintenance tasks, such as how to update, repair or modify the installation.

3.1 Installing QlikView Server

This documentation outlines the steps you need to follow to install and license QlikView Server. For a description of how to install QlikView Desktop, see: Installing QlikView Desktop.

Before Installing QlikView Server

Before installing QlikView Server, you need to consider:

- If Microsoft IIS is to be used as web server, it must be installed prior to QlikView Server.
- It is not possible to install QlikView Server to a server that acts as a domain controller.
- Internet protocol IPv4 or IPv6 is required for installation of QlikView Server.
- When installing QlikView Server/Publisher, several security groups are created. Several other security groups must be created following the installation. These must be properly configured to ensure that the appropriate services can run, and to ensure that users can access the appropriate functionality. Before you begin the installation, see **Security Groups** in *QlikView Publisher Repository (page 29)*.
- It is recommended not to move folder locations after the QlikView Server installation is complete, since many settings depend on the initial file locations. If the location of QlikView Server has to be changed after the installation, uninstall QlikView Server and then reinstall.
- Any previously defined tasks are deleted when the QlikView Publisher license is activated.

Setup Procedure

1. Download the QlikView Server installation executable from ≤ Product Downloads.
   - Microsoft Windows x64 version: *QlikViewServer_x64Setup.exe*

   For more information, see *Downloading installation files (page 111)*.
2. Run the QlikView Server installation executable.
3. If the User Account Control dialog is displayed, click **Yes** to allow the program to make changes on this computer.
4. Click **Next** in the Welcome dialog.
5. Select the region for the location of the server. Click **Next** to continue.
6. Read the license agreement, select **I accept the terms in the license agreement**, and click **Next** to continue.
7. Enter the customer information for QlikView Server. Click **Next** to continue.
8. All files are installed in the specified folder. To change the root folder for the installed files, click **Change** to specify the preferred location. Finally, click **Next** to continue.
9. Select the type of installation you want to perform:

- **Full installation, Single machine with QlikView Webserver:** Used to run all components on a single machine with QlikView Web Server as web server.
- **Full installation, Single machine with Microsoft IIS:** Used to run all components on a single machine with Microsoft IIS as web server. This option is only available if IIS is installed on the target machine.
- **Custom installation, select profiles:** If this option is selected you select the profiles you want to be included in the installation from the Profiles section in the dialog:
  - **QlikView Server:** Installs QlikView Server, Directory Service Connector, and the QlikView Server example documents.
  - **Reload/ Distribute Engine:** Installs the Reload Engine and the QlikView Distribution Service.
  - **Management Console:** Installs the QlikView Management Service together with the QlikView Management Console (QMC).
  - **Webserver:** Installs the QlikView Web Server.

To make further configuration of features to be installed, click **Config**. When done, click **Next**.

To use pre-defined configuration of features, click **Next**.

10. Set the account that the QlikView Server and Publisher services are to run under. Click **Next** to continue.

    The account that is used to run the QlikView services must have local administrator privileges.

    You can also select **I want to specify the account to be used for the services later**.
11. Select the IIS Website from the drop-down list and click **Next**.

    *This step is only applicable if **Full installation, Single machine with Microsoft IIS** was selected in **Step 8**. If not, proceed directly to the next step.*
12. Select the Service Authentication method:
    - **Use digital certificates:** Authenticate communication between QlikView servers using digital certificates and SSL/TSL. This alternative is recommended in environments where not all servers have access to a common Windows Active Directory or when the security provided by certificate authentication is required. Note that digital certificates are **only** supported by Windows Server 2008 R2 and later.
    - **Use QlikView Administrators Group:** Authenticate communication between QlikView services based on membership in the local Windows group QlikViewAdministrators. This alternative can be used in environments where all servers that are part of the QlikView installation can authenticate using a common Windows Active Directory.

    Click **Next** to continue.
13. Click **Install** to start the installation.

> *This may take several minutes to complete.*

14. Click **Finish** when the installation is complete.
15. Log off from Windows® and then log on again, so that group memberships added during the installation are updated.

> *It may be sufficient to log off from Windows and then log on again. However, it is recommended to restart the machine to enable the QlikView Server functionality.*

Logging the Installation

The setup procedure is logged when running the QlikView Server installation executable. The log files are as follows:

- Microsoft Windows x64 version: *QlikViewServerx64.wil*

The log files are stored in the *Temp* folder of the user (for example, *%UserProfile%\AppData\Local\Temp*). Each time an installation is executed, a new file is generated, over-writing the previous log file.

Obtaining the MSI package

If the MSI package is needed for the installation, proceed as follows to extract it from the *.exe* file:

1. Start the installation from the *.exe* file and wait until the first dialog opens.
2. Locate the MSI file (often stored with a random name, for example, *ed34g.msi*) in the *Temp* folder in *%UserProfile%\AppData\Local*.
3. Copy the *.msi* file to another location.
4. Exit the *.exe* installation.
5. Install QlikView Server using the *.msi* file.

Completing the Installation

After successfully installing QlikView Server, a license must be registered in QlikView Management Console (QMC) to activate the installed software.

> *If access is denied when starting QMC, log off from Windows and then log on again, so that group memberships added during the installation are updated.*

> *Running real-time anti-virus protection on the server degrades the performance of QlikView Server. It is recommended that the user documents, source documents, log directories, and .pgo files are excluded from the anti-virus scanning.*

Running Microsoft IIS

Handling Timeouts

> *This is only needed when using very large QlikView documents that return timeouts.*

Proceed as follows to handle timeouts:

1. Open the *%ProgramFiles%\QlikView\Server\QlikViewClients\QlikViewAjax\web.config* file in a text editor (for example, Notepad).
2. Search for the following text:
   ```
   <httpRuntime requestValidationMode="2.0" />
   ```
3. Edit the text so that it becomes:
   ```
   <httpRuntime requestValidationMode="2.0" executionTimeout="900"/>
   ```
4. Save the file.

Enabling ASP.NET

If Microsoft IIS is used as web server, enable ASP.NET to ensure proper operation of the QlikView Server sample pages and the extended functions (for example, QlikView Server tunnel).

Optimizing the Performance

To optimize the performance when running Microsoft IIS and AJAX, turn on compression in the web server.

For information on how to configure IIS http compression:
≤ HTTP Compression

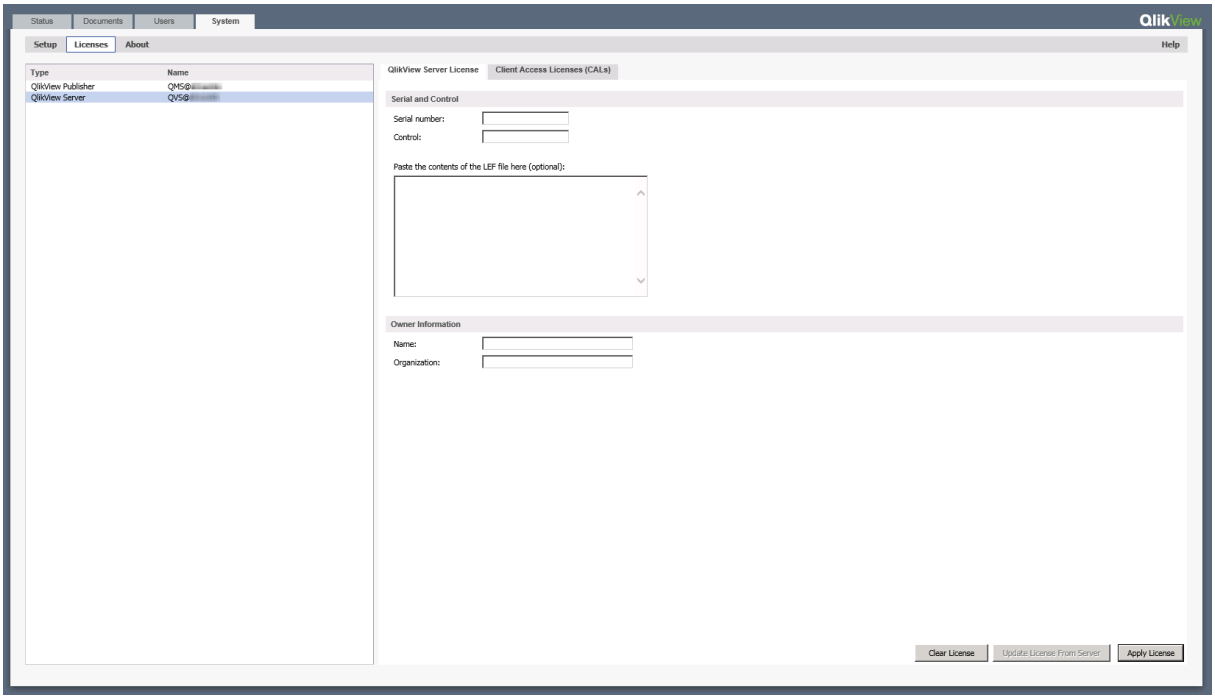
Licensing

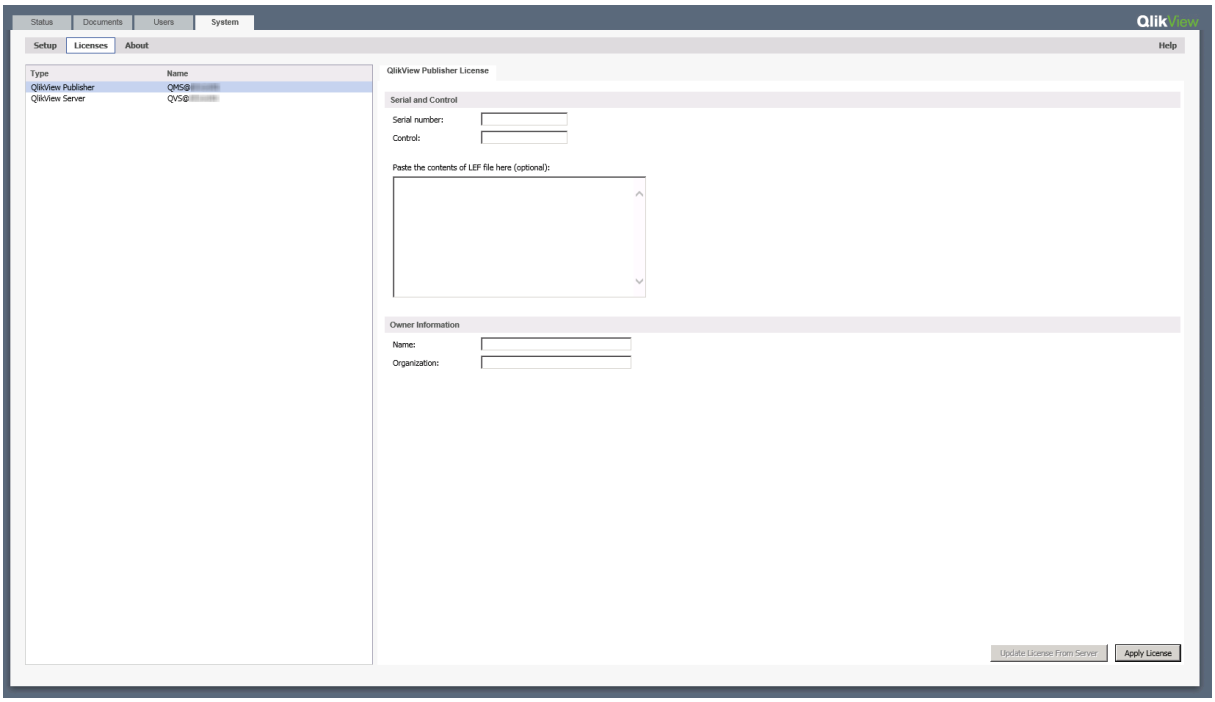
The licensing is used to authenticate QlikView Server and allow it to run on a specific machine.

Proceed as follows to enter the license for QlikView Server:

1. Go to **System**>**Licenses** in the QMC.
2. Select a QlikView Server or Publisher.
3. Fill in the **Serial number** and **Control** fields on the **QlikView Server License** or **QlikView Publisher License** tab (depending on whether QlikView Server or Publisher was chosen).

   > *Any previously defined tasks are deleted when the QlikView Publisher license is activated.*





QlikView Server/Publisher License tab in QMC

The license is checked every time a document is opened. If the time limit specified by the License Enabler File (LEF) is reached, the QVS automatically enters offline mode, which means that it is reachable from the QMC, but not operational.

The License Enabler File (LEF), *lef.txt*, for QlikView Server is automatically saved in *%ProgramData%\QlikTech*.

The *PubLef.txt* file for QlikView Publisher is saved in *%ProgramData%\QlikTech\ManagementService\Publisher LEF*.

Click **Update License from Server** to download a new *lef.txt* file from the QlikView LEF server. This is primarily used when updating the number of Client Access Licenses (CALs).

If the LEF information cannot be accessed through the Internet, it can be obtained from the local vendor. In that case, copy the entire *lef.txt* file to the location mentioned above, or paste the LEF data using the corresponding field on the QlikView Server/Publisher License tab in QMC. Contact the local vendor for specific instructions.

3.2 Downloading installation files

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

Do the following:

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

3.3 Configuring servers with digital certificates

When you choose digital certificates as your Service Authentication method, the certificates create trust between the services running on QlikView server machines. The certificates are installed when you create a new instance of QlikView.

On a stand alone deployment, all services run on the same machine. If you install QlikView nodes in a multi-server environment, you should only install the services you want to enable on each server. If you perform a complete install each time you install a node, then you will create multiple instances of the QlikView Management Service (QMS). If you have more than one QMS service running this will cause a mismatch in your certificates, as the QMS is responsible for distributing certificates to the other nodes in the deployment. When you run the installer, always select custom install and only install the services that you need to enable.

It is also recommended that you use the same Windows Administrator on all servers in the QlikView configuration.

Configuring security

To make your QlikView deployment as secure as possible, ensure that you configure secure socket layer (SSL) security on all your QlikView servers.

Enabling SSL on QlikView servers

To enable certificate service authentication between servers using SSL for Directory Service Connector (DSC), QlikView Web Server (QVWS), QlikView Management Service (QMS), QlikView Distribution Service (QDS), and QlikView Server (QVS):

1. Stop the QlikView Management Service, which runs the QlikView Management Console.
2. Run Notepad as an administrator.
3. Open the QMS configuration file in Notepad.
4. Change the key `UseWinAuthentication` value from `true` to `false`.
5. Save your changes.
6. Start the QMS service.

To verify that certificates are correctly set on the server that executes the QMS service run the Microsoft Management Console (MMC) from the Start menu.

Now repeat the steps above for the DSC, QDS, QVWS and IIS services in your system.

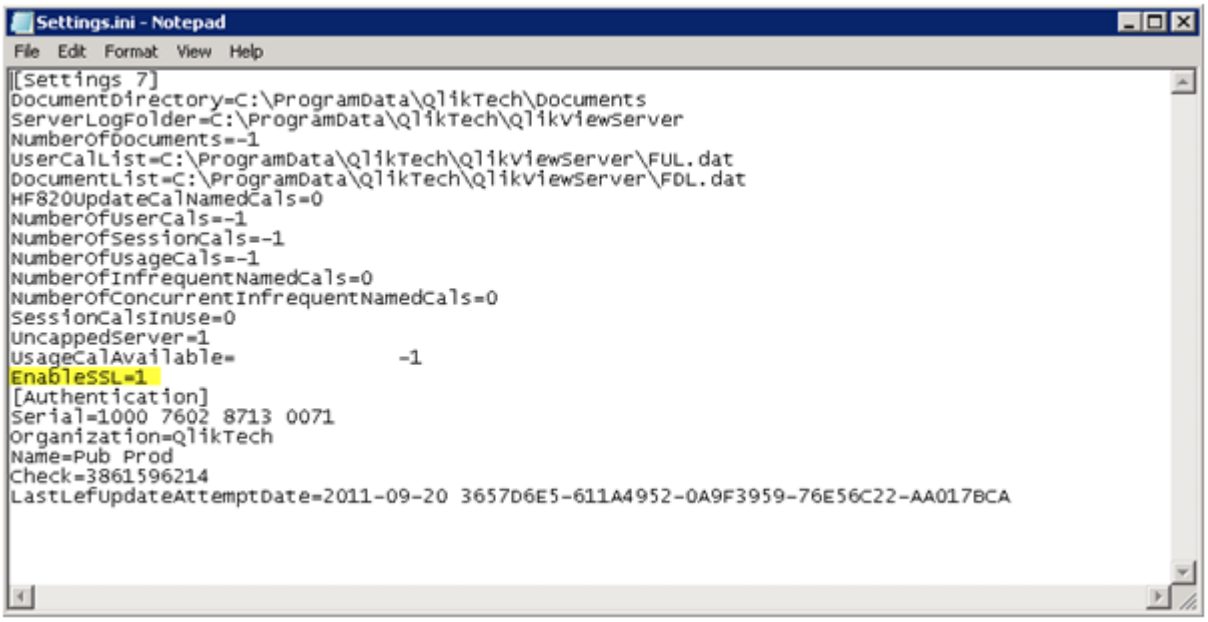
To configure certificate trust with IIS and QlikView Server use port 4750 (the same port as QVWS). The certificate trust used to enable HTTPS access for users of the web server is also used.

Enabling SSL for QlikView Server (QVS)

Complete the following additional step to configure SSL on the QlikView server service (QVS).

To edit the QVS service *Settings.ini* file:

1. Stop the QVS service.
2. Run Notepad as an administrator.
3. In Notepad, open the *Settings.ini* file.
4. Add `EnableSSL=1` in the `[Settings 7]` section.



5. Save your changes.
6. Start the QVS service.

Enable SSL for the QlikView Service Dispatcher (License service)

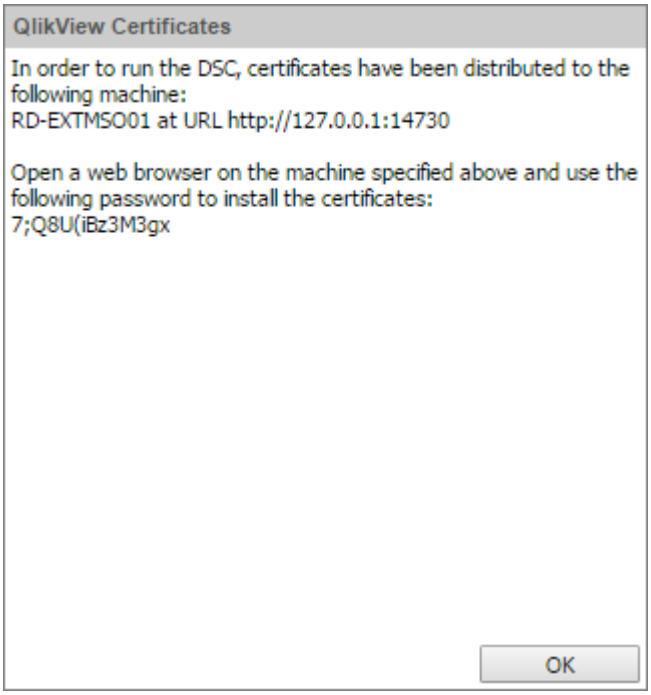
You need to enable secure socket layer (SSL) security for the license service. The License service is managed via the Service Dispatcher.

1. Stop the Service Dispatcher.
2. Run Notepad as an administrator.
3. In Notepad, open the *Services.conf* file (*C:\Program Files\QlikView\ServiceDispatcher\services.conf*)
4. Under the *[licenses.parameters]* section, change
   `-qv-auth-mode=ntlm`
   to
   `-qv-auth-mode=cert`
5. Save your changes.
6. Start the Service Dispatcher.

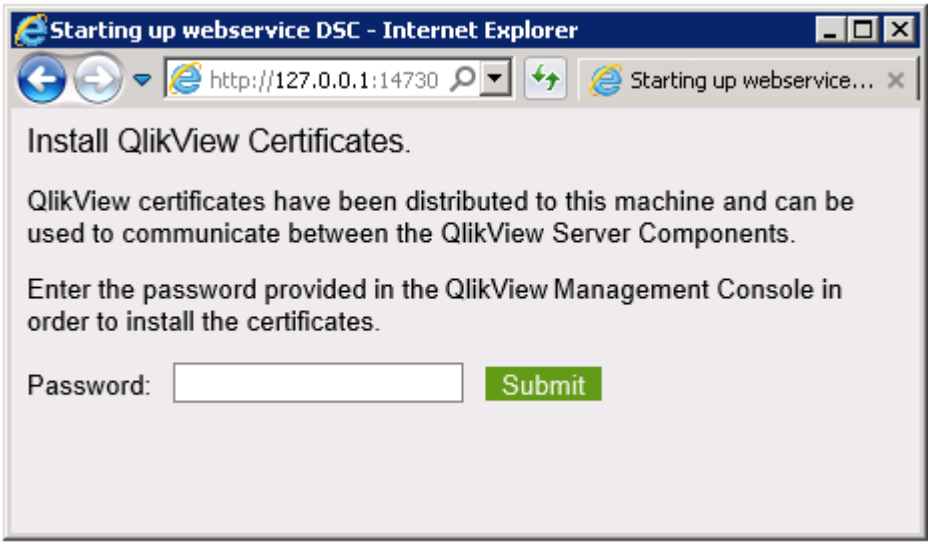
Adding QlikView services

To add QlikView services:

1. Open the QlikView Management Console.
2. Click the **System** tab, and then click **Setup** to see all the QlikView services.
3. To add a new service, click on the **Add** icon, to the right in the QlikView Servers pane.
4. Enter the **Service URL** in the text box, and click **Apply.** The new entry will be available in the tree view in the left pane. Add each service as a new service and then delete the existing service.
5. When you add a service, the **QlikView Certificates** window is displayed.



6. On the server where you are adding the new service, open a web browser and enter the URL and port provided by the QlikView Management Console **QlikView Certificates** window (14720, 14730 or 14750).
7. Enter the password provided by the QlikView Management Console **QlikView Certificates** window.



8. If successful, a message is displayed that confirms the password is correct and that the QlikView service can gain access via that port.

At this point, you can check to see if the certificates are properly installed on the servers that execute the additional QlikView services by running the MMC from the Start menu.

Updating certificates

You need to generate a new set of certificates when your certificates have expired, are about to expire, or if you want to generate new encryption keys for your sensitive data. Remember to make a copy of your old certificates.

Certificates expire after 10 years, but you can update them at any time. The expiration date of the certificates is displayed in the QMC. When 30 days or less remain before the expiration date, the QlikView Management Console displays a warning.

> *Certificates should not be replaced, but updated. Removing existing certificates may result in undecryptable data.*

Besides expiration of certificates, there may be other reasons to update, for example replacing a computer or changing a computer name, since one of the certificates is linked to the computer name.

To update certificates, follow these steps on each machine in the cluster:

1. Shut down all QlikView services (in any order).
2. If the machine currently has valid certificates that should be replaced, enable the configuration flag **InstallingNewCertificatesAndCryptoKey** for all QlikView services except the Qlik License Service.
3. Start up all QlikView services (in any order).
4. Click the **System** tab, and then click **Setup**.
5. Select a service, and then click the **General** tab for the service.
6. Click the **Apply** button in the bottom right of the window, and then follow the instructions to install the certificate.
7. Repeat the above procedure for each service (in any order) that requires updated certificates.
8. Shut down all QlikView services (in any order).
9. If you enabled the configuration flag **InstallingNewCertificatesAndCryptoKey** in a previous step, now disable it for all services.
10. Start up all services. Start the QlikView Management Service (QMS) first.

At start-up, having new certificates (containing a new encryption key), the services will re-encrypt all their sensitive data with the new encryption key.

> *It is strongly recommended not to delete the old certificates (although they now are virtually obsolete). If you later need to restore an older backup of your data you will need the previous certificates (with the corresponding encryption key) to decrypt it.*

> *When you update the certificates for your installation, you must restart the QlikView Management Service (QMS) before the Qlik License Service. Starting the services in this order ensures that the correct set of certificates is exported and made available to the Qlik License Service. You can manage the status of the Qlik License Service by starting and stopping the Qlik Service Dispatcher.*

Setting **InstallingNewCertificatesAndCryptoKey** flag

If you enable this flag, by setting it to true, any existing certificates installed on the server machine are ignored (except for extracting the CryptoAlgorithm). The flag is used by the DSC, QDS, and QVWS, but not by the QMS, and is disabled (set to false) by default.

Enable this flag when updating certificates, so you can receive new certificates. After the certificates have been updated, set the flag to false for all services.

To enable the flag, add the following line:

`InstallingNewCertificatesAndCryptoKey=True`
to the following configuration files:

- *C:\Program Files\QlikView\Distribution Service\QVDistributionService.exe.config*
- *C:\Program Files\QlikView\Directory Service Connector\QVDirectoryServiceConnector.exe.config*
- *C:\Program Files\QlikView\Server\Web Server\QVWebServer.exe.config*

Service failure due to undecryptable data

At start-up, each service validates all its encrypted data entries to ensure they are accessible. If the service encounters data that cannot be decrypted, it reports an error and stops execution.

