

AZ-220.VCEplus.premium.exam.60q

Number: AZ-220  
Passing Score: 800  
Time Limit: 120 min  
File Version: 1.0



**Website:** <https://vceplus.com>

**VCE to PDF Converter:** <https://vceplus.com/vce-to-pdf/>

**Facebook:** <https://www.facebook.com/VCE.For.All.VN/>

**Twitter :** [https://twitter.com/VCE\\_Plus](https://twitter.com/VCE_Plus)

**AZ-220**

**Microsoft Azure IoT Developer (beta)**



## Testlet 1

### Case Study

This is a case study. **Case studies are not timed separately. You can use as much exam time as you would like to complete each case.** However, there may be additional case studies and sections on this exam. You must manage your time to ensure that you are able to complete all questions included on this exam in the time provided.

To answer the questions included in a case study, you will need to reference information that is provided in the case study. Case studies might contain exhibits and other resources that provide more information about the scenario that is described in the case study. Each question is independent of the other question on this case study.

At the end of this case study, a review screen will appear. This screen allows you to review your answers and to make changes before you move to the next sections of the exam. After you begin a new section, you cannot return to this section.

#### To start the case study

To display the first question on this case study, click the **Next** button. Use the buttons in the left pane to explore the content of the case study before you answer the questions. Clicking these buttons displays information such as business requirements, existing environment, and problem statements. If the case study has an **All Information** tab, note that the information displayed is identical to the information displayed on the subsequent tabs. When you are ready to answer a question, click the **Question** button to return to the question.

#### Existing Environment. Current State of Development

Contoso produces a set of Bluetooth sensors that read the temperature and humidity. The sensors connect to IoT gateway devices that relay the data.

All the IoT gateway devices connect to an Azure IoT hub named iothub1.

#### Existing Environment. Device Twin

You plan to implement device twins by using the following JSON sample.



```
{
  "deviceId": "device_n",
  "etag": "AAAAAAAAAAQ=",
  "deviceEtag": "NDcwMTU4Mzk=",
  "status": "enabled",
  "statusUpdateTime": "0001-01-01T00:00:00Z",
  "connectionState": "Disconnected",
  "lastActivityTime": "0001-01-01T00:00:00Z",
  "cloudToDeviceMessageCount": 0,
  "authenticationType": "sas",
  "x509Thumbprint": {
    "primaryThumbprint": null,
    "secondaryThumbprint": null
  },
  "version": 11,
  "properties": {
    "desired": {
      "fanSpeed": 70,
      "$metadata": {
        "$lastUpdated": "2019-10-16T09:43:42.2944169Z",
        "$lastUpdatedVersion": 4,
        "fanSpeed": {
          "$lastUpdated": "2019-10-16T09:43:42.2944169Z",
          "$lastUpdatedVersion": 4
        }
      }
    },
    "$version": 4
  },
  "reported": {
    "fanSpeed": 80,
    "metadata": {
      "$lastUpdated": "2019-10-16T09:43:42.4035171Z",
      "fanSpeed": {
        "$lastUpdated": "2019-10-16T09:43:42.4035171Z"
      }
    }
  },
  "$version": 7
}
},
"capabilities": {
  "lotEdge": false
}
}
```



#### Existing Environment. Azure Stream Analytics

Each room will have between three to five sensors that will generate readings that are sent to a single IoT gateway device. The IoT gateway device will forward all the readings to iotHub1 at intervals of between 10 and 60 seconds.

You plan to use a gateway pattern so that each IoT gateway device will have its own IoT Hub device identity.

You draft the following query, which is missing the `GROUP BY` clause.

```
SELECT
    AVG(temperature),
    System.TimeStamp() AS AsaTime
FROM
    Iothub
```

You plan to use a 30-second period to calculate the average temperature reading of the sensors.

You plan to minimize latency between the condition reported by the sensors and the corresponding alert issued by the Stream Analytics job.

#### Existing Environment. Device Messages

The IoT gateway devices will send messages that contain the following JSON data whenever the temperature exceeds a specified threshold.

```
{
  "event": {
    "payload": "Temperature = 26.23 Humidity = 78.70597746416186 Button = 0",
    "properties": {
      "application": {
        "level": "critical"
      }
    }
  }
}
```

The `level` property will be used to route the messages to an Azure Service Bus queue endpoint named `criticalep`.

#### Existing Environment. Issues

You discover connectivity issues between the IoT gateway devices and `iothub1`, which cause IoT devices to lose connectivity and messages.

#### Requirements. Planning Changes Contoso

plans to make the following changes:

- Use Stream Analytics to process and view data.
- Use Azure Time Series Insights to visualize data.
- Implement a system to sync device statuses and required settings.
- Add extra information to messages by using message enrichment.
- Create a notification system to send an alert if a condition exceeds a specified threshold.
- Implement a system to identify what causes the intermittent connection issues and lost messages.

#### Requirements. Technical Requirements

Contoso must meet the following requirements:

- Use the built-in functions of IoT Hub whenever possible.
- Minimize hardware and software costs whenever possible.
- Minimize administrative effort to provision devices at scale.
- Implement a system to trace message flow to and from `iothub1`.
- Minimize the amount of custom coding required to implement the planned changes.
- Prevent read operations from being negatively affected when you implement additional services.

#### QUESTION 1

HOTSPOT

You create a new IoT device named device1 on iothub1. Device1 has a primary key of Uihuih76hbHb.

How should you complete the device connection string? To answer, select the appropriate options in the answer area.

**NOTE:** Each correct selection is worth one point.

**Hot Area:**

**Correct Answer:**

**Section: [none]**

**Explanation**

**Explanation/Reference:**

Explanation:

Box 1: iothub1

The Azure IoT hub is named iothub1.

Box 2: azure-devices.net

The format of the device connection string looks like:

HostName={YourIoTHubName}.azure-devices.net;DeviceId=MyNodeDevice;SharedAccessKey={YourSharedAccessKey}

Box 1: device1

Device1 has a primary key of Uihuih76hbHb.

Reference:

<https://docs.microsoft.com/en-us/azure/iot-hub/quickstart-control-device-dotnet>



## Question Set 2

### QUESTION 1

**Note:** This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution.

**After you answer a question in this question, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.**

You have an Azure IoT solution that includes an Azure IoT hub, a Device Provisioning Service instance, and 1,000 connected IoT devices.

All the IoT devices are provisioned automatically by using one enrollment group.

You need to temporarily disable the IoT devices from the connecting to the IoT hub.

Solution: From the Device Provisioning Service, you disable the enrollment group, and you disable device entries in the identity registry of the IoT hub to which the IoT devices are provisioned.

Does the solution meet the goal?

- A. Yes
- B. No

**Correct Answer:** A

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

You may find it necessary to deprovision devices that were previously auto-provisioned through the Device Provisioning Service.

In general, deprovisioning a device involves two steps:

1. Disenroll the device from your provisioning service, to prevent future auto-provisioning. Depending on whether you want to revoke access temporarily or permanently, you may want to either disable or delete an enrollment entry.
2. Deregister the device from your IoT Hub, to prevent future communications and data transfer. Again, you can temporarily disable or permanently delete the device's entry in the identity registry for the IoT Hub where it was provisioned.

Reference: <https://docs.microsoft.com/bs-latn-ba/azure/iot-dps/how-to-unprovision-devices>

### QUESTION 2

**Note:** This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution.

**After you answer a question in this question, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.**

You have an Azure IoT solution that includes an Azure IoT hub, a Device Provisioning Service instance, and 1,000 connected IoT devices.

All the IoT devices are provisioned automatically by using one enrollment group.

You need to temporarily disable the IoT devices from the connecting to the IoT hub.

Solution: You delete the enrollment group from the Device Provisioning Service.

Does the solution meet the goal?

- A. Yes
- B. No

**Correct Answer:** B

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

Instead, from the Device Provisioning Service, you disable the enrollment group, and you disable device entries in the identity registry of the IoT hub to which the IoT devices are provisioned.

Reference: <https://docs.microsoft.com/bs-latn-ba/azure/iot-dps/how-to-unprovision-devices>

### QUESTION 3

**Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution.**

**After you answer a question in this question, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.**

You have an Azure IoT solution that includes an Azure IoT hub, a Device Provisioning Service instance, and 1,000 connected IoT devices.

All the IoT devices are provisioned automatically by using one enrollment group.

You need to temporarily disable the IoT devices from the connecting to the IoT hub.

Solution: From the IoT hub, you change the credentials for the shared access policy of the IoT devices.

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer: B**

**Section: [none]**

**Explanation**

**Explanation/Reference:**

Reference: <https://docs.microsoft.com/bs-latn-ba/azure/iot-dps/how-to-unprovision-devices>



### QUESTION 4 HOTSPOT

You have an Azure IoT hub.

You plan to deploy 1,000 IoT devices by using automatic device management.

The device twin is shown below.



```
{
  "deviceId": "ContosoHyperDriveEngine1",
  "etag": "AAAAAAAAAAw=",
  "deviceEtag": "MTYyNDk20kw",
  "status": "enabled",
  "statusUpdateTime": "0001-01-01t00:00:00Z",
  "connectionTime": "Disconnected",
  "lastActivityTime": "0001-01-01T00:00:00Z",
  "cloudToDeviceMessageCount": 0,
  "authenticationType": "sas",
  "x509Thumbprint": {
    "primaryThumbprint": null,
    "secondaryThumbprint": null
  },
  "version": 13,
  "tags": {
    "engine": {
      "warpCorVersion": "1.2.65b",
      "warpDriveType": "WM105a"
    }
  },
  "properties": {
    "desired": {
      "$metadata": {
        "$lastUpdated": "2019-10-17T18:43:33.7599556Z"
      },
      "$version": 1
    },
    "reported": {
      "$metadata": {
        "$lastUpdated": "2019-10-17T18:43:33.7599556Z"
      },
      "$version": 1
    }
  }
}
```



You need to configure automatic device management for the deployment.

Which target Condition and Device Twin Path should you use? To answer, select the appropriate options in the answer area.

**NOTE:** Each correct selection is worth one point.

**Hot Area:**



### Answer Area

Target Condition:

properties.desired.warpDriveType='WM105a'
properties.reported.warpDriveType='WM105a'
tags.engine.warpDriveType='WM105a'

Device Twin Path:

properties.desired.warpOperating
properties.reported.warpOperating
properties.warpOperating

Correct Answer:

### Answer Area

Target Condition:

properties.desired.warpDriveType='WM105a'
properties.reported.warpDriveType='WM105a'
tags.engine.warpDriveType='WM105a'

Device Twin Path:

properties.desired.warpOperating
properties.reported.warpOperating
properties.warpOperating

Section: [none]

Explanation

Explanation/Reference:

Explanation:

Box 1: tags.engine.warpDriveType='WM105a'

Use tags to target twins. Before you create a configuration, you must specify which devices or modules you want to affect. Azure IoT Hub identifies devices and using tags in the device twin, and identifies modules using tags in the module twin.

Box 2: properties.desired.warpOperating

The twin path, which is the path to the JSON section within the twin desired properties that will be set.

For example, you could set the twin path to properties.desired.chiller-water and then provide the following JSON content:

```
{  
  "temperature": 66,  
  "pressure": 28  
}
```

Reference: <https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-automatic-device-management>

**QUESTION 5** You plan to deploy a standard tier Azure IoT hub.

You need to perform an over-the-air (OTA) update on devices that will connect to the IoT hub by using scheduled jobs.

What should you use?

- A. a device-to-cloud message
- B. the device twin reported properties
- C. a cloud-to-device message
- D. a direct method

**Correct Answer:** D

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

Releases via the REST API.

All of the operations that can be performed from the Console can also be automated using the REST API. You might do this to automate your build and release process, for example.

You can build firmware using the Particle CLI or directly using the compile source code API.

Note: Over-the-air (OTA) firmware updates are a vital component of any IoT system. Over-the-air firmware updates refers to the practice of remotely updating the code on an embedded device.

Reference:

<https://docs.particle.io/tutorials/device-cloud/ota-updates/>

**QUESTION 6** You have an IoT device that gathers data in a CSV file named Sensors.csv.

You deploy an Azure IoT hub that is accessible at ContosoHub.azure-devices.net.

You need to ensure that Sensors.csv is uploaded to the IoT hub.

Which two actions should you perform? Each correct answer presents part of the solution.

**NOTE:** Each correct selection is worth one point.

- A. Upload Sensors.csv by using the IoT Hub REST API.
- B. From the Azure subscription, select the IoT hub, select **Message routing**, and then configure a route to storage.
- C. From the Azure subscription, select the IoT hub, select **File upload**, and then configure a storage container.
- D. Configure the device to use a `GET` request to ContosoHub.azure-devices.net/devices/ContosoDevice1/files/notifications.

**Correct Answer:** AC

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

C: To use the file upload functionality in IoT Hub, you must first associate an Azure Storage account with your hub. Select File upload to display a list of file upload properties for the IoT hub that is being modified.

For Storage container: Use the Azure portal to select a blob container in an Azure Storage account in your current Azure subscription to associate with your IoT Hub. If necessary, you can create an Azure Storage account on the Storage accounts blade and blob container on the Containers

A: IoT Hub has an endpoint specifically for devices to request a SAS URI for storage to upload a file. To start the file upload process, the device sends a POST request to {iot hub}.azure-devices.net/devices/{deviceId}/files with the following JSON body:

```
{
  "blobName": "{name of the file for which a SAS URI will be generated}"
}
```

Incorrect Answers:

D: Deprecated: initialize a file upload with a GET. Use the POST method instead.

