Red Hat AMQ 7.2

Evaluating AMQ Online on OpenShift Container Platform

For use with AMQ Online 1.0

Last Updated: 2019-03-11

Red Hat AMQ 7.2 Evaluating AMQ Online on OpenShift Container Platform

For use with AMQ Online 1.0

Legal Notice

Abstract

This guide describes how to install and manage AMQ Online to evaluate its potential use in a production environment.

Table of Contents

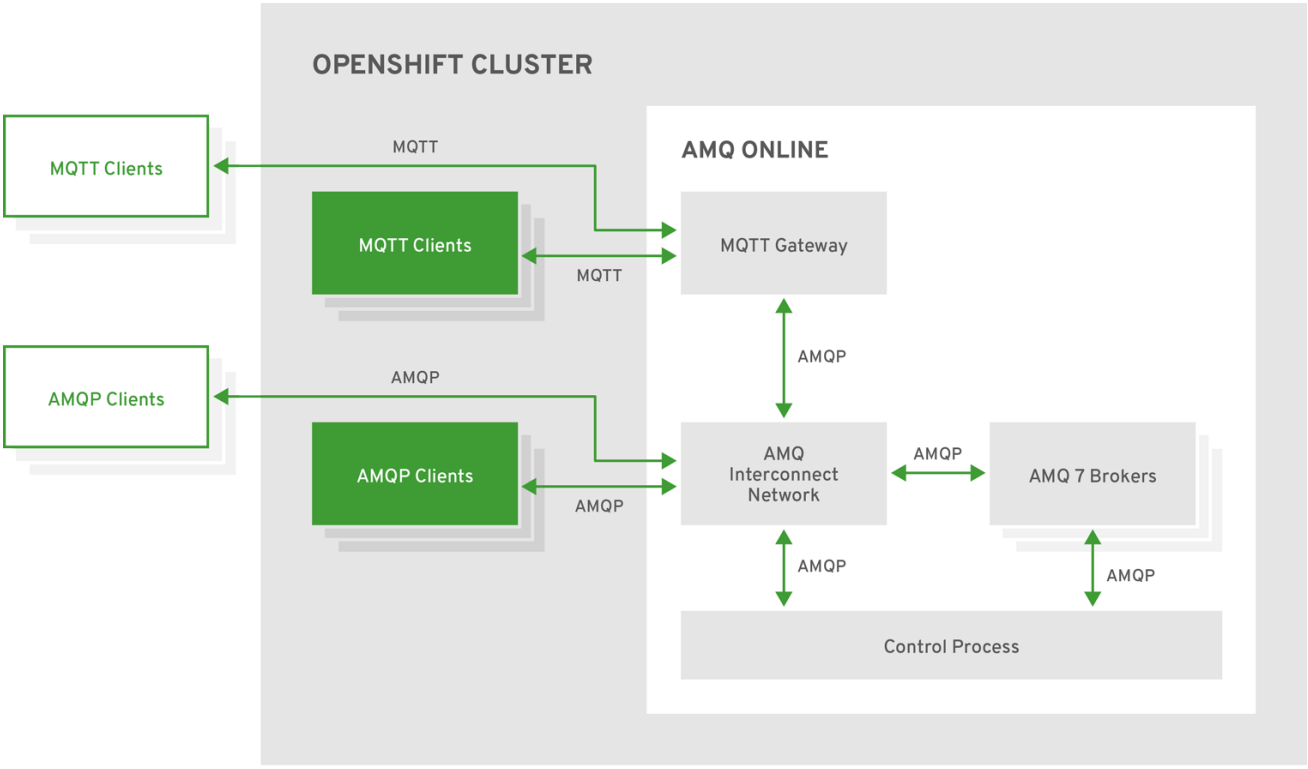
CHAPTER 1. INTRODUCTION

1.1. AMQ ONLINE OVERVIEW

Red Hat AMQ Online is an OpenShift-based mechanism for delivering messaging as a managed service. With Red Hat AMQ Online, administrators can configure a cloud-native, multi-tenant messaging service either in the cloud or on premise. Developers can provision messaging using the AMQ Online console. Multiple development teams can provision the brokers and queues from the console, without requiring each team to install, configure, deploy, maintain, or patch any software.

