MCD-LEVEL-2 Q&As

MuleSoft Certified Developer - Level 2 (Mule 4)

QUESTION 1

A scatter-gather router is configured with four routes:Route A, B, C and D.

Route C false.

A. Error,errorMesage.payload.results [`2\\']

B. Payload failures[`2\\']

C. Error,errorMessage,payload.failures[`2\\']

D. Payload [`2\\']

Correct Answer: C

The result of accessing route C failure is Error,errorMessage,payload.failures[`2\\']. This is because a scatter-gather router returns an aggregated message that contains an array of results from each route and an array of failures from each route. The failures array contains error objects with information about each failed route execution. To access route C failure, which is the third route (index 2), the developer needs to use Error.errorMessage.payload.failures[`2\\'] expression.

References:https://docs.mulesoft.com/mule-runtime/4.3/scatter-gather-reference#scatter-gather-output

---

QUESTION 2

The Center for Enablement team published a common application as a reusable module to the central Nexus repository.

How can the common application be included in all API implementations?

A. Download the common application from Naxus and copy it to the src/main/resources folder in the API

B. Copy the common application\\'s source XML file and out it in a new flow file in the src/main/mule folder

C. Add a Maven dependency in the PCM file with multiple-plugin as

D. Add a Maven dependency in the POM file with jar as

Correct Answer: D

To include a common application as a reusable module in all API implementations, the developer should add a Maven dependency in the POM file with jar as . This way, the developer can reuse Mule code from another application by packaging it as a JAR file and adding it as a dependency in the POM file of the API implementation. The classifier element specifies that it is a JAR file. References:https://docs.mulesoft.com/mule-runtime/4.3/mmp-concept#add-a-maven-dependency-to-the-pom-file

---

QUESTION 3

Which configurations are required for HTTP Listener to enable mTLS authentication?

A. Set an appropriate reconnection strategy and use persistent connections for the listener

B. Set an appropriate keystore configuration and use persistent connections for the listener

C. Set an appropriate keystore and truststoreconfiguration for the listener

D. Set an appropriate truststore configuration and reconnection strategy for the listener

Correct Answer: C

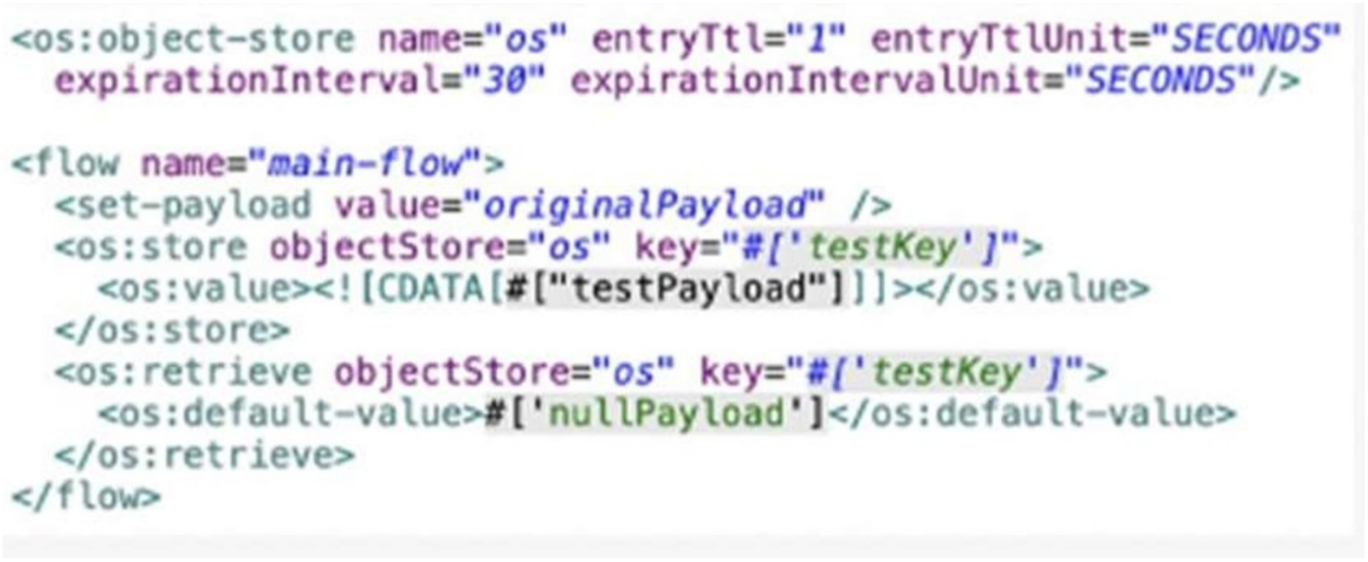
To enable mTLS authentication for HTTP Listener, the developer needs to set an appropriate keystore and truststore configuration for the listener. The keystore contains the certificate and private key of the Mule application that are used to prove its identity to clients. The truststore contains the certificates of trusted clients that are allowed to access the Mule application. References:https://docs.mulesoft.com/mule-runtime/4.3/tls-configuration#mutual-authentication

