Question #1 *Topic 1*

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

After you answer a question in this 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

Suggested Answer: *A*

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

ipindado2020 Highly Voted 4 years, 7 months ago

A is ok.

upvoted 7 times

Mers17 Most Recent 3 years, 11 months ago

A is okay

upvoted 4 times

tedsi 4 years, 7 months ago

Key word here is 'temporarily' so disabling is right strategy.

upvoted 4 times

Question #2 *Topic 1*

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

After you answer a question in this 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

Suggested Answer: *B*

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

ipindado2020 Highly Voted 👍 4 years, 7 months ago

B is ok

upvoted 8 times

Mers17 Most Recent 4 years, 11 months ago

B - No

upvoted 4 times

Question #3 *Topic 1*

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

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

ipindado2020 Highly Voted 4 years, 7 months ago
B is ok
upvoted 7 times

Mers17 Most Recent 3 years, 11 months ago
B- No is correct answer
upvoted 4 times

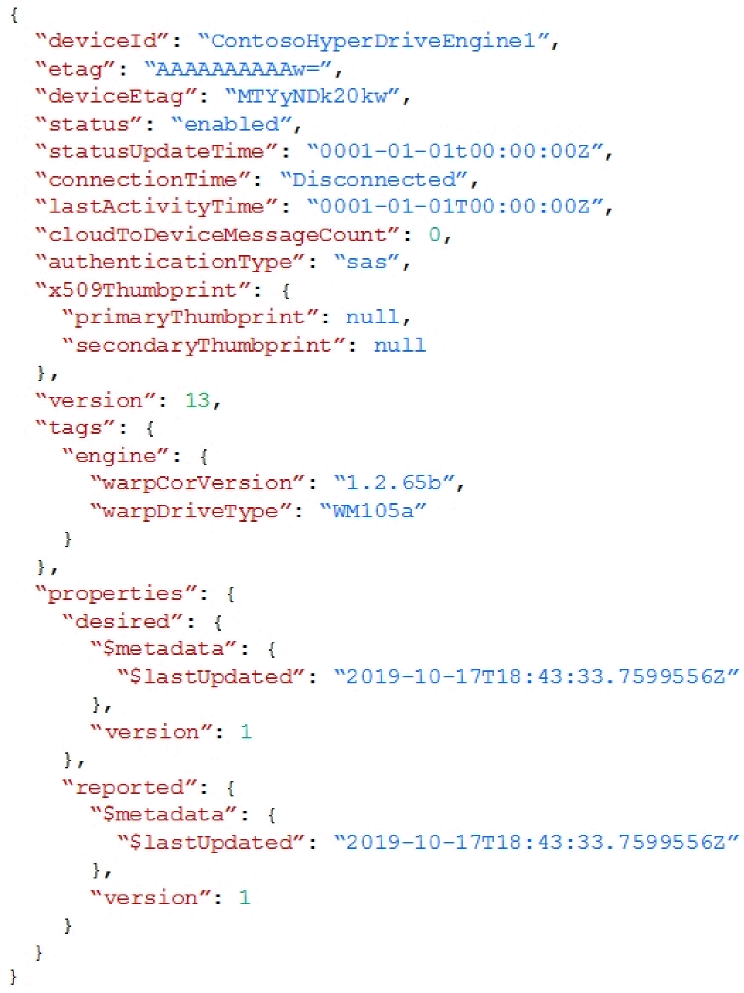
exam67 4 years ago
changing credentials may be effective in preventing the device to connect to IoT hub, but in order to be a "temporary" effect, you should then be prepared to restore the credentials later, therefore make a backup copy of it
upvoted 4 times

HOTSPOT -

You have an Azure IoT hub.

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

The device twin is shown below.