There are two reasons why a service cannot decrypt data:

1. The certificate is missing - The certificate containing the required encryption key is missing. To solve this problem, re-install the certificate from a backup, then re-start the service.
2. The encrypted data is undecryptable - To solve this problem, erase the undecryptable data.

How to erase corrupted data

To erase undecryptable data by temporarily enabling the hidden configuration `EraseUndecryptableData` flag:

1. Stop the service.
2. Run Notepad as an administrator.
3. Open the configuration file in Notepad.
4. Add the `EraseUndecryptableData` entry and set it to `true`.
5. Save the file.
6. Restart the service.

   When the service starts, only the undecryptable part of the data is erased.

7. Stop the service, open the configuration file and remove the `EraseUndecryptableData` entry.
8. Save the file and restart the service.
   The service starts normally.

In the QMC, re-enter the erased data. All the undecryptable data entries have already been listed in the service log file, and this indicates what you need to re-enter in the QMC.

3.4 Silent Installation

When running a silent installation, QlikView is installed with a limited set of or no dialogs at all. This means all features, properties, and user selections have to be known when creating the silent installation package. There are also some standard properties in Windows Installer Service that may be required.

To prepare a silent installation, the MSI file has to be extracted from the QlikView *Setup.exe* file.

A silent installation can be run with different interface levels:

Interface levels

Command	Type
/qn	Completely silent.
/qb	Basic user interface.

Add a *+* sign at end of the interface levels command to get a modal dialog at the end of the installation saying “Finished” and if it was successful or not.

The following silent installation command lines are recommended for QlikView:

msiexec /i QlikViewServerx64.msi Addlocal="all" IS_NET_API_LOGON_USERNAME="Domain\username" IS_NET_API_LOGON_PASSWORD="password" /qn+

Alternatively:

QlikViewServer_x64Setup.exe /s /v"/qn+ Addlocal="all" IS_NET_API_LOGON_USERNAME="Domain\username" IS_NET_API_LOGON_PASSWORD="password""

The command line above installs all features completely silently with a modal dialog at the end of the installation.

If just a limited set of the features are to be installed, change *all* to the name of the feature instead. If several features are to be installed, separate them with commas.

The following features can be installed:

- DirectoryServiceConnector
- ManagementService
- QVS
- QvsDocs
- WebServer

- DistributionService
- SupportTools
- QvsClients with the sub-features Plugin and AjaxZfc
- MsIIS with the sub-features QvTunnel and QlikView Settings Service

> *For the sub-features to be included in the installation, they have to be included in the list of features to be installed.*

msiexec /i QlikViewServerx64.msi ADDLOCAL="all" DEFAULTWEBSITE="2" /qn+

This command line installs all features, including the virtual directories to another website than the default one. This requires a machine with Microsoft Internet Information Services (IIS) installed and more than one website on it. The site number also has to be known. Set *DEFAULTWEBSITE* to the site number where the virtual directories are to be installed. To find the number of the website, check IIS.

The installation procedure can be logged, using the following command:

*msiexec /i QlikViewServerx64.msi ADDLOCAL="all" DEFAULTWEBSITE="2"/L*v log.txt /qn+*

Settings

The following settings are good to know when designing a silent installation package:

Silent installation settings

Setting	Description
Prerequisites	.NET Framework 4.8 or higher *For .NET Framework 4.8 to work, Windows 10 must be updated to the Anniversary update Build 1607 or later.*
Default installation folder (INSTALLDIR)	ProgramFilesFolder\QlikView
Windows Installer Version	3.1 Schema 301
Default language	English (United States) 1033
Require Administrative Privileges	Yes
INSTALLEVEL	100, all features is set to 101 by default
Features	There is a hidden feature called “Install”. Do not remove it.
IIS	Four virtual directories and an Application pool are installed
Services	Five services are installed

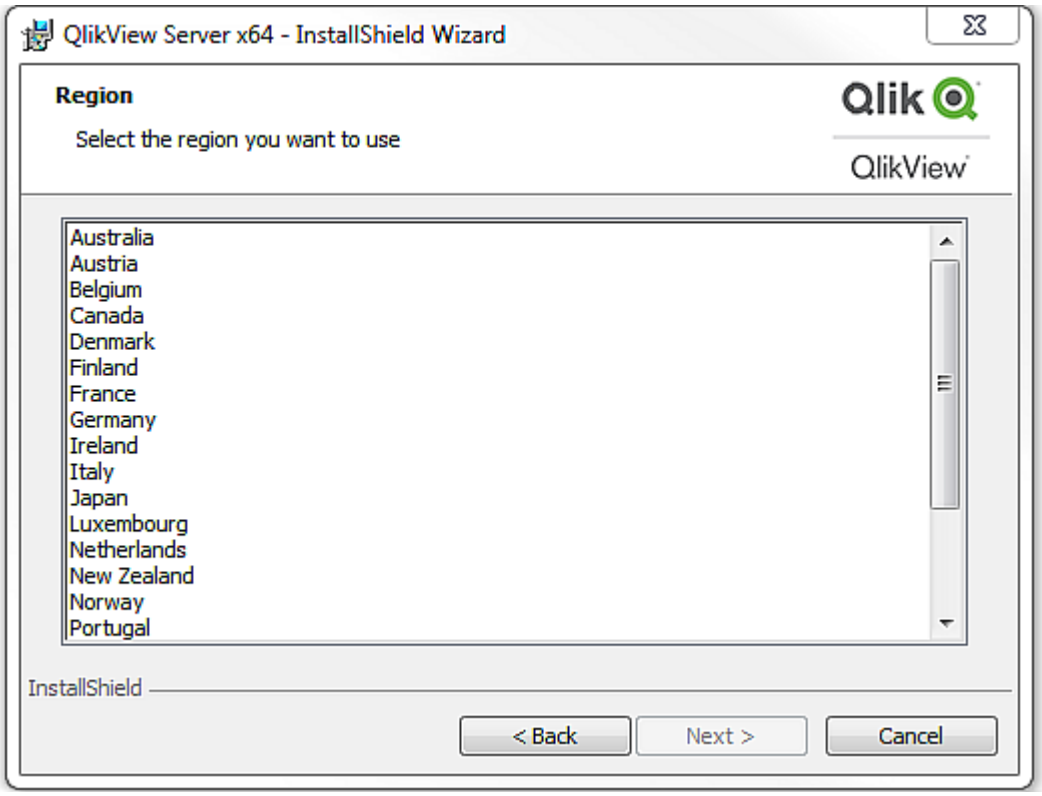
Dialogs

The QlikView installation has a number of dialogs, one of which is a Custom Setup dialog and one of which is a Website dialog. All dialogs set important properties. To find the value of a property, do a test installation with verbose logging. Note that the property values may differ depending on the language and operating system used.

Region

This dialog is used for specifying the region.

Property: *REGION_LIST*

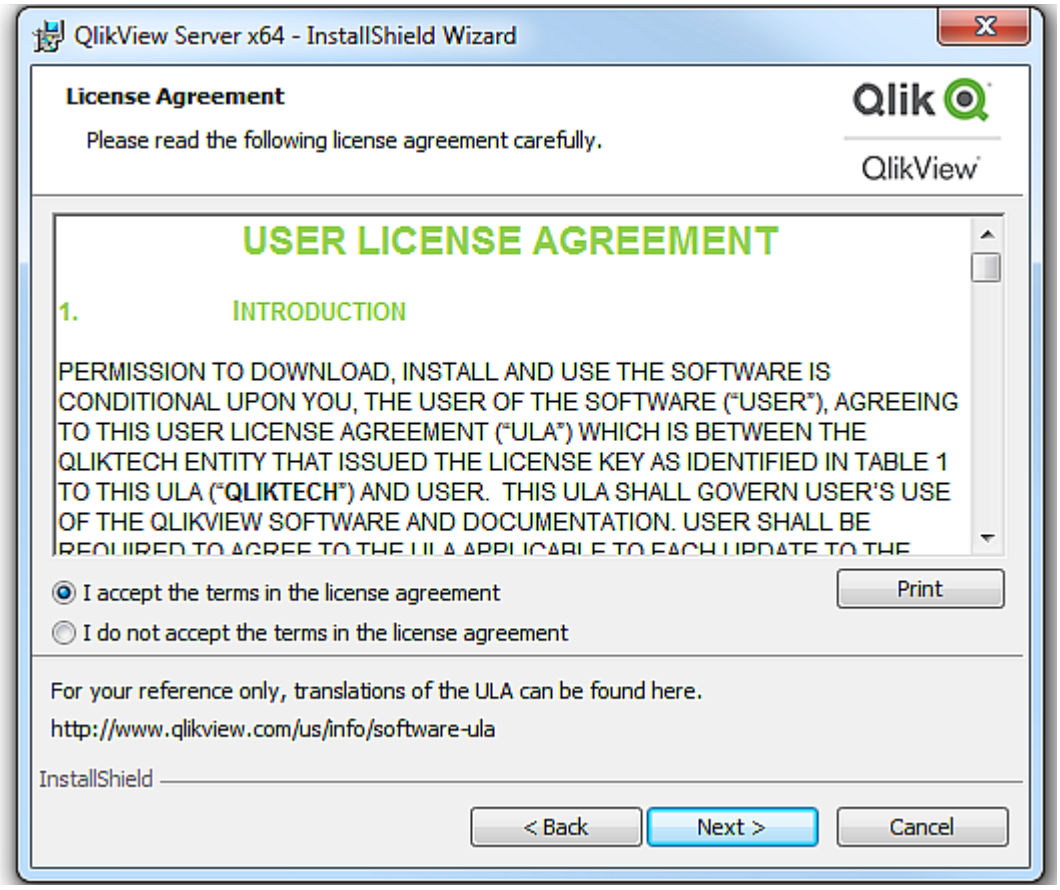


Region dialog

License Agreement

This dialog displays the license agreement for the selected region.

Radio button: *AgreeToLicense = "Yes"*



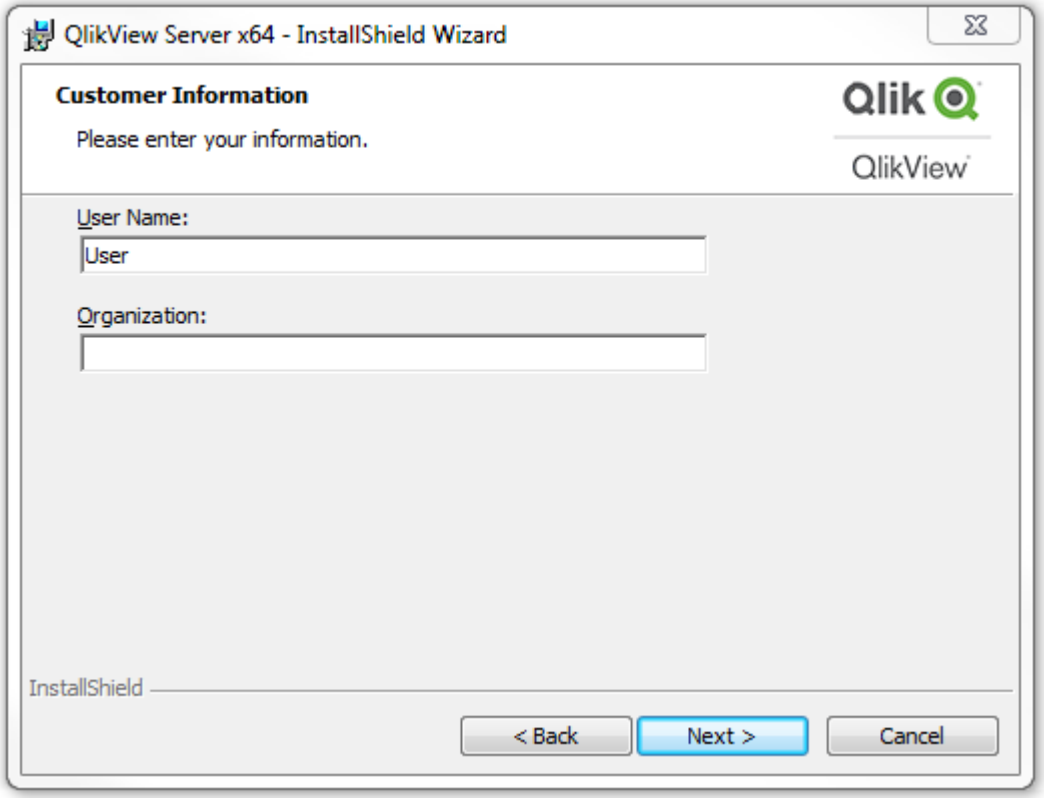
License dialog

Customer Information

This dialog is used for entering the customer information.

Properties:

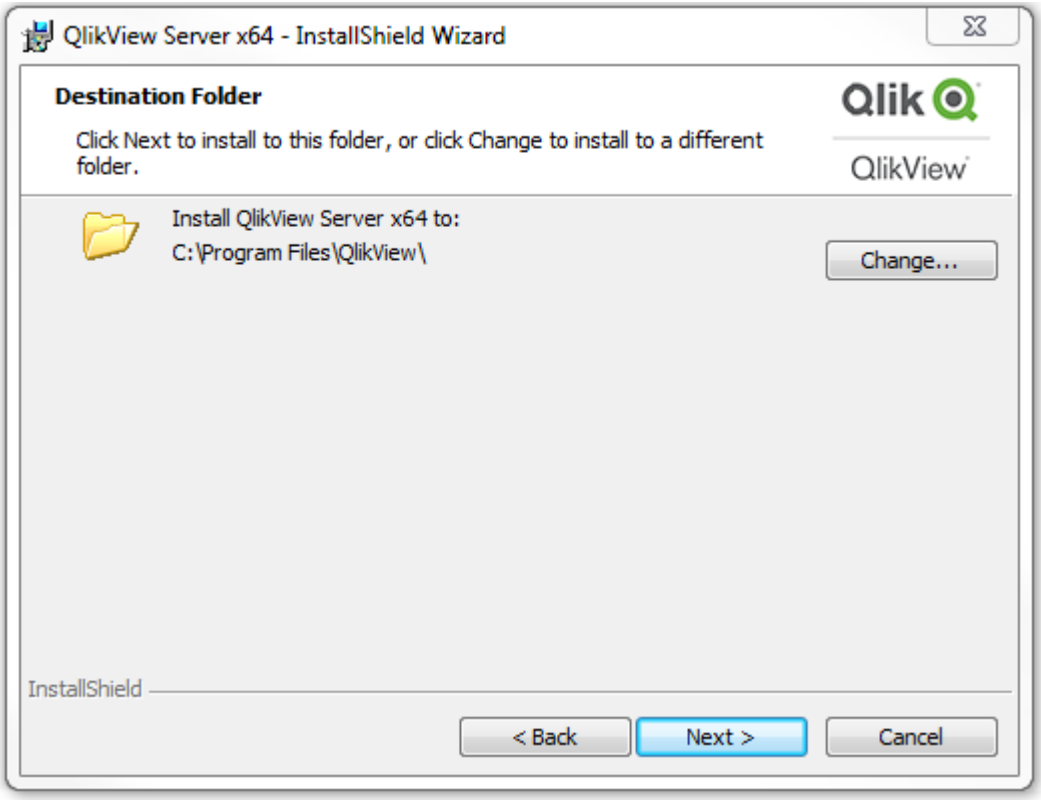
- *USERNAME*
- *COMPANYNAME*



Customer information dialog

Destination Folder

This dialog is used to set the default folder for the installation.



Destination folder dialog

Profiles

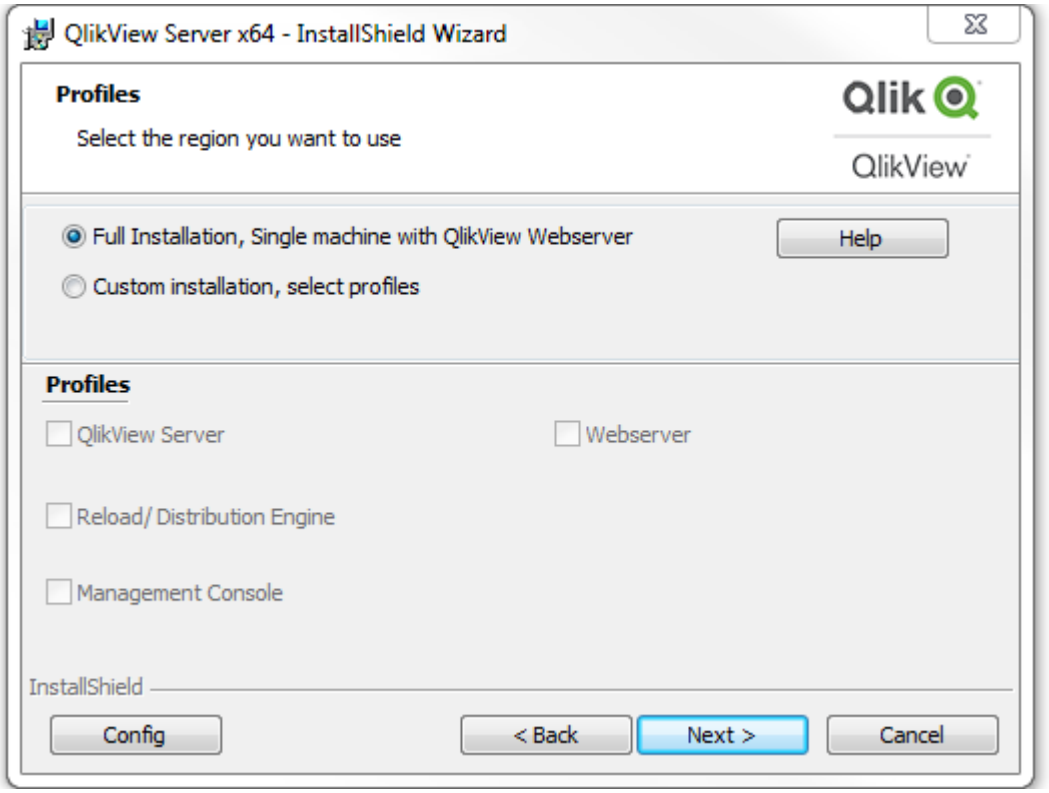
This dialog has several properties connected to it, since there are multiple profiles to choose from.

Select **Full Installation, Single machine with QlikView Webserver** to install everything, including QlikView Web Server, needed to run QlikView on a single machine. To use IIS instead, select **Full Installation, Single machine with IIS** (this option is only available if IIS is installed on the target machine).

To perform a custom installation, select **Custom installation, select profiles** and then select the profiles to install. The **Webserver** profile allows the user to choose between QlikView Web Server and IIS (if IIS is installed on the target machine).

Properties:

- *PROPQVS*: QlikView Server
- *PROPDS*: Publisher
- *PROPQMC*: Management Console
- *PROPWEB*, *PROPIIS* = 1 or 2: Webserver
- *PROPIIS* (if IIS is installed) or *PROPSTATE*: Single Machine Install



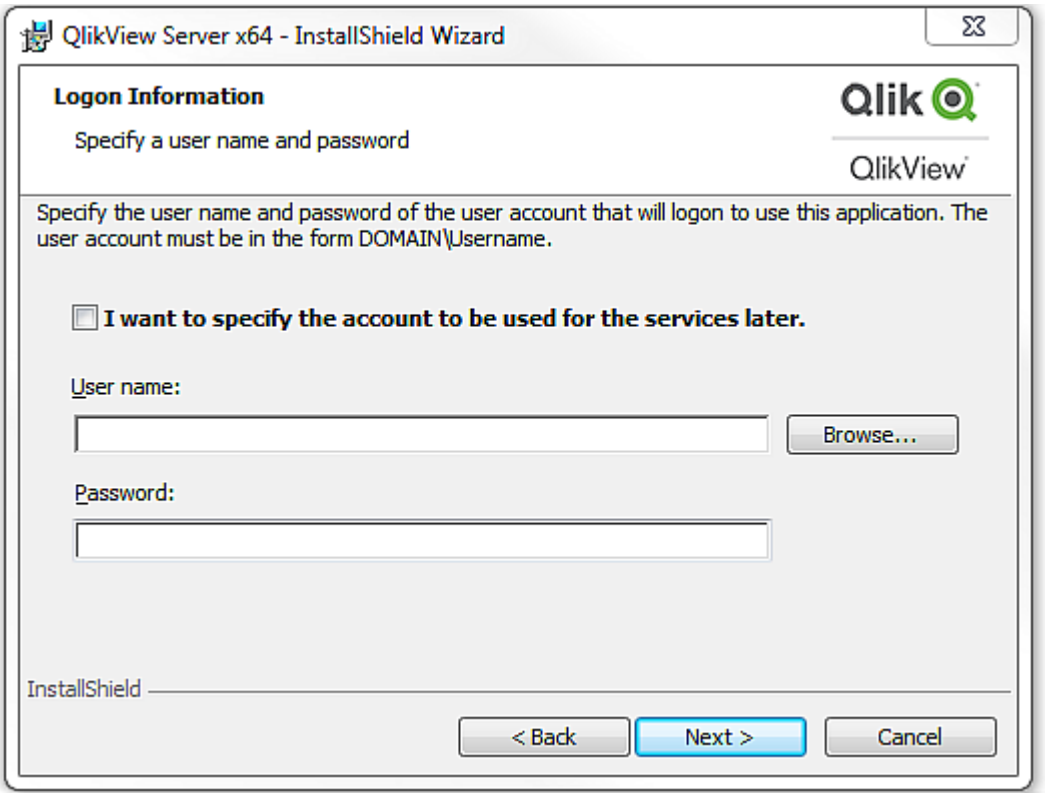
Profiles dialog

Logon Information

This dialog, which is optional to use, is used to specify the user that is to run the services that are installed. When clicking **Next**, a Custom Action checks that the entered user is valid. The Custom Action, which is implemented by InstallShield, requires the machine to be part of a Domain to work properly.

Properties:

- *LOCALSERVICE*
- *IS_NET_API_LOGON_USERNAME*
- *IS_NET_API_LOGON_PASSWORD*

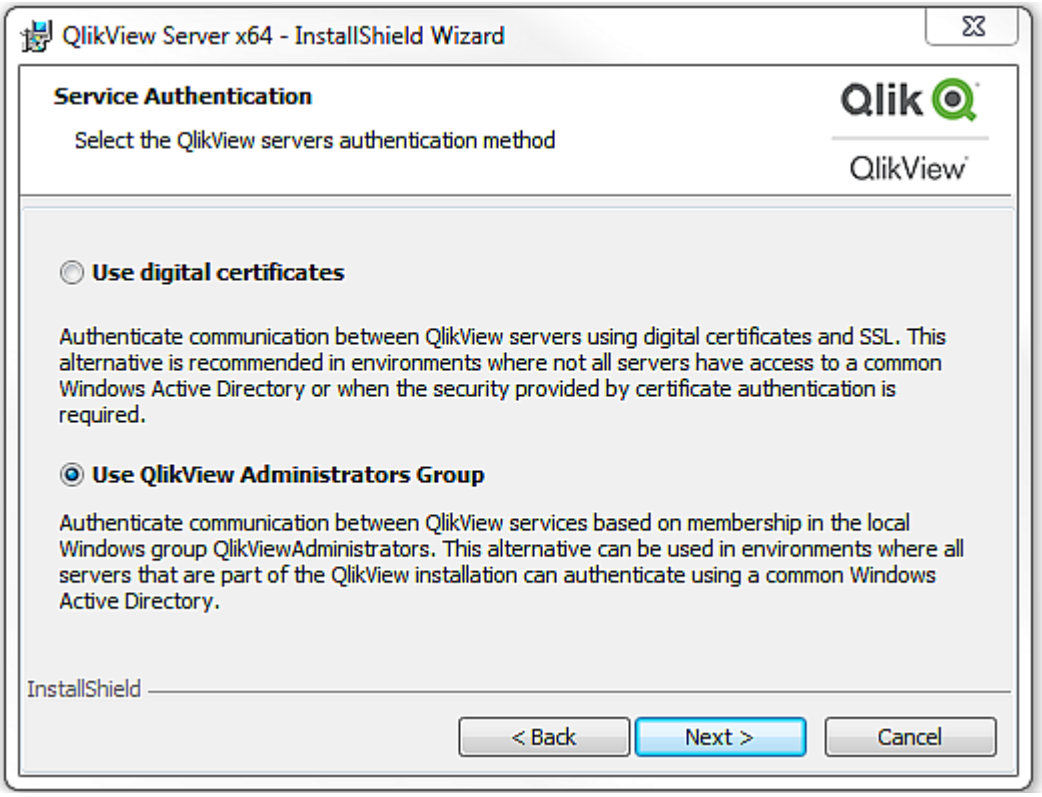


Logon information dialog

Service Authentication

This dialog is used to select the type of service authentication. QlikView Administrators Group is selected by default.

