EQUELLA®

Clustering Configuration Guide

Version 6.4

Document History

Document No.	Reviewed		Finalised		Published	
1	22/05/2015		22/05/2015		22/05/2015	

May 2015 edition.

Property of:

Pearson
Level 1, 11 Elizabeth Street
Hobart, Australia, 7000

Contact documentation@equella.com for matters relating to documentation.

Table of Contents

EQUELLA clustering overview

The purpose of this guide is to provide system administrators with an understanding of the EQUELLA clustering process.

EQUELLA clustering involves two or more EQUELLA instances sharing a common Apache Zookeeper, database and filestore behind a load balancer. The load balancer directs clients to one of the members (EQUELLA instances) of the cluster. A simple clustering scheme is shown in Figure 1.
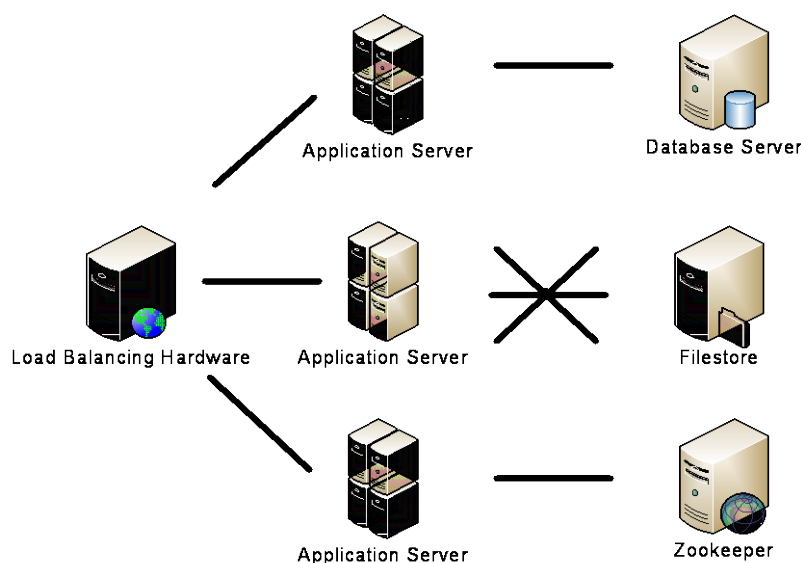

Figure 1 Example simple clustering scheme

Each of these components will typically be on dedicated hardware.

Clustering environment

Prior to setting up clustering the following conditions must be met:

- EQUELLA must be installed on all cluster nodes, but not started. If you are using Apache™ as your load balancer *and* Apache is installed on the same hardware as one of your EQUELLA nodes, then you *must not* bind EQUELLA to port 80, because Apache will need that port.

- All application servers, Zookeeper and database instances must be able to communicate with each other via TCP.

- When clustered, EQUELLA requires a shared filestore to house the cluster file repository. This filestore must be ready and available to all hardware nodes in the cluster prior to operating EQUELLA. If each node has a unique device path to the common filestore, those device paths should be noted for ease of configuration.

- Once EQUELLA is installed on all hosts of the cluster, the network file share must be determined. Also ensure that the user who is designated to start EQUELLA services on each node is the same (in Windows™, this must be a network user) and that the user has write permissions to the filestore (on each node).
- EQUELLA must have access through any firewalls.

Install Apache Zookeeper

Apache Zookeeper is used for coordination and discovery services between EQUELLA nodes, and is a required component for a clustered setup.

Zookeeper itself should be installed in a cluster, referred to in the Zookeeper documentation as a *quorum*. While a single instance of Zookeeper will suffice, it is recommended that a 3-node quorum is configured for production environments for resilience to errors. Please note that it is *highly recommended* that a quorum consists of an odd number of nodes – typically either 3 or 5.

The Zookeeper instance or quorum can be run on its own hardware or virtual machines, or installed alongside the EQUELLA application servers. A single Zookeeper quorum can also be utilised by multiple EQUELLA clusters.

