Technical White Paper: Clustering QlikView Publisher / Distribution Services

Clustering QlikView Publisher for Resilience and Horizontal Scalability

Version 1.0 – QlikView 10 and above

Technical White Paper: Clustering QlikView Publisher

Contents:

Introduction

This paper provides an overview of QlikView Publisher and its ability to be in a clustered deployment for either scalability or resilience or both. The paper will also address the architectural and installation requirements and options for building a clustered and resilient QlikView Publisher deployment.

QlikView Publisher is an optional module for QlikView Server that enables scheduling, administration, and management tools that provide a single control point 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 distributes data stored in QlikView documents to users within and outside the organization. By reducing data, each user can be presented with the information that concerns him/her. QlikView Publisher provides the ability to maintain centralized control of all the QlikView files and to schedule when and how they are reloaded and distributed. The QlikView Publisher service and user interface are now fully integrated into QlikView Server and the QlikView Management Console (QMC).

QlikView Publisher can automatically reload files and distribute them to "QlikView AccessPoint (QVS)" or by E-Mail or in an intranet. As an example, QlikView Publisher automates the production of a file for each sales person containing only their sales targets and current performance and has it automatically made available to the sales person in the way they have requested (email, on an FTP site, on QlikView Server).

QlikView Publisher, as a distribution service, ensures that the right information reaches the right user at the right time. As the use of business analysis spreads throughout the organization, controlling the distribution of analysis becomes increasingly important. QlikView Publisher provides for complete control of the distribution of a company's QlikView applications, automating the data refresh process for QlikView application data. In addition, it ensures that applications are distributed to the users through QlikView AccessPoint.

QlikView Publisher ensures that users only have access to the documents, reports, and data they need to get their jobs done. Based upon security criteria set by an administrator, Publisher can slice a QlikView application into multiple QlikView documents, add row level security, and distribute finalized QlikView documents or PDF reports to all authorized users. QlikView Publisher integrates directly with existing enterprise security infrastructure, making administration and distribution of business analysis and reports secure and efficient.

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. Out of box support for SNMP enables integration with enterprise system monitoring tools. External enterprise scheduling tools can trigger Publisher tasks using web service calls or tasks can be scheduled and/or executed on demand by a QlikView administrator.
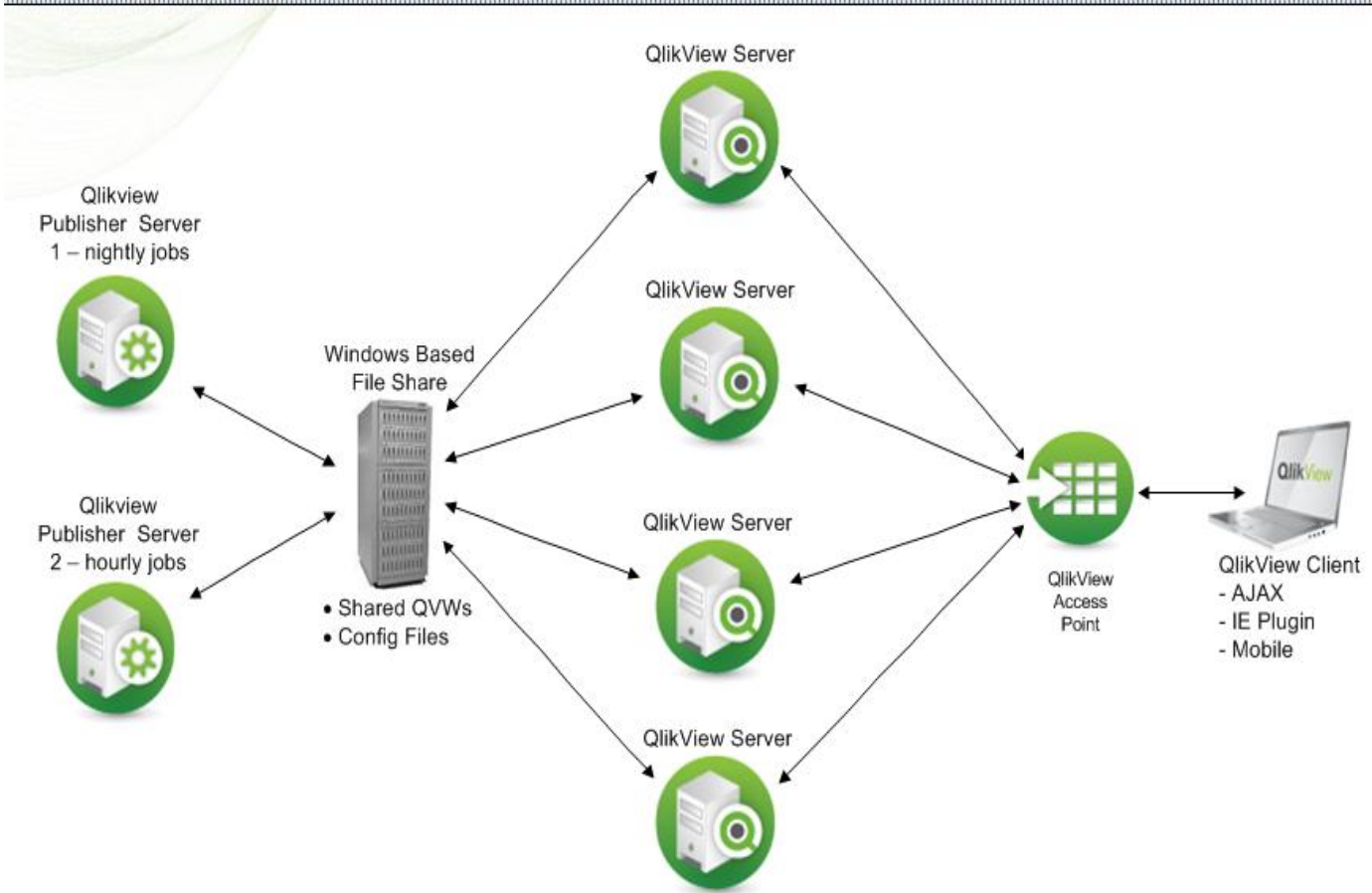
Figure 1: A two server clustered QlikView Publisher each server configured for processing different tasks and load balancing, also shown is a 4 server load balanced QlikView Server using the QlikView AccessPoint for load balancing.