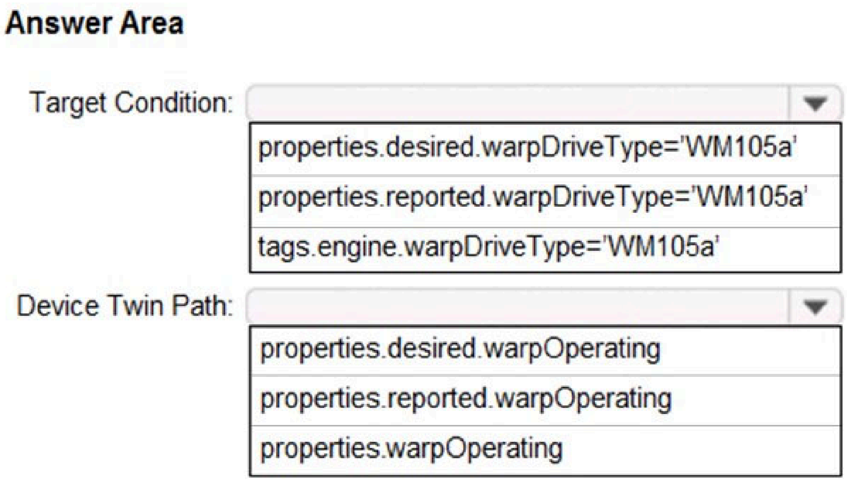
```
{
  “deviceId”: “ContosoHyperDriveEngine1”,
  “etag”: “AAAAAAAAAAw=”,
  “deviceEtag”: “MTYyNDk20kw”,
  “status”: “enabled”,
  “statusUpdateTime”: “0001-01-01t00:00:00Z”,
  “connectionTime”: “Disconnected”,
  “lastActivityTime”: “0001-01-01T00:00:00Z”,
  “cloudToDeviceMessageCount”: 0,
  “authenticationType”: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

You need to configure automatic device management for the deployment.

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

NOTE: Each correct selection is worth one point.

Hot Area:



Answer Area

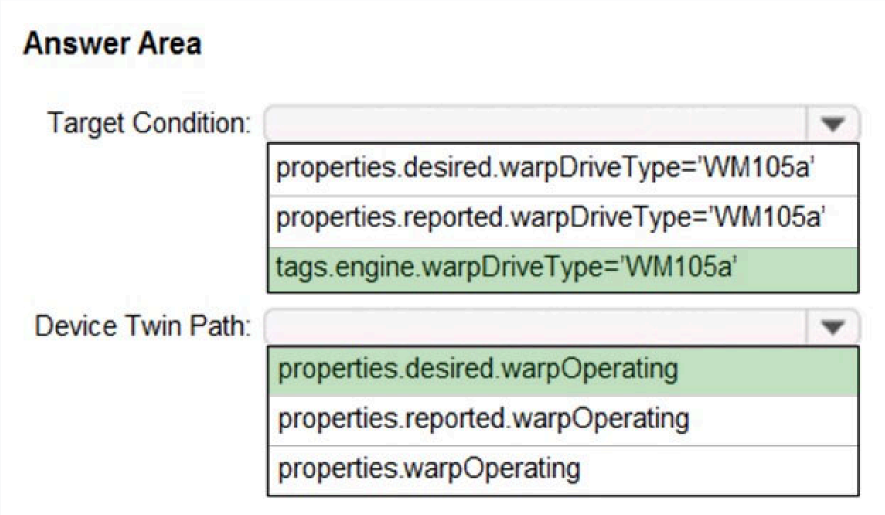
Target Condition: ▼

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

Device Twin Path: ▼

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

Suggested 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

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

⊟ **getazusername** Highly Voted 4 years, 7 months ago

1. Use tags to target twins
2. twin path e.g to properties.desired.chiller-water

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

upvoted 9 times

⊟ **liberty123** Most Recent 3 years, 3 months ago

Answer is correct

upvoted 3 times

⊟ **exam67** 4 years ago

an automatic device management configuration has *target condition*, *target content* and optionally *target metrics*. I assume 'device twins path' is used to indicate "target content". Answer is correct then.

upvoted 2 times

Question #5 *Topic 1*

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

Suggested Answer: *D*
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/

Community vote distribution

C (70%) D (30%)

⊟ **LiamRT** Highly Voted 4 years, 7 months ago
Schedule jobs on multiple devices. Azure IoT Hub enables a number of building blocks like device twin properties and tags and direct methods. Typically, back-end apps enable device administrators and operators to update and interact with IoT devices in bulk and at a scheduled time. Jobs execute device twin updates and direct methods against a set of devices at a scheduled time.
https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-devguide-jobs
Looks like D may be the correct answer.
upvoted 8 times

⊟ **angelsrp** Highly Voted 4 years, 11 months ago
Ans is B.

We use device twins JSON document to make changes in the device with the reporterd/desired properties.

https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-devguide-device-twins
upvoted 7 times

⊟ **not_a_robot** 4 years, 11 months ago
The answer is D, you don't set the report prop. Report prop is what's sent back to IoT Hub from the device.
upvoted 13 times

⊟ **satishk4u** 3 years, 4 months ago
Correct. Report Prop is from Device to IOT Hub.
upvoted 1 times

⊟ **MasDen** 4 years, 6 months ago
It is possible to use jobs to update devices using desired properties. So, answer should be B
https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-devguide-jobs#jobs-to-update-device-twin-properties
upvoted 4 times

⊟ **feyey** Most Recent 2 years, 2 months ago
Selected Answer: C
While it is possible to use a direct method to trigger an OTA update on devices that connect to an Azure IoT hub, it is not the recommended method.

Direct methods are typically used for invoking a specific action on a device in real-time, and they can be useful for managing device settings or executing a specific function on a device. However, when it comes to performing OTA updates, using cloud-to-device messages is usually the more appropriate method.

Cloud-to-device messages can be used to send firmware updates or other configuration changes to devices, and they can be scheduled to occur at specific times or intervals using Azure IoT Hub's Jobs feature.

Therefore, the recommended method for performing an OTA update on devices that connect to an Azure IoT hub by using scheduled jobs is to use a cloud-to-device message. Option C is the correct answer.

upvoted 4 times

KrishnaSK1 2 years, 5 months ago

You can use a direct method to initiate device management actions (such as reboot, factory reset, and firmware update) from a back-end app in the cloud. The device is responsible for:

Handling the method request sent from IoT Hub.
Initiating the corresponding device-specific action on the device.
Providing status updates through reported properties to IoT Hub.
https://learn.microsoft.com/en-us/training/modules/examine-device-management-concepts-methods/3-device-management-patterns

Answer: D

upvoted 1 times

liberty123 3 years, 3 months ago

Selected Answer: D

Agree with D

upvoted 1 times

imtiazL 3 years, 4 months ago

Answer is D
https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-node-node-schedule-jobs

upvoted 2 times

d0bermannn 3 years, 5 months ago

Selected Answer: D

D, see link
https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-devguide-c2d-guidance
said that targets for Cloud-to-device messages is 'Single device by deviceId.'
so no way for C

upvoted 2 times

metron 3 years, 7 months ago

Selected Answer: C

The key item in the question is 'will connect'. Which means D is likely wrong, since it works directly on connected devices. On the other hand, C is right (many devices will connect, and decide when to read the messages...since the messages can be queued in each device's queue, and it won't matter if some are connected or not.

upvoted 3 times

Robert12345Robert 2 years, 8 months ago

Jobs only support:
-Update desired properties
-Update tags
-Invoke direct methods
So C can not be the answer.

upvoted 2 times

exam67 4 years ago

my vote is for "C. a cloud-to-device message". There is a key statement in the question about the devices being usually disconnected and connecting at scheduled time. In that case a direct-method will likely fail. A cloud to device message instead is queued and will be picked up by the device as soon as it connects to IoT Hub.

https://docs.microsoft.com/en-us/azure/architecture/example-scenario/iot/cloud-to-device

upvoted 1 times

d0bermannn 3 years, 5 months ago

link
https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-devguide-c2d-guidance
said that targets for Cloud-to-device messages is 'Single device by deviceId.'
so D
upvoted 2 times

SanjuB 4 years, 4 months ago
D is the correct answer. No confusion.
upvoted 5 times

MasDen 4 years, 6 months ago
The answer is B.
"IoT Hub will use the device twin properties to transfer the configuration change request to the device and monitor the progress"
https://docs.microsoft.com/en-us/learn/modules/automate-iot-devices-management-with-azure-iot-hub/3-firmware-update-mechanism

There are not many differences between updating devices manually or through jobs.
upvoted 2 times

MasDen 4 years, 6 months ago
My fault. Answer is D, due to we can use desired properties not reported. Because we don't have the option to use desired properties we can only one possible option: direct method
upvoted 6 times

d0bermannn 3 years, 6 months ago
best comment here
upvoted 1 times

JanFJ 3 years, 5 months ago
why we don't have the option to use desired properties ?
upvoted 1 times

Question #6 *Topic 1*

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.

Suggested Answer: *AC*

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

angelsrp Highly Voted 👍 4 years, 11 months ago

Ans AC:

https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-devguide-file-upload

upvoted 10 times

BoomJosh Highly Voted 👍 4 years, 3 months ago

Appeared for exam on 3/24/2021 and successfully cleared it, this question was there.

upvoted 5 times

satishk4u Most Recent 🕒 3 years ago

AC is correct.

upvoted 2 times

Question #7 *Topic 1*

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

Suggested Answer: *D*
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

⊟ **tedsi** Highly Voted 👍 4 years, 7 months ago
Easy way to think of this: can't be Basic tiers due to Edge requirement and based on data flow has to be S3
upvoted 6 times

⊟ **pdeng** 2 years, 6 months ago
86.4m msg/day = 2.5k devices * [6kb / 4kb/unit] * (24h * 60min * (60sec / 5sec))
S2: (6 million messages/day per unit)
S3: (300 million messages/day per unit)
upvoted 1 times

⊟ **BillBaits** Most Recent ⏱ 3 years, 8 months ago
No answer is correct, since the maximum supported message size is 4kb
https://azure.microsoft.com/en-us/pricing/details/iot-hub/
upvoted 2 times

⊟ **longnt** 3 years ago
Kamilelo is corrected. 6KB message will be count as two per "message meter size" limit.
upvoted 1 times

⊟ **Kamilelo** 3 years, 6 months ago
D is correct. In link above you have "Message meter size" not "Maximum message size"
upvoted 6 times

⊟ **ipindado2020** 4 years, 7 months ago
D is ok
upvoted 4 times

Question #8 *Topic 1*

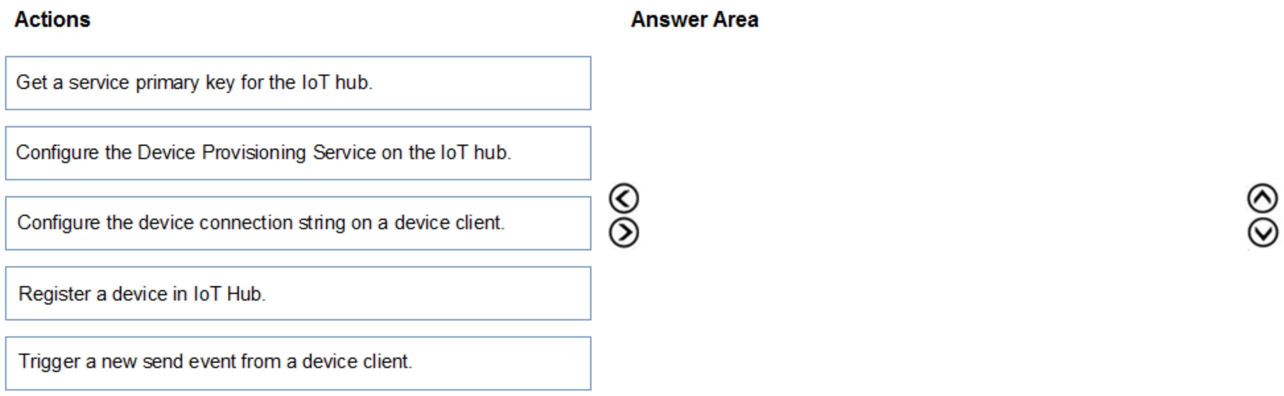
DRAG DROP -

You deploy an Azure IoT hub.

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

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

Select and Place:

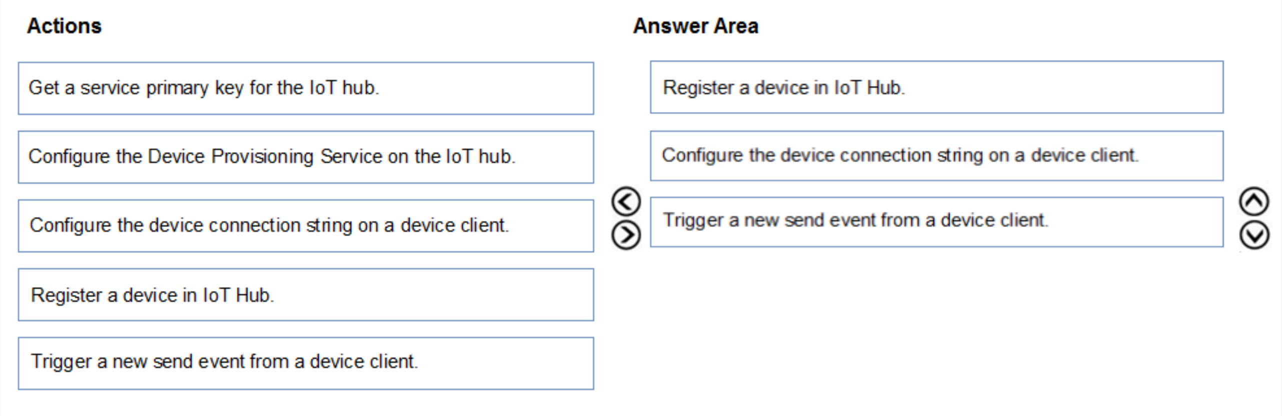


Actions

- Get a service primary key for the IoT hub.
- Configure the Device Provisioning Service on the IoT hub.
- Configure the device connection string on a device client.
- Register a device in IoT Hub.
- Trigger a new send event from a device client.

Answer Area

Suggested Answer:



Actions

- Get a service primary key for the IoT hub.
- Configure the Device Provisioning Service on the IoT hub.
- Configure the device connection string on a device client.
- Register a device in IoT Hub.
- Trigger a new send event from a device client.

Answer Area

- Register a device in IoT Hub.
- Configure the device connection string on a device client.
- Trigger a new send event from a device client.

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

angelsrp Highly Voted 4 years, 11 months ago

Given answers are correct, but i think this link will help a little bit more:

https://docs.microsoft.com/en-us/learn/modules/manage-iot-devices/3-connect-device

upvoted 14 times

Sudhansu21 Most Recent 3 years, 4 months ago

Came in Exam Feb 2022.

upvoted 4 times

Sudhansu21 3 years, 4 months ago

Correct.

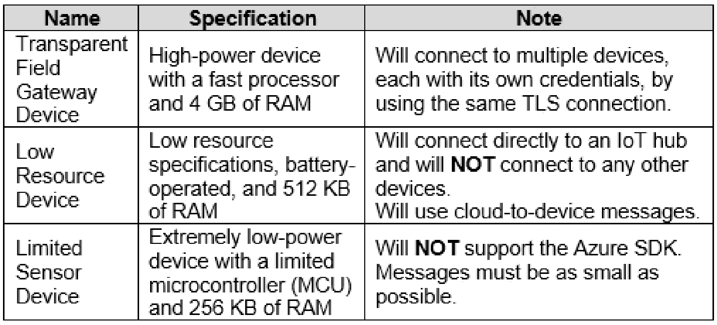
upvoted 2 times

Question #9 *Topic 1*

DRAG DROP -

You have an Azure IoT hub.

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



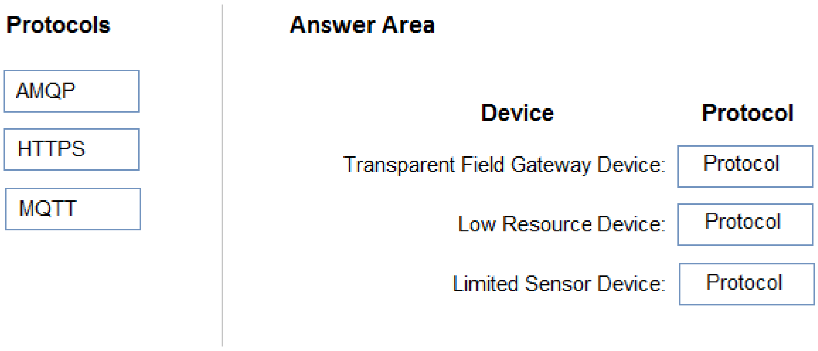
Name	Specification	Note
Transparent Field Gateway Device	High-power device with a fast processor and 4 GB of RAM	Will connect to multiple devices, each with its own credentials, by using the same TLS connection.
Low Resource Device	Low resource specifications, battery-operated, and 512 KB of RAM	Will 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.

Select and Place:



Protocols

AMQP

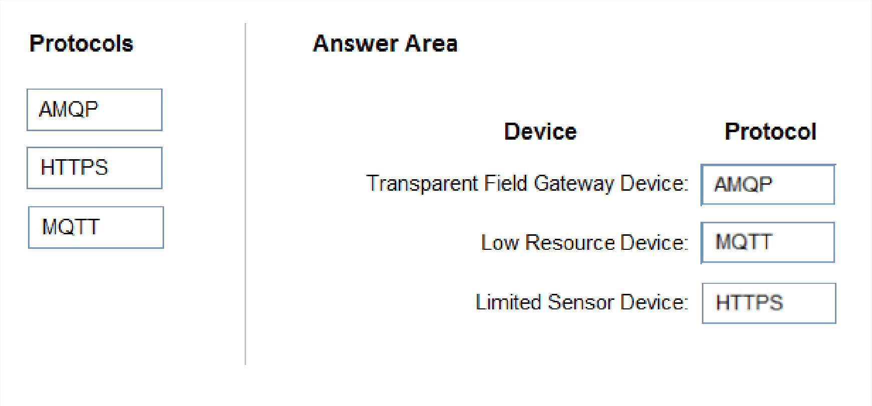
HTTPS

MQTT

Answer Area

Device	Protocol
Transparent Field Gateway Device:	Protocol
Low Resource Device:	Protocol
Limited Sensor Device:	Protocol

Suggested Answer:



Protocols

AMQP

HTTPS

MQTT

Answer Area

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

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

trickerk Highly Voted 3 years, 10 months ago

Given answers are correct according https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-devguide-protocols:

- Transparant Field Gateway Device: AMQP (will connect to multiple devices)
- Low Resource Device: MQTT or AMQP (will use cloud-to-device messages) -> MQTT (will not connect to another devices)
- Limited Sensor Device: HTTPS (will not support the Azure SDK Messages)

upvoted 5 times

KrishnaSK1 Most Recent 2 years, 5 months ago

answer: MQTT

https://blog.hansevision.com/connecting-sensors-to-azure-iot-hub

upvoted 1 times

pdeng 2 years, 6 months ago

I would vote for MQTT for the last box.

SSL/TLS on HTTPS is very heavy, MQTT is lighter.

MQTTS will be similar to HTTPS, but it is not listed in the options.

upvoted 3 times

pdeng 2 years, 6 months ago

Sorry, the MQTT here actually means MQTTS.

MQTT is smaller binary, HTTP is ascii.

upvoted 2 times

coramella 3 years, 5 months ago

Here you have this paragraph: https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-mqtt-support#example-in-c-using-mqtt-without-an-azure-iot-sdk on how implement mqtt in C when azure Sid is not supported. I would use mqtt for the last one.

upvoted 1 times

zartheus 4 years, 2 months ago

Limited sensor device should probably be MQTT instead of HTTPS as it requires minimum message size. Ref: "Payload size. MQTT and AMQP are binary protocols, which result in more compact payloads than HTTPS." https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-devguide-protocols

upvoted 2 times

BoomJosh 4 years, 3 months ago

Appeared for exam on 3/24/2021 and successfully cleared it, this question was there.

upvoted 3 times

shail_az900 4 years, 3 months ago

what was your answer

upvoted 2 times

BoomJosh 4 years, 3 months ago

Given answers are correct

upvoted 10 times

sballmer 4 years, 5 months ago

I would go for HTTPS, it is said "Extremely low power device". MQTT is a low-ressource protocol indeed but creates an active and constant connection which might be incompatible with a "low-power device". HTTPS allow the device to wake up, send data and get back to sleep, which is better for a low-power device...

upvoted 3 times

krishnakomal 4 years, 10 months ago

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

upvoted 1 times

EyeeyeeyeeyeeyeeyeeyeeyeSPIDER 4 years, 11 months ago

The last quetsion should be MQTT:

Low resource devices. The MQTT and HTTPS libraries have a smaller footprint than the AMQP libraries. As such, if the device has limited resources (for example, less than 1-MB RAM), these protocols might be the only protocol implementation available.

Payload size. MQTT and AMQP are binary protocols, which result in more compact payloads than HTTPS.

upvoted 4 times

angelsrp 4 years, 11 months ago

HTTPS is fine.

https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-device-sdk-platform-support
upvoted 3 times

ipindado2020 4 years, 7 months ago
Both should be valid.... Hard question. The fact that there is no SDK does not mean that mqtt is not supported by the device.... Maybe means that http integration "can be" easier, but nothing more....
MS recommends using MQTT as a general rule if device supports it

https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-devguide-no-sdk

So I go for MQTT, as should have smaller payload when reusing the connection.
upvoted 1 times

LiamRT 4 years, 5 months ago
A device can use the MQTT protocol to connect to an IoT hub using any of the following options.
Libraries in the Azure IoT SDKs.
The MQTT protocol directly.
https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-mqtt-support
Would such a small device support the 2nd option? I doubt it.
upvoted 1 times

tedsi 4 years, 7 months ago
Is not MQTT because you'd need Azure SDK, so has to be HTTPS.
upvoted 4 times

takagiko 3 years, 6 months ago
MQTT is a common protocol and is not specific to Azure SDK. For low resource IoT devices running on batteries, MQTT should generally be used rather than HTTPS. https://medium.com/mqtt-buddy/mqtt-vs-http-which-one-is-the-best-for-iot-c868169b3105
upvoted 1 times

Question #10 *Topic 1*

You create an Azure IoT hub by running the following command. az iot hub create --resource-group MyResourceGroup --name MyIotHub --sku B1 --location westus --partition-count 4

What does MyIotHub support?

A. Device Provisioning Service

B. cloud-to-device messaging

C. Azure IoT Edge

D. device twins

Suggested Answer: *A*

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

getazusername Highly Voted 👍 4 years, 7 months ago

Answer is a https://azure.microsoft.com/de-de/pricing/details/iot-hub/

You can see in the table, DPS is included in basic tier.

upvoted 10 times

