HOTSPOT

You create a new IoT device named device1 on iothub1. The primary key value assigned to device1 is 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.

EXPLANATION

Answer:

Answer Area

HostName= [azure-devices.net / criticalep / device1 / **iothub1** / tracestate] . [**azure-devices.net** / criticalep / device1 / iothub1 / tracestate] ;DeviceId= [azure-devices.net / criticalep / **device1** / iothub1 / tracestate] :SharedAccessKey=Uihuih76hbHb

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

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 section, 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

EXPLANATION

Answer: A
Your Answer: No answer

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

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 section, 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

EXPLANATION

Answer: B
Your Answer: No answer

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

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 section, 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

EXPLANATION

Answer: B
Your Answer: No answer

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

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.

EXPLANATION

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

Explanation:

Box 1: tags.engine.warpDriveType='VM105a'
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

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

EXPLANATION

Answer: D
Your Answer: No answer

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/

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.

EXPLANATION

Answer: AC
Your Answer: No answer

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

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

EXPLANATION

Answer: D
Your Answer: No answer

Explanation:
2500* 6 KB * 12 = 180,000 KB/minute = 180 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

DRAG DROP

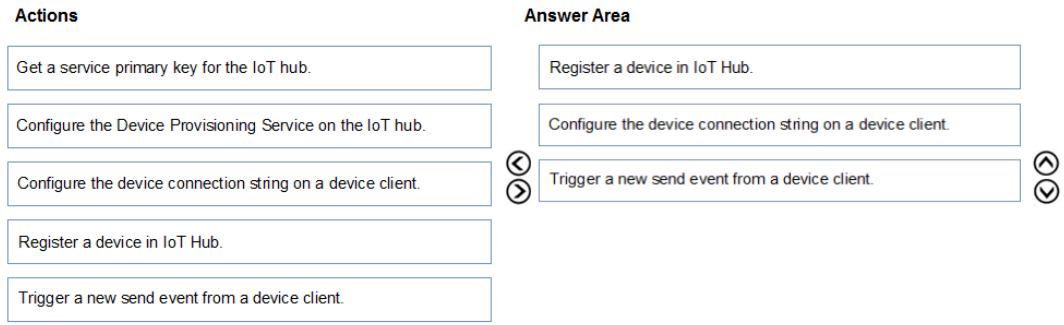
You deploy an Azure IoT hub.

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.

EXPLANATION

Answer:



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

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 **NOT** 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 **NOT** 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.

EXPLANATION

Answer:

Protocols

AMQP

HTTPS

MQTT

Answer Area

Device	Protocol
Transparent Field Gateway Device:	AMQP
Low Resource Device:	MQTT
Limited Sensor Device:	HTTPS

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

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 MylotHub support?

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

EXPLANATION

Answer: A
Your Answer: No answer

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

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

EXPLANATION

Answer: ACE
Your Answer: No answer

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

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 upstream protocol of the devices to use MQTT over TCP.
- 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. Integrate cellular communication hardware onto the devices and avoid the use of the external networks.

EXPLANATION

Answer: B
Your Answer: No answer

Explanation:
MQTT over WebSockets uses port 443.

Note: Devices can communicate with IoT Hub in Azure using various protocols. Typically, the choice of protocol is driven by the specific requirements of the solution. The following table lists the outbound ports that must be open for a device to be able to use a specific protocol:

Protocol	Port
MQTT	8883
MQTT over WebSockets	443
AMQP	5671
AMQP over WebSockets	443
HTTPS	443

prepaway
praw731500

Incorrect Answers:
A: MQTT over TCP uses port 883.
C: ExpressRoute uses BGP, which uses TCP port 179.
D: HTTPS proxy also uses port 443, but it would be a more expensive solution.

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

You have 100 devices that connect to an Azure IoT hub named Hub1. The devices connect by using a symmetric key.

You deploy an IoT hub named Hub2.

You need to migrate 10 devices from Hub1 to Hub2. The solution must ensure that the devices retain the existing symmetric key.

What should you do?

- A. Add a desired property to the device twin of Hub2. Update the endpoint of the 10 devices to use Hub2.
- B. Add a desired property to the device twin of Hub1. Recreate the device identity on Hub2.
- C. Recreate the device identity on Hub2. Update the endpoint of the 10 devices to use Hub2.
- D. Disable the 10 devices on Hub1. Update the endpoint of the 10 devices to use Hub2.

EXPLANATION

Answer: B
Your Answer: No answer

Explanation:
Desired properties. Used along with reported properties to synchronize device configuration or conditions. The solution back end can set desired properties, and the device app can read them. The device app can also receive notifications of changes in the desired properties.

Reference:
https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-devguide-device-twins

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 diagnostic logs and set up alerts for the connected devices count metric.

EXPLANATION

Answer: D
Your Answer: No answer

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

HOTSPOT

You are writing code to provision IoT devices by using the Device Provisioning Service.

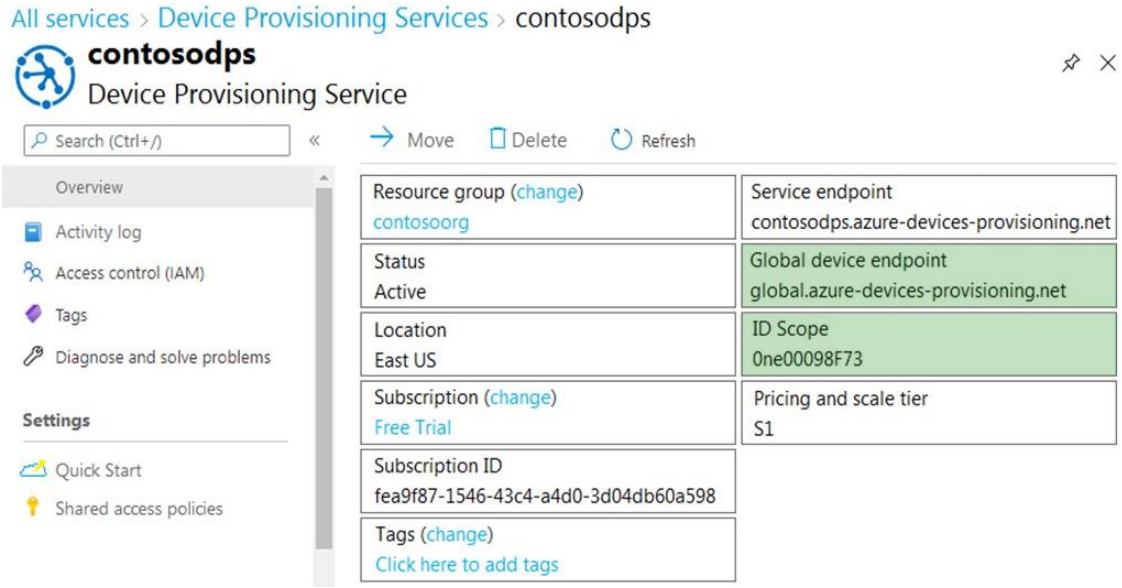
Which two details from the Overview blade of the Device Provisioning Service are required to provision a new IoT client device? To answer, select the appropriate detail in the answer area.

NOTE: Each correct selection is worth one point.

EXPLANATION

Answer:

Answer Area



Explanation:

Box 1: ID Scope
In the Azure portal, select the Overview blade for your Device Provisioning service and copy the ID Scope value. The ID Scope is generated by the service and guarantees uniqueness. It is immutable and used to uniquely identify the registration IDs.

Box 2: Global device endpoint
The global_prov_uri variable, which allows the IoT Hub client registration API IoTHubClient_LL_CreateFromDeviceAuth to connect with the designated Device Provisioning Service instance.

Example code:
static const char* global_prov_uri = "global.azure-devices-provisioning.net";
static const char* id_scope = "[ID Scope]";

Reference:
https://docs.microsoft.com/en-us/azure/iot-dps/tutorial-set-up-device

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 section, 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

EXPLANATION

Answer: B
Your Answer: No answer

Explanation:
Instead add tags to the device twin. Desired properties are synced, while tags are not.

Incorrect Answers:
A: 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://docs.microsoft.com/de-de/azure/iot-hub/iot-hub-devguide-device-twins

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

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.

EXPLANATION

Answer: D
Your Answer: No answer

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/

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.

EXPLANATION

Answer: AC
Your Answer: No answer

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

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.

EXPLANATION

Answer: CE
Your Answer: No answer

Explanation:
In general, deprovisioning a device involves two steps:
- 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.
- 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/en-us/azure/iot-dps/how-to-unprovision-devices

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.