The QlikView Publisher performs these main functions:

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

QlikView Source Documents, created using the QlikView Developer, reside in the Source Documents folder in ProgramData\QlikTech\SourceDocuments (QlikView default folder) using Windows Server 2008 and \Documents and Settings\All Users\Application Data\QlikTech\SourceDocuments for Windows Server 2003. This is the default QlikView location for Windows Server 2003, for a Publisher cluster this folder will need to be relocated to a shared folder designated in the QMC Publisher configuration. These source files contain either a) scripts within QVW files to extract data from the various data sources (e.g. data warehouses, Excel files, SAP, Salesforce.com) or b) the actual binary data extracts themselves within QVD files or c) a binary load from another QVW, inheriting its data model in one line of code.

Tasks are created by Administrators for data distribution and data reloads and stored in the QlikView Publisher repository as a collection of XML files or in a SQLServer database. When a task is executed, QlikView Publisher invokes QlikView Batch (QVB), which is comparable to QlikView Desktop without the GUI. QVB reloads the documents, which are stored in the Source Documents folder(s) and creates an associative QlikView database, which is stored within the 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 (QVS) using the encrypted QVP protocol or to a mail server and/or a file directory. QlikView Publisher can use the Directory Service Connector (DSC) to determine where and to whom the documents are distributed or this can be provided manually.

The User Documents folder is located at ProgramData\QlikTech\Documents (QlikView default folder) using Windows Server 2008 and for Windows Server 2003 \Documents and Settings\All Users\Application Data\QlikTech\Documents (QlikView default folder). The User Documents folder is the repository used by QlikView Server (QVS).

QlikView Publisher adds significant functionality to QlikView Server's standard reload and AccessPoint capability. QlikView Publisher includes functionality to handle field-level security and access control from central administration services like Active Directory or LDAP directories. QlikView Publisher enables complex distribution models for QlikView documents.
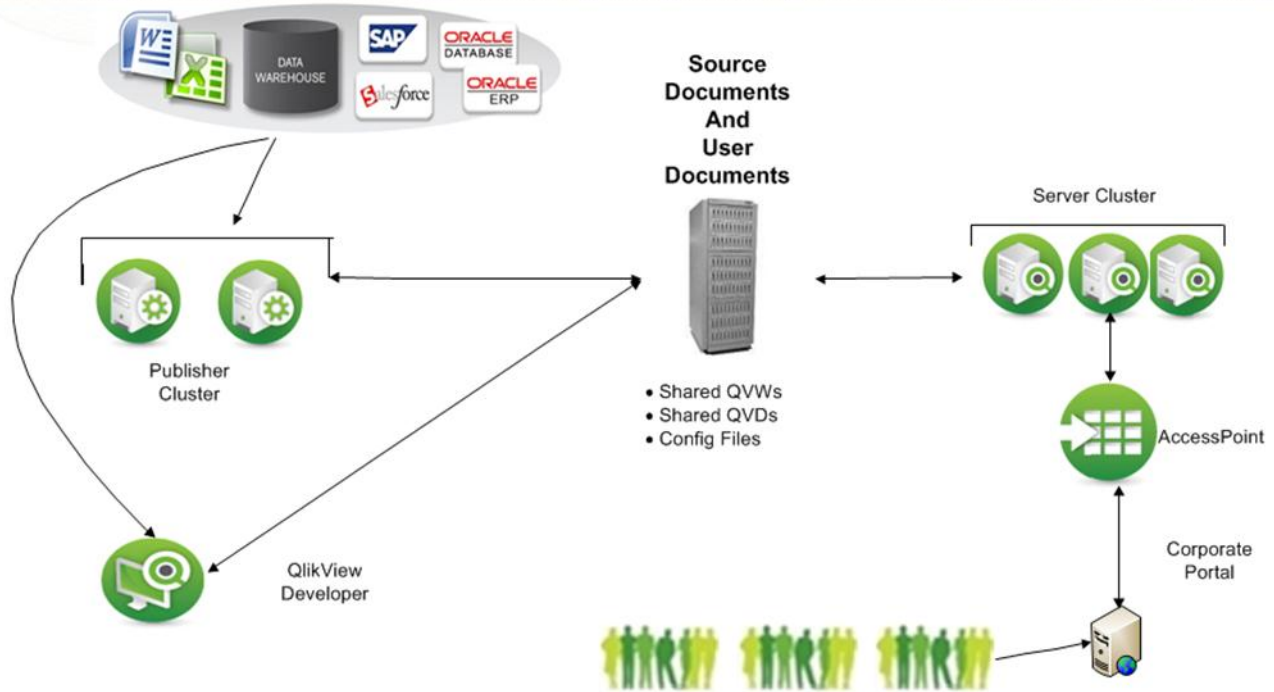
Figure 2: A two server clustered QlikView Publisher with each server configured for processing tasks and load balancing. Also shown is a 3 server load balanced QlikView Server cluster using the QlikView AccessPoint for load balancing. Documents created by QlikView Developer are stored in the Source Documents folder. QlikView Publisher tasks execute to retrieve data and store the result in the User Documents folder.