Question #11 *Topic 1*

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

Suggested Answer: *ACE*

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

getazusername Highly Voted 4 years, 7 months ago

IoT Protocol Ports:

MQTT 8883

MQTT over WebSockets 443

AMQP 5671

AMQP over WebSockets 443

HTTPS 443

upvoted 13 times

ipindado2020 Highly Voted 4 years, 7 months ago

ACE is ok

upvoted 8 times

Sudhansu21 Most Recent 3 years, 4 months ago

Came in Exam Feb 2022.

upvoted 3 times

BoomJosh 4 years, 3 months ago

Appeared for exam on 3/24/2021 and successfully cleared it, this question was there.

upvoted 7 times

Question #12 | *Topic 1*

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?

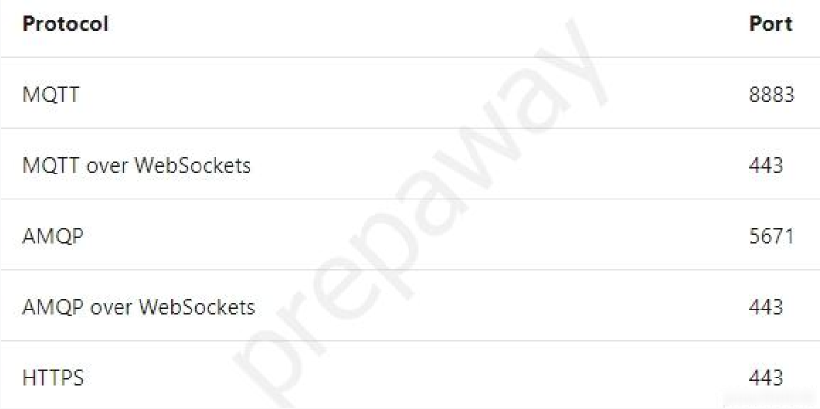
A. Configure the upstream protocol of the devices to use MQTT over TCP.

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.

Suggested Answer: *B*
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

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

⊟ **BoomJosh** Highly Voted 4 years, 3 months ago
Appeared for exam on 3/24/2021 and successfully cleared it, this question was there. Only difference was instead of MQTT, AMQP was in the option.
upvoted 5 times

⊟ **jracabrera** 4 years, 1 month ago
Thanks. Makes sense. Because AMQP is for cloud/field gateways.
upvoted 4 times

Question #13 *Topic 1*

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.

Suggested Answer: *B*
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

Community vote distribution
B (100%)

pdeng 2 years, 6 months ago
I would go for C.

The assumption is the connection string is hard-coded in device.
upvoted 2 times

Robert12345Robert 2 years, 9 months ago
tricky question, the endpoint is not changing by itself, you should do some programming on the device to support this. That is why I think answer C is the better and more accurate.
upvoted 3 times

hotwheelsinsf 2 years, 4 months ago
lots more working re-defining the endpoints....
upvoted 1 times

classicryder 3 years, 1 month ago
why not A?
upvoted 1 times

d0bermannn 3 years, 5 months ago
Selected Answer: B
nice question, that is the point where dev.twin must be used
upvoted 1 times

tita_tovenaar 4 years ago
should be B.
scenario is mentioned in following doc:
https://docs.microsoft.com/en-us/azure/iot-dps/how-to-reprovision
upvoted 2 times

d0bermannn 3 years, 6 months ago
your link have another option, but B still correct, as there are many ways to skin a cat)
upvoted 1 times

JeeBi 4 years, 1 month ago
When creating a device, you can provide the symmetric keys. So I would go for answer C: recreate the devices using the same keys and the change the endpoint.
upvoted 3 times

jracabrera 4 years, 1 month ago

Thanks JeeBi. I think you are right about recreate the device using the same keys. But, in real life, instead go to 10 devices and change the endpoint, you would use the the twin desired properties for your devices to consume the IoT Hub host name. In that way you can use Hub2. For me, the better is option B.

upvoted 4 times

⊟ **lukeross1504** 4 years, 3 months ago

is the answer correct?

upvoted 3 times

⊟ **jracabrera** 4 years, 1 month ago

For me, yes it is. Option B.

upvoted 4 times

⊟ **BoomJosh** 4 years, 3 months ago

Appeared for exam on 3/24/2021 and successfully cleared it, this question was there.

upvoted 2 times

⊟ **alexa123456789** 4 years, 2 months ago

what did you choose?

upvoted 8 times

Question #14 *Topic 1*

DRAG DROP -

You have an Azure subscription that contains an Azure IoT hub and 100 IoT devices.

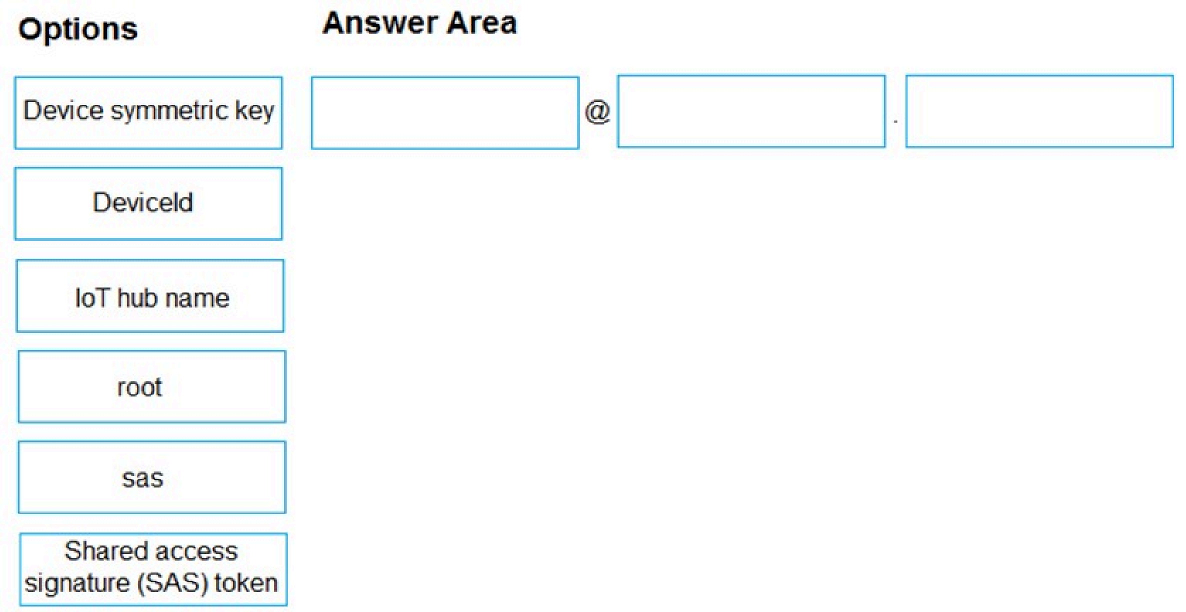
The devices connect to the IoT hub by using the Advanced Message Queuing Protocol (AMQP) protocol and authenticate to the IoT hub by using symmetric keys.

You need to configure the SASL PLAIN username for the AMQP connection.

How should you configure the username? To answer, drag the appropriate options to the correct targets. Each option may be used once, more than once, or not at all. You may need to drag the split bar between panes or scroll to view content.

NOTE: Each correct selection is worth one point.

Select and Place:



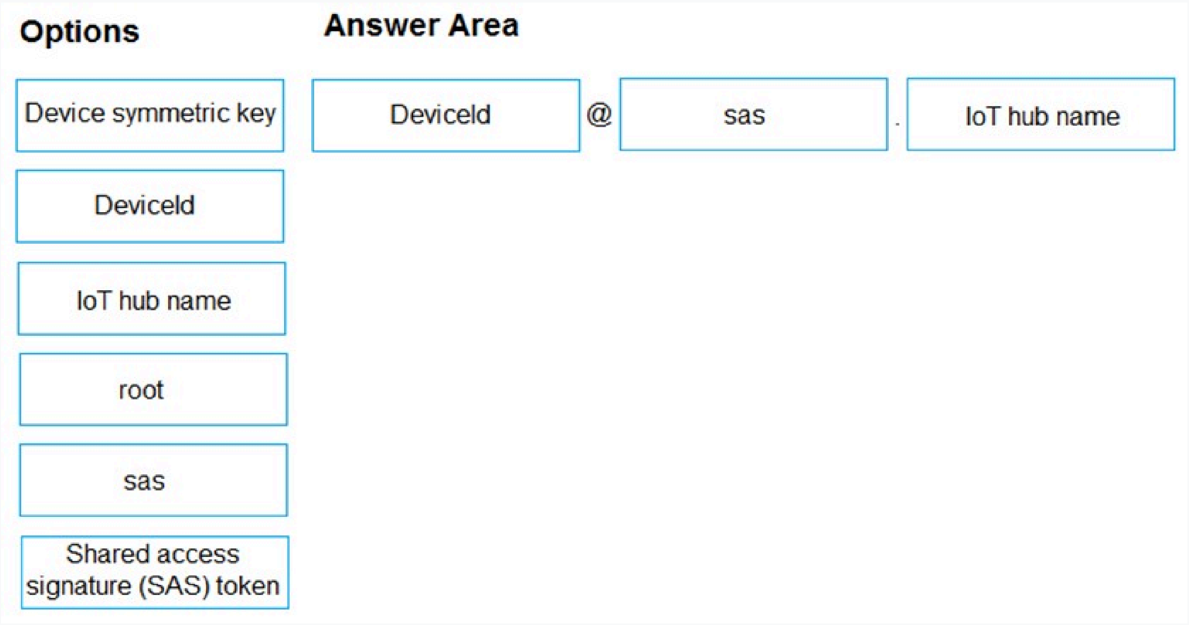
Options

- Device symmetric key
- DeviceId
- IoT hub name
- root
- sas
- Shared access signature (SAS) token

Answer Area

□ @ □ . □

Suggested Answer:



Options

- Device symmetric key
- DeviceId
- IoT hub name
- root
- sas
- Shared access signature (SAS) token

Answer Area

DeviceId @ sas . IoT hub name

Box 1: DeviceID -

If you use AMQP claims-based-security, the standard specifies how to transmit these tokens.

For SASL PLAIN, the username can be:

{policyName}@sas.root.{iothubName} if using IoT hub-level tokens.

{deviceId}@sas.{iothubname} if using device-scoped tokens.

Box 2: sas -

Box 3:IoT hub hame -

Reference:
https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-dev-guide-sas

Sudhansu21 Highly Voted 3 years, 4 months ago

Came in Exam Feb 2022.

upvoted 6 times

d0bermannn Highly Voted 3 years, 6 months ago

answer provided is correct as said in https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-dev-guide-sas?tabs=node, see here:

For SASL PLAIN, the username can be:

{policyName}@sas.root.{iothubName} if using IoT hub-level tokens.

{deviceId}@sas.{iothubname} if using device-scoped tokens.

upvoted 5 times

Question #15 *Topic 1*

You are configuring a production environment for an Azure IoT solution.
You plan to deploy 1,000 IoT devices. Each device will send one device-to-cloud message every hour. Each message will be 4 KB.
You need to deploy an Azure IoT hub that will support the IoT device deployment. The solution must meet the following requirements:
✑ Perform bulk device operations such as creating multiple device identities.
✑ Minimize costs
What should you deploy?

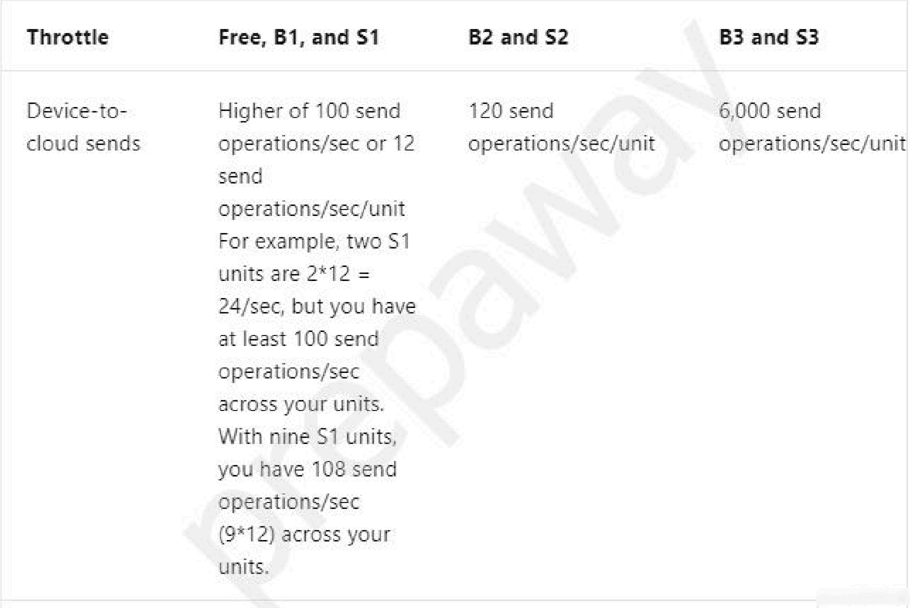
A. one unit of the B1 tier

B. one unit of the free tier

C. one unit of the S1 tier

D. one unit of the S2 tier

Suggested Answer: *B*
1000 /3600 device-to-cloud message/second is less than 1 per second. One unit of the free tier is sufficient.



Throttle	Free, B1, and S1	B2 and S2	B3 and S3
Device-to-cloud sends	Higher of 100 send operations/sec or 12 send operations/sec/unit For example, two S1 units are 2*12 = 24/sec, but you have at least 100 send operations/sec across your units. With nine S1 units, you have 108 send operations/sec (9*12) across your units.	120 send operations/sec/unit	6,000 send operations/sec/unit

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

Community vote distribution
A (100%)

⊟ **Kamilelo** Highly Voted 👍 3 years, 5 months ago
Selected Answer: A
It's 24000 4kb messages per day, Free tier can handle 8000 0,5kb messages per day so it's not enought. Answer should be A
upvoted 13 times

⊟ **d0bermannn** Highly Voted 👍 3 years, 6 months ago
Selected Answer: A
Do anybody really use free ter in prod.env? For me ans must be B1=A
upvoted 6 times

⊟ **feyey** Most Recent 🕒 2 years, 2 months ago
Selected Answer: A
The S1 tier provides the ability to perform bulk device operations and allows up to 400,000 messages per day. With 1,000 devices sending one message per hour, the total number of messages per day will be 24,000 (1000 devices * 24 hours). Each message is 4 KB, which translates to a total of 96 MB of data per day (24,000 messages * 4 KB). This is well within the message limit of the S1 tier (which allows up to 8,000,000 messages per month).

Additionally, the S1 tier is a cost-effective option, as it is designed for moderate-scale solutions and provides a balance between cost and features.

The free tier does not support bulk device operations, and the B1 and S2 tiers are designed for larger-scale solutions and are more expensive.

Therefore, deploying one unit of the S1 tier will meet the requirements of the Azure IoT solution while minimizing costs.

upvoted 1 times

PreetDP900 2 years, 3 months ago

So what is right answer???

I know i want to go for A as Free tier can't be used for prod environment.

upvoted 2 times

longnt 3 years ago

Selected Answer: A

Free tier message meter size is 0.5KB, it count 8 message for 4KB size. So in one hour 1000 device need 8000 messages, which use up all message/day of Freet tier, not enough.

upvoted 3 times

jh372 2 years, 4 months ago

FYI

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

upvoted 1 times

liberty123 3 years, 3 months ago

Selected Answer: A

Agree with B1

upvoted 4 times

coramella 3 years, 5 months ago

Selected Answer: A

Agree with b1. Documentation always says free is for dev and testing not prod.

upvoted 4 times

Question #16

Topic 1

You have an Azure IoT hub that receives messages from an IoT device. The messages are serialized as Protobuf.

You need the IoT hub to route the messages.

What should you do first?

A. From the Azure portal, add desired properties to the device twin.

B. Configure the IoT device to add application properties to the messages.

C. From the Azure portal, configure the IoT hub to add message enrichments.

D. Configure the IoT device to add ASCII-encoded properties to the body of the messages.

Suggested Answer: *A*

⊟ **Dilpreet23** 1 year, 11 months ago

Shouldn't this be B?

upvoted 1 times

Question #18 Topic 1

DRAG DROP

-

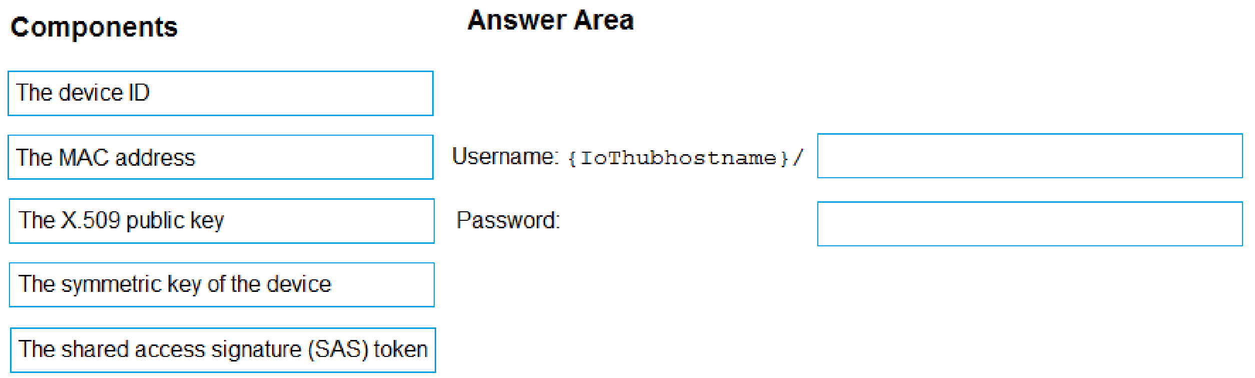
You have an Azure subscription that contains an Azure IoT hub and 100 IoT devices.

The devices connect to the IoT hub by using the Message Queuing Telemetry Transport (MQTT) protocol and authenticate to the IoT hub by using symmetric keys.

You need to configure the username and password for the MQTT connection.

What should you use? To answer, drag the appropriate components to the correct targets. Each component 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.



Components

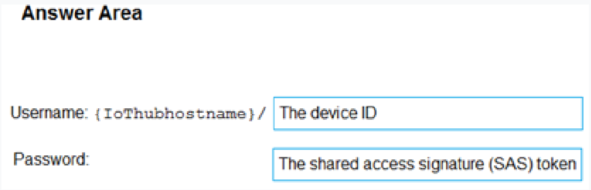
- The device ID
- The MAC address
- The X.509 public key
- The symmetric key of the device
- The shared access signature (SAS) token

Answer Area

Username: `{IoThubhostname}/` □

Password: □

Suggested Answer:



Answer Area

Username: `{IoThubhostname}/` The device ID

Password: The shared access signature (SAS) token

Tyrel 2 years ago

Correct.

https://learn.microsoft.com/en-us/azure/iot/iot-mqtt-connect-to-iot-hub

upvoted 1 times

Question #22 *Topic 1*

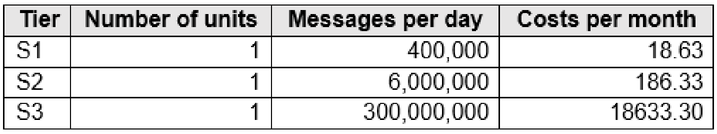
HOTSPOT

-

You have an Azure subscription that contains the following Azure IoT hub:

- Name: Hub1
- Tier: S1
- Number of units: 14

The subscription has the tiers and unit costs shown in the following table.

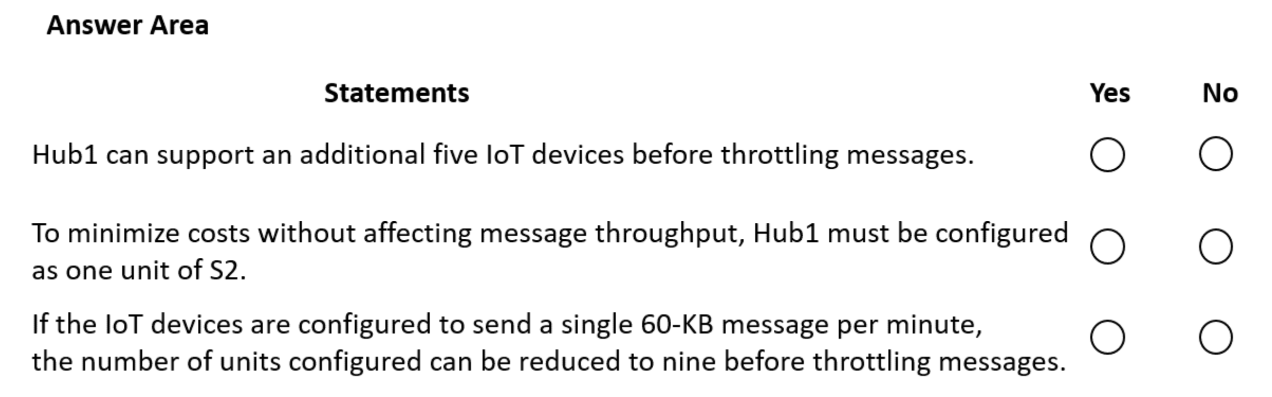


Tier	Number of units	Messages per day	Costs per month
S1	1	400,000	18.63
S2	1	6,000,000	186.33
S3	1	300,000,000	18633.30

You have 60 IoT devices that connect to Hub1. Each IoT device sends a single 1-KB message to Hub1 per second.

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

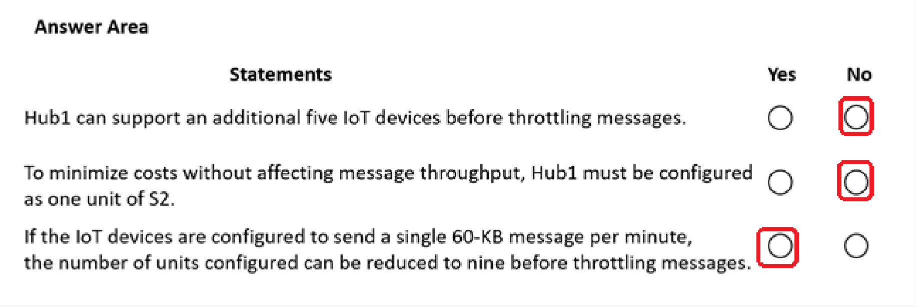
NOTE: Each correct selection is worth one point.



Answer Area

Statements	Yes	No
Hub1 can support an additional five IoT devices before throttling messages.	○	○
To minimize costs without affecting message throughput, Hub1 must be configured as one unit of S2.	○	○
If the IoT devices are configured to send a single 60-KB message per minute, the number of units configured can be reduced to nine before throttling messages.	○	○

Suggested Answer:



Answer Area

Statements	Yes	No
Hub1 can support an additional five IoT devices before throttling messages.	○	[✓ ○]
To minimize costs without affecting message throughput, Hub1 must be configured as one unit of S2.	○	[✓ ○]
If the IoT devices are configured to send a single 60-KB message per minute, the number of units configured can be reduced to nine before throttling messages.	[✓ ○]	○

kbronctjr 2 years ago

Is C correct? 60kb messages/min = 15 messages of 4 KB/min

15 * 60 devices * 60 min/h *24h/day = 1.296.000 msgs/day

1296000/400000 = 3,24. We could use as max 3 units before throtling

upvoted 2 times

kbronctjr 2 years ago

I think B is incorrrect:

We need to handle 60 * 3600 *24 = 5.184.000 msgs/day

5184000/400000 = 12.96 -> we need at least 13 S1 units

13 S1 units is more expensive than 1 S2 unit

upvoted 2 times

Billy223 2 years, 2 months ago

For the first point. It is NO. according to theory for a size 1=> 400.000 messages/day and unit. So, in one hand we have 14 units x 400.000 messages/day and unit= 5.600.000 messages/day max. On the other hand, we have (60+5) meesages/devices and sec x 60 sec/min x 60 min/h x 24 h/day = 5.616.000 messages/day. That´s why I think it is NO.
https://learn.microsoft.com/en-us/training/modules/examine-iot-hub-properties/2-iot-hub-tiers?ns-enrollment-type=learningpath&ns-enrollment-id=learn.wwl.implement-iot-device-communication-by-using-azure-iot-sdks

upvoted 3 times

hotwheelsinsf 2 years, 2 months ago

you might add some type of documentation llink on this

upvoted 1 times

123456789456789 2 years, 2 months ago

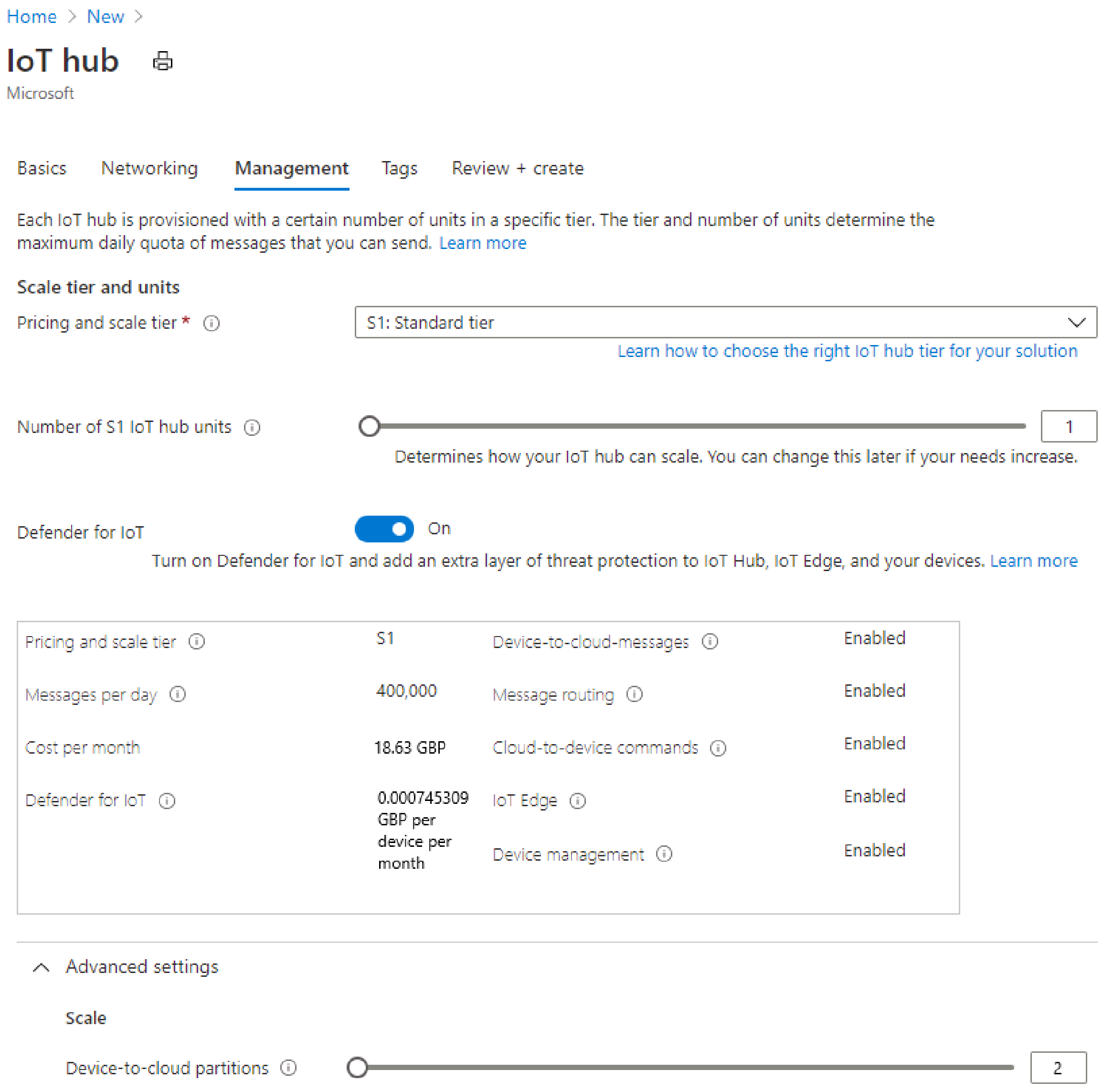
Hello, I think the third option is not correct. Where could you find that it only sends one single 60kb per minute ?

upvoted 1 times

HOTSPOT

-

You create an Azure IoT hub as shown in the following exhibit.



IoT hub

Microsoft

Each IoT hub is provisioned with a certain number of units in a specific tier. The tier and number of units determine the maximum daily quota of messages that you can send. Learn more

Scale tier and units

Pricing and scale tier * ⓘ: S1: Standard tier

Learn how to choose the right IoT hub tier for your solution

Number of S1 IoT hub units ⓘ: 1

Determines how your IoT hub can scale. You can change this later if your needs increase.

Defender for IoT: On

Turn on Defender for IoT and add an extra layer of threat protection to IoT Hub, IoT Edge, and your devices. Learn more

Pricing and scale tier ⓘ	S1	Device-to-cloud-messages ⓘ	Enabled
Messages per day ⓘ	400,000	Message routing ⓘ	Enabled
Cost per month	18.63 GBP	Cloud-to-device commands ⓘ	Enabled
Defender for IoT ⓘ	0.000745309 GBP per device per month	IoT Edge ⓘ	Enabled
		Device management ⓘ	Enabled

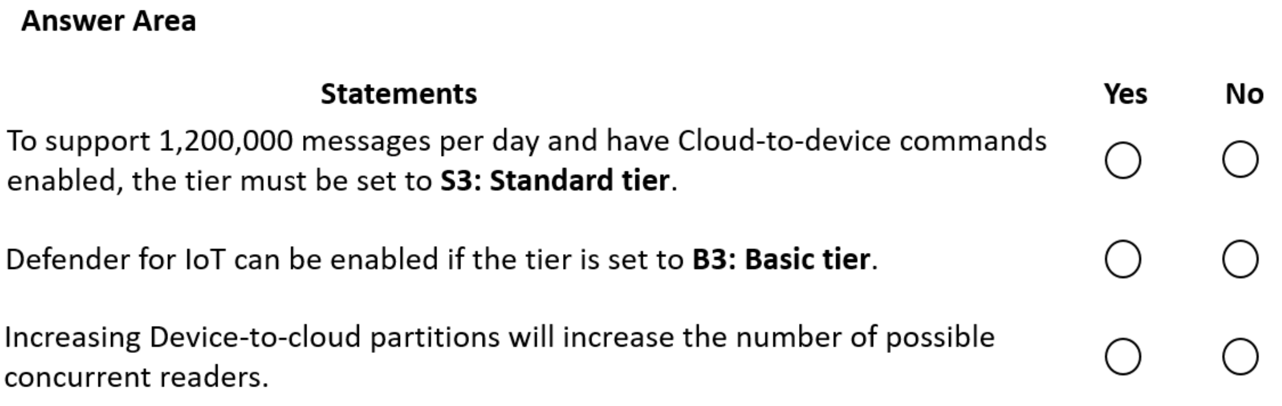
Advanced settings

Scale

Device-to-cloud partitions ⓘ: 2

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

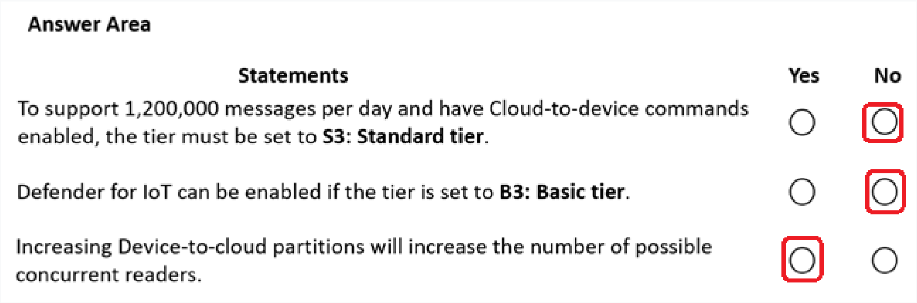
NOTE: Each correct selection is worth one point.



Answer Area

Statements	Yes	No
To support 1,200,000 messages per day and have Cloud-to-device commands enabled, the tier must be set to **S3: Standard tier**.	○	○
Defender for IoT can be enabled if the tier is set to **B3: Basic tier**.	○	○
Increasing Device-to-cloud partitions will increase the number of possible concurrent readers.	○	○

Suggested Answer:



Answer Area

Statements	Yes	No
To support 1,200,000 messages per day and have Cloud-to-device commands enabled, the tier must be set to **S3: Standard tier**.	○	[✓ ○]
Defender for IoT can be enabled if the tier is set to **B3: Basic tier**.	○	[✓ ○]
Increasing Device-to-cloud partitions will increase the number of possible concurrent readers.	[✓ ○]	○

PreetDP900 2 years, 2 months ago

Correct Answers.

I was confused for the first Question first but then understood. S2 will fulfill the message limits so no need to "Must have S3".

upvoted 3 times

Question #24 Topic 1

DRAG DROP

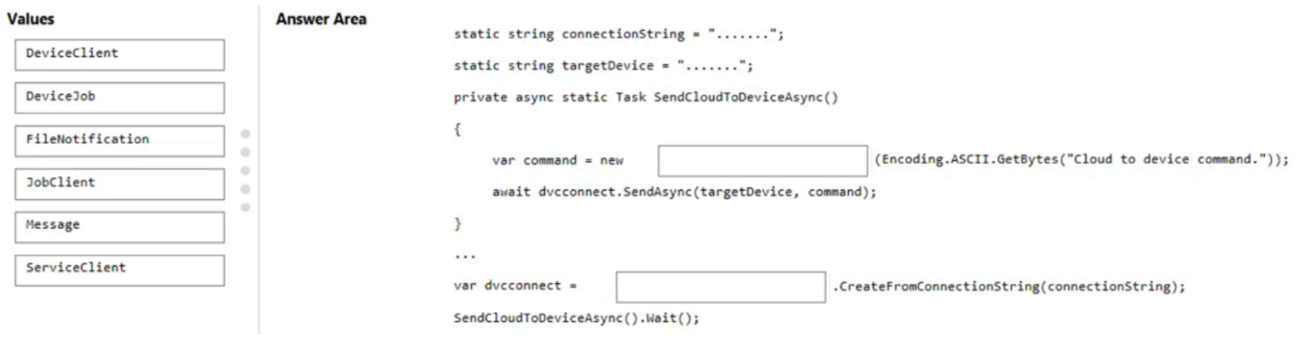
-

You are building an IoT device management application by using the Azure IoT Hub Service SDK.

You need to configure the application to send instructions via an IoT hub to IoT devices.

How should you complete the code? To answer, drag the appropriate values to the correct targets. 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.



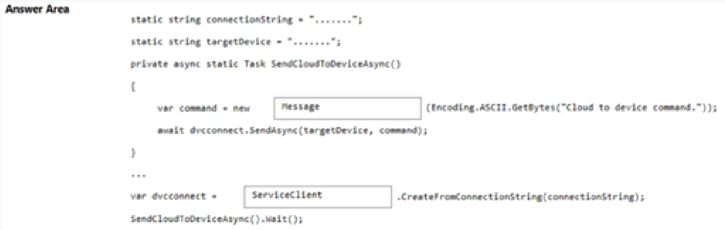
Values

- DeviceClient
- DeviceJob
- FileNotification
- JobClient
- Message
- ServiceClient

Answer Area

