



Red Hat AMQ 7.1

AMQ Broker 7.1 Release Notes

Release Notes for AMQ Broker

Red Hat AMQ 7.1 AMQ Broker 7.1 Release Notes

Release Notes for AMQ Broker

Legal Notice

Copyright © 2018 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution-Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux ® is the registered trademark of Linus Torvalds in the United States and other countries.

Java ® is a registered trademark of Oracle and/or its affiliates.

XFS ® is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL ® is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js ® is an official trademark of Joyent. Red Hat Software Collections is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack ® Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

Abstract

These release notes contain the latest information about new features, enhancements, fixes, and issues contained in the AMQ Broker 7.1 release.

Table of Contents

CHAPTER 1. FEATURES	3
Persistence	3
Security	3
Performance	3
CHAPTER 2. ENHANCEMENTS	4
CHAPTER 3. RESOLVED ISSUES	6
CHAPTER 4. KNOWN ISSUES	7
CHAPTER 5. IMPORTANT NOTES	10
Upgrading a 7.0 Broker Instance to 7.1	10
Change to the Management Console URL	10
CHAPTER 6. IMPORTANT LINKS	11

CHAPTER 1. FEATURES

The features added in this release, and that were not in previous releases of the AMQ, are outlined below.

Persistence

Failover

Brokers can be fully redundant, allowing a backup broker to take over connections from a primary broker in the event of a failure. Optionally, a shared volume can be configured between the primary and backup broker.

Delivery guarantee

Some messages that a developer sends from his application require a guarantee of delivery guaranteeing that messages get from the application sending them to the application consuming and processing them. If a network link goes down or the broker experiences a power outage while the message is in transit the message broker must resend undelivered messages once it returns online or the network connection is restored.

Zero persistence

Brokers can be configured to not persist any messages for scenarios in which no guarantee is required, but high performance is.

Security

Basic authentication

Brokers can authenticate clients using basic username and password credentials.

Role-based authentication

Brokers can authenticate and authorize clients to specific queues or message addresses based on roles assigned to the client.

SSL-encrypted connections

Connections between clients and broker or a broker and another broker are secure and utilize SSL to encrypt the connections.

Certificate-based authentication

Brokers can authenticate a client connection or a connection from another broker using certificates.

Performance

Dead connection detection

Brokers can detect dead connections when a client has exceeded specific lifespan that can be set either on the client's connection factory or globally on the server.

Slow consumer detection

Brokers can detect slow consumers and disconnect them from the server when this feature is enabled.

Refer to the ActiveMQ Artemis project for additional information:

<http://activemq.apache.org/artemis/index.html>

CHAPTER 2. ENHANCEMENTS

- **ENTMQBR-283 - AMQP - Send & Receive very large messages**

Previously, clients using the AMQP protocol could not receive messages larger than the value set for the `journal-buffer-size` element in the `broker.xml` configuration file. This limit is removed in 7.1. However, because the messages are stored in memory, their size is still limited by the amount of memory available.
- **ENTMQBR-357 - Implement a Kerberos Authentication Mechanism between clients and the broker**

Previously, the broker could not be used for authenticating clients that used Kerberos credentials. The 7.1 version of AMQ Broker allows Kerberos to be used for authentication between clients and brokers.
- **ENTMQBR-427 - limit non-ssl connection hangs up exceeding client(s)**

Previously, unauthenticated clients were not disconnected by the broker. The 7.1 version of AMQ Broker includes the configuration parameter `max-negotiate-time`, which limits the amount of time an unauthenticated client can remain connected.
- **ENTMQBR-432 - create more granular control of management authorisation**

Previously, securing access to the JMX management API was limited to granting users either full access or none at all. The 7.1 version of AMQ Broker allows for more granularity when granting user access to JMX objects and methods. The security section of the configuration now includes `whitelist`, `default-access`, and `role-access` sub-elements. See the section "Setting Role Based Access Control" in the product documentation titled "[Using AMQ Broker](#)".
- **ENTMQBR-483 - Implement large message support for OpenWire**