Why Cluster QlikView Publisher?

Clustering QlikView Services like Publisher provides for scaling the QlikView environment for both horizontal tasks scalability and resilience that provides for higher availability. Publisher's role in the QlikView solution is to distribute data and refresh data by criteria set by the QlikView Administrator. To accomplish this, Publisher executes many tasks, some large and some small, 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 and uses the Publisher load balancing algorithm. Publisher clusters are used to increase the scalability, availability and serviceability of data distribution and reloading of data.

Also, a Publisher Cluster license enables the configuration of Publisher services in clusters and standalone Publisher services. As an example, a Publisher cluster could be needed in the corporate office to handle the large volume of data and tasks and only a single Publisher service in a manufacturing plant where the manufacturing data is located and the Publisher service needs only to distribute documents using the manufacturing data source.

Clustering QlikView Publisher achieves the following objectives:

1   Horizontal Scalability

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

As an example, QlikView Publisher on a certain hardware server can process 100 concurrent tasks. As 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 now handle up to 200 concurrent tasks by configuring the additional server in a Publisher Cluster deployment. With added additional servers to the deployment, the Publisher services can process up to 200 concurrent tasks, 3 servers could process up to 300 concurrent tasks, if needed. In this scenario of 2 servers, the first 100 tasks would be allocated to Server A and the second 100 to Server B or if the servers were to be clustered, the tasks could be load balanced over the 2 servers.

2   Resilience

As the number of tasks on your deployment increases, the window on completing the tasks in time becomes increasingly important, clustering QlikView Distribution Services provides for resilience in the deployment. In the case above, where a single server can support 100 concurrent tasks, to build resilience into the deployment, we would consider deploying an additional server for a total of 3 servers. If a server were to "drop out" for example, due to a hardware failure, network connection issue, etc. the resilient cluster will still support up to 200 tasks. Having all 3 servers as active nodes will help reduce response times by not running all servers at 100% utilization and also limit the number of tasks and task chains affected if a node is 'lost'.

QlikView Publisher clusters are especially useful for mission-critical databases, services and files for business applications. They are based on several different redundant servers or "nodes" that replicate data, programs and server functions, so that when components fail, one of the nodes can resume service without any noticeable interruption in service. QlikView Publisher is a back-end service that provides data refreshing for QlikView users, if for some reason QlikView Publisher is not online, QlikView end users are not affected.

Requirements for a Clustered QlikView Publisher Deployment

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

1. Clustered QlikView Publisher license key
2. Shared storage area
3. Publisher load balancing strategies

1   QlikView Publisher License Keys

In a clustered environment, the QlikView Publisher servers are installed with the same license key. You can check this by examining the LEF for the following entry:

PRODUCTLEVEL;30;; where 30 is the code for Publisher.

NUMBER_OF_XS;N;; where N is the number of allowed QDS services.

A clustered QlikView Publisher deployment shares configuration and license information between themselves via the shared storage, so configuration and license management only needs to be performed once from the QlikView Management Console (QMC) for all nodes.

2   Shared Network Storage

This is required for storage of QlikView applications that are required on the cluster. QlikView Publisher recommends the storage of documents (qvw's), .meta data to be hosted on a Windows Based File Share. QlikView Publisher does support a SAN (NetApp, EMC, etc.) that is mounted to a Windows Server (2003, 2008) and then shared from that Windows server. Storage presented to a server via a SAN is required to appear as local attached storage. If SAN storage is used for Publisher, any distributed data that is accessed by QlikView Server should not reside on SAN storage.

The QlikView Distribution Services must have a shared application data directory and possibly shared source document directory as well, hence the requirement for shared network storage. All configured Publisher services should have reliable network access to the shared storage.

This is the 'Windows Based File Share' located on the left side of Figure 1 on page 4 above. A clustered QlikView deployment utilizes Windows Server based hardware.

QlikView Publisher Load Balancing and Simultaneous Tasks

The load balancing is determined by an internal ranking system based on the amount of memory available and CPU utilization. QlikTech recommends the default settings which have been extensively tested in the Scalability Center. If you desire to modify the load balancing formula, refer to QlikView Publisher Load Balancing Strategy in the Appendix.

The QlikView default number of simultaneous tasks executing per node is 4 with recommended maximum of 8 tasks per node. If there is a need to execute more than 10 Publisher tasks simultaneously per node, refer to Increasing the Number of Simultaneous Tasks in the Appendix.

Security

QlikView Publisher's role in the QlikView solution is to provide access to QlikView applications and data; therefore, it is important for QlikView Publisher to integrate with enterprise security solutions in addition to standard security features of QlikView Server.