```
static string connectionString = ".......";
static string targetDevice = ".......";
private async static Task SendCloudToDeviceAsync()
{
    var command = new [          ] (Encoding.ASCII.GetBytes("Cloud to device command."));
    await dvcconnect.SendAsync(targetDevice, command);
}
...
var dvcconnect = [          ] .CreateFromConnectionString(connectionString);
SendCloudToDeviceAsync().Wait();
```

Suggested Answer:



Answer Area

```
static string connectionString = ".......";
static string targetDevice = ".......";
private async static Task SendCloudToDeviceAsync()
{
    var command = new [Message] (Encoding.ASCII.GetBytes("Cloud to device command."));
    await dvcconnect.SendAsync(targetDevice, command);
}
...
var dvcconnect = [ServiceClient] .CreateFromConnectionString(connectionString);
SendCloudToDeviceAsync().Wait();
```

Tyrel 2 years ago

Correct.

upvoted 1 times

Question #25 *Topic 1*

You have an Azure IoT solution that includes an Azure IoT hub. The hub has the following configurations:

- Name: IoTHub1
- Retain for: 1 Day
- Azure region: East US
- Number of IoT hub units: 1
- Pricing and scale tier: B1 - Basic
- Consumer groups: $Default Only
- Connectivity method: Public endpoint (all networks)

You need to ensure that the solution supports IoT Hub jobs that update device twin properties.

What should you do first?

A. Create a device twin.

B. Create a module twin.

C. Create a shared access policy.

D. Upgrade to a standard tier IoT hub.

Suggested Answer: *D*

xixi4den 2 years, 1 month ago

device twins is not supported on Basic tier

https://learn.microsoft.com/en-us/azure/iot-hub/iot-hub-scaling

upvoted 3 times

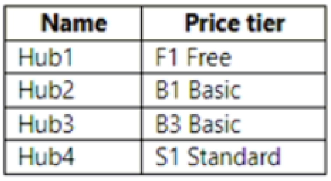
beltransrg2 2 years, 2 months ago

The ability to perform jobs that update device twin properties is supported on all tiers of Azure IoT Hub. Therefore, the correct answer would be A. Create a device twin.

upvoted 1 times

Question #26 *Topic 1*

You have an Azure subscription that contains the Azure IoT hubs shown in the following table.



Name	Price tier
Hub1	F1 Free
Hub2	B1 Basic
Hub3	B3 Basic
Hub4	S1 Standard

You plan to evaluate the Microsoft Defender for IoT agent-based solution in Built-in mode.

Which IoT hubs can you use for the evaluation?

A. Hub4 only

B. Hub3 and Hub4 only

C. Hub2, Hub3, and Hub4 only

D. Hub1, Hub2, Hub3, and Hub4

Suggested Answer: *A*

Tyrel 2 years ago

Correct.

upvoted 1 times

Question #27 *Topic 1*

You have an Azure IoT solution that contains an Azure IoT hub in the S1 - Standard-tier. The IoT hub has four built-in event endpoint partitions.

You need to increase the number of partitions to eight. The solution must minimize administrative effort.

What should you do?

A. From the Pricing and scale blade of the IoT hub, change the tier to S2 - Standard.

B. From the Pricing and scale blade of the IoT hub, increase the number of IoT Hub units to eight.

C. Create a new IoT hub and set Device-to-cloud partitions to eight.

D. Create a new IoT hub and set the number of IoT Hub units to eight.

Suggested Answer: *C*

Tyrel 2 years ago

Correct.

"The partition limit is chosen when an IoT hub is created, and can't be changed. The maximum limit of device-to-cloud partitions for basic tier and standard tier IoT hubs is 32."

upvoted 1 times

Question #28 *Topic 1*

HOTSPOT

-

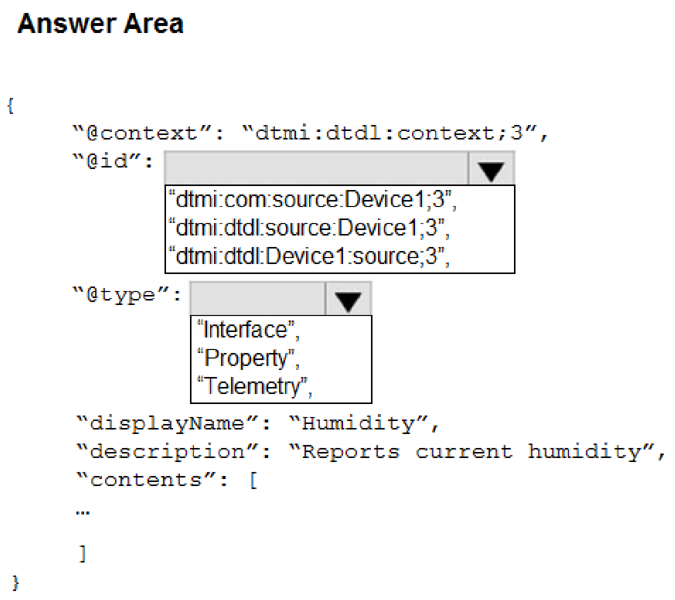
You have an Azure IoT solution that includes an IoT device named Device1.

You are creating an IoT Plug and Play model for Device1.

On Device1, you create a device model file in a folder named dtmi/com/source/.

How should you complete the model? To answer, select the appropriate options in the answer area.

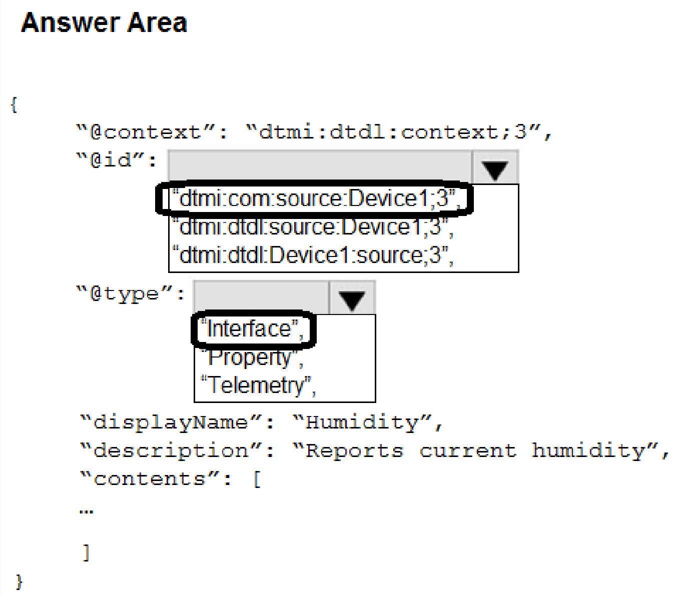
NOTE: Each correct selection is worth one point.



Answer Area

```
{
      “@context”: “dtmi:dtdl:context;3”,
      “@id”: ▼
            “dtmi:com:source:Device1;3”,
            “dtmi:dtdl:source:Device1;3”,
            “dtmi:dtdl:Device1:source;3”,
      “@type”: ▼
            “Interface”,
            “Property”,
            “Telemetry”,
      “displayName”: “Humidity”,
      “description”: “Reports current humidity”,
      “contents”: [
      …
      ]
}
```

Suggested Answer:



Answer Area

