Release Notes for Red Hat AMQ Clients
Abstract

These release notes contain the latest information about new features, enhancements, fixes, and issues contained in the AMQ Clients 2.8 release.
Table of Contents

CHAPTER 1. NEW AND CHANGED FEATURES ......................................................... 3
  1.1. AMQ PYTHON ................................................................. 3
  1.2. AMQ JMS ............................................................... 3
  1.3. AMQ SPRING BOOT STARTER ........................................ 3
  1.4. AMQ RESOURCE ADAPTER ........................................... 3

CHAPTER 2. FIXED ISSUES ............................................................................ 4
  2.1. AMQ JMS .................................................. 4
  2.2. AMQ RESOURCE ADAPTER ........................................... 4

CHAPTER 3. IMPORTANT NOTES .................................................................. 5
  3.1. AMQ C++ ............................................................... 5
  3.2. PREFERRED CLIENTS ..................................................... 5
  3.3. LEGACY CLIENTS ......................................................... 6
  3.4. UPSTREAM VERSIONS ..................................................... 6

CHAPTER 4. IMPORTANT LINKS ................................................................. 7
CHAPTER 1. NEW AND CHANGED FEATURES

1.1. AMQ PYTHON

- AMQ Python is now supported on Windows with Python 3.8, in addition to Python 3.6.

1.2. AMQ JMS

- If an AMQ JMS connection is lost, more detailed logging is now produced when the connection is restored.

1.3. AMQ SPRING BOOT STARTER

- AMQ Spring Boot Starter now supports Spring Boot 2.3.

1.4. AMQ RESOURCE ADAPTER

- AMQ Resource Adapter now includes an example using the WildFly application server, replacing the previous Thorntail example.
CHAPTER 2. FIXED ISSUES

2.1. AMQ JMS

- **CVE-2020-9488** - Improper validation of certificate with host mismatch in Apache Log4j SMTP appender
  This issue was resolved by updating the client’s Log4j dependency.

2.2. AMQ RESOURCE ADAPTER

- **CVE-2020-14297, CVE-2020-14307, CVE-2020-11113** - Vulnerabilities in the example code
  In earlier releases of the product, AMQ Resource Adapter contained an example program subject to the listed vulnerabilities.

  In this release, the vulnerabilities are addressed in a new example program.
CHAPTER 3. IMPORTANT NOTES

3.1. AMQ C++

- **Unsettled interfaces**
  The AMQ C++ messaging API includes classes and methods that are not yet proven and can change in future releases. Be aware that use of these interfaces might require changes to your application code in the future.

  These interfaces are marked *Unsettled API* in the API reference. They include the interfaces in the `proton::codec` and `proton::io` namespaces and the following interfaces in the `proton` namespace.

  - `listen_handler`
  - The `on_sender_drain_start` and `on_sender_drain_finish` methods on `messaging_handler`
  - The `draining` and `return_credit` methods on `sender`
  - The `draining` and `drain` methods on `receiver`

  API elements present in header files but not yet documented are considered unsettled and are subject to change.

- **Deprecated interfaces**
  Interfaces marked *Deprecated* in the API reference are scheduled for removal in a future release.

  This release deprecates the following interfaces in the `proton` namespace.

  - `void_function0` - Use the `work` class or C++11 lambdas instead.
  - `default_container` - Use the `container` class instead.
  - `url` and `url_error` - Use a third-party URL library instead.

3.2. PREFERRED CLIENTS

In general, AMQ clients that support the AMQP 1.0 standard are preferred for new application development. However, the following exceptions apply:

- If your implementation requires distributed transactions, use the AMQ Core Protocol JMS client.
- If you require MQTT or STOMP in your domain (for IoT applications, for instance), use community-supported MQTT or STOMP clients.

The considerations above do not necessarily apply if you are already using:

- The AMQ OpenWire JMS client (the JMS implementation previously provided in A-MQ 6)
- The AMQ Core Protocol JMS client (the JMS implementation previously provided with HornetQ)
3.3. LEGACY CLIENTS

- **Deprecation of the CMS and NMS APIs**
  The ActiveMQ CMS and NMS messaging APIs are deprecated in AMQ 7. It is recommended that users of the CMS API migrate to AMQ C++, and users of the NMS API migrate to AMQ .NET. The CMS and NMS APIs might have reduced functionality in AMQ 7.

- **Deprecation of the legacy AMQ C++ client**
  The legacy AMQ C++ client (the C++ client previously provided in MRG Messaging) is deprecated in AMQ 7. It is recommended that users of this API migrate to AMQ C++.

- **The Core API is unsupported**
  The Artemis Core API client is not supported. This client is distinct from the AMQ Core Protocol JMS client, which is supported.

3.4. UPSTREAM VERSIONS

- AMQ C++, AMQ Python, and AMQ Ruby are now based on Qpid Proton 0.32.0.
- AMQ JavaScript is now based on Rhea 1.0.24.
- AMQ .NET is now based on AMQP.Net Lite 2.4.0.
- AMQ JMS is now based on Qpid JMS 0.53.0.
- AMQ Core Protocol JMS is now based on ActiveMQ Artemis 2.11.0.
- AMQ OpenWire JMS is now based on ActiveMQ 5.11.0.
- AMQ JMS Pool is now based on Pooled JMS 1.1.1.
- AMQ Resource Adapter is now based on AMQP 1.0 Resource Adapter 1.0.2.
- AMQ Spring Boot Starter is now based on AMQP 1.0 JMS Spring Boot 2.3.3.
- AMQ Netty OpenSSL is now based on netty-tcnative 2.0.31.Final.
CHAPTER 4. IMPORTANT LINKS

- Red Hat AMQ 7 Supported Configurations
- Red Hat AMQ 7 Component Details
- AMQ Clients 2.7 Release Notes
- AMQ Clients 2.6 Release Notes
- AMQ Clients 2.5 Release Notes
- AMQ Clients 2.4 Release Notes
- AMQ Clients 2.3 Release Notes
- AMQ Clients 2.2 Release Notes
- AMQ Clients 2.1 Release Notes
- AMQ Clients 2.0 Release Notes
- AMQ Clients 1.2 Release Notes
- AMQ Clients 1.1 Release Notes

Revised on 2020-10-08 11:29:42 UTC