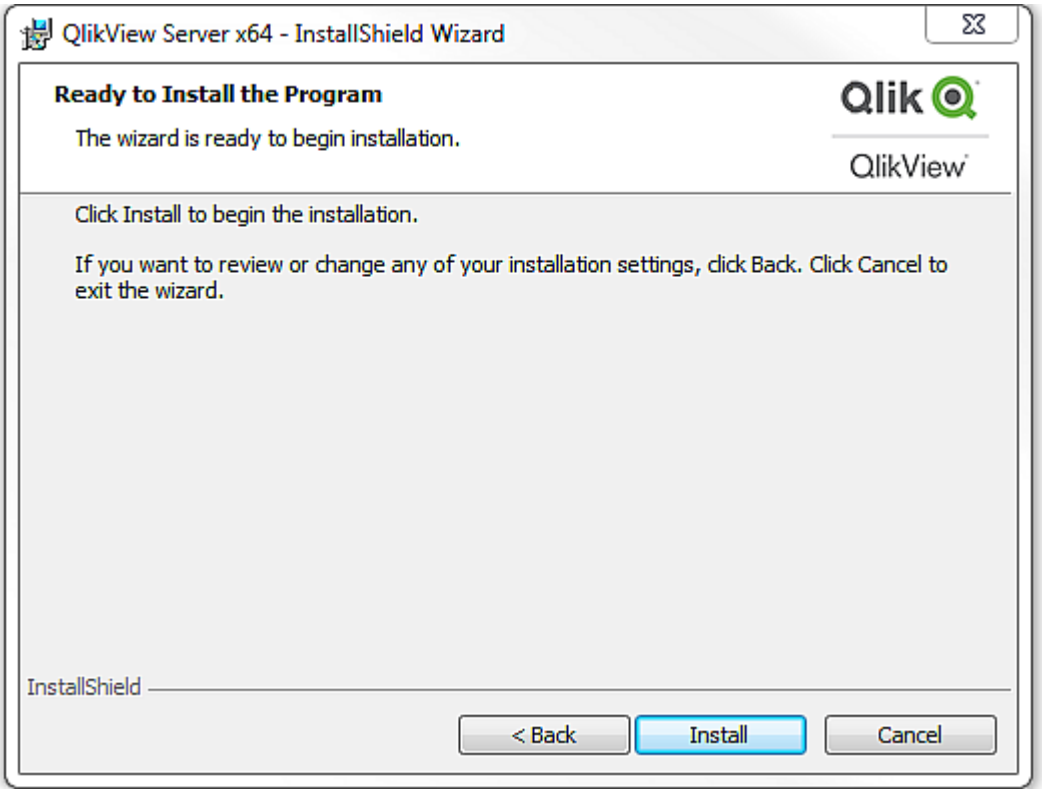
Property: *PROPCERT* (1 = Digital certificates, 2 = QlikView Administrators Group)



Service authentication dialog

Ready to Install

This is the last dialog. Click **Install** to start the installation.



Ready to install dialog

Additional Dialogs

Custom Setup

This dialog is displayed when clicking **Config** in the Profiles dialog.

Custom setup dialog

Website

This dialog is displayed when selecting IIS as web server in the Profiles dialog.

Property: *DEFAULTWEBSITE*

Website dialog

MST

When creating an MST file, the MSI file is customized without any changes being made directly in the MSI. The MST file works as a filter on top of the MSI and allows changes to be made to the installation. For example, the default installation folder for QlikView Server is *%ProgramFiles%\QlikView*, but if that is changed to *C:\QlikView* in the MST file, the default folder is changed. The same thing can be done with the dialogs, which means properties can be preset, so that the installation can be run with a limited set of dialogs.

To create an MST file, an MSI repackaging studio (for example, InstallShield AdminStudio) is needed.

> *Qlik does not supply any MST files and does not take any responsibility for MST files created by customers or partners.*

Silent uninstallation

The following silent uninstallation command line is recommended for QlikView:

QlikViewServer_x64Setup.exe /x /s /v"/qn"

The command line above removes all features completely silently.

Add a *+* sign at end of the interface levels command to get a modal dialog at the end of the installation saying “Finished” and if it was successful or not.

3.5 Configuring a proxy for Qlik License Service communication in QlikView Server

You can handle the communication between the Qlik License Service and the License Back-end with a proxy .

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

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

You can configure the communication between Qlik License Service and the Qlik License Back-end to be handled by a proxy.

In QlikView Server, configuration of a proxy for the Qlik License Service is done using command line parameters. Both HTTP and HTTPS scheme are supported.

With QlikView Server June 2020 or later NTLM and basic authentication capabilities to the licenses service when communicating over a HTTP tunnel are available. This allows you to require authentication on tunneling proxies and configure a more secure environment.

Do the following:

1. Stop the Qlik Service Dispatcher, which handles the execution of the Qlik License Service.
2. Navigate to the *service.conf* file, which by default is located in:
   %Program Files%\QlikView\ServiceDispatcher\service.conf
3. Locate the section `[licenses.parameters]`, which by default contains the following lines:
   ```
   [licenses.parameters]
   -qv-mode=true
   -app-settings="..\Licenses\appsettings.json"
   ```
4. Add the line `-proxy-uri=http://myproxy.example.com:8888` as shown below:
   ```
   [licenses.parameters]
   -qv-mode=true
   -proxy-uri=http://myproxy.example.com:8888
   -app-settings="..\Licenses\appsettings.json"
   ```
   Where "http://myproxy.example.com" is the address of your company's proxy, and "8888" is the port used by the proxy.

> *You can specify an IP address rather than a domain name as the proxy URI, for example* `-proxy-uri=http://10.76.124.124:1337`.

5. Browse to *%ProgramFiles%\QlikView\Licenses* and run `Encrypt-Password.ps1 [password for proxy access]`.

   Example:

   ```
   Encrypt-Password.ps1 123456
   ```

   Copy the generated encrypted password and use it in the next step.
6. To require authentication on tunneling proxies add the following lines to the *services.conf* file:

   ```
    -proxy-uri=[the uri of the proxy]
   -proxy-auth-mode=ntlm|basic|(leave empty for no authentication)
   -proxy-user=[username without domain]
   -proxy-encrypted-password=[password]
   -proxy-domain=[the domain] (only for NTLM)
   ```
7. Save and close the *services.conf* file.
8. Restart the Qlik Sense Service Dispatcher.
9. If you have a multi-node installation, repeat these steps for all the nodes in your installation.

3.6 Configuring preferred cipher suites for QlikView Server

You can rank the preferred cipher suites that Qlik License Service uses to encrypt and decrypt the signed key license.

The Qlik License Service is included in QlikView Server April 2019 and in later releases.

> *You can configure QlikView Server to use either mTLS or NTLM as your authentication protocol. If you use the NTLM service however, you cannot configure preferred cipher suites.*

If your Qlik License Service is set up to use certificate service authentication, then it uses Mutual TLS Authentication (mTLS). This protocol ensures that requests coming from both the server and client are trusted. The Qlik License Service listens on port 9200.

> *TLS 1.2 is supported since QlikView 12.0.*

The following list shows the supported cipher suites:

- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
- TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
- TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA25
- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384
- TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305
- TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305

- TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA
- TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA
- TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA
- TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA
- TLS_RSA_WITH_AES_128_GCM_SHA256
- TLS_RSA_WITH_AES_256_GCM_SHA384
- TLS_RSA_WITH_AES_128_CBC_SHA
- TLS_RSA_WITH_AES_256_CBC_SHA
- TLS_ECDHE_RSA_WITH_3DES_EDE_CBC_SHA
- TLS_RSA_WITH_3DES_EDE_CBC_SHA

To configure the preferred cipher suites for the Qlik License Service, do the following:

1. Open the *service.conf* file.
   The default path is *%Program Files%\QlikView\ServiceDispatcher\service.conf*.
2. Go to the following section:
   ```
   [license.parameters]
   -qv-mode=true
   -qv-auth-mode=cert
   ```
3. Add a comma-separated list of ciphers to his section, as shown below:
   ```
   [license.parameters]
   -qv-mode=true
   -cipher-suites=TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256,TLS_ECDHE_RSA_WITH_AES_128_CBC_
   SHA
   -qv-auth-mode=cert
   ```
4. Save the file and close.
5. Restart the QlikView Server Service Dispatcher, which handles execution of the Qlik License Service.
6. If you have a multi-node environment, repeat the steps above for each node.

3.7 Deploying MSI Packages with Group Policies

This chapter is mainly intended for the QlikView plugin.

General

A common problem today is how to deploy applications in a network environment where the users have limited rights, and how to deploy applications for a specific group of users. This section briefly describes how to deploy Microsoft Windows Installer (*.msi*) packages with group policies in an Active Directory environment.

The QlikView *.msi* packages require version 2.0 or higher of the Windows Installer service to be installed on the destination workstations.

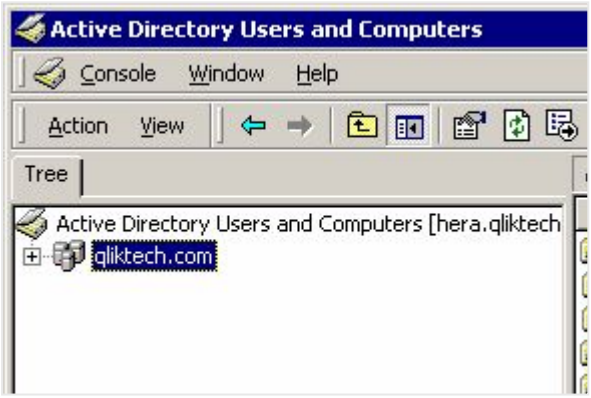
Deploying the MSI Package

When the *.msi* file has been obtained, it must be placed in a shared folder on the network. Make sure that all users and/or machines that are to install the application have read access to the folder. When the package has been made available to the users and/or machines, the Group policy object that will advertise the installation package can be created.

The package can be advertised to each user or each machine. Use the **User Configuration**>**Software Settings** container to advertise the package per user, and the **Computer Configuration**>**Software Settings** container to advertise per machine. Both containers are located in the Group Policy Object editor.

If the package is advertised per user, it can be either assigned or published. A package that is advertised per machine can only be published.

To publish a package per user means that it is listed (that is, advertised) in the Add programs from your network list in the Add/Remove programs dialog.



Add/Remove programs dialog

Each user must click the **Add** button to complete the installation.

To publish a package per machine means that the package is installed and accessible to all users on that machine the next time the machine is rebooted.

An advertised package that is assigned is also listed in the **Add programs from your network** list and can be added from there. This option also offers a few more ways to activate the installation package:

- Shortcuts (if the installation package adds any) on the desktop and/or Start Menu: The shortcuts are added and the installation package can be executed by clicking the appropriate shortcut.
- File association: The installation program is executed when the user tries to open a file that is associated with the advertised application.

There are a few more ways to execute the installation when it is advertised as assigned, but they are not applicable to any QlikView installations and therefore beyond the scope of this document.

The QlikView plugin installation package does not add any shortcuts or file associations. It is therefore not recommended to advertise QlikView installation packages with the assign option.

Advertising

To advertise means that the administrator gives the installation package permission to execute on an account with locked down permissions.

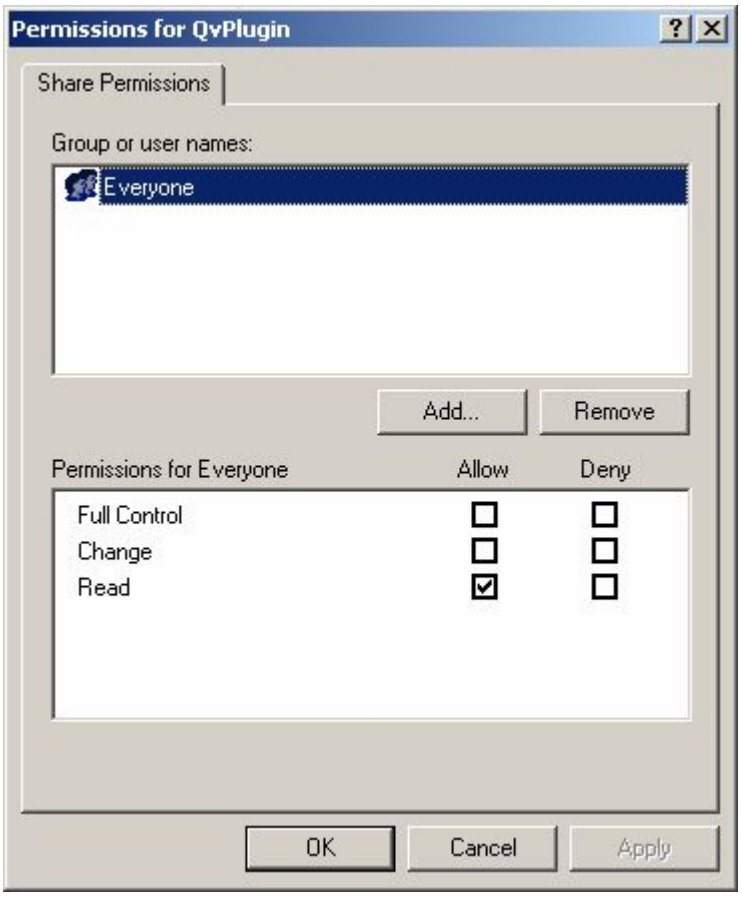
When the package is advertised, there are so called “entry points” loaded onto the destination system. Entry points are typically shortcuts, file associations, listing in the Add/Remove Programs dialog, and so on.

Step-by-step Guide

This section provides a step-by-step guide for creating a group policy for advertising of the QlikView plugin *.msi* package on a number of machines in the Active Directory.

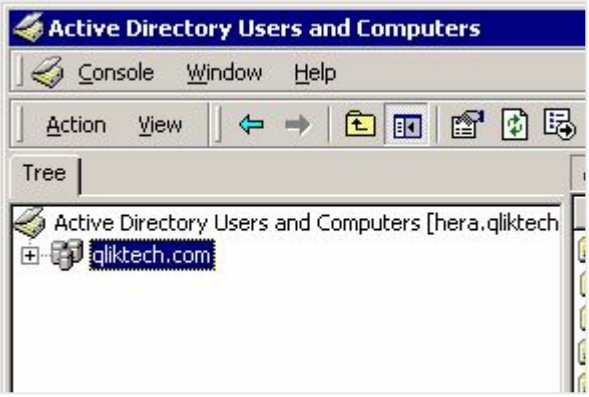
Proceed as follows to create a group policy:

1. Browse to the folder containing the *.msi* package. Share the folder with the network users with permission to install the package.



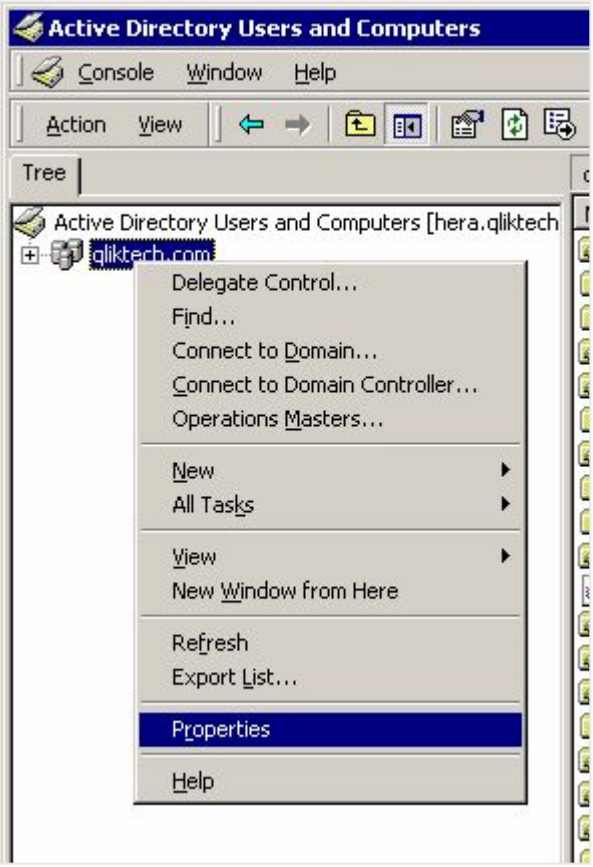
Sharing the folder

2. Open **Active Directory Users and Computers** and highlight the **Organizational Unit (OU)** where the package is to be deployed.



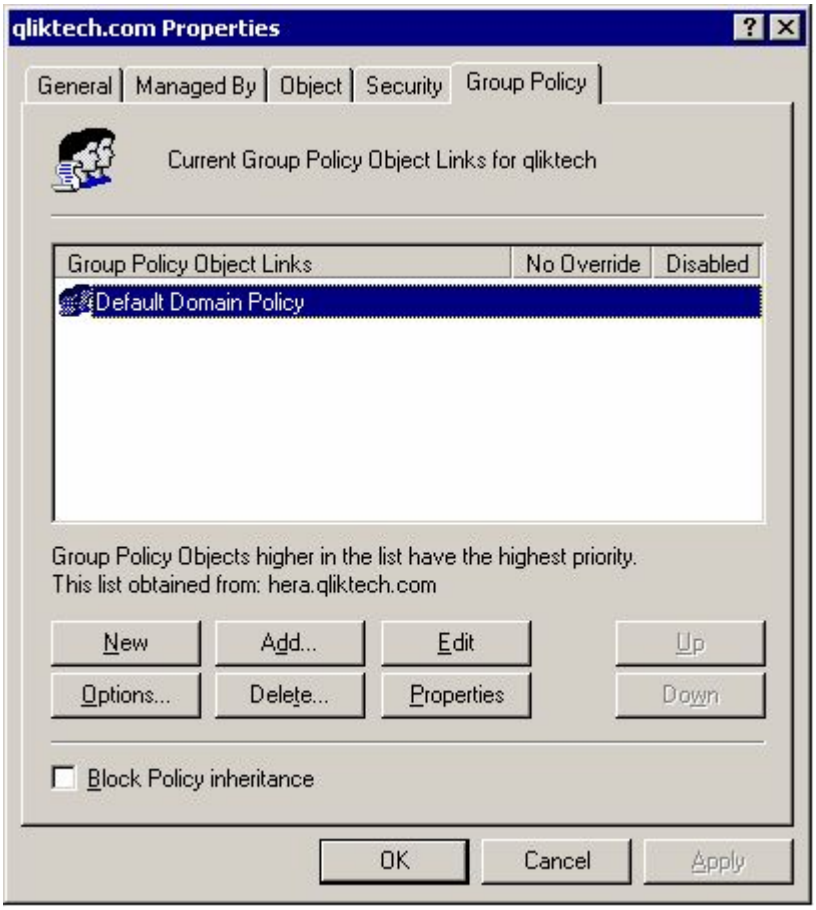
Highlighting the Organizational Unit where to deploy the package

3. Right-click and select **Properties**.



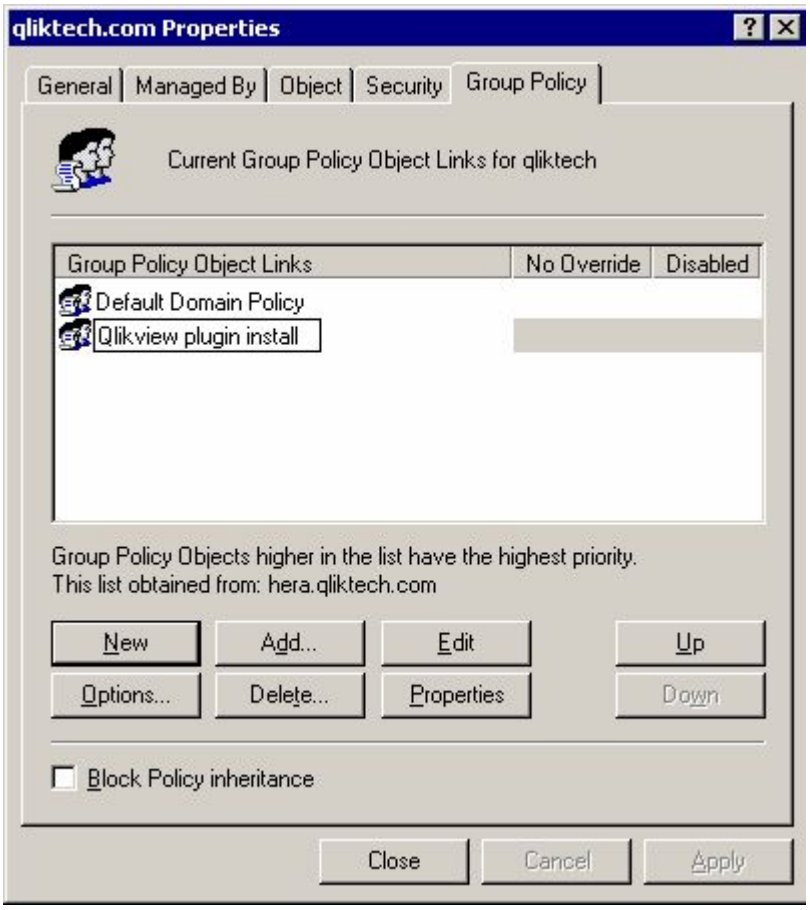
Selecting Properties

4. Go to the **Group Policy** tab, click **New**, and give the group policy object an appropriate name.



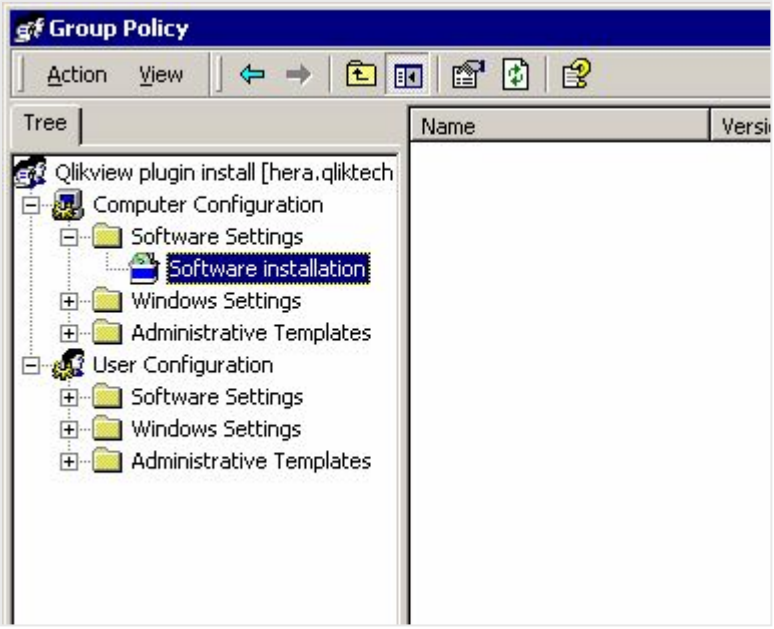
Providing a name

5. Highlight the new group policy object and click **Edit**.



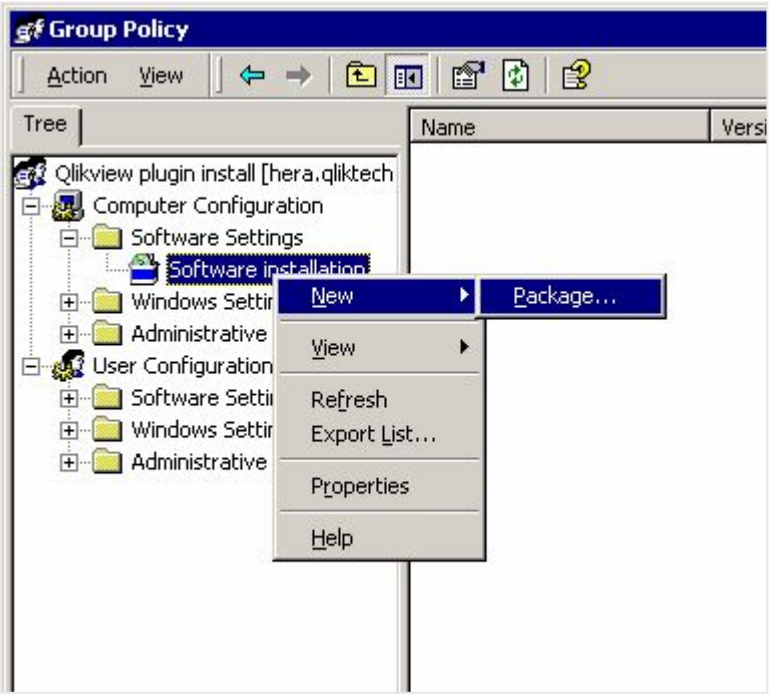
Highlighting the new group policy object

6. Expand **Computer Configuration**>**Software Settings** or **User Configuration**>**Software Settings**, depending on how the package is to be deployed. In this case, **Computer Configuration**>**Software Settings** is selected.



