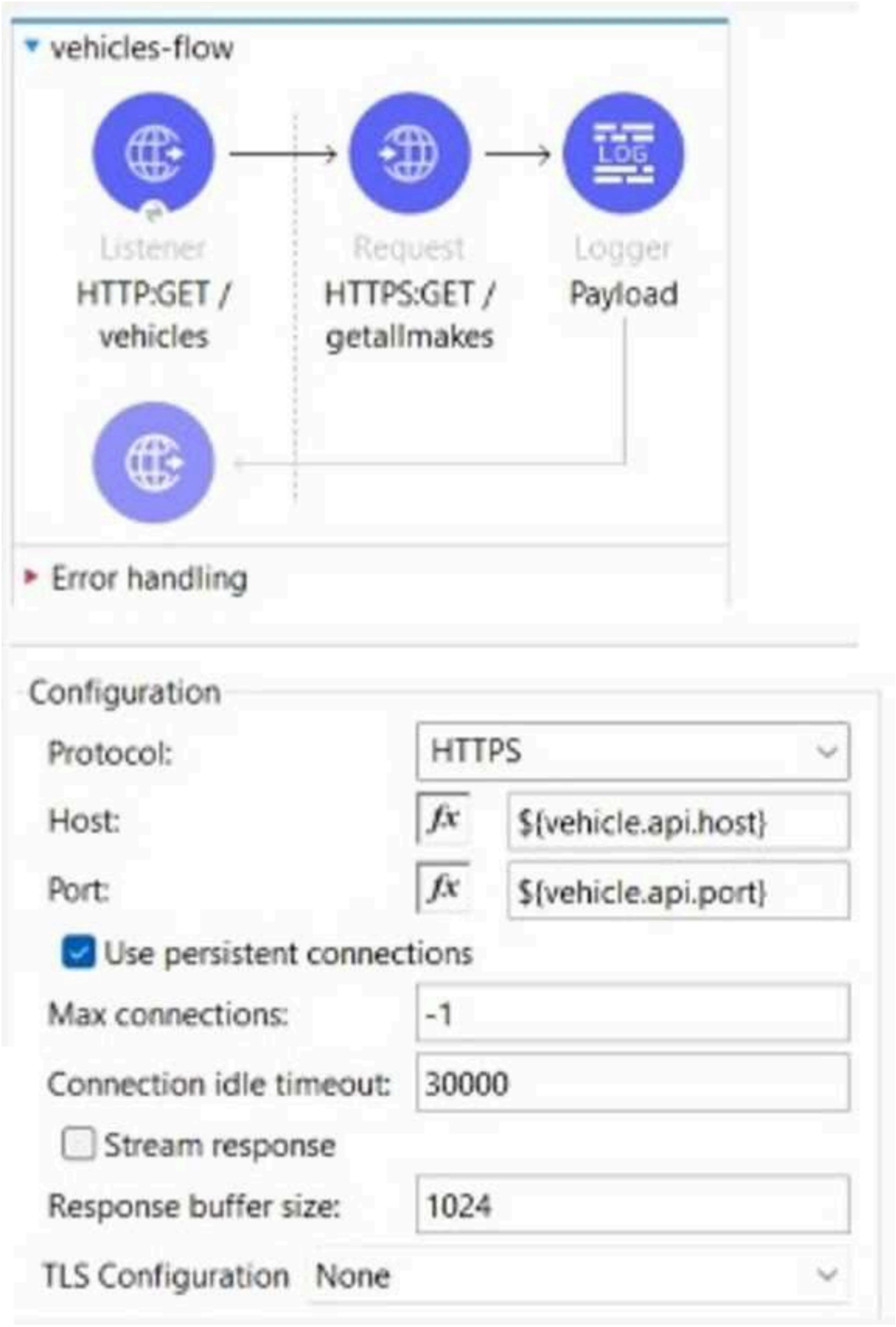
Version: 4.0

Question: 1

The flow is invoicing a target API. The API's protocol is HTTPS. The TLS configuration in the HTTP Request Configuration global element is set to None. A web client submits a request to http:localhost:8081/vehicles.



If the certificate of the target API is signed by a certificate authority (CA), what is true about the HTTP

Request operation when the flow executes?

A. The HTTP Request operation will succeed if the CA'S certificate is present in the JRE's default keystore
B. The HTTP Request operation will succeed if the CA's certificate is present in the JRE's default truststore.
C. The HTTP Request operation will always succeed regardless of the CA
D. The HTTP Request operation will always fail regardless of the CA

Answer: B

E xpl anati on:

The HTTP Request operation will use the default truststore of the JRE to validate the certificate of the target API. If the CA's certificate is present in the truststore, the operation will succeed. Otherwise, it will fail with a handshake exception. Reference: https://docs.mulesoft.com/mule-runtime/4.3/tls-config urati on# tl s-de faul t

Question: 2

When a client and server are exchanging messages during the mTLS handshake, what is being agreed on during the cipher suite exchange?

