MOBIA

MOBIA is a business technology integrator with over thirty years of experience and 500+ employees across Canada and USA. Our talented bench of technical engineers and trusted advisors deliver process improvements and business transformations within our core pillars of **Hybrid Infrastructure Solutions**, **Digital Transformation Services**, **Managed Services**, **Cybersecurity** and **Broadband** & **Wireless Services**.

Our inside-out approach allows us to understand business challenges from deep within the organization, mapping the impact as it ripples outward. This insight enables us to create future-proof solutions that maximize results and repeatedly exceed our client's expectations.

FirstOntario JBOSS Cluster Documentation

June 18, 2021

Rama Nayak, Delivery Consultant
Rama.Nayak@MOBIA.io

Ravindra Maurya, Team Lead, Delivery Consultant
Ravindra.Maurya@MOBIA.io

Nevin Pick, MOBIA Director of Digital Transformation Services
Nevin.Pick@MOBIA.io

Table of Contents

01 Project Summary

01.1 In Scope

- **EAP Cluster**
  - Install EAP on 2 Nodes
  - Configure a EAP TCP Cluster
  - Demonstrate Working Cluster using Demo Applications

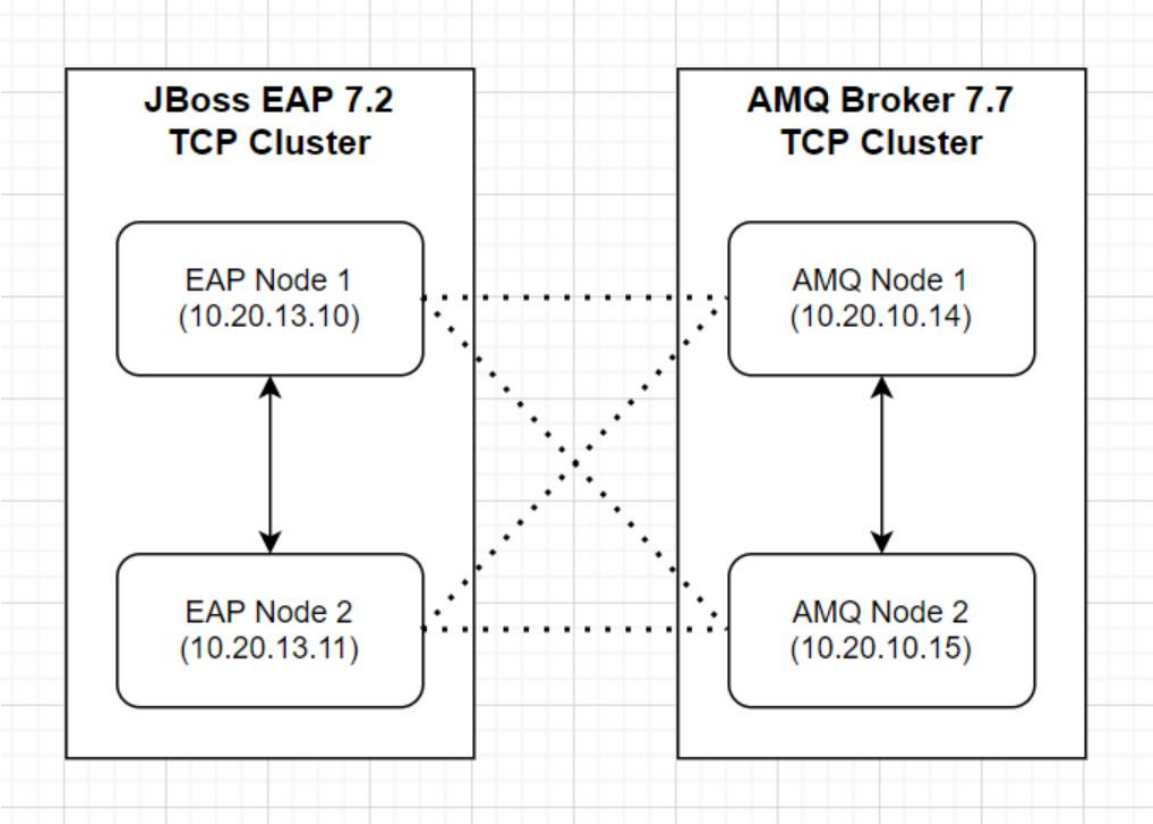
- **AMQ Cluster**
  - Install AMQ Messaging Server on 2 Nodes
  - Configure a AMQ TCP Cluster
  - Demonstrate Working Cluster
  - Integrate EAP & AMQ for JBoss Application
  - Demonstrate Working Messaging using Demo Applications

01.2 Out of Scope

- Deployment of Temenos Application over EAP Cluster
- Integration of Non-JBoss Applications with AMQ Messaging

01.3 Architecture

Based on the below architecture diagram, we have a 2 Node JBoss EAP TCP cluster which directly connects to another 2 Node AMQ Broker TCP Cluster for sending or receiving messages.



