



[Products & Services](#) › [Knowledgebase](#) › [Solutions](#) › Using Encrypted Property Placeholders for ActiveMQ SSL passwords in "etc/activemq.xml" file

Using Encrypted Property Placeholders for ActiveMQ SSL passwords in "etc/activemq.xml" file

○ Updated July 30 2014 at 3:06 PM

Issue

When activating ActiveMQ features in JBoss Fuse, the default etc/activemq.xml file is used as configuration.

In order to activate SSL, some configuration needs to be applied on the file, mainly to define the keystore and truststore ActiveMQ will use.

Here is an example of the configuration:

```
<broker xmlns="http://activemq.apache.org/schema/core" brokerName="${brokerName}"
...
<sslContext>
  <sslContext keyStore="wherever.jks" keyStorePassword="changeit" trustStore="wherever.jks" trustStorePassword="changeit" />
</sslContext>
...
</broker>
```

As you can see, the password to the keystore and truststore have to be written in clear in the file.

Is there a way to encrypt these passwords so they do not appear in clear text in the file? Something

like in etc/users.properties maybe?

Environment

- Fuse ESB Enterprise 7.1
- Fuse MQ Enterprise 7.1
- Red Hat JBoss A-MQ 6.0
- Red Hat JBoss Fuse 6.0
- Red Hat JBoss A-MQ 6.1
- Red Hat JBoss Fuse 6.1

Resolution

The section "Using Encrypted Property Placeholders" in both [Fuse ESB 7.1](https://access.redhat.com/site/documentation/en-US/Fuse_ESB_Enterprise/7.1/html-single/Security_Guide/index.html#ESBSecurityEncryptProperties) ([https://access.redhat.com/site/documentation/en-US/Fuse ESB Enterprise/7.1/html-single/Security_Guide/index.html#ESBSecurityEncryptProperties](https://access.redhat.com/site/documentation/en-US/Fuse_ESB_Enterprise/7.1/html-single/Security_Guide/index.html#ESBSecurityEncryptProperties)) and [JBoss Fuse 6.0](https://access.redhat.com/site/documentation/en-US/JBoss_Fuse/6.0/html-single/Security_Guide/index.html#ESBSecurityEncryptProperties) ([https://access.redhat.com/site/documentation/en-US/JBoss Fuse/6.0/html-single/Security_Guide/index.html#ESBSecurityEncryptProperties](https://access.redhat.com/site/documentation/en-US/JBoss_Fuse/6.0/html-single/Security_Guide/index.html#ESBSecurityEncryptProperties)) Security Guide described how to use Blueprint Property Placeholder extension and the namespace `xmlns:enc="http://karaf.apache.org/xmlns/jasypt/v1.0.0"`

to encrypt properties. However, it won't be useful here because "etc/activemq.xml" is a Spring xml but not a Blueprint xml and the namespace "http://karaf.apache.org/xmlns/jasypt/v1.0.0" only works in Blueprint but not Spring DM. Therefore, you will need to use a Spring equivalent property placeholder configurer instead. For instance, using Jasypt

EncryptablePropertyPlaceholderConfigurer just as described in ActiveMQ documentation page: <http://activemq.apache.org/encrypted-passwords.html> (<http://activemq.apache.org/encrypted-passwords.html>)

Note, there was some change since Jasypt 1.9.0 version where the "org.jasypt.spring.properties" package was no longer included in Jasypt main jar. You will have to add spring specific Jasypt jar to the classpath in order to use the relevant Jasypt Property Placeholder Configurer. Please take a look at below link for more detail:

http://www.jasypt.org/spring31.html#Encryptable_PropertySource_implementation_for_Spring_3.1
(http://www.jasypt.org/spring31.html#Encryptable_PropertySource_implementation_for_Spring_3.1)

There are two spring specific jasypt jars: jasypt-spring3 for Spring 3.0.x version and jasypt-spring31 for Spring 3.1.x and later versions. Since Fuse ESB Enterprise 7.1 only ships Spring 3.0.7.RELEASE you will need to use jasypt-spring3 for Fuse ESB Enterprise 7.1 while use jasypt-spring31 for JBoss Fuse 6.0 and 6.1 because in JBoss Fuse 6.x we ship Spring 3.1.x and 3.2.x respectively.