A. A protocol
B. The TLS version
C. An encryption algorithm
D. The Public key format

Answer: C

E xpl anati on:

A cipher suite is a set of cryptographic algorithms that are used to secure the communication between a client and a server. A cipher suite consists of four components: a key exchange algorithm, an authentication algorithm, an encryption algorithm, and a message authentication code (MAC) algorithm. During the cipher suite exchange, the client and the server agree on which encryption algorithm to use for encrypting and decrypting the data. Reference:
https:/ / docs. m ul e soft. com / m ul e -runti m e / 4. 3/ tl s-config urati on# ci phe r-sui te s

Question: 3

A custom policy needs to be developed to intercept all cutbound HTTP requests made by Mule appl i cati ons.
Which XML element must be used to intercept outbound HTTP requests?

A. It is not possible to intercept outgoing HTTP requests, only inbound requests
B. http-policy:source
C. htt-policy:operation
D. http-policy:processor

Answer: B

E xpl anati on:

The http-policy:processor element is used to intercept outbound HTTP requests made by Mule applications. It allows customizing the request before it is sent to the target API and modifying the response after it is received from the target API. Reference: https://docs.mulesoft.com/api-m anag e r/ 2. x / pol i cy -m ul e 4-custom -pol i cy # pol i cy -x m l -fil e

Question: 4

An API has been built to enable scheduling email provider. The front-end system does very little data entry validation, and problems have started to appear in the email that go to patients. A validate-customer'' flow is added validate the data.
What is he expected behavior of the 'validate-customer'' flow?

```
<flow name="validate-customer">
  <validation:all>
    <validation:is-email email="#[payload.customer.emailAddress]" message="invalid email address">
      <error-mapping sourceType="VALIDATION:INVALID_EMAIL" targetType="SCHEDULE:INVALID_EMAIL_ADDRESS"/>
    </validation:is-email>
    <validation:matches-regex value="#[payload.schedule.appointmentDate]"
      regex="^\d{4}-\d{2}-\d{2}$" message="Invalid appointment date">
      <error-mapping sourceType="VALIDATION:MISMATCH" targetType="SCHEDULE:INVALID_APPOINTMENT_DATE"/>
    </validation:matches-regex>
    <validation:is-not-null value="#[payload.customer.name]" message="Invalid customer name">
      <error-mapping sourceType="VALIDATION:NULL" targetType="SCHEDULE:INVALID_CUSTOMER_NAME"/>
    </validation:is-not-null>
  </validation:all>
</flow>
```

A. If only the email address Is invalid a VALIDATION.INVALID_EMAIL error is raised
B. If the email address is invalid, processing continues to see if the appointment data and customer name are also invalid
C. If the appointment date and customer name are invalid, a SCHEDULE:INVALID_APPOINTMENT_DATE error is raised
D. If all of the values are invalid the last validation error is raised:SCHEDULE:INVALID_CUSTOMER_NAME

Answer: A

E xpl anati on:

The validate-customer flow uses an until-successful scope to validate each field of the customer data. The until-successful scope executes its processors until they succeed or exhausts the maximum number of retries. If any processor fails, it raises an error and stops executing the remaining processors. Therefore, if only the email address is invalid, a VALIDATION.INVALID_EMAIL error is raised and the validation of appointment date and customer name is skipped. Reference: https:/ / docs. m ul e soft. com / m ul e -runti m e / 4. 3/ unti l -succe ssful -scope

Question: 5

When implementing a synchronous API where the event source is an HTTP Listener, a developer needs to return the same correlation ID back to the caller in the HTTP response header.

How can this be achieved?

A. Enable the auto-generate CorrelationID option when scaffolding the flow
B. Enable the CorrelationID checkbox in the HTTP Listener configuration
C. Configure a custom correlation policy
D. NO action is needed as the correlation ID is returned to the caller in the response header by default

Answer: D

E xpl anati on:

When implementing a synchronous API where the event source is an HTTP Listener, Mule automatically propagates some message attributes between flows via outbound and inbound properties. One of these attributes is correlation ID, which is returned to the caller in the response header by default as MULE_CORRELATION_ID. Reference: https://docs.mulesoft.com/mule-runti m e / 4. 3/about-m ul e -m e ssag e # m e ssag e -attri bute s