02 Prerequisites

Ensure that your environment meets the below requirements before you start implementing the architecture.

- Java JDK v1.8 or Higher
- JBoss EAP v7.2 Binaries
- AMQ Broker v7.7 Binaries

03 AMQ Cluster Plan

03.1 Broker setup

- Extract the **amq-broker-7.7.0.zip** file to the preferred installation location.

```
[rnayak@FCU-R20QAAMQ01 ~]$ cd /eap/AMQ_CLUSTER/amq-broker-7.7.0
[rnayak@FCU-R20QAAMQ01 amq-broker-7.7.0]$ ls
bin  docs  etc  examples  lib  LICENSE  NOTICE  README.html  schema  web
[rnayak@FCU-R20QAAMQ01 amq-broker-7.7.0]$
```

- Create a new Broker:

https://access.redhat.com/documentation/en-us/red_hat_amq/7.7/html/getting_started_with_amq_broker/creating-standalone-getting-started

```
[rnayak@FCU-R20QAAMQ01 bin]$ sudo su jboss-user
[jboss-user@FCU-R20QAAMQ01 bin]$ pwd
/eap/AMQ_CLUSTER/amq-broker-7.7.0/bin
[jboss-user@FCU-R20QAAMQ01 bin]$ ls
artemis  artemis.cmd  broker  lib
[jboss-user@FCU-R20QAAMQ01 bin]$ ./artemis create broker \
    --user admin \
    --password Admin123$ \
    --host `hostname -I` \
    --http-host `hostname -I` \
    --allow-anonymous
Creating ActiveMQ Artemis instance at: /eap/AMQ_CLUSTER/amq-broker-7.7.0/bin/broker
::::::::::::      You, seconds ago • Uncommitted changes
You can now start the broker by executing:
   "/eap/AMQ_CLUSTER/amq-broker-7.7.0/bin/broker/bin/artemis" run
Or you can run the broker in the background using:
   "/eap/AMQ_CLUSTER/amq-broker-7.7.0/bin/broker/bin/artemis-service" start
```

- Once Broker is created, we can edit the **Broker's configuration files** to control Broker operations.

```
[jboss-user@FCU-R20QAAMQ01 amq-broker-7.7.0]$ cd bin/
[jboss-user@FCU-R20QAAMQ01 bin]$ ls
artemis  artemis.cmd  broker  lib
[jboss-user@FCU-R20QAAMQ01 bin]$ ls broker
bin  data  etc  lib  lock  log  tmp
[jboss-user@FCU-R20QAAMQ01 bin]$ ls broker/etc/
artemis.profile          artemis-users.properties  broker.xml          logging.properties  management.xml
artemis-roles.properties  bootstrap.xml            jolokia-access.xml  login.config
[jboss-user@FCU-R20QAAMQ01 bin]$
```

- For Clustering and Queue creation etc., we can configure the default "**broker.xml**" file by appending features to it.

  Clustering: -
  https://access.redhat.com/documentation/en-us/red_hat_amq/7.7/html/configuring_amq_broker/setting-up-broker-cluster-configuring

  A cluster consists of multiple broker instances that have been grouped together. Broker clusters enhance performance by distributing the message processing load across multiple brokers. You can create a broker cluster by specifying a static list of brokers. This is called **static discovery**.

  A broker has one or more **connectors** that define how the broker can connect to other brokers in the cluster. A broker has one or more **cluster connections** which load balances messages for all addresses.

```
<core xmlns="urn:activemq:core" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:schemaLocation="urn:activemq:core ">

   <name>10.20.10.14</name>

   <connectors>
      <connector name="netty-connector">tcp://10.20.10.14:61616</connector>
      <connector name="broker2-connector">tcp://10.20.10.15:61616</connector>
   </connectors>
   <cluster-connections>
      <cluster-connection name="amq-cluster">
         <connector-ref>netty-connector</connector-ref>
         <retry-interval>500</retry-interval>
         <use-duplicate-detection>true</use-duplicate-detection>
         <message-load-balancing>STRICT</message-load-balancing>
         <max-hops>1</max-hops>
         <static-connectors>
            <connector-ref>broker2-connector</connector-ref>
         </static-connectors>
      </cluster-connection>
   </cluster-connections>

   <cluster-user>amq_cluster_user</cluster-user>
   <cluster-password>Abcd1234</cluster-password>
```