QlikView Publisher is viewed as a Back End process within the QlikView solution. From a security standpoint, it's important to understand that the Front End does not have any open ports to the Back End. It does not send any queries to data sources on the Back End, nor do any of the user documents (QVW's) contain any connection strings to data sources located on the back end. End users can only access QlikView documents that exist on the Front End, and never in the Back End. Within the Back End, the Windows file system is always in charge of authorization; QlikView is not responsible for access privileges.
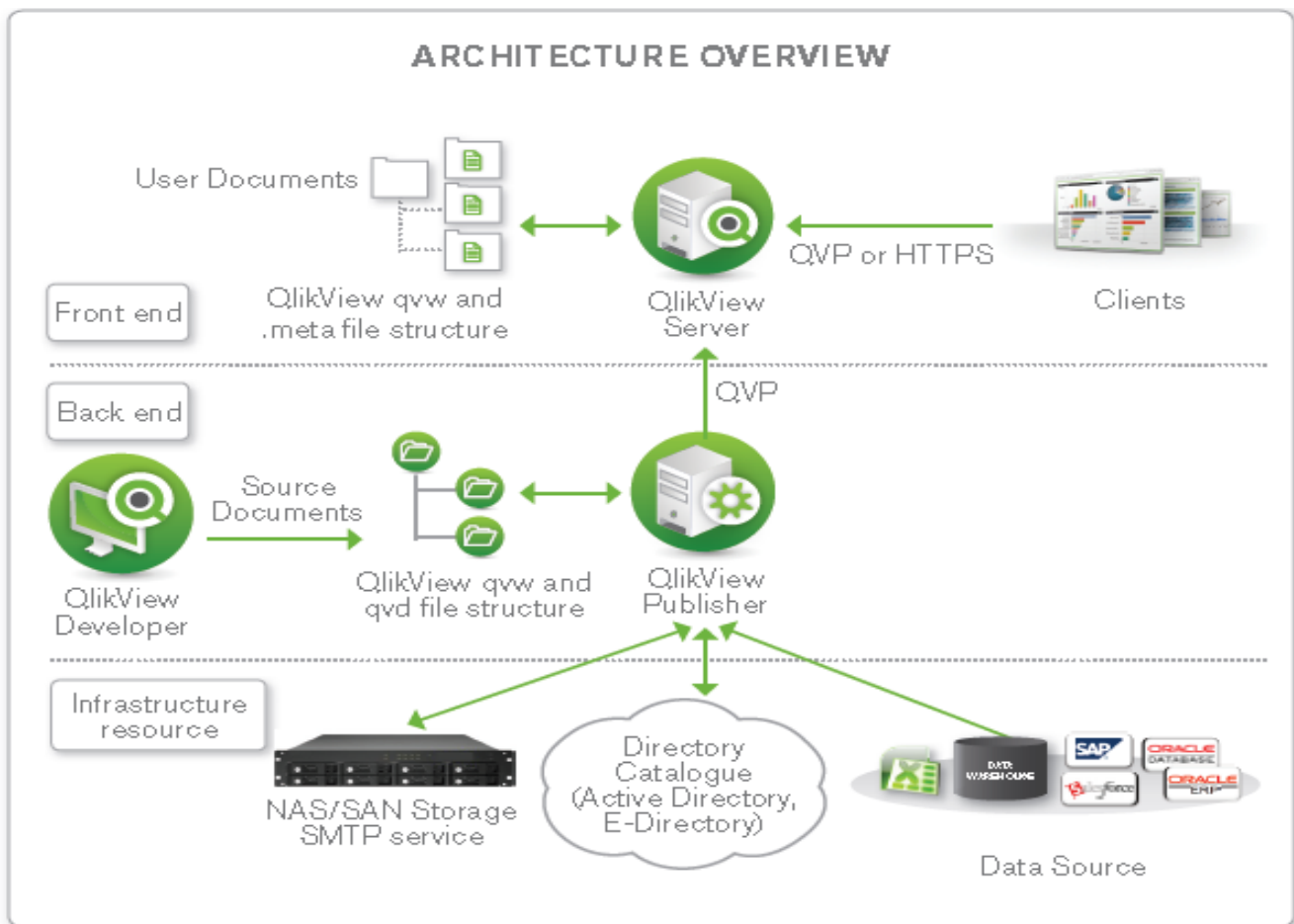

Figure 3: Shown is a simplified view of a standard QlikView deployment containing the location of the various QlikView products as well as both data and application locations.

1 Directory Services

In order to provide security for QlikView documents, QlikView Publisher can connect to an external directory service, such as Active Directory, LDAP, a database, or other sign-on solutions. The external directory service is an authentication source with which QlikView has established a trust relationship.

QlikView provides a built-in Directory Service Provider for Active Directory that allows QlikView Administrators to assign Active Directory users privileges to QlikView documents or portions thereof. QlikView Publisher leverages this built-in provider in order 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.

2 QlikView Server Authorization Modes

QlikView Server provides two mutually exclusive options for authorizing access to QlikView documents. Depending upon QlikView Server's authorization mode (NTFS or DMS), Publisher must populate the appropriate Access Control List (ACL) when assigning rights to a document. In the case of NTFS authorization, Publisher populates a standard NTFS ACL when sending documents to QlikView Server. In the case of DMS authorization, Publisher populates an ACL contained with a meta file associated with the application. Users browsing the local file system can easily recognize an application's .qvw file with its associated .qvw.meta file.

3 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 a data reduction based on users or groups defined within any external authentication source available through a custom or AD Directory Service Provider. Publisher accomplishes data reduction by the QlikView function Loop and Reduce, and is configured by a QlikView administrator via the QMC in the Documents → Source Documents tab. Publisher data reduction should not be confused with Section Access dynamic data reduction.

Configuring Publisher Clustering with QlikView 11

1 Assumption

Steps shown are performed using Windows Server 2008 R2.

2 Prerequisites

Following requirements have to be met before starting the QDS cluster configuration:

- A QlikView Publisher license that supports more than one distribution service. The Publisher LEF must contain an entry "**NUMBER_OF_XS;N**;;" where **N** is 2 or higher.
- AccessPoint (based on IIS or QVWS), Management Service (QMS), QVS, DSC are already installed on QlikView system in the network
- A domain user to run the QV services on every machine is available
- A shared storage device, QlikTech recommends a shared device mounted as a Windows Based File Share.
  > All QDS cluster nodes need read /write access to the following data centrally stored:
  > - Publisher status, configuration and log files
  > - QlikView Source Documents

3 Step-by-Step Installation

3.1 Prepare the shared storage device

Create folders for the files accessed by every Publisher cluster node, these folders are used in the screenshots as an example.

