- **Vendor:** Cisco
- **Exam Code:** 300-920
- **Exam Name:** Developing Applications for Cisco Webex and Webex Devices (DEVWBX)
- **New Updated Questions from Braindump2go (Updated in April/2020)**

QUESTION 12

```
{
    "message": "The user has sent too many requests in a given amount of time. Please refer to the
Retry-After response header to wait before making a new request.",
    "errors": [
        {
            "description" : "The user has sent too many requests in a given amount of time. Please
refer to the Retry-After response header to wait before making a new request."
        }
    ],
    "trackingId": "ROUTER_5D224C5A-C32B-01BB-007E-2D30B4D4007E"
}
```

Refer to the exhibit. What is the Webex Teams REST API HTTP response status code, based on this code snippet?

A. 401
B. 403
C. 429
D. 501

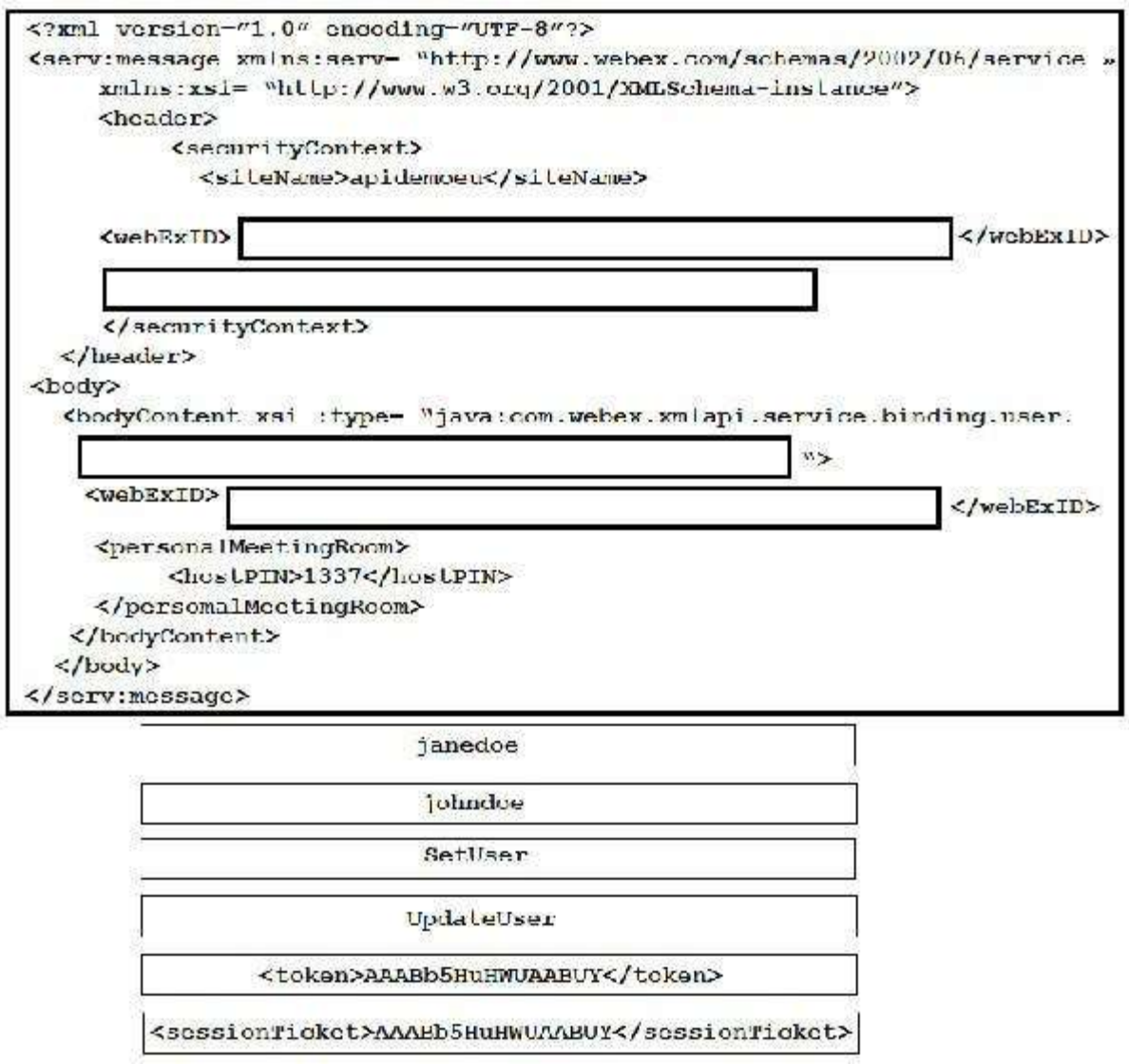
Correct Answer: C
Explanation

Explanation/Reference:
Reference: https://developer.webex.com/docs/api/v1/messages/get-message-details