Reference: <https://github.com/MicrosoftDocs/azure-docs/blob/master/articles/iot-hub/iot-hub-configure-file-upload.md>

**QUESTION 7** You plan to deploy an Azure IoT hub.

The IoT hub must support the following:

- Three Azure IoT Edge devices
- 2,500 IoT devices

Each IoT device will spend a 6 KB message every five seconds.

You need to size the IoT hub to support the devices. The solution must minimize costs.

What should you choose?

- A. one unit of the S1 tier
- B. one unit of the B2 tier
- C. one unit of the B1 tier
- D. one unit of the S3 tier

**Correct Answer:** D

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

$2500 * 6 \text{ KB} * 12 = 180,000 \text{ KB/minute} = 180 \text{ MB/Minute}$ .

B3, S3 can handle up to 814 MB/minute per unit.

Incorrect Answers:

A, C: B1, S1 can only handle up to 1111 KB/minute per unit

B: B2, S2 can only handle up to 16 MB/minute per unit.

Reference: <https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-scaling>

**QUESTION 8**

DRAG DROP

You deploy an Azure IoT hub.

You need to demonstrate that the IoT hub can receive messages from a device.



Which three actions should you perform in sequence? To answer, move the appropriate actions from the list of actions to the answer area and arrange them in the correct order.

**Select and Place:**

**Correct Answer:**

**Section: [none]**

**Explanation**

**Explanation/Reference:**

Explanation:

Step 1: Register a device in IoT Hub

Before you can use your IoT devices with Azure IoT Edge, you must register them with your IoT hub. Once a device is registered, you can retrieve a connection string to set up your device for IoT Edge workloads.

Step 2: Configure the device connection string on a device client.

When you're ready to set up your device, you need the connection string that links your physical device with its identity in the IoT hub.

Step 3: Trigger a new send event from a device client.

Reference: <https://docs.microsoft.com/en-us/azure/iot-edge/how-to-register-device>

## QUESTION 9

DRAG DROP

You have an Azure IoT hub.

You plan to attach three types of IoT devices as shown in the following table.

Name	Specification	Note
Transparent Field Gateway Device	High-power device with a fast processor and 4 GB of RAM	Will connect to multiple devices, each with its own credentials, by using the same TLS connection.
Low Resource Device	Low resource specifications, battery-operated, and 512 KB of RAM	Will connect directly to an IoT hub and will <b>NOT</b> connect to any other devices. Will use cloud-to-device messages.
Limited Sensor Device	Extremely low-power device with a limited microcontroller (MCU) and 256 KB of RAM	Will <b>NOT</b> support the Azure SDK. Messages must be as small as possible.



You need to select the appropriate communication protocol for each device.

What should you select? To answer, drag the appropriate protocols to the correct devices. Each protocol may be used once, more than once, or not at all. You may need to drag the split bar between panes or scroll to view content.

**NOTE:** Each correct selection is worth one point.

**Select and Place:**

**Correct Answer:**

**Section: [none]**

**Explanation**

**Explanation/Reference:**

Explanation:

Box 1: AMQP

Use AMQP on field and cloud gateways to take advantage of connection multiplexing across devices.

Box 2: MQTT

MQTT is used on all devices that do not require to connect multiple devices (each with its own per-device credentials) over the same TLS connection.

Box 3: HTTPS

Use HTTPS for devices that cannot support other protocols.

Reference: <https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-devguide-protocols>

**QUESTION 10** You create an Azure IoT hub by running the following command.

```
az iot hub create --resource-group MyResourceGroup --name MyIotHub --sku B1 --location westus --partition-count 4
```

What does MyIotHub support?

- A. Device Provisioning Service
- B. cloud-to-device messaging
- C. Azure IoT Edge
- D. device twins

**Correct Answer:** A

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

The Device Provisioning Service is included in the Basic Tiers (such as B1).

Incorrect Answers:

B, C, D: The Standard tier is needed for cloud-to-device messaging, Azure IoT Edge, and device twins.

Reference: <https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-scaling>

**QUESTION 11** You have an existing Azure IoT hub.

You need to connect physical IoT devices to the IoT hub.

You are connecting the devices through a firewall that allows only port 443 and port 80.

Which three communication protocols can you use? Each correct answer presents a complete solution.

**NOTE:** Each correct selection is worth one point.

- A. MQTT over WebSocket
- B. AMQP
- C. AMQP over WebSocket
- D. MQTT
- E. HTTPS

**Correct Answer:** ACE

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:



MQTT over WebSockets, AMQP over WebSocket, and HTTPS use port 443.

Reference: <https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-devguide-protocols>

**QUESTION 12** You have an Azure IoT solution that includes an Azure IoT hub and 100 Azure IoT Edge devices.

You plan to deploy the IoT Edge devices to external networks. The firewalls of the external networks only allow traffic on port 80 and port 443.

You need to ensure that the devices can connect to the IoT hub. The solution must minimize costs.

What should you do?

- A. Configure the devices for extended offline operations.
- B. Configure the upstream protocol of the devices to use MQTT over WebSocket.
- C. Connect the external networks to the IoT solution by using ExpressRoute.
- D. Configure the devices to use an HTTPS proxy.

**Correct Answer:** B

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

MQTT over WebSockets uses port 443.

Reference:

<https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-devguide-protocols>



## Testlet 1

### Case Study

This is a case study. **Case studies are not timed separately. You can use as much exam time as you would like to complete each case.** However, there may be additional case studies and sections on this exam. You must manage your time to ensure that you are able to complete all questions included on this exam in the time provided.

To answer the questions included in a case study, you will need to reference information that is provided in the case study. Case studies might contain exhibits and other resources that provide more information about the scenario that is described in the case study. Each question is independent of the other question on this case study.

At the end of this case study, a review screen will appear. This screen allows you to review your answers and to make changes before you move to the next sections of the exam. After you begin a new section, you cannot return to this section.

#### To start the case study

To display the first question on this case study, click the **Next** button. Use the buttons in the left pane to explore the content of the case study before you answer the questions. Clicking these buttons displays information such as business requirements, existing environment, and problem statements. If the case study has an **All Information** tab, note that the information displayed is identical to the information displayed on the subsequent tabs. When you are ready to answer a question, click the **Question** button to return to the question.

#### Existing Environment. Current State of Development

Contoso produces a set of Bluetooth sensors that read the temperature and humidity. The sensors connect to IoT gateway devices that relay the data.

All the IoT gateway devices connect to an Azure IoT hub named iothub1.

#### Existing Environment. Device Twin

You plan to implement device twins by using the following JSON sample.





```
{
  "deviceId": "device_n",
  "etag": "AAAAAAAAAAQ=",
  "deviceEtag": "NDcwMTU4Mzk=",
  "status": "enabled",
  "statusUpdateTime": "0001-01-01T00:00:00Z",
  "connectionState": "Disconnected",
  "lastActivityTime": "0001-01-01T00:00:00Z",
  "cloudToDeviceMessageCount": 0,
  "authenticationType": "sas",
  "x509Thumbprint": {
    "primaryThumbprint": null,
    "secondaryThumbprint": null
  },
  "version": 11,
  "properties": {
    "desired": {
      "fanSpeed": 70,
      "$metadata": {
        "$lastUpdated": "2019-10-16T09:43:42.2944169Z",
        "$lastUpdatedVersion": 4,
        "fanSpeed": {
          "$lastUpdated": "2019-10-16T09:43:42.2944169Z",
          "$lastUpdatedVersion": 4
        }
      }
    },
    "$version": 4
  },
  "reported": {
    "fanSpeed": 80,
    "metadata": {
      "$lastUpdated": "2019-10-16T09:43:42.4035171Z",
      "fanSpeed": {
        "$lastUpdated": "2019-10-16T09:43:42.4035171Z"
      }
    },
    "$version": 7
  }
},
"capabilities": {
  "lotEdge": false
}
}
```



#### Existing Environment. Azure Stream Analytics

Each room will have between three to five sensors that will generate readings that are sent to a single IoT gateway device. The IoT gateway device will forward all the readings to iotHub1 at intervals of between 10 and 60 seconds.

You plan to use a gateway pattern so that each IoT gateway device will have its own IoT Hub device identity.

You draft the following query, which is missing the `GROUP BY` clause.

```
SELECT
    AVG(temperature),
    System.TimeStamp() AS AsaTime
FROM
    Iothub
```

You plan to use a 30-second period to calculate the average temperature reading of the sensors.

You plan to minimize latency between the condition reported by the sensors and the corresponding alert issued by the Stream Analytics job.

#### Existing Environment. Device Messages

The IoT gateway devices will send messages that contain the following JSON data whenever the temperature exceeds a specified threshold.

```
{
  "event": {
    "payload": "Temperature = 26.23 Humidity = 78.70597746416186 Button = 0",
    "properties": {
      "application": {
        "level": "critical"
      }
    }
  }
}
```

The `level` property will be used to route the messages to an Azure Service Bus queue endpoint named `criticalep`.

#### Existing Environment. Issues

You discover connectivity issues between the IoT gateway devices and `iothub1`, which cause IoT devices to lose connectivity and messages.

#### Requirements. Planning Changes Contoso

plans to make the following changes:

- Use Stream Analytics to process and view data.
- Use Azure Time Series Insights to visualize data.
- Implement a system to sync device statuses and required settings.
- Add extra information to messages by using message enrichment.
- Create a notification system to send an alert if a condition exceeds a specified threshold.
- Implement a system to identify what causes the intermittent connection issues and lost messages.

#### Requirements. Technical Requirements

Contoso must meet the following requirements:

- Use the built-in functions of IoT Hub whenever possible.
- Minimize hardware and software costs whenever possible.
- Minimize administrative effort to provision devices at scale.
- Implement a system to trace message flow to and from `iothub1`.
- Minimize the amount of custom coding required to implement the planned changes.
- Prevent read operations from being negatively affected when you implement additional services.

**QUESTION 1** What should you do to identify the cause of the connectivity issues? A. Send cloud-to-device messages to the IoT devices.

- B. Use the heartbeat pattern to send messages from the IoT devices to iothub1.
- C. Monitor the connection status of the device twin by using an Azure function.
- D. Enable the collection of the Connections diagnostics logs and set up alerts for the connected devices count metric.

**Correct Answer:** D

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

Scenario: You discover connectivity issues between the IoT gateway devices and iothub1, which cause IoT devices to lose connectivity and messages.

To log device connection events and errors, turn on diagnostics for IoT Hub. We recommend turning on these logs as early as possible, because if diagnostic logs aren't enabled, when device disconnects occur, you won't have any information to troubleshoot the problem with.

Step 1:

1. Sign in to the Azure portal.
2. Browse to your IoT hub.
3. Select Diagnostics settings.
4. Select Turn on diagnostics.
5. Enable Connections logs to be collected.
6. For easier analysis, turn on Send to Log Analytics (see pricing).

Step 2:

Set up alerts for device disconnect at scale

To get alerts when devices disconnect, configure alerts on the Connected devices (preview) metric.

Reference:

<https://docs.microsoft.com/bs-cyrl-ba/azure/iot-hub/iot-hub-troubleshoot-connectivity>



## Question Set 2

### QUESTION 1

**Note:** This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution.

**After you answer a question in this question, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.**

You have devices that connect to an Azure IoT hub. Each device has a fixed GPS location that includes latitude and longitude.

You discover that a device entry in the identity registry of the IoT hub is missing the GPS location.

You need to configure the GPS location for the device entry. The solution must prevent the changes from being propagated to the physical device.

Solution: You use an Azure policy to apply tags to a resource group.

Does the solution meet the goal?

- A. Yes
- B. No

**Correct Answer:** B

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

Instead add the desired properties to the device twin.

Note: Device Twins are used to synchronize state between an IoT solution's cloud service and its devices. Each device's twin exposes a set of desired properties and reported properties. The cloud service populates the desired properties with values it wishes to send to the device. When a device connects it requests and/or subscribes for its desired properties and acts on them.

Reference: <https://azure.microsoft.com/sv-se/blog/deep-dive-into-azure-iot-hub-notifications-and-device-twin/>

### QUESTION 2

**Note:** This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution.

**After you answer a question in this question, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.**

You have devices that connect to an Azure IoT hub. Each device has a fixed GPS location that includes latitude and longitude.

You discover that a device entry in the identity registry of the IoT hub is missing the GPS location.

You need to configure the GPS location for the device entry. The solution must prevent the changes from being propagated to the physical device.

Solution: You add tags to the device twin.

Does the solution meet the goal?

- A. Yes
- B. No

**Correct Answer:** B

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

Instead add the desired properties to the device twin.

Note: Device Twins are used to synchronize state between an IoT solution's cloud service and its devices. Each device's twin exposes a set of desired properties and reported properties. The cloud service populates the desired properties with values it wishes to send to the device. When a device connects it requests and/or subscribes for its desired properties and acts on them.

Reference: <https://azure.microsoft.com/sv-se/blog/deep-dive-into-azure-iot-hub-notifications-and-device-twin/>

### QUESTION 3

**Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution.**

**After you answer a question in this question, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.**

You have devices that connect to an Azure IoT hub. Each device has a fixed GPS location that includes latitude and longitude.

You discover that a device entry in the identity registry of the IoT hub is missing the GPS location.

You need to configure the GPS location for the device entry. The solution must prevent the changes from being propagated to the physical device.

Solution: You add the desired properties to the device twin.

Does the solution meet the goal?

- A. Yes
- B. No

**Correct Answer: A**

**Section: [none]**

**Explanation**

**Explanation/Reference:**

Explanation:

Device Twins are used to synchronize state between an IoT solution's cloud service and its devices. Each device's twin exposes a set of desired properties and reported properties. The cloud service populates the desired properties with values it wishes to send to the device. When a device connects it requests and/or subscribes for its desired properties and acts on them.

