LISTSERV Maestro Admin Tech Doc 5

Multi Server Installation

May 22, 2014 | © L-Soft Sweden AB
lsoft.com

This document is a LISTSERV Maestro Admin Tech Doc. Each admin tech doc documents a certain facet of the LISTERV Maestro administration on a technical level. This document is number 5 of the collection of admin tech docs and explains the topic “Multi Server Installation”.

Last updated for LISTSERV Maestro 6.0-1 on May 22, 2014. The information in this document also applies to later LISTSERV Maestro versions, unless a newer version of the document supersedes it.

Information in this document is subject to change without notice. Companies, names, and data used in examples herein are fictitious unless otherwise noted. L-Soft Sweden AB does not endorse or approve the use of any of the product names or trademarks appearing in this document.

All of L-Soft's manuals for LISTSERV are available in ASCII-text format via LISTSERV and in popular word-processing formats via ftp.lsoft.com. They are also available on the World Wide Web at the following URL:

URL: http://www.lsoft.com/manuals.html

L-Soft invites comment on its manuals. Please feel free to send your comments by e-mail to: MANUALS@LSOFT.COM

Table of Contents

1 Distributing Components on Several Servers

You can distribute the components of LISTSERV Maestro in a way, that each of them runs on its own server – or you can run all of them on the same server, or in any combination.

Distributing components has several advantages:

- Load Distribution: Processor and disk load is shared between several servers, giving each component more “room” to operate.

- Separate Maintenance: You do not have to shut down or restart all components, whenever a maintenance task of one of them requires you to do so. The other components may continue running (of course, when you shut down a component that other components rely on, this is no longer entirely true, since the other components cannot run properly while the one component is down).

  Especially the Maestro Tracker component has very demanding uptime requirements: This component should constantly be running to be able to collect the tracking data from the mails that are sent. It can only do so while it is running and connected to the internet.

  Therefore it is not a good idea to shut down the server on which the Maestro Tracker component is running – this should only be done as a last resort.

  Other components however do not have these strict uptime requirements. For example, the Maestro User Interface component actually needs to be restarted if a new database connection is to be used or if a new JDBC database driver is installed (see following chapters). Therefore it may be a good idea to have the Maestro User Interface and the Maestro Tracker components on separate servers.

For high-end, (very) high-volume installations, a component distribution on four servers is recommended for optimal performance:

- User-Interface and Hub server: Contains the LISTSERV Maestro components Maestro User Interface and Administration Hub.

- Tracker server: Contains the Maestro Tracker component.

- Database server: Contains the database used by the Maestro User Interface component.

- LISTSERV server: Contains the LISTSERV external component.

1.1 Fresh Installation with Distributed Components

A fresh installation with distributed components is quite straight forward: To install any of the three LISTSERV Maestro components, simply run the LISTSERV Maestro setup on the server where you want to install the component(s) and then select the required components from the list, while leaving all components you want to install on other servers unchecked.

The other external components (database and LISTSERV) come with their own setups anyway. Simply execute their setups on the respective servers (see also the chapter about using a different database in this document).

1.2 Changing the Component Distribution of an Existing LISTSERV Maestro Instance

It is possible to change the distribution of the components even for an existing LISTSERV Maestro instance.

The most common example for this is probably the following: Initially you installed LISTSERV Maestro on a single server. Over time, as LISTSERV Maestro was used more and more, the workload on this server increased, and you now want to have a separate server for the TRK component, i.e. you want to split LISTSERV Maestro on two servers: One for LUI/HUB and one for TRK.

This, and all other scenarios that involve a change of the current component distribution, are a sub-case of the “moving a LISTSERV Maestro instance” scenario that is described in “LMA Admin Tech Doc 4 – Moving LISTSERV Maestro” . Please see this tech doc for more details.

2 Authenticating and Encrypting Communication Between LISTSERV Maestro Components

A LISTSERV Maestro installation consists of several components: The three main components are the Maestro User Interface (LUI), Administration Hub (HUB) and Maestro Tracker (TRK). These components communicate with each other over so called network sockets which are connected to ports. This happens via the “internal communications port” (default: port 1099) and the “Tracker communications port” (default: port 7000).

These ports are opened for incoming connections on the servers where the components are installed as described in “LMA Admin Tech Doc 6 – IP Addresses and Port” and are usually protected against unauthorized access by configuring the firewall that protects the server(s) in the right manner (see “LMA Admin Tech Doc 7 – Installing Behind a Firewall or Proxy”).

However, if you have stricter security requirements and need to make sure that there is no unauthorized access from anyone (either from inside or outside the firewall) to these ports and/or you want to make sure that no one can listen in on the communication between the components, then you can additionally use the Secure Sockets Layer (SSL) for the communication on these ports and employ its authentication and encryption features.