Queue creation: -
https://access.redhat.com/documentation/en-us/red_hat_amq/7.7/html/configuring_amq_broker/addresses

The default configuration defines two queues: **DLQ** (Dead Letter Queue) handles messages that arrive with no known destination, and **Expiry Queue** holds messages that have lived past their expiration and therefore should not be routed to their original destination.

```
      <addresses>
         <address name="DLQ">
            <anycast>
               <queue name="DLQ" />
            </anycast>
         </address>
         <address name="ExpiryQueue">
            <anycast>
               <queue name="ExpiryQueue" />
            </anycast>
         </address>
         <address name="SampleQ">
            <anycast>
               <queue name="SampleQ"/>
            </anycast>
         </address>
         <address name="jms.queue.SampleQ2">
            <anycast>
               <queue name="jms.queue.SampleQ2"/>
            </anycast>
         </address>
      </addresses>
```

- For accessing the AMQ management console, modify "**jolokia-access.xml**" file by allowing cors access to the server's public IP address.

Console Url: -
http://127.164.132.130:8161/console

```
<cors>
    <!-- Allow cross origin access from 10.20.10.14 ... -->
    <allow-origin>*://10.20.10.14*</allow-origin>
    <allow-origin>*://127.164.132.130*</allow-origin> <!-- Public IP -->
```

- Register AMQ Broker as a Linux Service:
https://access.redhat.com/documentation/en-us/red_hat_amq/7.7/html/managing_amq_broker/assembly-using-command-line-interface-managing

```
$ sudo vi /etc/systemd/system/amqbroker7.service
$ cat /etc/systemd/system/amqbroker7.service
[Unit]
Description=AMQ Broker
After=syslog.target network.target
      You, 17 hours ago • EAP-AMQ interconnection - 2
[Service]
ExecStart=/eap/AMQ_CLUSTER/amq-broker-7.7.0/bin/broker/bin/artemis run
Restart=on-failure
User=jboss-user
Group=jboss-user

# A workaround for Java signal handling
SuccessExitStatus=143

[Install]
WantedBy=multi-user.target

$ sudo systemctl daemon-reload
$ sudo systemctl enable amqbroker7
Created symlink from /etc/systemd/system/multi-user.target.wants/amqbroker7.service to /etc/systemd/system/
amqbroker7.service.
$ sudo systemctl start amqbroker7
$ sudo systemctl status amqbroker7 -l
$ sudo systemctl stop amqbroker7
```

- Validate if Broker is running or not:

```
$ sudo netstat -tlnup | grep java
tcp6       0      0 10.20.10.14:5672        :::*                    LISTEN      18589/java
tcp6       0      0 10.20.10.14:61613       :::*                    LISTEN      18589/java
tcp6       0      0 10.20.10.14:61616       :::*                    LISTEN      18589/java
tcp6       0      0 10.20.10.14:1883        :::*                    LISTEN      18589/java
tcp6       0      0 10.20.10.14:8161        :::*                    LISTEN      18589/java
tcp6       0      0 10.20.10.14:5445        :::*                    LISTEN      18589/java
```

- Modify Firewall rules:

```
$ sudo firewall-cmd --add-port=8161/tcp --permanent # Connect to AMQ Broker Admin Console
$ sudo firewall-cmd --add-port=61616/tcp --permanent # Communicate with AMQ Broker Clients
$ sudo firewall-cmd --reload
$ sudo firewall-cmd --list-all
:::::
  ports: 8161/tcp 61616/tcp
:::::
```