QUESTION 4

Refer to the exhibit.

A Mute Object Store is configured with an entry TTL of one second and an expiration interval of 30 seconds.

What is the result of the flow if processing between os\\'store and os:retrieve takes 10 seconds?


```
<os:object-store name="os" entryTtl="1" entryTtlUnit="SECONDS"
  expirationInterval="30" expirationIntervalUnit="SECONDS"/>

<flow name="main-flow">
  <set-payload value="originalPayload" />
  <os:store objectStore="os" key="#['testKey']">
    <os:value><![CDATA[#["testPayload"]]]></os:value>
  </os:store>
  <os:retrieve objectStore="os" key="#['testKey']">
    <os:default-value>#['nullPayload']</os:default-value>
  </os:retrieve>
</flow>
```

A. nullPayload

B. originalPayload

C. OS:KEY_NOT_FOUND

D. testPayload

Correct Answer: A

The result of the flow is nullPayload if processing between os:store and os:retrieve takes 10 seconds. This is because the entry TTL of the object store is one second, which means that any stored value expires after one second and is removed from the object store. The expiration interval of 30 seconds only determines how often the object store checks

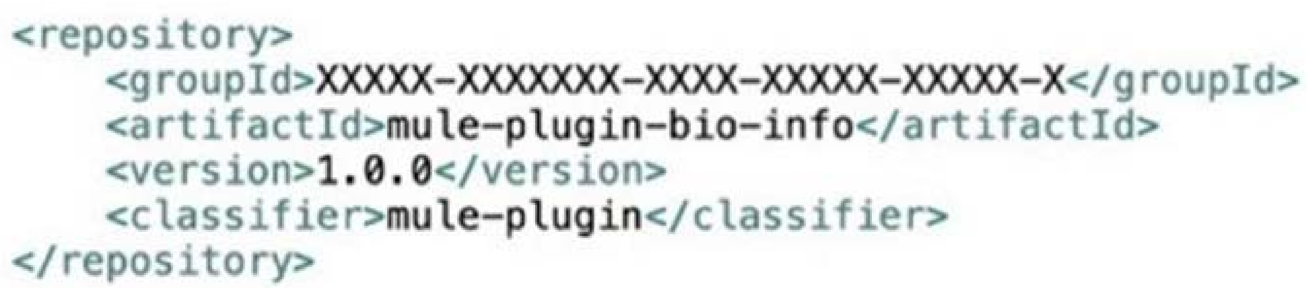
for expired values, but it does not affect the TTL. Therefore, when os:retrieve tries to get the value after 10 seconds, it returns nullPayload because the value has already expired and been removed.
References:https://docs.mulesoft.com/object-store/osv2-faq#how-does-the-time-to-live-work