- *\\server1\ProgramData\QlikTech\DistributionService* for the Application Folder
- *\\server1\ProgramData\QlikTech\SourceDocuments* for the Source Document Folder

3.2 Prepare the cluster nodes

On each planned QDS cluster node perform the following configuration
- Login as Administrator
- QlikTech recommends the use of a Firewall to secure the QlikView solution
    - Open necessary ports for QlikView in the Windows Firewall or external Firewall device.
    - QlikView requires these ports to "opened" for the QlikView services for a complete QlikView solution:

Service	Port
QlikView Distribution Service (Publisher) –required for Publisher	4720/tcp
Directory Service Connector – required for Publisher	4730/tcp
QlikView Management Service – required for Publisher	4780/tcp
QlikView Webserver / IIS Configuration	4750/tcp
QVS Configuration	4749/tcp
QVP Communication	4747/tcp
QMS (EDX Calls) – required for Publisher	4799/tcp

- Deactivate Internet Explorer Enhanced Security Configuration for Administrators
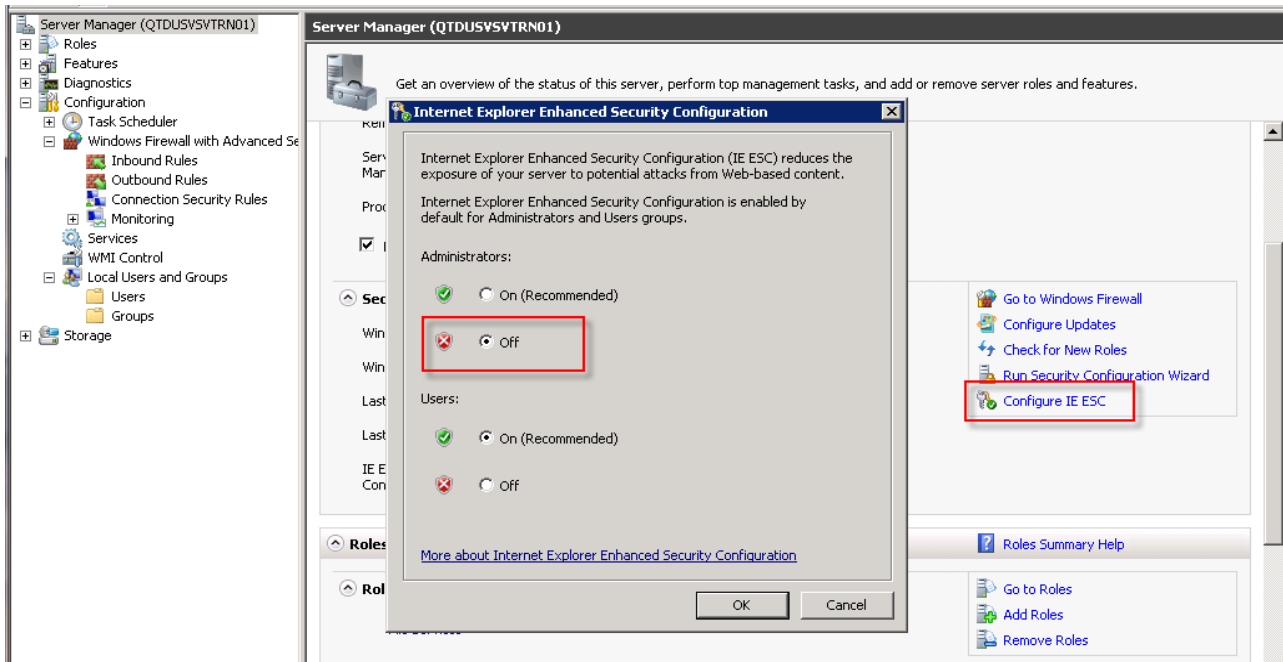
- By default, Windows Server 2003 and 2008 ship with Internet Explorer in enhanced security configuration, which is basically a locked down version that adds a bit of extra security to your servers for Web Browsing. When Internet Explorer Enhanced Security Configuration is enabled, it may create problems in viewing the management console and service content. You may be able to leave IE ESC on, but if you experience any issues, you will need to turn the feaure off for the Administrators group.
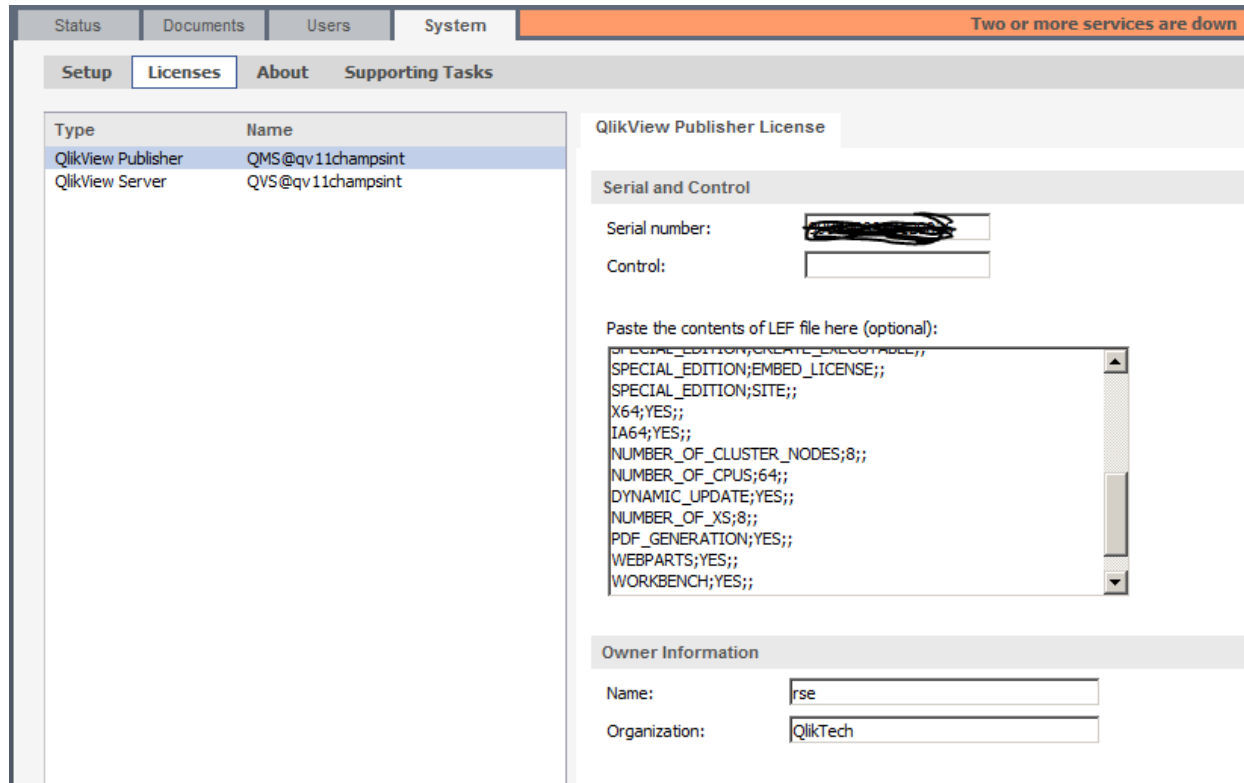
- Add Domain User the QlikView services should use to Local Administrators Group
- Start QlikView x64 server setup and select Custom Installation then install feature "Reload/Distribution Engine" only on each node where Publisher is to reside.
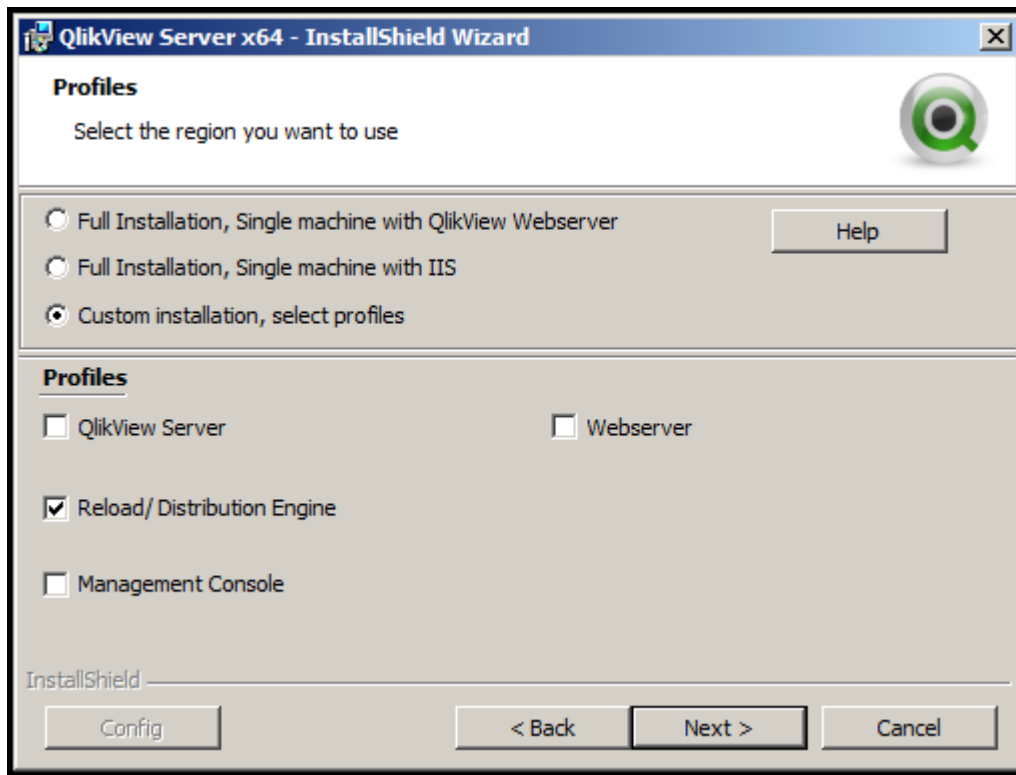
- Enter the QlikView service account credentials
- Finish the setup and restart the system immediately