03.2 Cluster validation

- Verify the logs in "**artemis.log**" that are generated every time a new Broker joins in or separates from the Cluster.

```
[rnayak@FCU-R20QAAMQ01 log]$ pwd
/eap/AMQ_CLUSTER/amq-broker-7.7.0/bin/broker/log
[rnayak@FCU-R20QAAMQ01 log]$ tail -n 10 artemis.log
::::::
2021-06-15 02:35:49,969 INFO  [org.apache.activemq.artemis] AMQ241004: Artemis Console available at http://10.20.10.
14:8161/console
2021-06-15 02:36:43,660 INFO  [io.hawt.web.LoginServlet] hawtio login is using 1800 sec. HttpSession timeout
2021-06-15 03:13:22,176 INFO  [org.apache.activemq.artemis.core.server] AMQ221027: Bridge
ClusterConnectionBridge@485b9634 [name=$.artemis.internal.sf.amq-cluster.2aa6713c-cda9-11eb-ba76-005056afe262,
queue=QueueImpl[name=$.artemis.internal.sf.amq-cluster.2aa6713c-cda9-11eb-ba76-005056afe262,
postOffice=PostOfficeImpl [server=ActiveMQServerImpl::serverUUID=a1f0ed57-cd9e-11eb-8138-005056aff001], temp=false]
@465ac23b targetConnector=ServerLocatorImpl (identity=(Cluster-connection-bridge::ClusterConnectionBridge@485b9634
[name=$.artemis.internal.sf.amq-cluster.2aa6713c-cda9-11eb-ba76-005056afe262, queue=QueueImpl[name=$.artemis.
internal.sf.amq-cluster.2aa6713c-cda9-11eb-ba76-005056afe262, postOffice=PostOfficeImpl
[server=ActiveMQServerImpl::serverUUID=a1f0ed57-cd9e-11eb-8138-005056aff001], temp=false]@465ac23b
targetConnector=ServerLocatorImpl [initialConnectors=[TransportConfiguration(name=netty-connector,
factory=org-apache-activemq-artemis-core-remoting-impl-netty-NettyConnectorFactory) ?port=61616&host=10-20-10-15],
discoveryGroupConfiguration=null]]::ClusterConnectionImpl@402695541[nodeUUID=a1f0ed57-cd9e-11eb-8138-005056aff001,
connector=TransportConfiguration(name=netty-connector,
factory=org-apache-activemq-artemis-core-remoting-impl-netty-NettyConnectorFactory) ?port=61616&host=10-20-10-14,
address=, server=ActiveMQServerImpl::serverUUID=a1f0ed57-cd9e-11eb-8138-005056aff001])) [initialConnectors=
[TransportConfiguration(name=netty-connector,
factory=org-apache-activemq-artemis-core-remoting-impl-netty-NettyConnectorFactory) ?port=61616&host=10-20-10-15],
discoveryGroupConfiguration=null]] is connected      You, seconds ago • Uncommitted changes
2021-06-15 07:06:23,810 INFO  [io.hawt.web.keycloak.KeycloakServlet] Keycloak integration is disabled
[rnayak@FCU-R20QAAMQ01 log]$
```

04 EAP Cluster Plan

04.1 Server setup

- Extract the **jboss-eap-7.2.0.zip** file to the preferred installation location.

```
[rnayak@FCU-R20QAWEB01 EAP-7.2.0]$ pwd
/eap/EAP_CLUSTER/EAP-7.2.0
[rnayak@FCU-R20QAWEB01 EAP-7.2.0]$ ls
appclient  auto.xml.variables  docs    installation   jboss-modules.jar  migration  standalone   version.txt
auto.xml   bin                 domain  JBossEULA.txt  LICENSE.txt        modules    uninstaller  welcome-content
[rnayak@FCU-R20QAWEB01 EAP-7.2.0]$
```

- Create and put sample "**t24-standalone-full-ha.xml**" file in standalone/configuration path.

```
[rnayak@FCU-R20QAWEB01 configuration]$ pwd
/eap/EAP_CLUSTER/EAP-7.2.0/standalone/configuration
[rnayak@FCU-R20QAWEB01 configuration]$ ls
application.keystore          logging.properties      standalone-full.xml           standalone.xml
t24-standalone-full-ha.xml      You, seconds ago • Uncommitted changes
application.keystore.backup   mgmt-groups.properties  standalone-ha.xml             standalone.xml.backup       T24.
xml
application-roles.properties  mgmt-users.properties   standalone-load-balancer.xml  standalone_xml_history
application-users.properties  standalone-full-ha.xml  standalone.orig.xml           standalone.xml_originialhb
[rnayak@FCU-R20QAWEB01 configuration]$
```

- Cluster and Queue related configurations are present in "**t24-standalone-full-ha.xml**".