Reference: <https://azure.microsoft.com/sv-se/blog/deep-dive-into-azure-iot-hub-notifications-and-device-twin/>

### QUESTION 4

You have three Azure IoT hubs named Hub1, Hub2, and Hub3, a Device Provisioning Service instance, and an IoT device named Device1.

Each IoT hub is deployed to a separate Azure region.

Device enrollment uses the Lowest latency allocation policy.

The Device Provisioning Service uses the Lowest latency allocation policy.

Device1 is auto-provisioned to Hub1 by using the Device Provisioning Service.

Device1 regularly moves between regions.

You need to ensure that Device1 always connects to the IoT hub that has the lowest latency.

What should you do?

- A. Configure device attestation that uses X.509 certificates.
- B. Implement device certificate rolling.
- C. Disenroll and reenroll Device1.
- D. Configure the re-provisioning policy.

**Correct Answer: D**

**Section: [none]**

**Explanation**

**Explanation/Reference:**

Explanation:

Automated re-provisioning support.

Microsoft added first-class support for device re-provisioning which allows devices to be reassigned to a different IoT solution sometime after the initial solution assignment. Re-provisioning support is available in two options:

- Factory reset, in which the device twin data for the new IoT hub is populated from the enrollment list instead of the old IoT hub. This is common for factory reset scenarios as well as leased device scenarios.
- Migration, in which device twin data is moved from the old IoT hub to the new IoT hub. This is common for scenarios in which a device is moving between geographies.

Reference: <https://azure.microsoft.com/en-us/blog/new-year-newly-available-iot-hub-device-provisioning-service-features/>

**QUESTION 5** You have an Azure subscription that contains a resource group named RG1.

You need to deploy the Device Provisioning Service. The solution must ensure that the Device Provisioning Service can accept new device enrollments.

You create a Device Provisioning Service instance.

Which two actions should you perform next? Each correct answer presents part of the solution.

**NOTE:** Each correct selection is worth one point.

- A. From the **Linked IoT hubs** blade of the Device Provisioning Service, link an Azure IoT hub.
- B. From the Azure portal, create a new Azure IoT hub.
- C. From the Manage allocation policy blade of the Device Provisioning Service, configure an allocation policy.
- D. From the Certificates blade of the Device Provisioning Service, upload an X.509 certificate to the Device Provisioning Service.

**Correct Answer:** D

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

A: The Device Provisioning Service can only provision devices to IoT hubs that have been linked to it.

C: Allocation policy. The service-level setting that determines how Device Provisioning Service assigns devices to an IoT hub. There are three supported allocation policies: ▪

Lowest latency: devices are provisioned to an IoT hub with the lowest latency to the device.

- Evenly weighted distribution
- Static configuration via the enrollment list

Reference: <https://docs.microsoft.com/bs-latn-ba/azure/iot-dps/concepts-service>

**QUESTION 6**

You have 10,000 IoT devices that connect to an Azure IoT hub. The devices do not support over-the-air (OTA) updates.

You need to decommission 1,000 devices. The solution must prevent connections and autoenrollment for the decommissioned devices.

Which two actions should you perform? Each correct answer presents part of the solution.

**NOTE:** Each correct selection is worth one point.

- A. Update the `connectionState` device twin property on all the devices.
- B. Blacklist the X.509 root certification authority (CA) certificate for the enrollment group.
- C. Delete the enrollment entry for the devices.
- D. Remove the identity certificate from the hardware security module (HSM) of the devices.
- E. Delete the device identity from the device registry of the IoT hub.

**Correct Answer:** BC

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

B: X.509 certificates are typically arranged in a certificate chain of trust. If a certificate at any stage in a chain becomes compromised, trust is broken. The certificate must be blacklisted to prevent Device Provisioning Service from provisioning devices downstream in any chain that contains that certificate.

C: Individual enrollments apply to a single device and can use either X.509 certificates or SAS tokens (in a real or virtual TPM) as the attestation mechanism. (Devices that use SAS tokens as their attestation mechanism can be provisioned only through an individual enrollment.) To blacklist a device that has an individual enrollment, you can either disable or delete its enrollment entry.

To blacklist a device that has an individual enrollment, you can either disable or delete its enrollment entry.

Reference: <https://docs.microsoft.com/en-us/azure/iot-dps/how-to-revoke-device-access-portal>

**QUESTION 7 HOTSPOT**

You have an Azure IoT Central application that has a custom device template.

You need to configure the device template to support the following activities:

- Return the reported power consumption.
- Configure the desired fan speed.
- Run the device reset routine. ▪

Read the fan serial number.

Which option should you use for each activity? Each correct answer presents part of the solution.

**NOTE:** Each correct selection is worth one point.

**Hot Area:**

**Correct Answer:**

**Section: [none]**

**Explanation**

**Explanation/Reference:**

Explanation:

Box 1: Measurement

Telemetry/measurement is a stream of values sent from the device, typically from a sensor. For example, a sensor might report the ambient temperature.

Box 2: Property

The template can provide a writeable fan speed property

Properties represent point-in-time values. For example, a device can use a property to report the target temperature it's trying to reach. You can set writeable properties from IoT Central. Box

3: Settings

Box 4: Command

You can call device commands from IoT Central. Commands optionally pass parameters to the device and receive a response from the device. For example, you can call a command to reboot a device in 10 seconds.

Reference: <https://docs.microsoft.com/en-us/azure/iot-central/core/howto-set-up-template>

**QUESTION 8**

**DRAG DROP**

You have an Azure IoT Central application that includes a Device Provisioning Service instance.

You need to connect IoT devices to the application without first registering the devices.

In which order should you perform the actions? To answer, move all actions from the list of actions to the answer area and arrange them in the correct order.





**Select and Place:**

**Correct Answer:**

**Section: [none]**

**Explanation**

**Explanation/Reference:**

Explanation:

Step: With DPS (Device Provisioning Service) you can generate device credentials and configure the devices offline without registering the devices through IoT Central UI.

Connect devices that use SAS tokens without registering

1. Copy the IoT Central application's group primary key
2. Use the dps-keygen tool to generate the device SAS keys. Use the group primary key from the previous step. The device IDs must be lower-case: dps-keygen -mk:<group primary key> -di:<device ID>
3. The OEM flashes each device with a device ID, a generated device SAS key, and the application ID scope value.
4. When you switch on a device, it first connects to DPS to retrieve its IoT Central registration information.

The device initially has a device status Unassociated on the Devices page and isn't assigned to a device template. On the Devices page, Migrate the device to the appropriate device template. Device provisioning is now complete, the device status is now Provisioned, and the device can start sending data.

On the Administration > Device connection page, the Auto approve option controls whether you need to manually approve the device before it can start sending data.

Reference: <https://docs.microsoft.com/en-us/azure/iot-central/core/concepts-get-connected>

**QUESTION 9** You have an Azure IoT Central application.

You need to connect an IoT device to the application.



Which two settings do you require in IoT Central to configure the device? Each correct answer presents part of the solution.

**NOTE:** Each correct selection is worth one point.

- A. Group SAS Primary Key
- B. the IoT hub name
- C. Scope ID
- D. Application Name
- E. Device ID

**Correct Answer:** CE

**Section: [none]**

**Explanation**

**Explanation/Reference:**

Explanation:

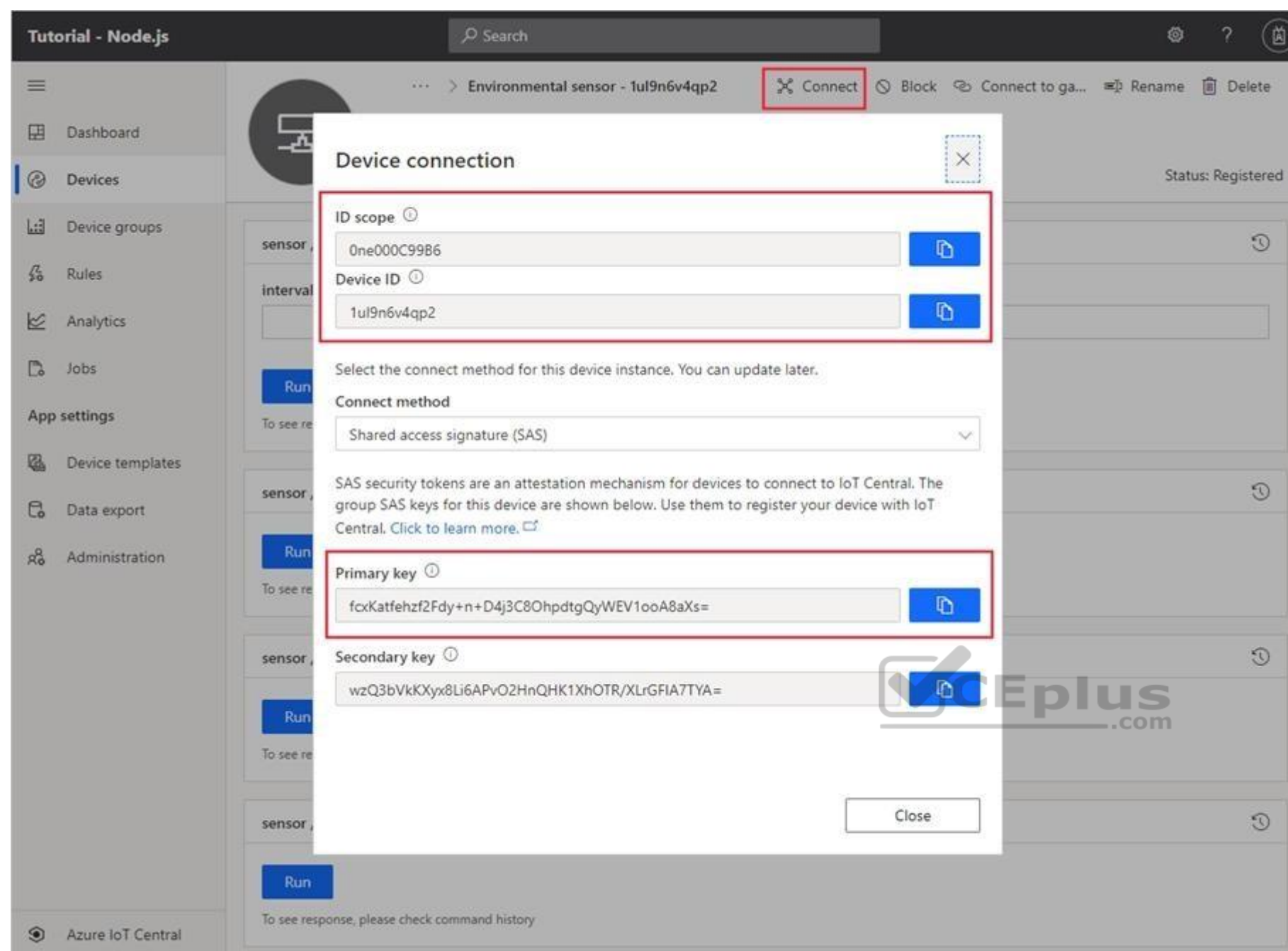
In your Azure IoT Central application, add a real device to the device template 1.

On the Devices page, select the Environmental sensor device template.

2. Select + New.

3. Make sure that Simulated is Off. Then select Create.

Click on the device name, and then select Connect. Make a note of the device connection information on the Device Connection page - ID scope, Device ID, and Primary key. You need these values when you create your device code:



Reference: <https://docs.microsoft.com/bs-cyrl-ba/azure/iot-central/core/tutorial-connect-device-python>

**QUESTION 10** You have an existing Azure IoT hub.

You use IoT Hub jobs to schedule long running tasks on connected devices.

Which three operations do the IoT Hub jobs support directly? Each correct answer presents a complete solution.

**NOTE:** Each correct selection is worth one point.

- A. Trigger Azure functions.
- B. Invoke direct methods.
- C. Update desired properties.
- D. Send cloud-to-device messages.
- E. Disable IoT device registry entries.

F. Update tags.

**Correct Answer:** BCF

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

Consider using jobs when you need to schedule and track progress any of the following activities on a set of devices: ▪

Invoke direct methods

▪ Update desired properties ▪

Update tags

Reference:

<https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-devguide-jobs>

**QUESTION 11** You have an Azure IoT hub.

You need to recommend a solution to scale the IoT hub automatically.

What should you include in the recommendation?

A. Create an SMS alert in IoT Hub for the Total number of messages used metric.

B. Create an Azure function that retrieves the quota metrics of the IoT hub.

C. Configure autoscaling in Azure Monitor.

D. Emit custom metrics from the IoT device code and create an Azure Automation runbook alert.

**Correct Answer:** B

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

Note: IoT Hub is scaled and priced based on an allowed number of messages per day across all devices connected to that IoT Hub. If you exceed the allowed message threshold for your chosen tier and number of units, IoT Hub will begin rejecting new messages. To date, there is no built-in mechanism for automatically scaling an IoT Hub to the next level of capacity if you approach or exceed that threshold.

Reference:

<https://docs.microsoft.com/en-us/samples/azure-samples/iot-hub-dotnet-autoscale/iot-hub-dotnet-autoscale/>

**QUESTION 12** You have an Azure IoT hub that uses a Device Provisioning Service instance.

You create a new individual device enrollment that uses symmetric key attestation.

Which detail from the enrollment is required to auto provision the device by using the Device Provisioning Service?