QUESTION 13
DRAG DROP

Drag and drop the code snippets onto the exhibit to create a valid Webex Meetings API request allowing Jane (an admin) to reset John's PMR pin. Not all options are used.

Select and Place:



Correct Answer:

```
<?xml version="1.0" encoding="UTF-8"?>
<serv:message xmlns:serv= "http://www.webex.com/schemas/2002/06/service "
      xmlns:xsi= "http://www.w3.org/2001/XMLSchema-instance">
      <header>
            <securityContext>
              <siteName>apidemoeu</siteName>

      <webExID>                    johndoe                    </webExID>
                        SetUser
      </securityContext>
  </header>
<body>
  <bodyContent xsi :type= "java:com.webex.xmlapi.service.binding.user.
            <token>AAABb5HuHWUAABUY</token>                    ">
    <webExID>                    UpdateUser                    </webExID>
      <personalMeetingRoom>
            <hostPIN>1337</hostPIN>
      </personalMeetingRoom>
    </bodyContent>
  </body>
</serv:message>
```

janedoe

johndoe

SetUser

UpdateUser

<token>AAABb5HuHWUAABUY</token>

<sessionTicket>AAABb5HuHWUAABUY</sessionTicket>

Explanation

Explanation/Reference:

QUESTION 14

```
1 const webex = windows.webex = Webex.init();
2
3 webex.once('ready' , () => {
4   const jwt = document.getElementById('context').value;
5
6    webex.authorization.requestAccessTokenFromJwt({ jwt })
7        .then(() => {
8            if (webex.canAuthorize) {
9                alert('Authenticated!');
10           }
11       })
12       .catch((err) => {
13           alert('Authentication.error:' +err);
14       });
15 });
```

Refer to the exhibit. On line 4, the script retrieves a context from a DOM element that was generated from a server-side component. How does that server-side component obtain the value for the 'context' element?

A. by opening a dialog asking the end-user to paste his personal access token
B. by completing an authorization code grant flow using the identifier and secret of an OAuth integration
C. by embedding the access token of a Bot account
D. by creating a guest token using the identifier and secret of a Guest Issuer application

Correct Answer: B
Explanation

Explanation/Reference:

QUESTION 15

```
 1  const xapi = require('xapi');
 2  let payload = JSON.stringify({'destination': 'new'});
 3       xapi.command('HttpClient Port', {
 4          Header: "Content-Type: application/json",
 5          Url: "https://192.168.0.2/v1/motion",
 6       },
 7          payload)
 8       .then((result) => {
i 9          console.log(result)
i 10     })
```

Severity ▾ | Enter filter | ☐ Show history

```
01:19:35 Untitled > Loading...
01:19:36 [system] > Starting macros...
01:19:36 [system] > Macros ready.
01:19:36 Untitled > 'Unhandled promise rejection' ( code: 0,
           message: 'HttpClientPostResult',
           data:
           ( status: 'Error',
             Message: 'SSL peer certificate or SSH remote
key was not OK' } }
```

Refer to the exhibit. What causes the error message?

A. xapi must be enabled for promises.
B. HttpClient AllowInsecureHTTPS has not been enabled.
C. The NODE_TLS_REJECT_UNAUTHORIZED environment variable must be set to 0.
D. HttpClient must be changes to HttpsClient.