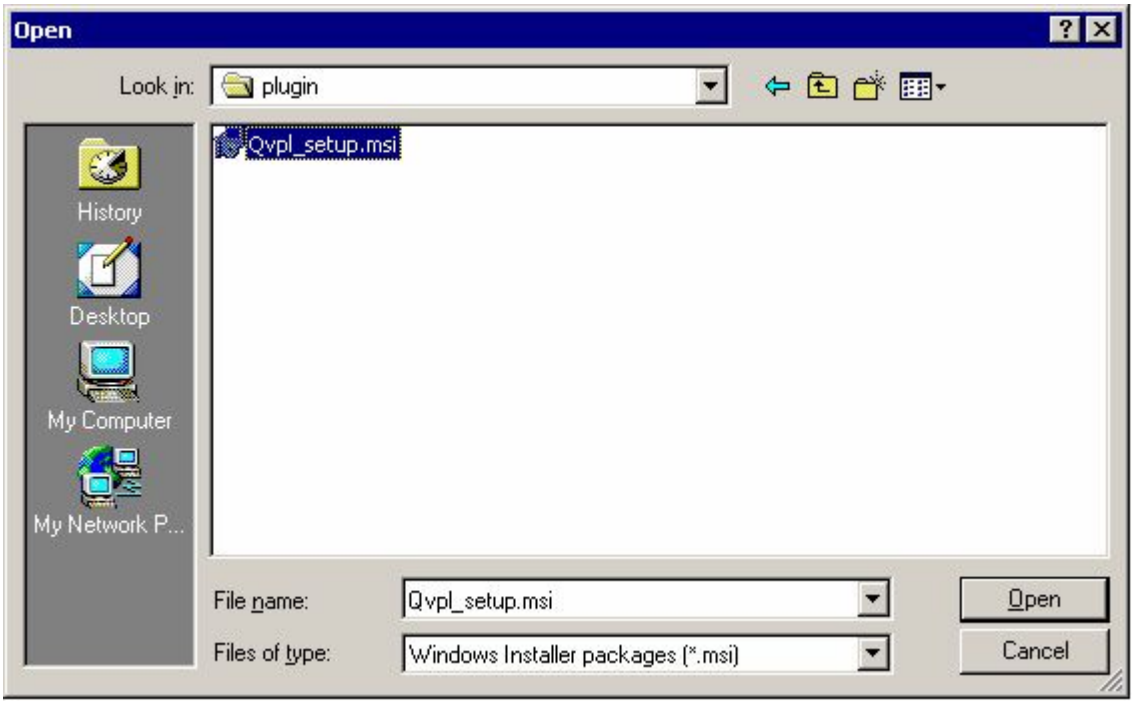
Selecting Software Settings

7. Right-click **Software installation** and select **New**>**Package...** A pop-up window, asking where to locate the installation package, is displayed.



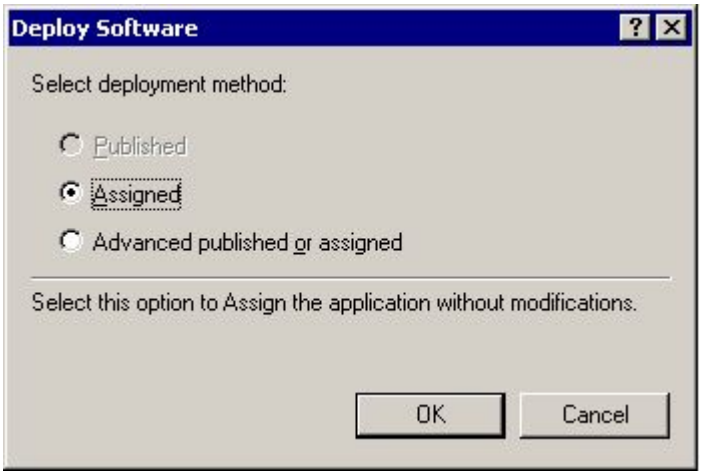
Creating a new package

8. Browse to the installation package (in this case, *QvPluginSetup.msi*), select it, and click **Open**.



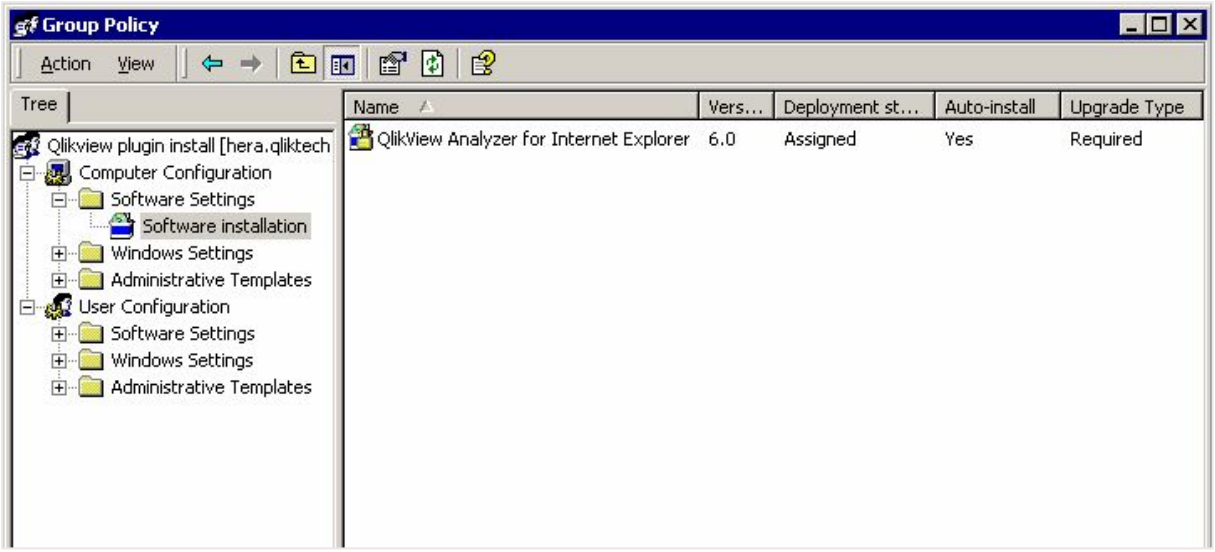
Opening the installation package

9. Select the deployment method **Assigned** and click **OK**. Since the installation is to be applied to the **Computer Configuration**, only the **Assigned** deployment method can be used.



Selecting deployment method

10. The deployment rule is now ready for use. All machines in the OU get this deployment automatically. What actually happens is that when a machine is rebooted, the installation program is executed, so that any user that logs on to a machine in that OU can run the installed program. The rule can be applied to many different OUs.



Deployment rule is ready for use

4 QlikView Upgrades and Updates

In this section, you find information on how to upgrade QlikView Server to the latest release. Here, you can also read about the requirements necessary for the upgrade to be successful, such as having a valid *Maintenance contract on upgrade (page 139)*. In the *Upgrading and migrating QlikView Server (page 140)* page you can also find information on how to migrate a QlikView Server deployment to a different machine or cluster of machines.

4.1 Maintenance contract on upgrade

When upgrading QlikView Server and QlikView Desktop, it is important that you have a valid maintenance contract. If you attempt to upgrade without a valid maintenance contract, the QlikView installation is restricted to an **unlicensed mode** with limited functionalities.

Every maintenance contract has a limited validity period and a specific end date. This means that your maintenance contract is not valid for a version of QlikView released after the end date indicated in your contract. However, the maintenance contract remains valid for any version of QlikView (Desktop or Server depending on what installation you have) released before the end date of the contract.

The information regarding the maintenance contract, including validity period and end date, is stored in the License Enabler File (LEF). You can verify the validity period of your maintenance contract by checking the end date stored in the LEF file. For QlikView Server, check the LEF file in the QMC. For QlikView Desktop, locate the LEF file in your local drive. Usually, the LEF is automatically transferred and stored in your computer during installation. However, there are instances when this procedure fails and the LEF file is not successfully transferred. For more information about this scenario, see the License Enabler File Editor page.

The validity of the maintenance contract is automatically checked during upgrade. If you are unsure whether your maintenance contract is valid for the QlikView version you want to upgrade to, you should refrain from upgrading. When you attempt to upgrade without a valid maintenance contract, your QlikView installation (Desktop or Server) is restricted to an **unlicensed mode** with limited functionalities. In QlikView Desktop, the unlicensed mode is called Personal Edition.

> *If you launch the upgrade but your maintenance contract is not valid for that specific QlikView version, a warning message is displayed. This warning message is displayed **only** when the maintenance contract is not valid for the requested QlikView version.*

In case your QlikView Desktop or Server installation is restricted to the unlicensed mode, you can revert it to the previous version. Follow this downgrade procedure.

Restoring an older QlikView Desktop installation

To restore a previous installation of QlikView Desktop, you must uninstall the current instance and install the older version for which you own a valid maintenance contract.

Restoring an older QlikView Server installation

Before installing a newer version of QlikView Server, read carefully the procedure in *Backup and upgrade preparation (page 149)*. This procedure helps you create a backup of your QlikView Server installation, including the QlikView Publisher Repository (QVPR) database. Creating this backup before upgrading allows you to successfully restore a previous version of QlikView Server in case the newer installation is restricted to the unlicensed mode.

To restore a previous installation of QlikView Server, follow these steps:

- Uninstall the unlicensed instance of QlikView Server
- Install an older version of QlikView Server for which you have a valid maintenance contract.
- Restore the QVPR and data directory backups.

4.2 Upgrading and migrating QlikView Server

When you upgrade QlikView Server, you can either upgrade on the same server, or you can upgrade and migrate to a different server. This topic outlines the steps you need to follow to upgrade to a newer version of QlikView Server, and includes the steps for migrating your installation, including certificates, to another server with a different machine name.

This documentation describes how to upgrade and migrate an installation running QlikView Server 12.00 or later. If your installation runs QlikView Server 11.20 or an earlier version, see: *Upgrading and migrating QlikView Server from 11.20 to November 2017 or later (page 145)*. For a description of how to upgrade QlikView Desktop, see: Upgrading and Updating QlikView Desktop.

At the upgrade, shared files with the *.Shared* file name extension are not automatically converted to *.Tshared* files. For more information on how to convert files, see *Cleaning and converting the shared files (page 82)*. All bookmarks and user objects in shared files are preserved during the upgrade.

Requirements

If you are using Qlik NPrinting, check that the version of QlikView you are upgrading to is compatible with your current version of Qlik NPrinting. Consult the Qlik NPrinting system requirements at Qlik NPrinting Engine Engine at System requirements for Qlik NPrinting Engine. For more information about upgrading Qlik NPrinting, see Upgrading Qlik NPrinting.

Best practices

For a successful upgrade of QlikView, take the following basic practices into account:

- Ensure that you have created an appropriate backup of your QlikView Server installation. For a detailed list of the files and folders to backup, see *Backup and upgrade preparation (page 149)*.
- Perform the upgrade during a scheduled downtime. All QlikView services must be stopped for the upgrade to be successful.
- Licensing information and settings are saved by default when QlikView Server is removed. They are applied to any subsequent installation of QlikView Server on the system.

- Ensure that you have a valid maintenance contract before upgrading your QlikView Server installation. Attempting to upgrade without a valid maintenance contract will result in limited functionality of QlikView Server. See: *Maintenance contract on upgrade (page 139)*.
- If Digital Certificate Authentication is used for communication between QlikView services, the new certificates created during the upgrade must be installed on all machines, except for the machine running the QlikView Management Service (QMS). See: *Updating certificates (page 115)*.

Upgrading on the same machine

When you upgrade your QlikView deployment on the same machine, you upgrade each server independently. If you have multiple servers, repeat these instructions for each server. Before you begin, plan your upgrade to maximize deployment up-time.

Maximize up-time

Follow this procedure to maximize system up-time for end users:

> *Before you begin, make sure you back up your QlikView deployment.*
>
> *Backup and upgrade preparation (page 149)*

1. Stop the QlikView Management Service (QMS). When you do this, the QlikView Management Console becomes unavailable.
2. Upgrade the QlikView services in the following order (let the installer restart the services):
   a. QlikViewWeb Server (QVWS)
   b. Directory Service Connector (DSC)
   c. QlikView Server (QVS)
   d. QlikView Distribution Service (QDS)
   e. QlikViewManagement Service (QMS)
3. Start the QMS, so that the QlikView Management Console becomes available again.

Upgrading a single-node deployment

To upgrade QlikView Server on a single-server site:

> *Before you begin, make sure you back up your QlikView site.*
>
> *Backup and upgrade preparation (page 149)*

1. If your installation uses digital certificates for authentication, create a backup of the certificates. Certificates are updated automatically when you upgrade a QlikView Server installation on the same machine. However, we recommend that you create a backup of your certificates anyway. See: *Backing up and restoring certificates (page 155)*.
2. Download the latest version of QlikView Server from ≤ Product Downloads.
   For more information, see *Downloading installation files (page 111)*.

3. Stop the QlikView services. During the upgrade process all the services are stopped and restarted automatically, but we still recommend that you stop all services before upgrading. To avoid interrupting running tasks, shut down only the QlikView Distribution Service (QDS) by performing a graceful shutdown right before the upgrade procedure. For more information on graceful shutdown, see: Shutting down a QlikView Distribution Service.
4. Run the installation program as an administrator, and follow the on screen instructions. For a step-by-step description of the install procedure, see: *Installing QlikView Server (page 106)*.
5. During the installation, select a service authentication method. Choose either **Use digital certificates** or **Use QlikView Administrators Group**. If you previously used digital certificates then you should also choose this option when you upgrade.
6. Restart your machine once the installation process has finished to ensure that all services start up correctly.
7. Open QlikView Management Console and apply your license information for QlikView Server, and QlikView Publisher.
8. Restart the server to apply the license information.

Upgrading a multi-server deployment

To upgrade a multi-server site:

1. Stop all services on each machine before you perform the upgrade.
2. If your installation uses digital certificates for authentication, create a backup of the certificates stored on the machine running the QlikView Management Service. Certificates are updated automatically when you upgrade QlikView Server on the same machine. However, we recommend that you create a backup of your certificates anyway. See: *Backing up and restoring certificates (page 155)*.
3. Perform the upgrade procedure on each machine of the multi-server setup.
4. If digital certificates are used, install the new certificates on all machines running QlikView services except for the machine running the QlikView Management Service (QMS). For a detailed description of this procedure, see: *Updating certificates (page 115)*.

Upgrading to a different machine

The simplest way to upgrade to a different machine is to first upgrade on the current machine and then to migrate to the new target machine. If you are upgrading a multi-server site to a new machine, repeat these steps for each server.

Install QlikView Server on the target machine

1. Download the latest version of QlikView Server from ≤ Product Downloads.
   For more information, see *Downloading installation files (page 111)*.
2. Launch the QlikView Server install wizard, and follow the on screen instructions. For a step-by-step description of the installation procedure, see: *Installing QlikView Server (page 106)*.
3. During the installation, select a service authentication method. Choose either **Use digital certificates** or **Use QlikView Administrators Group**. Make sure to select **Use digital certificates** if the installation on your current machine uses certificates as the authentication method.

4. Restart your machine once the installation process has finished to ensure that all services start up correctly.
5. Open the QlikView Management Console and apply your license information for QlikView Server, and QlikView Publisher.

> *As a QlikView Server license is applicable to only one installation at a time, shut down the installation on the current machine before applying your license information to the new installation on you target machine.*

Migrate and restore your QlikView Server backup

Once you have installed the latest version of QlikView Server on both your current and target machines, you can proceed to migrate the content of your installation from the current machine and restore it on the target machine. The procedure varies depending on whether your installation uses QlikView Administrators Group or digital certificates as the authentication method.

Before starting the migration procedure, make sure to create a backup of your current QlikView Server installation. Without an appropriate backup, you won't be able to restore your QlikView Server installation on the target machine. See *Backup and upgrade preparation (page 149)*.

> *If your installation uses digital certificates for authentication, it is vital that you create a backup of the certificates on your current machine. See: Backing up and restoring certificates (page 155).*

Migrating an installation that uses QlikView Administrators Group

1. Once you have installed the latest version of QlikView Server on both the current and target machine, create an appropriate backup of the current machine. See *Backup and upgrade preparation (page 149)*.
2. Stop all QlikView services on both current machine and target machine.
3. On the target machine, remove or rename the *%ProgramData%\QlikTech\ManagementService\QVPR* folder, as this will be replaced with your backed up version.
4. Copy the QVPR folder from your current machine to your target machine (make a note of the folder name).
5. Edit all *.xml* files in the following folders, replacing all references to the current machine name with the target machine name.
   - *%ProgramData%\QlikTech\ManagementService*
   - *%ProgramData%\QlikTech\ManagementService\QVPR*
6. Restart the QlikView services. Start the QlikView Management Service first, then wait a minute, and start the other services in any order.
7. Restore the SourceDocuments folder and Mounted Folders.

- If you are storing your source documents in the default *%ProgramData%\QlikTech\SourceDocuments* folder, then move all your source documents to the same location on the target machine.
- If you are storing your source documents in a different folder location, then add the source documents folder path in the QlikView Management Console. To do this see the *Add* section in Source Folders.
- If you are distributing tasks to mounted folders, reinsert the path to the mounted folders. To do this see the *Mounter Folders* section in Folders.

8. Shut down your old server machine.

Migrating an installation that uses digital certificates

When you migrate a QlikView Server installation that uses certificates, some settings are encrypted. These settings cannot be decrypted if QlikView cannot access the certificates originally used for the encryption. Restoring the certificates from your current machine to the target machine for the migration allows you to decrypt the migrated settings. Once decrypted, these settings are encrypted again using the encryption key stored in the certificates from the target machine.

1. Once you have installed the latest version of QlikView Server on both the current and target machine, create an appropriate backup of the current machine, including certificates. See *Backup and upgrade preparation (page 149)*.
2. Stop all QlikView services on both current machine and target machine.
3. Copy the certificates backup from your current machine to your target machine and save it in a preferred location.
4. Restore the certificates from the current machine to the target machine using the MMC (Microsoft Management Console). For a step-by-step description of this procedure, see: *Restoring certificates (page 156)*. Once the old certificates are restored, you should see two sets of certificates in the MMC, with two different expiration dates.
5. On the target machine, remove or rename the *%ProgramData%\QlikTech\ManagementService\QVPR* folder, as this will be replaced with your backed up version.
6. Copy the QVPR folder from your current machine to your target machine (make a note of the folder name).
7. Edit all *.xml* files in the following folders, replacing all references to the current machine name with the target machine name.
   - *%ProgramData%\QlikTech\ManagementService*
   - *%ProgramData%\QlikTech\ManagementService\QVPR*
8. Restart the QlikView services. Start the QlikView Management Service first, then wait a minute, and start the other services in any order.
9. Restore the SourceDocuments folder and Mounted Folders.
   - If you are storing your source documents in the default *%ProgramData%\QlikTech\SourceDocuments* folder, then move all your source documents to the same location on the target machine.

- If you are storing your source documents in a different folder location, then add the source documents folder path in the QlikView Management Console. To do this see the *Add* section in Source Folders.
- If you are distributing tasks to mounted folders, reinsert the path to the mounted folders. To do this see the *Mounter Folders* section in Folders.

If the decryption of the migrated files was successful, your tasks, bookmarks and all other custom settings should now be in place and visible in the QlikView Management Console.

10. Shut down your old server machine.

Upgrading and migrating a multi-server deployment

When upgrading and migrating a multi-server installation, perform the above procedures for each machine in your multi-server deployment.

Here you find a summary of the necessary steps:

1. Perform the upgrade procedure on each machine of the multi-server setup.
2. Perform a backup of each current machine in your multi-server installation.
3. If your installation uses digital certificates for authentication, create a backup of the certificates on the machine running the QlikView Management Service (QMS).
4. Install a running, licensed version of QlikView Server on each of the target machines.
5. If your installation uses digital certificates for authentication, restore the certificates on the target machine where the QMS is installed.
6. Migrate and restore the backup specific for each machine of the installation. For example, the QVPR folder must be migrated (and the machine name changed) only to the target machine running the QlikView Management Service.
7. Restore the SourceDocuments folder.
8. Shut down the old server machines.
9. If Digital Certificate Authentication is used, install the new certificates on all machines running QlikView services except for the machine running the QlikView Management Service (QMS). For a detailed description of this procedure, see: *Updating certificates (page 115)*

Upgrading and migrating QlikView Server from 11.20 to November 2017 or later

When you upgrade QlikView Server, you can either upgrade on the same server, or you can upgrade and migrate to a different server. This topic outlines the steps you need to follow to upgrade from QlikView Server 11.20 to a newer version, and includes the steps for migrating to another server with a different machine name.

Before you perform the upgrade and migration, ensure that you have created an appropriate backup of your QlikView Server deployment. For a detailed list of the files and folders to backup, see *Backup and upgrade preparation (page 149)*.

For a detailed list of changes and issues when upgrading from QlikView Server 11.20, read the following Qlik Support article: QlikView 11.20 End of Life Upgrade: Known Issues and changes in product behaviour.

> *When you upgrade from QlikView Server 11.20 to QlikView Server 12.10 or later, your installation might encounter a variety of issues due to backend file system. QlikView Server 12.10 and later versions are more disk intense and require bigger file server than QlikView 11.20. When planning your QlikView deployment, it is important to keep in mind the type of storage and resource capacity. For more information, read the following Qlik Support article: QlikView and its backend File Share System.*

Upgrading on the same machine

To upgrade QlikView Server on the same machine:

1. Create a backup of your QlikView Server 11.20 installation. It is always important to create a backup of your installation, however this step is not essential if you are upgrading on the same machine. See *Backup and upgrade preparation (page 149)*.
2. Download the latest version of QlikView Server from ≤ Product Downloads.
   For more information, see *Downloading installation files (page 111)*.
3. Stop the QlikView services. During the upgrade process all the services are stopped by the installer, and restarted automatically, but it is still recommended that you stop all services before proceeding with an upgrade.
4. Run the installation program as an administrator, and follow the on screen instructions. For a step-by-step description of the install procedure, see: *Installing QlikView Server (page 106)*.

   > *When you upgrade from a QlikView Server 11.20 installation using certificates to QlikView Server November 2017 or later, remove the certificates before starting the upgrade. When you launch the installation program, a warning window is displayed if certificates are detected. You remove the certificates using the Microsoft Management Console (MMC). For more information, see: Removing certificates (page 157)*

5. During the installation, select a service authentication method. Choose either **Use digital certificates** or **Use QlikView Administrators Group**. If you previously used digital certificates then you should also choose this option when you upgrade.

   > *If you are upgrading from QlikView 11.20 to November 2017 or later and choose digital certificates, there is no need to backup and restore your old certificates. New certificates are generated and installed automatically the first time you start the QlikView Management Service.*

6. Restart your machine once the installation process has finished to ensure that all services start up correctly.
7. Open the QlikView Management Console, and apply your license information for QlikView Server, and QlikView Publisher.
8. Restart the server to apply the license information.

Upgrading on a multi-server deployment

To upgrade a multi-server installation:

- Stop all services on each machine before you perform the upgrade.
- Perform the upgrade procedure on each machine of the multi-server setup.

Upgrading and migrating to a different machine

To upgrade and migrate QlikView Server to a different machine follow the steps in this section.

Upgrade QlikView Server to a different machine:

1. On your current machine, create a backup of your QlikView Server 11.20 installation. See *Backup and upgrade preparation (page 149)*.
2. Download the latest version of QlikView Server from the ≤ Product Downloads.
3. Launch the QlikView Server upgrade wizard, and follow the on screen instructions. For a step-by-step description of the installation procedure, see: *Installing QlikView Server (page 106)*.
4. During the installation, select a service authentication method. Choose either **Use digital certificates** or **Use QlikView Administrators Group**. If you previously used digital certificates then you should also choose this option when you upgrade.

   > *If you are upgrading from QlikView 11.20 to November 2017 or later and choose digital certificates, there is no need to backup and restore your old certificates. New certificates are generated and installed automatically the first time you start the QlikView Management Service.*

5. Restart your machine once the installation process has finished to ensure that all services start up correctly.
6. Open the QlikView Management Console, and apply your license information for QlikView Server, and QlikView Publisher.
7. You will be prompted to restart the server after applying the license information.