A. the registration ID of the enrollment

B. the primary key of the enrollment

C. the device identity of the IoT hub

D. the hostname of the IoT hub

**Correct Answer:** C

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:



An enrollment is the record of devices or groups of devices that may register through auto-provisioning. The enrollment record contains information about the device or group of devices, including:

- the attestation mechanism used by the device
- the optional initial desired configuration
- desired IoT hub
- the desired device ID

Note: Azure IoT auto-provisioning can be broken into three phases:

1. Service configuration - a one-time configuration of the Azure IoT Hub and IoT Hub Device Provisioning Service instances, establishing them and creating linkage between them.
2. Device enrollment - the process of making the Device Provisioning Service instance aware of the devices that will attempt to register in the future. Enrollment is accomplished by configuring device identity information in the provisioning service, as either an "individual enrollment" for a single device, or a "group enrollment" for multiple devices.
3. Device registration and configuration

Reference:

<https://docs.microsoft.com/en-us/azure/iot-dps/concepts-service#enrollment>

### QUESTION 13

You have an Azure IoT hub that uses a Device Provisioning Service instance to automate the deployment of Azure IoT Edge devices.

The IoT Edge devices have a Trusted Platform Module (TPM) 2.0 chip.

From the Azure portal, you plan to add an individual enrollment to the Device Provisioning Service that will use the TPM of the IoT Edge devices as the attestation mechanism.

Which detail should you obtain before you can create the enrollment.

- A. the scope ID and the Device Provisioning Service endpoint
- B. the primary key of the Device Provisioning Service shared access policy and the global device endpoint
- C. the X.509 device certificate and the certificate chain
- D. the endorsement key and the registration ID

**Correct Answer:** D

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

The TPM simulator's Registration ID and the Endorsement key, are used when you create an individual enrollment for your device.

Reference:

<https://docs.microsoft.com/en-us/azure/iot-edge/how-to-auto-provision-simulated-device-linux>

### Question Set 1

#### QUESTION 1

You develop a custom Azure IoT Edge module named temperature-module.

You publish temperature-module to a private container registry named mycr.azurecr.io

You need to build a deployment manifest for the IoT Edge device that will run temperature-module.

Which three container images should you define in the manifest? Each correct answer presents part of the solution.

**NOTE:** Each correct selection is worth one point.

- A. mcr.microsoft.com/azureiotedge-simulated-temperature-sensor:1.0
- B. mcr.microsoft.com/azureiotedge-agent:1.0
- C. mcr.microsoft.com/iotedgedev:2.0
- D. mycr.azurecr.io/temperature-module:latest
- E. mcr.microsoft.com/azureiotedge-hub:1.0

**Correct Answer:** BDE



**Section: [none]**

**Explanation**

**Explanation/Reference:**

Explanation:

Each IoT Edge device runs at least two modules: \$edgeAgent and \$edgeHub, which are part of the IoT Edge runtime. IoT Edge device can run multiple additional modules for any number of processes. Use a deployment manifest to tell your device which modules to install and how to configure them to work together.

Reference: <https://docs.microsoft.com/en-us/azure/iot-edge/module-composition>

**QUESTION 2**

DRAG DROP

You need to install the Azure IoT Edge runtime on a new device that runs Windows 10 IoT Enterprise.

Which four actions should you perform in sequence? To answer, move the appropriate actions from the list of actions to the answer area and arrange them in the correct order.

**Select and Place:**



## Actions

From an elevated PowerShell prompt, run the following command.

```
.{Invoke-WebRequest -useb https://aka.ms/
iotedge-win} |
Invoke-Expression; Initialize-IoTEdge
```

From Azure IoT Hub, create an IoT Edge device.

From a Bash prompt, run the following commands.

```
curl https://packages.
microsoft.com/keys/microsoft.asc |
gpg --dearmor > microsoft.gpg
sudo cp ./microsoft.gpg /etc/apt/trusted.gpg.d/
```

From an elevated PowerShell prompt, run the following command.

```
.{Invoke-WebRequest -useb https://aka.ms/
iotedge-win} |
Invoke-Expression; Deploy-IoTEdge
```

Enter the IoT Edge device connection string.

From a Bash prompt, run the following commands.

```
sudo apt-get install moby-engine
```

## Answer Area



Correct Answer:



## Actions

From an elevated PowerShell prompt, run the following command.

```
•{Invoke-WebRequest -useb https://aka.ms/
iotedge-win} |
Invoke-Expression; Initialize-IoTEdge
```

From Azure IoT Hub, create an IoT Edge device.

From a Bash prompt, run the following commands.

```
curl https://packages.
microsoft.com/keys/microsoft.asc |
gpg --dearmor > microsoft.gpg
sudo cp ./microsoft.gpg /etc/apt/trusted.gpg.d/
```

From an elevated PowerShell prompt, run the following command.

```
•{Invoke-WebRequest -useb https://aka.ms/
iotedge-win} |
Invoke-Expression; Deploy-IoTEdge
```

Enter the IoT Edge device connection string.

From a Bash prompt, run the following commands.

```
sudo apt-get install moby-engine
```

## Answer Area

From Azure IoT Hub, create an IoT Edge device.

From an elevated PowerShell prompt, run the following command.

```
•{Invoke-WebRequest -useb https://aka.ms/
iotedge-win} |
Invoke-Expression; Deploy-IoTEdge
```

From an elevated PowerShell prompt, run the following command.

```
•{Invoke-WebRequest -useb https://aka.ms/
iotedge-win} |
Invoke-Expression; Initialize-IoTEdge
```

Enter the IoT Edge device connection string.



Section: [none]  
Explanation

Explanation/Reference:  
Explanation:

Step 1: From Azure IoT Hub, create an IoT Edge Device Step 2: Deploy-IoTEdge



The Deploy-LoTEdge command checks that your Windows machine is on a supported version, turns on the containers feature, and then downloads the moby runtime and the IoT Edge runtime. The command defaults to using Windows containers.

```
{Invoke-WebRequest -useb https://aka.ms/iotedge-win} | Invoke-Expression; ` Deploy-LoTEdge
```

Step 3: Initialize-LoTEdge

The Initialize-LoTEdge command configures the IoT Edge runtime on your machine. The command defaults to manual provisioning with Windows containers. {Invoke-WebRequest -useb https://aka.ms/iotedge

Step 4: Enter the IoT Edge device connection string.

When prompted, provide the device connection string that you retrieved in step 1. The device connection string associates the physical device with a device ID in IoT Hub.

Reference: <https://docs.microsoft.com/en-us/azure/iot-edge/module-composition>

### QUESTION 3

You have an Azure IoT solution that includes an Azure IoT Hub named Hub1 and an Azure IoT Edge device named Edge1. Edge1 connects to Hub1.

You need to deploy a temperature module to Edge1.

What should you do?

- A. From the Azure portal, navigate to Hub1 and select IoT Edge. Select **Edge1**, and then select **Manage Child Devices**. From a Bash prompt, run the following command:  
`az iot edge set-modules -device-id Edge1 -hub-name Hub1 -content C:\deploymentMan1.json`
- B. Create an IoT Edge deployment manifest that specifies the temperature module and the route to \$upstream. From a Bash prompt, run the following command: `az iot hub monitor-events-device-id Edge1 -hub-name Hub1`
- C. From the Azure portal, navigate to Hub1 and select IoT Edge. Select **Edge1**, select **Device Twin**, and then set the deployment manifest as a desired property. From a Bash prompt, run the following command `az iot hub monitor-events-device-id Edge1 -hub-name Hub1`
- D. Create an IoT Edge deployment manifest that specifies the temperature module and the route to \$upstream. From a Bash prompt, run the following command: `az iot edge set-modules -device-id Edge1 -hub-name Hub1 -content C:\deploymentMan1.json`

**Correct Answer: D**

**Section: [none]**

**Explanation**

**Explanation/Reference:**

Explanation:

You deploy modules to your device by applying the deployment manifest that you configured with the module information.

Change directories into the folder where your deployment manifest is saved. If you used one of the VS Code IoT Edge templates, use the deployment.json file in the config folder of your solution directory and not the deployment.template.json file.

Use the following command to apply the configuration to an IoT Edge device:

```
az iot edge set-modules --device-id [device id] --hub-name [hub name] --content [file path]
```

Reference: <https://docs.microsoft.com/en-us/azure/iot-edge/how-to-deploy-modules-cli>

### QUESTION 4

DRAG DROP

Your company is creating a new camera security system that will use Azure IoT Hub.

You plan to use an Azure IoT Edge device that will run Ubuntu Server 18.04.

You need to configure the IoT Edge device.

Which three actions should you perform in sequence? To answer, move the appropriate actions from the list of actions to the answer area and arrange them in the correct order.

Select and Place:

### Actions

Create an individual device enrollment by using the Device Provisioning Service.

Run the following commands.

```
sudo apt-get install moby-engine  
sudo apt-get install moby-cli  
sudo apt-get install iotedge
```

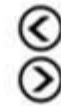
Add the connection string to the `/etc/iotedge/config.yaml` file, and then run the following command.

```
sudo systemctl restart iotedge
```

Install the IoT edge repository for Ubuntu Server 18.04 on the physical device. From IoT Hub, create a new IoT Edge device.

From IoT Hub, create an IoT Edge device registry entry.

### Answer Area



Correct Answer:

## Actions

Create an individual device enrollment by using the Device Provisioning Service.

Run the following commands.

```
sudo apt-get install moby-engine
sudo apt-get install moby-cli
sudo apt-get install iotedge
```

Add the connection string to the /etc/iotedge/config.yaml file, and then run the following command.

```
sudo systemctl restart iotedge
```

Install the IoT edge repository for Ubuntu Server 18.04 on the physical device. From IoT Hub, create a new IoT Edge device.

From IoT Hub, create an IoT Edge device registry entry.

## Answer Area

Run the following commands.

```
sudo apt-get install moby-engine
sudo apt-get install moby-cli
sudo apt-get install iotedge
```

From IoT Hub, create an IoT Edge device registry entry.



Add the connection string to the /etc/iotedge/config.yaml file, and then run the following command.

```
sudo systemctl restart iotedge
```



Section: [none]

Explanation

Explanation/Reference:

Explanation:

Step 1: Run the following commands

Install the container runtime.

Azure IoT Edge relies on an OCI-compatible container runtime. For production scenarios, we recommended that you use the Moby-based engine provided below. The Moby engine is the only container engine officially supported with Azure IoT Edge. Docker CE/EE container images are compatible with the Moby runtime.

Install the Moby engine. sudo  
apt-get install moby-engine

Install the Moby command-line interface (CLI). The CLI is useful for development but optional for production deployments. sudo  
apt-get install moby-cli

Install the security daemon. The package is installed at /etc/iotedge/.  
 sudo apt-get install iotedge

Step 2: From IoT Hub, create an IoT Edge device registry entry.

Note: In your IoT Hub in the Azure portal, IoT Edge devices are created and managed separately from IoT devices that are not edge enabled.

1. Sign in to the Azure portal and navigate to your IoT hub.
2. In the left pane, select IoT Edge from the menu.
3. Select Add an IoT Edge device.
4. Provide a descriptive device ID. Use the default settings to auto-generate authentication keys and connect the new device to your hub.
5. Select Save.

Retrieve the connection string in the Azure portal

1. When you're ready to set up your device, you need the connection string that links your physical device with its identity in the IoT hub.
2. From the IoT Edge page in the portal, click on the device ID from the list of IoT Edge devices.
3. Copy the value of either Primary Connection String or Secondary Connection String.

Step 3: Add the connection string to..

To manually provision a device, you need to provide it with a device connection string that you can create by registering a new device in your IoT hub.

Open the configuration file. sudo  
 nano /etc/iotedge/config.yaml

Find the provisioning configurations of the file and uncomment the Manual provisioning configuration section. Update the value of device\_connection\_string with the connection string from your IoT Edge device. Save and close the file.

After entering the provisioning information in the configuration file, restart the daemon: sudo  
 systemctl restart iotedge

Reference: <https://docs.microsoft.com/en-us/azure/iot-edge/how-to-install-iot-edge-linux>



## QUESTION 5

You have the devices shown in the following table.

Name	Type	Hardware configuration
Device1	Azure Sphere microcontroller unit (MCU)	4 MB of RAM ARM processor
Device2	Raspberry Pi single board computer (SBC)	1 GB of RAM ARM processor
Device3	Desktop computer	8 GB of RAM x64 processor
Device4	Apple iPhone	4 GB of RAM ARM processor

You are implementing a proof of concept (POC) for an Azure IoT solution.

You need to deploy an Azure IoT Edge device as part of the POC.

On which two devices can you deploy IoT Edge? Each correct answer presents a complete solution.

**NOTE:** Each correct selection is worth one point.

- A. Device1
- B. Device2
- C. Device3
- D. Device4

**Correct**

**Answer:**

BC

**Section: [none]**

**Explanation**

**Explanation/Reference:**

Explanation:

Azure IoT Edge runs great on devices as small as a Raspberry Pi3 to server grade hardware.

Tier 1.

The systems listed in the following table are supported by Microsoft, either generally available or in public preview, and are tested with each new release.

Operating System	AMD64	ARM32v7	ARM64
Raspbian Stretch		✓	
Ubuntu Server 16.04	✓		Public preview
Ubuntu Server 18.04	✓		Public preview
Windows 10 IoT Core, build 17763	✓		
Windows 10 IoT Enterprise, build 17763	✓		
Windows Server 2019, build 17763	✓		
Windows Server IoT 2019, build 17763	✓		



Reference:

<https://docs.microsoft.com/en-us/azure/iot-edge/support>

**Testlet 1**

**Case Study**

This is a case study. **Case studies are not timed separately. You can use as much exam time as you would like to complete each case.** However, there may be additional case studies and sections on this exam. You must manage your time to ensure that you are able to complete all questions included on this exam in the time provided.

To answer the questions included in a case study, you will need to reference information that is provided in the case study. Case studies might contain exhibits and other resources that provide more information about the scenario that is described in the case study. Each question is independent of the other question on this case study.