Clustering: -
https://access.redhat.com/solutions/3021711

```
<stack name="tcp">
    <transport type="TCP" socket-binding="jgroups-tcp"/>
    <protocol type="org.jgroups.protocols.TCPPING">
        <property name="initial_hosts">
            ${custom.jboss.jgroups.tcp.initial_hosts}
        </property>
        <property name="port_range">
            0
        </property>
    </protocol>
    <protocol type="MERGE3"/>
    <protocol type="FD_SOCK" socket-binding="jgroups-tcp-fd"/>
    <protocol type="FD_ALL"/>
    <protocol type="VERIFY_SUSPECT"/>
    <protocol type="pbcast.NAKACK2"/>
    <protocol type="UNICAST3"/>
    <protocol type="pbcast.STABLE"/>
    <auth-protocol type="AUTH">
        <plain-token>
            <shared-secret-reference clear-text="${custom.jboss.jgroups.tcp.password:CHANGE ME!!}"/>
        </plain-token>
    </auth-protocol>
    <protocol type="pbcast.GMS"/>
    <protocol type="MFC"/>
    <protocol type="FRAG2"/>
</stack>
```

Queue related configurations: -
https://developers.redhat.com/blog/2018/12/06/how-to-integrate-a-remote-red-hat-amq-7-cluster-on-red-hat-jboss-eap-7

JBoss EAP 7 includes a default configuration for the messaging-activemq subsystem with the full or full-ha configuration. In order to connect to a remote server, we have to configure the below steps.

1. Define the **remote socket bindings** to connect to each RHAMQ 7 broker deployed in its cluster.

```
    <outbound-socket-binding name="remote-artemis-1">
        <remote-destination host="10.20.10.14" port="61616"/>
    </outbound-socket-binding>
    <outbound-socket-binding name="remote-artemis-2">
        <remote-destination host="10.20.10.15" port="61616"/>
    </outbound-socket-binding>
</socket-binding-group>
```

2. Define new **remote connectors** and a new **pooled connection factory** called “remote-artemis”.

```
<subsystem xmlns="urn:jboss:domain:messaging-activemq:4.0">
    <server name="default" id-cache-size="2000">
        <remote-connector name="remote-artemis-1" socket-binding="remote-artemis-1"/>
        <remote-connector name="remote-artemis-2" socket-binding="remote-artemis-2"/>
        <pooled-connection-factory name="remote-artemis" entries="java:/RemoteJmsXA java:jboss/RemoteJmsXA java:jboss/
        DefaultJMSConnectionFactory" connectors="remote-artemis-1 remote-artemis-2" ha="true" user="admin" password="Admin123$"
        min-pool-size="15" max-pool-size="30" statistics-enabled="true">      You, 21 hours ago • EAP-AMQ interconnection - 2
            <inbound-config rebalance-connections="true" setup-attempts="-1" setup-interval="5000"/>
        </pooled-connection-factory>
    </server>
</subsystem>
```

3. Define a new **external context** to declare the Queues and Topics in the RHAMQ 7 cluster. This step will define a local JNDI entry to connect to the remote resources.

```
<subsystem xmlns="urn:jboss:domain:naming:2.0">
    <bindings>
        <external-context name="java:global/remoteContext" module="org.apache.activemq.artemis" class="javax.naming.InitialContext">
            <environment>
                <property name="java.naming.factory.initial" value="org.apache.activemq.artemis.jndi.ActiveMQInitialContextFactory"/>
                <property name="queue.SampleQ" value="SampleQ"/>
            </environment>
        </external-context>
    </bindings>
    <remote-naming/>
</subsystem>
```

- Create a Server admin user:

```
[jboss-user@FCU-R20QAWEB02 bin]$ pwd
/eap/EAP_CLUSTER/EAP-7.2.0/bin
[jboss-user@FCU-R20QAWEB02 bin]$ ./add-user.sh admin Admin123$
Added user 'admin' to file '/eap/EAP_CLUSTER/EAP-7.2.0/standalone/configuration/mgmt-users.properties'
Added user 'admin' to file '/eap/EAP_CLUSTER/EAP-7.2.0/domain/configuration/mgmt-users.properties'
```

