Red Hat JBoss A-MQ 6.1

Installation Guide

Installing Red Hat JBoss A-MQ on a computer

Last Updated: 2017-10-13

Red Hat JBoss A-MQ 6.1 Installation Guide

Installing Red Hat JBoss A-MQ on a computer

JBoss A-MQ Docs Team
Content Services
fuse-docs-support@redhat.com

Legal Notice

Abstract

It is easy to install Red Hat JBoss A-MQ and tailor the installation to a particular environment.

Table of Contents

CHAPTER 1. INSTALLATION PREREQUISITES

Abstract

Before attempting to install and use Red Hat JBoss A-MQ, make sure your system meets the minimum requirements.

1.1. SUPPORTED PLATFORMS

Red Hat tests and supports Fuse products in the configurations listed at Red Hat JBoss A-MQ Supported Configurations.

1.2. JAVA RUNTIME

For details of the Java runtimes supported by Red Hat JBoss A-MQ, see Red Hat JBoss A-MQ Supported Configurations.

1.3. SUPPORTED STANDARDS AND PROTOCOLS

Red Hat JBoss A-MQ supports the standards and protocols listed at Red Hat JBoss A-MQ Supported Standards and Protocols.

1.4. HARDWARE REQUIREMENTS

The minimum hardware specifications to install Red Hat JBoss A-MQ are:

- 300 MB of free disk space
- 2GB of RAM

In addition to the disk space required for the base installation, a running system will require space for caching, persistent message stores, and other functions.

CHAPTER 2. INSTALLING

Abstract

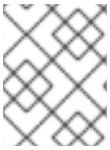
Red Hat JBoss A-MQ is installed by unpacking an archive system on a system. This provides an easy way for a developer to get up and running.

GETTING THE ARCHIVE

You can download the Red Hat A-MQ archive from the Red Hat Customer Portal>Downloads>Red Hat JBoss Middleware>Downloads page, after you register and login to your customer account.

Once logged in:

1. Select **A-MQ**, listed under **Integrated Platforms**, in the sidebar menu.
2. Select **6.1.0** from the **Version** drop-down list on the **Software Downloads** page.
3. Click the **Download** button next to the Red Hat JBoss A-MQ 6.1.0 distribution file you want to download.



NOTE

Trial kits are also available from the developer Web site, at http://jboss.org/products/amq. *These kits are not supported by Red Hat*

UNPACKING THE ARCHIVE

Red Hat JBoss A-MQ is packaged as a **.zip** file. Using a suitable archive tool, such as **Zip** or **Gunzip**, unpack Red Hat JBoss A-MQ into a directory to which you have full access.

WARNING

Do not unpack the archive file into a folder that has spaces in its path name. For example, do not unpack into **C:\Documents and Settings\Greco Roman\Desktop\amq**.

WARNING

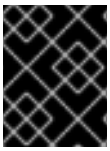
Do not unpack the archive file into a folder that has any of the following special characters in its path name: #, %, ^, ".

CHAPTER 3. ADDING A REMOTE CONSOLE USER

Abstract

The server's remote command console is not configured with a default user. Before you can connect to the server's console remotely, you must add a user to the configuration.

Red Hat JBoss A-MQ is not installed with a default user for the remote console. Before you can remotely manage a server, you must add a user by editing ***InstallDir*/etc/users.properties**.



IMPORTANT

The information in this file is unencrypted so it is not suitable for environments that require strict security.

To add a user:

1. Open ***InstallDir*/etc/users.properties** in your favorite text editor.
2. Locate the line **# admin=admin,admin**.

   This line specifies a user **admin** with the password **admin** and the role **admin**.
3. Uncomment the line by removing the leading #.
4. Change the first **admin** to the desired user name.
5. Change the second **admin** to the desired password.
6. Leave the last **admin** as-is.
7. Save the changes.

CHAPTER 4. VERIFYING THE INSTALLATION

Abstract

Once the installer has completed running it is a good idea to run a simple test to ensure that Red Hat JBoss A-MQ was properly installed.

JBoss A-MQ ships with a simple client utility that can be used to verify that the software was successfully installed. You use it to create a message producer and a message consumer that connect to the broker. If they run successfully, and you see—by checking the Fuse Management Console—that the broker processed the messages, then you can be confident that JBoss A-MQ was installed successfully.

To verify that JBoss A-MQ is properly installed:

1. Log in as the user with ownership permissions for the JBoss A-MQ installation.
2. Open a command shell at ***`InstallDir`***.
3. Start the broker using the **`start`** command.