At the end of this case study, a review screen will appear. This screen allows you to review your answers and to make changes before you move to the next sections of the exam. After you begin a new section, you cannot return to this section.

**To start the case study**

To display the first question on this case study, click the **Next** button. Use the buttons in the left pane to explore the content of the case study before you answer the questions. Clicking these buttons displays information such as business requirements, existing environment, and problem statements. If the case study has an **All Information** tab, note that the information displayed is identical to the information displayed on the subsequent tabs. When you are ready to answer a question, click the **Question** button to return to the question.

**Existing Environment. Current State of Development**

Contoso produces a set of Bluetooth sensors that read the temperature and humidity. The sensors connect to IoT gateway devices that relay the data.

All the IoT gateway devices connect to an Azure IoT hub named iothub1.

**Existing Environment. Device Twin**

You plan to implement device twins by using the following JSON sample.





```
{
  "deviceId": "device_n",
  "etag": "AAAAAAAAAAQ=",
  "deviceEtag": "NDcwMTU4Mzk=",
  "status": "enabled",
  "statusUpdateTime": "0001-01-01T00:00:00Z",
  "connectionState": "Disconnected",
  "lastActivityTime": "0001-01-01T00:00:00Z",
  "cloudToDeviceMessageCount": 0,
  "authenticationType": "sas",
  "x509Thumbprint": {
    "primaryThumbprint": null,
    "secondaryThumbprint": null
  },
  "version": 11,
  "properties": {
    "desired": {
      "fanSpeed": 70,
      "$metadata": {
        "$lastUpdated": "2019-10-16T09:43:42.2944169Z",
        "$lastUpdatedVersion": 4,
        "fanSpeed": {
          "$lastUpdated": "2019-10-16T09:43:42.2944169Z",
          "$lastUpdatedVersion": 4
        }
      }
    },
    "$version": 4
  },
  "reported": {
    "fanSpeed": 80,
    "metadata": {
      "$lastUpdated": "2019-10-16T09:43:42.4035171Z",
      "fanSpeed": {
        "$lastUpdated": "2019-10-16T09:43:42.4035171Z"
      }
    }
  },
  "$version": 7
}
},
"capabilities": {
  "lotEdge": false
}
}
```



### Existing Environment. Azure Stream Analytics

Each room will have between three to five sensors that will generate readings that are sent to a single IoT gateway device. The IoT gateway device will forward all the readings to iotHub1 at intervals of between 10 and 60 seconds.

You plan to use a gateway pattern so that each IoT gateway device will have its own IoT Hub device identity.



You draft the following query, which is missing the `GROUP BY` clause.

```
SELECT
    AVG(temperature),
    System.TimeStamp() AS AsaTime
FROM
    Iothub
```

You plan to use a 30-second period to calculate the average temperature reading of the sensors.

You plan to minimize latency between the condition reported by the sensors and the corresponding alert issued by the Stream Analytics job.

#### Existing Environment. Device Messages

The IoT gateway devices will send messages that contain the following JSON data whenever the temperature exceeds a specified threshold.

```
{
  "event": {
    "payload": "Temperature = 26.23 Humidity = 78.70597746416186 Button = 0",
    "properties": {
      "application": {
        "level": "critical"
      }
    }
  }
}
```

The `level` property will be used to route the messages to an Azure Service Bus queue endpoint named `criticalep`.

#### Existing Environment. Issues

You discover connectivity issues between the IoT gateway devices and `iothub1`, which cause IoT devices to lose connectivity and messages.

#### Requirements. Planning Changes Contoso

plans to make the following changes:

- Use Stream Analytics to process and view data.
- Use Azure Time Series Insights to visualize data.
- Implement a system to sync device statuses and required settings.
- Add extra information to messages by using message enrichment.
- Create a notification system to send an alert if a condition exceeds a specified threshold.
- Implement a system to identify what causes the intermittent connection issues and lost messages.

#### Requirements. Technical Requirements

Contoso must meet the following requirements:

- Use the built-in functions of IoT Hub whenever possible.
- Minimize hardware and software costs whenever possible.
- Minimize administrative effort to provision devices at scale.
- Implement a system to trace message flow to and from `iothub1`.
- Minimize the amount of custom coding required to implement the planned changes.
- Prevent read operations from being negatively affected when you implement additional services.

**QUESTION 1** You plan to deploy Azure Time Series Insights.

What should you create on `iothub1` before you deploy Time Series Insights?

- A. a new message route
- B. a new consumer group
- C. a new shared access policy
- D. an IP filter rule

**Correct Answer:** B

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

Create a dedicated consumer group in the IoT hub for the Time Series Insights environment to consume from. Each Time Series Insights event source must have its own dedicated consumer group that isn't shared with any other consumer. If multiple readers consume events from the same consumer group, all readers are likely to exhibit failures.

Reference: [https://docs.microsoft.com/en-us/azure/time-series-insights/time-series-insights-how-to-add-an-event-source-  
iothub](https://docs.microsoft.com/en-us/azure/time-series-insights/time-series-insights-how-to-add-an-event-source-iothub)

**QUESTION 2** How should you complete the GROUP BY clause to meet the Streaming Analytics requirements?

- A. GROUP BY HoppingWindow(Second, 60, 30)
- B. GROUP BY TumblingWindow(Second, 30)
- C. GROUP BY SlidingWindow(Second, 30)
- D. GROUP BY SessionWindow(Second, 30, 60)

**Correct Answer:** B

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

Scenario: You plan to use a 30-second period to calculate the average temperature reading of the sensors.

Tumbling window functions are used to segment a data stream into distinct time segments and perform a function against them, such as the example below. The key differentiators of a Tumbling window are that they repeat, do not overlap, and an event cannot belong to more than one tumbling window.

Incorrect Answers:

A: Hopping window functions hop forward in time by a fixed period. It may be easy to think of them as Tumbling windows that can overlap, so events can belong to more than one Hopping window result set.

Reference: [https://docs.microsoft.com/en-us/azure/stream-analytics/stream-analytics-window-  
functions](https://docs.microsoft.com/en-us/azure/stream-analytics/stream-analytics-window-functions)

**QUESTION 3** HOTSPOT

You need to use message enrichment to add additional device information to messages sent from the IoT gateway devices when the reported temperature exceeds a critical threshold.

How should you configure the enrich message values? To answer, select the appropriate options in the answer area.

**NOTE:** Each correct selection is worth one point.

**Hot Area:**



### Answer Area

\$iothubname	desired.pressure
\$twin	fanSpeed.reported
\$twin.properties	reported.fanSpeed
\$twin.results	temperature
\$twin.tags	temperature.reported

Correct Answer:

### Answer Area

\$iothubname	desired.pressure
\$twin	fanSpeed.reported
\$twin.properties	reported.fanSpeed
\$twin.results	temperature
\$twin.tags	temperature.reported

Section: [none]  
Explanation

Explanation/Reference:

Reference:

<https://docs.microsoft.com/bs-cyrl-ba/azure/iot-hub/iot-hub-message-enrichments-overview> Testlet 2

### Case Study

This is a case study. **Case studies are not timed separately. You can use as much exam time as you would like to complete each case.** However, there may be additional case studies and sections on this exam. You must manage your time to ensure that you are able to complete all questions included on this exam in the time provided.

To answer the questions included in a case study, you will need to reference information that is provided in the case study. Case studies might contain exhibits and other resources that provide more information about the scenario that is described in the case study. Each question is independent of the other question on this case study.

At the end of this case study, a review screen will appear. This screen allows you to review your answers and to make changes before you move to the next sections of the exam. After you begin a new section, you cannot return to this section.

### To start the case study

To display the first question on this case study, click the **Next** button. Use the buttons in the left pane to explore the content of the case study before you answer the questions. Clicking these buttons displays information such as business requirements, existing environment, and problem statements. If the case study has an **All Information** tab, note that the information displayed is identical to the information displayed on the subsequent tabs. When you are ready to answer a question, click the **Question** button to return to the question.

### Requirements. Planned Changes

ADatum is developing an Azure IoT solution to monitor environmental conditions. The IoT solution consists of hardware devices and cloud services. All the devices will communicate directly to Azure IoT Hub.

The hardware devices will be deployed to the branch offices and will collect data about various environmental conditions such as temperature, humidity, air quality, and noise level. The devices will be wired by using Power over Ethernet (PoE) connections.

ADatum is developing the solution in the following three phases: proof of value (POV), pilot, and production.

#### **Requirements. POV Requirements**

The POV phase will demonstrate that a technical solution is viable. During this phase, 100 devices will be deployed to the main office and Azure Stream Analytics will be connected to an IoT hub to generate real-time alerts. Stream Analytics will perform the following processing:

- Calculate the median rate of the telemetry across the entire devices that exceed the median rate by a factor of 4.
- Compare the current telemetry to the specified thresholds and issue alerts when telemetry values are out of range. ▪

Ensure that all message content during this phase is human readable to simplify debugging. **Requirements. Pilot**

#### **Requirements**

During the pilot phase, devices will be deployed to 10 offices. Each office will have up to 1,000 devices.

During this phase, you will add Azure Time Series Insights in parallel to Stream Analytics to support real-time graphs and queries in a dashboard web app.

The pilot deployment must minimize operating costs.

#### **Requirements. Production Requirements**

The production phase will include all the offices.

The production deployment will have one IoT hub in each Azure region. Devices must connect to the IoT hub in their region.

The production phase must meet the following requirements:

- Ensure that the IoT solution can support performance and scale targets.
- Ensure that the IoT solution support up to 1,000 devices per office. ▪

Minimize operating costs of the IoT solution.

#### **Requirements. Technical Requirements**

Datum identifies the following requirements for the planned IoT solution:

- The solution must generate real-time alerts when a fire condition is detected in an office. All the devices in that office must trigger an audible alarm siren within 10 seconds of the alert. ▪ A dashboard UI must display alerts and the system status in real time and must allow device operators to make adjustments to the system.

- Each device will send hourly updates to IoT Hub. Condition alerts will be sent immediately.
- Multiple types of devices will collect telemetry that has different schemas.
- IoT Hub must perform message routing based on the message body.
- Direct methods must be used for cloud-to-device communication.
- Reports must be provided monthly, quarterly, and annually.
- Stored data queries must be as efficient as possible.
- The device message size will be under 4 KB. ▪

Development effort must be minimized.

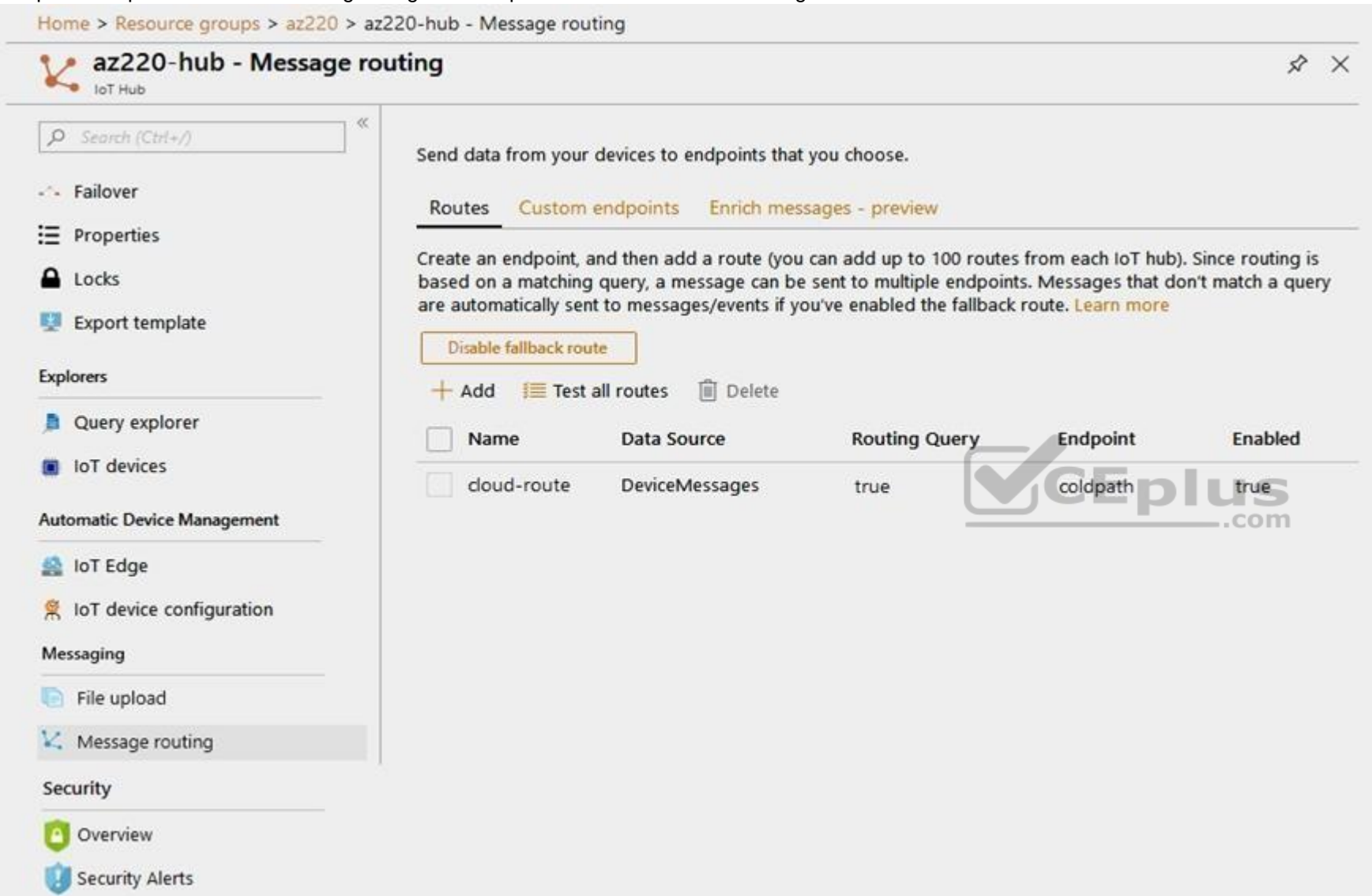
#### **Requirements. Throttle and Quotas**

The relevant throttles and quotas for various IoT Hub tiers are shown in the following table.