Migrate and restore your backup:

1. Stop all QlikView services on both the current machine and the target machine.
2. On the target machine, remove or rename the *ProgramData\QlikTech\ManagementService\QVPR* folder, as this will be replaced with your backed up version.
3. On the target machine, remove or rename the `qvpr_<TargetMachineName>.ini` file located in *ProgramData\QlikTech\ManagementService*.
4. Copy the QVPR folder and the `qvpr_<CurrentMachineName>.ini` file from your current machine to your target machine (make a note of the folder name).
5. Rename the `.ini` file from `qvpr_<CurrentMachineName>.ini` to `qvpr_<TargetMachineName>.ini`.
6. In the .xml files in the *ProgramData\QlikTech\ManagementService* folder, and in the `Config.xml` file, change all references to the current machine name to point to the target machine name.
7. Restart the QlikView services. Start the QlikView Management Service first, then wait a minute, and start the other services in any order.
8. Restore the **SourceDocuments** folder.
   When you restore the source documents folder, you have two options:

- If you are storing your source documents in the default *ProgramData\QlikTech\SourceDocuments* folder, then move all your source documents to the same location on the target machine.
- If you are storing your source documents in a different folder location, then add the source documents folder path in the QlikView Management Console. To do this see the *Add* section in Source Folders.

9. Shut down your old server machine.

Upgrading and migrating a multi-server deployment

When upgrading and migrating a multi-server installation from QlikView Server 11.20 to QlikView Server November 2017 or later, perform the above procedure for each machine in your multi-server deployment.

Summary of steps:

1. Perform a backup of each machine in your multi-server installation.
2. Install a running, licensed version of QlikView Server on each of the target machines.
3. Migrate and restore the backup specific for each machine of the installation. For example, the QVPR folder must be migrated (and the machine name changed) only to the target machine running the QlikView Management Service.
4. Restore the Source Documents folder.
5. Shut down the old server machines.

5 Backup and Restore QlikView

In this section, you will find information on how to create a complete backup of your QlikView Server installation. If your QlikView Server installation uses digital certificates for authentication, it is vital that you create a backup of your certificates and keep them in a secure location. Here, you will find dedicated documentation on how to backup and restore certificates.

5.1 Backup and upgrade preparation

When you upgrade from an older version of QlikView to the latest version, ensure that you have prepared your environment correctly by performing appropriate backups. Before upgrading, you should back up all important files to a safe location, including any customizations that you have created since your original QlikView deployment. This topic aims to provide a basic checklist of files you need to back up, and other important considerations. The guidance in this topic applies to a standalone, or multi-server deployment using a QlikView web server.

Backing up files

You can manually back up files, or create your own backup script to automatically back up files to your chosen location. In a QlikView deployment, the most important files to back up are contained in the QlikTech folder in *ProgramData*, and in the QlikView folder under *Program Files*. Make a copy of both of these directories to ensure a successful backup.

QlikView Server data directories

In a single-node QlikView Server deployment, the most important directory to back up is the QlikTech folder located in *C:\ProgramData\QlikTech*. This directory contains a sub folder for each of the QlikView services. Each sub folder contains configuration and settings files that you may have edited if you have customized your deployment. It is important that you back up these files if you want to restore your original configuration when you upgrade your QlikView server.

Use the QlikView Management Console to get an overview of all configuration files contained in the application data folder. In the QMC, you can see the location of the configuration files, file paths, and other custom settings that you may have changed.

When you back up a QlikView Server deployment, you typically back up the following:

- QVPR database (backed up to a .zip file)
- Configuration files (.config files)
- Settings files (.ini files)
- Log files
- Documents
- Bookmarks (stored in *.Shared*, or *.TShared* files)
- User objects (stored in *.Shared*, or *.TShared* files)
- Tasks (stored in the QVPR database)

When you upgrade from a QlikView Server 11.20 installation using certificates to QlikView Server November 2017 or later, remove the certificates before starting the upgrade. When you launch the installation program, a warning window is displayed if certificates are detected. You remove the certificates using the Microsoft Management Console (MMC). For more information, see: Removing certificates (page 157)

The following tables provide more information about these items, and where they are stored in your deployment.

ProgramData

ProgramData folders

Folder name	Description of content
DirectoryServiceConnector	Configuration and settings files Log files Resources folder and service key
DistributionService	Configuration and settings files Log files A version of the tasks is sent from the QVPR database to the QlikView Distribution Service, so if the Distribution Service folder is lost, the tasks can still be restored from the QVPR database backup.
Documents	*.qvf* or *.qvw* files and other files related to your documents Bookmarks, and user Objects are stored in *.Shared* files
ManagementService	This is a crucial folder to back up as it includes the QVPR database, and a *Backup* folder. The QVPR database is automatically backed up here every day as a *.zip* file. This is the only data directory that is automatically backed up, to reduce the risk of file corruption. Configuration and settings files Log files Tasks - ensure that you back up this folder to save all your tasks.
QlikView Documentation	PDF help documentation
QlikViewBatch	This folder contains QVB log files, and only needs to be backed up if you have enabled logging. In QlikView November 2017, and later this folder also contains some QVS related log files.

QlikViewServer	This folder is very important to back up. It contains the *Settings.ini*, which is the equivalent to the *.exe.config* files in Program Files for the other services.
SourceDocuments	This folder contains the Source Documents, used by the QlikView Publisher to create User Documents.
WebServer	Log files Configuration file `service_key`

The *Program Files* folder contains configuration files that are not accessible from the QMC, so these files can only be edited manually. The most important files here to back up are the QlikView services config files, which contain important configuration, and settings files.

Program Files

Program Files folders

Folder name	Description of contents
DirectoryServiceConnector	Contains the QVDirectoryServiceConnector configuration file. Important to back up if you have made manual changes.
DistributionService	Contains the QVDistributionService configuration file. Important to back up if you have made manual changes.
Examples	Contains sub folders, for example QlikView documents and all related data.
ManagementService	Contains the QVManagementService configuration file. Important to back up if you have made manual changes.
QvPlugin	Languages to support localized help content.
QvProtocol	Contains *qvp.dll*
Server	The Web Server sub folder which contains the QVWebServer configuration file. Important to back up if you have made manual changes.
Themes	If you have created custom themes, it is important to include this folder in the backup.
Web	Contains the *Web.config* configuration file. Important to back up if you have made manual changes.

QVPR database files

The QVPR database is backed up as a .zip file in the following location: *C:\ProgramData\QlikTech\ManagementService\QVPR\Backups*. The backup file contains .xml and .bak files that need to be migrated to your new QlikView environment.

This is part of the backup of data directories, but is the most important component because it contains configuration files and settings for the QlikView Publishing Repository. By default, the QVPR is backed up daily as a zip file to *C:\ProgramData\QlikTech\ManagementService\QVPR\Backups*. You can change the frequency of backup, and the location where backup files are stored in the QMC.

QlikView Web Server or Microsoft IIS

When upgrading between major versions of QlikView, the Microsoft IIS settings get automatically reverted to default. If your installation uses a Microsoft IIS web server, it is important that you backup your Microsoft IIS settings before performing the upgrade.

This topic focuses on how to create a back up when using the QlikView Web Server. If you use a Microsoft IIS web server, refer to the Microsoft documentation for instructions on how to back up your IIS websites, certificates, and configuration files. The principles for backup are similar for both types of web server. However, if you have created, for example, a custom authentication solution that runs with IIS, either refer to your own documentation for backup locations, and changes, or contact the consultant that originally created the customization for you.

Log files

Log files contain important information that might help to troubleshoot problems in your deployment. The log files can contain many entries, so if storage is a problem, only select the files you want to keep. However, it is recommended that you back up all log files.

Licenses

Licensing information, and settings are saved by default when a QlikView Server is removed. This information is then reapplied to any subsequent installation of QlikView Server. QlikView uses two licenses which are added to the following locations when you upgrade:

- QlikView Server license - This license file (LEF) is stored in *C:\ProgramData\QlikTech*.
- QlikView Publisher license - This license file (LEF) is stored in *C:\ProgramData\QlikTech\ManagementService\Publisher LEF*.

Certificates

When you install QlikView you can choose the QlikView Administrators Group for digital authentication or you can choose to install digital certificates. If you choose digital certificates, and you are running QlikView 12.00, or later, then it is very important that you back up the certificates.

For more information on how certificates work in QlikView, see: *Certificate Trust (page 161)*.

A single, standalone QlikView server always uses the following three certificates:

Certificates

Location	Issued To	Issued By	Description
Local Computer / Personal	*<machine-name>*	QlikViewCA	Server
Local Computer / Personal	QVProxy	QlikViewCA	Client

Location	Issued To	Issued By	Description
Local Computer / Trusted Root Certification Authorities	QlikViewCA	QlikViewCA	Root

Certificates are used to authenticate the different services in the QlikView installation, and secure data that may be sensitive. When you upgrade and restore, you need the certificates for authentication so that QlikView services can start up and run correctly.

In QlikView 12.00, and later it is very important that you never delete certificates, and always back them up to a secure location.

If you are running QlikView 11.20 or earlier, a different method of encryption is used. This means that old certificates cannot be restored in the new installation, and new certificates need to be generated.

When you upgrade from a QlikView Server 11.20 installation using certificates to QlikView Server November 2017 or later, remove the certificates before starting the upgrade. When you launch the installation program, a warning window is displayed if certificates are detected. You remove the certificates using the Microsoft Management Console (MMC). For more information, see: Removing certificates (page 157)

Updating certificates

In a multi-server QlikView environment, the QlikView Management Service (QMS) is your certificate authority, acting like a central node for handling and distributing certificates. The QMS needs to have the root, service, and client certificates installed before you can generate certificates to add more machines.

When you upgrade or restore a new machine to your QlikView deployment, you distribute certificates using the QlikView Management Console. There must be an exchange of certificates between the QMS and the machine you want to add. To do this, in the QMC, you enter the URL to the machine you want to add. When you click **Apply**, a pop-up window opens and displays the password you need to enter on the remote server machine. After this password has been accepted, the service or services on the remote server machine must be restarted. Once the services have been restarted, the process of generating and distributing a certificate to another machine is complete.

For more information, see: *Updating certificates (page 115)*.

Backing up certificates

Use the MMC (Microsoft Management Console) to backup all digital certificates to your chosen location. For a step-by-step description of the procedure, see: *Backing up and restoring certificates (page 155)*.

Backing up custom content

You may have added custom content to your deployment, such as a custom authentication, or custom security solution. This customization may have been created by your own organization, or by an external

consulting company, but whatever the origin, you are responsible for documenting and backing up your own customizations. If you are unsure about how to back up or migrate any customizations, we recommend you contact the consultant that originally created them.

Changing default file locations

As part of your QlikView deployment it may have been necessary to change some default file locations. When you perform an upgrade, the original folder locations will be restored and any custom file locations will be lost. Before you back up, you can use the QMC to get an overview of all paths to your configuration and settings files that by default are stored in the QlikTech application data folder under *ProgramData*. To keep folders in your own chosen locations after upgrade, you need to create a backup of any custom file paths you have created. Open the relevant configuration file to see the custom locations you have added.

Configuration files

The QlikView services all have configuration files that you can edit to suit the requirements of your deployment. There are configuration and settings files for each QlikView service, and these are stored in the sub folders in your QlikView data directories. For more details, see the following table: *QlikView Server data directories (page 149)*. If you have a created customizations of your own, you will have edited the configuration file for the relevant QlikView service.

From QlikView November 2018, you can monitor all non-default values applied to the services in your deployment (except for the QlikView Server Service, QVS) directly from the QlikView Management Console (QMC). In the QMC, go to **status** > **services** and select one of the services in your QlikView Server deployment. If one or more custom config values are in place, they are listed in the information tab on the right-hand side of the screen. The list shows which config settings are modified, on which machine, and what is the existing value compared to the default value. Discrepancies between machines running the same service are listed as well. For more information, see: Services.

If you restore your changed configuration and settings files to your upgraded deployment, you may overwrite important changes that Qlik has added in subsequent release and updates. Therefore, to avoid loss, it is important to note down all the changes, or customizations that you have made to your configuration and settings files. Once you have completed the upgrade process, you need to append your changes to the new config files. If you follow this approach, it will ensure that your deployment includes all the latest config files, as well as all the customizations from your earlier deployment.

For example, if you have created your own customizations in the QlikView Distribution Service, you will have made changes to the *QVDistributionService.exe.config* file. To back this up, navigate to *C:\Program Files\QlikView\Distribution Service* folder, and make a copy of the *QVDistributionService.exe.config* file. Then, after upgrade, append your customizations to this file.

Multi-server deployments

If backing up QlikView from another machine, cluster, or other location, make a backup of each machine that you want to upgrade. Follow the same procedure as you would for a single server installation. In the case of multi-server deployments using a shared application data folder (file share) in a different location, ensure that this folder is also included in the backup.

5.2 Backing up and restoring certificates

Backing up certificates

It is vital that you back up your certificates and keep them in a secure location. If the certificates are lost, your sensitive data will be lost.

Service failure due to undecryptable data (page 116).

Here is a list of the three QlikView certificates that you must backup on the server running the QlikView Management Service (QMS):

Certificates

Location	Issued To	Issued By	Description
Local Computer / Personal	*<machine-name>*	QlikViewCA	Server
Local Computer / Personal	QVProxy	QlikViewCA	Client
Local Computer / Trusted Root Certification Authorities	QlikViewCA	QlikViewCA	Root

Since the QlikView Management Service (QMS) creates and distributes certificates to all services in the QlikView installation, it is optional to back up the certificates on the servers running the other services. If certificates are missing for any of these services, the QMS distributes new certificates to the machines that are part of the deployment.

Use the MMC (Microsoft Management Console) to backup the certificates to your chosen location. For more information on the MMC, see: *Using Microsoft Management Console (page 158)*.

To backup certificates:

1. Open the MMC.
2. Click **File**, and then click **Add/Remove Snap in**.
3. Select **Certificates** and then click **Add**.
4. Select **Computer account**, and click **Next**.
5. Select **Local computer**. Click **Finish** and then click **OK** on the main window.
6. Expand the **Certificates** node, and select the following certificate folders:
   - **Personal**
   - **Trusted Root certificate Authorities**
7. Right click the certificate that you want to back up, click **All Tasks**, and then click **Export**.
8. In the **Certificate Export Wizard**, select **Yes, export private key**, and click **Next**.
9. Select **Export all extended properties** and **Include all certificates in the certification path if possible**. Then, click **Next**.

> *Make sure you export the private key and export all extended properties.*

10. Enter and confirm a password. Then click **Next**.
11. Enter a file name, and choose a location for your backup, then click **Next**.
12. Click **Finish** to create the backup.

For more information on locating your certificates, and how to back them up, see *Certificate Trust (page 161)*.

Restoring certificates

If the certificates are missing for any reason, the services will close down and information can be found in the log files. If certificates were previously backed up, they can be restored using the MMC (Microsoft Management Console) on the machine running the QlikView Management Service. If there is no backup to use for restoring, the inaccessible data (the protected secret information) has to be cleared and later on reentered.

The following tab lists the three certificates that need to be restored:

Certificates

Location	Issued To	Issued By	Description
Local Computer / Personal	*<machine-name>*	QlikViewCA	Server
Local Computer / Personal	QVProxy	QlikViewCA	Client
Local Computer / Trusted Root Certification Authorities	QlikViewCA	QlikViewCA	Root

To restore the certificates:

1. Open the MMC.
2. Click **File**, and then click **Add/Remove Snap in**.
3. Select **Certificates** and then click **Add**.
4. Select **Computer account**, and click **Next**.
5. Select **Local computer**. Click **Finish** and then click **OK** on the main window.
6. Expand the **Certificates** node, and select the following certificate folders:
   - **Personal**
   - **Trusted Root certificate Authorities**
7. Right click the **Certificates** folder under **Trusted Root Certification Authorities**, click **All Tasks**, and then click **Import**.
8. In the **Certificate Import Wizard**, browse to the location where you stored the certificates backup. To visualize the certificates files, select **Personal Information Exchange (*.pfx;*.p12)** format in the drop-down menu next to **File name**.
9. Select the Root certificate and click **Open**. Then, click **Next**.

10. Enter the password that was created when the certificates were exported. Select **Mark this key as exportable** and **Include all extended properties**. Click **Next**.
11. In the following windows, click **Next** and then **Finish**.
12. If the import was successful, the certificate is now listed in the MMC.
13. Repeat steps 7 to 12 to import Server and Client certificates in the **Certificates** folder under **Personal**.

Services failure due to missing certificates

If the QMS service fails, a new set of certificates with a new random SecretsKey is created at startup. The QMS may now be asked for certificates by other services.

If any of the other services fails, the service starts in a special mode where the service can handle certificates from the QMS. You need to browse to a certain port on the local machine and enter a password presented by the QlikView Management Console. After this, the service must be restarted and will then run in its normal mode, using the newly received certificates and keys.

Service failure due to undecryptable data (page 116)

Restoring certificates when migrating a QlikView Server installation

When you migrate a QlikView Server installation that uses certificates, some settings are encrypted. These settings cannot be decrypted if QlikView cannot access the certificates originally used for the encryption. Restoring the certificates from your current machine to the target machine for the migration allows you to decrypt the migrated settings. Once decrypted, these settings are encrypted again using the encryption key stored in the certificates from the target machine.

For more information on migrating a QlikView Server installation, see: *Upgrading and migrating QlikView Server (page 140)*.

Removing certificates

We recommend to never remove your certificates. If certificates are lost, your sensitive data will be lost. However, you have to remove certificates in very specific situations, like when you upgrade QlikView Server from 11.20 to November 2017 or later.

Use the Microsoft Management Console (MMC) to remove the certificates. See: Using Microsoft Management Console.

1. Open the MMC.
2. Click **File**, and then click **Add/Remove Snap in**.
3. Select **Certificates** and then click **Add**.
4. Select **Computer account**, and click **Next**.
5. Select **Local computer**. Click **Finish** and then click **OK** on the main window.
6. Expand the **Certificates** node, and select the following certificate folders:
   - **Personal**
   - **Trusted Root certificate Authorities**
7. Delete only the following certificates:

Certificates

Location	Issued To	Issued By	Description
Local Computer / Personal	*<machine-name>*	QlikViewCA	Server
Local Computer / Personal	QVProxy	QlikViewCA	Client
Local Computer / Trusted Root Certification Authorities	QlikViewCA	QlikViewCA	Root

Make sure you delete only the certificates listed above.

Configuration files

The following table lists the location of each of the configuration files that may need editing.

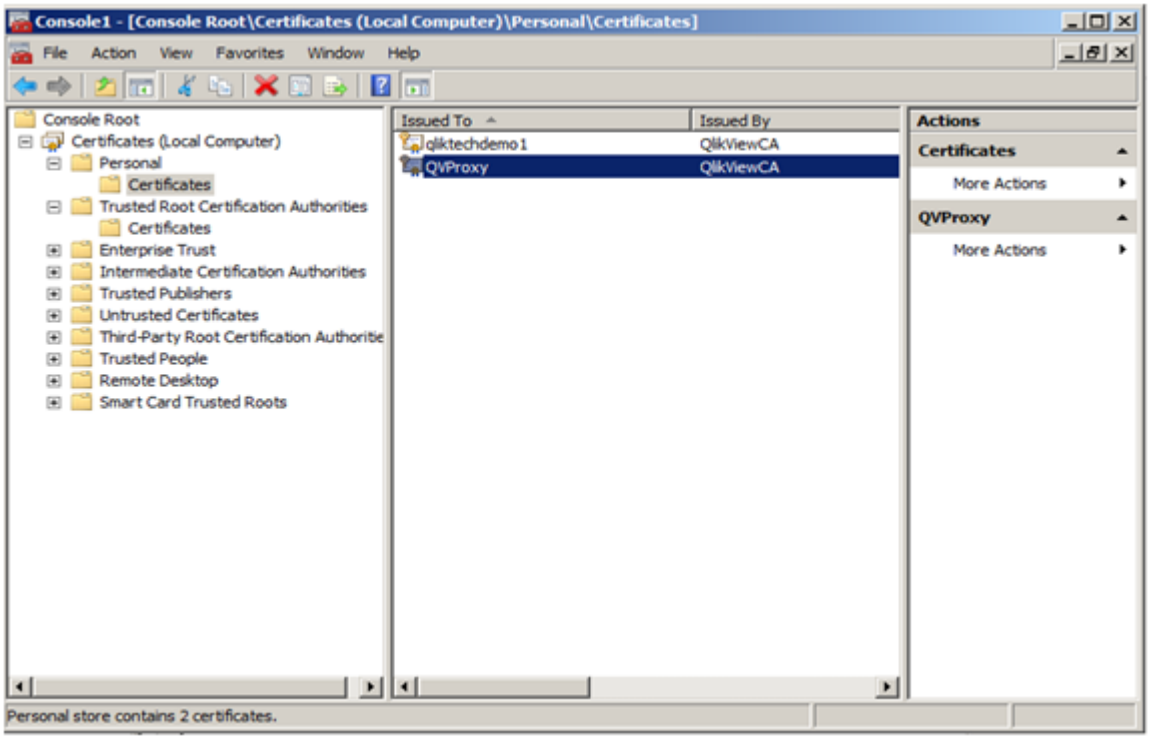
Configuration files

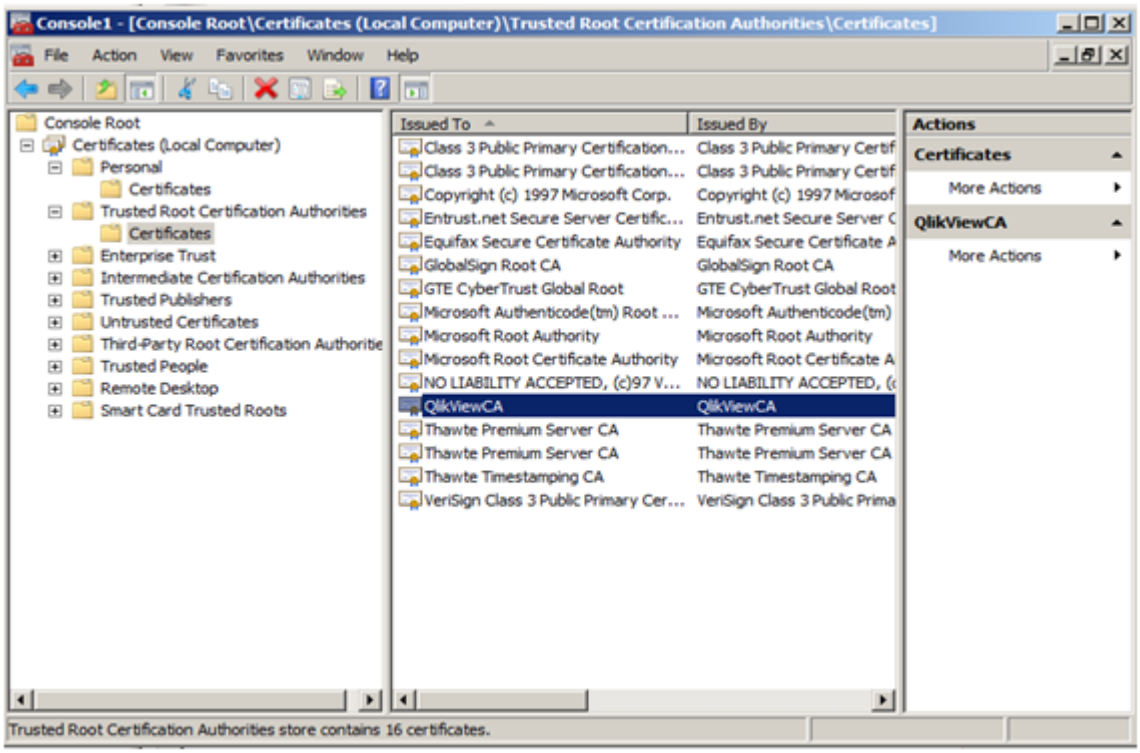
Service	Default Path
QMS	*C:\Program Files\QlikView\Management Service\QVManagementService.exe.config*
DSC	*C:\Program Files\QlikView\Directory ServiceConnector\QVDirectoryServiceConnector.exe.config*
QDS	*C:\Program Files\QlikView\Distribution Service\QVDistributionService.exe.config*
QVWS	*C:\Program Files\QlikView\Server\Web Server\QVWebServer.exe.config*
IIS	*C:\Program Files\QlikView\Server\Web Server Settings\QVWebServerSettingsService.exe.config* *C:\Program Files\QlikView\Server\QlikViewClients\QlikViewAjax\web.config*
QVS	*C:\ProgramData\QlikTech\QlikViewServer\Settings.ini*

Using Microsoft Management Console

Certificates can be visually confirmed in the QlikView Management Console with the certificate snap-in added. The QlikView certificates are located in the **Personal**>**Certificates** and **Trusted Root Certification Authorities**>**Certificates** folders.

The figures below show properly installed certificates in a QlikView Server configuration. Within the QlikView Management Console, all QlikView services on servers have certificates deployed as shown in the figures.