A detailed overview of what Zookeeper provides as a service can be found at http://zookeeper.apache.org/doc/trunk/zookeeperOver.html

Installing a standalone Zookeeper

1. Go to *http://www.apache.org/dyn/closer.cgi/zookeeper/,*select a mirror and then navigate to 'stable' directory and download 3.4.6.
2. Unzip the file to your chosen directory.
3. Copy **/conf/zoo_sample.cfg** to **/conf/zoo.cfg** and edit the **dataDir** directory to point to a new directory.
4. Start ZooKeeper by running the appropriate following command:

    [Unix]   bin/zkServer.sh start

    [Windows] bin\zkServer

Configuring a Zookeeper quorum

1. Follow the above instructions for installing standalone Zookeeper instances for each node in the quorum.
2. Stop each of the services.
3. Append the following properties to each **/conf/zoo.conf**, changing **zoo1**, **zoo2** and **zoo3** to be the hostname or IP address of each node in the quorum. Replicate the **server.<n>** property if you have more than 3 nodes:

    server.1=zoo1:2888:3888

    server.2=zoo2:2888:3888

    server.3=zoo3:2888:3888

4. On each Zookeeper node create a **myid** file in the **dataDir** set in the **zoo.conf** file containing the number **<n>** assigned to the node in the **zoo.conf**. For example when configuring the **zoo1** node the **myid** file would contain a **1** as that is the ID listed in the **zoo.conf.**

5. Restart each Zookeeper node.

Further information on Zookeeper customization can be found at http://zookeeper.apache.org/doc/trunk/zookeeperStarted.html

Configure EQUELLA on the first host

The following sections describe configuration modifications to various files suitable for Windows Server™ or Unix™ using two nodes and a software load balancer.

Edit the EQUELLA configuration files

Edit the following files, substituting the correct values for the variables. (All configuration files can be found in the <path-to-equella>**learningedge-config** directory.)

mandatory-config.properties

In the <path-to-equella>**learningedge-config\mandatory-config.properties** file edit the **filestore.root** property to the network location of the shared file store. For example, a Windows system using an SMB share would look like:

```
filestore.root = //some.host/share.name/
```

Edit the **admin.url** property to be the EQUELLA address of the hosting server:

```
admin.url = http://equellaadmin.myinstitution.com
```

If Apache HTTP Server is being used as a load balancer, and AJP is being used to communicate with the application servers, then the **ajp.port** property should be uncommented and set to the relevant port:

```
ajp.port = 8009
```

optional-config.properties

In the <path-to-equella>**learningedge-config\optional-config.properties** file, edit the **zookeeper.instances** variable to include the **<address>:<port>** of the Zookeeper instance. The **<address>** portion can be an IP address or hostname. Multiple Zookeeper instances should be separated by commas:

```
zookeeper.instances = 192.168.1.177:2181,zookeeper2.my.lan:2181
```

If you are using the Zookeeper quorum for more than one EQUELLA cluster, then the **zookeeper.prefix** property must be set with a unique name for the cluster:

```
zookeeper.prefix = equella63staging
```

The property **zookeeper.nodeId** can be set in order to append a human readable string to the NODE ID which makes identifying nodes much easier. If this property is not set, the EQUELLA application will try to use the hostname of the server on which EQUELLA is running. Failing that, it will revert to a plain UUID NODE ID. Note that this string should be unique between cluster nodes.

```
zookeeper.nodeId = nodeX
```

EQUELLA will do its best to determine the public IP address of the machine it is running on. If the machine has more than one up, non-virtual network interface then EQUELLA will present an error in its logs and fail to start. To resolve this, the **messaging.bindAddress** property must be set with an IP address accessible by the other application server nodes:

```
messaging.bindAddress = 192.168.1.100
```

Finally, EQUELLA will attempt to bind to port 8999 for communications between application servers. If this port is already bound to by another service, then the **messaging.bindPort** property may be set to an alternative. Note that this port *does not* need to match other application servers:

```
messaging.bindAddress = 192.168.1.100
```

If AJP is the chosen protocol for communications between the load balancer and the application servers, the **jvmroute.id** property should be set to a unique value for each node in the cluster which facilitates sticky sessions:

```
jvmroute.id = NODE1
```

NOTE: Although AJP is a supported protocol, it is not recommended.

Configure EQUELLA on subsequent hosts

To configure EQUELLA on subsequent hosts, install EQUELLA and repeat the steps from the Configure EQUELLA on the first host section on page 5 on each node of your cluster.

Ensure the variables used to configure the first host are updated where appropriate to refer to the second (or subsequent) hosts, to maintain uniqueness.

Test the clustered environment

The following tests ensure the clustered environment is working correctly by testing each server is being accessed, cluster tasks are being delegated correctly and that a single server failure is handled without error.