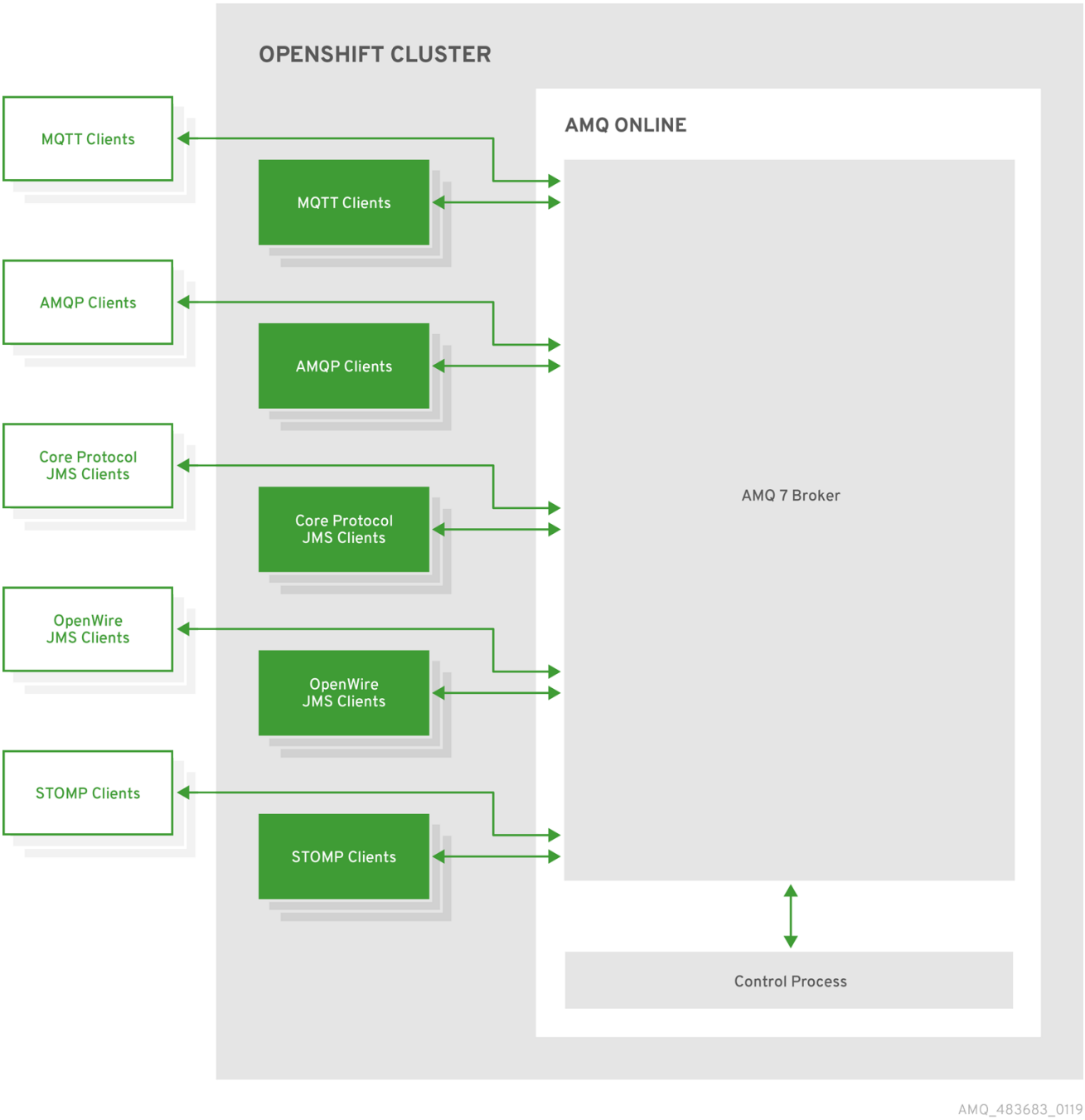
AMQ Online can provision different types of messaging depending on your use case. A user can request messaging resources by creating an address space. AMQ Online currently supports two address space types, standard and brokered, each with different semantics. The following diagrams illustrate the high-level architecture of each address space type:

Figure 1.1. Standard address space



AMQ_483683_0119

Figure 1.2. Brokered address space



1.2. SUPPORTED FEATURES

The following table shows the supported features for AMQ Online 1.0:

Table 1.1. Supported features reference table

Feature		Brokered address space	Standard address space
Address type	Queue	Yes	Yes
	Topic	Yes	Yes

Feature		Brokered address space	Standard address space
	Multicast	No	Yes
	Anycast	No	Yes
	Subscription	No	Yes
Messaging protocol	AMQP	Yes	Yes
	MQTT	Yes	Technology preview only
	CORE	Yes	No
	OpenWire	Yes	No
	STOMP	Yes	No
Transports	TCP	Yes	Yes
	WebSocket	Yes	Yes
Durable subscriptions	JMS durable subscriptions	Yes	No
	"Named" durable subscriptions	No	Yes
JMS	Transaction support	Yes	No
	Selectors on queues	Yes	No
	Message ordering guarantees (including prioritization)	Yes	No
Scalability	Scalable distributed queues and topics	No	Yes

1.3. SUPPORT STATEMENT AND RESOURCES

The AMQ Online 1.0 release is provided as a service to customers who want to try the AMQ Online features and work with support when problems are encountered. Support for Beta releases is limited to commercially reasonable effort and non-production use cases, and all support cases must be opened with a severity of 4. Patches will not be provided, but bug fixes might be incorporated in future releases. To contact support, visit Open a Support Case.

1.4. SUPPORTED CONFIGURATIONS

For more information about AMQ Online supported configurations see Red Hat AMQ 7 Supported Configurations.

1.5. DOCUMENT CONVENTIONS

1.5.1. Variable text

This document contains code blocks with variables that you must replace with values specific to your installation. In this document, such text is styled as italic monospace.

For example, in the following code block, replace ***my-namespace*** with the namespace used in your installation:

```
sed -i 's/amq-online-infra/my-namespace/' install/bundles/enmasse-with-
standard-authservice/*.yaml
```

CHAPTER 2. GETTING STARTED

This guide will walk through the process of setting up AMQ Online on OpenShift with clients for sending and receiving messages.

Prerequisites

- To install AMQ Online, the OpenShift client tools are required.
- An OpenShift cluster is required.
- A user on the OpenShift cluster with **`cluster-admin`** permissions is required to set up the required cluster roles and API services.

2.1. DOWNLOADING AMQ ONLINE

Procedure

- Download and extract the **`amq-online-install.zip`** file from the AMQ Online download site.

NOTE

Although container images for AMQ Online are available in the Red Hat Container Catalog, we recommend that you use the YAML files provided instead.

2.2. INSTALLING AMQ ONLINE USING A YAML BUNDLE

The simplest way to install AMQ Online is to use the predefined YAML bundles.

Procedure

1. Log in as a user with **`cluster-admin`** privileges:

```
oc login -u system:admin
```

2. (Optional) If you want to deploy to a namespace other than **`amq-online-infra`** you must run the following command and substitute **`amq-online-infra`** in subsequent steps:

```
sed -i 's/amq-online-infra/my-namespace/' install/bundles/amq-
online/*.yaml
```

3. Create the project where you want to deploy AMQ Online:

```
oc new-project amq-online-infra
```

4. Deploy using the **`amq-online`** bundle:

```
oc apply -f install/bundles/amq-online
```

2.3. CREATING ADDRESS SPACES USING THE COMMAND LINE

In AMQ Online, you create address spaces using standard command-line tools.

Procedure

1. Log in as a messaging tenant:

```
oc login -u developer
```

2. Create the project for the messaging application:

```
oc new-project myapp
```

3. Create an address space definition:

```
apiVersion: enmasse.io/v1beta1
kind: AddressSpace
metadata:
  name: myspace
spec:
  type: standard
  plan: standard-unlimited
```

4. Create the address space:

```
oc create -f standard-address-space.yaml
```

5. Check the status of the address space:

```
oc get addressspace myspace -o jsonpath={.status.isReady}
```

The address space is ready for use when the previous command outputs **`true`**.

2.4. CREATING ADDRESSES USING THE COMMAND LINE

You can create addresses using the command line.