- Start the server manually:

```
[rnayak@FCU-R20QAWEB01 init.d]$ pwd
/eap/EAP_CLUSTER/EAP-7.2.0/bin/init.d
[rnayak@FCU-R20QAWEB01 init.d]$ ls
t24-standalone.sh
[rnayak@FCU-R20QAWEB01 init.d]$ sudo su jboss-user
[jboss-user@FCU-R20QAWEB01 init.d]$ sh t24-standalone.sh
```

```
[rnayak@FCU-R20QAWEB01 init.d]$ cat t24-standalone.sh
#/bin/bash

hostip="127.0.0.1"

# get assigned private ip:
getHostIP() {
 echo "pre-while-hostip-$hostip"
 while [ "$hostip" == "127.0.0.1" ]
  do
   echo "do-hostip-$hostip"
   sleep 5
   hostip=`hostname -I`
  done
 echo "done-hostip-$hostip"
}
getHostIP

# constants:
initial_hosts="10.20.13.10[7600],10.20.13.11[7600]"
messaging_cluster_password="Abcd1234"
jgroups_tcp_password="Wxyz1234"

# start server:
sh /eap/EAP_CLUSTER/EAP-7.2.0/bin/standalone.sh --server-config=t24-standalone-full-ha.xml -Djboss.bind.address=$hostip -Djboss.bind.address.
management=$hostip -Djboss.bind.address.private=$hostip -Djboss.bind.address.unsecure=$hostip -Djboss.server.name=Server-$hostip -Djboss.node.
name=Node-$hostip -Djboss.tx.node.id=TxNode-$hostip -Djboss.messaging.cluster.password=$messaging_cluster_password -Dcustom.jboss.jgroups.tcp.
password=$jgroups_tcp_password -Dcustom.jboss.jgroups.tcp.initial_hosts=$initial_hosts
[rnayak@FCU-R20QAWEB01 init.d]$
```

- Register Jboss EAP as a Linux Service:
  https://access.redhat.com/solutions/1320133

```
$ sudo vi /etc/systemd/system/jbosseap7.service
$ cat /etc/systemd/system/jbosseap7.service
[Unit]
Description=JBoss EAP
After=NetworkManager.service

[Service]
Type=simple
User=jboss-user
Group=jboss-user
ExecStart=/bin/sh /eap/EAP_CLUSTER/EAP-7.2.0/bin/init.d/t24-standalone.sh
Restart=always
RestartSec=5
TimeoutStartSec=5
TimeoutStopSec=5

[Install]
WantedBy=multi-user.target

$ sudo systemctl daemon-reload
$ sudo systemctl enable jbosseap7
Created symlink from /etc/systemd/system/multi-user.target.wants/jbosseap7.service to /etc/systemd/system/jbosseap7.service.
$ sudo systemctl start jbosseap7
$ sudo systemctl status jbosseap7 -l
$ sudo systemctl stop jbosseap7
```

- Validate if Server is running or not:

```
$ sudo netstat -tlnup | grep java
tcp        0      0 10.20.13.10:7600        0.0.0.0:*               LISTEN      888/java
tcp        0      0 10.20.13.10:8080        0.0.0.0:*               LISTEN      888/java
tcp        0      0 10.20.13.10:8443        0.0.0.0:*               LISTEN      888/java
tcp        0      0 10.20.13.10:40095       0.0.0.0:*               LISTEN      888/java
tcp        0      0 10.20.13.10:9990        0.0.0.0:*               LISTEN      888/java
tcp        0      0 10.20.13.10:3528        0.0.0.0:*               LISTEN      888/java
tcp        0      0 10.20.13.10:8009        0.0.0.0:*               LISTEN      888/java
udp        0      0 224.0.1.105:23364       0.0.0.0:*                           888/java
```

- Modify Firewall rules:

```
$ sudo firewall-cmd --add-port=9990/tcp --permanent # Connect to EAP Admin Console
$ sudo firewall-cmd --add-port=8080/tcp --permanent # Connect to EAP Applications
$ sudo firewall-cmd --add-port=7600/tcp --permanent # Communicate with EAP Cluster Members
$ sudo firewall-cmd --reload
$ sudo firewall-cmd --list-all
:::::
  ports: 9990/tcp 8080/tcp 7600/tcp
:::::
```