```
{
      “@context”: “dtmi:dtdl:context;3”,
      “@id”: ▼
            [✓ “dtmi:com:source:Device1;3”,]
            “dtmi:dtdl:source:Device1;3”,
            “dtmi:dtdl:Device1:source;3”,
      “@type”: ▼
            [✓ “Interface”,]
            “Property”,
            “Telemetry”,
      “displayName”: “Humidity”,
      “description”: “Reports current humidity”,
      “contents”: [
      …
      ]
}
```

Tyrel 2 years ago

Correct answer.

upvoted 1 times

Question #1 *Topic 2*

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.

Suggested Answer: *AC*
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

Community vote distribution
AB (64%) AC (36%)

alexgrdi89 Highly Voted 👍 5 years ago
Answer A and C
upvoted 22 times

angelsrp Highly Voted 👍 4 years, 11 months ago
i think answers are A and D, by default allocation policy is set to "Evenly weighted distribution", so, you dont need to manage the policy, instead, you need to upload and verify a X.509 certificate.

https://docs.microsoft.com/en-us/learn/modules/securely-provision-iot-devices-at-scale-with-device-provisioning-service/4-exercise-create-dps-resource-root-certificate-group-enrollment
upvoted 19 times

adi38911 4 years, 10 months ago
Do not agree to this answer as question does not mention anything about which attestation mechanism to use, If at all one is not bothered about Generating CA signing certificate for devices then option A & C makes more sense since with individual enrollments you can accept device enrollments using SAS key as well and thats the basic difference when creating enrollments, individual enrollments are possible with SAS key and certificate both but group enrollments are only possible with certificate so in my opinion if question had something on the attestation mechanism, i would have better choice to select the correct option but with plane vanilla set up of DPS i would go with Option A & C
upvoted 5 times

IMARRA Most Recent 🕓 1 year, 11 months ago
Selected Answer: AB
Clearly explained here:
https://learn.microsoft.com/en-us/azure/iot-dps/quick-setup-auto-provision
upvoted 1 times

feyey 2 years, 2 months ago
Selected Answer: AC
A. From the Linked IoT hubs blade of the Device Provisioning Service, link an Azure IoT hub: The Device Provisioning Service needs to be linked to an Azure IoT hub to enable it to provision devices for the hub. The linked IoT hub acts as the target hub for devices provisioned by the service. This step is required to enable the Device Provisioning Service to accept new device enrollments.

C. From the Manage allocation policy blade of the Device Provisioning Service, configure an allocation policy: An allocation policy determines how devices are assigned to IoT hubs when they are provisioned by the Device Provisioning Service. It specifies the IoT hub to which a device should be provisioned based on criteria such as device ID, device group, and tags. Configuring an allocation policy is an important step for ensuring that the Device Provisioning Service can efficiently accept new device enrollments.

Therefore, options A and C are the correct

upvoted 1 times

feyey 2 years, 2 months ago

Selected Answer: AC

A. From the Linked IoT hubs blade of the Device Provisioning Service, link an Azure IoT hub: The Device Provisioning Service needs to be linked to an Azure IoT hub to enable it to provision devices for the hub. The linked IoT hub acts as the target hub for devices provisioned by the service. This step is required to enable the Device Provisioning Service to accept new device enrollments.

C. From the Manage allocation policy blade of the Device Provisioning Service, configure an allocation policy: An allocation policy determines how devices are assigned to IoT hubs when they are provisioned by the Device Provisioning Service. It specifies the IoT hub to which a device should be provisioned based on criteria such as device ID, device group, and tags. Configuring an allocation policy is an important step for ensuring that the Device Provisioning Service can efficiently accept new device enrollments.

Therefore, options A and C are the correct

upvoted 1 times

feyey 2 years, 2 months ago

Selected Answer: AC

A. From the Linked IoT hubs blade of the Device Provisioning Service, link an Azure IoT hub: The Device Provisioning Service needs to be linked to an Azure IoT hub to enable it to provision devices for the hub. The linked IoT hub acts as the target hub for devices provisioned by the service. This step is required to enable the Device Provisioning Service to accept new device enrollments.

C. From the Manage allocation policy blade of the Device Provisioning Service, configure an allocation policy: An allocation policy determines how devices are assigned to IoT hubs when they are provisioned by the Device Provisioning Service. It specifies the IoT hub to which a device should be provisioned based on criteria such as device ID, device group, and tags. Configuring an allocation policy is an important step for ensuring that the Device Provisioning Service can efficiently accept new device enrollments.

Therefore, options A and C are the correct

upvoted 1 times

feyey 2 years, 2 months ago

Selected Answer: AB

To ensure that the Device Provisioning Service can accept new device enrollments, the next two actions that should be performed are:

A. From the Linked IoT hubs blade of the Device Provisioning Service, link an Azure IoT hub: The Device Provisioning Service needs to be linked to an Azure IoT hub to enable it to provision devices for the hub. The linked IoT hub acts as the target hub for devices provisioned by the service. This step is required to enable the Device Provisioning Service to accept new device enrollments.

B. From the Azure portal, create a new Azure IoT hub: Before the Device Provisioning Service can start accepting new device enrollments, a new Azure IoT hub needs to be created. This IoT hub will act as the destination for devices that are provisioned by the Device Provisioning Service. If an IoT hub already exists, it can be used instead of creating a new one. This step is also required to enable the Device Provisioning Service to accept new device enrollments.

Therefore, options A and B are the correct actions

upvoted 1 times

satishk4u 2 years, 11 months ago

Selected Answer: AB

Should be A and B

upvoted 1 times

zb99 2 years, 11 months ago

Selected Answer: AB

Only two steps that are not optional.

upvoted 1 times

rsamant 3 years ago

Answer is A & D

Allocation policy is not mandatory.

https://docs.microsoft.com/en-us/azure/iot-dps/how-to-manage-enrollments

upvoted 2 times

liberty123 3 years, 3 months ago

Selected Answer: AC

Agree with AC

upvoted 1 times

Sophiamiaa 3 years, 5 months ago

A and D..

upvoted 2 times

coramella 3 years, 5 months ago

Selected Answer: AB

The answer makes no reference either to the authentication mechanism or to a different allocation policy, moreover the only existing resource is the resource the group sand so the iot hub must be created.

upvoted 3 times

Marusyk 3 years, 7 months ago

why it is not A and B?

upvoted 2 times

JeeBi 4 years, 1 month ago

I also choose A and B. The question does not state that certificates need to be used and the allocation policy gets a default value.

upvoted 8 times

jracabrera 4 years, 1 month ago

I agree with A and B.

upvoted 4 times

dinesh_tng 4 years, 2 months ago

There are only two required steps A & B. C and D are optional.

Step 1 - Create IoT Hub (As creation of Hub is not mentioned)

Step 2 - Link IoT Hub

upvoted 5 times

AnonymousJhb 4 years, 2 months ago

B is wrong, as per Microsoft recommended best practice, first deploy the IoT Hub, before DPS.

https://docs.microsoft.com/en-us/azure/iot-dps/quick-setup-auto-provision

Hence, A is next step and C.

D is a security option inside of C which is either individual or over-arching group enrollment

upvoted 5 times

dinesh_tng 4 years ago

technically, we can create DPS without IoT Hub. For for DPS to accept new enrolments, IoT Hub is must. Allocation policy is optional and CA will impact devices only from selected category. So my ans will be A and B.

upvoted 6 times

tita_tovenaar 4 years ago

correct. technically , Microsoft connects Allocation Policy to IoT Hubs. A DPS can have *custom* allocation policies

upvoted 3 times

GParreiras 4 years, 2 months ago

Correct answer is A and C

upvoted 3 times

Question #2 *Topic 2*

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.

Suggested Answer: *CE*
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

Community vote distribution
CE (100%)

bob2Be Highly Voted 4 years, 12 months ago
Should not hte answer be C & E.
The question asks for stopping the auto enrollment for 1000 devices out of the 10,000
upvoted 19 times

getazusername 4 years, 7 months ago
Yes, you need first to disconnect them and then prevent it from enrollment.
upvoted 4 times

FacuTheRock Highly Voted 4 years, 8 months ago
I think correct answers should be C and E.
The statement says "MUST PREVENT CONNECTIONS AND AUTO-ENROLLMENT".
By deleting the enrollment entry, we are preventing devices from auto-enrolling but not from connecting. That's why we need also to delete their identities in the Identity Registry
- Blacklisting the Root certificate would impact on the 10k devices, and we only want to disallow 1k devices.
Even when we can create intermmediate certificates to disallow certain path in the chain, that option is not listed, and Option B clearly mentions ROOT CERTIFICATE, so I dont agree with option B
upvoted 13 times

tedsi 4 years, 7 months ago
I agree that this is correct. You don't want to affect the remaining devices - so can't blacklist. And since this is a permanent removal, disenroll and deregistration is the right process.
upvoted 5 times

liberty123 Most Recent 3 years, 3 months ago
Selected Answer: CE
Agree with CE
upvoted 1 times

kokosek 4 years, 3 months ago
For me, B is a wrong answer. It's required do deprovision 1000 of 10000 devies. Blacklisting root certificate seems like it will disable enrollments with other intermediate/leaf certificates chained to that root cert. There is a possibility, that other devices require that root cert, so doing that may cause issues in real life scenario.

upvoted 3 times

BoomJosh 4 years, 3 months ago

Appeared for exam on 3/24/2021 and successfully cleared it, this question was there.

upvoted 3 times

LiamRT 4 years, 5 months ago

To deprovision a device that has an individual enrollment:

1. Disenroll the device from your provisioning service
2. Disable or delete the device in the identity registry of the IoT hub that it was provisioned to.

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

C & E it seems.

upvoted 8 times

ipindado2020 4 years, 7 months ago

Agree on BC

upvoted 1 times

ipindado2020 4 years, 7 months ago

Changed my mind to CE

upvoted 5 times

rjdask 4 years, 8 months ago

Confusion here seems to be surrounding the term: blacklist. Given answer is correct. The documentation actually states this as being "disallowed". X.509 certificates are typically arranged in a certificate chain of trust. If a certificate at any stage in a chain becomes compromised, trust is broken. "The certificate must be disallowed to prevent Device Provisioning Service from provisioning devices downstream in any chain that contains that certificate. To learn more about X.509 certificates and how they are used with the provisioning service, see X.509 certificates."

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

upvoted 1 times

angelsrp 4 years, 11 months ago

Correct ans are BC:

Deprovisioning process:

-Disenrollment (Blacklist individual devices or an enrollment group

-Deregister (Delete de device enrollment entries)

upvoted 3 times

thestillheron 4 years, 11 months ago

Guidance for this scenario has changed. To blacklist specific devices in an Enrollment group, without blacklisting the entire enrolment group, you can add the specific devices and their certificates as individually enrolled devices, and then disable them.

DPS first checks individual enrolments. If it finds a match that is disabled, it will refuse the connection, even if a non-blacklisted match exists in group enrolments:

https://docs.microsoft.com/en-us/azure/iot-dps/how-to-revoke-device-access-portal#blacklist-specific-devices-in-an-enrollment-group

upvoted 2 times

EyeeyeeyeeyeeyeeyeeyeeyeSPIDER 4 years, 11 months ago

B & C is the correct answer - blacklist the cert or delete the enrollment group

To blacklist the certificate, you can either disable or delete its enrollment group.

upvoted 3 times

Mardy 4 years, 12 months ago

I think it's B & E - https://docs.microsoft.com/en-us/azure/iot-dps/how-to-unprovision-devices

upvoted 4 times

redSandton 4 years, 11 months ago

I also think B&E because there are 2 basic steps to disenroll an automatically provisioned device ,first blacklist the certificate/disable the enrollment group .secondly delete/disable the device entry in the IoT hub identity registry

upvoted 1 times

Bengkel 4 years, 10 months ago

It states "For devices that use X.509 attestation, you may want to disable/delete an entry in the hierarchy of your existing enrollment groups". So this is not always mandatory. The page is clear Disenroll en Deregister = C & E

upvoted 5 times

⊟ **niceguy0371** 4 years, 10 months ago

I think it's B&E. 1st: The question is not clear if these 1.000 devices are in 1 enrollment group or if all 10,000 devices are in one enrollment group. 2nd: The question states that there are 10.000 devices and you only need to prevent 1000 devices for qutoenrollment. So, assuming that all the 10.000 devices are in the same enrollment group (because it's not specified), blacklisting the certificate affects all 10.000 devices.

So, you have to delete the enrollment entries for the 10000 devices and delete the device identity from the device registry in the IoT hub. Just like preventing one single IoT device from autoenrollment

upvoted 2 times

⊟ **niceguy0371** 4 years, 10 months ago

I mean C&E off course

upvoted 8 times

Question #3 *Topic 2*

HOTSPOT -

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

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

✑ Return the reported power consumption.

✑ Configure the desired fan speed.

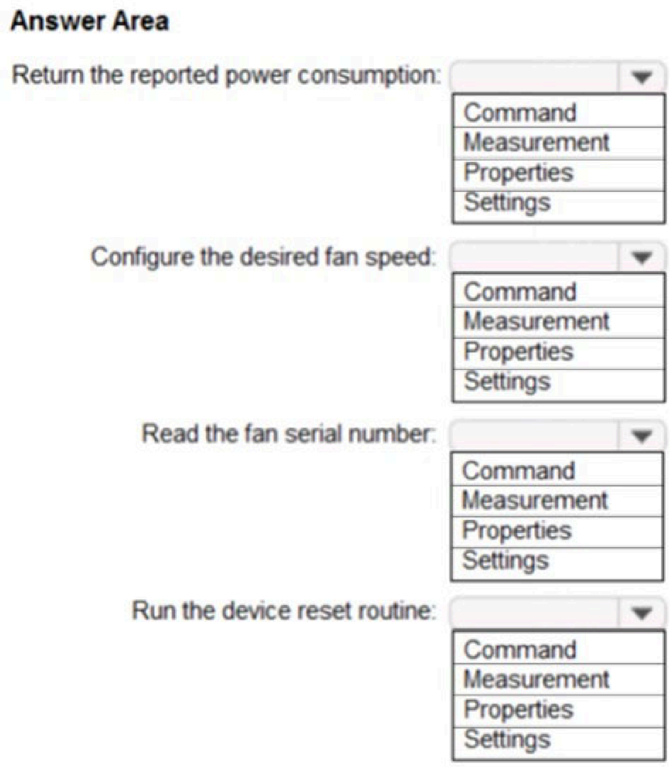
✑ Run the device reset routine.

✑ Read the fan serial number.

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

NOTE: Each correct selection is worth one point.

Hot Area:



Answer Area

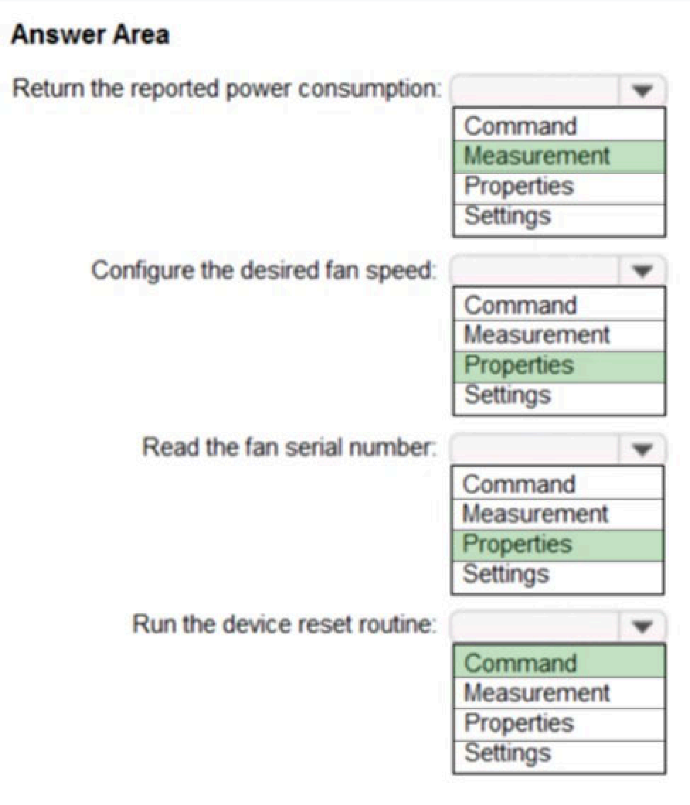
Return the reported power consumption: ▼
- Command
- Measurement
- Properties
- Settings

Configure the desired fan speed: ▼
- Command
- Measurement
- Properties
- Settings

Read the fan serial number: ▼
- Command
- Measurement
- Properties
- Settings

Run the device reset routine: ▼
- Command
- Measurement
- Properties
- Settings

Suggested Answer:



Answer Area

Return the reported power consumption:
- Command
- [✓ Measurement]
- Properties
- Settings

Configure the desired fan speed:
- Command
- Measurement
- [✓ Properties]
- Settings

Read the fan serial number:
- Command
- Measurement
- [✓ Properties]
- Settings

Run the device reset routine:
- [✓ Command]
- Measurement
- Properties
- Settings

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

Arockia Highly Voted 3 years, 7 months ago

The question Options are wrong. The correct answer is:

a. Return the reported power consumption --> Telemetry

b. Configure the desired fan speed --> Properties

c. Read the fan serial number --> Cloud Properties

d. Run the device reset routine --> command

upvoted 11 times

ridhz Most Recent 4 years ago

I am not sure but "Run device reset routine" sounds like a "Settings" answer.

Am I right?

upvoted 1 times

dedede 4 years ago

I am not sure but "Configure the desired fan speed" sounds like "Settings" answer.

Am I right?

upvoted 2 times

⊟ **exam67** 4 years ago

No, because there is no "settings" when it comes to device template "interfaces"

https://docs.microsoft.com/en-us/azure/iot-central/core/concepts-device-templates#interfaces

upvoted 3 times

Question #4 *Topic 2*

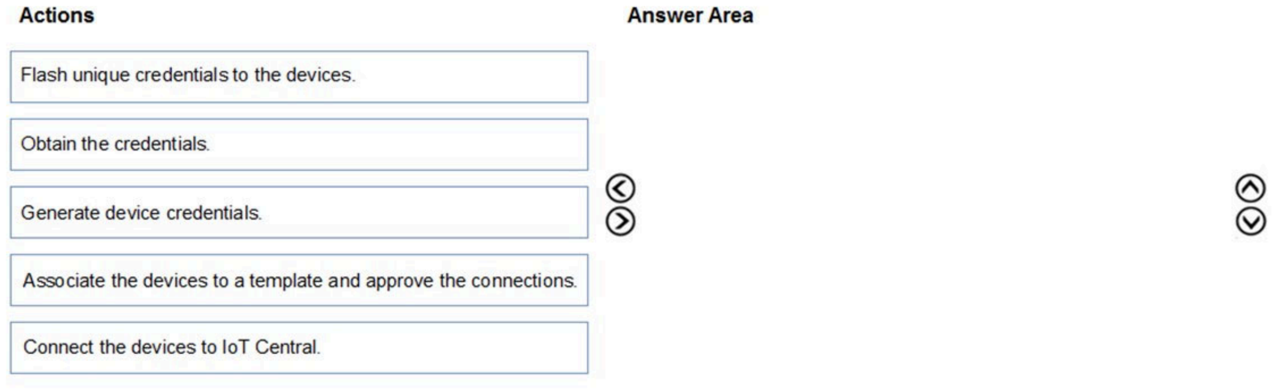
DRAG DROP -

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

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

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

Select and Place:

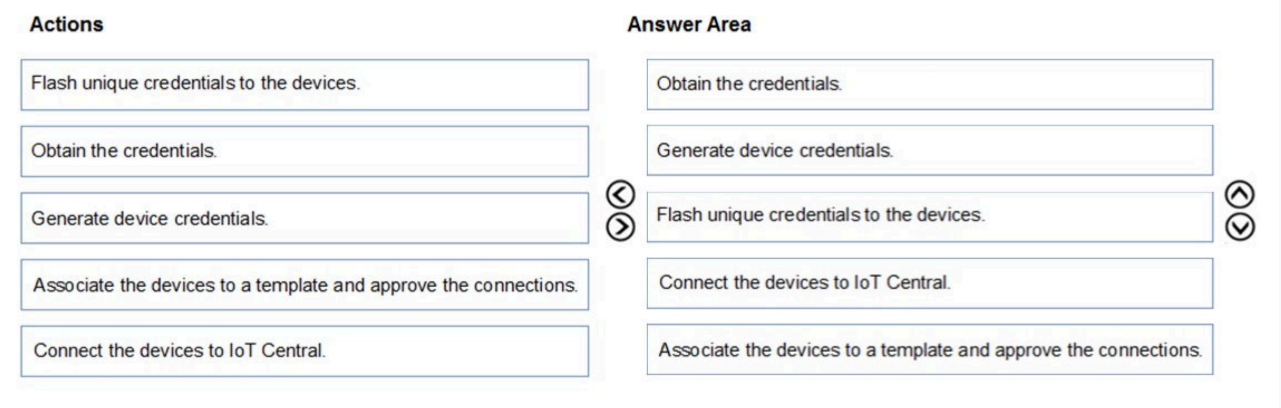


Actions

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

Answer Area

Suggested 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.

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

⊟ **theiotguy** Highly Voted 5 years ago

I think it should be

- Obtain cred
- Generate cred
- Flash cred
- Connect
- Associate

We need to obtain the IoT Central application's group primary key first to generate the device credentials.

https://docs.microsoft.com/en-us/azure/iot-central/core/concepts-get-connected#connect-without-registering-devices

upvoted 51 times

⊟ **EyeeyeeyeeyeeyeeyeeyeeyeSPIDER** Highly Voted 4 years, 11 months ago theiotguy is right. The question is not well written. The first step should be written as "Obtain the group primary key from the IoT Central enrollment group". The second step should be "User the group primary key to generate device credentials"

Purposely obtuse in order to confuse. Poor question.

upvoted 12 times

Sudhansu21 Most Recent 3 years, 4 months ago

Came in Exam Feb 2022.

upvoted 2 times

BoomJosh 4 years, 4 months ago

The given answers seem to be correct. Check this link https://docs.microsoft.com/en-us/azure/iot-central/core/tutorial-connect-device?pivots=programming-language-csharp

upvoted 2 times

SabSal 4 years, 5 months ago

Question appeared on exam

upvoted 3 times

> **Exterminator1512** 4 years, 5 months ago
>
> You gave it recently?
> How many questions did appear?
>
> upvoted 1 times

Yong2020 5 years ago

I am no expert but according to the content, the sequence should be
- Generate cred
- Obtain cred
- Flash cred
- Connect
- Associate

upvoted 7 times

Yong2020 5 years ago

To make sense, "obtain the credential" should be moved to step 3

upvoted 3 times

Question #5 *Topic 2*

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

Suggested Answer: *AC*
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

Community vote distribution
AC (100%)

AshokDhanekula Highly Voted 3 years, 11 months ago
We need the Scope ID and Device ID to configure the device in IoT Central. Answer should be C & E.
upvoted 11 times

pdeng 2 years, 6 months ago
https://learn.microsoft.com/en-us/azure/iot-central/core/quick-deploy-iot-central
upvoted 1 times

exnaniantwort Most Recent 2 years, 6 months ago
Selected Answer: AC
Get connection information
When you run the sample device application later in this tutorial, you need the following configuration values:

ID scope: In your IoT Central application, navigate to Permissions > Device connection groups. Make a note of the ID scope value.
Group primary key: In your IoT Central application, navigate to Permissions > Device connection groups > SAS-IoT-Devices. Make a note of the shared access signature Primary key value.
https://learn.microsoft.com/en-us/azure/iot-central/core/tutorial-connect-device?pivots=programming-language-javascript
upvoted 3 times

RajeevP26 3 years, 5 months ago
Not sure about the exact answer, but below seems to be the relevant link
https://docs.microsoft.com/en-us/azure/iot-central/core/concepts-get-connected
upvoted 1 times

d0bermannn 3 years, 6 months ago
Selected Answer: AC
A&C is correct as we see on links
https://docs.microsoft.com/bs-cyrl-ba/azure/iot-central/core/tutorial-connect-device?pivots=programming-language-ansi-c
And as IoT Central uses DPS under the hood for device connect
https://docs.microsoft.com/en-us/azure/iot-dps/concepts-service
"The device ID is the ID as it appears in IoT Hub. The desired device ID may be set in the enrollment entry, but it is not required to be set. "
upvoted 4 times

Arockia 3 years, 7 months ago

C & E --> Scope ID, Device ID.
upvoted 2 times

⊟ **trickerk** 3 years, 10 months ago

Given answer is correct according https://docs.microsoft.com/bs-cyrl-ba/azure/iot-central/core/tutorial-connect-device:

When you run the sample device application later in this tutorial, you need the following configuration values:
- ID scope: In your IoT Central application, navigate to Administration > Device Connection. Make a note of the ID scope value.
- 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.

upvoted 4 times

⊟ **exam67** 4 years ago

I vote for C+E. A device application need the device-id to connect. SAS key may be also correct, but there are other enrollment methods available
upvoted 2 times

Question #6 *Topic 2*

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.

Suggested Answer: *BCF*

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

trickerk Highly Voted 👍 3 years, 10 months ago

Given answer is correct. The question explanation and reference are clear.

upvoted 9 times

trickerk 3 years, 10 months ago

- Update desired properties
- Update tags
- Invoke direct methods

upvoted 7 times

ipindado2020 Highly Voted 👍 4 years, 7 months ago

BCF is ok

upvoted 8 times

JeeBi Most Recent ⏲ 4 years, 1 month ago

Tags are not visible on the device. Should be BCD, adding 'D. Send cloud-to-device messages.'

upvoted 2 times

berkejf 4 years, 1 month ago

You are wrong. IoT Hub jobs is running on IoT hub not on device. tag can be updated by IoT hub jobs. Correct answer is BCF

upvoted 6 times

Question #7 *Topic 2*

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.

Suggested Answer: *B*
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/

userfriendly Highly Voted 4 years, 7 months ago
B.
Here is explanation:
https://docs.microsoft.com/en-us/samples/azure-samples/iot-hub-dotnet-autoscale/iot-hub-dotnet-autoscale/
The solution consists of three Azure Functions, each one playing a specific part in the Azure Durable Functions framework
IotHubScaleInit – this function is executed on a regular timer. This function checks to see if an instance of the Orchestrator function is running and, if not, starts one. In essence, it's used to "kick off" the solution and make sure it's always running. For the sample, it is set for once per hour, but can be set to any period.
IotHubScaleOrchestrator – this function implements the "Orchestrator" for the solution. It's role in the pattern is to manage the execution of the worker function (asynchronously, and safely), and to, once it's done, re-schedule itself for execution after a specific number of minutes.
IotHubScaleWorker – this is the function that performs the actions of checking to see if the IoTHub needs to be scales and, if so, scaling it.
upvoted 10 times

SanjuB Highly Voted 4 years, 4 months ago
No confusion B is the correct answer.
Create Azure Function to retrieve the Quota Metrics of the IoT Hub
upvoted 7 times

getazusername Most Recent 4 years, 7 months ago
If you are approaching the allowed message limit on your IoT hub, you can use these steps to automatically scale to increment an IoT Hub unit in the same IoT Hub tier.
https://docs.microsoft.com/de-de/azure/iot-hub/iot-hub-scaling
https://docs.microsoft.com/de-de/samples/azure-samples/iot-hub-dotnet-autoscale/iot-hub-dotnet-autoscale/
upvoted 2 times

angelsrp 4 years, 11 months ago
Ans is B, the answer could be D but is talking about iothub "code".
upvoted 5 times

Question #8 *Topic 2*

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

Suggested Answer: *A*
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

Mardy Highly Voted 4 years, 11 months ago
Ans is A Registration ID is a required field
upvoted 25 times

Bengkel 4 years, 10 months ago
When adding an enrollment the device ID should be provided, during provisioning the registration ID is required. I agree answer should be A.
upvoted 5 times

dinesh_tng 4 years, 2 months ago
Device ID is optional, whereas registration ID is mandatory. If Device ID is not provided, it is autofill by Registration ID. I agree answer should be A.
upvoted 6 times

pdeng 2 years, 6 months ago
Reference to https://learn.microsoft.com/en-us/azure/iot-dps/media/quick-create-simulated-device-symm-key/create-individual-enrollment-nodejs.png
upvoted 1 times

LiamRT 4 years, 5 months ago
yes, but the Reg. ID is not a 'detail from the enrollment'. B, Primary Key is correct IMO.
upvoted 2 times

ADC88 4 years, 2 months ago
Correct and confirmed here: https://docs.microsoft.com/en-us/rest/api/iot-dps/createorupdateindividualenrollment/createorupdateindividualenrollment#request-body
upvoted 4 times

redSandton Highly Voted 4 years, 11 months ago
I would say the answer is A , Registration ID is mandatory
https://docs.microsoft.com/en-us/azure/iot-dps/concepts-device
upvoted 9 times

satishk4u Most Recent 3 years ago
Ans is A Registration ID for individual device enrollment
upvoted 1 times

[Removed] 3 years, 1 month ago
I think the question wants to say "which field is mandatory in 'Add enrollment screen'". I that case, only registration id is mandatory.
Very confusing question, I totally agree.

upvoted 1 times

⊟ **dinesh_tng** 4 years ago

It shall be B. Reason - Device Enrollment is already done, now need Primary Key which will be used to create SAS token

upvoted 2 times

⊟ **Ashwinee** 4 years, 3 months ago

Confusing question, question says which details is required, then ans is A. but to auto provision device id is used !

upvoted 1 times

⊟ **SanjuB** 4 years, 4 months ago

The Device Identity of the IoT Hub

Is the correct one. Checked the documents and couple of other sites.

upvoted 1 times

⊟ **Exterminator1512** 4 years, 5 months ago

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

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

GIven answer is correct
Too much confusion in the comments

upvoted 3 times

⊟ **RowanX** 4 years, 5 months ago

Answer B is correct. The RegID will be used to generate a symmetric key (e.g. the primary and secondary key) which will be used to authenticate with the DPS during provisioning.

upvoted 2 times

⊟ **MasDen** 4 years, 6 months ago

Answer is B.
Registration ID is a device identifier not enrollment group, so it is not A.
"Open the enrollment and copy the value of your generated Primary Key."
https://docs.microsoft.com/en-us/azure/iot-dps/how-to-legacy-device-symm-key

upvoted 2 times

⊟ **rjdask** 4 years, 8 months ago

A and/or B for a symmetric device. https://docs.microsoft.com/bs-latn-ba/azure/iot-dps/quick-create-device-symmetric-key-python#prepare-the-device-provisioning-code