EXPLANATION

Answer:

Answer Area

Return the reported power consumption:	Measurement
Configure the desired fan speed:	Properties
Read the fan serial number:	Properties
Run the device reset routine:	Command

Explanation:

A device template in Azure IoT Central is a blueprint that defines the:
- Telemetry a device sends to IoT Central.
- Properties a device synchronizes with IoT Central.
- Commands that IoT Central calls on a device.

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: Properties
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: Properties

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

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.

EXPLANATION

Answer:

Actions	Answer Area
Flash unique credentials to the devices.	Obtain the credentials.
Obtain the credentials.	Generate device credentials.
Generate device credentials.	Flash unique credentials to the devices.
Associate the devices to a template and approve the connections.	Connect the devices to IoT Central.
Connect the devices to IoT Central.	Associate the devices to a template and approve the connections.

Explanation:

Step 1: Obtain the credential
Obtain the group primary key from the IoT Central enrollment group.

Step 2: Generate device credentials
The group primary key used to generate device credentials

Step 3: Flash unique credentials to the devices
The OEM flashes each device with a device ID, a generated device SAS key, and the application ID scope value.

Step 4: Connect the devices to IoT Central

Step 5: Associate the devices to a template and approve the connections

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

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

EXPLANATION

Answer: AC
Your Answer: No answer

Explanation:
Required connection information:
- Group primary key: In your IoT Central application, navigate to Administration > Device Connection > SAS-IoT-Devices. Make a note of the shared access signature Primary key value.
- ID scope: In your IoT Central application, navigate to Administration > Device Connection. Make a note of the ID scope value.

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

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.

EXPLANATION

Answer: BCF
Your Answer: No answer

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

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.

EXPLANATION

Answer: B
Your Answer: No answer

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/

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

EXPLANATION

Answer: A
Your Answer: No answer

Explanation:
The registration ID is used to uniquely identify a device registration with the Device Provisioning Service. The device ID must be unique in the provisioning service ID scope. Each device must have a registration ID.

Note: An individual enrollment is an entry for a single device that may register. Individual enrollments may use either X.509 leaf certificates or SAS tokens (from a physical or virtual TPM) as attestation mechanisms. The registration ID in an individual enrollment is alphanumeric, lowercase, and may contain hyphens.

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

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

EXPLANATION

Answer: D
Your Answer: No answer

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

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 section, 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

EXPLANATION

Answer: B
Your Answer: No answer

Explanation:
Instead tags should be added to the Device twin.

Tags: A section of the JSON document that the solution back end can read from and write to. Tags are not visible to device apps.

Reference:
https://docs.microsoft.com/de-de/azure/iot-hub/iot-hub-devguide-device-twins

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

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 section, 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

EXPLANATION

Answer: A
Your Answer: No answer

Explanation:
Tags are not synced.

Tags: A section of the JSON document that the solution back end can read from and write to. Tags are not visible to device apps.

Reference:
https://docs.microsoft.com/de-de/azure/iot-hub/iot-hub-devguide-device-twins

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

You have an existing Azure IoT hub.

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

Which two 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.

EXPLANATION

Answer: BC
Your Answer: No answer

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

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

Each device has a property tag named city that is used to store the location of the device.

You need to update the properties on all the devices located at an office in the city of Seattle as quickly as possible. Any new devices in the Seattle office that are added to the IoT hub must receive the updated properties also.

What should you do?

- A. From Automatic Device Management, create an IoT device configuration.
- B. From the IoT hub, generate a query for the target devices.
- C. Create a scheduled job by using the IoT Hub service SDKs.
- D. Deploy an Azure IoT Edge transparent gateway to the Seattle office and deploy an Azure Stream Analytics edge job.

EXPLANATION

Answer: A
Your Answer: No answer

Explanation:
Automatic device management in Azure IoT Hub automates many of the repetitive and complex tasks of managing large device fleets. With automatic device management, you can target a set of devices based on their properties, define a desired configuration, and then let IoT Hub update the devices when they come into scope. This update is done using an automatic device configuration or automatic module configuration, which lets you summarize completion and compliance, handle merging and conflicts, and roll out configurations in a phased approach.

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

You have an Azure IoT Central application.

You add an IoT device named Oven1 to the application. Oven1 uses an IoT Central template for industrial ovens.

You need to send an email to the managers group at your company as soon as the oven temperature falls below 400 degrees.

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

NOTE: Each correct selection is worth one point.

- [ ] A. Create a SendGrid account in the same resource group as the IoT Central application.
- [ ] B. Add a condition that has Time Aggregation set to Off.
- [ ] C. Add a condition that has Aggregation set to Minimum.
- [ ] D. Add the Manager role to the IoT Central application.
- [ ] E. From IoT Central, create a telemetry rule for the template.

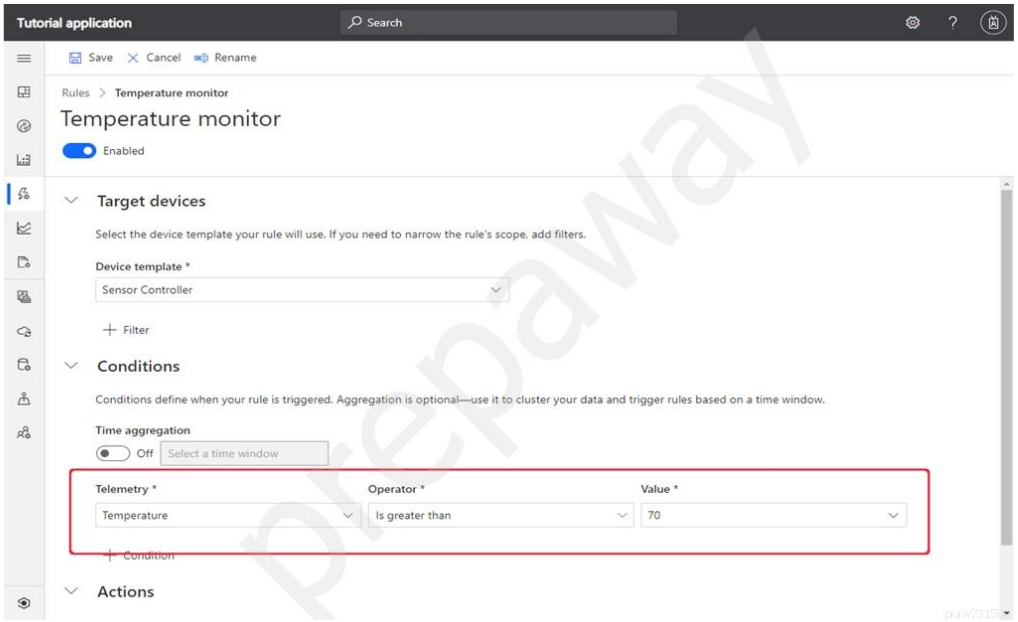
EXPLANATION

Answer: BE
Your Answer: No answer

Explanation:
Devices use telemetry to send numerical data from the device. A rule triggers when the selected telemetry crosses a specified threshold.

E: To create a telemetry rule, the device template must include at least one telemetry value. The rule monitors the temperature reported by the device and sends an email when it falls below 400 degrees.

B: Configure the rule conditions.
Conditions define the criteria that the rule monitors. In this tutorial, you configure the rule to fire when the temperature exceeds 70° F.
1. Select Temperature in the Telemetry dropdown.
2. Next, choose Is less than as the Operator and enter 400 as the Value.



3. Optionally, you can set a Time aggregation. When you select a time aggregation, you must also select an aggregation type, such as average or sum from the aggregation drop-down.
- Without aggregation, the rule triggers for each telemetry data point that meets the condition.
- With aggregation, the rule triggers if the aggregate value of the telemetry data points in the time window meets the condition.

Reference:
https://docs.microsoft.com/en-us/azure/iot-central/core/tutorial-create-telemetry-rules

You have an Azure IoT solution that includes multiple Azure IoT hubs in different geographic locations and a single Device Provision Service instance.

You need to configure device enrollment to assign devices to the appropriate IoT hub based on the following requirements:

- The registration ID of the device
- The geographic location of the device

The load between the IoT hubs in the same geographic location must be balanced.

What should you use to assign the devices to the IoT hubs?

- A. Static configuration (via enrollment list only)
- B. Lowest latency
- C. Evenly weighted distribution
- D. Custom (Use Azure Function)

EXPLANATION

Answer: A
Your Answer: No answer

Explanation:
Set the Device Provisioning Service allocation policy
The allocation policy is a Device Provisioning Service setting that determines how devices are assigned to an IoT hub. There are three supported allocation policies:
- Lowest latency: Devices are provisioned to an IoT hub based on the hub with the lowest latency to the device.
- Evenly weighted distribution (default): Linked IoT hubs are equally likely to have devices provisioned to them. This is the default setting. If you are provisioning devices to only one IoT hub, you can keep this setting.
- Static configuration via the enrollment list: Specification of the desired IoT hub in the enrollment list takes priority over the Device Provisioning Service-level allocation policy.