04.2 Cluster validation

- Verify the logs that are generated in "**server.log**" every time a new Server joins in or separates from the Cluster.

```
[rnayak@FCU-R20QAWEB01 ~]$ cd /eap/EAP_CLUSTER/EAP-7.2.0/standalone/log/
[rnayak@FCU-R20QAWEB01 log]$ ls
audit.log              browser.log     console.log       server.hb.log  server.log.2021-06-15
backupgc.log.current   console.hb.log  gc.log.0.current  server.log     server.log.2021-06-16
[rnayak@FCU-R20QAWEB01 log]$
```

```
2021-06-17 06:02:49,595 INFO  [org.infinispan.CLUSTER] (thread-264,ejb,Node-10.20.13.10) ISPN000094: Received new cluster view for channel ejb:
[Node-10.20.13.10|3] (2) [Node-10.20.13.10, Node-10.20.13.11]
2021-06-17 06:02:49,595 INFO  [org.infinispan.CLUSTER] (thread-264,ejb,Node-10.20.13.10) ISPN100000: Node Node-10.20.13.11 joined the cluster
2021-06-17 06:02:49,596 INFO  [org.infinispan.CLUSTER] (thread-264,ejb,Node-10.20.13.10) ISPN000094: Received new cluster view for channel ejb:
[Node-10.20.13.10|3] (2) [Node-10.20.13.10, Node-10.20.13.11]
2021-06-17 06:02:49,597 INFO  [org.infinispan.CLUSTER] (thread-264,ejb,Node-10.20.13.10) ISPN100000: Node Node-10.20.13.11 joined the cluster
2021-06-17 06:02:49,598 INFO  [org.infinispan.CLUSTER] (thread-264,ejb,Node-10.20.13.10) ISPN000094: Received new cluster view for channel ejb:
[Node-10.20.13.10|3] (2) [Node-10.20.13.10, Node-10.20.13.11]
2021-06-17 06:02:49,599 INFO  [org.infinispan.CLUSTER] (thread-264,ejb,Node-10.20.13.10) ISPN100000: Node Node-10.20.13.11 joined the cluster
2021-06-17 06:02:49,600 INFO  [org.infinispan.CLUSTER] (thread-264,ejb,Node-10.20.13.10) ISPN000094: Received new cluster view for channel ejb:
[Node-10.20.13.10|3] (2) [Node-10.20.13.10, Node-10.20.13.11]
2021-06-17 06:02:49,601 INFO  [org.infinispan.CLUSTER] (thread-264,ejb,Node-10.20.13.10) ISPN100000: Node Node-10.20.13.11 joined the cluster
```

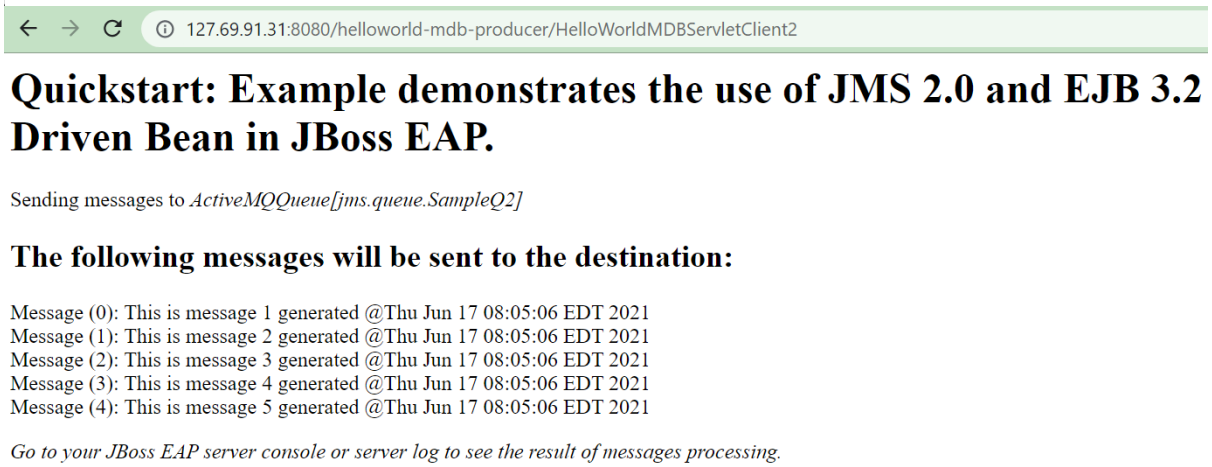
```
2021-06-17 06:02:44,839 INFO  [org.infinispan.CLUSTER] (thread-264,ejb,Node-10.20.13.10) ISPN000094: Received new cluster view for channel ejb:
[Node-10.20.13.10|2] (1) [Node-10.20.13.10]
2021-06-17 06:02:44,840 INFO  [org.infinispan.CLUSTER] (thread-264,ejb,Node-10.20.13.10) ISPN100001: Node Node-10.20.13.11 left the cluster
2021-06-17 06:02:44,841 INFO  [org.infinispan.CLUSTER] (thread-264,ejb,Node-10.20.13.10) ISPN000094: Received new cluster view for channel ejb:
[Node-10.20.13.10|2] (1) [Node-10.20.13.10]
2021-06-17 06:02:44,842 INFO  [org.infinispan.CLUSTER] (thread-264,ejb,Node-10.20.13.10) ISPN100001: Node Node-10.20.13.11 left the cluster
2021-06-17 06:02:44,843 INFO  [org.infinispan.CLUSTER] (thread-264,ejb,Node-10.20.13.10) ISPN000094: Received new cluster view for channel ejb:
[Node-10.20.13.10|2] (1) [Node-10.20.13.10]
2021-06-17 06:02:44,844 INFO  [org.infinispan.CLUSTER] (thread-264,ejb,Node-10.20.13.10) ISPN100001: Node Node-10.20.13.11 left the cluster
2021-06-17 06:02:44,844 INFO  [org.infinispan.CLUSTER] (thread-264,ejb,Node-10.20.13.10) ISPN000094: Received new cluster view for channel ejb:
[Node-10.20.13.10|2] (1) [Node-10.20.13.10]
2021-06-17 06:02:44,845 INFO  [org.infinispan.CLUSTER] (thread-264,ejb,Node-10.20.13.10) ISPN100001: Node Node-10.20.13.11 left the cluster
```