3.3 Configure QDS Cluster in the QMC

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

- On the System→ Setup tab, add first QDS cluster node below Distribution Services
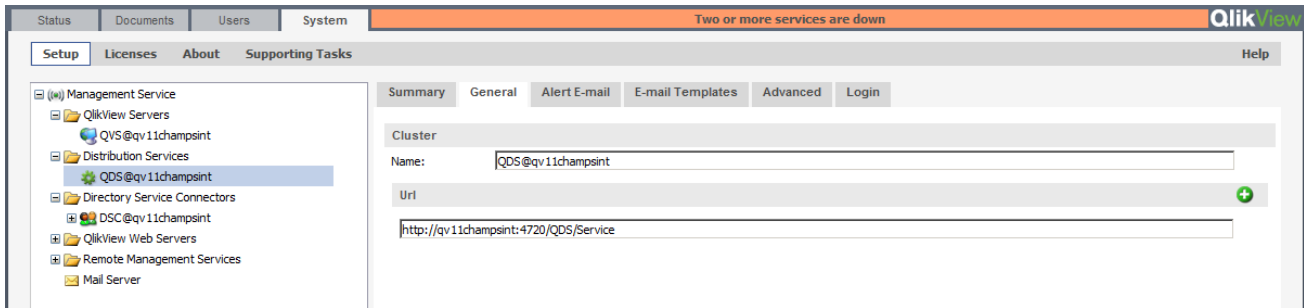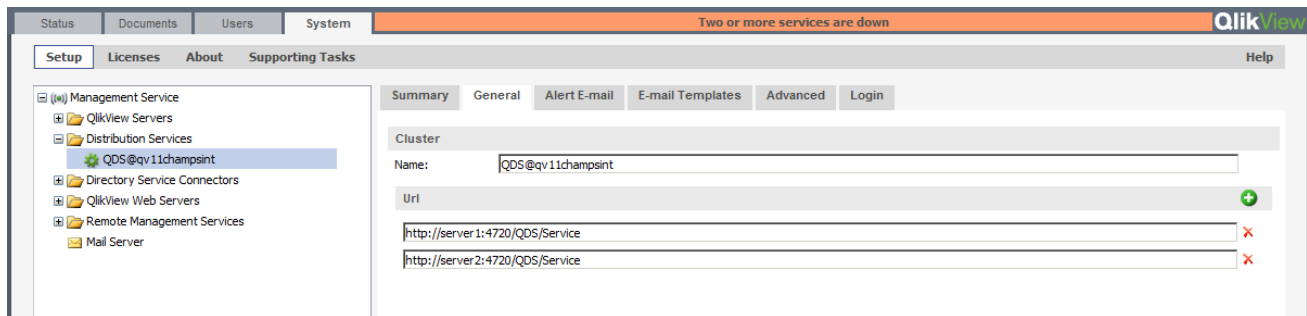

- Switch Application Data Folder & Source Folders to Shared Device folder paths using UNC-syntax
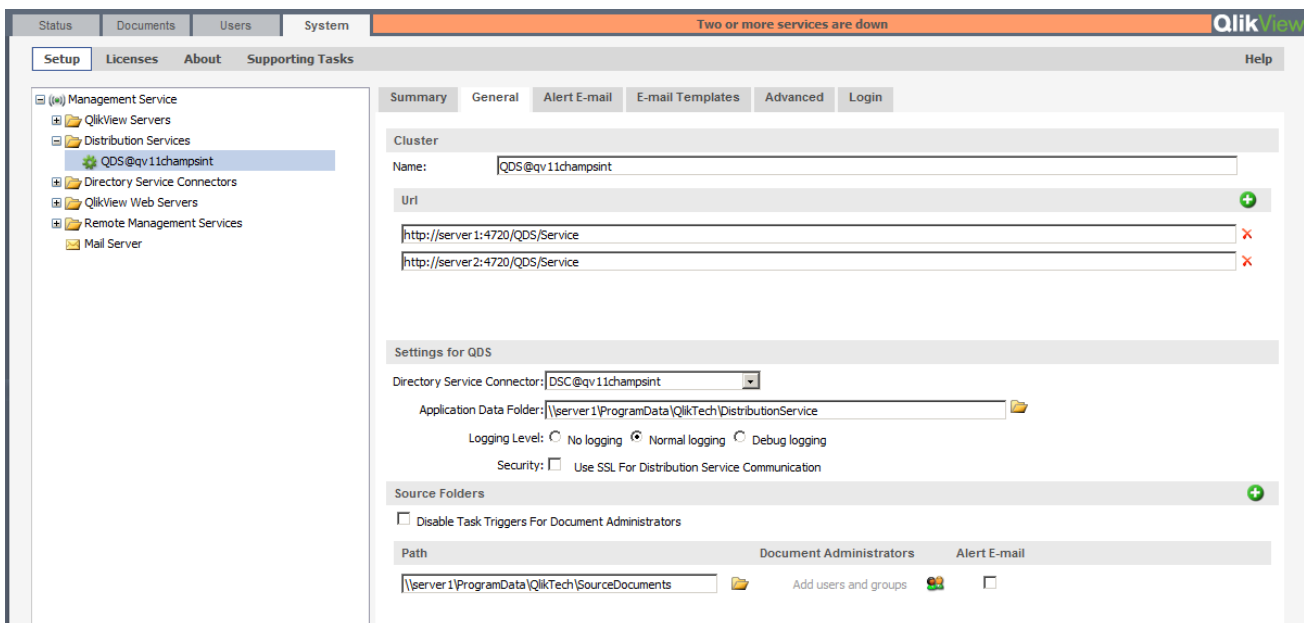
- Press Apply and restart QDS service manually
- Add each additional QDS cluster node in URL format

- Press Apply and restart QDS service on all nodes manually

Summary

This document proposed to provide an understanding of the infrastructure requirements for clustering QlikView Publisher services to help in planning your clustered deployment.

As a recap these are the things to consider:

- Why am I clustering – resilience or additional QlikView Publisher resources or both?

- How many QlikView Publisher servers will I cluster?

- Do I have a 'cluster enabled' QlikView Publisher License Key?

    - Does it have the relevant number of servers set?

- Shared Storage infrastructure in place?

If you have further questions or require assistance in building your QlikView Publisher cluster please contact your local QlikTech office for assistance from our Professional Services Team.

Appendix:

1   QlikView Publisher Load Balancing Strategy

The load balancing is determined by an internal ranking system based on the amount of memory available and CPU utilization. QlikTech recommends the default settings which have been extensively tested in the Scalability Center. If you choose to change the default settings, as to how the ranking is done, you can do so by editing the configuration file QlikViewDistributionService.exe.config. The key (below) is written in JavaScript.