Procedure

1. Create an address definition:

```
apiVersion: enmasse.io/v1beta1
kind: Address
metadata:
    name: myspace.myqueue
spec:
    address: myqueue
    type: queue
    plan: standard-small-queue
```

NOTE

Prefixing the name with the address space name is required to ensure addresses from different address spaces do not collide.

2. Create the address:

```
oc create -f standard-small-queue.yaml
```

3. List the addresses:

```
oc get addresses -o yaml
```

2.5. CREATING USERS USING THE COMMAND LINE

In AMQ Online users can be created using standard command-line tools.

Prerequisites

- You must have already created an address space.

Procedure

1. Save the user definition to a file:

```
apiVersion: user.enmasse.io/v1beta1
kind: MessagingUser
metadata:
  name: myspace.user1
spec:
  username: user1
  authentication:
    type: password
    password: cGFzc3dvcmQ= # Base64 encoded
  authorization:
    - addresses: ["myqueue", "queue1", "queue2", "topic*"]
      operations: ["send", "recv"]
    - addresses: ["anycast1"]
      operations: ["send"]
```

2. Create the user and associated user permissions:

```
oc create -f user-example1.yaml
```

3. Confirm that the user was created:

```
oc get messagingusers
```

2.6. SENDING AND RECEIVING MESSAGES

Prerequisites

- Installed Apache Qpid Proton Python bindings.
- An address space named **myspace** must be created.
- An address named **myqueue** must be created.
- A user named **user1** with password **password** must be created.

Procedure

1. Save Python client example to a file:

```
from __future__ import print_function, unicode_literals
import optparse
from proton import Message
from proton.handlers import MessagingHandler
from proton.reactor import Container

class HelloWorld(MessagingHandler):
    def __init__(self, url):
        super(HelloWorld, self).__init__()
        self.url = url

    def on_start(self, event):
        event.container.create_receiver(self.url)
        event.container.create_sender(self.url)

    def on_sendable(self, event):
        event.sender.send(Message(body="Hello World!"))
        event.sender.close()

    def on_message(self, event):
        print("Received: " + event.message.body)
        event.connection.close()

parser = optparse.OptionParser(usage="usage: %prog [options]")
parser.add_option("-u", "--url",
default="amqps://localhost:5672/myqueue",
                  help="url to use for sending and receiving
messages")
opts, args = parser.parse_args()

try:
    Container(HelloWorld(opts.url)).run()
except KeyboardInterrupt: pass
```

2. Retrieve the address space messaging endpoint host name:

```
oc get addressspace myspace -o 'jsonpath={.status.endpointStatuses[?
(@.name=="messaging")].externalHost}'
```

   Use the output as the host name in the following step.

3. Run the client:

```
python client-example1.py -u
amqps://user1:password@messaging.example1.com:443/myqueue
```

We have seen how to set up AMQ Online on OpenShift, and how to communicate with it using an AMQP client.

APPENDIX A. USING YOUR SUBSCRIPTION

AMQ Online is provided through a software subscription. To manage your subscriptions, access your account at the Red Hat Customer Portal.

Accessing your account

1. Go to access.redhat.com.
2. If you do not already have an account, create one.
3. Log in to your account.

Activating a subscription

1. Go to access.redhat.com.
2. Navigate to **My Subscriptions**.
3. Navigate to **Activate a subscription** and enter your 16-digit activation number.

Downloading zip and tar files

To access zip or tar files, use the Red Hat Customer Portal to find the relevant files for download. If you are using RPM packages, this step is not required.

1. Open a browser and log in to the Red Hat Customer Portal **Product Downloads** page at access.redhat.com/downloads.
2. Locate the **Red Hat AMQ Online** entries in the **JBOSS INTEGRATION AND AUTOMATION** category.
3. Select the desired AMQ Online product. The Software Downloads page opens.
4. Click the **Download** link for your component.

Registering your system for packages

To install RPM packages on Red Hat Enterprise Linux, your system must be registered. If you are using zip or tar files, this step is not required.

1. Go to access.redhat.com.
2. Navigate to **Registration Assistant**.
3. Select your OS version and continue to the next page.
4. Use the listed command in your system terminal to complete the registration.

To learn more see How to Register and Subscribe a System to the Red Hat Customer Portal.

Revised on 2019-03-11 19:09:49 UTC