Reference:
https://docs.microsoft.com/en-us/azure/iot-dps/tutorial-provision-multiple-hubs

You are developing an Azure IoT Central application.

You add a new custom device template to the application.

You need to add a fixed location value to the device template. The value must be updated by the physical IoT device, read-only to device operators, and not graphed by IoT Central.

What should you add to the device template?

- A. a Location property
- B. a Location telemetry
- C. a Cloud property

EXPLANATION

Answer: A
Your Answer: No answer

Explanation:
For example, a builder can create a device template for a connected fan that has the following characteristics:
- Sends temperature telemetry
- Sends location property

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

DRAG DROP

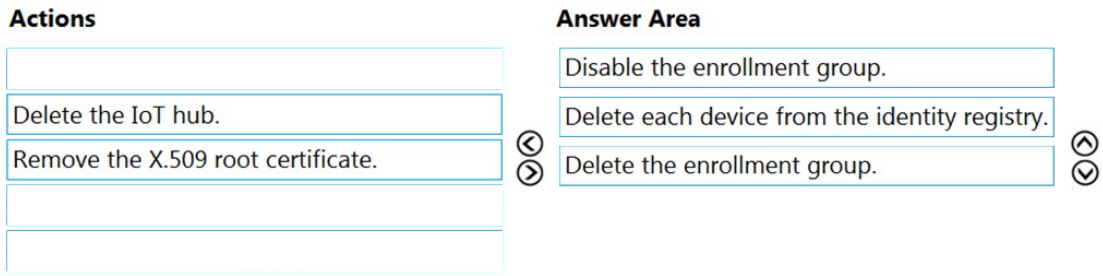
You have an Azure IoT solution that includes an Azure IoT hub, a Device Provisioning Service instance, and 1,000 connected IoT devices. The IoT devices are allocated to tour enrollment groups. Each enrollment group is configured to use certificate attestation.

You need to decommission all the devices in a single enrollment group and the enrollment group itself.

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.

EXPLANATION

Answer:



Explanation:

To deprovision all of the devices that have been provisioned through an enrollment group:

1. Disable the enrollment group to disallow its signing certificate.
2. Use the list of provisioned devices for that enrollment group to disable or delete each device from the identity registry of its respective IoT hub.
3. After disabling or deleting all devices from their respective IoT hubs, you can optionally delete the enrollment group. Be aware, though, that, if you delete the enrollment group and there is an enabled enrollment group for a signing certificate higher up in the certificate chain of one or more of the devices, those devices can re-enroll.

Reference:
https://docs.microsoft.com/en-us/azure/iot-dps/how-to-unprovision-devices

You have an Azure IoT hub that uses a Device Provision Service instance.

You plan to deploy 100 IoT devices.

You need to confirm the identity of the devices by using the Device Provision Service.

Which three device attestation mechanisms can you use? Each correct answer presents a complete solution.

NOTE: Each correct selection is worth one point.

- [ ] A. X.509 certificates
- [ ] B. Trusted Platform Module (TPM) 2.0
- [ ] C. Trusted Platform Module (TPM) 1.2
- [ ] D. Symmetric key
- [ ] E. Device Identity Composition Engine (DICE)

EXPLANATION

Answer: ABD
Your Answer: No answer

Explanation:
The Device Provisioning Service supports the following forms of attestation:
- X.509 certificates based on the standard X.509 certificate authentication flow.
- Trusted Platform Module (TPM) based on a nonce challenge, using the TPM 2.0 standard for keys to present a signed Shared Access Signature (SAS) token. This does not require a physical TPM on the device, but the service expects to attest using the endorsement key per the TPM spec.
- Symmetric Key based on shared access signature (SAS) Security tokens, which include a hashed signature and an embedded expiration.

Reference:
https://docs.microsoft.com/en-us/azure/iot-dps/concepts-service#attestation-mechanism

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 section, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.

You have a Standard tier Azure IoT hub and a fleet of IoT devices.

The devices connect to the IoT hub by using either Message Queuing Telemetry Transport (MQTT) or Advanced Message Queuing Protocol (AMQP).

You need to send data to the IoT devices and each device must respond. Each device will require three minutes to process the data and respond.

Solution: You update the twin desired property and check the corresponding reported property.

Does this meet the goal?

A. Yes

B. No

EXPLANATION

Answer: A
Your Answer: No answer

Explanation:
IoT Hub provides three options for device apps to expose functionality to a back-end app:
- Twin's desired properties for long-running commands intended to put the device into a certain desired state. For example, set the telemetry send interval to 30 minutes.
- Direct methods for communications that require immediate confirmation of the result. Direct methods are often used for interactive control of devices such as turning on a fan.
- Cloud-to-device messages for one-way notifications to the device app.

Reference:
https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-devguide-c2d-guidance

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 section, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.

You have a Standard tier Azure IoT hub and a fleet of IoT devices.

The devices connect to the IoT hub by using either Message Queuing Telemetry Transport (MQTT) or Advanced Message Queuing Protocol (AMQP).

You need to send data to the IoT devices and each device must respond. Each device will require three minutes to process the data and respond.

Solution: You use direct methods and check the response.

Does this meet the goal?

- A. Yes
- B. No

EXPLANATION

Answer: B
Your Answer: No answer

Explanation:
IoT Hub provides three options for device apps to expose functionality to a back-end app:
- Twin's desired properties for long-running commands intended to put the device into a certain desired state. For example, set the telemetry send interval to 30 minutes.
- Direct methods for communications that require immediate confirmation of the result. Direct methods are often used for interactive control of devices such as turning on a fan.
- Cloud-to-device messages for one-way notifications to the device app.

Reference:
https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-devguide-c2d-guidance

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 section, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.

You have a Standard tier Azure IoT hub and a fleet of IoT devices.

The devices connect to the IoT hub by using either Message Queuing Telemetry Transport (MQTT) or Advanced Message Queuing Protocol (AMQP).

You need to send data to the IoT devices and each device must respond. Each device will require three minutes to process the data and respond.

Solution: You use cloud-to-device messages and watch the cloud-to-device feedback endpoint for successful acknowledgement.

Does this meet the goal?

- A. Yes
- B. No

EXPLANATION

Answer: B
Your Answer: No answer

Explanation:
IoT Hub provides three options for device apps to expose functionality to a back-end app:
- Twin's desired properties for long-running commands intended to put the device into a certain desired state. For example, set the telemetry send interval to 30 minutes.
- Direct methods for communications that require immediate confirmation of the result. Direct methods are often used for interactive control of devices such as turning on a fan.
- Cloud-to-device messages for one-way notifications to the device app.

Reference:
https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-devguide-c2d-guidance

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 section, 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 connecting to the IoT hub.

Solution: You disconnect the Device Provisioning Service from the IoT hub.

Does this meet the goal?