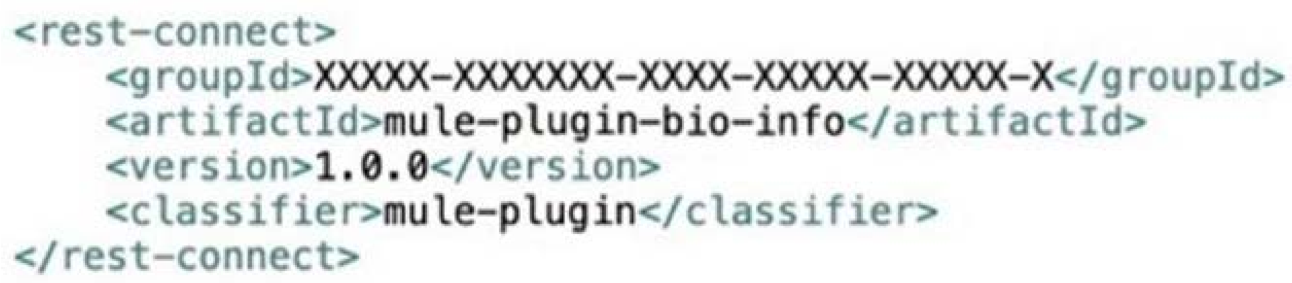
QUESTION 5

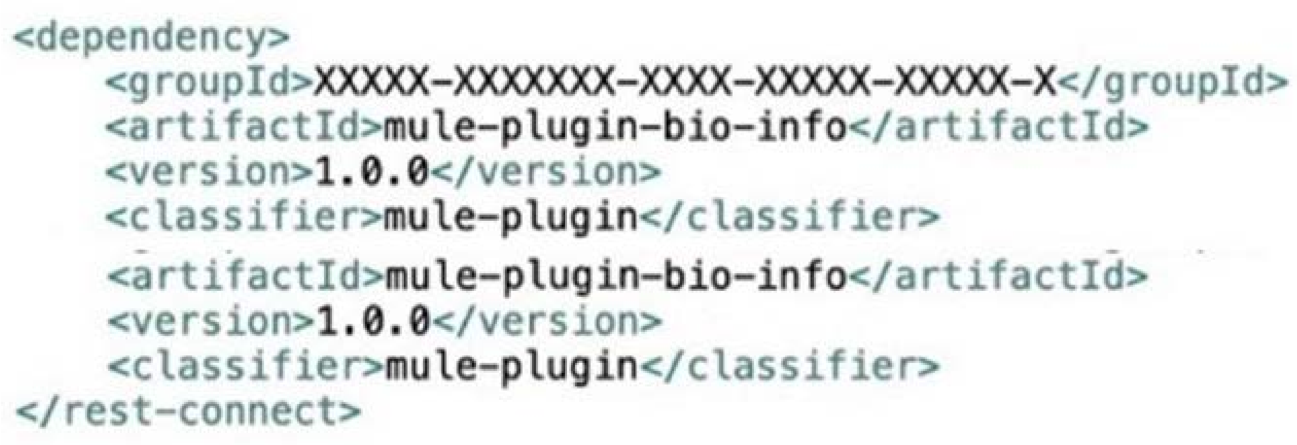
Refer to the exhibits.

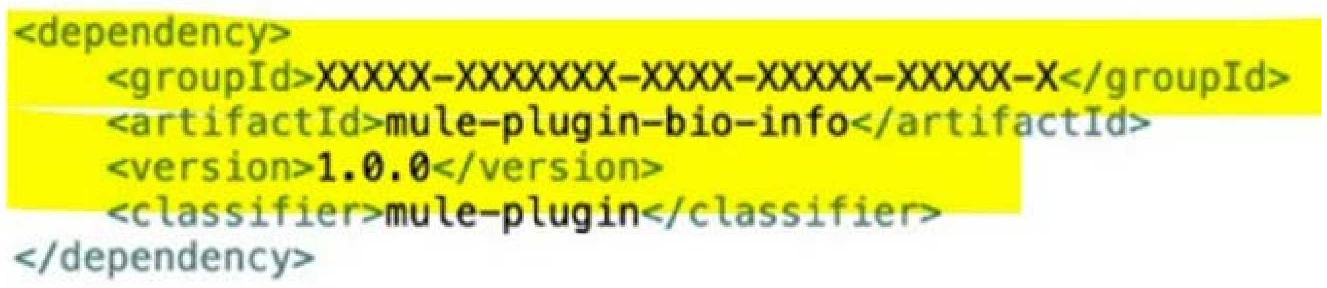
Bioinfo System API is implemented and published to Anypoint Exchange. A developer wants to invoke this API using its REST Connector.