Windows	**`bin\start.bat`**
Unix	**`./bin/start`**

4. Run the producer client using **`java -jar extras/mq-client.jar producer --user admin --password admin`**.

   The producer will connect to the broker and produce 100 messages. Example 4.1, “Test Producer Output” shows the producer's output.

   Example 4.1. Test Producer Output

   ```
   $ java -jar extras/mq-client.jar producer --user admin --password
   admin
   Using destination: queue://TEST, on broker:
   failover://tcp://localhost:61616
   [org.apache.activemq.transport.failover.FailoverTransport] :
   Successfully connected to tcp://localhost:61616
   [org.fusesource.mq.ProducerThread] : Sent 'test message: 0'
   [org.fusesource.mq.ProducerThread] : Sent 'test message: 1'
   [org.fusesource.mq.ProducerThread] : Sent 'test message: 2'
   [org.fusesource.mq.ProducerThread] : Sent 'test message: 3'
   [org.fusesource.mq.ProducerThread] : Sent 'test message: 4' ...
   [org.fusesource.mq.ProducerThread] : Sent 'test message: 99'
   [org.fusesource.mq.ProducerThread] : Producer thread finished
   Produced: 100
   $
   ```

5. Run the consumer client using **`java -jar extras/mq-client.jar consumer --user admin --password admin`**.

The consumer will connect to the broker and consume 100 messages. Example 4.2, “Test Consumer Output” shows the consumer's output.

Example 4.2. Test Consumer Output

```
$ java -jar extras/mq-client.jar consumer --user admin --password
admin
Using destination: queue://TEST, on broker:
failover://tcp://localhost:61616
[org.apache.activemq.transport.failover.FailoverTransport] :
Successfully connected to tcp://localhost:61616 Waiting for: 100
messages [org.fusesource.mq.ConsumerThread] : Received test
message: 0 [org.fusesource.mq.ConsumerThread] : Received test
message: 1 [org.fusesource.mq.ConsumerThread] : Received test
message: 2 [org.fusesource.mq.ConsumerThread] : Received test
message: 3 [org.fusesource.mq.ConsumerThread] : Received test
message: 4 ... [org.fusesource.mq.ConsumerThread] : Received test
message: 99 [org.fusesource.mq.ConsumerThread] : Consumer thread
finished Consumed: 100 messages
$
```

CHAPTER 5. OFFLINE MODE

Abstract

You can run the JBoss A-MQ container in offline mode (that is, without an Internet connection). But if you are planning to deploy custom applications to the container, it might be necessary to download additional dependencies (to a local Maven repository) before you can run the container in offline mode with these applications.

If you want to run the JBoss A-MQ container in offline mode, it is necessary to distinguish between the following kinds of dependency:

- *Runtime dependencies*—the dependencies required to run the JBoss A-MQ container, in its default configuration.
- *Build-time dependencies*—the dependencies required to build a custom application (which might include third-party libraries).

Here is a summary of what can be done in offline mode and what needs to be done in online mode (with an Internet connection):

- *Running the JBoss A-MQ container in its default configuration*—is supported in offline mode. The default configuration of the JBoss A-MQ container is specified by the **`featuresBoot`** property in the **`etc/org.apache.karaf.features.cfg`** file. The requisite dependencies are all provided in the **`system/`** sub-directory of the installation.
- *Installing additional features*—is, in general, *not* supported in offline mode. In principle, you can use the **`features:install`** command to install any of the features from the standard feature repositories (as specified by the **`featuresRepositories`** property in the **`etc/org.apache.karaf.features.cfg`** file), but the majority of these features must be downloaded from the Internet and are thus not supported in offline mode.
- *Deploying custom applications*—is, in general, *not* supported in offline mode. There may be some cases where an application with a minimal set of build-time dependencies is deployable offline, but in general, custom applications would have third-party dependencies that require an Internet connection (so that JAR files can be downloaded by Apache Maven).

If you do need to deploy an application with dependencies that are not available offline, you can use the Maven dependency plug-in to download the application's dependencies into a Maven offline repository. This customized Maven offline repository can then be distributed internally to any machines that do not have an Internet connection. For more details of this approach, see section "Offline Repository for a Maven Project" in "Fabric Guide".

CHAPTER 6. MONITORING FUSE RESOURCES

Abstract

Using Red Hat JBoss Operations Network (JON), you can discover, import, and monitor Red Hat JBoss A-MQ.

IMPORTANT

The Red Hat JBoss ON 3.2.0 Plugin Pack for Red Hat JBoss A-MQ 6.0 and 6.1 is not installed with the JON base distribution. You must download and install it separately after you have installed JON.