- A. Yes
- B. No

EXPLANATION

Answer: B
Your Answer: No answer

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

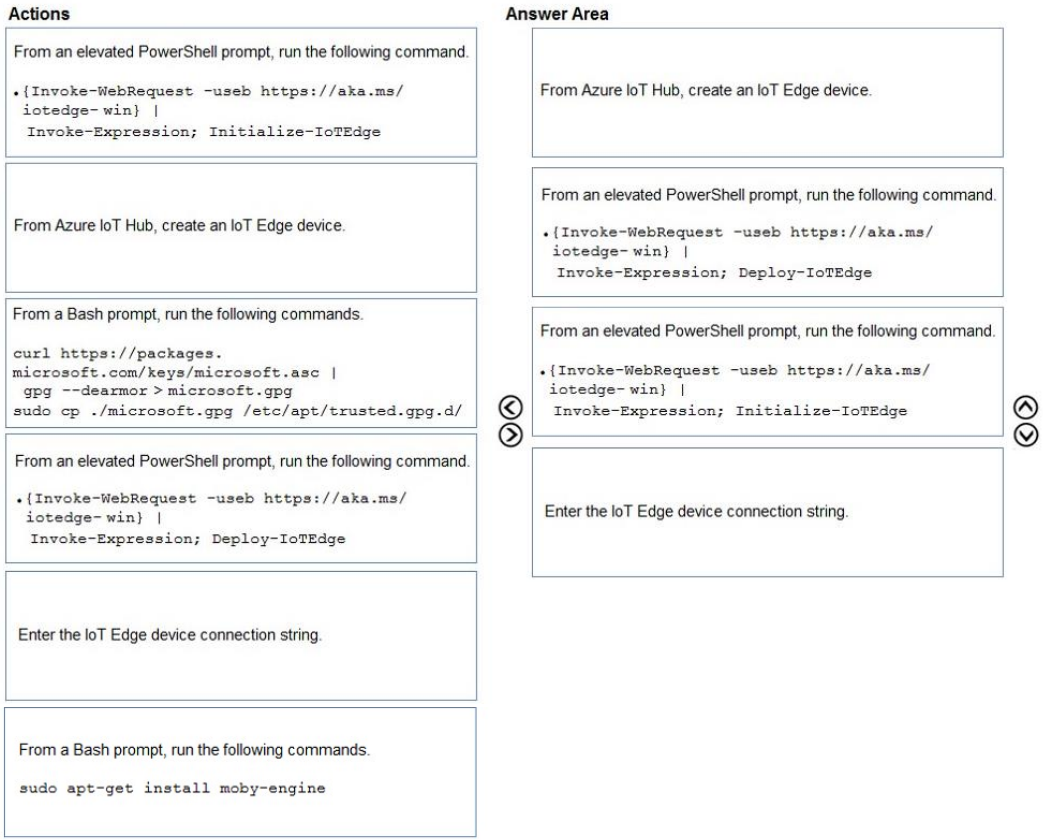
DRAG DROP

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.

EXPLANATION

Answer:



Explanation:

Step 1: From Azure IoT Hub, create an IoT Edge Device

Step 2: Deploy-IoTEdge
The Deploy-IoTEdge 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-IoTEdge

Step 3: Initialize-IoTEdge
The Initialize-IoTEdge 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

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
```

EXPLANATION

Answer: D
Your Answer: No answer

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

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.

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.

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

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
```

Explanation:

Step1: Install the IoT edge repository for Ubuntu Server 18.04 on the physical device. From IoT hub, create a new IoT Edge device.

Prepare your device to access the Microsoft installation packages.
Install the repository configuration that matches your device operating system.

Ubuntu Server 18.04: curl https://packages.microsoft.com/config/ubuntu/18.04/multiarch/prod.list > ./microsoft-prod.list

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.

Step 2: 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 3: Add the connection string to the /etc/iotedge/config.yaml file,..
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

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

EXPLANATION

Answer: BC
Your Answer: No answer

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

HOTSPOT

You have the following device twin for the IoT device.


```
{
  "deviceId": "device1",
  "etag": "AAAAAAAAAk=",
  "deviceEtag": "NDcwMTU4Mzk=",
  "status": "enabled",
  "statusUpdateTime": "0001-01-01T00:00:00Z",
  "connectionState": "Disconnected",
  "lastActivityTime": "2019-10-21T22:45:57.9732805Z",
  "cloudToDeviceMessageCount": 0,
  "authenticationType": "sas",
  "x509Thumbprint": {
    "primaryThumbprint": null,
    "secondaryThumbprint": null
  },
  "version": 17,
  "properties": {
    "desired": {
      "$metadata": {
        "$lastUpdated": "2019-10-24T19:40:46.4809147Z",
        "$lastUpdatedVersion": 9
      },
      "$version": 9
    },
    "reported": {
      "fanSpeed": 73,
      "$metadata": {
        "$lastUpdated": "2019-10-24T19:41:28.8839751Z",
        " fanSpeed": {
        "$lastUpdated": "2019-10-24T19:41:28.8839751Z"
        }
      },
      "$version": 8
    }
  },
  "capabilities": {
    "iotEdge": false
  }
}
```


For each of the following statements, select Yes if the statement is true. Otherwise, select No.

NOTE: Each correct selection is worth one point.

EXPLANATION

Answer:

Statements	Yes	No
You can add a property that contains multiple nested values to the device twin.	◯	○
The device twin will set `fanSpeed` for the physical IoT device to `73`.	◯	○
You can change the device identity of the physical IoT device by modifying the `deviceId` property.	○	◯

Explanation:

Box1: Yes

Box 2: Yes
Fanspeed 73 is a reported property.

Box 3: No
The deviceID property is read only.

Reference:
https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-devguide-device-twins

You are deploying an Azure IoT Edge solution that includes multiple IoT Edge devices.

You need to configure module-to-module routing.

To which section of the deployment manifest should you add the routes?

- A. `storeAndForwardConfiguration`
- B. `$edgeHub`
- C. `modules`
- D. `systemModules`

EXPLANATION

Answer: B
Your Answer: No answer

Explanation:
Routes are declared in the $edgeHub desired properties.

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

You have an IoT device that has the following configurations:

- Hardware: Raspberry Pi
- Operating system: Raspbian

You need to deploy Azure IoT Edge to the device.

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 IoT Edge runtime.
- B. Install the IoT Edge security daemon.
- C. Run the `Deploy-IoTEdge` PowerShell cmdlet on the IoT Edge device.
- D. Install the container runtime.

EXPLANATION

Answer: AB
Your Answer: No answer

Explanation:
The Azure IoT Edge runtime is what turns a device into an IoT Edge device. The runtime can be deployed on devices as small as a Raspberry Pi or as large as an industrial server.

The IoT Edge security daemon provides and maintains security standards on the IoT Edge device. The daemon starts on every boot and bootstraps the device by starting the rest of the IoT Edge runtime.

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

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`

EXPLANATION

Answer: BDE
Your Answer: No answer

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

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

EXPLANATION

Answer: B
Your Answer: No answer

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

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)`

EXPLANATION

Answer: B
Your Answer: No answer

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

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.

EXPLANATION

Answer:

Answer Area

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

Reference:
https://docs.microsoft.com/bs-cyrl-ba/azure/iot-hub/iot-hub-message-enrichments-overview

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.

EXPLANATION

Answer: AC
Your Answer: No answer

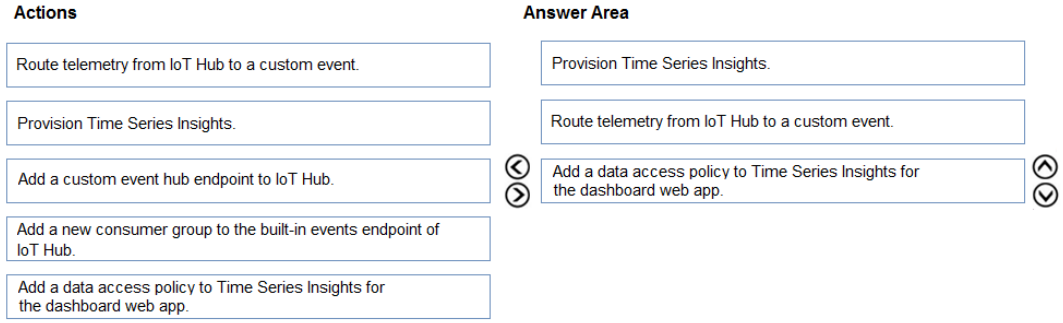
DRAG DROP

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.

EXPLANATION

Answer:



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

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

EXPLANATION

Answer: A
Your Answer: No answer

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

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

What should you recommend?

A. XML

B. Avro

C. JSON

EXPLANATION

Answer: C
Your Answer: No answer

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

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.

EXPLANATION

Answer: C
Your Answer: No answer

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.

EXPLANATION

Answer:

Actions	Answer Area
Configure the Time Series Insights event source to connect to an existing IoT hub.	Create a dedicated consumer group in the built-in events endpoints of the IoT hub.
Create an Azure event hub.	Add a new Time Series Insights event source.
Add a new Time Series Insights event source.	Configure the Time Series Insights event source to connect to an existing IoT hub.
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.	

Explanation:

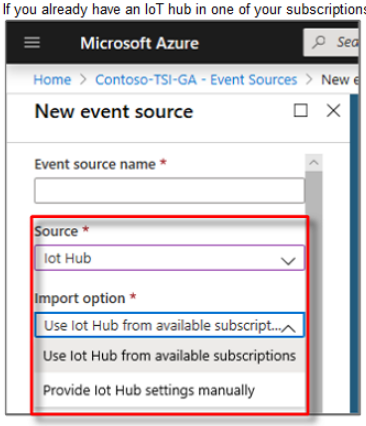
Step 1: Create a dedicated consumer group..
Add a consumer group to your IoT hub.
Applications use consumer groups to pull data from Azure IoT Hub. To reliably read data from your IoT hub, provide a dedicated consumer group that's used only by this Time Series Insights environment.

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.

Step 3: Configure the Time Series event source to connect to an existing IOT 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-iothub

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`".

EXPLANATION

Answer: CE
Your Answer: No answer

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

You have an Azure IoT hub.

