Building Kaa server from source

This page describes how to build the Kaa server from the source code available on GitHub.

Before building the Kaa server from source, ensure that Oracle JDK 7 and Apache Maven are installed on your machine.

Fetching source code

It is allowed to use any Git client to fetch the Kaa source code from the repository.

Set up your Git configuration (at least the username and email) and download Kaa repository as follows:

```
git clone https://github.com/kaaproject/kaa.git
```

Building Kaa with default Java endpoint SDK

To build the Kaa server with the default Java endpoint SDK into Debian/RPM packages, execute the following command.

NOTE
The Debian build will work correctly on both Linux and Windows operation systems, while the RPM build will work only on Linux operated machines with the RPM tool installed.

NOTE
For the mvn command, the build number and git commit variables are set to emulate Jenkins build variables that are substituted automatically on the build machine.

Building Kaa with C/C++ endpoint SDK and default Java endpoint SDK

To build C/C++ endpoint SDK libraries, ensure all the necessary C/C++ components are installed as described in the C endpoint SDK and C++ endpoint SDK sections.

After the required components are installed, you can build the C/C++ endpoint SDK and all server components by executing the following command.

```
mvn -P compile-client-c,compile-client-cpp,compile-gwt,build-rpm clean install
```

Available maven profiles

Maven profile	Description
build-rpm	As implied in the profile name, it will force to generate .rpm packages. This is useful if you are going to install Kaa on .rpm based Linux distribution (Red Hat Linux, Oracle Linux, etc.)
cassandra-dao	Forces Kaa to use Cassandra NoSQL storage. If none is set MongoDB used by default.
mongo-dao	Forces Kaa to use MongoDB NoSQL storage. Enabled by default.
compile-gwt	Compiles administration user interface. Can be skipped during regular builds.
compile-client-c	Compiles C endpoint. Note, -DskipTests option disables only Java tests. If you want disable C tests too, you should manually comment out test section in client build script.
compile-client-cpp	Compiles C++ endpoint. Note, -DskipTests option disables only Java tests. If you want disable C++ tests too, you should manually comment out test section in client build script.
compile-thrift	Forces Thrift compiler to generate code from thrift files.
cassandra	Compiles Cassandra log appender.
cdap	Compiles CDAP log appender.
couchbase	Compiles Couchbase log appender.

oracle-nosql	Compiles Oracle NoSQL log appender.
jenkins	Forces Kaa to execute integration tests.

Running Kaa

The following script can be used to start all the Kaa components.

```
for x in `cd /etc/init.d ; ls kaa-*` ; do sudo service $x start ; done
```

A similar script, as shown in the following example, can be used to restart or stop all the components.

```
for x in `cd /etc/init.d ; ls kaa-*` ; do sudo service $x restart ; done
for x in `cd /etc/init.d ; ls kaa-*` ; do sudo service $x stop ; done
```