Red Hat AMQ 7.2

AMQ Interconnect 1.3 Release Notes

Release Notes for AMQ Interconnect

Red Hat AMQ 7.2 AMQ Interconnect 1.3 Release Notes

Release Notes for AMQ Interconnect

Legal Notice

Abstract

These release notes contain the latest information about new features, enhancements, fixes, and issues contained in the AMQ Interconnect 1.3 release.

Table of Contents

CHAPTER 1. FEATURES

AMQ Interconnect 1.3 is 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, interior mode, and as of version 1.3, edge 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-1993 - Allow policy for sources and targets to handle multiple wildcards**
  The syntax for vhost policies has been improved in this version of AMQ Interconnect. You can now use multiple wildcard characters in source and target patterns.

- **ENTMQIC-1990 - Allow restriction of TLS and SSL protocol versions to be used in connections**
  In this version of AMQ Interconnect, you can now specify the SSL/TLS protocol version to be used in connections. You can use this capability to block versions of the protocol that have been shown to have security vulnerabilities.

- **ENTMQIC-1988 - Enhanced statistics available from router**
  This version of AMQ Interconnect adds several new statistics to the router management schema. You can now view more detailed statistics, including rolled-up, per-router statistics.

- **ENTMQIC-1987 - Edge router mode for improved scale and efficiency**
  AMQ Interconnect 1.3 introduces a new "edge" router operating mode. You can now create router networks containing both edge and interior routers in which clients connect to the edge routers, and the edge routers connect to the interior routers. This topology enables you to scale the router network to thousands of individual routers without suffering the memory use issues, route computational load, and delivery disruption that could occur in router networks containing large numbers of interior routers.

CHAPTER 3. TECHNOLOGY PREVIEWS

- In AMQ Interconnect 1.2, exchange/binding-style message routing is available as a Technology Preview feature only.

  Technology Preview features are not supported with Red Hat production service-level agreements (SLAs) and might not be functionally complete. Red Hat does not recommend using them for production. For more information, see Red Hat Technology Preview Features Support Scope.

CHAPTER 4. RESOLVED ISSUES

- **ENTMQIC-1985 - Management agent misreports the SSL/TLS version**
  In previous versions of AMQ Interconnect, when using SSL/TLS in a connection, the management agent reported the protocol version to be `TLSv1/SSLv3` when the protocol actually being used on the wire was `TLSv1.2`.

  This was an issue in the underlying `qpid-proton` library and was fixed in `qpid-proton` 0.20.

- **ENTMQIC-1997 - Router crash when executing `pn_class_incref ()` in `libqpid-proton-core.so.10`**
  In previous versions of AMQ Interconnect, a defect caused the router to crash under certain load conditions. This issue has been corrected, and the crashes are no longer seen.

- **ENTMQIC-1929 - qdmanage tool filtering by "type" is shady**
  When using the `qdmanage` tool, you can now use the `--type` argument to operate on a particular type of entity. Previously, `qdmanage` ignored `--type` for certain commands.

- **ENTMQIC-1978 - cannot set address phase via qdmanage tool**
  Previously, if you used the `qdmanage` tool to create addresses or auto-links with a non-default phase, the phase number was not correctly interpreted by the tool. This has been corrected and it is now possible to use `qdmanage` to configure non-default uses of address phase.

- **ENTMQIC-1986 - Detach not echoed back in multi-hop link route due to wrong session being closed**
  Previously, link-routed links would sometimes fail to close completely under certain load conditions. This defect has been corrected, and routed-links now close cleanly and completely.

- **ENTMQIC-2000 - enable sasl plugin for websocket connections**
  Previously, authentication was not usable in HTTP/Websocket listeners. This defect has been corrected, and you can now configure SASL client authentication for HTTP/Websocket listeners.

- **ENTMQIC-2004 - Transactional behaviour different in client router broker scenario versus client broker**
  Previously, if a link was repeatedly drained and then restarted, flow-control credit would not always be accounted correctly (particularly when using the AMQ JMS client). This issue has been corrected, and credit is now correctly tracked over multiple drain-restart cycles.

- **ENTMQIC-1992 - Enabling AMQ 7.1 Broker failback breaks failover support**
  Previously, if a router was connected to the master instance of a broker cluster, and then failed over to the backup instance, it was possible for the router to lose track of the master instance and fail to reconnect to it after it was back online. This issue has been fixed to ensure that a router can always reconnect to the original master instance in a broker cluster.

- **ENTMQIC-2003 - Allow address translation on link routes**
  In this version of AMQ Interconnect, you can now use address translation for link routes. The `addExternalPrefix` and `delExternalPrefix` configuration attributes have been added, which enable you to add or remove a prefix in the address to form a different address for the route-container than the address used in the router network.

- **ENTMQIC-2023 - `qdrouterd` accumulates memory on connections with link attach+detach loop**
  In previous versions of AMQ Interconnect, a defect caused memory usage to build up when a long-lived connection carried many link attaches and detaches. The memory for the links was not freed until the connection was closed. This defect has been corrected, and link memory is

reclaimed when the link is closed.

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

- AMQ Interconnect 1.2.2 Resolved Issues

- AMQ Interconnect 1.2.1 Resolved Issues

CHAPTER 5. 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-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
      ...
  }
  ```

Revised on 2019-01-17 17:10:58 UTC