Console1 - [Console Root\Certificates (Local Computer)\Personal\Certificates]
File Action View Favorites Window Help
Console Root
Certificates (Local Computer)
Personal
Certificates
Trusted Root Certification Authorities
Certificates
Enterprise Trust
Intermediate Certification Authorities
Trusted Publishers
Untrusted Certificates
Third-Party Root Certification Authoritie
Trusted People
Remote Desktop
Smart Card Trusted Roots
Issued To
Issued By
qliktechdemo1
QlikViewCA
QVProxy
QlikViewCA
Actions
Certificates
More Actions
QVProxy
More Actions
Personal store contains 2 certificates.


Console1 - [Console Root\Certificates (Local Computer)\Trusted Root Certification Authorities\Certificates]
File Action View Favorites Window Help
Console Root
Certificates (Local Computer)
Personal
Certificates
Trusted Root Certification Authorities
Certificates
Enterprise Trust
Intermediate Certification Authorities
Trusted Publishers
Untrusted Certificates
Third-Party Root Certification Authoritie
Trusted People
Remote Desktop
Smart Card Trusted Roots
Issued To
Issued By
Class 3 Public Primary Certification...
Class 3 Public Primary Certif
Class 3 Public Primary Certification...
Class 3 Public Primary Certif
Copyright (c) 1997 Microsoft Corp.
Copyright (c) 1997 Microsof
Entrust.net Secure Server Certific...
Entrust.net Secure Server C
Equifax Secure Certificate Authority
Equifax Secure Certificate A
GlobalSign Root CA
GlobalSign Root CA
GTE CyberTrust Global Root
GTE CyberTrust Global Root
Microsoft Authenticode(tm) Root ...
Microsoft Authenticode(tm)
Microsoft Root Authority
Microsoft Root Authority
Microsoft Root Certificate Authority
Microsoft Root Certificate A
NO LIABILITY ACCEPTED, (c)97 V...
NO LIABILITY ACCEPTED, (c
QlikViewCA
QlikViewCA
Thawte Premium Server CA
Thawte Premium Server CA
Thawte Premium Server CA
Thawte Premium Server CA
Thawte Timestamping CA
Thawte Timestamping CA
VeriSign Class 3 Public Primary Cer...
VeriSign Class 3 Public Prima
Actions
Certificates
More Actions
QlikViewCA
More Actions
Trusted Root Certification Authorities store contains 16 certificates.

6 Security

The security of QlikView Server/Publisher consists of the following parts:

- Certificates: In QlikView Server, if you choose digital authentication, you use certificates for authentication and authorization. A certificate provides trust between servers. In addition, dynamic encryption keys are used for sensitive data.
- Protection of the platform: How the platform itself is protected and how it needs to communicate and operate.
- Authentication: Who is the user and how can the user prove it? QlikView uses standard authentication protocols, such as Integrated Windows Authentication (IWA), HTTP headers, and ticketing, to authenticate every user requesting access to data.
- Document level authorization: Is the user allowed to access the document or not? QlikView uses server-side capabilities such as Document Metadata Service (DMS) or Windows NTFS to determine access privileges at file level.
- Data level authorization: Is the user allowed to see all of the data or just parts of it? QlikView implements row and field level data security, using a combination of document-level capabilities (Section Access) and server-side data reduction capabilities (QlikView Publisher).

6.1 Certificates

A certificate is a data file that contains keys that are used to encrypt communication between a client and a server in a domain. Certificates also confirm that the domain is known by the organization that issued the certificate. A certificate includes information about the keys, information about the identity of the owner, and the digital signature of an organization that has verified that the content of the certificate is correct. The pair of keys (public and private keys) are used to encrypt communication.

Qlik products use certificates when they communicate with each other. They also use certificates within products, for communication between components that are installed on different computers. These are standard TLS certificates.

The organization that issues the certificate, the Certificate Authority, is said to “sign” the certificate. You can arrange to get certificates from a certificate authority, to show your domain is known. You can also issue and sign your own (“self-signed certificates”).

Some common errors

Because it generally important for security to know whether a site is known, browsers will display error messages related to certificates and might block communication.

Some common errors are related to the certificate authority. For example, if there is no certificate authority or if the certificate has expired, the default level of security in most browsers will stop communication with a message about “unsigned certificates”, “expired certificates”, or similar terms. If your security administrators know that the certificate is still good, you can create an exception so the error is ignored for that certificate.

Other common errors are related to how the domain is named. For example, companyname.com is a different domain from www.companyname.com, and localhost is a different domain from a server name. A fully qualified domain name is an unambiguous name for a domain. For example, a server at companyname.com might be named mktg-SGK, and can be referred to that way, but the fully qualified domain name is mktg-SGK.companyname.com. (This is called whitelisting.)

Encryption and keys

The kind of encryption used in certificates in Qlik products requires a pair of keys (asymmetric encryption). One key, the public key, is shared. The other key, the private key, is used only by the owner.

PEM is an ASCII text format for public certificates. It is portable across platforms.

You can get certificates and key pairs from certificate authorities or you can generate them. To get a certificate signed, you will need to also generate a signing request.

Certificate Trust

In QlikView Server, if you choose digital authentication, you use certificates for authentication and authorization. A certificate provides trust between servers machines. In addition, dynamic encryption keys are used for sensitive data. The default configuration in QlikView relies on Windows trust (hard-coded cryptographic keys).

> *Certificates contain encryption keys so it is vital to keep a backup of the certificates in a safe place. See: Backing up and restoring certificates (page 155)*

> *You must reference the QlikView Server by its machine name, and not by the IP address or fully qualified domain name.*

Architecture

In a QlikView Server installation, certificates authenticate and authorize communication between services running on multiple servers. The certificates include a SecretsKey that handles encryption and decryption of data such as passwords and connection strings.

Configuring certificates in a multiple server deployment within QlikView removes the dependency on a QlikView Administration Group for establishing trust . You can also use certificates to build a trust domain between QlikView services that are located in different domains without having to share an Active Directory (AD) or other user directories.

> *The configuration steps described here only provide a trust domain between the QlikView services. The use of SSL/TLS and certificates for securing end-user communication has to be configured separately.*

QlikView Server uses the following digital certificates for authentication and authorization:

Certificates

Location	Issued To	Issued By	Description
Local Computer / Personal	*<machine-name>*	QlikViewCA	Server
Local Computer / Personal	QVProxy	QlikViewCA	Client
Local Computer / Trusted Root Certification Authorities	QlikViewCA	QlikViewCA	Root

Certificates are managed from the Microsoft Management Console (MMC).

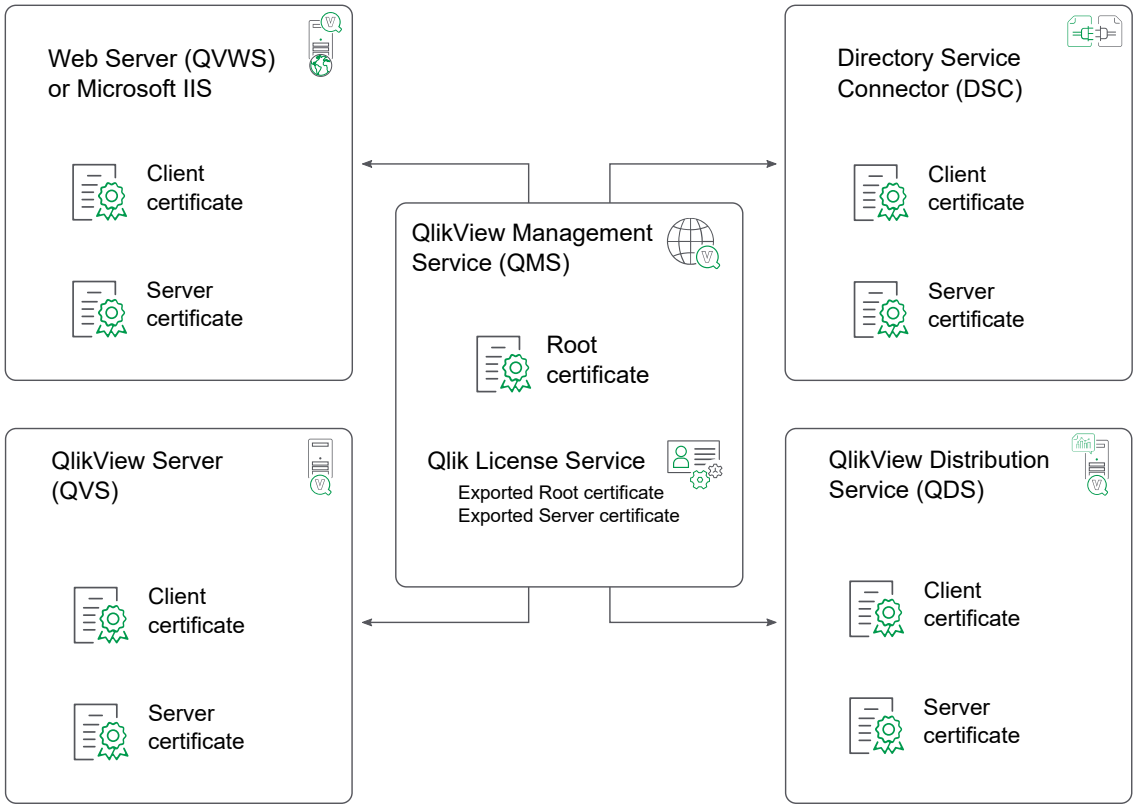
The architecture is based on the QlikView Management Service (QMS) acting as the certificate manager or Certificate Authority (CA). The QMS can create and distribute certificates to all services in the QlikView installation.

QMS 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 QMS server. All servers with QlikView services that are to participate in the installation receive certificates signed using the root certificate when added to the QMS. The QMS (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 QlikView services. The certificate enables the QMS to validate the authenticity of the service. This means that the QMS is responsible for saying “yes, this service deployed on this server is a service in my installation”.

After the servers have received certificates, the communication between the QlikView services is encrypted using HTTPS (SSL/TLS encryption). The certificates only secure the communication between the services on the servers. The certificates do not secure the communication with the end user (that is, the certificates are not used for QlikView plug-in, client, or web server communication with the QVS).

The following diagram shows a multi-node QlikView Server deployment where the QMS (the Certificate Authority) distributes the certificates to the machines where the other services are installed.



Qlik License Service

In QlikView April 2019 or later, the Qlik License Service is always installed and actively used only when QlikView Server is licensed using a signed key. The Qlik License Service is installed on the machine running the QlikView Management Service (QMS), and handles certificates differently from the other services.

When the QlikView Management Service (QMS) is started for the first time, the Root and Server certificates are automatically exported and made available to the Qlik License Service. The certificates are exported as the following file:

- root.pem
- server.pem
- server_key.pem
  This file contains the Server certificate key.

By default, these files are stored in the following location:
%ProgramData%\QlikTech\LicenseService\Exported Certificates.

When you update the certificates for your installation, you must restart the QlikView Management Service (QMS) before the Qlik License Service. Starting the services in this order ensures that the correct set of certificates is exported and made available to the Qlik License Service. You can manage the status of the Qlik License Service by starting and stopping the Qlik Service Dispatcher.

Requirements

The following requirements must be fulfilled for the certificate trust to function properly:

- Certificate trust cannot be partially implemented. It is either used by all services in the QlikView installation or not at all.
- Certificate trust is only supported by Windows Server 2008 and later.
- Make sure that all machines use QlikView Server 12.00 or later. In QlikView Server 11.20 or earlier, a different method of encryption is used. Old certificates are not compatible with an installation running QlikView 12.00 or later and new certificates need to be generated.
- If it is an initial installation of QlikView Server, install and configure the QlikView services without any modification. Prior to configuring the use of certificates, start and stop the services on the servers (that is, machines) where the QlikView services are deployed.
- Section Access management must not be configured in environments where certificate trust is configured.
- Ensure that you back up the following three certificates on the machine running the QlikView Management Service (QMS) every time they are updated:

  Certificates

Location	Issued To	Issued By	Description
Local Computer / Personal	*<machine-name>*	QlikViewCA	Server
Local Computer / Personal	QVProxy	QlikViewCA	Client
Local Computer / Trusted Root Certification Authorities	QlikViewCA	QlikViewCA	Root

  For more information on how to backup certificates, see: *Backing up and restoring certificates (page 155)*.

In addition, the technical requirements described in the following sections also have to be fulfilled.

Certificate ports

This section describes the ports that you need to open when configuring certificate trust.

The ports that are listed in the following table are needed for service to service communication and have to be configured as “open”.

For more information on QlikView ports, see: *Ports (page 19)*.

Firewall configuration changes might be necessary, depending on the location of the QlikView servers within the resulting network and the routing of the QVS communication.

Ports for service to service communication

Service	Ports	SSL/TSL -enabled Ports
QlikView Server	4747, 4749	4749
QlikView Distribution Service	4720	4720
QlikView Web Server	4750, 80, 443	4750, 443
QlikView Management Service	4780, 4799	4780, 4799
Directory Service Connector	4730	4730

The ports that are listed in the following table are needed for the certificate installation procedure on the local server.

The ports are not used for service to service communication.

Ports for certificate installation

Service	Ports
QlikView Distribution Service	14720
Directory Service Connector	14730
QlikView Web Server	14750

The following table lists the protocols that are used for communication on the ports that are specified in this section.

Protocols for port communication

Service	Ports
QlikView Server	QVPX over SSL/TSL
All other services	SOAP over SSL/TSL

To install the distributed certificates for the respective services, physical access to the console or remote access to the console (for example, using remote desktop functionality) is needed.

6.2 Protection of the Platform

Functionality

The functionality for downloading documents and/or print and export to Microsoft Excel can be restricted at the user level for each document on the server.

Special Accounts

Supervision Account

The supervision account is granted access to all documents that are created by tasks in QlikView Publisher. The characteristics of the supervision account are as follows:

- Provides access to all files on the QVS
- Does not provide any access to the QlikView Management Console (QMC)
- Respects the types of clients that are allowed for each document (for example, a supervision account cannot open a QlikView document using the AJAX client, if the AJAX client has been blocked by the user that created the task)

Anonymous User Account

When QVS is started for the first time on a machine, a Windows account is created for anonymous users. The account name is `IQVS_name`, where `name` is the name of the machine in the local network.

If the machine in question is a domain server, the anonymous account is created as a domain account. If not, it is created as a local machine account.

Each folder and file that is to be available for anonymous clients must be given read privileges for the anonymous account.

> *Start QVS and let it create the anonymous account before attempting to grant any privileges. Do not try to create the anonymous account manually.*

QlikView Administrators

The QlikView Administrators group is used for granting access to the QlikView Management Console (QMC) as well as authorization of communication between services, if Windows Authentication is used.

Communication

Protection of AJAX Client

The AJAX client uses HTTP or HTTPS as the protocol for communication between the client browser and the QlikView Web Server (QVWS) or Microsoft IIS. It is strongly recommended to protect the communication between the browser and the web server using SSL/TSL encryption over the HTTP protocol (that is, HTTPS). If the communication is not encrypted, it is sent as clear text.

The communication between the web server and QVS uses QVP as described below.

Protection of Plugin

The QlikView plugin can communicate with QVS in two ways:

- If the plugin has the ability to communicate with QVS using QVP (port 4747), the security is applied as follows:
  Server Communication (page 167)
- If the communication cannot use QVP or if the client chooses it in the plugin, the communication is tunneled using HTTP to the web server.

If HTTPS is enabled on the web server, the tunnel is encrypted using SSL/TLS.

Server Communication

The QVS communication uses the QVP protocol, which is encrypted by default. The QVP protocol can be protected using 1024-bit RSA for key exchange and 256-bit AES GCM for data encryption, provided the Microsoft Enhanced Cryptographic Provider is installed. If the Microsoft Base Cryptographic Provider is used, the protection of the communication is 512-bit RSA for key exchange and 128-bit AES CBC for data encryption.

Services Communication

The services that are part of the QlikView platform (that is, QVS, DSC, QMC, QDS, and QVWS) all communicate using web services. The web services authenticate using Integrated Windows Authentication (IWA).

SSL and TLS support

The following table shows QlikView support for SSL and TLS.

SSL vs TLS support

-	SSL v3.0	TLS v1.3	TLS v1.2	TLS v1.1	TLS v1.0
QlikView May 2023	√	√	√	-	-
QlikView May 2022	√	-	√	√	√

6.3 Authentication

QlikView requires that the user is authenticated when establishing a session via QlikView Server (either through a browser or when downloading and opening a document via the QlikView Desktop client). Although the majority of implementations require users to be authenticated, QlikView can also be configured to allow anonymous access

In the QlikView context, the authentication of a user is almost always done against an external entity that is then used to pass the externally authenticated user identity to QlikView Server. In such a scenario, QlikView relies on the authentication to be performed prior to accessing QlikView, and that some token of identity is transmitted to, and trusted by, QlikView.

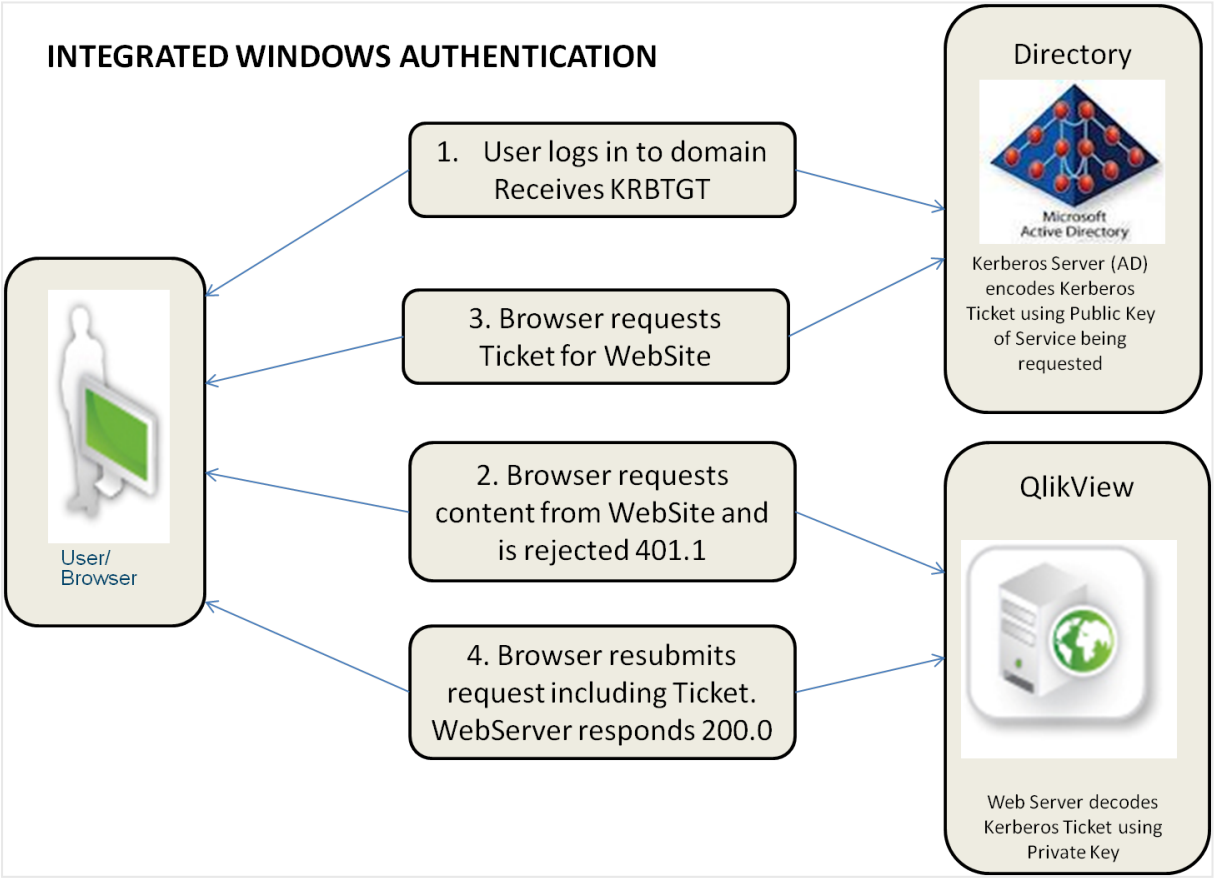
Authentication when Using QlikView Server in a Windows User Environment

Authentication to a QlikView Server in an environment based on Windows users (for example, incorporating Active Directory) is straightforward. The process is as follows:

1. The user credentials are validated when the user logs in to the Windows operating system on the client machine.
2. Later when the user wants to establish a session with a QlikView Server (QVS) (for example, via a browser on the desktop), QVS can use the built-in Integrated Windows Authentication (IWA).
3. The identity of the logged-in user is communicated to QlikView Server using either the Kerberos or the NTLM security solution. This solution provides single sign-on capabilities right out of the box. In case the authentication exchange fails to identify the user, the browser prompts the user for a Windows user account name and password.

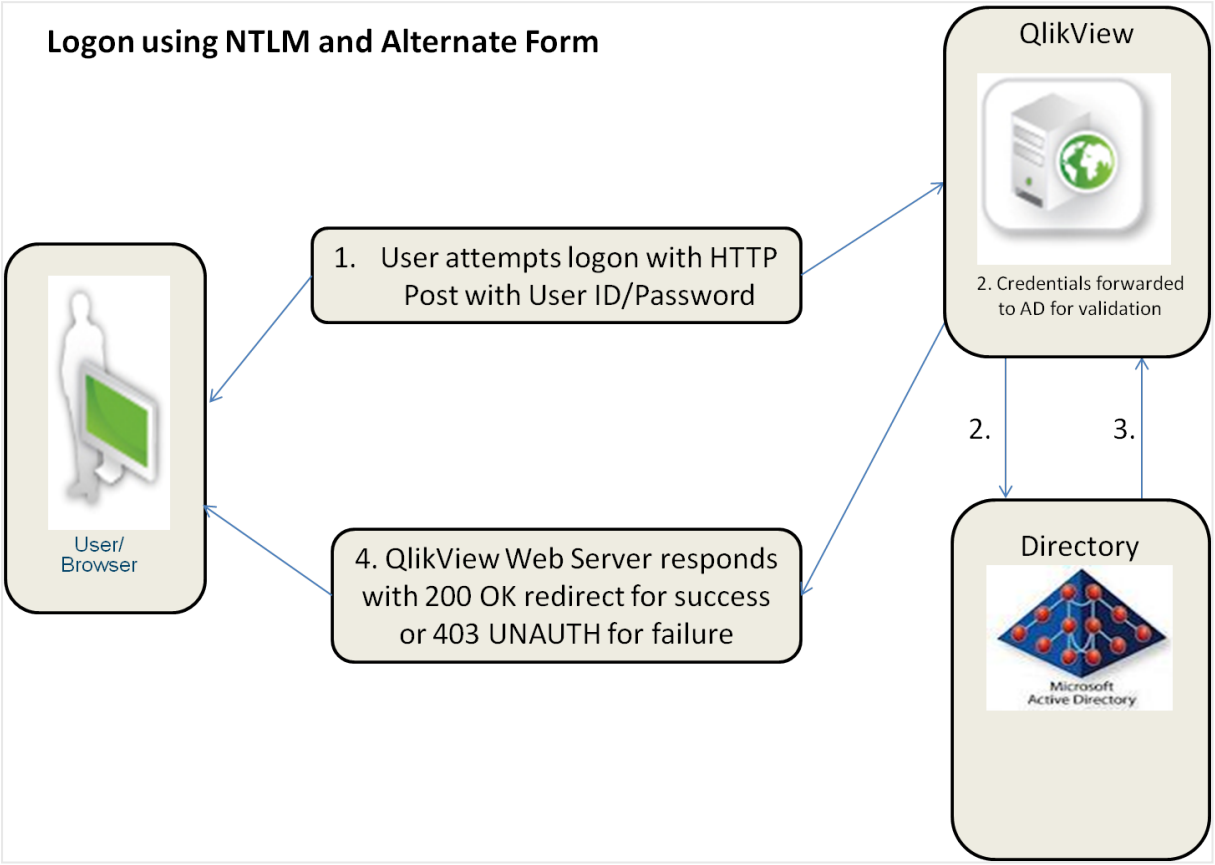
> *User groups cannot be transferred to the QlikView system. They have to be resolved by the Directory Service Connector (DSC) or not at all.*

The figure below shows the standard authentication flow for IWA:



Authentication when using QlikView Server in a Windows user environment

The figure below shows the authentication flow for the combination of NTLM and alternate login, which differs from the standard flow for IWA:



Authentication using NTLM and alternate form

The authentication process differs based on the environment:

- Local Area Network (LAN): IWA is most common and most suitable for recognizing Windows users on a LAN. The act of authentication is performed when logging in the workstation, and this identity is leveraged by QlikView.
- Multi-domain environment: The internal company network IWA should be avoided in architectures where there is a multi-domain environment with no trust relationship between the domain of the workstation and the domain of the server, or when used across a reverse proxy. In such an environment, configure the QlikView deployment to use either an existing external SSO service or a QlikView custom ticket exchange to expose an authenticated identity to QlikView.