Tier	Direct method	Device-to-cloud message	Price per month
B1	40/sec/unit	400,000/day/unit	\$10/unit
S1	40/sec/unit	400,000/day/unit	\$25/unit
S2	120/sec/unit	6,000,000/day/unit	\$250/unit

### Requirements. IoT Hub Routing

You plan to implement IoT Hub routing during the POV phase as shown in the following exhibit.



Home > Resource groups > az220 > az220-hub - Message routing

**az220-hub - Message routing**

Send data from your devices to endpoints that you choose.

Routes Custom endpoints Enrich messages - preview

Create an endpoint, and then add a route (you can add up to 100 routes from each IoT hub). Since routing is based on a matching query, a message can be sent to multiple endpoints. Messages that don't match a query are automatically sent to messages/events if you've enabled the fallback route. [Learn more](#)

[Disable fallback route](#)

+ Add Test all routes Delete

Name	Data Source	Routing Query	Endpoint	Enabled
cloud-route	DeviceMessages	true	coldpath	true

**QUESTION 1** You need to configure Stream Analytics to meet the POV requirements.

What are two ways to achieve the goal? Each correct answer presents a complete solution.

**NOTE:** Each correct selection is worth one point.

- A. From IoT Hub, create a custom event hub endpoint, and then configure the endpoint as an input to Stream Analytics.
- B. Create a Stream Analytics module, and then deploy the module to all IoT Edge devices in the fleet.
- C. Create an input in Stream Analytics that uses the built-in events endpoint of IoT Hub as the source.
- D. Route telemetry to an Azure Blob storage custom endpoint, and then configure the Blob storage as a reference input for Stream Analytics.

**Correct Answer:** AC

**Section: [none]**

**Explanation**

**Explanation/Reference:**

**QUESTION 2**

DRAG DROP

You need to add Time Series Insights to the solution to meet the pilot requirements.

Which three actions should you perform in sequence? To answer, move the appropriate actions from the list of actions to the answer area and arrange them in the correct order.

**Select and Place:**

**Correct Answer:**

**Section: [none]**

**Explanation**

**Explanation/Reference:**

Explanation:

Step 1: Provision Time Series Insights  
Select Provision new IoT Hub to create a new IoT hub.

Step 2: Route telemetry from IoT Hub to a custom event.

Step 3: Add a data access policy to Time Series Insights for the dashboard web app

Scenario: Requirements. Pilot Requirements

During the pilot phase, devices will be deployed to 10 offices. Each office will have up to 1,000 devices.

During this phase, you will add Azure Time Series Insights in parallel to Stream Analytics to support real-time graphs and queries in a dashboard web app.

The pilot deployment must minimize operating costs.

Incorrect Answers:

No need to use an endpoint.

Reference: <https://docs.microsoft.com/en-us/azure/time-series-insights/time-series-insights-update-create-environment>

**QUESTION 3** You need to store the real-time alerts generated by Stream Analytics to meet the technical requirements.

Which type of Stream Analytics output should you configure?

- A. Azure Blob storage
- B. Microsoft Power BI
- C. Azure Cosmos DB
- D. Azure SQL Database

**Correct Answer: A**

**Section: [none]**

**Explanation**

**Explanation/Reference:**

Explanation:

When you create a Time Series Insights Preview pay-as-you-go (PAYG) SKU environment, you create two Azure resources:

- An Azure Storage general-purpose V1 blob account for cold data storage.
- An Azure Time Series Insights Preview environment that can be configured for warm data storage.



Reference: <https://docs.microsoft.com/en-us/azure/time-series-insights/time-series-insights-update-storage-ingress>

**QUESTION 4** You need to recommend the format of telemetry messages to meet the POV requirements.

What should you recommend?

- A. XML
- B. Avro
- C. JSON

**Correct Answer: C**

**Section: [none]**

**Explanation**

**Explanation/Reference:**

Explanation:

Scenario: POV Requirements

- Ensure that all message content during this phase is human readable to simplify debugging.

Avro uses a binary format, so it is not human readable.

The more lightweight JSON (Javascript object notation) has become a popular alternative to XML for various reasons. A couple obvious ones are: ▪

Less verbose- XML uses more words than necessary

- JSON is faster- Parsing XML software is slow and cumbersome.

Reference: <https://blog.cloud-elements.com/json-better-xml/>

**QUESTION 5** During the POV phase, telemetry from IoT Hub stops flowing to the hot path. The cold path continues to work.

What should you do to restore the hot path?

- A. Disable the fallback route.
- B. Run the Test all routes action.
- C. Create an explicit route for the hot path.
- D. Modify cold-route to send only some telemetry data to the cold path.

**Correct Answer: C**

**Section: [none]**

**Explanation**

**Explanation/Reference:**



### Question Set 3

**QUESTION 1** You have 100 devices that connect to an Azure IoT hub.

You plan to use Azure functions to process all the telemetry messages from the devices before storing the messages.

You need to configure the functions binding for the IoT hub.

Which two configuration details should you use to configure the binding? Each correct answer presents part of the solution.

**NOTE:** Each correct selection is worth one point.

- A. the name of the resource group that contains the IoT hub
- B. the IoT hub's connection string shared access key that has Service connect permissions
- C. the connection string of the Azure Event Hub-compatible endpoint from the IoT Hub built-in endpoints
- D. the Azure Event-Hub compatible name

**Correct Answer:** CD

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

EventHubName: Functions 2.x and higher. The name of the event hub. When the event hub name is also present in the connection string, that value overrides this property at runtime.

Connection: The name of an app setting that contains the connection string to the event hub's namespace. Copy this connection string by clicking the Connection Information button for the namespace, not the event hub itself. This connection string must have send permissions to send the message to the event stream.

Reference: <https://docs.microsoft.com/en-us/azure/azure-functions/functions-bindings-event-iot-output>



### QUESTION 2

DRAG DROP

You have an instance of Azure Time Series Insights and an Azure IoT hub that receives streaming telemetry from IoT devices.

You need to configure Time Series Insights to receive telemetry from the devices.

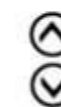
Which three actions should you perform in sequence? To answer, move the appropriate actions from the list of actions to the answer area and arrange them in the correct order.

**Select and Place:**

### Actions

- Configure the Time Series Insights event source to connect to an existing IOT hub.
- Create an Azure event hub.
- Add a new Time Series Insights event source.
- Increase the events retention to seven days for the built-in endpoints of the IoT hub.
- Create a dedicated consumer group in the built-in events endpoints of the IoT hub.

### Answer Area

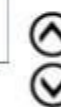


Correct Answer:

### Actions

- Configure the Time Series Insights event source to connect to an existing IOT hub.
- Create an Azure event hub.
- Add a new Time Series Insights event source.
- Increase the events retention to seven days for the built-in endpoints of the IoT hub.
- Create a dedicated consumer group in the built-in events endpoints of the IoT hub.

### Answer Area



Section: [none]  
Explanation

Explanation/Reference:  
Explanation:

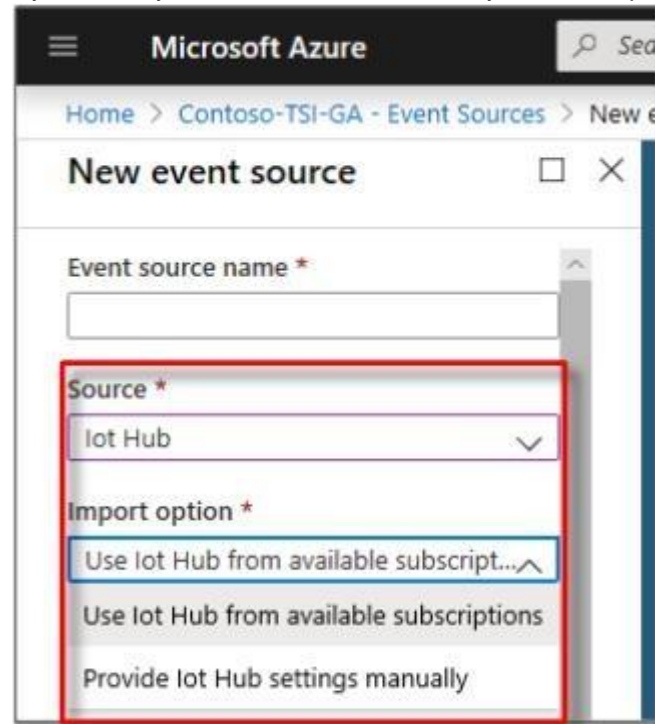
- Step 1: Create a dedicated consumer group..  
Add a consumer group to your IoT hub.  
Applications use consumer groups to pull data from Azure IoT Hub. To reliably read data from your IoT hub, provide a dedicated consumer group that's used only by this Time Series Insights environment.
- Step 2: Add a new Time Series Insights event source.  
Add a new event source
1. Sign in to the Azure portal.
  2. In the left menu, select All resources. Select your Time Series Insights environment.

3. Under Settings, select Event Sources, and then select Add.
4. In the New event source pane, for Event source name, enter a name that's unique to this Time Series Insights environment. For example, enter event-stream.  
hub

Step 4: For Source, select IoT Hub.

Step 5: Select a value for Import option:

If you already have an IoT hub in one of your subscriptions, select Use IoT Hub from available subscriptions. This option is the easiest approach.



Reference: <https://docs.microsoft.com/en-us/azure/time-series-insights/time-series-insights-how-to-add-an-event-source-iot-hub>

**QUESTION 3** You have 1,000 devices that connect to a standard tier Azure IoT hub.

All the devices are commissioned and send telemetry events to the built-in IoT Hub endpoint.

You configure message enrichment on the events endpoint and set the enrichment value to `$twin.tags.ipV4`.

When you inspect messages on the events endpoint, you discover that all the messages are stamped with a string of `"$twin.tags.ipV4"`.

What are two possible causes of the issue? Each correct answer presents a complete solution.

**NOTE:** Each correct selection is worth one point.

- A. The `ipV4` tag is a restricted twin property that is unavailable for message enrichment.
- B. A standard tier IoT hub does not support device twin properties in message enrichments.
- C. The device sending the message has no device twin.
- D. Message enrichment cannot be added to messages going to a built-in endpoint.
- E. The device twin path used for the value of the enrichment does not exist.
- F. The device twin property value used for message enrichment is set to `"$twin.tags.ipV4"`.

**Correct Answer:** CE

**Section: [none]**

**Explanation**

**Explanation/Reference:**

Explanation:

In some cases, if you are applying an enrichment with a value set to a tag or property in the device twin, the value will be stamped as a string value. For example, if an enrichment value is set to \$twin.tags.field, the messages will be stamped with the string "\$twin.tags.field" rather than the value of that field from the twin. This happens in the following cases:

- (C) Your IoT Hub is in the standard tier, but the device sending the message has no device twin.
- (E) Your IoT Hub is in the standard tier, but the device twin path used for the value of the enrichment does not exist. For example, if the enrichment value is set to \$twin.tags.location, and the device twin does not have a location property under tags, the message is stamped with the string "\$twin.tags.location".
- Your IoT Hub is in the basic tier. Basic tier IoT hubs do not support device twins.

Reference: <https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-message-enrichments-overview>



## Testlet 1

### Case Study

This is a case study. **Case studies are not timed separately. You can use as much exam time as you would like to complete each case.** However, there may be additional case studies and sections on this exam. You must manage your time to ensure that you are able to complete all questions included on this exam in the time provided.

To answer the questions included in a case study, you will need to reference information that is provided in the case study. Case studies might contain exhibits and other resources that provide more information about the scenario that is described in the case study. Each question is independent of the other question on this case study.

At the end of this case study, a review screen will appear. This screen allows you to review your answers and to make changes before you move to the next sections of the exam. After you begin a new section, you cannot return to this section.

#### To start the case study

To display the first question on this case study, click the **Next** button. Use the buttons in the left pane to explore the content of the case study before you answer the questions. Clicking these buttons displays information such as business requirements, existing environment, and problem statements. If the case study has an **All Information** tab, note that the information displayed is identical to the information displayed on the subsequent tabs. When you are ready to answer a question, click the **Question** button to return to the question.

#### Existing Environment. Current State of Development

Contoso produces a set of Bluetooth sensors that read the temperature and humidity. The sensors connect to IoT gateway devices that relay the data.

All the IoT gateway devices connect to an Azure IoT hub named iothub1.

#### Existing Environment. Device Twin

You plan to implement device twins by using the following JSON sample.



```
{
  "deviceId": "device_n",
  "etag": "AAAAAAAAAAQ=",
  "deviceEtag": "NDcwMTU4Mzk=",
  "status": "enabled",
  "statusUpdateTime": "0001-01-01T00:00:00Z",
  "connectionState": "Disconnected",
  "lastActivityTime": "0001-01-01T00:00:00Z",
  "cloudToDeviceMessageCount": 0,
  "authenticationType": "sas",
  "x509Thumbprint": {
    "primaryThumbprint": null,
    "secondaryThumbprint": null
  },
  "version": 11,
  "properties": {
    "desired": {
      "fanSpeed": 70,
      "$metadata": {
        "$lastUpdated": "2019-10-16T09:43:42.2944169Z",
        "$lastUpdatedVersion": 4,
        "fanSpeed": {
          "$lastUpdated": "2019-10-16T09:43:42.2944169Z",
          "$lastUpdatedVersion": 4
        }
      }
    },
    "$version": 4
  },
  "reported": {
    "fanSpeed": 80,
    "metadata": {
      "$lastUpdated": "2019-10-16T09:43:42.4035171Z",
      "fanSpeed": {
        "$lastUpdated": "2019-10-16T09:43:42.4035171Z"
      }
    }
  },
  "$version": 7
}
},
"capabilities": {
  "lotEdge": false
}
}
```



#### Existing Environment. Azure Stream Analytics

Each room will have between three to five sensors that will generate readings that are sent to a single IoT gateway device. The IoT gateway device will forward all the readings to iotHub1 at intervals of between 10 and 60 seconds.