To install the **`Red Hat JBoss ON 3.2.0 Plugin Pack for Red Hat JBoss A-MQ 6.0 and 6.1`**:

1. Navigate to the Red Hat Customer Portal>Downloads>JBoss Enterprise Middleware>Downloads page.
2. Select JBoss ON for Fuse under **`Management`** in the sidebar menu.
3. Make sure **`3.2.0`** appears in the **`Version`** drop-down list on the Software Downloads page.
4. Click **`Download`** next to Red Hat JBoss ON 3.2.0 Plugin Pack for Red Hat JBoss A-MQ 6.0 and 6.1 .
5. Unzip the plugin package (**`JON-3.2.0-Fuse/jon-plugin-pack-fuse-3.2.0.GA.zip`**) into a temporary directory (for example, ***`/installDir/server/`***).
6. Install the jar files in one of two ways:
   - Hot Deploy

     Copy the unpacked jar files into the JBoss Operations Network's ***`/installDir/server/`*****`plugins/`** directory. For details, see the JBoss Operations Network Installation Guide.
   - JON Server UI

     For details, see the JBoss Operations Network Installation Guide , or watch the video Installing JBoss Operations Network - Part 4, Installing JON Agent Plugins .

APPENDIX A. INSTALLING THE APACHE ACTIVEMQ STANDARD DISTRIBUTION

Abstract

Red Hat JBoss A-MQ includes a standard distribution of Apache ActiveMQ in the installation's `extras` directory.

If you simply want to use a standard distribution of Apache ActiveMQ you can use the archived version in the installation's `extras` directory.

The `apache-activemq-5.9.0.redhat-610<build#>.zip` archive file is provided.

You can copy this archive file to the desired location and decompress it using the appropriate utility for your platform.

WARNING

Do not unpack the archive file into a folder that has spaces in its path name. For example, do not unpack into `C:\Documents and Settings\Greco Roman\Desktop\fusesrc`.

APPENDIX B. INSTALLING THE CLIENT PACKAGES

Abstract

Red Hat JBoss A-MQ provides JMS clients as individual packages that you download from the Red Hat Customer Portal.

Table B.1. JMS client packages

Package Name	Description
`jboss-a-mq-cpp-6.1.0.redhat-SNAPSHOT.zip`	Provides the CMS client library for Linux and for Windows. Contains two archives: **`activemq-cpp-library-3.8.2-src.tar.gz`** and **`activemq-cpp-library-3.8.2-src.zip`**.
`jboss-a-mq-nms-6.1.0.redhat-SNAPSHOT.zip`	Provides the .NET Messages client library. Contains two archives: **`Apache.NMS.ActiveMQ-1.6.2-src.zip`** and **`Apache.NMS.ActiveMQ-1.6.2-bin.zip`**.
`jboss-a-mq-amqp-client-6.1.0.redhat-SNAPSHOT.zip`	Provides the Qpid client API. Contains three jar files: **`qpid-amqp-1-0-client-0.26.jar`**, **`qpid-amqp-1-0-client-jms-0.26.jar`**, and **`qpid-amqp-1-0-common-0.26.jar`**.

Copy the archive files to the desired location and decompress them using the appropriate utility for your platform.

After extracting the archives, you can distribute their contents with your client applications that need them. In the case of CMS, you need to build the C++ library from source and include that in your CMS application.

WARNING

Do not unpack the archive files into a folder that has spaces in its path name. For example, do not unpack into **`C:\Documents and Settings\Greco Roman\Desktop\fusesrc`**.

APPENDIX C. RED HAT JBOSS FUSE MAVEN REPOSITORIES

Abstract

Red Hat JBoss Fuse and Red Hat JBoss A-MQ strongly support Maven, an open source build system available from Apache Maven. To use Maven to build your projects, you need to specify, in a Maven `settings.xml` file, where required artifacts are located.

For details on setting up Maven to work with Red Hat JBoss Fuse, see *Building with Maven* in *Red Hat JBoss Fuse Deploying into the Container* on the Red Hat Customer Portal .

The following repositories contain artifacts your projects may need:

- JBoss Fuse repository

  Provides access to the artifacts in the Red Hat JBoss Fuse Maven repository. This repository is required.

  `https://repo.fusesource.com/nexus/content/groups/public/`

- JBoss Fuse Snapshots repository

  Provides access to the artifacts in the Red Hat JBoss Fuse snapshot kit. This repository is not required, but it is useful if you want to experiment with building your project using different versions of the supplied archetypes.

  `https://repo.fusesource.com/nexus/content/groups/public-snapshots/`

- Apache Public repository

  Provides access to the artifacts in the Apache Maven repository. This repository is not required, but it is useful to include as Red Hat JBoss Fuse depends on various Apache artifacts.

  `https://repository.apache.org/content/groups/public/`