upvoted 1 times

⊟ **Sudipta3009** 4 years, 9 months ago

Definitely A is the Answer

upvoted 7 times

⊟ **angelsrp** 4 years, 11 months ago

Ans is A:
https://docs.microsoft.com/en-us/azure/iot-dps/about-iot-dps

upvoted 7 times

⊟ **dcpavelescu** 4 years, 11 months ago

I will go with A) as "registration ID" is a mandatory field when adding a device individual enrollment in DPS

upvoted 8 times

Question #9 *Topic 2*

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

Suggested Answer: *D*
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

ADC88 Highly Voted 4 years, 2 months ago
Correct
upvoted 9 times

Passionate Most Recent 3 years, 10 months ago
Correct
upvoted 4 times

Question #10 *Topic 2*

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

Suggested Answer: *B*
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/

Community vote distribution

B (100%)

thestillheron Highly Voted 4 years, 11 months ago
The explanation of the answer is slightly inaccurate.
The question specifies that this information should not be propagated to the device.

For that reason, you should not put the information in the desired properties. Information in the "desired properties" will propagate to the devices.

You should instead put that information in the device tags. Information in the devices tags do not propagate to the physical device.
upvoted 16 times

AnonymousJhb 4 years, 1 month ago
WRONG!
A desired property: is WRITTEN by a back-end application and READ ONLY by a device.
A reported property is WRITTEN by a device and READ ONLY by a back-end application.
A tag is set by a back-end application and is NEVER sent to a device.
https://docs.microsoft.com/en-us/azure/iot-hub/tutorial-device-twins
(some clown has copied the same answers from 1 question to all the questions of this sim)
Read the question:
You need to configure the GPS location for the device entry. The solution must prevent the changes from being propagated (=NEVER) to the physical device.
=TAGS!
upvoted 3 times

exam67 Highly Voted 4 years, 1 month ago
The answer given is "no" and it is correct. The reason is that "resource tags" are not appropriate in this case. "Twin tags" are instead appropriate. In Azure every resource may have tags, those are called "resource tags", but are used for organizing resources. In this case "twin tags" are needed, which are a completely different concept
upvoted 6 times

d0bermannn 3 years, 6 months ago
best explanation here
upvoted 1 times

liberty123 Most Recent 3 years, 3 months ago
Selected Answer: B
Given answer is correct, Instead tags should be added to the Device twin.

upvoted 2 times

⊟ **srama79** 4 years, 7 months ago

I think the answer should be Yes ... it has to be tags ... i think the answer give is inaccurate.

upvoted 2 times

⊟ **srama79** 4 years, 7 months ago

Might be the Answer is right as a NO ... since we dont apply tags to the resource group, but to the device twin. I think the answer is right, but the explanation is misleading

upvoted 4 times

⊟ **joy5** 4 years, 8 months ago

A tag is set by a back-end application and is never sent to a device

upvoted 2 times

⊟ **tommy1244** 4 years, 9 months ago

Should the answer be yes?

upvoted 2 times

⊟ **certstowinirl** 4 years, 8 months ago

The answer is no. It should say "Device Tags" not resource tags.

upvoted 11 times

Question #11 *Topic 2*

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

Suggested Answer: *A*
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/

AkashKarve Highly Voted 👍 5 years ago
Ans should be "No". Bu using desired properties, changes will be propagated to device
upvoted 19 times

getazusername 4 years, 7 months ago
Yeah,
Answer is No because solution is: 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.
https://docs.microsoft.com/de-de/azure/iot-hub/iot-hub-devguide-device-twins
upvoted 6 times

tita_tovenaar 4 years ago
so youre saying the answer is Yes which is correct.
upvoted 3 times

AnonymousJhb 4 years, 1 month ago
WRONG!
you are correct, the answer is tags.
A desired property: is WRITTEN by a back-end application and READ ONLY by a device.
A reported property is WRITTEN by a device and READ ONLY by a back-end application.
A tag is set by a back-end application and is NEVER sent to a device.
https://docs.microsoft.com/en-us/azure/iot-hub/tutorial-device-twins
(some clown has copied the same answers from 1 question to all the questions of this sim)
upvoted 3 times

AnonymousJhb 4 years, 1 month ago
Read the question:
You need to configure the GPS location for the device entry. The solution must prevent the changes from being propagated (=NEVER) to the physical device.
=TAGS!
upvoted 1 times

dariuszdbr Highly Voted 👍 4 years, 3 months ago
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?

The answer should be Yes.

Updating/Adding Tags to Device Twin will not propagate changes to the device.

upvoted 13 times

AnonymousJhb 4 years, 1 month ago

you are correct, the answer is tags.

A desired property: is WRITTEN by a back-end application and READ ONLY by a device.

A reported property is WRITTEN by a device and READ ONLY by a back-end application.

A tag is set by a back-end application and is NEVER sent to a device.

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

(some clown has copied the same answers from 1 question to all the questions of this sim)

upvoted 4 times

Stephan99 Most Recent 4 years, 1 month ago

Yes

" The solution must prevent the changes from being propagated to the physical device."

upvoted 4 times

Ashwinee 4 years, 3 months ago

Ans is No. 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.

upvoted 1 times

ipindado2020 4 years, 7 months ago

B is the way

upvoted 3 times

EyeeyeeyeeyeeyeeyeeyeSPIDER 4 years, 11 months ago

The answer should be NO

upvoted 10 times

Question #12 *Topic 2*

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.

Suggested Answer: *BC*
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

⊟ **exam67** Highly Voted 👍 4 years ago
correct
upvoted 5 times

Question #13 *Topic 2*

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.

Suggested Answer: *A*
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

dcprint Highly Voted 4 years, 2 months ago
Very happy to know that you have appeared the exam on 24/3/21 and clearing it too.

Instead of commenting this , mention the correct answer.
upvoted 11 times

IMARRA Most Recent 2 years, 11 months ago
why not B?
upvoted 2 times

trickerk 3 years, 10 months ago
Given answer is correct
upvoted 3 times

cburdea 3 years, 7 months ago
What is the given answer? They changed a couple of answers recently
upvoted 1 times

tita_tovenaar 4 years ago
not sure why C can't be true (too). From https://docs.microsoft.com/en-us/azure/iot-develop/about-iot-sdks:

IoT Hub
The IoT Hub service SDKs allow you to build applications that easily interact with your IoT Hub to manage devices and security. You can use these SDKs to send cloud-to-device messages, invoke direct methods on your devices, update device properties, and more.
upvoted 3 times

d0bermannn 3 years, 5 months ago
Agreed, there is more then one way to skin a cat)
And SDK can do it also, and I would say SDK works under the hood of Automatic device management.
But MS always ask how to do something easiest and simpliest way, so A.
btw SDK is the only way to do it on basic tier because Automatic device management use twins
upvoted 1 times

d0bermannn 3 years, 5 months ago
disregard my words of basic tier,as we have no any twins there)
upvoted 1 times

⊟ **BoomJosh** 4 years, 3 months ago

Appeared for exam on 3/24/2021 and successfully cleared it, this question was there.

upvoted 3 times

⊟ **dcprint** 4 years, 2 months ago

Very happy to know that you have appeared the exam on 24/3/21 and cleared it too.

Instead of commenting this , mention the correct answer.

upvoted 17 times

Question #14 *Topic 2*

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.

Suggested Answer: *BE*

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

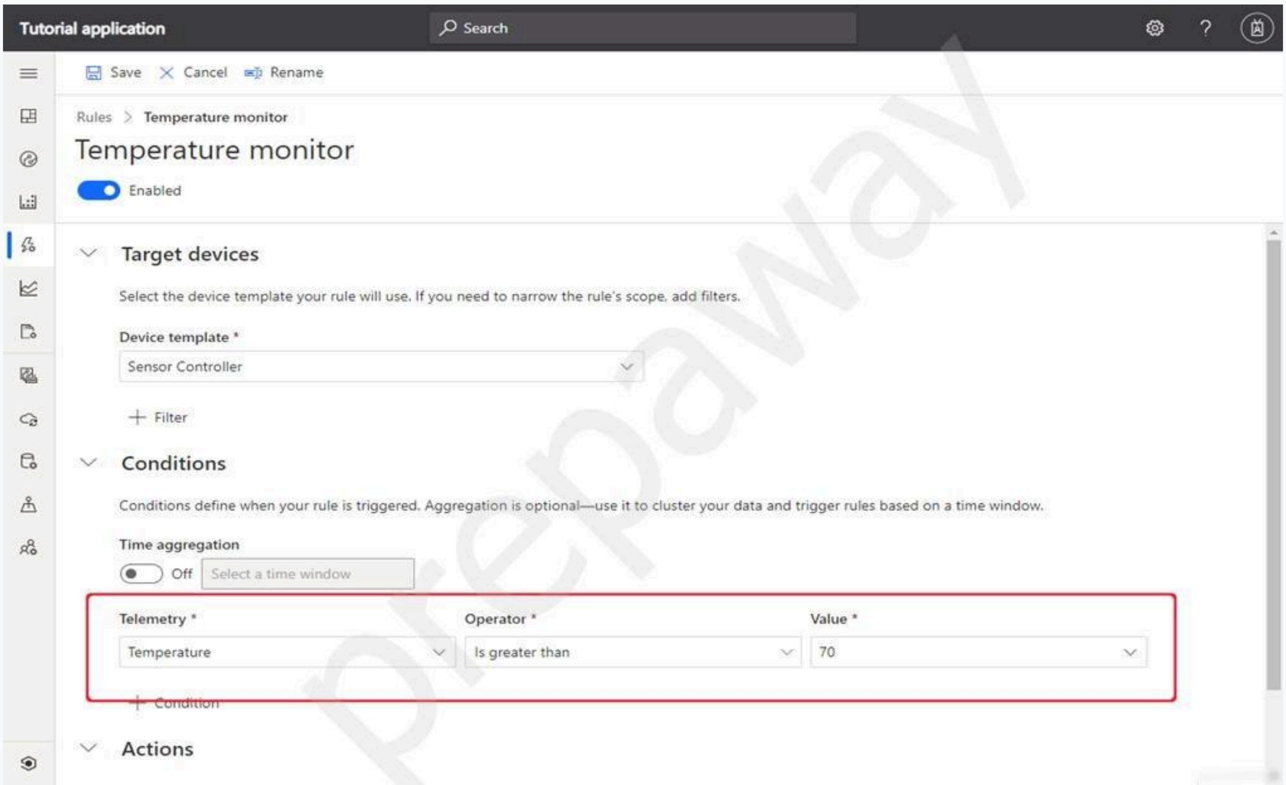
E: To create a telemetry rule, the device template must include at least one telemetry value. The rule monitors the temperature reported by the device and sends an email when it falls below 400 degrees.

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.



Rules > Temperature monitor

Temperature monitor

Enabled

Target devices

Select the device template your rule will use. If you need to narrow the rule's scope, add filters.

Device template *: Sensor Controller

Conditions

Conditions define when your rule is triggered. Aggregation is optional—use it to cluster your data and trigger rules based on a time window.

Time aggregation: Off — Select a time window

Telemetry *: Temperature

Operator *: Is greater than

Value *: 70

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

tita_tovenaar Highly Voted 4 years ago

answers are correct.

A is technically possible but unnecessary and complete overkill, so you would 'lose' a step. The other options are nonsense.

upvoted 8 times

BoomJosh Highly Voted 👍 4 years, 3 months ago

Appeared for exam on 3/24/2021 and successfully cleared it, this question was there.

upvoted 5 times

Question #15 *Topic 2*

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)

Suggested Answer: *A*

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

Community vote distribution

D (88%) 13%

⊟ **AlanYu** Highly Voted 👍 4 years, 1 month ago

I think the answer should be D? It requires dps to allocate devices based on the registration ID of the device and the geographic location of the device. And also the load-balance should be made. But the static configuration can not achieve this requirement. The logic can be written in the Azure Function. So I think answer is D.

upvoted 16 times

⊟ **AlanYu** 4 years, 1 month ago

https://docs.microsoft.com/en-us/azure/iot-dps/how-to-use-custom-allocation-policies?tabs=windows

upvoted 2 times

⊟ **tita_tovenaar** 4 years ago

you’re right. The load balancer is of second order. Geogr. comes first. Only custom config can do this.

upvoted 4 times

⊟ **berkejf** Highly Voted 👍 4 years, 1 month ago

answer should be D. it require to base registration id and geographic location to assign device to IoT hub. It only can be done by Customised Function App.

upvoted 7 times

⊟ **IMARRA** Most Recent ⊙ 1 year, 11 months ago

Selected Answer: D

"Static configuration: devices are provisioned to a single IoT hub, which must be specified on the enrollment." So Answer should be D.

https://learn.microsoft.com/en-us/azure/iot-dps/how-to-use-allocation-policies#understand-allocation-policies

upvoted 1 times

⊟ **feyey** 2 years, 2 months ago

Selected Answer: D

"Custom (Use Azure Function)" allocation policy option.

This option allows you to use a custom Azure Function to dynamically allocate devices to IoT hubs based on your specific criteria, including the device's registration ID and geographic location. You can write a function that takes the device's metadata as input and returns the IoT hub endpoint that the device should be provisioned to. This enables you to apply more complex allocation logic beyond the standard allocation policy options provided by Azure IoT Hub.

Using the custom allocation policy, you can implement a load balancing strategy that distributes devices evenly between IoT hubs in the same geographic location, based on the current load and availability of each hub. This can help ensure that the devices are distributed efficiently and prevent overloading of any particular IoT hub.

Therefore, the correct answer is option D: Custom (Use Azure Function).

upvoted 1 times

⊟ **IMARRA** 2 years, 11 months ago

why not Evenly weighted distribution?

upvoted 1 times

⊟ **exnaniantwort** 2 years, 6 months ago

It misses the requirement "The load between the IoT hubs in the same geographic location must be balanced".

upvoted 2 times

⊟ **zb99** 2 years, 11 months ago

Selected Answer: D

Only way to include ID in allocation logic is with a custom Function.

upvoted 1 times

⊟ **Plee** 3 years, 3 months ago

Selected Answer: D

IMO answer should be D

upvoted 1 times

⊟ **liberty123** 3 years, 3 months ago

Selected Answer: A

I see Static configuration is not wrong, it works!

upvoted 1 times

⊟ **coramella** 3 years, 5 months ago

Selected Answer: D

“ This article demonstrates a custom allocation policy using an Azure Function written in C#. Two new IoT hubs are created representing a Contoso Toasters Division and a Contoso Heat Pumps Division. Devices requesting provisioning must have a registration ID with one of the following suffixes to be accepted for provisioning:
-contoso-tstrsd-007: Contoso Toasters Division
-contoso-hpsd-088: Contoso Heat Pumps Division
The devices will be provisioned based on one of these required suffixes on the registration ID.” https://docs.microsoft.com/en-us/azure/iot-dps/how-to-use-custom-allocation-policies?tabs=windows

upvoted 1 times

⊟ **d0bermannn** 3 years, 6 months ago

Selected Answer: D

D looks ok:
https://docs.microsoft.com/en-us/azure/iot-dps/how-to-use-custom-allocation-policies?tabs=windows

upvoted 2 times

⊟ **Stephan99** 4 years, 1 month ago

It states "The load between the IoT hubs in the same geographic location must be balanced."
If geographic location is the same, I suppose that the latency is also the same.
Why not select "Evenly weighted distribution"?

upvoted 5 times

⊟ **JeeBi** 4 years, 1 month ago

It states "The load between the IoT hubs in the same geographic location must be balanced." This would not be the case when using lowest latency.
So I would choose A.

upvoted 1 times

ADC88 4 years, 2 months ago

I think should be the answer B: Lowest Latency to cover the requirement about geographic location

upvoted 1 times

> **tita_tovenaar** 4 years ago
>
> no, lowest latency would push devices to the closest hub. thats not necessarily an equal distribution between hubs.
>
> upvoted 4 times

Question #16 *Topic 2*

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

Suggested Answer: *A*

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

Community vote distribution

A (100%)

⊟ **Miko1** Highly Voted 4 years, 2 months ago

corrrrrect

upvoted 9 times

⊟ **coramella** Most Recent 3 years, 5 months ago

Selected Answer: A

" must be updated by the physical IoT device" so A is correct.

upvoted 3 times

Question #17 *Topic 2*

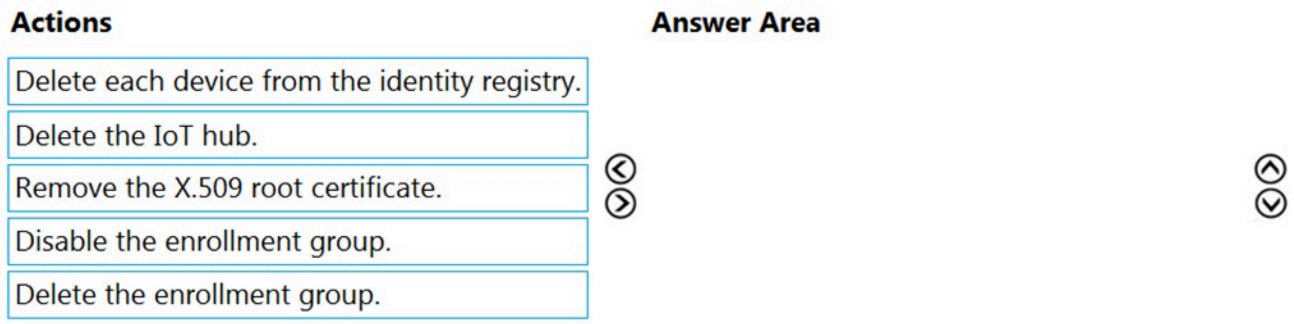
DRAG DROP -

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 four 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.

Select and Place:

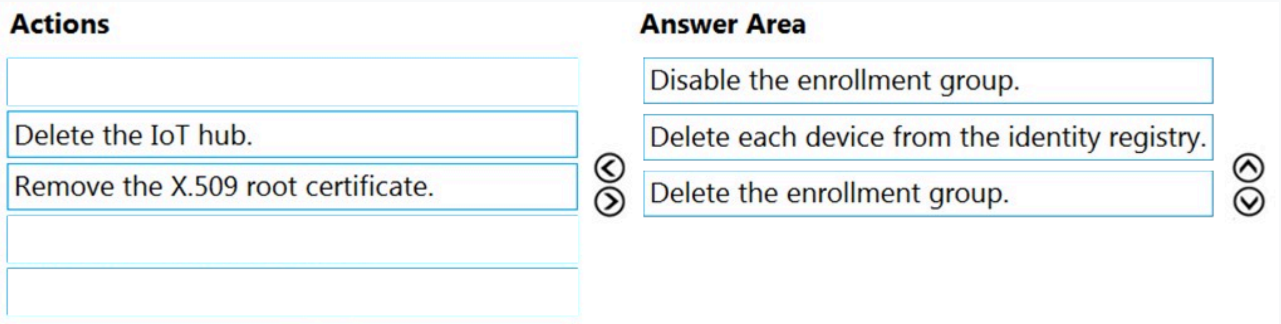


Actions

- Delete each device from the identity registry.
- Delete the IoT hub.
- Remove the X.509 root certificate.
- Disable the enrollment group.
- Delete the enrollment group.

Answer Area

Suggested Answer:



Actions

- Delete the IoT hub.
- Remove the X.509 root certificate.

Answer Area

- Disable the enrollment group.
- Delete each device from the identity registry.
- Delete the enrollment group.

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

sirtomash Highly Voted 👍 4 years, 2 months ago

The answer is fully correct!

upvoted 10 times

Question #18 *Topic 2*

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)

Suggested Answer: *ABD*
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

Community vote distribution

ABD (100%)

⊟ **tita_tovenaar** Highly Voted 4 years ago
careful!
Dice is technically a good answer too, see for example https://azure.microsoft.com/en-us/blog/azure-iot-supports-new-security-hardware-to-strengthen-iot-security/

since Dice is new, not many vendors are supported yet. as such, it is *not yet* a complete solution
upvoted 5 times

⊟ **nqthien041292** Most Recent 3 years, 3 months ago
Selected Answer: ABD
Vote ABD
upvoted 3 times

Question #19 *Topic 2*

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

Suggested Answer: *A*
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

Ashwinee Highly Voted 4 years, 3 months ago
Each device will require three minutes to process the data and respond. hence twin properties is correct
upvoted 7 times

Momindo Most Recent 2 years, 9 months ago
why would you "send data" using device twin properties, device twins are made for configurations right?
upvoted 1 times

Ouss7 2 years, 11 months ago
twin's desired properties are one-way communication
upvoted 1 times

Question #20 *Topic 2*

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

Suggested Answer: *B*
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

trickerk Highly Voted 👍 3 years, 10 months ago
Fleet of devices has minimum 2 devices (it's obivious), thus 2x3min = 6mi. Direct methods has maximum 5 minutes of time out period.
So given answer is correct: No.
upvoted 5 times

Ouss7 2 years, 11 months ago
A Direct method communicate with one device , so either we have 1 or 1000 devices it doesn't matter, it doesn't impact the timeout of one device which is maximum 5 mins
upvoted 2 times

Kamilelo Most Recent ⏲ 2 years, 5 months ago
IMHO anser is B) NO
a Direct Method call requires HTTPS call and here there are only MQTT and AMQP
"Direct methods are HTTPS-only from the cloud side and MQTT, AMQP, MQTT over WebSockets, or AMQP over WebSockets from the device side."
https://learn.microsoft.com/en-us/azure/iot-hub/iot-hub-devguide-direct-methods
upvoted 2 times

Jan91 2 years, 3 months ago
Direct methods could also be over AMQP or MQTT, it is stated in the link you provided
upvoted 1 times

Robert12345Robert 2 years, 8 months ago
The timeout is per device:
https://learn.microsoft.com/en-us/azure/iot-hub/iot-hub-devguide-direct-methods
So A.
upvoted 2 times

dinesh_tng 4 years ago
Direct methods are synchronous and either succeed or fail after the timeout period....WHEREAS QUESTION SAYS - "each device must respond"
upvoted 4 times

berkejf 4 years, 1 month ago
Answer should be Yes. Direct method can set timeout up to 300 sec which allow device to response in 3 mins.
upvoted 2 times

⊟ **borondy** 4 years, 3 months ago

Are we sure? The Direct Method has a maximum 300s method timeout. So it should work in concept.

upvoted 3 times

⊟ **dinesh_tng** 4 years ago

Since there are "fleet of IoT devices", this count can be huge....Direct Meathod is good for immediate actions and on limited number of devices.

upvoted 5 times

⊟ **trickerk** 3 years, 10 months ago

Fleet of devices has minimum 2 devices (it's obivious), thus 2x3min = 6mi. Direct methods has maximum 5 minutes of time out period.

So given answer is correct: No.

upvoted 1 times

Question #21 *Topic 2*

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

Suggested Answer: *B*
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

jiaher Highly Voted 👍 3 years, 12 months ago
Cloud to Device messages are one-way only so answer should be No.
upvoted 7 times

Question #22 *Topic 2*

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

After you answer a question in this 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

Suggested Answer: *B*

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

Community vote distribution

B (100%)

trahd 2 years, 2 months ago

Selected Answer: B

B is correct

upvoted 2 times

moni9367 2 years, 3 months ago

B is ok

upvoted 2 times

Joey_Chou 3 years, 1 month ago

B is correct.

upvoted 3 times

d0bermannn 3 years, 6 months ago

Selected Answer: B

disconnect the Device Provisioning Service from the IoT hub do not prevent device already provisioned from connecting to the iotHub

upvoted 3 times

Question #23 *Topic 2*

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

After you answer a question in this 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

Suggested Answer: *B*

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/

Community vote distribution

B (100%)

mk07 Highly Voted 5 years ago

shouldn't this be device tag? it says changes should not propagate to device.

upvoted 27 times

getazusername 4 years, 7 months ago

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

upvoted 4 times

EyeeyeeyeeyeeyeeyeeyeeyeSPIDER Highly Voted 4 years, 11 months ago

Correct, the answer should be YES here

upvoted 11 times

coramella Most Recent 3 years, 5 months ago

Selected Answer: B

Desired props are propagated to device. No is the answer

upvoted 4 times

JeeBi 4 years, 1 month ago

https://docs.microsoft.com/en-us/learn/modules/remotely-monitor-devices-with-azure-iot-hub/7-write-code-device-twins

=> Tags: information on the device that isn't visible to the device.

Changes should not be sent to the device, so I would go with tags and answer B 'No'

upvoted 4 times

Ashwinee 4 years, 3 months ago

i think ans is No. tags should be used

upvoted 5 times

sballmer 4 years, 5 months ago

I think No is the right answer, in my opinion "adding a tag to the device twin" means adding a data to the desired property, which will be transmited to the device and we must prevent changes being propagated to the device.

The solution is to add a tag to the device, not to the device twin.

upvoted 2 times

pbleep 4 years, 5 months ago

Tags are not desired properties and are not seen by the device and therefore would not trigger a response. Desired properties themselves are seen by devices and would trigger a response

upvoted 4 times

Lihz 4 years, 7 months ago

I think should be A.

upvoted 1 times

ipindado2020 4 years, 7 months ago

A is the way

upvoted 1 times

srama79 4 years, 7 months ago

Answer should be YES ..

upvoted 2 times

spartons12345 4 years, 8 months ago

I think the answer should be Yes

upvoted 3 times

Question #24 *Topic 2*

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.

Suggested Answer: *D*
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/

⊟ **ipindado2020** Highly Voted 4 years, 7 months ago
D is ok
upvoted 9 times

⊟ **exnaniantwort** Most Recent 2 years, 6 months ago
https://learn.microsoft.com/en-us/azure/iot-dps/how-to-reprovision
upvoted 2 times

Question #25 *Topic 2*

You have an Azure IoT Central solution that includes multiple IoT devices. The devices report temperature, humidity, and pressure.
You need to export the sensor data captured during a 48-hour period as a CSV file.
What should you use in IoT Central?

A. Devices

B. Jobs

C. Device groups

D. Analytics

Suggested Answer: *D*
Azure IoT Central provides rich analytics capabilities to analyze historical trends and correlate telemetry from your devices. To get started, select Analytics on the left pane.
The analytics user interface has three main components:
✑ Data configuration panel: On the configuration panel, select the device group for which you want to analyze the data. Next, select the telemetry that you want to analyze and select the aggregation method for each telemetry. The Group By control helps to group the data by using device properties as dimensions.
✑ Time control: Use the time control to select the duration for which you want to analyze the data.
✑ Chart control: The chart control visualizes the data as a line chart.
Reference:
https://docs.microsoft.com/en-us/azure/iot-central/core/howto-create-analytics

Community vote distribution

D (100%)

⊟ **coffecold** 2 years, 6 months ago

Nowadays it is called data-explorer according to the link

upvoted 1 times

⊟ **coffecold** 2 years, 6 months ago

Under "Analyse", not "Analytics"

upvoted 2 times

⊟ **d0bermannn** 3 years, 6 months ago

Selected Answer: D

D looks correct as link provided said

upvoted 4 times

Question #26 Topic 2

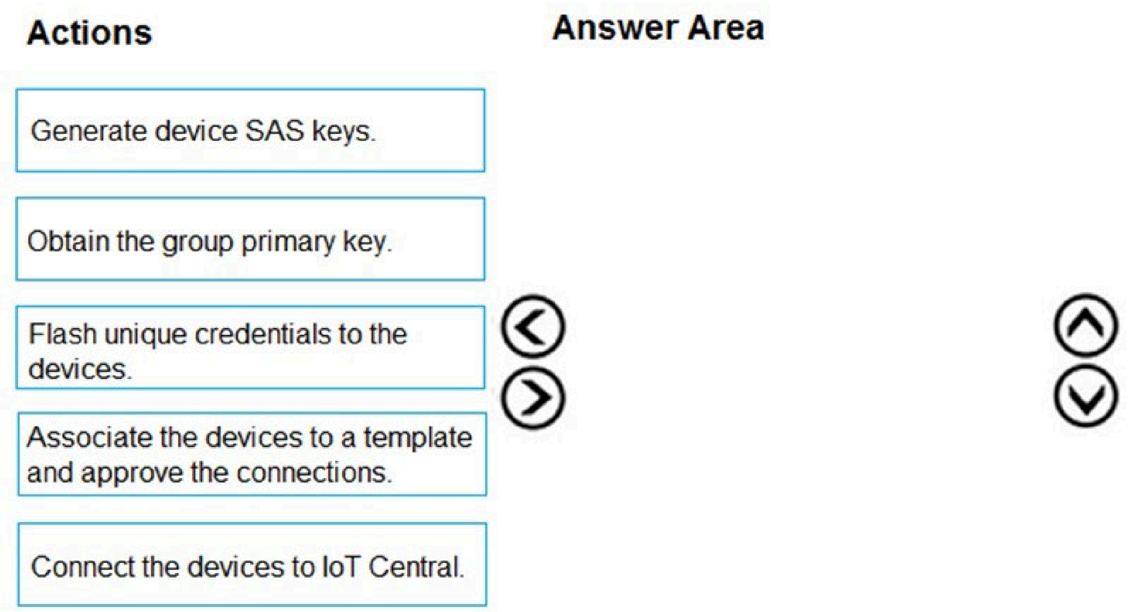
DRAG DROP -

You have an Azure IoT Central application.

You need to connect IoT devices that use SAS tokens to the application without first registering the devices.

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

Select and Place:



Actions

Generate device SAS keys.

Obtain the group primary key.

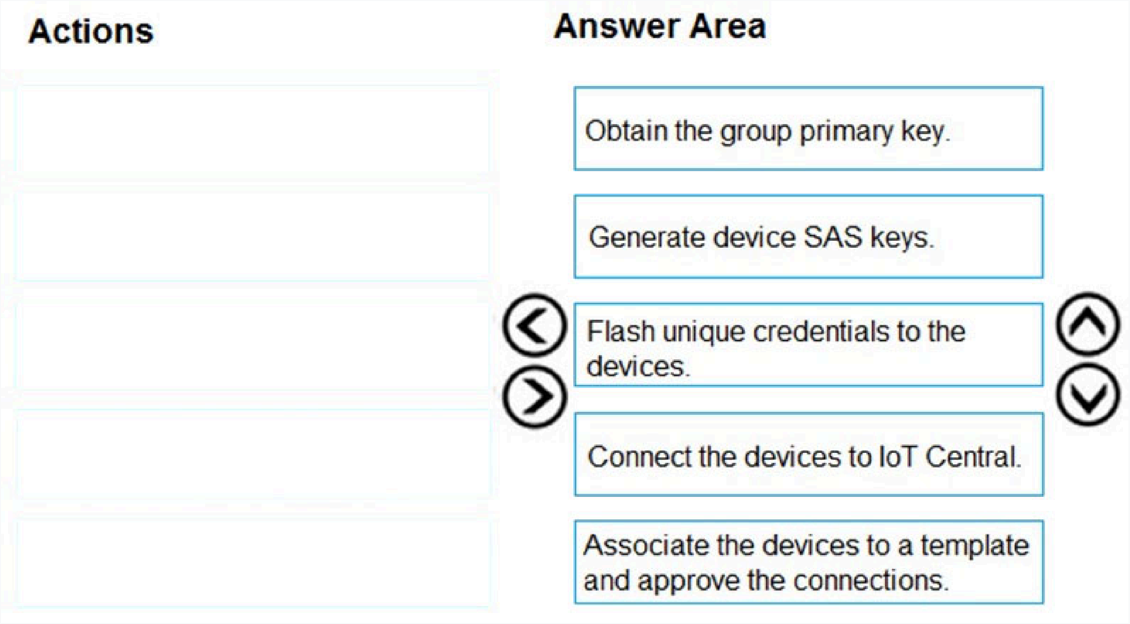
Flash unique credentials to the devices.

Associate the devices to a template and approve the connections.

Connect the devices to IoT Central.

Answer Area

Suggested Answer:



Actions

Answer Area

Obtain the group primary key.

Generate device SAS keys.

Flash unique credentials to the devices.

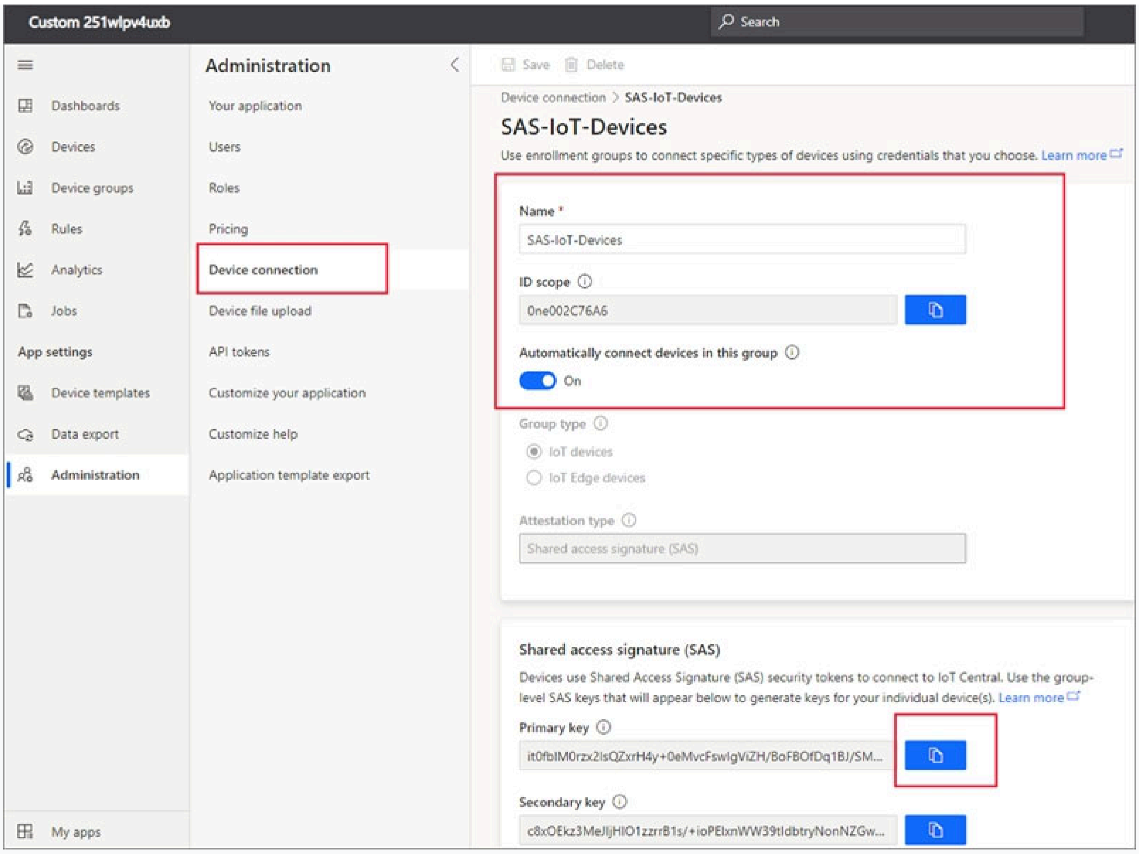
Connect the devices to IoT Central.

Associate the devices to a template and approve the connections.

Automatically register devices that use SAS tokens:

Step 1: Obtain the group primary key

1. Copy the group primary key from the SAS-IoT-Devices enrollment group:



Device connection > SAS-IoT-Devices

SAS-IoT-Devices

Use enrollment groups to connect specific types of devices using credentials that you choose. Learn more

Name * SAS-IoT-Devices

ID scope 0ne002C76A6

Automatically connect devices in this group: On

Group type
● IoT devices
○ IoT Edge devices

Attestation type: Shared access signature (SAS)

Shared access signature (SAS)

Devices use Shared Access Signature (SAS) security tokens to connect to IoT Central. Use the group-level SAS keys that will appear below to generate keys for your individual device(s). Learn more

Primary key: it0fbIM0rzx2lsQZxrH4y+0eMvcFswlgViZH/BoFBOfDq1BJ/SM...

Secondary key: c8xOEkz3MeJljHIO1zzrrB1s/+ioPElxnWW39tIdbtryNonNZGw...

Step 2: Generate device SAS Keys.

2. Use the az iot central device compute-device-key command to generate the device SAS keys. Use the group primary key from the previous step.

Step 3: Flash unique credentials to the devices.

3. As an OEM, flash each device with the device ID, the generated device SAS key, and the application ID scope value. The device code should also send the model ID of the device model it implements.

Step 4: Connect the devices to IoT Central

4. When you switch on a device, it first connects to DPS to retrieve its IoT Central registration information.

5. The device uses the information from DPS to connect to, and register with, your IoT Central application.

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

The IoT Central application uses the model ID sent by the device to associate the registered device with a device template.

Reference:

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

⊟ **d0bermannn** Highly Voted 3 years, 6 months ago

repeated question with more reliable phrases, but thanks anyway, answer is correct

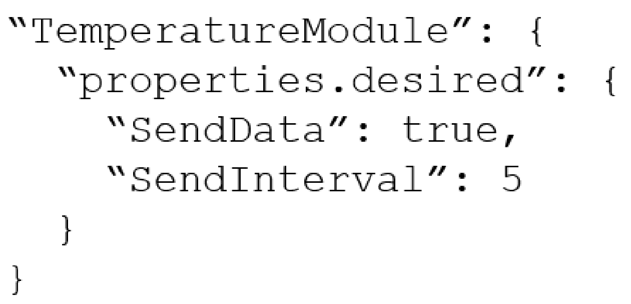
upvoted 6 times

HOTSPOT

-

You have an Azure IoT Edge automatic deployment named D1 that deploys a temperature module to five IoT Edge devices.

D1 has a deployment priority of 10 and the following module configuration.

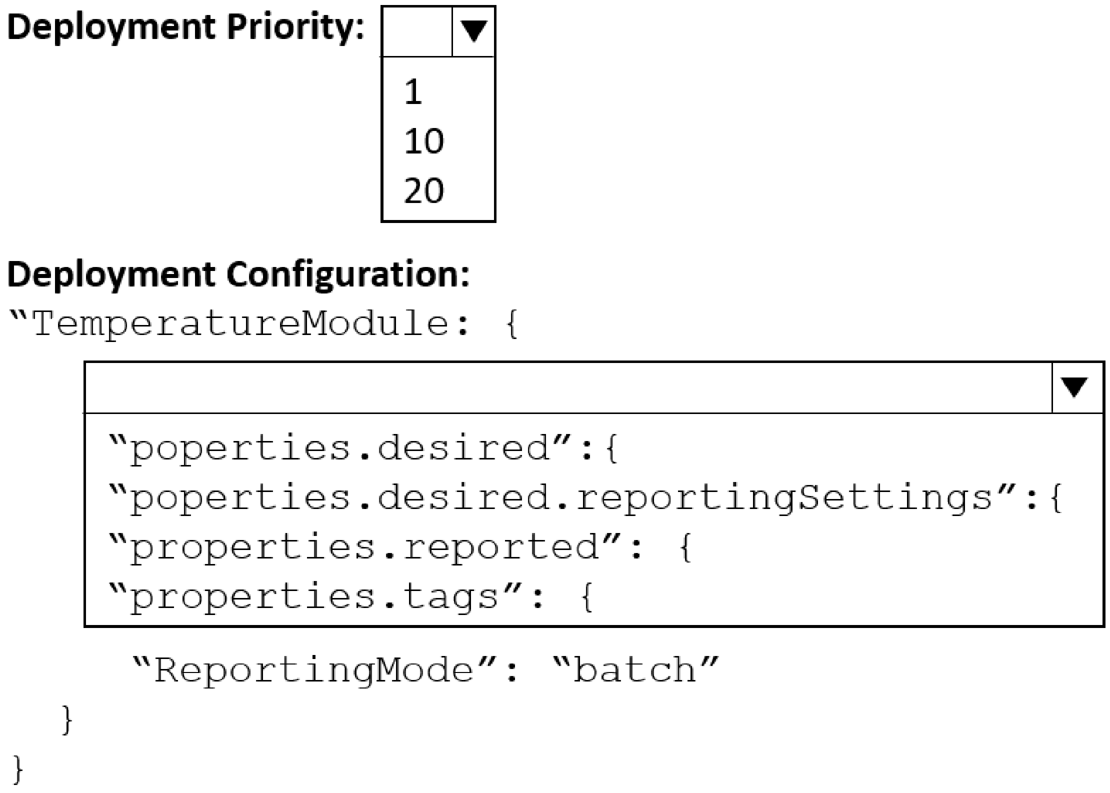