You plan to use a gateway pattern so that each IoT gateway device will have its own IoT Hub device identity.



You draft the following query, which is missing the `GROUP BY` clause.

```
SELECT
    AVG(temperature),
    System.TimeStamp() AS AsaTime
FROM
    Iothub
```

You plan to use a 30-second period to calculate the average temperature reading of the sensors.

You plan to minimize latency between the condition reported by the sensors and the corresponding alert issued by the Stream Analytics job.

#### Existing Environment. Device Messages

The IoT gateway devices will send messages that contain the following JSON data whenever the temperature exceeds a specified threshold.

```
{
  "event": {
    "payload": "Temperature = 26.23 Humidity = 78.70597746416186 Button = 0",
    "properties": {
      "application": {
        "level": "critical"
      }
    }
  }
}
```

The `level` property will be used to route the messages to an Azure Service Bus queue endpoint named `criticalep`.

#### Existing Environment. Issues

You discover connectivity issues between the IoT gateway devices and `iothub1`, which cause IoT devices to lose connectivity and messages.

#### Requirements. Planning Changes Contoso

plans to make the following changes:

- Use Stream Analytics to process and view data.
- Use Azure Time Series Insights to visualize data.
- Implement a system to sync device statuses and required settings.
- Add extra information to messages by using message enrichment.
- Create a notification system to send an alert if a condition exceeds a specified threshold.
- Implement a system to identify what causes the intermittent connection issues and lost messages.

#### Requirements. Technical Requirements

Contoso must meet the following requirements:

- Use the built-in functions of IoT Hub whenever possible.
- Minimize hardware and software costs whenever possible.
- Minimize administrative effort to provision devices at scale.
- Implement a system to trace message flow to and from `iothub1`.
- Minimize the amount of custom coding required to implement the planned changes.
- Prevent read operations from being negatively affected when you implement additional services.

**QUESTION 1** You need to enable telemetry message tracing through the entire IoT solution.

What should you do?



- A. Monitor device lifecycle events.
- B. Upload IoT device logs by using the File upload feature.
- C. Enable the DeviceTelemetry diagnostic log and stream the log data to an Azure event hub.
- D. Implement distributed tracing.

**Correct Answer:** D

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

IoT Hub is one of the first Azure services to support distributed tracing. As more Azure services support distributed tracing, you'll be able trace IoT messages throughout the Azure services involved in your solution.

Note:

Enabling distributed tracing for IoT Hub gives you the ability to:

- Precisely monitor the flow of each message through IoT Hub using trace context. This trace context includes correlation IDs that allow you to correlate events from one component with events from another component. It can be applied for a subset or all IoT device messages using device twin.
- Automatically log the trace context to Azure Monitor diagnostic logs.
- Measure and understand message flow and latency from devices to IoT Hub and routing endpoints.

Start considering how you want to implement distributed tracing for the non-Azure services in your IoT solution.

Reference:

<https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-distributed-tracing>

## Question Set 2

**QUESTION 1** You have 10 IoT devices that connect to an Azure IoT hub named Hub1.

From Azure Cloud Shell, you run `az iot hub monitor-events --hub-name Hub1` and receive the following error message: "az iot hub: 'monitor-events' is not in the 'az iot hub' command group. See 'az iot hub --help'." You need to ensure that you can run the command successfully.

What should you run first?

- A. `az iot hub monitor-feedback --hub-name Hub1`
- B. `az iot hub generate-sas-token --hub-name Hub1`
- C. `az iot hub configuration list --hub-name Hub1`
- D. `az extension add --name azure-cli-iot-ext`

**Correct Answer:** D

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

Execute `az extension add --name azure-cli-iot-ext` once and try again.

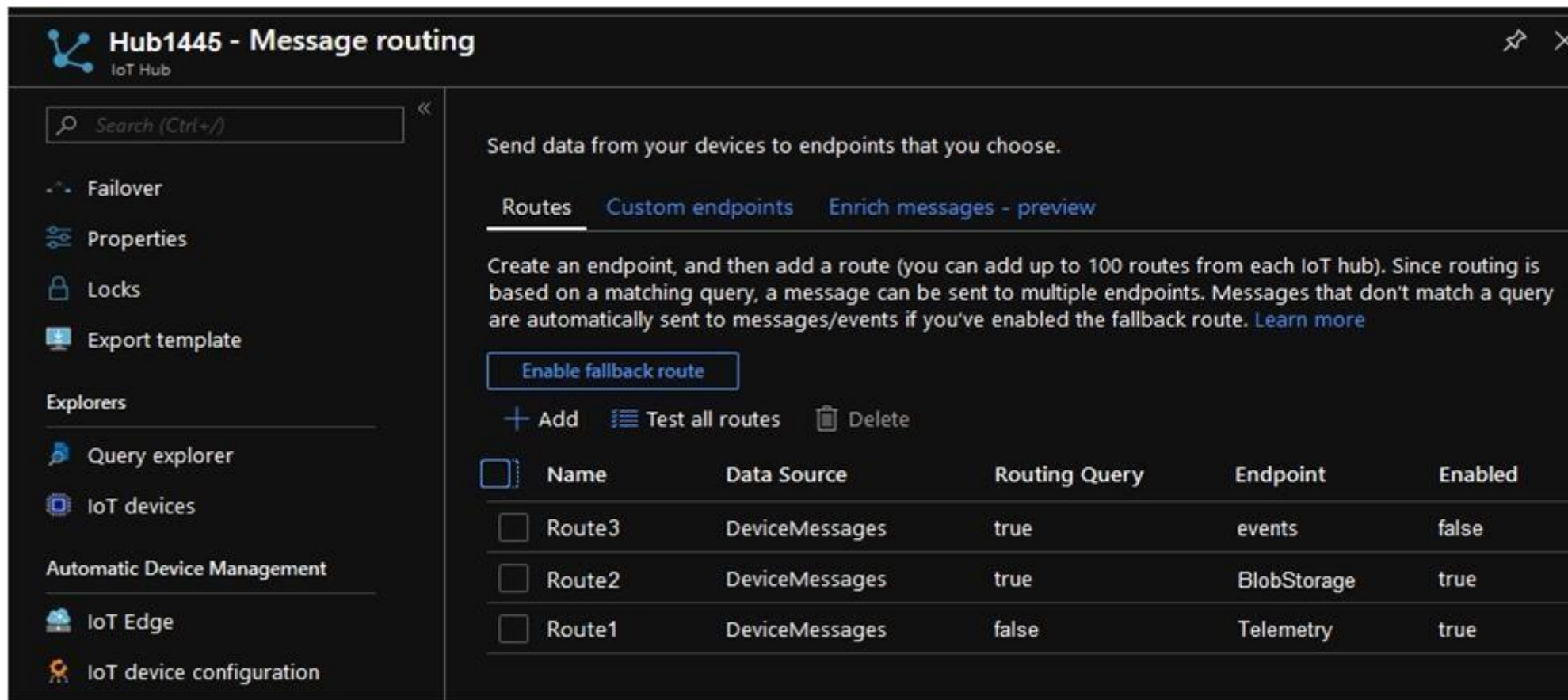
In order to read the telemetry from your hub by CLI, you have to enable IoT Extension with the following commands: Add:  
`az extension add --name azure-cli-iot-ext`

Reference: <https://github.com/MicrosoftDocs/azure-docs/issues/20843>



## QUESTION 2

You have an Azure Stream Analytics job that connects to an Azure IoT hub named Hub1445 as a streaming data source. Hub1445 is configured as shown in the exhibit. (Click the **Exhibit** tab.)



Hub1445 - Message routing

Send data from your devices to endpoints that you choose.

Routes Custom endpoints Enrich messages - preview

Create an endpoint, and then add a route (you can add up to 100 routes from each IoT hub). Since routing is based on a matching query, a message can be sent to multiple endpoints. Messages that don't match a query are automatically sent to messages/events if you've enabled the fallback route. [Learn more](#)

[Enable fallback route](#)

[+ Add](#) [Test all routes](#) [Delete](#)

<input type="checkbox"/>	Name	Data Source	Routing Query	Endpoint	Enabled
<input type="checkbox"/>	Route3	DeviceMessages	true	events	false
<input type="checkbox"/>	Route2	DeviceMessages	true	BlobStorage	true
<input type="checkbox"/>	Route1	DeviceMessages	false	Telemetry	true

The Stream Analytics job fails to receive any messages from the IoT hub.

What should you do to resolve the issue?

- A. Change the Route1 route query to **true**.
- B. Enable the Route3 route.
- C. Disable the Route2 route.
- D. Enable the fallback route.

**Correct Answer:** A

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

The device telemetry is usually passed as JSON from the device through the IoT Hub - this is handled nicely by Azure Streaming Analytics queries.

The IoT Hub message routing should be configured as follows:

Data source: Device Telemetry Messages

Routing query: true (as the routing query is an expression that evaluates to true or false for each received message, the simplest way to send all messages to the endpoint is to just supply true as the query).

Reference: <https://darenmay.com/blog/azure-iot-streaming-analytics-data-lake-analytics-and-json/>

**QUESTION 3** You are troubleshooting an Azure IoT hub.

You discover that some telemetry messages are dropped before they reach downstream processing.

You suspect that IoT Hub throttling is the root cause.

Which log in the Diagnostics settings of the IoT hub should you use to capture the throttling error events?

- A. Routes
- B. DeviceTelemetry
- C. Connections
- D. C2DCommands

**Correct Answer:** B

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

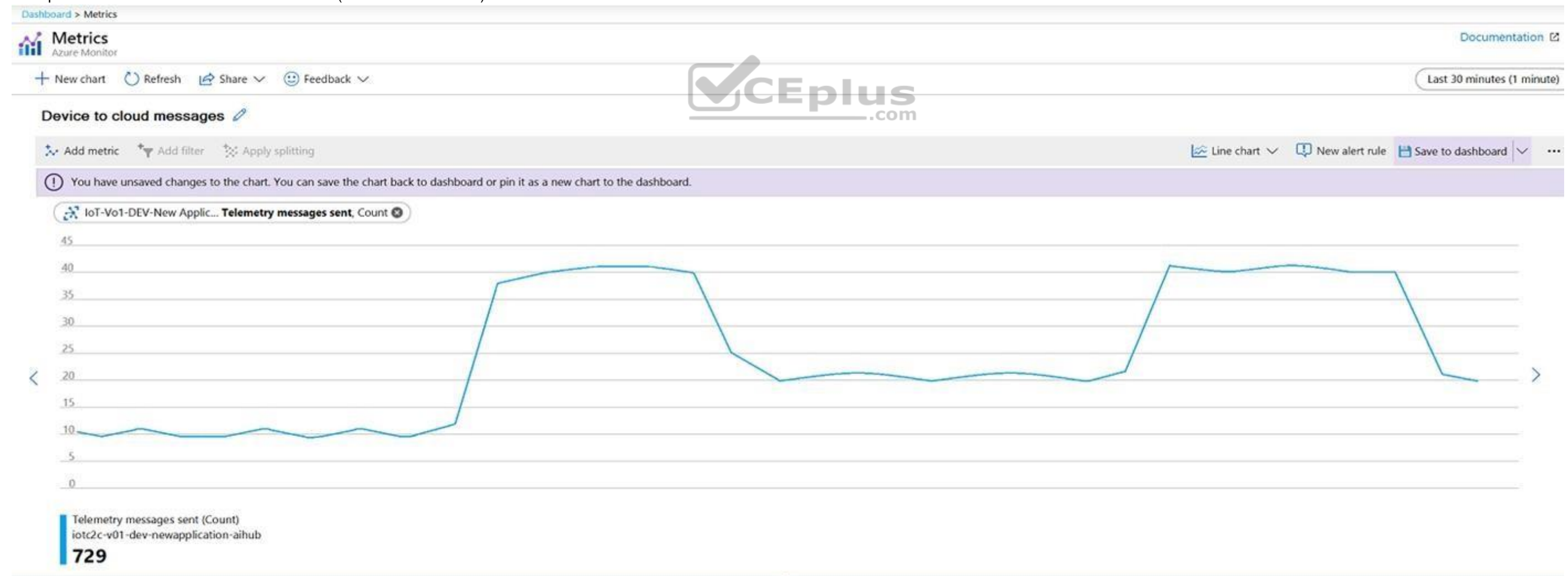
The device telemetry category tracks errors that occur at the IoT hub and are related to the telemetry pipeline. This category includes errors that occur when sending telemetry events (such as throttling) and receiving telemetry events (such as unauthorized reader). This category cannot catch errors caused by code running on the device itself.

Note: The metric d2c.telemetry.ingress.sendThrottle is the number of throttling errors due to device throughput throttles.

Reference: <https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-monitor-resource-health>

**QUESTION 4** You have 20 devices that connect to an Azure IoT hub.

You open **Azure Monitor** as shown in the exhibit. (Click the **Exhibit** tab.)



You discover that telemetry is not being received from five IoT devices.