Previously, clients using the OpenWire protocol could not receive messages that were larger than the value set for the `journal-buffer-size` element in the `broker.xml` configuration file. This limit is removed in 7.1. However, because messages are stored in memory, their size is still limited by the amount of memory available.
- **ENTMQBR-773 - Provide option to disable/hide advisories created from openwire clients**

The broker automatically creates one or more advisory queues when an OpenWire client connected to the broker. Creating a large number of advisory queues can affect system resources. The 7.1 version of AMQ Broker includes two new configuration parameters. The first, `supportAdvisory`, controls the use of advisory messages, and `suppressInternalManagementObjects` controls their display in the console.
- **ENTMQBR-789 - Add support for Ldap role mapping of kerberos authenticated users**

The 7.1 version of AMQ Broker can query an LDAP directory to discover the roles assigned to a Kerberos-authenticated user. Discovered roles can be mapped to roles created in the broker configuration. See the "Security" chapter of the product documentation titled [Using AMQ Broker](#) for more information.
- **ENTMQBR-790 - Add support for authentication to ldap via kerberos for ldapLoginModule**

Previously, the broker could not connect to an LDAP provider by using Kerberos credentials. In the 7.1 version of AMQ Broker, you can enable the broker to use Kerberos by adding configuration to the `login.config` configuration file. See the "Security" chapter of the product documentation titled [Using AMQ Broker](#) for more information.
- **ENTMQBR-811 - Enable the ability to configure maxFramePayloadSize for STOMP over WebSocket**

Previously, when using STOMP client and WebSockets, you could not send messages with a

payload greater than 65 kilobytes in size, which was the default limit. Using 7.1 version of AMQ Broker, you can limit the size of a message directly by adding the `maxWebSocketFrameSize` parameter to the connection string URI and providing the appropriate value in kilobytes.

- **ENTMQBR-827 - Lock down jolokia by default in a-mq**
Previously, the AMQ Console was accessible from any host. In 7.1, the Console can be accessed only from localhost by default. You can change the access rules using the new configuration file `management-access.xml`. See the product documentation titled "[Using AMQ Console](#)" for more information about configuration options.
- **ENTMQBR-898 - [AMQ7, Plugin support] Artemis Plugin Support**
With the 7.1 version of AMQ Broker, you can develop plug-ins that intercept events in the broker for custom handling. Events that a plug-in could intercept include security-related method calls and the creation of queues and addresses.

CHAPTER 3. RESOLVED ISSUES

- **ENTMQBR-533 - Large message support is documented, but supported only with the Core protocol**

Previously, sending and receiving large messages was supported only when you use the Core protocol. The 7.1 version of AMQ Broker also supports large messages when you use the OpenWire and AMQP protocols. Note that the STOMP protocol has its own method for handling large messages, which has been supported since 7.0.
- **ENTMQBR-701 - Clarify reserved characters in AMQ 7**

In prior releases, it was unclear which characters should not be used when creating destination names, addresses, client IDs, and other configuration elements. This information was added to the product documentation titled [Using AMQ Broker](#) under the section "Address and Queue Naming Requirements".
- **ENTMQBR-771 - Persistent messages sent with QPID AMQP JMS client are not reflected as durable by the broker**

Previously, persistent messages sent by a AMQ JMS client were not displayed as durable when browsing messages by using the AMQ Console or JMX method call. This issue is resolved in version 7.1.
- **ENTMQBR-793 - JMS selectors don't work with topic-to-queue diverts (with AMQP)**

Previously, when a divert was used to route messages from a topic to a queue, no messages were consumed after a consumer attached to the target queue using a selector expression. This issue is fixed in version 7.1.
- **ENTMQBR-803 - Remove unsupported examples shipped with AMQ7**

Previous distributions of AMQ included examples from the upstream community that are not supported by Red Hat. The unsupported examples are removed in the AMQ Broker 7.1 release.
- **ENTMQBR-884 - [Hawtio, Artemis panel] after connecting to hawtio "Artemis" panel is not displayed**

Previously, the Artemis tab would not display in the AMQ Console when used in some environments. The only tabs displayed were **Connect** and **Dispatch Router**. This issue is resolved in the 7.1 release.