Correct Answer: B
Explanation

Explanation/Reference:
Reference: https://help.webex.com/en-us/nthg6le/Sending-HTTP-Requests-from-a-Board-Room-or-Desk-Device

QUESTION 16

```
1   const jsxapi = require('jsxapi');
2   const speaker = require('./speaker')
3   const xapi = jsxapi.connect('ssh://10.131.202.230', {
4        username: 'admin',
5        password: ''
6   });
7   let state = 0
8   function toggle(state) {
9        if (state === 0) {return 1}
10       else {return 0}
11  }
12  xapi.on('ready', () => {
13     xapi.event
14        .on('                    ', (x) => {
15           if (x.PanelId === 'panel_1') {
16             state = toggle(state)
17           if (state === 1) {
18              speaker.control({"on": true});
19           } else {
20              speaker.control({"on": false})
21           }
22        }
23     })
24  });
```

Refer to the exhibit. An 'Action Button' with identifier 'panel_1' is deployed to the Touch10 interface of a Room Series device. Which event must be inserted into line 14 to turn the music on/off in the conference room?

A. UserInterface Extensions Event Pressed
B. UserInterface Extensions Panel Clicked
C. UserInterface Extensions Widget GetValue
D. UserInterface Extensions Widget Action

Correct Answer: A
Explanation

Explanation/Reference:
Reference: https://www.cisco.com/c/dam/en/us/td/docs/telepresence/endpoint/ce81/sx-mx-in-room-control-guide-ce81.pdf

QUESTION 17
A Webex Teams bot is deployed but soon it stops responding. Which two explanations are the cause of the issue? (Choose two.)

A. A new webhook was created, which marks the old webhook as inactive.
B. The web server that is set to receive webhooks is not configured to return a 200 message. And the webhook is disabled.
C. The webhook secret is expired and must be refreshed.
D. The refresh token is not being used.
E. The bot owner regenerated the access token on developer.webex.com.

Correct Answer: AB
Explanation

Explanation/Reference:
Reference: https://developer.authorize.net/api/reference/features/webhooks.html

QUESTION 18

```
<xsd:complexType name= "lstRecordingResponse">
     <xsd:complexContent>
          <xsd:extension base= "serv:bodyContentType">
               <xsd:sequence>
                    <xsd:element name= "matchingRecords" type=
"serv:matchingRecordsType" minOccurs="0"/>
                    <xsd:element name= "recording" type= "ep:recordingType" minOccurs=
"0" maxOccurs= "unbounded"/>
...
<xsd:complexType name= "recordingType">
     <xsd:sequence>
          <xsd:element name= "recordingID" type= "xsd:int"/>
          <xsd:element name= "hostWebExID" type= "xsd:string"/>
          <xsd:element name= "name" type= "xsd:string"/>
          <xsd:element name= "createTime" type= "xsd:string"/>
          <xsd:element name= "sreamURL" type= "xsd:string"/>
          <xsd:element name= "fileURL" type= "xsd:string"/>
          <xsd:element name= "password" type= "xsd:string" minOccurs= "0" />
```

Refer to the exhibit. A snippet from the XSD schema of the Webex Meeting XML API 'LstRecordingResponse' element is listed in the exhibit. Assuming that a variable named 'resp' exists that contains the XML response from a successful 'LstRecording' request, which code snippet correctly generates a simple report that lists meeting names and recording file download links?

A.
```
list = (new window.DOMParser()).parseFromString(resp, 'application/json')
for (const item of lst.getElementsByTagNameNS('*', 'matchingRecords')){
document.writeln(
 <p> Meeting Name:${item.getElementsByTagNameNS('*', 'name') [0].textContent}<br>');
document.write(
'Download:${item.getElementsByTagNameNS('*', 'fileURL') [0].textContent}<br>');
```

B.
```
list = (new window.DOMParser()).parseFromString(resp, 'LstRecordingResponse')
for (const item of lst.getElementsByTagNameNS('*', 'matchingRecords')){
document.writeln(
 <p> Meeting Name:${item.getElementsByTagNameNS('*', 'name') [0].textContent}<br>');
document.write(
'Download:${item.getElementsByTagNameNS('*', 'fileURL') [0].textContent}<br>');
```

C.
```
list = (new window.DOMParser()).parseFromString(resp, 'text/xml')
for (const item of lst.getElementsByTagNameNS('*', 'matchingRecords')){
document.writeln(
 <p> Meeting Name:${item.getElementsByTagNameNS('*', 'hostWebExID') [0] }<br>');
document.write(
'Download:${item.getElementsByTagNameNS('*', 'streamURL') [0].textContent}<br>');
```

D.
```
list = (new window.DOMParser()).parseFromString(resp, 'text/xml')
for (const item of lst.getElementsByTagNameNS('*', 'recording')){
document.writeln(
 <p> Meeting Name:${item.getElementsByTagNameNS('*', 'name') [0].textContent}<br>');
document.write(
'Download:${item.getElementsByTagNameNS('*', 'fileURL') [0].textContent}<br>');
```

Correct Answer: A
Explanation

Explanation/Reference:

QUESTION 19
Which xAPI access mechanism requires separate connections for commands and notifications?