Display each server's unique ID

To view details about all of the registered application nodes in the cluster:

1. Login to the EQUELLA server administration.
2. Select **Health check.**

You can also select the **Enable cluster debugging** checkbox to show debug information in the page footer. An example is shown in Figure 1.

Test cluster node server configuration

Additionally, further information about each node's services can be found by clicking the drop-down arrow. An example is shown in Figure 2.

Figure 2 Node server information

To test each server is running and being called correctly:

1. Log in to an EQUELLA instance and identify the unique cluster ID that is displayed in the bottom left-hand corner of the window, and then stop that instance.

2. Force a refresh after the service has stopped (Ctrl + R / Ctrl + F5).

3. The unique cluster ID should now identify a different machine in the cluster.

4. Repeat this process until all the cluster servers have been identified and are running correctly.

Test server fail over

This test stops a server part way through a resource contribution to simulate a server failure.

1. Ensure all the cluster web servers are running.

2. Create a test user (refer to the *EQUELLA User Management Configuration Guide* for further information).

3. Log in to EQUELLA using the new user and open a contribution wizard with multiple pages.

4. Enter some details on the first wizard page, and then navigate to another page in the wizard. (This commits the changes for temporary storage. Without this, the server wouldn't know that anything had been written.)

5. Leave the contribution wizard open, wait approximately 40 seconds and stop the server identified by the cluster ID. Note that user sessions are only persisted to the database every 30 seconds to reduce load.

Once the server has stopped:

6. Return to the open contribution wizard.

7. Navigate to the wizard page containing the data entered in step 4.

A correctly configured cluster will perform this series of tasks without error.

Appendix A

Configure Apache™ HTTPD as a load balancer

Download and install Apache HTTP Server 2.4

Download Apache HTTP 2.4 appropriate for your machine's architecture from http://httpd.apache.org/download.cgi#apache24.

1. Install HTTPD 2.4 with default options. Make a note of the installation directory.
2. Ensure that the **mod_proxy**, **mod_proxy_balancer, mod_lbmethod_byrequests** and **mod_headers** modules are enabled.
3. Start and test that HTTPD is working.
4. Shut down HTTPD.

To configure Apache as a load balancer

Create a virtual host file for your site, where you can configure the load balancing proxy, or create one in Apache's *httpd.conf* file (or equivalent).

```
<VirtualHost *:80>
    ServerName <EXTERNAL-SERVER-NAME>

    ProxyPass / balancer://mycluster/ nocanon
    ProxyPreserveHost On

    # Using mod header (cookie created by loadbalancer)
    Header add Set-Cookie \
        "ROUTEID=.%{BALANCER_WORKER_ROUTE}e; path=/" \
        env=BALANCER_ROUTE_CHANGED

    <Proxy balancer://mycluster/ >
        BalancerMember http://<NODE-1>:<port> route=1
        BalancerMember http://<NODE-2>:<port> route=2
        BalancerMember http://<NODE-3>:<port> route=3
        ProxySet stickysession=ROUTEID
    </Proxy>
</VirtualHost>
```

To configure Apache as load balancer and SSL terminator

When using Apache to terminate SSL and act as the cluster load balancer, the **https.port** property should first be uncommented in the <path-to-equella>**learningedge-config\mandatory-config.properties** file.

```
https.port = 8443
```

The following example configuration will terminate SSL and direct the then non-secure communications to EQUELLA. You will also need to ensure that **mod_ssl** is enabled:

```
<VirtualHost *:443>
    ServerName <EXTERNAL-SERVER-NAME>
```

```
        ProxyPass / balancer://mycluster/ nocanon
        ProxyPreserveHost On

        # Using mod header (cookie created by loadbalancer)
        Header add Set-Cookie \
            "ROUTEID=.%{BALANCER_WORKER_ROUTE}e; path=/" \
            env=BALANCER_ROUTE_CHANGED

        <Proxy balancer://mycluster/ >
            BalancerMember http://<NODE-1>:<https.port> route=1
            BalancerMember http://<NODE-2>:<https.port> route=2
            BalancerMember http://<NODE-3>:<https.port> route=3
            ProxySet stickysession=ROUTEID
        </Proxy>

        SSLEngine On
        SSLProxyEngine On
        SSLCertificateFile <path-to-cert.pem>
        SSLCertificateKeyFile <path-to-cert.key>
    </VirtualHost>
```

Contact Client Support

We are always happy to help.

If your organisation has a support agreement with EQUELLA then help is available at
http://equella.custhelp.com.