For additional details about issues resolved in specific releases, see the following articles:

- [AMQ 7 - 7.0.1 Resolved Issues](#)
- [AMQ 7 - 7.0.2 Resolved Issues](#)
- [AMQ 7 - 7.0.3 Resolved Issues](#)

CHAPTER 4. KNOWN ISSUES

- **ENTMQBR-17 - AMQ222117: unable to start cluster connection**
A broker cluster may fail to initialize properly in environments that support IPv6. The failure is due to a `SocketException` that is indicated by the log message `Can't assign requested address`. To work around this issue, set the `java.net.preferIPv4Stack` system property to `true`.
- **ENTMQBR-463 - Attributes in clustering settings has order restrictions, would be nice to either have better error message or simply ignore the order**
Currently the sequence of the elements in the cluster connection configuration has to be in a specific order. The workaround is to adhere to the order in the configuration schema.
- **ENTMQBR-520 - Receiving from address named same as queue bound to other address should not be allowed**
A queue with the same name as an address must only be assigned to address. Creating a queue with the same name as an existing address, but bound to an address with a different name, is an invalid configuration. Doing so can result in incorrect messages being routed to the queue.
- **ENTMQBR-569 - Conversion of IDs from OpenWire to AMQP results in sending IDs as binary**
When communicating cross-protocol from an A-MQ 6 OpenWire client to an AMQP client, additional information is encoded in the application message properties. This is benign information used internally by the broker and can be ignored.
- **ENTMQBR-599 - Define truststore and keystore by artemis cli**
Creating a broker instance by using the `--ssl-key`, `--ssl-key-password`, `--ssl-trust`, and `--ssl-trust-password` parameters does not work. To work around this issue, set the corresponding properties manually in `bootstrap.xml` after creating the broker.
- **ENTMQBR-617 - Cert based authentication doesn't work with AMQP and OpenWire clients**
Authentication based on SSL certificates is not supported for AMQP and OpenWire clients.
- **ENTMQBR-636 - Journal breaks, causing NullPointerException, under perf load (mpt)**
To prevent IO-related issues from occurring when the broker is managing heavy loads, verify that the JVM is allocated with enough memory and heap space. See the section titled "Tuning the VM" in the [Performance Tuning](#) chapter of the ActiveMQ Artemis documentation.
- **ENTMQBR-648 - JMS Openwire client is unable to send messages to queue with defined 'purgeOnNoConsumer' or queue 'filter'**
Using an A-MQ 6 JMS client to send messages to an address that has a queue with `purgeOnNoConsumer` set to `true` fails if the queue has no consumers. It is recommended that you do not set the `purgeOnNoConsumer` option when using A-MQ 6 JMS clients.
- **ENTMQBR-652 - List of known amq-jon-plugin bugs**
This version of `amq-jon-plugin` has known issues with the MBeans for broker and queue.

Issues with the broker MBean:

- Closing a connection throws `java.net.SocketTimeoutException` exception
- `listSessions()` throws `java.lang.ClassCastException`
- Adding address settings throws `java.lang.IllegalArgumentException`
- `getConnectorServices()` operation cannot be found

- `listConsumersAsJSON()` operation cannot be found
- `getDivertNames()` operation cannot be found
- Listing network topology throws `IllegalArgumentException`
- Remove address settings has wrong parameter name

Issues with the queue MBean:

- `expireMessage()` throws argument type mismatch exception
- `listDeliveringMessages()` throws `IllegalArgumentException`
- `listMessages()` throws `java.lang.Exception`
- `moveMessages()` throws `IllegalArgumentException` with error message argument type mismatch
- `removeMessage()` throws `IllegalArgumentException` with error message argument type mismatch
- `removeMessages()` throws exception with error Can't find operation `removeMessage` with 2 arguments
- `retryMessage()` throws argument type mismatch `IllegalArgumentException`
- **ENTMQBR-655 - [AMQP] Unable to send message when populate-validated-user is enabled**
The configuration option `populate-validated-user` is not supported for messages produced using the AMQP protocol.
- **ENTMQBR-742 - JMS Queues are not being auto-deleted for Openwire and AMQP clients**
Addresses and queues with `<auto-delete-addresses>` and `<auto-delete-queues>` set to `true` are not deleted when AMQ OpenWire JMS clients disconnect from them.
- **ENTMQBR-781 - [Error Codes] same Error Code(s) defined in different places with different reasons**
Logging error codes are not unique for all broker log messages. Use the logger name, `org.apache.activemq.artemis.core.server` for example, to identify uniqueness.
- **ENTMQBR-897 - Openwire client/protocol issues with special characters in destination name**
Currently AMQ OpenWire JMS clients cannot access queues and addresses that include the following characters in their name: comma (','), hash ('#'), greater than ('>'), and whitespace.
- **ENTMQBR-927 - broker.xml does not validate against artemis-configuration.xsd**
The `broker.xml` configuration file does not validate against its XML schema.
- **ENTMQBR-940 - NPE thrown by `QuorumVoteMessage.decode()`**
If a broker that is not configured with an `<ha-policy>` is added to a cluster that is using `<replication>` for its policy, the broker is not included during quorum votes taken after a failure within the cluster. Instead, an exception is thrown that the broker is missing an `<ha-policy>`. It is recommended that you configure all brokers in a cluster to use the same `<ha-policy>`.

- **ENTMQBR-943** - [AMQ7, Openwire, Compression] consuming openwire compressed bytemessage throws `java.util.zip.DataFormatException: incorrect header check`
When using the OpenWire protocol to send small, compressed `ByteMessages` that have JMS properties set for a `Queue`, an exception is thrown on the consumer side when it tries to decompress the message. See the Knowledge Base article on the Red Hat Customer Portal for more details: <https://access.redhat.com/solutions/3269061>.
- **ENTMQBR-944** - [A-MQ7, Hawtio, RBAC] User gets no feedback if operation access was denied by RBAC
The console can indicate that an operation attempted by an unauthorized user was successful when it was not.
- **ENTMQBR-945** - Non-persistent messages lost in non-failure scenarios when authorization fails, because delivery mode defaults to asynchronous
Clients may not notice the loss of non-persistent messages. Message loss is often associated with an obvious failure, such as the broker stopping or the storage provider disconnecting. However, the asynchronous behavior of the default delivery mode can lead to the loss of non-persistent messages in other situations too, such as failing authorization or trying to use a queue that is not configured.
- **ENTMQBR-956** - [AMQ7, AMQP, JMX, DeliveringCount] JMX operation `"listDeliveringMessages()"` returns empty string when `deliveringCount > 0`
The JMX operation `listDeliveringMessages()` does not return messages that were consumed in a transaction by an AMQ JMS client.

CHAPTER 5. IMPORTANT NOTES

Upgrading a 7.0 Broker Instance to 7.1

AMQ Broker 7.1 includes configuration files and settings that were not included with previous versions. Upgrading a broker instance to 7.1 requires adding these new files and settings to your existing 7.0.x broker instances. For more information about upgrading to 7.1, see the section "Upgrading a Broker Instance from 7.0.x to 7.1.0" in the product documentation titled [Using AMQ Broker](#).

Change to the Management Console URL

AMQ Broker 7.1 includes a change to the default URL of the Red Hat AMQ Console. In previous versions, the default URL was `http://IP:PORT/hawtio/`. With 7.1, the default URL becomes `http://IP:PORT/console/`. The change to the URL also impacts broker instances that are upgraded to 7.1. For more information about how to access the 7.1 version of the AMQ Console, see the section "Accessing AMQ Console" in the product documentation titled [Using AMQ Console](#).

CHAPTER 6. IMPORTANT LINKS

- [Red Hat AMQ 7 Supported Configurations](#)
- [Red Hat AMQ 7 Component Details](#)
- [Red Hat JBoss AMQ 7.0 Release Notes](#)

Revised on 2018-05-21 15:48:30 EDT