Authentication with a QlikView Server Using an Existing Single Sign-on Software Package

In environments where an SSO infrastructure already exists (for example, CA SiteMinder®, IBM® WebSeal, or Oracle® Oblix), QlikView can use the HTTP header injection method of single sign-on provided by the SSO infrastructure. This means single sign-on is provided right out of the box. The SSO infrastructure software packages can be configured as follows:

- Repeat user get access: The software packages can be configured to protect a resource. When a user requests access to QlikView, the SSO package grants access, if the user has previously signed in to the SSO authentication page.
- New user log in: If the user does not have an existing session with the SSO package, the user is redirected to the SSO package login page. After logging in, the user is redirected to the original URL that the user requested.

In both cases, if the user has properly authenticated to the SSO software, the username is injected into an HTTP header and the value in that header is what the QlikView server accepts as the authenticated identity of the user.

> *Unless SSO software is in place, the HTTP header method of authenticating to a QlikView Server must not be used. HTTP headers can easily be spoofed. All of the SSO software packages mentioned above provide protection against this type of spoofing attacks, if the software package is the only path for users to access the content.*

QlikView does not recommend or endorse any specific tool or product for providing identity in HTTP headers. The approach is highly suited to extranet deployments wherein the users may not exist in the internal Active Directory. The act of authentication is performed by the reverse proxy or ISAPI filter that intercepts the attempt of the end user to interact with QlikView content.

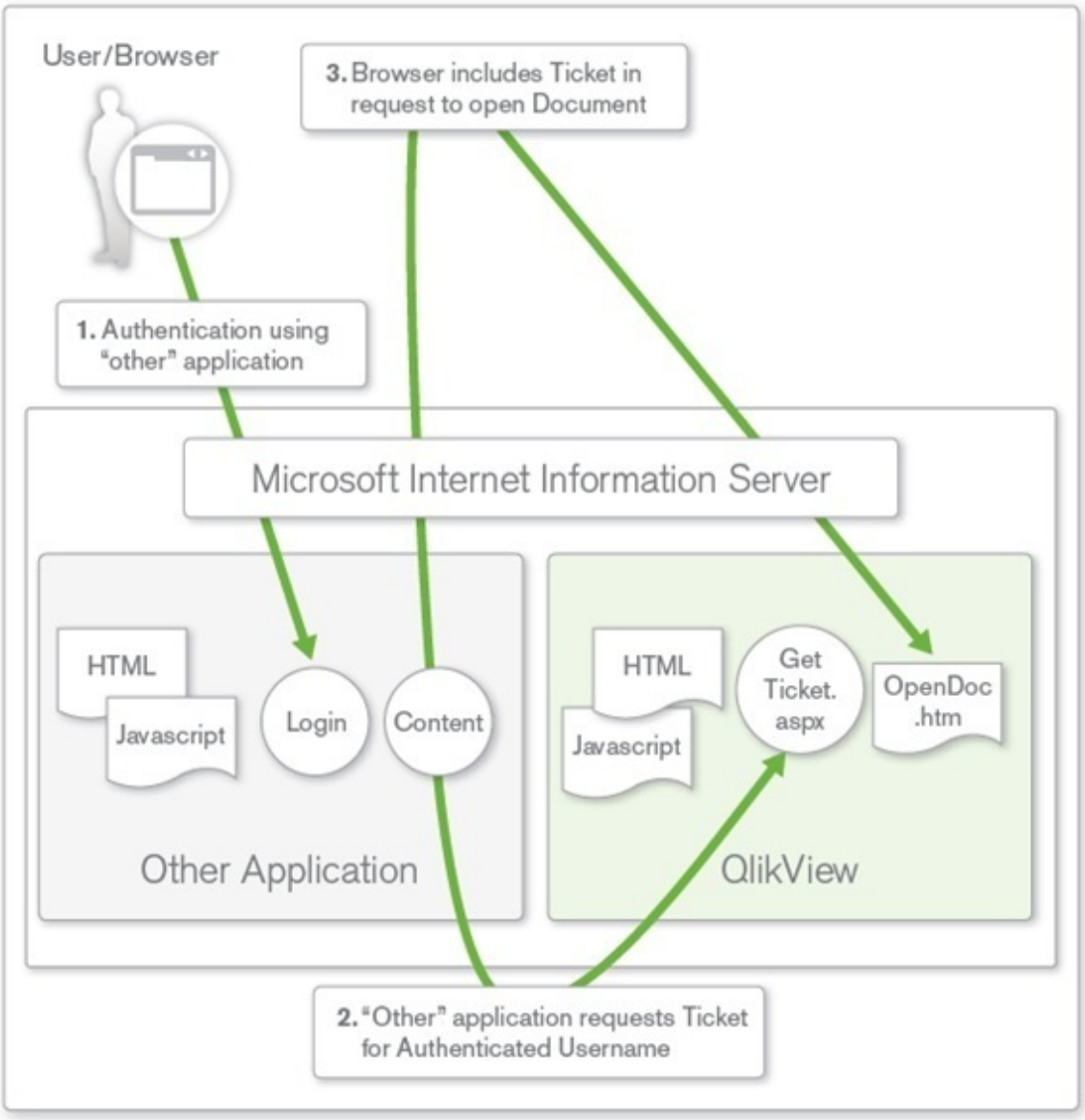
Authentication Using neither IWA nor Single Sign-on Software

QlikView provides a third method for single sign-on, Custom Ticket Exchange (CTE), when neither of the methods described above is suitable.

CTE relies on the user having authenticated previously to another system:

1. The third-party system is granted the privilege and responsibility to request an authentication token (called a “ticket” in QlikView) from QVS on behalf of the authenticated user of the third-party system. It is the responsibility of the third-party system to only request tickets for users that have been properly authenticated (for example, QVS has no knowledge of the authentication status of the user).
2. The system then passes the authentication token to the user, who uses it in a request to open a session with QVS.
3. QVS checks that the ticket is valid and then opens a session for the authenticated user.

Ticketed authentication is mainly applicable when embedding QlikView content in third-party applications and portals, and is rarely used for providing general access to QlikView. Typically a small amount of custom development is needed to implement the request and passing of the ticket for the CTE method to work.

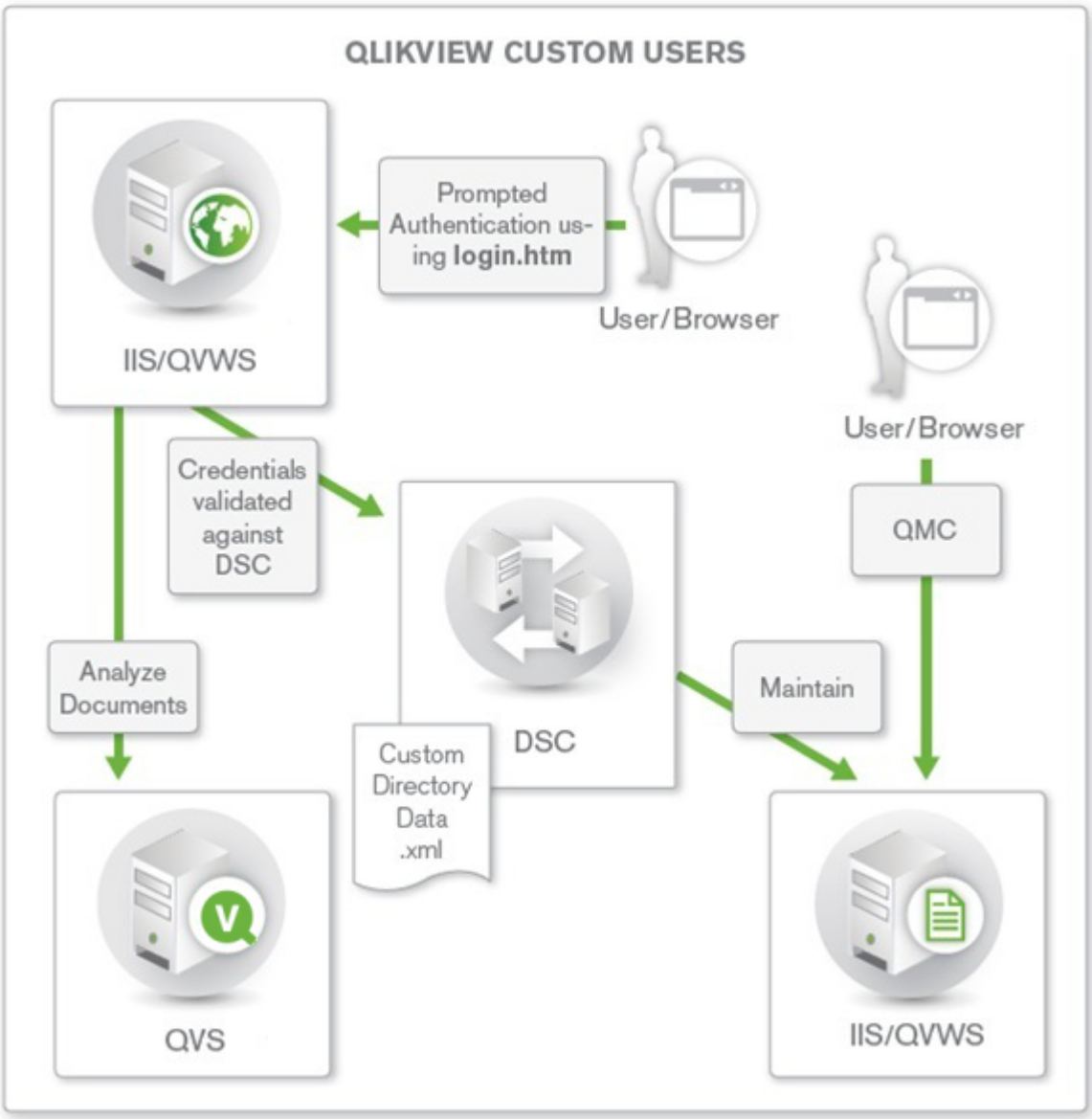


Authentication using neither IWA nor single sign-on software

QlikView Server Authentication Using Custom Users

The three methods described above all use a single sign-on principle, where the user ID and password are stored externally to QlikView Server and an external entity is responsible for the authentication. Less common, although possible, is the ability to store the user credentials in the QlikView Server environment using the Custom Users functionality in QlikView Publisher. In this case, users and passwords are defined and stored within the QlikView environment and the web tier of the QlikView deployment is responsible for forms authentication. This solution is suitable for smaller, standalone QlikView Server deployments, and must not be used in environments where the user definitions are to be available to multiple systems. In such environments, it is highly recommended to use one of the three single sign-on solutions described above.

Each coexistent form of authentication may require a distinct web server instance. Several web servers can forward user requests to the same QVS instance(s).



QlikView Server authentication using Custom Users

6.4 Authorization

Once a user has been authenticated (that is, the system knows who the user is), the first step in assigning the security privileges has been completed. The second step is to understand the authority or access rights that the user has to applications, data, or both. This step is referred to as Authorization. At a fundamental level, an administrator populates an Access Control List (ACL) with a list of users and/or groups and what they are to have access to. When the time comes for a user to request access, the system looks up the authenticated identity of the user in the ACL and verifies if the administrator has granted the user enough privileges to do so.

Direct access to a QlikView document using QlikView Desktop is always governed by the Windows NTFS file security. Access to the web-based QlikView Management Console (QMC) is restricted to Windows users that are members of a particular local Windows group.

Document Level Authorization

Once a user has been authenticated, QlikView Server typically handles authorization on its own. QlikView Server provides the choice between storing the ACL information as Windows NTFS privileges (applicable only when the user is authenticated using a Windows user identity) or by storing the ACL information in the internal repository, Document Metadata Service (DMS), in QlikView. The choice of NTFS or DMS affects the access to all documents in QlikView Server.

NTFS vs. DMS

QlikView Server can use the NTFS privileges of the Windows file system to store authorization information. When in NTFS authorization mode, QlikView Server controls access to a given QlikView document by determining if the authenticated user has NTFS privileges to the underlying QlikView document file (*.qvw* or *.qvf*). This is based on the operating system privileges and Windows NTFS is used for the ACL. The privileges of the authenticated user are configured by a server administrator using standard Windows Explorer functionality via directory properties options.

As an alternative to Windows NTFS, QlikView can use its own ACL, DMS. Unlike NTFS, this allows non-Windows users and groups to be authorized to access applications and data. DMS integrates fully with the existing Directory Service Provider (for example, Active Directory, other LDAP) where Group Membership has been recorded – this is a mechanism by which QlikView Server can re-use existing enterprise accounts and group structures. The permitted users or groups are recorded in a meta file that resides next to the QlikView document, and it is managed using QMC.

NTFS is the default document authorization model, suitable when all users and groups are identified in Active Directory or locally on the QlikView Server host. The NTFS permissions may be inherited from the directory that the QlikView documents are in, or may be assigned using QlikView Publisher distribution tasks.

DMS is required when the authenticated user identity is not a Windows user account. The DMS permissions are explicitly assigned using QMC, or may be assigned using QlikView Publisher distribution tasks.

When authenticating a user via a web ticket, the user is not a proper Windows user, even if sending in the user name in Active Directory format (for example, QLIKVIEW\jsmith). This means that DMS authorization should be used when using web tickets.

Data Level Authorization

Data level authorization allows access to be granted or denied on a document level and even to specific data in a document.

There are two types of data level authorizations:

- Dynamic data reduction: Determines if the user is allowed to view the data when the user tries to access it.
- Static data reduction: Performed by QlikView Publisher, determines if the user is allowed to view the data when it is prepared for the user.

Static and dynamic reduction of data can be used on its own, but can also be combined to deliver data level authorization.

Dynamic Data Reduction

Dynamic data reduction is done in QlikView using the concept of Section Access, which is part of the QlikView document.

Section Access Management is configured in the QlikView Management Console (QMC). For information, see the QMC help.

Static Data Reduction

For larger deployments and/or those in need of centralized control of authorization capabilities, QlikView Server/Publisher are used. Departments or functions often have a “master” application that contains all relevant data covering all analysis needs, and this master document needs to be separated (“reduced”) according to the needs and access privileges of the intended audience. QlikView Publisher reloads the QlikView document with available data, refreshes the Section Access tables, and splits the large QlikView document into smaller documents based on values in a particular field.

This “reduction and distribution” allows for a file containing many data fields to be broken up by the contents of a field and distributed to authorized users or groups according to their access privileges.

One of the benefits of reducing and distributing source files in this manner is that the documents that are created in this process contain no explicit reference to the source data (for example, a database connection string) in their script environments. Therefore, if a user interacts with the document via QlikView Desktop, the user cannot see the location of the source data. All of the data pertinent to the user needs is contained in the document.

An administrator can use QMC to create tasks on source *.qvw*, *.qvf*or *.qvd* files to accomplish this. At a basic level, the steps are as follows:

An administrator can use QMC to create tasks on source *.qvw*, *.qvf*or *.qvd* files to accomplish this. At a basic level, the steps are as follows:

1. On the source document (either *.qvw*, *.qvf* or *.qvd*), apply the data reduction criteria (for example, choose the field name on which to reduce the data).
2. Apply the distribution criteria to the newly created (reduced) files:
    a. Assign the authorization privileges using either DMS or NTFS ACLs.
    b. Choose the type of distribution (for example, *.qvw* or *.qvf* files or *.pdf* report).
    c. Choose the location for the newly created files.
3. Apply the notification criteria for the completion of the task (for example, e-mail notification).

The newly created files only contain the data that the user or group is authorized to see, since the data has been “reduced” from the master document in accordance to the reduction criteria. This is why the process is termed “Static Data Reduction”. Hence, there is no risk of an unauthorized person viewing data, since only authorized data exists in each file.

6.5 QVD Encryption

You can encrypt sensitive data in QVD files with customer supplied key pairs which allows you to control who gets access to your data. The encryption keys are managed through certificates, that must be stored in a certificate store for the user running the QlikView Distribution Service (QDS).

The encryption is configured in the *settings.ini* file where encryption is enabled and the certificate thumbprint is added. QVD encryption is not enabled by default.

The engine reads and then uses the thumbprint to get the key from the Windows CNG key store. The engine then generates a new data encryption key (DEK) which is used to encrypt the data.

> *A DEK is never reused which ensures that if one file is compromised, the encryption is still valid for all other files.*

The following is encrypted:

- Data (tables and fields)

The QVD header is not encrypted. Encryption parameters are stored in the QVD header as extra meta-data.

> *You must reload an existing QVD for it to be encrypted after QVD encryption has been enabled in the settings.ini file.*

Older versions of Qlik Sense and QlikView returns an error when reading encrypted QVDs files.

Encryption certificates overview

Encryption keys are best managed through certificates. The certificates must be stored in a certificate store for the user running the QlikView Distribution Service (QDS).

The encryption certificate functions as a shell around the encryption key. The key can be fetched even if the certificate has expired, and therefore there is no need to renew an expired encryption certificate.

Encryption keys

The encryption solution uses two types of keys:

- Data encryption keys
- Key encryption keys

Data encryption keys

Data encryption keys (DEK) are auto-generated keys for AES-256 encryption of the data. A new key is generated for each object that is encrypted.

Key encryption keys

Key encryption keys (KEK) are private and public key pair for secure, asymmetric encryption of the data encryption keys. The public key is used to encrypt the data and the private key is used to decrypt the data encrypted by the public key.

> *Only keys using the RSA algorithm are supported.*

The key used for key encryption is specified in the *settings.ini* file. It is stored in a Microsoft Cryptography Next Generation (CNG) Key Storage Provider. It is contained in a certificate stored in a Windows Certificate Store.

Using QVD encryption

This is the common workflow for using the QVD encryption feature in QlikView.

1. Create an encryption certificate: *Creating encryption certificates using Windows PowerShell (page 178)*
2. Enable QVD encryption and specify the key: *Enabling QVD encryption (page 176)*
3. For multi-node deployments, export the encryption certificate: *Exporting encryption certificates using Windows PowerShell (page 180)*
4. For multi-node deployments, import the encryption certificate on all nodes: *Importing encryption certificates using Windows PowerShell (page 183)*

> *Make sure to back up the certificate. You may not be able to open your encrypted QVD if the certificate is lost. It is your responsibility to safe keep the certificate backup for as long as it is needed.*

Encrypting QVD files shared with Qlik Sense

If you have QVD files used in both QlikView and Qlik Sense Enterprise on Windows, make sure that the same thumbprint is defined for both products.

Enabling QVD encryption

The Qlik associative engine is configured by defining the encryption key thumbprint in the *settings.ini* file. Enable QVD encryption by defining `enableEncryptQvd=1`. Then copy the value of the *Thumbprint* field from the certificate and paste it into the `encryptionKeyThumbprint` field in *settings.ini*.

> *The certificate must be stored in a certificate store for the user running the QlikView Distribution Service (QDS).*

Do the following:

1. Open the Certificate Manager tool (certmgr.msc).
2. Locate the certificate.
3. Right click the certificate and select **Open**.
4. On the **Details** tab, select the **Thumbprint** field and copy the value.
5. Locate the appropriate *settings.ini* file.
   For QlikView Desktop, the *settings.ini* file can be found in *C:\Users\<user>\AppData\Roaming\QlikTech\QlikView*.
   For QlikView Server, the *settings.ini* file can be found in *C:\Windows\System32\config\systemprofile\AppData\Roaming\QlikTech\QlikViewBatch*.
6. In the *settings.ini* file, enable QVD encryption: `enableEncryptQvd=1`. Then paste the Thumbprint value into the `encryptionKeyThumbprint` field.

   Example:

   ```
   enableEncryptQvd=1
   encryptionKeyThumbprint=563888bb6aea55eb0d33d9d8b909e0d2ef26ffbd
   ```
7. Save the *settings.ini* file.

QlikView accepts Secure Hash Algorithm 1 (SHA-1) thumbprints in the 40-digit hexadecimal string form without spaces.

Example:

If your certificate thumbprint contain spaces, like **`56 38 88 bb 6a ea 55 eb 0d 33 d9 d8 b9 09 e0 d2 ef 26 ff bd`**, you enter it in the `encryptionKeyThumbprint` field as follows:

```
encryptionKeyThumbprint=563888bb6aea55eb0d33d9d8b909e0d2ef26ffbd
```

> *If your organization has a key rotation policy, you may need to update the thumbprint definition when the key is changed.*
> *Remember to keep the certificate containing the old key on the server until QVDs have been saved with the new key.*

Managing encryption certificates

There are many tools available for managing certificates but this documentation will focus on creating and distributing certificates using Windows PowerShell and Microsoft Management Console.

If other tools are used, the requirements are:

- a RSA key is used
- the key is stored in a CNG KeyStorageProvider
- the certificate is stored in a certificate store for the user running the Engine

Make sure to back up the certificate. You may not be able to open your encrypted QVD if the certificate is lost. It is your responsibility to safe keep the certificate backup for as long as it is needed.

The encryption certificate should be exported to every node in the deployment.

Creating encryption certificates using Windows PowerShell

It is not necessary to use certificates issued by a certificate authority (CA), you can also issue and sign your own self-signed certificates. Encryption certificates that you create must be stored in a certificate store for the user running the QlikView Distribution Service (QDS).

To create the new encryption certificate, use the **`New-SelfSignedCertificate`** cmdlet to create a self-signed certificate.

Syntax: Windows Server 2016 and later

```
PS C:\Users\johndoe.ACME> New-SelfSignedCertificate -Subject <Certifcate name> -KeyAlgorithm
RSA -KeyLength <Key length, e.g.4096> -Provider "Microsoft Software Key Storage Provider" -
KeyExportPolicy ExportableEncrypted
-CertStoreLocation "cert:\CurrentUser\My"
```

Syntax: Windows Server 2012 R2

```
PS C:\Users\johndoe.ACME> New-SelfSignedCertificate -DnsName <Certifcate name> -
CertStoreLocation "cert:\CurrentUser\My"
```

New-SelfSignedCertificate cmdlet parameters Windows Server 2016 and later

The following parameters should at minimal be defined when creating the certificate using PowerShell for Windows Server 2016 and later.

For complete documentation, see the ≤ Microsoft New-SelfSignedCertificate documentation.

-Subject

Specifies the string that appears in the subject of the new certificate. This cmdlet prefixes **`CN=`** to any value that does not contain an equal sign. For multiple subject relative distinguished names (also known as RDNs), separate each subject relative distinguished name with a comma (,). If the value of the relative distinguished name contains commas, separate each subject relative distinguished name with a semicolon (;).

```
-Subject <Certifcate name>
```

-KeyAlgorithm

Specifies the name of the algorithm that creates the asymmetric keys that are associated with the new certificate. Must be **`RSA`**.

```
-KeyAlgorithm RSA
```

-KeyLength

Specifies the length, in bits, of the key that is associated with the new certificate.

```
-KeyLength <Key length, e.g.4096>
```

-Provider

Specifies the name of the KSP or CSP that this cmdlet uses to create the certificate. Should be **`Microsoft Software Key Storage Provider`**.

```
-Provider "Microsoft Software Key Storage Provider"
```

-KeyExportPolicy

Specifies the policy that governs the export of the private key that is associated with the certificate. The acceptable values for this parameter are:

- Exportable
- ExportableEncrypted (default)
- NonExportable

```
-KeyExportPolicy ExportableEncrypted
```

-CertStoreLocation

Specifies the certificate store in which to store the new certificate. If the current path is *Cert:\CurrentUser* or *Cert:\CurrentUser\My*, the default store is **`Cert:\CurrentUser\My`**. Otherwise, you must specify **`Cert:\CurrentUser\My`** for this parameter.

```
-CertStoreLocation "cert:\CurrentUser\My"
```

New-SelfSignedCertificate cmdlet parameters Windows Server 2012 R2

The following parameters should at minimal be defined when creating the certificate using PowerShell for Windows Server 2012 R2.

> *For complete documentation, see the* ≤ Microsoft New-SelfSignedCertificate documentation.

-DnsName

Specifies one or more strings to put into the Subject Alternative Name extension of the certificate. The first DNS name is also saved as Subject Name and Issuer Name.

```
-DnsName <Certifcate name>
```

-CertStoreLocation

Specifies the certificate store in which to store the new certificate. If the current path is *Cert:\CurrentUser* or *Cert:\CurrentUser\My*, the default store is **`Cert:\CurrentUser\My`**. Otherwise, you must specify **`Cert:\CurrentUser\My`** for this parameter.

```
-CertStoreLocation "cert:\CurrentUser\My"
```

New-SelfSignedCertificate defaults Windows Server 2012 R2

The following defaults apply for the **`New-SelfSignedCertificate`** cmdlet in Windows Server 2012 R2:

- Key algorithm: RSA
- Key length: 2048
- Extended key usage (EKU): Client authentication and Server authentication
- Key usage: Digital signature, Key encipherment (a0)
- Validity: one year

Example: creating a data encryption certificate using PowerShell for Windows Server 2016 and later

In this example, the user called test is creating a self-signed exportable encrypted certificate with the subject `MyTestCert` and a key length of `4096` bits. The certificate is to be stored in `Cert:\CurrentUser\My`.