The following subchapters describe how you can configure your LISTSERV Maestro components to use SSL for their inter-component communication.

Note: Authenticating and encrypting the communication between the components of LISTSERV Maestro as described in this chapter is a separate issue than securing LISTSERV Maestro access with HTTPS), even though the two have many similarities and can even be combined. If what you are looking for is actually to authenticate and encrypt the communication between the user and the Maestro User Interface or Administration Hub (and not the communication between the separate components of LISTSERV Maestro) then please see “LMA Admin Tech Doc 9 – Securing Access with HTTPS” instead.

2.1 General Considerations

Before you continue, it is recommended that you read the section “Introduction to Secure Communication” in “LMA Admin Tech Doc 9 – Securing Access with HTTPS” and its general discussion of authentication, encryption and certificates, which does apply in the same manner to using SSL for the inter-component communication. In the following, it is assumed that you already understand the role of server certificates and trusted root certificates.

Securing the inter-component communication is an “all-or-nothing” affair, i.e. you can either secure all communication (on the “internal communication port” and the “Tracker communication port”) between all three components (LUI, HUB and TRK) or none of it. You cannot secure only the communication between some components (or on some ports) but not between the others. (However, securing inter-component communication is independent of securing user access via HTTPS, i.e. you can secure either the inter-component communication, or the user access, or both of them, or none of them, depending on your security needs.)

Also, securing inter-component communication happens on a per-server basis: It does not matter how many of the LISTSERV Maestro components are installed on one server, for the purpose of securing the communication, we deal with them as one single server entity (which, when secured, will secure the communication of all components installed on it). Therefore, in the following sub-chapters, we will not speak of the individual LISTSERV Maestro components, but only of separate LISTSERV Maestro servers (or just short “servers”). For any given LISTSERV Maestro installation, there may be one, two or three such servers, depending on how the components are distributed. And because secure inter-component communication is an “all-or-nothing” affair, we need to secure all of those servers or none of them.

Note, that securing the inter-component communication deals only with the communication on the described ports between the three main components LUI, HUB and TRK. Communication to the system database or any user database and communication to LISTSERV is not part of this discussion.

2.2 Obtaining and Installing Server Certificates

A fully authenticated secure communication via SSL is divided into two parts: First the authentication, that the two communication partners really are the entities they claim they are, and second the encryption of all data passed between the two.

In the SSL protocol, the two partners first authenticate each other by presenting their signed server certificates to each other. Each partner then checks the signature of the other’s certificate to see if it has been signed by a trusted certificate authority (CA). Once the two partners have thus authenticated each other, they then agree on an encryption which they then employ for all subsequent communication.

As a consequence of this, if we want to have authenticated SSL communication between two servers, both servers must possess a server certificate which has been signed by a certificate authority (CA) which the other server accepts as a trusted authority.

So the first step in securing the inter-component communication is to obtain signed server certificates for all LISTSERV Maestro servers. Obtaining and installing these certificates is very similar to the steps described for the certificate that is required to secure user-access with HTTPS (see “LMA Admin Tech Doc 9 – Securing Access with HTTPS”), so we do not repeat the description of the required steps here, but instead simply include references to the appropriate sub-chapters of chapter **Error! Reference source not found.**.

Before we launch into the detailed steps of obtaining and installing such a certificate, first some issues that need to be taken into consideration:

- You need to obtain a signed server certificate for all LISTSERV Maestro servers (one for each server). You cannot simply obtain any server certificate and use it on a server of your choice (or use the same certificate on several servers): The certificate is always bound to the explicit server name that you chose when you created the certificate. Should you want to rename your server, you would have to obtain a new certificate for the new name.

- Make sure that your servers have the right names configured and are using these correct names for network communication. Sometimes, if the operating system is misconfigured or if a server has several names assigned, the name that a server uses to identify himself during network communication is not the one you would expect. In that case, if the server certificate was created with one name but the server identifies itself with another name, the SSL communication will not work, because the communication partner cannot match the certificate to the server.

  For example, if a server has several names assigned, it may happen that the sever uses one name to identify itself to communication partners which reside on the same server, and a second name for partners on different servers but in the same network zone, and a third name for servers in a different zone. In that case, you either need to consolidate your server so that it uses the same name for all these different types of connections, or you actually must obtain and install one certificate for each of these names (using several certificates for the same server, with different server-names, is an untested feature – test and employ at your own risk!).

- It is recommended that you obtain all server certificates from the same CA, because then you only have to deal with one trusted root authority certificate (see below for details).

- If for a certain server you already have obtained and installed (in a keystore file of your choosing) a signed certificate for the purpose of securing user access to this server with HTTPS (see "LMA Admin Tech Doc 9 – Securing Access with HTTPS"), then you do not need to obtain another certificate for this server: The same certificate can then also be used for securing the inter-component communication on that server. Simply supply the path and password of your existing keystore file when performing the step described in chapter "2.3 Enabling Secure Inter-Component Communication".