You plan to implement IoT Hub events by using Azure Event Grid.

You need to send an email when the following events occur:

- Device Created
- Device Deleted
- Device Connected
- Device Disconnected

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

NOTE: Each correct selection is worth one point.

- A. From the IoT hub, configure an event subscription that has API management as the Endpoint Type.
- B. From the IoT hub, configure an event subscription that has Web Hook as the Endpoint Type.
- C. Create an Azure logic app that has a Request trigger.
- D. From the IoT hub, configure an event subscription that has Service Bus Queue as the Endpoint Type.
- E. Create an Azure logic app that has a scheduled trigger.

EXPLANATION

Answer: BC
Your Answer: No answer

Explanation:
For non-telemetry events like DeviceConnected, DeviceDisconnected, DeviceCreated and DeviceDeleted, the Event Grid filtering can be used when creating the subscription.

C: Azure Event Grid enables you to react to events in IoT Hub by triggering actions in your downstream business applications.
A trigger, such as a Request trigger, is a specific event that starts your logic app.

Reference:
https://docs.microsoft.com/en-us/azure/event-grid/publish-iot-hub-events-to-logic-apps

HOTSPOT

You create an Azure Stream Analytics job that has the following query.

```
SELECT
    Count(*) AS dailyCount,
    System.Timestamp() AS time
INTO FunctionOutput
FROM IotHubInput TIMESTAMP BY deviceTime
GROUP BY TumblingWindow(hour, 24)
```

The job is configured to have an Azure IoT Hub input and an output to an Azure function.

For each of the following statements, select Yes if the statement is true. Otherwise, select No.

NOTE: Each correct selection is worth one point.

EXPLANATION

Answer:

Answer Area

Statements	Yes	No
The function will be invoked at midnight UTC.	○	○
The function will be invoked only when the IoT hub receives telemetry.	○	○
When the Stream Analytics job is restarted, the function can be invoked more than once in a 24-hour period.	○	○

Explanation:

Box 1: Yes
All time handling operations in Azure Stream Analytics are in UTC.

Box 2: No
Tumbling windows are a series of fixed-sized, non-overlapping and contiguous time intervals.

Box 3: Yes

Reference:
https://docs.microsoft.com/en-us/stream-analytics-query/time-management-azure-stream-analytics

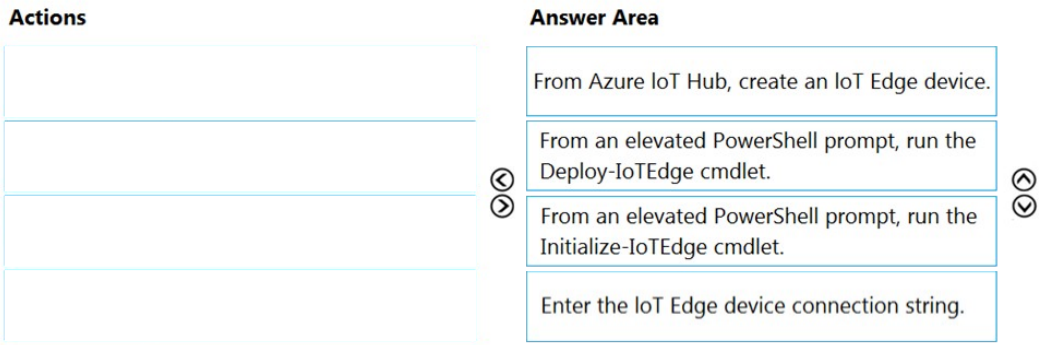
DRAG DROP

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

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

EXPLANATION

Answer:



Explanation:

Step 1: From Azure IoT hub, create an IoT Edge device
In the Azure Cloud Shell, enter the following command to create a device named myEdgeDevice in your hub.
az iot hub device-identity create --device-id myEdgeDevice --edge-enabled --hub-name {hub_name}

View the connection string for your device, which links your physical device with its identity in IoT Hub. Copy the value of the connectionString key from the JSON output and save it. This value is the device connection string. You'll use this connection string to configure the IoT Edge runtime in the step 3.

Step 2: From an elevated PowerShell prompt, run the Deploy-IoTEdge cmdlet.
Install the Azure IoT Edge runtime on your IoT Edge device.
1. Run PowerShell as an administrator.
2. Run the Deploy-IoTEdge command, which performs the following tasks:
   - Checks that your Windows machine is on a supported version.
   - Turns on the containers feature.
   - Downloads the moby engine and the IoT Edge runtime.

Step 3: From an elevated PowerShell prompt, run the Initialize-IoTEdge cmdlet

Step 4: Enter the IoT Edge device connection string.
Configure the IoT Edge device with a device connection string.

Reference:
https://docs.microsoft.com/en-us/azure/iot-edge/quickstart

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 section, 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 Stream Analytics job that receives input from an Azure IoT hub and sends the outputs to Azure Blob storage. The job has compatibility level 1.1 and six streaming units.

You have the following query for the job.

```
SELECT COUNT(*) AS Count, TollBoothID
INTO BlobOutput
FROM IotHubInput
GROUP BY TumblingWindow(minute, 3), TollBoothID
```

You plan to increase the streaming unit count to 12.

You need to optimize the job to take advantage of the additional streaming units and increase the throughput.

Solution: You change the query to the following.

```
WITH Step1 AS (
SELECT COUNT(*) AS Count, TollBoothID, PartitionID
FROM IoTHubInput PARTITION BY PartitionID
GROUP BY TumblingWindow(minute, 3), TollBoothID, PartitionID
)
SELECT SUM(Count) AS Count, TollBoothID
INTO BlobOutput
FROM Step1
GROUP BY TumblingWindow(minute, 3), TollBoothID
```

Does this meet the goal?

- A. Yes
- B. No

EXPLANATION

Answer: A
Your Answer: No answer

Explanation:
Max number of Streaming Units with one step and with no partitions is 6.

Reference:
https://docs.microsoft.com/en-us/azure/stream-analytics/stream-analytics-parallelization

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 section, 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 Stream Analytics job that receives input from an Azure IoT hub and sends the outputs to Azure Blob storage. The job has compatibility level 1.1 and six streaming units.

You have the following query for the job.

```
SELECT COUNT(*) AS Count, TollBoothID
INTO BlobOutput
FROM IotHubInput
GROUP BY TumblingWindow(minute, 3), TollBoothID
```

You plan to increase the streaming unit count to 12.

You need to optimize the job to take advantage of the additional streaming units and increase the throughput.

Solution: You change the query to the following.

```
SELECT COUNT(*) AS Count, TollBoothID
INTO BlobOutput
FROM IotHubInput PARTITION BY PartitionID
GROUP BY TumblingWindow(minute, 3), TollBoothID, PartitionID
```

Does this meet the goal?

- A. Yes
- B. No

EXPLANATION

Answer: B
Your Answer: No answer

Explanation:
Max number of Streaming Units with one step and with no partitions is 6.

Reference:
https://docs.microsoft.com/en-us/azure/stream-analytics/stream-analytics-parallelization

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 section, 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 Stream Analytics job that receives input from an Azure IoT hub and sends the outputs to Azure Blob storage. The job has compatibility level 1.1 and six streaming units.

You have the following query for the job.

```
SELECT COUNT(*) AS Count, TollBoothID
INTO BlobOutput
FROM IotHubInput
GROUP BY TumblingWindow(minute, 3), TollBoothID
```

You plan to increase the streaming unit count to 12.

You need to optimize the job to take advantage of the additional streaming units and increase the throughput.

Solution: You change the compatibility level of the job to 1.2.

Does this meet the goal?

- A. Yes
- B. No

EXPLANATION

Answer: B
Your Answer: No answer

Explanation:
Max number of Streaming Units with one step and with no partitions is 6.

Reference:
https://docs.microsoft.com/en-us/azure/stream-analytics/stream-analytics-parallelization

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

EXPLANATION

Answer: CD
Your Answer: No answer

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.