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

AverageCPULoad – The average CPU load of all running QVBs.
MemoryUsage – The total memory usage for the entire application.
TotalMemory – The total amount of memory on the server.
NumberOfQlikViewEngines – The number of the QlikView engines currently in use.
MaxQlikViewEngines – The configured value of max QlikView engines.
NumberOfRunningTasks – The number of currently running tasks.

2   Increasing the Number of Simultaneous Tasks

The QlikView default number of simultaneous tasks executing per node is 4 with recommended maximum of 8 tasks per node. If there is a need to execute more than 10 Publisher tasks simultaneously per 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 10 or more simultaneous tasks. Another option would be to add addition servers for Publisher tasks.

Backup the Windows Server registry. Change the Windows Server registry setting:

From: 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

(Default for SharedSection is 1024,3072,512 in 32bit or 1024,3072,768 in x64)

Read more on http://blogs.msdn.com/ntdebugging/archive/2007/07/05/desktop-heap-part-2.aspx

To:

Change the GDI and User handle max count in the registry to **SharedSection=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

Also, change Max number of simultaneous QlikView engines for distribution setting in QMC to the number of engines needed.
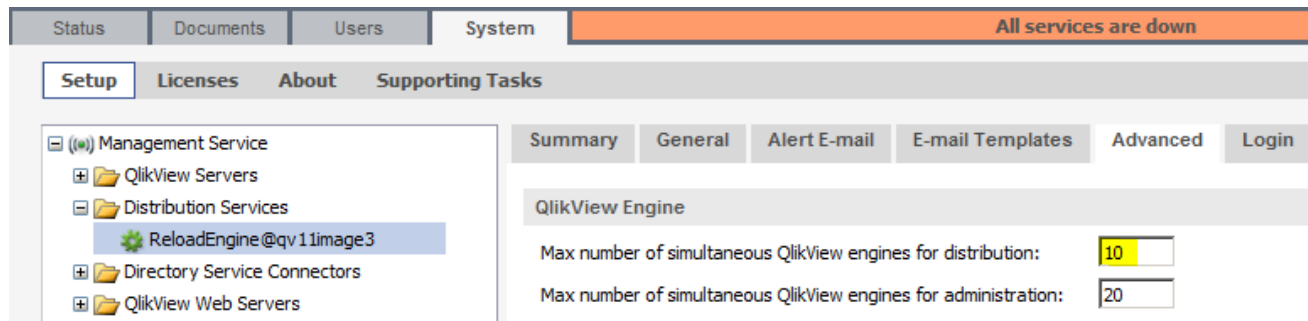
3   Trouble Shooting

If the log message "The network BIOS command limit has been reached" occurs in the Debug-Cluster log, you need to increase the limit for long-term sessions in the registry. Failure to do so may result in tasks not being run.

Increase the following parameters in the registry:

      HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\lanmanworkstation\parameters\MaxCmds

And

      HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\lanmanserver\parameters\MaxMpxCt

This issue only occurs on Windows Server 2000, Windows XP and Windows Server 2003!

More information is available on:

http://blogs.msdn.com/b/ntdebugging/archive/2007/01/04/desktop-heap-overview.aspx and

http://support.microsoft.com/kb/810886.

For QlikView 10 and 11 these settings in the config.xml file on the server where the QlikView Publisher service is installed, usually under:

> "C:\Documents and Settings\All Users\Application Data\QlikTech\DistributionService" for Windows 2003 Server or

> "C:\ProgramData\QlikTech\DistributionService on Windows 2008 Server".

4 Definitions/Terminology

Cluster:

'A computer cluster is a group of linked computers, working together closely so that in many respects they form a single computer. The components of a cluster are commonly, but not always, connected to each other through fast local area networks. Clusters are usually deployed to improve performance and/or availability over that provided by a single computer, while typically being much more cost-effective than single computers of comparable speed or availability.' [1]

High-availability (HA) clusters:

'High-availability clusters (also known as failover clusters) are implemented primarily for the purpose of improving the availability of services which the cluster provides. They operate by having redundant nodes, which are then used to provide service when system components fail. The most common size for an HA cluster is two nodes, which is the minimum requirement to provide redundancy. HA cluster implementations attempt to manage the redundancy inherent in a cluster to eliminate single points of failure.' [2]

Load-balancing clusters:

---

[1] http://en.wikipedia.org/wiki/Computer_cluster
[2] http://en.wikipedia.org/wiki/Computer_cluster#High-availability_.28HA.29_clusters

'Load-balancing clusters operate by distributing a workload evenly over multiple back end nodes. Typically the cluster will be configured with multiple redundant load-balancing front ends.' [3]

Node

A single QlikView Distribution service instance on a server.

Active Node

An Active Node is accepting and processing work.

Passive Node

A Passive Node is inactive, waiting to process work should an active node in the cluster fail.

Network Load Balancer

'In computer networking, load balancing is a technique to spread work between two or more computers, network links, CPUs, hard drives, or other resources, in order to get optimal resource utilization, throughput, or response time. Using multiple components with load balancing, instead of a single component, may increase reliability through redundancy. The balancing service is usually provided by a dedicated program or hardware device (such as a multilayer switch).' [4]

Storage Area Network

'A storage area network (SAN) is architecture to attach remote computer storage devices (such as disk arrays, tape libraries and optical jukeboxes) to servers in such a way that, to the operating system, the devices appear as locally attached.' [5]

---

[3] http://en.wikipedia.org/wiki/Computer_cluster#Load-balancing_clusters
[4] http://en.wikipedia.org/wiki/Load_balancer
[5] http://en.wikipedia.org/wiki/Storage_area_network