You need to identify the names of the devices that are not generating telemetry and visualize the data.

What should you do first?

- A. Add the Number of throttling errors metric and archive the logs to an Azure storage account.
- B. Configure diagnostics for Routes and stream the logs to Azure Event Hubs.
- C. Add the Telemetry messages sent metric and archive the logs to an Azure Storage account.
- D. Configure diagnostics for Connections and send the logs to Azure Log Analytics.

**Correct Answer:** D

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

To log device connection events and errors, turn on diagnostics for IoT Hub. We recommend turning on these logs as early as possible, because if diagnostic logs aren't enabled, when device disconnects occur, you won't have any information to troubleshoot the problem with.

1. Sign in to the Azure portal.
2. Browse to your IoT hub.
3. Select Diagnostics settings.
4. Select Turn on diagnostics.
5. Enable Connections logs to be collected.
6. For easier analysis, turn on Send to Log Analytics

### Diagnostics settings

Save
Discard
Delete

**Name**

log-connection-errors-events-to-log-analytics ✓

☐ Archive to a storage account

☐ Stream to an event hub

☒ Send to Log Analytics

Log Analytics  
iot-log-everything-workspace >

LOG

☒ Connections

### Diagnostics settings

Save
Discard
Delete

**Name**

log-connection-errors-events-to-log-analytics ✓

☐ Archive to a storage account

☐ Stream to an event hub

☒ Send to Log Analytics

Log Analytics  
iot-log-everything-workspace >

LOG

☒ Connections

Reference: <https://docs.microsoft.com/bs-cyrl-ba/azure/iot-hub/iot-hub-troubleshoot-connectivity>

#### QUESTION 5

You have an Azure IoT solution that includes a standard tier Azure IoT hub and an IoT device.

The device sends one 100-KB device-to-cloud message every hour.

You need to calculate the total daily message consumption of the device.

What is the total daily message consumption of the device?

- A. 24
- B. 600
- C. 2,400
- D. 4,800

**Correct Answer:** B

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

100 KB \* 24 is around 2,400 bytes.

The 100 KB message is divided into 4 KB blocks, and it is billed for 25 messages. 25 times 24 is 600

Note: The maximum message size for messages sent from a device to the cloud is 256 KB. These messages are metered in 4 KB blocks for the paid tiers so for instance if the device sends a 16 KB message via the paid tiers it will be billed as 4 messages.

Reference:

<https://azure.microsoft.com/en-us/pricing/details/iot-hub/>

**QUESTION 6** You have 1,000 devices that connect to an Azure IoT hub.

You are performing a scheduled check of deployed IoT devices.

You plan to run the following command from the Azure CLI prompt.

```
az iot hub query --hub-name hub1 --query-command "SELECT * FROM devices WHERE connectionState = 'Disconnected'"
```

What does the command return?

- A. the Device Disconnected events
- B. the device twins
- C. the Connections logs
- D. the device credentials

**Correct Answer:** A

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

The IoT Hub publishes the Microsoft.Devices.DeviceDisconnected event type, which is published when a device is disconnected from an IoT hub.

Reference: <https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-event-grid#event-types>

**QUESTION 7** You have an Azure IoT solution that includes several Azure IoT hubs.

A new alerting feature was recently added to the IoT devices. The feature uses a new device twin reported property named `alertCondition`.

You need to send alerts to an Azure Service Bus queue named MessageAlerts. The alerts must include `alertCondition` and the name of the IoT hub.

Which two actions should you perform? Each correct answer presents part of the solution.

**NOTE:** Each correct selection is worth one point.

- A. Configure File upload for each IoT hub. Configure the device to send a file to an Azure Storage container that contains the device name and status message.
- B. Add the following message enrichments:  
Name = iotHubName  
Value = \$twin.tag.location  
Endpoint = MessageAlert
- C. Create an IoT Hub routing rule that has a data source of Device Twin Change Events and select the endpoint for MessageAlerts.
- D. Add the following message enrichments:  
Name = iotHubName  
Value = \$iothubname  
Endpoint = MessageAlert
- E. Create an IoT Hub routing rule that has a data source of Device Telemetry Messages and select the endpoint for MessageAlerts.

**Correct Answer:** BD

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

B: Message enrichments is the ability of the IoT Hub to stamp messages with additional information before the messages are sent to the designated endpoint. One reason to use message enrichments is to include data that can be used to simplify downstream processing. For example, enriching device telemetry messages with a device twin tag can reduce load on customers to make device twin API calls for this information.

D: Applying enrichments

The messages can come from any data source supported by IoT Hub message routing, including the following examples:

- -->device twin change notifications -- changes in the device twin
- device telemetry, such as temperature or pressure
- device life-cycle events, such as when the device is created or deleted



Reference: <https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-message-enrichments-overview>

## QUESTION 8

DRAG DROP

You have 100 devices that connect to an Azure IoT hub.

You need to be notified about failed local logins to a subnet of the devices.

Which three actions should you perform in sequence? To answer, move the appropriate actions from the list of actions to the answer area and arrange them in the correct order.

**Select and Place:**

**Correct Answer:**

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

Step 1: Enable Azure Security Center for IoT

Security alerts, such as failed local IoT hub logins, are stored in AzureSecurityOfThings.SecurityAlert table in the Log Analytics workspace configured for the Azure Security Center for IoT solution.

Step 2: Select a device security group

Update a device security group..

Step 3: Create a custom alert rule



..by creating a custom alert rule

Reference: <https://docs.microsoft.com/bs-latn-ba/azure/asc-for-iot/how-to-security-data-access> <https://docs.microsoft.com/en-us/rest/api/securitycenter/devicesecuritygroups/createorupdate>



## Question Set 1

### QUESTION 1

You have an Azure IoT Edge device.

You need to modify the credentials used to access the container registry.

What should you modify?

- A. the @edgeHub module twin
- B. the IoT Edge module
- C. the \$edgeAgent module twin
- D. the Azure IoT Hub device twin

**Correct Answer:** C

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

The module twin for the IoT Edge agent is called \$edgeAgent and coordinates the communications between the IoT Edge agent running on a device and IoT Hub. The desired properties are set when applying a deployment manifest on a specific device as part of a single-device or at-scale deployment.

These properties include:

▪ runtime.settings.registryCredentials.{registryId}.username ▪  
runtime.settings.registryCredentials.{registryId}.password

Reference: <https://docs.microsoft.com/en-us/azure/iot-edge/module-edgeagent-edgehub>



**QUESTION 2** You enable Azure Security Center for IoT.

You need to onboard a device to Azure Security Center.

What should you do?

- A. Add the azureiotsecurity module identity to the Azure IoT Hub device identity.
- B. Open incoming TCP port 8883 on the device.
- C. Modify the connection string of the device.
- D. Install an X.509 certificate on the hardware security module (HSM) of the device.

**Correct Answer:** A

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

Use the following workflow to deploy and test your Azure Security Center for IoT security agents:

1. Enable Azure Security Center for IoT service to your IoT Hub
2. If your IoT Hub has no registered devices, Register a new device.
3. Create an azureiotsecurity security module for your devices.

Azure Security Center for IoT makes use of the module twin mechanism and maintains a security module twin named azureiotsecurity for each of your devices.

Note: To manually create a new azureiotsecurity module twin for a device use the following instructions:

1. In your IoT Hub, locate and select the device you wish to create a security module twin for.
2. Click on your device, and then on Add module identity.

3. In the Module Identity Name field, enter azureiotsecurity.
4. Click Save.

Reference:

<https://docs.microsoft.com/en-us/azure/asc-for-iot/quickstart-create-security-twin>

### QUESTION 3

You have an Azure IoT solution that includes an Azure IoT hub, 100 Azure IoT Edge devices, and 500 leaf devices.

You need to perform a key rotation across the devices.

Which three types of entities should you update? Each correct answer presents part of the solution.

**NOTE:** Each correct selection is worth one point.

- A. the \$edgeHub module identity
- B. the \$edgeAgent module identity
- C. the leaf module identities
- D. the IoT Edge device identities
- E. the iothubowner policy credentials
- F. the leaf device identities

**Correct Answer:** ADF

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

To get authorization to connect to IoT Hub, devices and services must send security tokens signed with either a shared access or symmetric key. These keys are stored with a device identity in the identity registry.

An IoT Hub identity registry can be accessed like a dictionary, by using the deviceId or moduleId as the key.

Reference: [https://docs.microsoft.com/bs-latn-ba/azure/iot-dps/how-to-control-](https://docs.microsoft.com/bs-latn-ba/azure/iot-dps/how-to-control-access)

[access https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-devguide-identity-registry](https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-devguide-identity-registry)

**QUESTION 4** You have an Azure IoT hub that is being taken from prototype to production.

You plan to connect IoT devices to the IoT hub. The devices have hardware security modules (HSMs).

You need to use the most secure authentication method between the devices and the IoT hub. Company policy prohibits the use of internally generated certificates.

Which authentication method should you use?

- A. an X.509 self-signed certificate
- B. a certificate thumbprint
- C. a symmetric key
- D. An X.509 certificate signed by a root certification authority (CA).

**Correct Answer:** D

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

Purchase X.509 certificates from a root certificate authority (CA). This method is recommended for production environments.

The hardware security module, or HSM, is used for secure, hardware-based storage of device secrets, and is the most secure form of secret storage. Both X.509 certificates and SAS tokens can be stored in the HSM Reference:

<https://docs.microsoft.com/en-us/azure/iot-dps/concepts-security>

#### QUESTION 5

DRAG DROP

You have an Azure IoT solution that includes an Azure IoT hub.

You receive a root certification authority (CA) certificate from the security department at your company.

You need to configure the IoT hub to use the root CA certificate.

Which four actions should you perform in sequence? To answer, move the appropriate actions from the list of actions to the answer area and arrange them in the correct order.

**Select and Place:**

**Correct Answer:**

**Section: [none]**

**Explanation**

**Explanation/Reference:**

Reference: <https://docs.microsoft.com/bs-latn-ba/azure/iot-hub/iot-hub-security-x509-get-started>

**QUESTION 6** You use Azure Security Center in an Azure IoT solution.

You need to exclude some security events. The solution must minimize development effort.

What should you do?



- A. Create an Azure function to filter security messages.
- B. Add a configuration to the code of the physical IoT device.
- C. Add configuration details to the device twin object.
- D. Create an azureiotsecurity module twin and add configuration details to the module twin object.

**Correct Answer: D**

**Section: [none]**

**Explanation**

**Explanation/Reference:**

Explanation:

Properties related to every Azure Security Center for IoT security agent are located in the agent configuration object, within the desired properties section, of the azureiotsecurity module.

To modify the configuration, create and modify this object inside the azureiotsecurity module twin identity.

Note: Azure Security Center for IoT's security agent twin configuration object is a JSON format object. The configuration object is a set of controllable properties that you can define to control the behavior of the agent.

These configurations help you customize the agent for each scenario required. For example, automatically excluding some events, or keeping power consumption to a minimal level are possible by configuring these properties.

Reference: <https://docs.microsoft.com/en-us/azure/asc-for-iot/how-to-agent-configuration>

**QUESTION 7** You have an Azure IoT hub that uses a Device Provisioning Service instance.

You have 1,000 legacy IoT devices that only support MAC address or serial number identities. The device do **NOT** have a security feature that can be used to securely identify the device or a hardware security module (HSM).

You plan to deploy the devices to a secure environment.

You need to configure the Device Provisioning Service instance to ensure that all the devices are identified securely before they receive updates.

Which attestation mechanism should you choose?

- A. Trusted Platform Module (TPM) 1.2 attestation
- B. symmetric key attestation
- C. X.509 certificates

**Correct Answer:** B

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:


A common problem with many legacy devices is that they often have an identity that is composed of a single piece of information. This identity information is usually a MAC address or a serial number. Legacy devices may not have a certificate, TPM, or any other security feature that can be used to securely identify the device. The Device Provisioning Service for IoT hub includes symmetric key attestation. Symmetric key attestation can be used to identify a device based off information like the MAC address or a serial number.




Reference: <https://docs.microsoft.com/bs-latn-ba/azure/iot-dps/how-to-legacy-device-symm-key>

#### **QUESTION 8**


From the Device Provisioning Service, you create an enrollment as shown in the exhibit. (Click the **Exhibit** tab.)




**enrollment1**  
 Enrollment Group Details

 Save
  Refresh
  Regenerate keys

**Settings**
 Registration Records

 You can view and update attestation information, set how you want to assign devices to hubs, define the re-provisioning policy and set the initial twin state of provisioning devices.

**Attestation Type**  
 Symmetric Key

Primary Key  
 \*\*\*\*\*

Secondary Key  
 \*\*\*\*\*

IoT Edge device ⓘ  
 True False

Select how you want to assign devices to hubs  
 Evenly weighted distribution

Select the IoT hubs this group can be assigned to: ⓘ  
 iothub-contoso.azure-devices.net

Link a new IoT hub

Select how you want device data to be handled on re-provisioning \* ⓘ  
 Re-provision and migrate data

Enable entry ⓘ  
 Enable Disable

You need to deploy a new IoT device.

What should you use as the device identity during attestation?

- A. a self-signed X.509 certificate
- B. the random string of alphanumeric characters
- C. the HMACSHA256 hash of the device's registration ID
- D. the endorsement key of the device's Trusted Platform Module (TPM)

**Correct Answer: C**

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

Each device uses its derived device key with your unique registration ID to perform symmetric key attestation with the enrollment during provisioning. To generate the device key, use the key you copied from your DPS enrollment to compute an HMAC-SHA256 of the unique registration ID for the device and convert the result into Base64 format.

Reference: <https://docs.microsoft.com/en-us/azure/iot-edge/how-to-auto-provision-symmetric-keys>