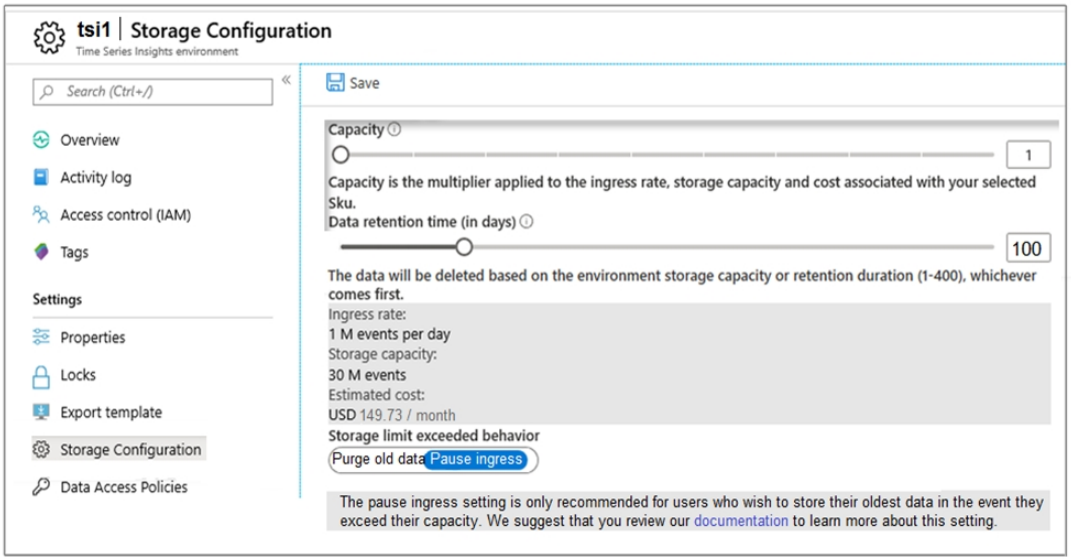
Connection: The name of an app setting that contains the connection string to the event hub's namespace. Copy this connection string by clicking the Connection Information button for the namespace, not the event hub itself. This connection string must have send permissions to send the message to the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

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

HOTSPOT

You have an Azure IoT hub named Hub1 and an Azure Time Series Insights environment named tsi1. Tsi1 connects to Hub1. The solution has been operational for 6 months.

Tsi1 is configured as shown in the following exhibit.



Hub1 receives 1 million messages per day. Each message is up to 1 KB and is formatted as JSON.

Hub1 has seven days of retained telemetry.

For each of the following statements, select Yes if the statement is true. Otherwise, select No.

NOTE: Each correct selection is worth one point.

EXPLANATION

Answer:

Answer Area

Statement	Yes	No
Tsi1 will display 100 days of telemetry.	○	○
Tsi1 will display telemetry that arrived three months ago.	○	○
Tsi1 will display real-time data after the Time Series Insights environment has been connected to the event source of Hub1 for two days.	○	○

Reference:
https://docs.microsoft.com/en-us/azure/time-series-insights/time-series-insights-overview

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.

EXPLANATION

Answer: D
Your Answer: No answer

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

During the POV phase, you connect a device to IoT Hub and start sending telemetry messages.

You need to verify the content of the messages received by IoT Hub during the POV phase.

What should you use?

- A. the Monitoring settings of IoT Hub or a Postman call to the IoT Hub REST API
- B. Azure Monitor or Azure Log Analytics
- C. Microsoft Visual Studio Code that uses the IoT Hub Toolkit or Azure CLI that uses the IoT Hub extension
- D. Splunk or Grafana

EXPLANATION

Answer: B
Your Answer: No answer

Reference:
https://docs.microsoft.com/en-us/azure/iot-hub/tutorial-use-metrics-and-diags

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`

EXPLANATION

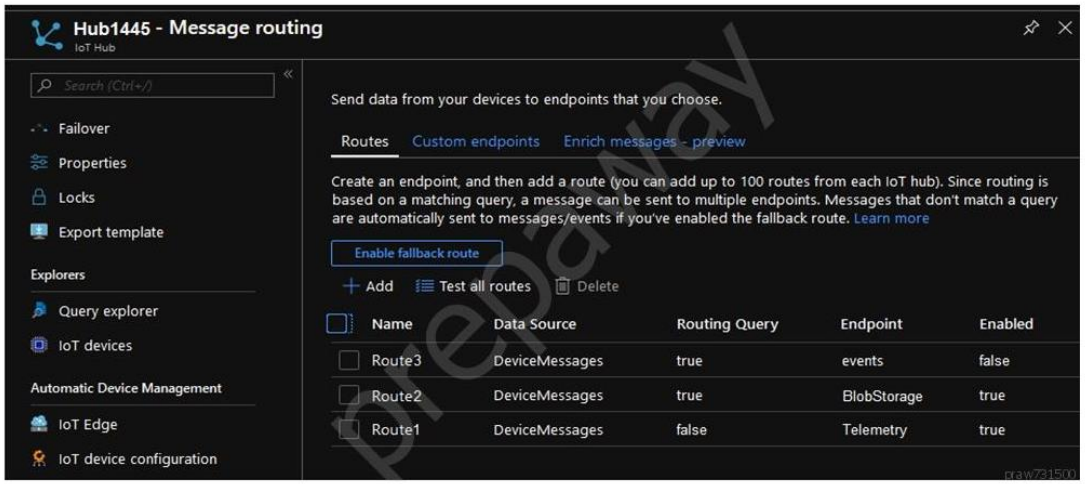
Answer: D
Your Answer: No answer

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

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.



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.

EXPLANATION

Answer: B
Your Answer: No answer

Explanation:
Events are not getting through as Route3, with an events Endpoint, is not enabled.

Note: Azure IoT Hub is a highly scalable publish-subscribe event ingestor optimized for IoT scenarios.

The default timestamp of events coming from an IoT Hub in Stream Analytics is the timestamp that the event arrived in the IoT Hub, which is EventEnqueuedUtcTime.

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

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

EXPLANATION

Answer: B
Your Answer: No answer

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

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

You open **Azure Monitor** as shown in the exhibit.

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.

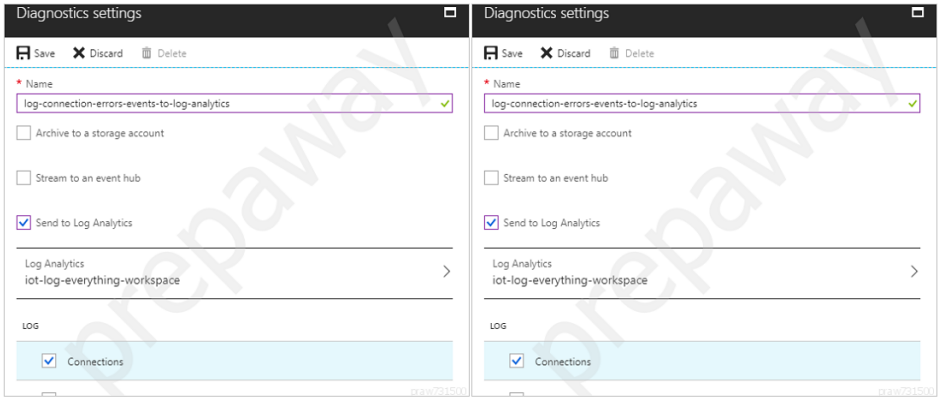
What should you do first?

- 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.

EXPLANATION

Answer: D
Your Answer: No answer

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



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

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

EXPLANATION

Answer: B
Your Answer: No answer

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/

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

EXPLANATION

Answer: B
Your Answer: No answer

Explanation:
Device twin queries: IoT Hub exposes the device twins as a document collection called devices. For example, the following query retrieves the whole set of device twins:
SELECT * FROM devices

Reference:
https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-devguide-query-language#device-and-module-twin-queries

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 = MessageAlerts
  ```
- [ ] 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 = MessageAlerts
  ```
- [ ] E. Create an IoT Hub routing rule that has a data source of Device Telemetry Messages and select the endpoint for MessageAlerts.

EXPLANATION

Answer: CD
Your Answer: No answer

Explanation:
C: Device twins change notifications allow back end applications to be notified in real-time of twin changes and react as business logic requires. For instance, when building an application that notifies an operator directly that a device has a firmware update pending and is "ready to update". This is accomplished by creating a routing rule from the "Twin Change Events" source and filtering all events that involve the software update status.

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

https://azure.microsoft.com/en-us/blog/enhancements-to-azure-iot-hub-improve-management-of-the-full-device-lifecycle/

DRAG DROP

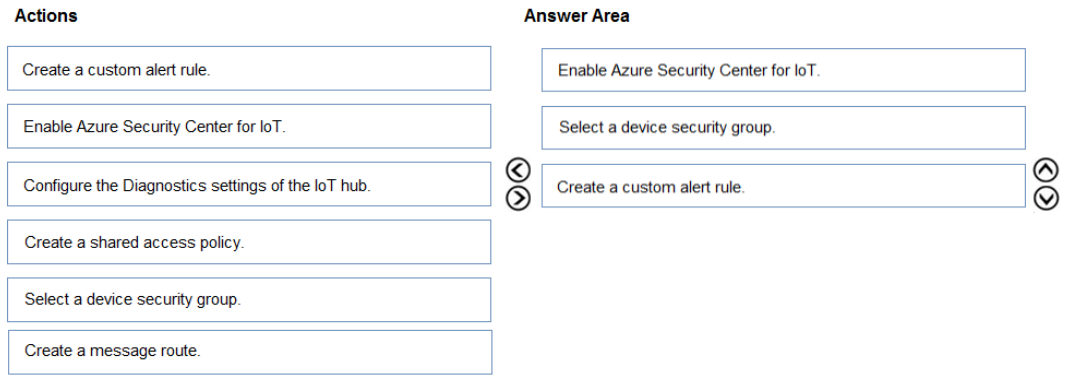
You have 100 devices that connect to an Azure IoT hub.