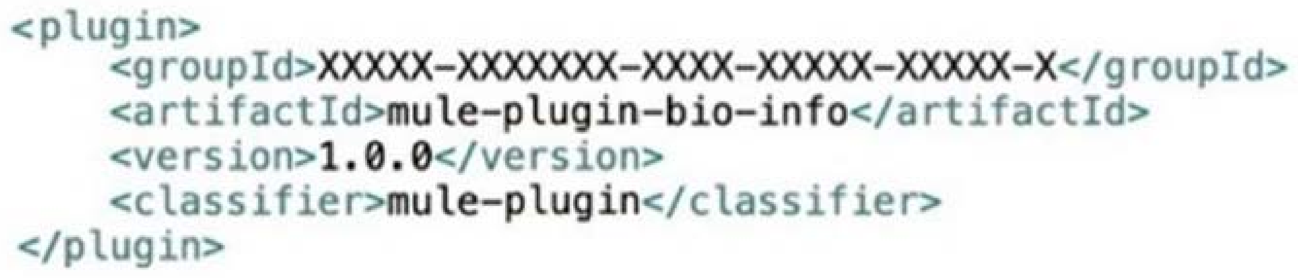
What should be added to the POM?

A.

```
<repository>
    <groupId>XXXXX-XXXXXXX-XXXX-XXXXX-XXXXX-X</groupId>
    <artifactId>mule-plugin-bio-info</artifactId>
    <version>1.0.0</version>
    <classifier>mule-plugin</classifier>
</repository>
```

B.

```
<rest-connect>
    <groupId>XXXXX-XXXXXXX-XXXX-XXXXX-XXXXX-X</groupId>
    <artifactId>mule-plugin-bio-info</artifactId>
    <version>1.0.0</version>
    <classifier>mule-plugin</classifier>
</rest-connect>
```

C.

```
<dependency>
    <groupId>XXXXX-XXXXXXX-XXXX-XXXXX-XXXXX-X</groupId>
    <artifactId>mule-plugin-bio-info</artifactId>
    <version>1.0.0</version>
    <classifier>mule-plugin</classifier>
    <artifactId>mule-plugin-bio-info</artifactId>
    <version>1.0.0</version>
    <classifier>mule-plugin</classifier>
</rest-connect>
```

D.

```
<dependency>
    <groupId>XXXXX-XXXXXXX-XXXX-XXXXX-XXXXX-X</groupId>
    <artifactId>mule-plugin-bio-info</artifactId>
    <version>1.0.0</version>
    <classifier>mule-plugin</classifier>
</dependency>
```

E.

```
<plugin>
    <groupId>XXXXX-XXXXXXX-XXXX-XXXXX-XXXXX-X</groupId>
    <artifactId>mule-plugin-bio-info</artifactId>
    <version>1.0.0</version>
    <classifier>mule-plugin</classifier>
</plugin>
```

A. Option A

B. Option B

C. Option C

D. Option D

E. Option E

Correct Answer: E

To invoke Bioinfo System API using its REST Connector, option E should be added to the pom.xml file. This option adds a dependency for Bioinfo System API REST Connector with its group ID, artifact ID, version, classifier, and type. The classifier specifies that it is a REST Connector (raml-client), and the type specifies that it is a Mule plugin (mule-plugin). References:https://docs.mulesoft.com/apikit/4.x/apikit-4-generate-from-rest-api-task#add-the-api-dependency-to-thepom-file