A. Serial
B. WebSocket
C. HTTP/HTTPS
D. SSH

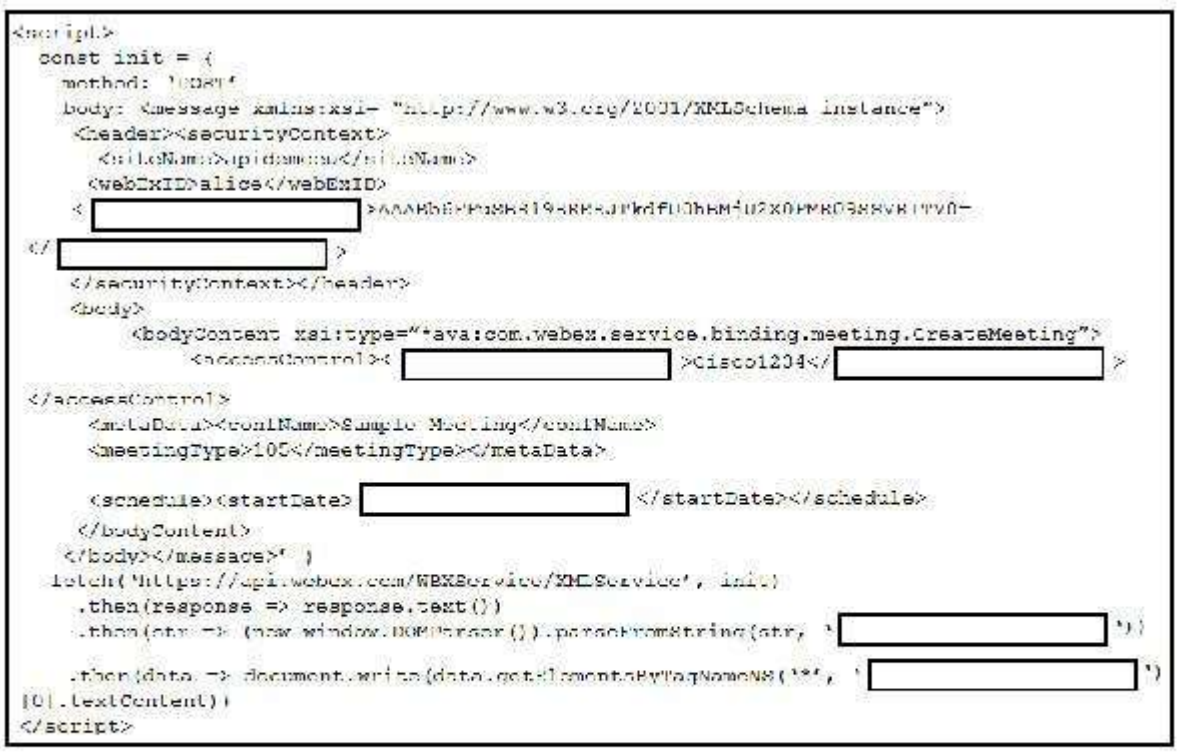
Correct Answer: D
Explanation

Explanation/Reference:
Reference: https://github.com/CiscoDevNet/labs-xapi/blob/master/labs/collab-xapi-intro/5.md