Obtaining a server certificate involves three basic steps, which are explained in more detail in the following subchapters:

- Create an unsigned certificate with the name of the server you want to secure

- Create a certificate signing request (CSR) from that certificate and send it to a certification authority (CA). The CA first verifies that you really are who you claim to be and then returns a signed version of your certificate to you.

- Replace the unsigned certificate with the signed certificate you got back from the CA.

The administration of the certificates happens with a command line tool called "`keytool`", that is installed together with Java. For more information about this tool, and further discussion about certificates and secure communication, see the documentation of this tool at Oracle's website:

> http://docs.oracle.com/javase/7/docs/technotes/tools/windows/keytool.html
>
> and
>
> http://docs.oracle.com/javase/7/docs/technotes/tools/solaris/keytool.html

Important: The steps described in the following sub-chapters, for obtaining and installing a server certificate, must be executed for each server that is to be secured (i.e. for each server where any of the three LISTSERV Maestro main components LUI, HUB or TRK is installed).

2.2.1 Securing the Trusted Root Certificate Keystore

On each server: As a first step, if you have not already done so, you need to secure the default keystore for trusted root certificates that is shipped with LISTSERV Maestro.

See the section of the same name in "LMA Admin Tech Doc 9 – Securing Access with HTTPS" for details.

2.2.2 Creating an Unsigned Server Certificate

On each server: As a first step to get a server certificate, you need to create an unsigned certificate with the name of this server.

See the section of the same name in "LMA Admin Tech Doc 9 – Securing Access with HTTPS" for details.

2.2.3 Performing a Certificate Signing Request (CSR)

On each server: As the second step, you need to generate a certificate signing request (CSR) from the unsigned server certificate and submit it to a CA of your choice.

See the section of the same name in "LMA Admin Tech Doc 9 – Securing Access with HTTPS" for details.

2.2.4 Installing the Signed Server Certificate

On each server: As the third step, you need to replace the unsigned version of the server certificate which you have in your keystore file with the signed version that you received back from your CA.

See the section of the same name in "LMA Admin Tech Doc 9 – Securing Access with HTTPS" for details.

2.3 Enabling Secure Inter-Component Communication

After you have successfully obtained and installed server certificates for all LISTSERV Maestro servers, you need to perform one final administration step to actually enable secure inter-component communication:

On **each** LISTSERV Maestro server, create an INI-file with the following path and name:

```
[maestro_install_folder]/commands/ssl.ini
```

(The filename is lowercase "`ssl.ini`", i.e. with the lowercase letter "l", as in "SSL").

This file must be a text file according to the LISTSERV Maestro INI-file rules with exactly the following three entries:

```
SSLAllowedClients=CLIENT_LIST
SSLKeystorePath=KEYSTORE_FILE
SSLKeystorePassword=PASSWORD
```

with the following replacements:

`CLIENT_LIST`: A comma separated list of the host names of all LISTSERV Maestro servers of your LISTSERV Maestro installation (this may be one, two or three names, depending on how many servers the LISTSERV Maestro components are distributed on). This list must not contain spaces, linebreaks or anything but the host names and the separating commas. The host names must be the same host names as have been used for the respective server certificates. The name of the server on which this `ssl.ini` file is stored must be included too!

KEYSTORE_FILE: The absolute path to the keystore file (including drive letter) in which the signed server certificate for this server can be found. You cannot use a relative path name but must supply the full path to the file. Remember to use either forward slashes ("/") or double backslashes ("\\") as the folder separator, since in INI-files the single backslash has a special meaning as an escape character (see the rules for INI files described in "LMA Admin Tech Doc 1 – Configuration Files").

PASSWORD: The password of the keystore file.

Security Issue: As you see, the password to the keystore and the certificate therein is included as plain text in this file. This can be a security breach, if unauthorized persons have access to this file. You should therefore employ the appropriate operating system or file system security measures, so that only authorized persons can access this file.

After you have created this file on all LISTSERV Maestro servers, restart LISTSERV Maestro (on all servers) to begin using secure inter-component communication.

2.4 Summary

Secure inter-component communication (via SSL) between the LISTSERV Maestro components is enabled, if:

- Signed server certificates have been obtained and installed for **all** LISTSERV Maestro servers (and if necessary the trusted root certificates have been installed too), as described in section 2.2.
- The `ssl.ini` file has been created on **all** LISTSERV Maestro servers, as described in section 2.3.
- LISTSERV Maestro has been restarted on **all** servers since the above steps have been completed.

The fact that secure inter-component communication is enabled can be seen in the log-files, where the log-file of each component must contain a message among the startup messages like the following one (with "XYZ" being the name of the component):

```
XYZ: Initializing Secure Sockets Layer (SSL) for component communication
```