



## **Red Hat AMQ 7.1**

# **AMQ Interconnect 1.1 Release Notes**

Release Notes for AMQ Interconnect



# Red Hat AMQ 7.1 AMQ Interconnect 1.1 Release Notes

---

Release Notes for AMQ Interconnect

## 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 Interconnect 1.1 release.

## Table of Contents

<b>CHAPTER 1. FEATURES</b> .....	<b>3</b>
<b>CHAPTER 2. ENHANCEMENTS</b> .....	<b>4</b>
<b>CHAPTER 3. RESOLVED ISSUES</b> .....	<b>5</b>
<b>CHAPTER 4. KNOWN ISSUES</b> .....	<b>6</b>



# CHAPTER 1. FEATURES

AMQ Interconnect 1.1 represents the initial offering of this component as part of a complete messaging solution. Deploying an interconnect router (or network of routers) provides the following benefits over using brokers alone:

## Operating modes

The interconnect component can operate in stand-alone mode or interior mode.

## Brokerless messaging

Direct, brokerless messaging when broker queueing is not needed. This feature is useful for request-response/RPC patterns.

## Anycast or Multicast

Direct delivery can be configured as multicast (all subscribers receive a copy of each produced message) or anycast (one subscriber receives a copy of each produced message).

## Inexpensive HA and resiliency

High availability and resiliency for the network does not require high-cost clustering; it is achieved through redundant topologies much like one would use in deploying an IP network.

## Scale up queues and topics

A messaging system including interconnect can offer a greater number of queues and topics than can be offered by a single broker or a cluster of brokers.

## Queue/topic distribution

A single queue or topic can be distributed across multiple brokers to provide increased user scale and throughput.

## Security

Access to a broker can be secured, hardened, and limited. In addition, the broker does not need to be deployed in a client-facing DMZ (De-Militarized Zone in front of a firewall).

## Encryption

Connections between clients and a broker or a broker and another broker can be secured by using SSLTLS (Secure Socket Layer Top Level Specification) or SASL (Simple Authentication and Security Layer) at the interconnect level to encrypt the connections.

## Elasticity

Brokers can be added and removed to handle changes in load or to accommodate broker maintenance.

## Multi-tenancy

Queues, topics, and destinations can be partitioned by user/application/account such that multiple users can use the same messaging infrastructure without interfering with each other.

Refer to the Apache Qpid Dispatch Router project for additional information:

<http://qpid.apache.org/components/dispatch-router/index.html>

## CHAPTER 2. ENHANCEMENTS

- **ENTMQIC-1905 - Support for large messages - Message Streaming**

AMQ Interconnect has been enhanced to better support large messages. Previously, when receiving a message, the router waited to receive the entire message before sending it to its receivers. This led to slow throughput for large messages.

Starting in AMQ Interconnect 1.1, the router sends frames to the receivers when they arrive at the router, and releases the memory immediately after sending the buffer to all receivers. This keeps the memory stable on the router even if there are a large number of senders sending large messages. In addition, messages are delivered more quickly to the receivers, thus increasing throughput because frames are sent when they are received and not held until the entire message arrives at the router.



### NOTE

If a fast sender sends messages to a slow receiver, it is possible that memory could pile up on the router.

In addition, multicast streaming is now supported as part of this enhancement.



## CHAPTER 3. RESOLVED ISSUES

- **ENTMQIC-1966 - warning: %postun(qpid-dispatch-router-0.8.0-9.el6.i686) scriptlet failed, exit status 2**

After installing AMQ Interconnect 1.0.1, it was necessary to check the service status and restart the service if it was running:

```
# service qdrouterd status
qdrouterd (pid 3996) is running...

# service qdrouterd restart
Shutting down qdrouterd services: [ OK ]
Starting qdrouterd services: [ OK ]

# service qdrouterd status
qdrouterd (pid 4384) is running...
```

This issue is resolved in the 1.1 release.

## CHAPTER 4. KNOWN ISSUES

- **ENTMQIC-61 - Memory pools are never returned to heap**

Several heavily used data objects (deliveries, messages, links, buffers, etc.) are managed by AMQ Interconnect in pools for efficient allocation. In AMQ Interconnect 1.1, objects in these pools are not returned to the heap at any time. This means that the memory used in large bursts of activity will not be freed, but will remain available for use thereafter.

This might be observed as an increase in memory usage that does not decrease after a burst of activity is completed. Subsequent bursts of activity will use the same memory that was used previously.

Methods of returning large amounts of pooled objects back to the heap are being developed.

- **ENTMQIC-1979 - Kerberos authentication problem with amq-jms client**

Kerberos (GSSAPI) authentication between the AMQ JMS client and the AMQ Interconnect router does not work and is not supported in this release.

- **ENTMQIC-1980 - Symbolic ports in HTTP listeners do not work**

When configuring a listener in the router with the `http` option enabled (for console or WebSocket access), the `port` attribute must be expressed numerically. Symbolic port names do not work with HTTP listeners.

If a listener is configured as:

```
listener {
    ...
    port: amqp
    http: yes
    ...
}
```

It should be changed to:

```
listener {
    ...
    port: 5672
    http: yes
    ...
}
```

- **ENTMQIC-1985 - Management agent misreports the SSL/TLS version**

When using SSL/TLS in a connection, the management agent reports the protocol version to be **TLSv1/SSLv3** when the protocol actually being used on the wire is **TLSv1.2**.

This is an issue in the underlying **qpidd-proton** library and is fixed in **qpidd-proton 0.20**.

*Revised on 2018-05-21 15:51:26 EDT*