For jasypt-spring31 jar, ServiceMix provides custom build bundles from 1.9.0 and onwards. For

instance "org.apache.servicemix.bundles.jasypt-spring31/1.9.0_1". However, there isn't a custom build bundle for jasypt-spring3 jar. Therefore, you will have to directly install the plain jar using "wrap:" protocol.

1. Install jasypt-spring3 or jasypt-spring31 bundle during container start but right before mq-fabric bundle to be activated.

Respectively, for Fuse ESB Enterprise 7.1, you can insert following line

```
<bundle dependency="true">wrap:mvn:org.jasypt/jasypt-spring3/1.9.0</
```

to "activemq" feature in the file:

```
system/org/apache/activemq/activemq-karaf/5.7.0.fuse-71-047/activemq-karaf-5.7.0.fuse-71-047-features.xml
```

For JBoss Fuse 6.0, insert the following line:

```
<bundle dependency="true">mvn:org.apache.servicemix.bundles/org.apac
```

to "activemq" feature in the file:

```
system/org/apache/activemq/activemq-karaf/5.8.0.redhat-60024/activemq-karaf-5.8.0.redhat-60024-features.xml
```

For JBoss Fuse 6.1, insert the following line:

```
<bundle dependency="true">mvn:org.apache.servicemix.bundles/org.apac
```

to "activemq" feature in the file:

```
system/org/apache/activemq/activemq-karaf/5.9.0.redhat-610379/activemq-karaf-5.9.0.redhat-610379-features-core.xml
```

2. Add similar configuration to "etc/activemq.xml" file as described in ActiveMQ documentation page:
[<http://activemq.apache.org/encrypted-passwords.html>]

```

<bean id="environmentVariablesConfiguration" class="org.jasypt.ency
  <property name="algorithm" value="PBEWithMD5AndDES" />
  <property name="passwordEnvName" value="ACTIVEMQ_ENCRYPTION_PASS
</bean>

<bean id="configurationEncryptor" class="org.jasypt.encrypted.pbe.S
  <property name="config" ref="environmentVariablesConfiguration"
</bean>

<bean id="propertyConfigurer" class="org.jasypt.spring31.properties.
  <constructor-arg ref="configurationEncryptor" />
  <property name="location" value="file:${karaf.base}/etc/credenti
  <property name="properties">
    <bean class="org.fusesource.mq.fabric.ConfigurationPropertie
  </property>
</bean>

```

Note, for Fuse ESB Enterprise 7.1, you will need to change the "propertyConfigurer" bean slightly to take jasypt-spring3 version into account:

```

<bean id="propertyConfigurer" class="org.jasypt.spring3.properties.E
...

```

3. Encrypt your keystore and truststore password. If you are using the karaf container, you'll need to navigate to your {AMQHome}/extras/ directory. There, you'll find a file called apache-activemq-5.9.0.redhat-610379-bin.zip. You'll need to unzip this into your extras folder, go into the apache-activemq-5.9.0.redhat-610379 directory and run:

```

./bin/activemq encrypt --password activemq --input mypassword

```

Where activemq is your encryption password and mypassword is the password to your key or truststore. This should output the password. To test the generated password, you can run the following command.

```

./bin/activemq decrypt --password activemq --input o93Cx1j1PqvhqH8X

```

4. Create a file "credentials-enc.properties" under "etc/" folder of either Fuse ESB Enterprise 7.1 container or JBoss Fuse 6.x container and put your encrypted keystore password there, something like:

```

keystore.password=ENC(Cf3Jf3tM+UrS0oaKU50od5CuBa8rxjoL)
truststore.password=ENC(eeWjNyX6FY8Fjp3E+F6qTytV11bZITdP)

```

Note, you can use built in encrypt/decrypt utilities from activemq to encrypt your clear text password to prepare "etc/credentials-enc.properties" file. For more details regarding to

activemq build in encrypt/decrypt utilities, please take a look at the same ActiveMQ documentation link mentioned above.