You need to be notified about failed local logins to a subset 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.

EXPLANATION

Answer:



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

You have an Azure IoT solution that includes a basic tier Azure IoT hub named Hub1 and a Raspberry Pi device named Device1. Device1 connects to Hub1.

You back up Device1 and restore the backup to a new Raspberry Pi device.

When you start the new Raspberry Pi device, you receive the following error message in the diagnostic logs of `Hub1: "409002 LinkCreationConflict."`

You need to ensure that Device1 and the new Raspberry Pi device can run simultaneously without error.

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

NOTE: Each correct selection is worth one point.

- [ ] A. On the new Raspberry Pi device, modify the connection string.
- [ ] B. From Hub1, modify the device shared access policy.
- [ ] C. Upgrade Hub1 to the standard tier.
- [ ] D. From Hub1, create a new consumer group.
- [ ] E. From Hub1, create a new IoT device.

EXPLANATION

Answer: AE
Your Answer: No answer

Explanation:
Note: Symptoms
You see the error 409002 LinkCreationConflict in logs along with device disconnection or cloud-to-device message failure.

Cause
Generally, this error happens when IoT Hub detects a client has more than one connection. In fact, when a new connection request arrives for a device with an existing connection, IoT Hub closes the existing connection with this error.

Reference:
https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-troubleshoot-error-409002-linkcreationconflict#symptoms

https://devblogs.microsoft.com/iotdev/understand-different-connection-strings-in-azure-iot-hub/

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

You discover that some of the devices fail to send data to the IoT hub.

You need to ensure that you can use Azure Monitor to troubleshoot the device connectivity issues.

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

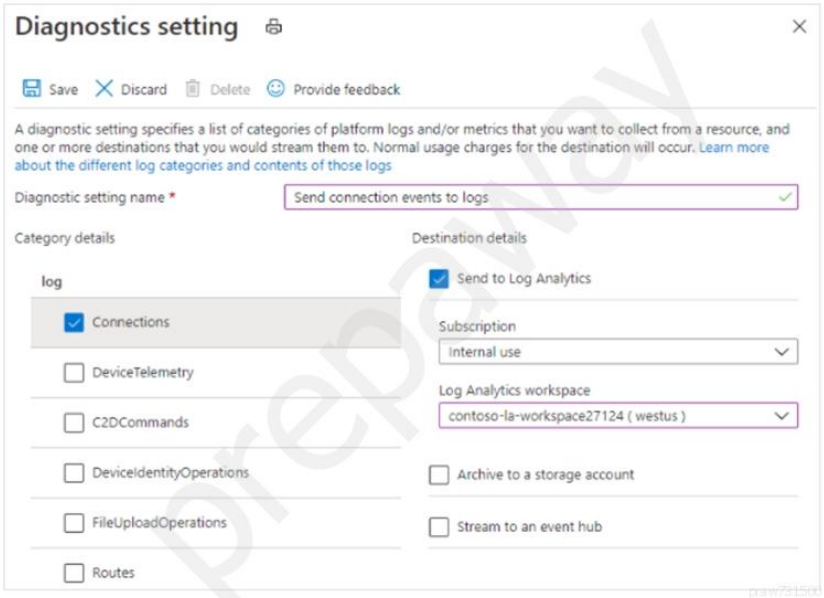
NOTE: Each correct selection is worth one point.

- A. From the Diagnostics settings of the IoT hub, select **Archive to a storage account.**
- B. Collect the DeviceTelemetry, Connections, and Routes logs.
- C. Collect all metrics.
- D. From the Diagnostics settings of the IoT hub, select **Send to Log Analytic.**
- E. Collect the JobsOperations, DeviceStreams, and FileUploadOperations logs.

EXPLANATION

Answer: BD
Your Answer: No answer

Explanation:
The IoT Hub resource logs connections category emits operations and errors having to do with device connections. The following screenshot shows a diagnostic setting to route these logs to a Log Analytics workspace:



Note: Azure Monitor: Route connection events to logs:
IoT hub continuously emits resource logs for several categories of operations. To collect this log data, though, you need to create a diagnostic setting to route it to a destination where it can be analyzed or archived. One such destination is Azure Monitor Logs via a Log Analytics workspace, where you can analyze the data using Kusto queries.

Reference:
https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-troubleshoot-connectivity

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

You plan to deploy 10,000 IoT devices.

You need to validate the performance of the IoT solution while 10,000 concurrently connected devices stream telemetry. The solution must minimize effort.

What should you deploy?

- A. an Azure IoT Device Simulation from Azure IoT Solution Accelerator
- B. an Azure function, an IoT Hub device SDK, and a timer trigger
- C. Azure IoT Central application and a template for the retail industry
- D. an Azure IoT Edge gateway configured as a protocol translation gateway

EXPLANATION

Answer: A
Your Answer: No answer

Explanation:
The IoT solution accelerators are complete, ready-to-deploy IoT solutions that implement common IoT scenarios. The scenarios include connected factory and device simulation.

Use the Device Simulation solution accelerator to run simulated devices that generate realistic telemetry. You can use this solution accelerator to test the behavior of the other solution accelerators or to test your own custom IoT solutions.

Reference:
https://docs.microsoft.com/en-us/azure/iot-accelerators/about-iot-accelerators

You have an Azure IoT Central application that monitors 100 IoT devices.

You need to generate alerts when the temperature of a device exceeds 100 degrees. The solution must meet the following requirements:

- Minimize costs
- Minimize deployment time

What should you do?

A. Perform a data export to Azure Service Bus.

B. Create an email property in the device templates.

C. Perform a data export to Azure Blob storage and create an Azure function.

D. Create a rule that uses an email action.

EXPLANATION

Answer: D
Your Answer: No answer

Explanation:
You can create rules in IoT Central that trigger actions, such as sending an email, in response to telemetry-based conditions, such as device temperature exceeding a threshold.

Reference:
https://docs.microsoft.com/en-us/azure/iot-central/core/howto-configure-rules-advanced

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 section, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.

You have 20 IoT devices deployed across two floors of a building. The devices on the first floor must be set to 60 degrees. The devices on the second floor must be set to 80 degrees.

The device twins are configured to use a tag that identifies the floor on which the twins are located.

You create the following automatic configuration for the devices on the first floor.

```
{
    "id": "first_floor_devices",
    "schemaVersion": null,
    "labels": {
     "Version": "1"
    },
    "content": {
     "deviceContent": {
         "properties.desired.ac": {
          "temperature": 60
         }
       }
    },
    "targetCondition": "tags.floor-'first'",
    "createdTimeUtc": "2020-12-08T04:06:56.651Z",
    "lastUpdatedTimeUtc":  "2020-12-08T04:06:56.651Z",
    "priority": 1,
...
}
```

You create the following automatic configuration for the devices on the second floor.

```
{
    "id": "second_floor_devices",
    "schemaVersion": null,
    "labels": {
     "Version": "1"
    },
    "content": {
     "deviceContent": {
         "properties.desired.ac": {
          "temperature": 80
         }
       }
    },
    "targetCondition": "*",
    "createdTimeUtc": "2020-12-08T04:11:08.561Z",
    "lastUpdatedTimeUtc":  "2020-12-09T18:50:55.070Z",
    "priority": 10,
...
}
```

The IoT devices on the first floor report that the temperature is set to 80 degrees.

You need to ensure that the first-floor devices are set to the correct temperature.

Solution: In the automatic configuration for the second-floor devices, you set `targetCondition` to `"tags.floor='second'"`.

Does this meet the goal?

- A. Yes
- B. No

EXPLANATION

Answer: A
Your Answer: No answer

Reference:
https://docs.microsoft.com/en-us/azure/iot-edge/module-deployment-monitoring?view=iotedge-2020-11

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

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 section, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.

You have 20 IoT devices deployed across two floors of a building. The devices on the first floor must be set to 60 degrees. The devices on the second floor must be set to 80 degrees.

The device twins are configured to use a tag that identifies the floor on which the twins are located.