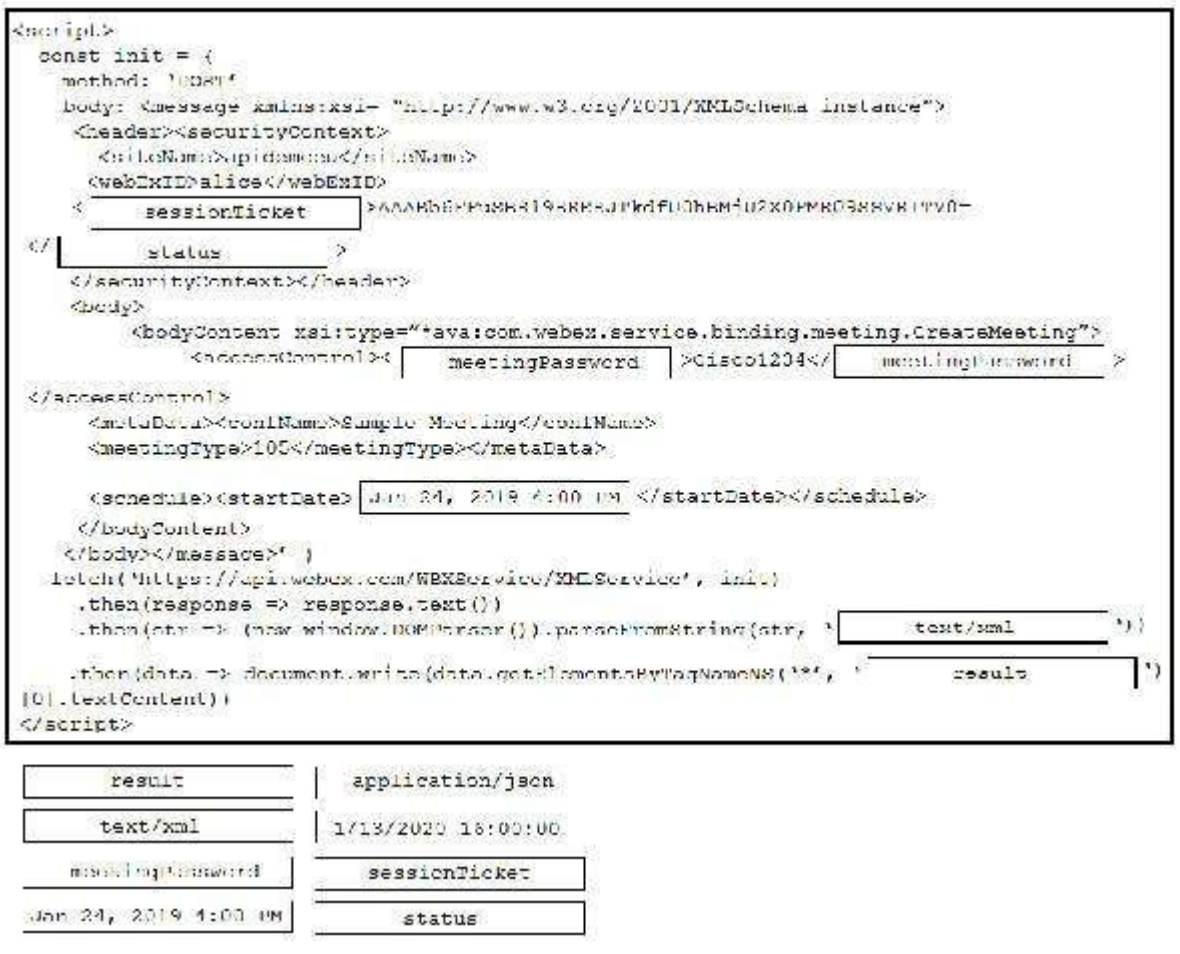
QUESTION 20
DRAG DROP

Drag and drop the code to complete the JavaScript code snippet to create a meeting using the Webex Meetings XML API. Options may be used more than once.

Select and Place:



Correct Answer:



Explanation

Explanation/Reference:

QUESTION 21
Which two capabilities are currently supported by the Webex Meetings XML API? (Choose two.)

A. Request a recording link for playback.
B. Send a text message to the meeting host.
C. Request permissions to schedule on someone else's behalf.
D. Schedule a new meeting.
E. Send a problem report.

Correct Answer: CD
Explanation

Explanation/Reference:
Reference: https://community.cisco.com/t5/cloud-collaboration/api-for-scheduling-a-webex-meeting-on-behalf-of-other-user/td-p/3472319

QUESTION 22
Which two statements about Webex Teams refresh tokens are true? (Choose two.)

A. The refresh token is useless without the client ID and client secret.
B. An attacker can use the refresh token to send messages on behalf of the user.
C. The refresh token is used to generate a new access token.
D. A new refresh token cannot be granted until the client ID is invalidated.
E. The refresh token does not expire.

Correct Answer: BC
Explanation

Explanation/Reference:
Reference: https://auth0.com/learn/refresh-tokens/