5. Finally configure <sslContext> to the "etc/activemq.xml" file using property placeholder like:

```
<sslContext>
  <sslContext keyStore="wherever.jks"
                keyStorePassword="${keystore.password}"
                trustStore="wherever.jks"
                trustStorePassword="${truststore.password}"
  </sslContext>
...
<transportConnectors>
  <transportConnector name="openwire" uri="tcp://0.0.0.0:0?max
  <transportConnector name="nio+ssl" uri="nio+ssl://0.0.0.0:61
</transportConnectors>
```

SBR [FuseSource \(/sbr/fusesource\)](#)

Product(s)

[Red Hat JBoss Fuse \(https://access.redhat.com/search/browse/solutions#?&portal_product=Red+Hat+JBoss+Fuse\)](#)

Component [fuse \(https://access.redhat.com/search/browse/solutions#?&portal_component=fuse\)](#)

Category [Learn more \(https://access.redhat.com/search/browse/solutions#?&portal_category=Learn+more\)](#)

Tags [fuse \(https://access.redhat.com/search/browse/solutions#?&portal_tag=fuse\)](#)

Comments



[\(/user/3316263\)](#)

[Joe Luo \(/user/3316263\)](#)

Red Hat

Active Contributor

300 points

firstly add jasypt-spring31 bundle to "activemq" feature that is defined in the following file to make sure that the jasypt-spring31 bundle gets installed during features boot:

system/org/apache/activemq/activemq-karaf/5.9.0.redhat-610379/activemq-karaf-5.9.0.redhat-610379-features-core.xml

Something like:

[June 16 2014 at 9:40 AM \(https://access.redhat.com/solutions/742423#comment-773313\)](#)

- **Private Comment**

```
<feature name="activemq" description="ActiveMQ broker"
...
  <bundle dependency="true">mvn:org.apache.serv:
  <bundle dependency="true">mvn:org.apache.serv:
  <bundle dependency="true">mvn:org.apache.serv:
...

```

Then change property placeholder configurer in the "etc/activemq.xml" file like:

```
<!--
  <bean class="org.springframework.beans.factory.config.PropertyPlaceholderConfigurer"
    <property name="properties">
      <bean class="org.fusesource.mq.fabric.Configuration"
    </property>
  </bean>
-->

  <bean id="environmentVariablesConfiguration" class="org.fusesource.mq.fabric.Configuration"
    <property name="algorithm" value="PBEWithMD5AndDES"
    <property name="passwordEnvName" value="ACTIVE_MQ_PASSWORD"
  </bean>

  <bean id="configurationEncryptor" class="org.fusesource.mq.fabric.ConfigurationEncryptor"
    <property name="config" ref="environmentVariablesConfiguration"
  </bean>

  <bean id="propertyConfigurer" class="org.fusesource.mq.fabric.ConfigurationPropertyPlaceholderConfigurer"
    <constructor-arg ref="configurationEncryptor"
    <property name="location" value="file:${karaf.home}/etc/activemq.xml"
    <property name="properties">
      <bean class="org.fusesource.mq.fabric.Configuration"
    </property>
  </bean>

```

The rest steps should be exactly the same.



(/user

/5801323)

Eugene Curry (/user

/5801323)

Red Hat

Newbie

5 points

I am running JBoss Fuse 6.1 GA.

The only way i can get the dependency to install/start is if i specify
dependency="false" in the bundle declaration inside the feature file:

```
<bundle dependency="false">mvn:org.apache.serv
```

February 5 2015 at 2:45 PM (<https://access.redhat.com/solutions/742423#comment-881083>) - Edited

Add new comment

[Formatting Help](#)

Mark comment as private

SUBMIT COMMENT

[Privacy Policy \(http://www.redhat.com/en/about/privacy-policy\)](http://www.redhat.com/en/about/privacy-policy) | [Customer Portal Terms of Use \(https://access.redhat.com/solutions/742423#comment-881083\)](https://access.redhat.com/solutions/742423#comment-881083)
| [All Policies and Guidelines \(http://www.redhat.com/en/about/all-policies-guidelines\)](http://www.redhat.com/en/about/all-policies-guidelines)

Copyright © 2015 Red Hat, Inc.