Type the following command in Microsoft PowerShell:

```
PS C:\Users\test> New-SelfSignedCertificate -Subject MyTestCert -KeyAlgorithm RSA -KeyLength
4096 -Provider "Microsoft Software Key Storage Provider" -KeyExportPolicy ExportableEncrypted
-CertStoreLocation "cert:\CurrentUser\My"
```

Result:

When the certificate has been created, the following is displayed in Microsoft PowerShell:

```
   PSParentPath: Microsoft.PowerShell.Security\Certificate::CurrentUser\My

Thumbprint                                Subject
----------                                -------
563888BB6AEA55EB0D33D9D8B909E0D2EF26FFBD  CN=MyTestCert
```

Exporting encryption certificates using Windows PowerShell

To export a encryption certificate, use the **`Export-PfxCertificate`** cmdlet.

Syntax:

```
PS C:\Users\johndoe.ACME> Export-PfxCertificate -cert cert:\currentuser\My\<certificate
thumbprint> -FilePath <FileName>.pfx -Password <Password or variable>
```

Export-PfxCertificate cmdlet parameters

The following parameters should at minimal be defined when exporting the certificate.

> *For complete documentation, see the* ≤ Microsoft Export-PfxCertificate documentation.

-cert

Specifies the path to the certificate to be exported.

```
-cert cert:\currentuser\My\<certificate thumbprint>
```

-FilePath

Specifies the path for the PFX file to be exported.

```
-FilePath <FileName>.pfx
```

-Password

Specifies the password used to protect the exported PFX file. The password should be in the form of secure string. This parameter must be specified, or an error will be displayed.

```
-Password <Password or variable>
```

Example: exporting a data encryption certificate

In this example the user called test will export the encryption certificate previously created to a PFX file.

1. First, create a secure string of the plain text password string and store it in the `$mypwd` variable. For this he is using the **`ConvertTo-SecureString`** cmdlet.
   Type the following command in Microsoft PowerShell:
   ```
   PS C:\Users\test> $mypwd = ConvertTo-SecureString -String "MyPassword" -Force -
   AsPlainText
   ```
2. Then proceed with the actual exporting of the encryption certificate with thumbprint `563888bb6aea55eb0d33d9d8b909e0d2ef26ffbd` using the **`Export-PfxCertificate`** cmdlet. The password variable created in the previous step is called to protect the exported PFX file. Type the following command in Microsoft PowerShell:
   ```
   PS C:\Users\test> Export-PfxCertificate -cert
   cert:\currentuser\My\563888bb6aea55eb0d33d9d8b909e0d2ef26ffbd -Filepath MyTestCert.pfx -
   Password $mypwd
   ```

Result:

When the certificate has been exported, the following is displayed in Microsoft PowerShell:

```
    Directory: C:\Users\test


Mode                LastWriteTime         Length Name
----                -------------         ------ ----
-a----       11/20/2019     11:21           4294 MyTestCert.pfx
```

Backing up encryption certificates using Microsoft Management Console

You should always have a back up of the certificate. If the certificate is lost from the server, or in case of a hard disk failure, you may not be able to open your encrypted app. It is your responsibility to keep safe the certificate backup for as long as it is needed.

You can use the same procedure as for exporting when backing up your certificate, see *Exporting encryption certificates using Windows PowerShell (page 180)*.

Another way of backing up your encryption certificates is to do it with Microsoft Management Console. The below example shows how to export or back up your SSL certificate with a private key using Microsoft Management Console.

Do the following:

1. On the Windows Server where the SSL certificate is installed, open the Microsoft Management Console: type `mmc` in the Windows search menu and open it.
2. In the Console window, click **File > Add/Remove Snap-in**.
3. In the Add or Remove Snap-ins window, select **Certificates** from the Available snap-ins pane on the left side and then click **Add >**.
4. In the dialog, select **My user account** and then click **Next**.
5. In the Add or Remove Snap-ins window, click **OK**.
6. In the Console window, in the Console Root pane on the left side, expand Certificates (Current user) and locate the certificate that you want to export or back up.
7. In the center pane, right-clock on the certificate that you want to export or back up, and then click **All Tasks > Export**.
8. In the Certificate Export Wizard, on the Welcome to the Certificate Export Wizard page, click **Next**.
9. On the Export Private Key page, select **Yes, export the private key**, and then click **Next**.
10. On the Export File Format page, select **Personal Information Exchange - PKCS #12 (.PFX)** and then check **Include all certificates in the certification path if possible**.

    *Do not select **Delete the private key if the export is successful**.*

    Click **Next**.
11. On the Security page, check the **Password** box, then create and confirm the password.

    This password will be required when you import or restore the certificate with private key.

    Then check the **Group or user name** box. If applicable, select the Active Directory user or group account to which you want to assign access to the certificate with private key. Then click **Add**.
    Click **Next**.
12. On the File to Export page, click **Browse** to specify the save location and the file name of the back up file and then click **Save**.
    Back on the File to Export page, click **Next**.
13. On the Completing the Certificate Export Wizard page, verify that the settings are correct and then click **Finish**.
14. You should receive a message stating that the export was successful, and the SSL certificate with private key is now saved to the location that you selected .

Importing encryption certificates using Windows PowerShell

To import an encryption certificate on for example other machines, use the **`Import-PfxCertificate`** cmdlet.

> *Encryption certificates that you import must be stored in a certificate store for the user running the QlikView Distribution Service (QDS).*

Syntax:

```
PS C:\Users\johndoe.ACME> Import-PfxCertificate -CertStoreLocation cert:\currentuser\My -
FilePath <FileName>.pfx [-Exportable] -Password $mypwd
```

Import-PfxCertificate cmdlet parameters

The following parameters should at minimal be defined when importing the certificate.

> *For complete documentation, see the* ≤ Microsoft Import-PfxCertificate documentation.

-CertStoreLocation

Specifies the path of the store to which certificates will be imported. If this parameter is not specified, then the current path is used as the destination store.

```
-CertStoreLocation cert:\currentuser\My
```

-FilePath

Specifies the path for the PFX file.

```
-FilePath <FileName>.pfx
```

-Exportable

Optional.

Specifies whether the imported private key can be exported. If this parameter is not specified, then the private key cannot be exported.

```
-Exportable
```

-Password

Specifies the password for the imported PFX file in the form of a secure string.

```
-Password $mypwd
```

Example: importing a data encryption certificate

In this example the user called test2 will import the encryption certificate with thumbprint `563888BB6AEA55EB0D33D9D8B909E0D2EF26FFBD` previously exported to a PFX file.

1. First, create a secure string of the plain text password string and store it in the `$mypwd` variable. For this he is using the **`ConvertTo-SecureString`** cmdlet.
   Type the following command in Microsoft PowerShell:
   ```
   PS C:\Users\test2>  $mypwd = ConvertTo-SecureString -String "MyPassword" -Force -
   AsPlainText
   ```
2. Then proceed with the actual importing of the PFX file using the **`Import-PfxCertificate`** cmdlet. The password variable created in the previous step is called to access the PFX file. Type the following commands in Microsoft PowerShell:
   ```
   PS C:\Users\test2>  Import-PfxCertificate -CertStoreLocation cert:\currentuser\My -
   FilePath MyTestCert.pfx -Exportable -Password $mypwd
   ```

Result:

When the certificate has been exported, the following is displayed in Microsoft PowerShell:

```
   PSParentPath: Microsoft.PowerShell.Security\Certificate::CurrentUser\My

Thumbprint                                Subject
----------                                -------
563888BB6AEA55EB0D33D9D8B909E0D2EF26FFBD  CN=MyTestCert
```

Restoring encryption certificates using Microsoft Management Console

You can use the same procedure as for importing when restoring your certificate, see *Importing encryption certificates using Windows PowerShell (page 183)*.

If you backed up your certificate using Microsoft Management Console, as described in *Backing up encryption certificates using Microsoft Management Console (page 181)*, then follow the example below to restore your SSL certificate.

> *Encryption certificates that you restore must be stored in a certificate store for the user running the QlikView Distribution Service (QDS).*

Do the following:

1. On the Windows Server where you want to install the SSL certificate, open the Microsoft Management Console: type `mmc` in the Windows search menu and open it.
2. In the Console window, click **File > Add/Remove Snap-in**.
3. In the Add or Remove Snap-ins window, select **Certificates** from the Available snap-ins pane on the left side and then click **Add >**.
4. In the dialog, select **My user account** and then click **Next**.
5. In the Add or Remove Snap-ins window, click **OK**.
6. In the Console window, in the Console Root pane on the left side, expand Certificates (Current user), right-click on the Personal folder, and then select **All Tasks > Import**.
7. In the Welcome to the Certificate Import Wizard window, click **Next**.

8. On the File to import page, Click **Browse** to locate and select the PFX file that you want to import, and then click **Next**.

   Make sure to select All files (.*) in the file type drop-down of the File Explorer window, as it by default is set to search for X.509 Certificate (*.cert,*.crt) file types only.*

9. On the Private key protection page, type the password that was created when the SSL certificate was exported / backed up.
   Then check the **Mark this key as exportable** box. This means you can back up or export the SSL certificate when needed.
   Then also check the **Include all extended properties** box.
   Click **Next**.
10. On the Certificate Store page, select **Place all certificates in the following store** and then click **Browse**.
    In the Select Certificate Store window, select **Personal** and click **OK**.
    Back on the Certificate Store page, click **Next**.
11. Verify that all settings are correct on the Completing the Certificate Import Wizard page, and then click **Finish**.
12. You should receive a message stating that the import was successful, and the SSL certificate with private key is now saved to the Personal store (folder).

7 Licensing QlikView

Licenses let you manage QlikView software usage in your organization.

7.1 Overview

QlikView Server deployments are licensed via a serial and control number, or a signed key. Your QlikView Server license is based either on access types, or on CALs (Client Access License). A QlikView Server installation can also include a QlikView Publisher license.

For detailed information on QlikView licensing options, read Qlik's legal agreements, product terms, and Licensing Service Reference Guide:
≤ Qlik Legal Agreements
≤ Qlik Product Terms
≤ Qlik Licensing Service Reference Guide

7.2 Unified license

As of the April 2019 releases of QlikView and Qlik Sense, QlikView customers can use a unified license in multiple deployments. A unified license shares the same signed key between:

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

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

For detailed instructions on how to apply a signed key to your QlikView Server deployment and configure Professional and Analyzer access, see: *Configure Professional and Analyzer access in QlikView Server (page 189)*.

7.3 QlikView Server license

QlikView Server license terms and access allocations are defined by the License Enabler File (LEF), or by the License Definition, depending on the license activation method. A QlikView Server license is either based on access types, or on CALs (Client Access License).

User-based and capacity-based licenses

A user-based license grants a predefined number of access allocations that can be assigned to unique and identified users. In QlikView Server, user-based licenses are either Professional and Analyzer Users licenses, or Client Access Licenses (CALs).

A capacity-based license grants a predefined number of time allocations for accessing QlikView that can be used by identified or anonymous users. In QlikView Server, capacity-based licenses are either based on Analyzer Capacity access, or on CALs.

Access types

Professional and Analyzer Users licenses (user-based) and Analyzer Capacity licenses (capacity-based) can be combined. These licenses are subscription based, and activated using a signed key. You can see the details of your license in the License Definition, located in the QlikView Management Console. See: QlikView Server License.

- Professional and Analyzer access (user-based) are allocated to users just as in Qlik Sense. The License Definition determines the distribution of the two access types.
- Analyzer Capacity (capacity-based) is similar to Analyzer access regarding available features. Analyzer Capacity is available to identified or anonymous users. Users share the monthly analyzer time allotment, which is consumed in units of six minutes.

CALs

User and Document CALs (user-based) and Session and Usage CALs (capacity-based) can be combined. CALs can be subscription based or perpetual. They are activated using a license key composed of a serial number and a control number.

- User and Document CALs (user-based): are allocated to unique and identified users for accessing QlikView documents. The LEF file determines the types and amount of CALs available for your installation.
- Session and Usage CALs (capacity-based): allow any user, identified or anonymous, to access and consume QlikView documents. The LEF file determines the types and amount of CALs available for your installation.

Restrictions

You cannot combine access types (Professional, Analyzer, and Analyzer Capacity) and CALs (User, Document, Session, and Usage). Your QlikView Server license can be composed of both user-based access and capacity-based access quotas. For example:

- If you purchase a license based on access types, your license can contain different quotas of Professional access, Analyzer access, and Analyzer Capacity access.
- If you purchase a license based on CALs, your license can contain different quotas of User, Document, Session, or Usage CALs.

Professional and Analyzer access dynamic assignment

QlikView Server supports dynamic assignment of Professional and Analyzer access types. You can enable dynamic assignment for Professional and Analyzer access in the QlikView Management Console, see Professional and Analyzer access.

How dynamic assignment works:

- When dynamic assignment is enabled for both Professional and Analyzer access types: a user logging in is automatically assigned Professional access, if available. If not available, the user is assigned Analyzer access. If Analyzer access is not available, the user is assigned Analyzer Capacity access. If Analyzer Capacity is not available, the user cannot access QlikView.

- When dynamic assignment is only enabled for Professional access type: a user logging in is automatically assigned Professional access, if available. If not available, the user is assigned Analyzer Capacity access. If Analyzer Capacity is not available, the user cannot access QlikView.
- When dynamic assignment is only enabled for Analyzer access type: a user logging in is automatically assigned Analyzer access, if available. If not available, the user is assigned Analyzer Capacity access. If Analyzer Capacity is not available, the user cannot access QlikView.

If you enable dynamic assignment, users may get a double allocation. This might impact the number of available licenses for your installation.

QlikView Server signed key

When you license QlikView with a signed key, your license information, such as license definition and access allocations, are stored in a License Back-end outside the QlikView deployment. Connected deployments use the same License Back-end. Users listed in the connected deployments are shared with the QlikView installation, together with their access allocations. This can affect the number of Professional and Analyzer access allocations available in the QlikView installation.

QlikView Server license key

A license key is composed of a serial number and a control number. It is used when activating a QlikView Server license based on CALs. CALs are only used for licensing, not with user authentication for data access purposes.

A QlikView Server license key can be installed on as many servers as needed, provided that only the licensed number of nodes are running at any given time.

A cold standby environment can be installed and ready-to-run, but cannot be live (Windows services cannot be started) and in use prior to the live environment being shut down.

To connect to a QlikView Server installation, each client needs a Client Access License (CAL). CALs are purchased with QlikView Server and associated with the server serial number. CALs cannot be transferred to a QlikView client program or between different QlikView Server clusters. A separate CAL is needed for each cluster.

7.4 QlikView Publisher license

A QlikView Publisher license adds more functionality to your QlikView Server installation, such as advanced reload capabilities and distribution models. A dedicated LEF file determines the capabilities of your QlikView Publisher license.
This license is perpetual, and is activated using a license key composed of a serial number and a control number.

7.5 QlikView Desktop

QlikView Desktop supports the following licensing options:

- Local Client License: a complete and licensed version of QlikView Desktop. The Local Client License is defined by a License Enabler File (LEF) and activated using a license key.
- Personal Edition. an unlicensed version of QlikView Desktop meant for individuals, students, or small start-ups. For more information on QlikViewPersonal Edition, see: QlikView Personal Edition.

QlikView Desktop can be connected to QlikView Server installations using Professional and Analyzer access or CAL access.

7.6 Configure Professional and Analyzer access in QlikView Server

In QlikView April 2019 or later you can apply Professional and Analyzer Users licenses. A Professional and Analyzer Users license is activated using a signed key. When moving from a CAL license to a Professional and Analyzer Users license, you must assign the new access types to users. If use a unified license, you can share users and their Professional and Analyzer access with connected QlikView and Qlik Sense deployments.

For more information on licensing QlikView, see: *Licensing QlikView (page 186)*

To switch from using a CAL license to a Professional and Analyzer Users license, you must:

- Activate the Professional and Analyzer Users license by applying the signed key to QlikView Server Service (QVS).
- Assign Professional and Analyzer access to users.

Restrictions

When you license QlikView with a Professional and Analyzer Users license, all QlikView Server Service (QVS) instances must share the same license and signed key. If you apply different signed keys to different QVS instances in the same installation, the last signed key applied propagates to the other QVS instances, overwriting the previous signed keys.

There is a maximum number of QVS instances allowed when you license a QlikView installation with a Professional and Analyzer Users license. You can see your license details in the **License Definition** box, in the QlikView Management Console.

Activating the Professional and Analyzer Users license

Do the following:

1. In the QlikView Management Console, navigate to **System** and open the **Licenses** tab.
2. Select **QlikView Server** to open the QlikView Server license menu.
3. In the **QlikView Server License** tab, select the **Use Signed Key License** check box.
   The menu changes to show the fields for activating QlikView Server using a signed key.
4. Enter the signed key in the dedicated field, and select **Apply License**.

5. A pop-up window specifies that QlikView Server (QVS) needs to be restarted. Select **OK** to restart it and apply the new license.
6. Once QlikView Server (QVS) has been restarted, the new license is applied. In the **QlikView Server License** tab, the **License Definition** box should now be populated with the details of your license. This text cannot be edited.
7. Repeat this process for each QlikView Server cluster in your QlikView installation.

Allocating Professional and Analyzer access

Once you have applied the Professional and Analyzer Users license to your QlikView Server installation, you must allocate Professional and Analyzer access to users. If you are switching from a CAL license, you must manually grant users one of the new access types. For more information, see: *Users previously assigned with CAL access (page 191)*

> *You can only grant Professional and Analyzer access to users listed in one of the Directory Service Providers for your QlikView installation.*

Allocating Professional access

Do the following:

1. In the QlikView Management Console, navigate to **System** and open the **Licenses** tab.
2. Select **QlikView Server** and open the **Professional and Analyzer access** tab.
3. Select **Professional Access**.
4. Under **Assigned Users**, click the `Manage Users` icon. An access assignment window opens.
5. Search for a user in the dedicated search field. You can search for multiple users at the same time by inputting a semicolon-separated list. The user or users matching your search are listed under **Search Result**.
6. Select the user you want to grant Professional access, and click **Add**.
7. Select **OK** to confirm the access allocation. The access assignment window closes.
8. Select **Apply** to confirm the access allocation.
9. Users with Professional access are now listed under **Assigned Users**.

Removing an access allocation

To remove Professional access from a user, click the `Delete` icon. Select **Apply** to confirm.

If you want to cancel an access removal before it is applied, click the `Restore` icon in the user row. This option is only available before you select **Apply**.

Allocating Analyzer access

Do the following:

1. In the QlikView Management Console, navigate to **System** and open the **Licenses** tab.
2. Select **QlikView Server** and open the **Professional and Analyzer access** tab.

3. Select **Analyzer Access**.
4. Under **Assigned Users**, click the `Manage Users` icon. An access assignment window opens.
5. Search for a user in the dedicated search field. You can search for multiple users at the same time by inputting a semicolon-separated list. The user or users matching your search are listed under **Search Result**.
6. Select the user you want to grant Analyzer access, and click **Add**.
7. Select **OK** to confirm the access allocation. The access assignment window closes.
8. Select **Apply** to confirm the access allocation.
9. Users with Analyzer access are now listed under **Assigned Users**.

Removing an access allocation

To remove Analyzer access from a user, click the `Delete` icon. Select **Apply** to confirm.

If you want to cancel an access removal before it is applied, click the `Restore` icon in the user row. This option is only available before you select **Apply**.

Professional and Analyzer access dynamic assignment

QlikView Server supports dynamic assignment of Professional and Analyzer access types. You can enable dynamic assignment for Professional and Analyzer access in the QlikView Management Console, see Professional and Analyzer access.

How dynamic assignment works:

- When dynamic assignment is enabled for both Professional and Analyzer access types: a user logging in is automatically assigned Professional access, if available. If not available, the user is assigned Analyzer access. If Analyzer access is not available, the user is assigned Analyzer Capacity access. If Analyzer Capacity is not available, the user cannot access QlikView.
- When dynamic assignment is only enabled for Professional access type: a user logging in is automatically assigned Professional access, if available. If not available, the user is assigned Analyzer Capacity access. If Analyzer Capacity is not available, the user cannot access QlikView.
- When dynamic assignment is only enabled for Analyzer access type: a user logging in is automatically assigned Analyzer access, if available. If not available, the user is assigned Analyzer Capacity access. If Analyzer Capacity is not available, the user cannot access QlikView.

If you enable dynamic assignment, users that have been manually assigned a Professional or Analyzer access get a double allocation. This might impact the number of available licenses for your installation. You can remove manual allocations and rely entirely on dynamic assignment to avoid double allocation of access. See: Professional and Analyzer access.

Users previously assigned with CAL access

When switching from a CAL license to a Professional and Analyzer Users license, users lose their allocated CALs. You must allocate Professional and Analyzer access manually. *AllocatedCALinfo.txt* is automatically generated when you switch from CAL to Professional and Analyzer Users license, and saved

under *%ProgramData%\QlikTech\QlikViewServer*. This text file contains a semicolon-separated list of users that were assigned with Named User CALs and Document CALs before switching to Professional and Analyzer Users licenses.

Use the *AllocatedCALinfo.txt* file to help you allocate the proper access type to QlikView users. Copy the semicolon-separated list into the Professional Access or Analyzer Access assignment window in the QlikView Management Console to retrieve the users listed and grant them the new access types. For a step-by-step description of the access assignment procedure, see: *Allocating Professional access (page 190)* and *Allocating Analyzer access (page 190)*.

Sharing users across deployments

As of the April 2019 releases of QlikView and Qlik Sense, QlikView customers can use a unified license in multiple deployments. A unified license shares the same signed key between:

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

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

When you license QlikView with a signed key, your license information, such as license definition and access allocations, is stored in a License Back-end outside the QlikView deployment. The License Back-end contacted by a Qlik License Service part of the QlikView Server deployment. Port 443 must be open to allow the communication with the License Back-end. See: *Architecture (page 10)* and *Ports (page 19)*.

If you use unified licensing, connected deployments use the same License Back-end. Users listed in connected deployments are shared with the QlikView installation, along with their access allocations. This can affect the number of Professional and Analyzer access allocations available in the QlikView installation.

When you license QlikView using a unified license, users shared between connected deployments might be displayed via their user names instead of full name. This happens when users are registered in the connected deployment, but not in QlikView Server. To properly display shared users' information, you must register them in the QlikView Server deployment. You register shared users by setting up a new Directory Service Provider that contacts the connected deployment and retrieves users' information.

7.7 OEM

General

The OEM feature prevents abuse of QlikView Servers sold under an Original Equipment Manufacturer (OEM) license and protects the revenue streams of both the OEM products and the full QlikView product. In addition, the feature helps avoid channel conflicts between QlikView OEM partners, QlikView reseller partners, and QlikView direct account managers.

The OEM feature includes the following restrictions:

- A QlikView Server delivered to a customer by an OEM partner cannot run other QlikView applications than the ones delivered by the OEM partner.
- A QlikView application delivered to a customer by an OEM partner cannot run on another QlikView Server than the one delivered by the OEM partner.

Detailed Function Description

The functions of the OEM feature are as follows:

A tag with a key is defined in the QlikView Server License Enabler File (LEF) as `OEM_PRODUCT_ID`. This LEF tag is issued once for each OEM partner and their QlikView Desktop, and QlikView Server licenses with matching tags are delivered for each QlikView Server deployment requiring this feature.

The User Preferences dialog in QlikView Desktop allows an OEM developer to embed a hash key in the QlikView document file. The hash key, which is based on the `OEM_PRODUCT_ID` key present in the QlikView Desktop license of the OEM partner, is a capitalized 40 character hex string that is stored in the Document Properties and Document metadata. In the dialog, the partner can label all keys generated for the QlikView document files. The same key can be used for multiple documents belonging to the same customer.

A QlikView Server with the `OEM_PRODUCT_ID` tag in its LEF only permits the publishing of QlikView document files with a matching key on that QlikView Server. A standard, non-OEM QlikView Server by default opens any QlikView document file, with the exception of document files containing a specific key that some OEM partners are issued with to prevent opening with any other QlikView Server than the one with a matching `OEM_PRODUCT_ID`.

The table below provides a few examples of the results of the OEM functionality.

QEM functionality examples

-	*Normal.qvw*	*OEM 1.qvw*	*OEM 2.qvw*
Normal QlikView Server	File opened	File not opened	File not opened
OEM 1 (No license lease)	File not opened	File opened	File not opened
OEM 2 (No license lease)	File not opened	File not opened	File opened

In QlikView Desktop, a document file containing a `PRODUCT_ID` is opened in user mode.