05 EAP - AMQ Communication

For testing communication purpose, deploy below 2 applications in EAP Cluster.

1. **helloworld-mdb-producer.war** (for sending messages to the Queue)
2. **helloworld-mdb-consumer.jar** (for receiving messages from the Queue)

05.1 Send messages to the Queues



05.2 Receive messages from the Queues

Logs will appear in EAP "**server.log**" file.

```
2021-06-17 08:02:57,710 INFO  [class org.jboss.as.quickstarts.mdb.HelloWorldQueueMDB] (Thread-65 (ActiveMQ-client-global-threads)) Received
Message from queue 'SampleQ': This is message 1 generated @Thu Jun 17 08:02:57 EDT 2021
2021-06-17 08:02:57,713 INFO  [class org.jboss.as.quickstarts.mdb.HelloWorldQueueMDB] (Thread-66 (ActiveMQ-client-global-threads)) Received
Message from queue 'SampleQ': This is message 2 generated @Thu Jun 17 08:02:57 EDT 2021
2021-06-17 08:02:57,714 INFO  [class org.jboss.as.quickstarts.mdb.HelloWorldQueueMDB] (Thread-68 (ActiveMQ-client-global-threads)) Received
Message from queue 'SampleQ': This is message 4 generated @Thu Jun 17 08:02:57 EDT 2021
2021-06-17 08:02:57,716 INFO  [class org.jboss.as.quickstarts.mdb.HelloWorldQueueMDB] (Thread-67 (ActiveMQ-client-global-threads)) Received
Message from queue 'SampleQ': This is message 3 generated @Thu Jun 17 08:02:57 EDT 2021
2021-06-17 08:02:57,722 INFO  [class org.jboss.as.quickstarts.mdb.HelloWorldQueueMDB] (Thread-69 (ActiveMQ-client-global-threads)) Received
Message from queue 'SampleQ': This is message 5 generated @Thu Jun 17 08:02:57 EDT 2021
      You, seconds ago • Uncommitted changes
2021-06-17 08:05:06,895 INFO  [class org.jboss.as.quickstarts.mdb.HelloWorldQueueMDB2] (Thread-70 (ActiveMQ-client-global-threads)) Received
Message from queue 'jms.queue.SampleQ2': This is message 2 generated @Thu Jun 17 08:05:06 EDT 2021
2021-06-17 08:05:06,896 INFO  [class org.jboss.as.quickstarts.mdb.HelloWorldQueueMDB2] (Thread-71 (ActiveMQ-client-global-threads)) Received
Message from queue 'jms.queue.SampleQ2': This is message 1 generated @Thu Jun 17 08:05:06 EDT 2021
2021-06-17 08:05:06,898 INFO  [class org.jboss.as.quickstarts.mdb.HelloWorldQueueMDB2] (Thread-72 (ActiveMQ-client-global-threads)) Received
Message from queue 'jms.queue.SampleQ2': This is message 3 generated @Thu Jun 17 08:05:06 EDT 2021
2021-06-17 08:05:06,899 INFO  [class org.jboss.as.quickstarts.mdb.HelloWorldQueueMDB2] (Thread-73 (ActiveMQ-client-global-threads)) Received
Message from queue 'jms.queue.SampleQ2': This is message 4 generated @Thu Jun 17 08:05:06 EDT 2021
2021-06-17 08:05:06,909 INFO  [class org.jboss.as.quickstarts.mdb.HelloWorldQueueMDB2] (Thread-230 (ActiveMQ-client-global-threads)) Received
Message from queue 'jms.queue.SampleQ2': This is message 5 generated @Thu Jun 17 08:05:06 EDT 2021
```