You create the following automatic configuration for the devices on the first floor.

```
{
    "id": "first_floor_devices",
    "schemaVersion": null,
    "labels": {
     "Version": "1"
    },
    "content": {
     "deviceContent": {
         "properties.desired.ac": {
          "temperature": 60
         }
       }
    },
    "targetCondition": "tags.floor-'first'",
    "createdTimeUtc": "2020-12-08T04:06:56.651Z",
    "lastUpdatedTimeUtc":  "2020-12-08T04:06:56.651Z",
    "priority": 1,
...
}
```

You create the following automatic configuration for the devices on the second floor.

```
{
    "id": "second_floor_devices",
    "schemaVersion": null,
    "labels": {
     "Version": "1"
    },
    "content": {
     "deviceContent": {
         "properties.desired.ac": {
          "temperature": 80
         }
       }
    },
    "targetCondition": "*",
    "createdTimeUtc": "2020-12-08T04:11:08.561Z",
    "lastUpdatedTimeUtc":  "2020-12-09T18:50:55.070Z",
    "priority": 10,
...
}
```

The IoT devices on the first floor report that the temperature is set to 80 degrees.

You need to ensure that the first-floor devices are set to the correct temperature.

Solution: In the automatic configuration for the second-floor devices, you set `Version` to `2`.

Does this meet the goal?

- A. Yes
- B. No

EXPLANATION

Answer: B
Your Answer: No answer

Reference:
https://docs.microsoft.com/en-us/azure/iot-edge/module-deployment-monitoring?view=iotedge-2020-11

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

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.

EXPLANATION

Answer: D
Your Answer: No answer

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

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 devices 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

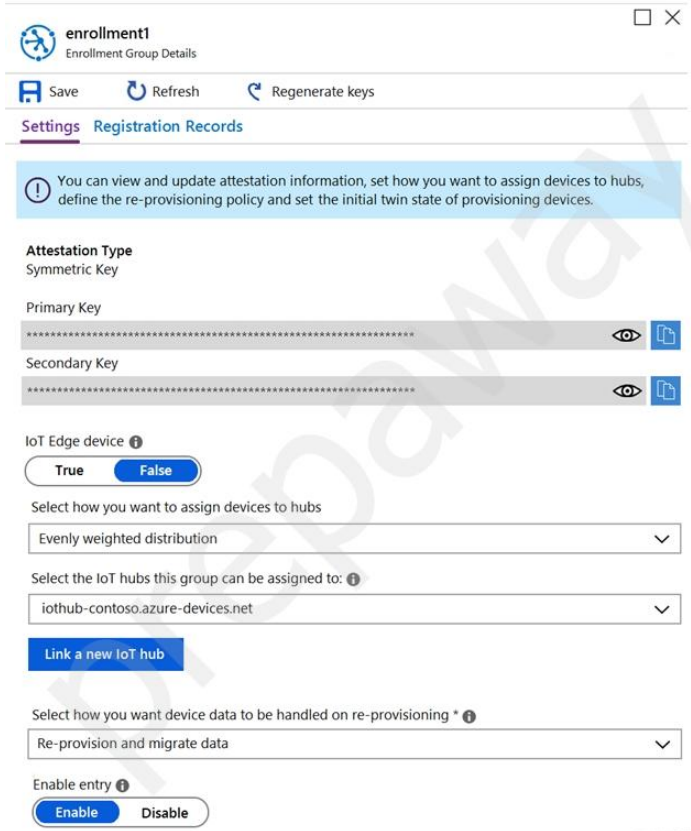
EXPLANATION

Answer: B
Your Answer: No answer

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

From the Device Provisioning Service, you create an enrollment as shown in the exhibit.



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 HMAC-SHA256 hash of the device's registration ID
- D. the endorsement key of the device's Trusted Platform Module (TPM)

EXPLANATION

Answer: C
Your Answer: No answer

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

DRAG DROP

You have an Azure IoT Edge solution.

You plan to deploy an Azure Security Center for IoT security agent. You need to configure the security agent to meet the following requirements:

- Connection events must be reported as high priority.
- High priority events must be collected every seven minutes.

How should you configure the `azureiotsecurity` module twin? To answer, drag the appropriate values to the correct locations. Each value 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.

EXPLANATION

Answer:

```
"reported": {

"lowPriorityMessageFrequency": {

"eventPriorityProcessCreate": {
"aggregationIntervalConnectionCreate": {
```

Answer Area

```
"desired": {
  "ms_iotn:urn_azureiot_Security_SecurityAgentConfiguration": {
    "highPriorityMessageFrequency": {
      "value": "PT7M"
    },
    "eventPriorityConnectionCreate": {
      "value": "High"
    }
  }
}
```

Explanation:

Box 1: "desired": {

To configure connection events as high priority and collect high priority events every 7 minutes, use the following configuration.

```
"desired": {
   "ms_iotn:urn_azureiot_Security_SecurityAgentConfiguration": {
      "highPriorityMessageFrequency": {
         "value": "PT7M"
      },
      "eventPriorityConnectionCreate": {
         "value": "High"
      }
```

Box 2: "highPriorityMessageFrequency ": {

Box 3: "eventPriorityConnectionCreate": {

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

You have an Azure IoT hub that has a hostname of contoso-hub.azure-devices.net and an MCU-based IoT device named Device1. Device1 does **NOT** support Azure IoT SDKs.

You plan to connect Device1 to the IoT hub by using the Message Queuing Telemetry Transport (MQTT) protocol and to authenticate by using X.509 certificates.

You need to ensure that Device1 can authenticate to the IoT hub.

What should you do?

- A. Create an Azure key vault and enable the encryption of data at rest for the IoT hub by using a customer-managed key.
- B. Enable a hardware security module (HSM) on Device1.
- C. From the Azure portal, create an IoT Hub Device Provisioning Service (DPS) instance and add a certificate enrollment for Device1.
- D. Add the DigiCert Baltimore Root Certificate to Device1.

EXPLANATION

Answer: D
Your Answer: No answer

Explanation:
The connection to Azure IoT Hub with MQTT is secured using TLS.
The Azure IoT Hub library requires the provisioning of the following certificates and a private key for a successful TLS connection:
1. Baltimore CyberTrust Root certificate - Server certificate, used to verify the server's certificate while connecting.
2. Device certificate - generated by the procedures described in Creating Azure IoT Hub certificates , used by Azure IoT Hub to authenticate the device.
3. Private key of the device.

Reference:
https://developer.nordicsemi.com/nRF_Connect_SDK/doc/latest/nrf/include/net/azure_iot_hub.html

HOTSPOT

You are planning a proof of concept (POC) that will use an Azure IoT hub.

You have two self-signed client authentication certificates named Cert1 and Cert2. Cert1 has a basic constraint that contains `Subject Type=CA`. Cert2 has a basic constraint that contains `Subject Type=End Entity`.

You need to identify which certificates to use.

What should you identify? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

EXPLANATION

Answer:

Answer Area

Certificate you can use to authenticate a leaf device to IoT Hub during testing:

- Cert1 only
- **Cert2 only**
- Both Cert1 and Cert2
- Neither certificate

Certificate that you can upload to IoT Hub as a verified certificate:

- Cert1 only
- **Cert2 only**
- Both Cert1 and Cert2
- Neither certificate

Explanation:

Box 1: Cert2 only
Cert2: The leaf certificate, or end-entity certificate, identifies the certificate holder. It has the root certificate in its certificate chain as well as zero or more intermediate certificates. The leaf certificate is not used to sign any other certificates. It uniquely identifies the device to the provisioning service and is sometimes referred to as the device certificate.

Box 2: Cert1 only
Cert1: A root certificate is a self-signed X.509 certificate representing a certificate authority (CA). It is the terminus, or trust anchor, of the certificate chain. Root certificates can be self-issued by an organization or purchased from a root certificate authority.

Reference:
https://docs.microsoft.com/en-us/azure/iot-dps/concepts-x509-attestation

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

EXPLANATION

Answer: C
Your Answer: No answer

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

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.

EXPLANATION

Answer: A
Your Answer: No answer

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

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

EXPLANATION

Answer: ADF
Your Answer: No answer

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-access

https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-devguide-identity-registry

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).

EXPLANATION

Answer: D
Your Answer: No answer

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

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.

EXPLANATION

Answer:

Actions	Answer Area
Generate a verification code.	Upload the root CA certificate to the IoT hub.
Upload the verification certificate.	Generate a verification code.
Upload the root CA certificate to the IoT hub.	Generate a verification certificate.
Copy the thumbprint from root CA certificate.	Upload the verification certificate.
Generate a verification certificate.	

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