```
“TemperatureModule”: {
  “properties.desired”: {
    “SendData”: true,
    “SendInterval”: 5
  }
}
```

You need to create a new layered deployment that will add a new twin property named ReportingMode. The new deployment must not overwrite the existing module configurations set by D1.

How should you configure the deployment? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.



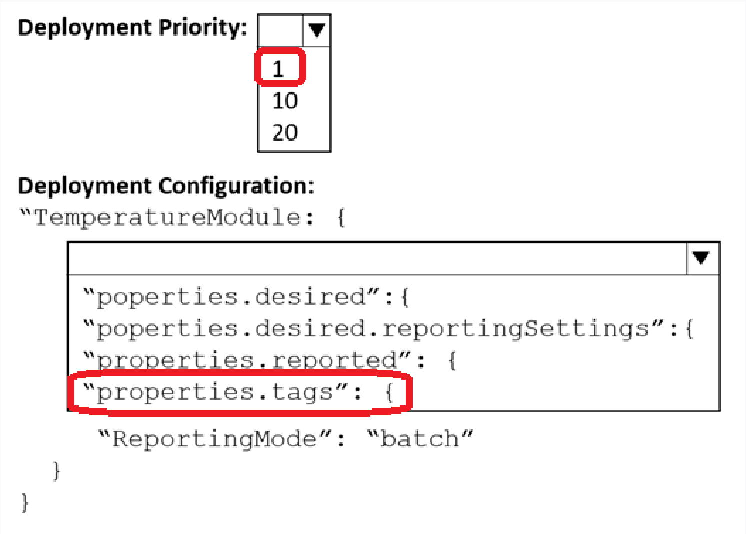
Deployment Priority:

- 1
- 10
- 20

Deployment Configuration:

```
“TemperatureModule: {
  [▼]
    “poperties.desired”:{
    “poperties.desired.reportingSettings”:{
    “properties.reported”: {
    “properties.tags”: {
     “ReportingMode”: “batch”
  }
}
```

Suggested Answer:



Deployment Priority: ▼

[✓ 1]
10
20

Deployment Configuration:

```
"TemperatureModule: {
    ▼
    "poperties.desired":{
    "poperties.desired.reportingSettings":{
    "properties.reported": {
    [✓ "properties.tags": {]
      "ReportingMode": "batch"
  }
}
```

⊟ **Jan91** 2 years, 3 months ago

Answer is wrong. I should say priority 20, because it should not overwrite and properties.desired.reportingSettings.

upvoted 1 times

⊟ **slafcemafce** 2 years, 2 months ago

Higher value is higher priority.
https://learn.microsoft.com/en-us/azure/iot-edge/module-deployment-monitoring?view=iotedge-1.4#priority

So I would go with priority 1

upvoted 4 times

Question #29 *Topic 2*

You have an Azure IoT solution that includes an Azure IoT hub and a Device Provisioning Service instance.

Several enrolled devices are stolen.

You need to prevent the stolen devices from connecting to the IoT solution. The solution must prevent the devices from re-enrollment and must be implemented as soon as possible.

What should you do?

A. Delete the devices from the IoT hub.

B. Delete the device enrollments from the Device Provisioning Service.

C. Disable the devices in the IoT hub and delete from the IoT hub.

D. Disable the device enrollments in the Device Provisioning Service and delete the devices from the IoT hub.

Suggested Answer: *B*

hotwheelsinsf 2 years, 1 month ago

shouldn't the answer be d. Disable and then delete?

upvoted 3 times

Badoic 2 years, 1 month ago

The requirement is "prevent the devices from re-enrollment" in DSP.
Just "disabling" the enrollment in DSP does not meet this req., the enrollment in DSP has to be deleted.

upvoted 2 times

Question #31 *Topic 2*

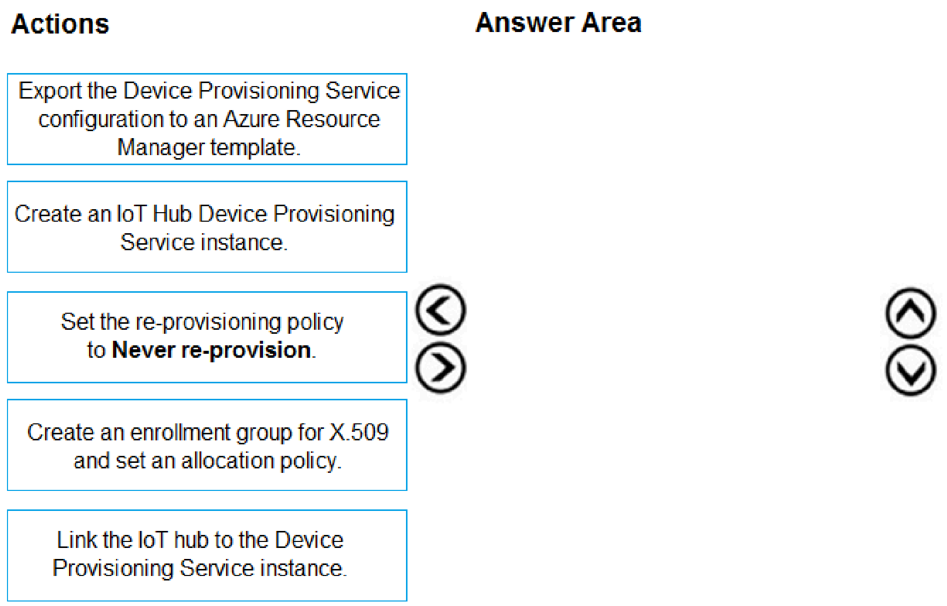
DRAG DROP

-

You have an Azure IoT hub.

You need to deploy a Device Provisioning Service instance that uses X.509 attestation to support new IoT devices.

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

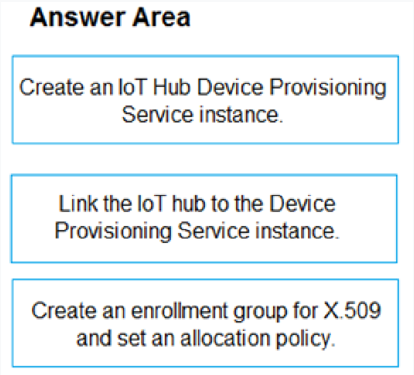


Actions

- Export the Device Provisioning Service configuration to an Azure Resource Manager template.
- Create an IoT Hub Device Provisioning Service instance.
- Set the re-provisioning policy to **Never re-provision**.
- Create an enrollment group for X.509 and set an allocation policy.
- Link the IoT hub to the Device Provisioning Service instance.

Answer Area

Suggested Answer:



Answer Area

1. Create an IoT Hub Device Provisioning Service instance.
2. Link the IoT hub to the Device Provisioning Service instance.
3. Create an enrollment group for X.509 and set an allocation policy.

⊟ **Badoic** 2 years, 1 month ago

Correct!

upvoted 2 times

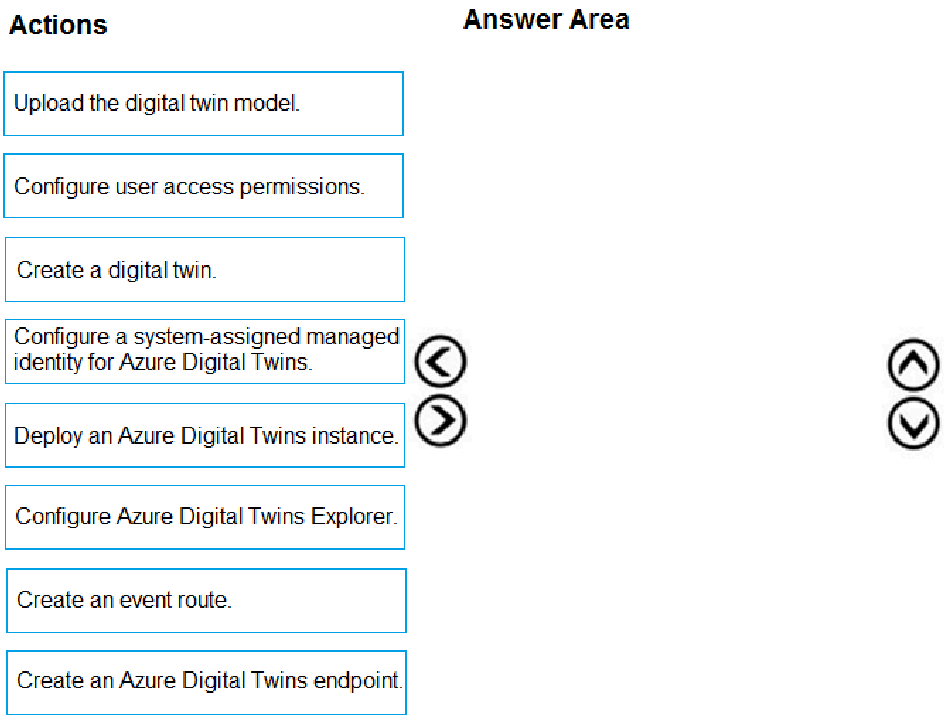
Question #33 *Topic 2*

DRAG DROP

-

You need to configure a digital twin to accept device telemetry data from the IoT hub.

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.



Actions

Upload the digital twin model.

Configure user access permissions.

Create a digital twin.

Configure a system-assigned managed identity for Azure Digital Twins.

Deploy an Azure Digital Twins instance.

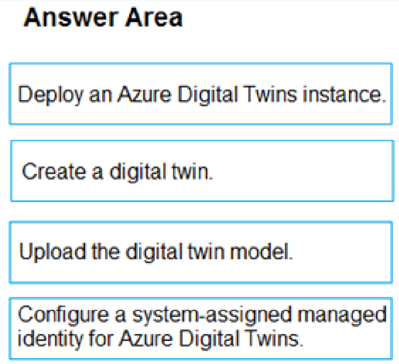
Configure Azure Digital Twins Explorer.

Create an event route.

Create an Azure Digital Twins endpoint.

Answer Area

Suggested Answer:



Answer Area

Deploy an Azure Digital Twins instance.

Create a digital twin.

Upload the digital twin model.

Configure a system-assigned managed identity for Azure Digital Twins.

geekgirl007 2 years ago

you have to upload the model before you can create a digital twin so correct order is below;

Deploy an Azure Digital Twins instance

Upload the digital twin model

Create a digital twin

Configure a system-assigned managed identity for Azure Digital Twins

upvoted 2 times

Question #34 *Topic 2*

You have an Azure IoT Central solution.

You need to verify that telemetry messages from devices arrive to IoT Central.

What should you use?

A. the Azure IoT explorer

B. the az command in Azure CLI

C. Azure Service Bus Explorer

D. the Azure IoT Tools for VS Code extension pack

Suggested Answer: *B*

alfespa17 2 years, 1 month ago

Correct answer is Azure IOT Explorer
https://learn.microsoft.com/en-us/azure/iot-develop/quickstart-send-telemetry-iot-hub?pivots=programming-language-java#view-telemetry

upvoted 3 times

Question #37 *Topic 2*

HOTSPOT

-

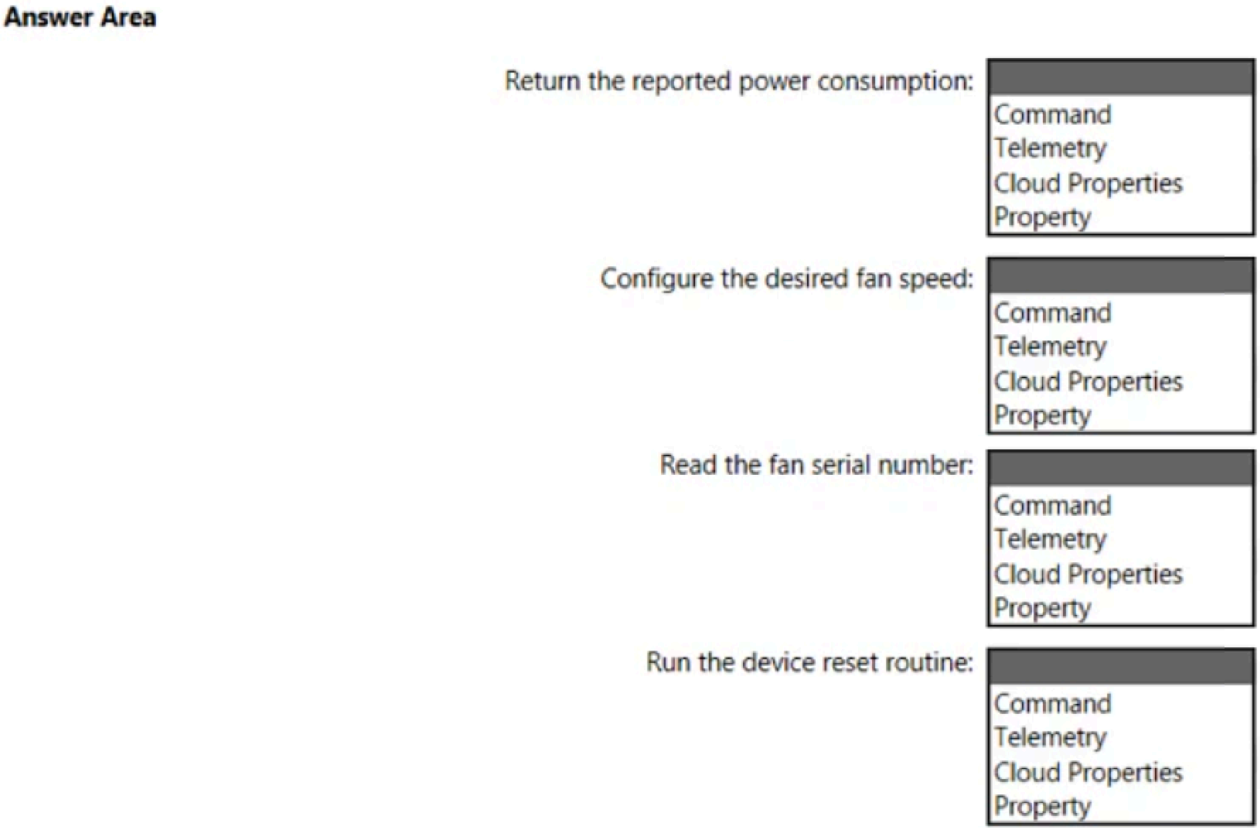
You have an Azure IoT Central application that has a custom device template.

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.



Answer Area

Return the reported power consumption:

Command
Telemetry
Cloud Properties
Property

Configure the desired fan speed:

Command
Telemetry
Cloud Properties
Property

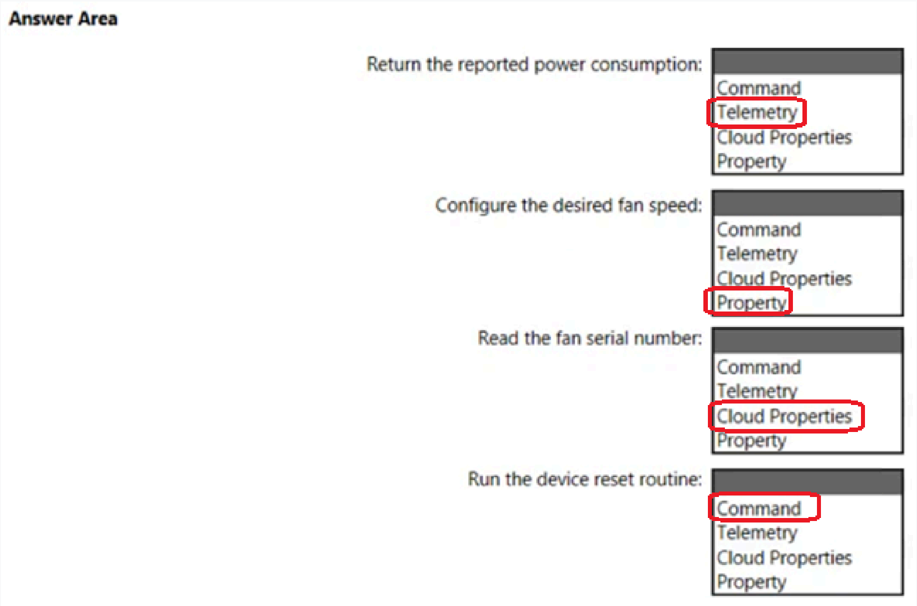
Read the fan serial number:

Command
Telemetry
Cloud Properties
Property

Run the device reset routine:

Command
Telemetry
Cloud Properties
Property

Suggested Answer:



Answer Area

Return the reported power consumption:
Command
[✓ Telemetry]
Cloud Properties
Property

Configure the desired fan speed:
Command
Telemetry
Cloud Properties
[✓ Property]

Read the fan serial number:
Command
Telemetry
[✓ Cloud Properties]
Property

Run the device reset routine:
[✓ Command]
Telemetry
Cloud Properties
Property

smudo1965 1 year, 11 months ago

Following this link https://learn.microsoft.com/en-us/azure/iot-central/core/howto-use-properties cloud properties are never exchanged with the device therefore answer is property

upvoted 1 times

hotwheelsinsf 2 years, 2 months ago

this question is the same as topic 2 question 3, but with different options... why don't you people try to be consistent in your information??????? You are really not helping your paying customers with this confusion...

upvoted 3 times

Question #38 *Topic 2*

You have an Azure IoT hub that uses a Device Provisioning Service (DPS) instance.

For 100 legacy devices, you plan to create a new device enrollment that will use symmetric key attestation. The solution must minimize administrative effort.

What should you use to derive the device key?

A. the subscription ID

B. the IoT hub name

C. the group master key

D. the primary key of the DPS shared access policy

Suggested Answer: *D*

trahd 2 years, 2 months ago

The multiple-choice options in the question are confusing.
Also, the question says "what should you do to derive the device key?". Better wording would be, "what should you do to derive the device key for each device?".

I will answer it in real world terms:
Since we are going for symmetric key attestation and minimizing administrative effort, it's best to use a symmetric key enrollment group.

In that case, for each device, "the derived device key is a hash of the device's registration ID and is computed using the symmetric key of the enrollment group."

Reference:
https://learn.microsoft.com/en-gb/azure/iot-dps/concepts-symmetric-key-attestation?tabs=windows#group-enrollments

upvoted 1 times

Question #40 *Topic 2*

You have an Azure IoT Hub deployment.

You plan to deploy 1,000 IoT devices that will have 1 MB of RAM. The devices will be deployed behind firewalls that block port 443.

You need to configure the communication protocol for the devices. The solution must ensure that each device uses unique credentials.

Which protocol should you use?

A. AMQP

B. MQTT over WebSockets

C. MQTT

D. AMQP over WebSockets

Suggested Answer: *C*

Yameo 2 years ago

It shouldn't be B? MQTT isn't restricted to 8883? I'm asking as I'm totally new to IoT.

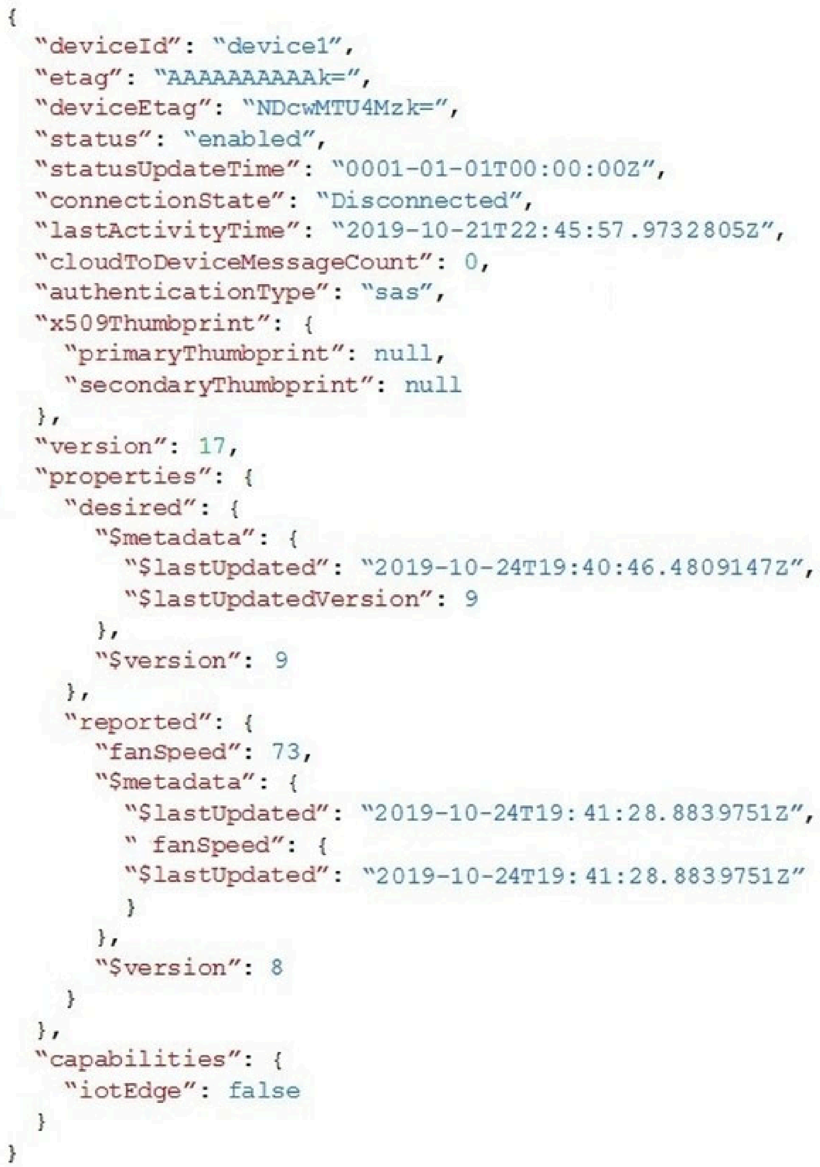
upvoted 1 times

> **Yameo** 2 years ago
>
> Ok, I misread, the port 443 is blocked, not the available one.
>
> upvoted 1 times

Question #1 Topic 3

HOTSPOT -

You have the following device twin for the IoT device.

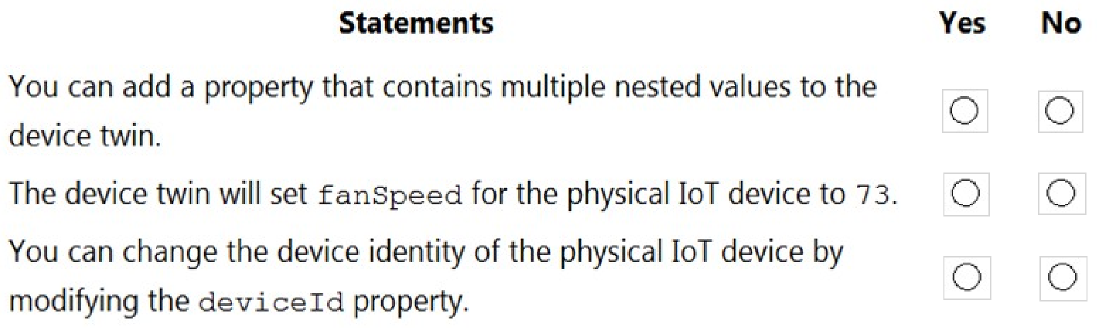


```
{
  “deviceId”: “device1”,
  “etag”: “AAAAAAAAAAk=”,
  “deviceEtag”: “NDcwMTU4Mzk=”,
  “status”: “enabled”,
  “statusUpdateTime”: “0001-01-01T00:00:00Z”,
  “connectionState”: “Disconnected”,
  “lastActivityTime”: “2019-10-21T22:45:57.9732805Z”,
  “cloudToDeviceMessageCount”: 0,
  “authenticationType”: “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.

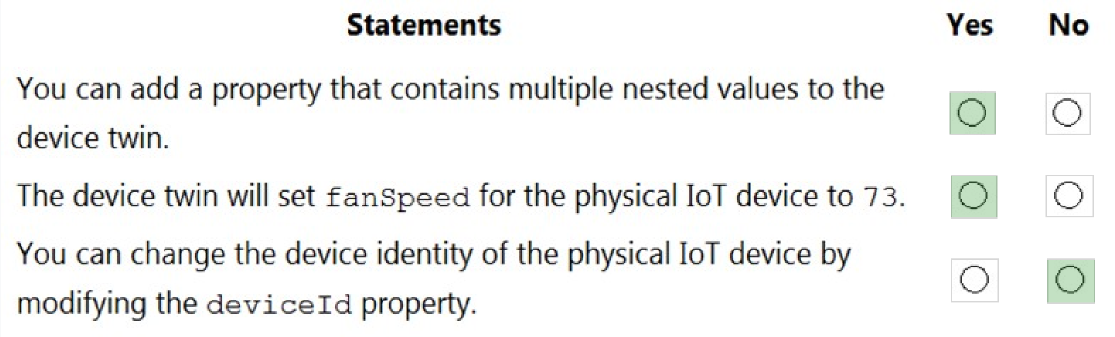
NOTE: Each correct selection is worth one point.

Hot Area:



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.	○	○

Suggested 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.	○	[✓ ○]

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

dariuszdbr Highly Voted 4 years, 3 months ago
I think that the correct answers should be:
Yes - Although, The maximum depth of JSON objects in tags, desired properties, and reported properties is 10 (https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-devguide-device-twins#tags-and-properties-format)
No - The device actually report that it's fanSpeed is equal to 73, so it will not be set, it's the current value reported to cloud
No - The deviceID property is read only.
upvoted 46 times

trickerk 3 years, 10 months ago
You're alright:
Yes https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-devguide-device-twins
No Reported is device-to-cloud (read) https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-devguide-device-twins
No https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-devguide-identity-registry
upvoted 6 times

harithzainudin 3 years, 7 months ago
indeed, you are correct. i agree
upvoted 5 times

liberty123 Most Recent 3 years, 3 months ago
Agree with
Yes
No
No
upvoted 2 times

d0bermannn 3 years, 5 months ago
ynn is correct fo sure
upvoted 2 times

Passionate 3 years, 10 months ago
This question was on test 16-Aug-2021.
upvoted 3 times

Marusyk 3 years, 7 months ago
and what?
upvoted 2 times

d0bermannn 3 years, 6 months ago
he 's abandoned his test and goes to spam here)
upvoted 5 times

Question #2 *Topic 3*

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

Suggested Answer: *B*

Routes are declared in the $edgeHub desired properties.

Reference:

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

Community vote distribution

B (100%)

d0bermannn 3 years, 6 months ago

Selected Answer: B

B. $edgeHub is correct as we see link provided

upvoted 4 times

Question #3 *Topic 3*

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.

Suggested Answer: *AB*
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

Community vote distribution
BD (100%)

mbn Highly Voted 4 years, 3 months ago
answer is BD
B: "Azure IoT Edge relies on an OCI-compatible container runtime. For production scenarios, we recommended that you use the Moby engine. The Moby engine is the only container engine officially supported with Azure IoT Edge."
https://docs.microsoft.com/en-us/azure/iot-edge/how-to-install-iot-edge?view=iotedge-2018-06#install-a-container-engine

D: "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."
https://docs.microsoft.com/en-us/azure/iot-edge/how-to-install-iot-edge?view=iotedge-2018-06#install-iot-edge

A is wrong because it refers to an update
upvoted 21 times

hotwheelsinsf Most Recent 2 years, 2 months ago
So what is the right answer???
upvoted 1 times

[Removed] 3 years, 1 month ago
I think it should be A,D?
https://docs.microsoft.com/en-us/azure/iot-edge/how-to-provision-single-device-linux-symmetric?view=iotedge-2020-11&tabs=azure-portal%2Cubuntu
says "install container engine" and "Install the IoT Edge runtime"
upvoted 1 times

[Removed] 3 years, 1 month ago
Sorry, A is talking about update, so I stand with B, D too.
upvoted 1 times

Plee 3 years, 3 months ago
Selected Answer: BD
Agree with BD
upvoted 1 times

liberty123 3 years, 3 months ago

Selected Answer: BD

Agree with BD

upvoted 1 times

d0bermannn 3 years, 6 months ago

Selected Answer: BD

B&D is correct as we see link

https://docs.microsoft.com/en-us/azure/iot-edge/how-to-provision-single-device-linux-symmetric?view=iotedge-2020-11&tabs=azure-cli

upvoted 3 times

kumarneeraj 4 years, 1 month ago

Answer is CD

You need to first deploy the IoT Edge runtime and then install it (initialize). How can you update the runtime without deploying it. The security daemon automatically gets added as a part of IoT Edge runtime.

upvoted 1 times

tita_tovenaar 4 years ago

wrong. powershell is not used for pure linux machines. only for linux containers on windows. as this is a RaspPi, it follows standard linux deployment without powershell:

https://docs.microsoft.com/en-us/azure/iot-edge/how-to-install-iot-edge?view=iotedge-2020-11

upvoted 3 times

Question #4 *Topic 3*

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

Suggested Answer: *BDE*
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

Community vote distribution
BDE (100%)

tita_tovenaar Highly Voted 4 years ago
A is incorrect as it is a simulator.
C is also incorrect because iotedgedev relates to builing your project, it's not deployed to the actual edge device. hence given answers are correct
upvoted 7 times

liberty123 Most Recent 3 years, 3 months ago
Selected Answer: BDE
Agree with BDE
upvoted 1 times

d0bermannn 3 years, 6 months ago
Selected Answer: BDE
BDE looks nice
upvoted 2 times

angelsrp 4 years, 11 months ago
Ans ABE:

https://hub.docker.com/_/microsoft-azureiotedge-agent
https://hub.docker.com/_/microsoft-azureiotedge-hub
https://docs.microsoft.com/en-us/azure/iot-edge/how-to-deploy-modules-portal
upvoted 1 times

angelsrp 4 years, 11 months ago
my bad, given answers are correct
upvoted 10 times

dinesh_tng 4 years, 2 months ago
B&E are there by default on mcr.microsoft.com as Agent and Hub Module. E is required on mycr.azurecr.io as that is private repo.

A is incorrect, as it points to microsoft public repo.
upvoted 3 times

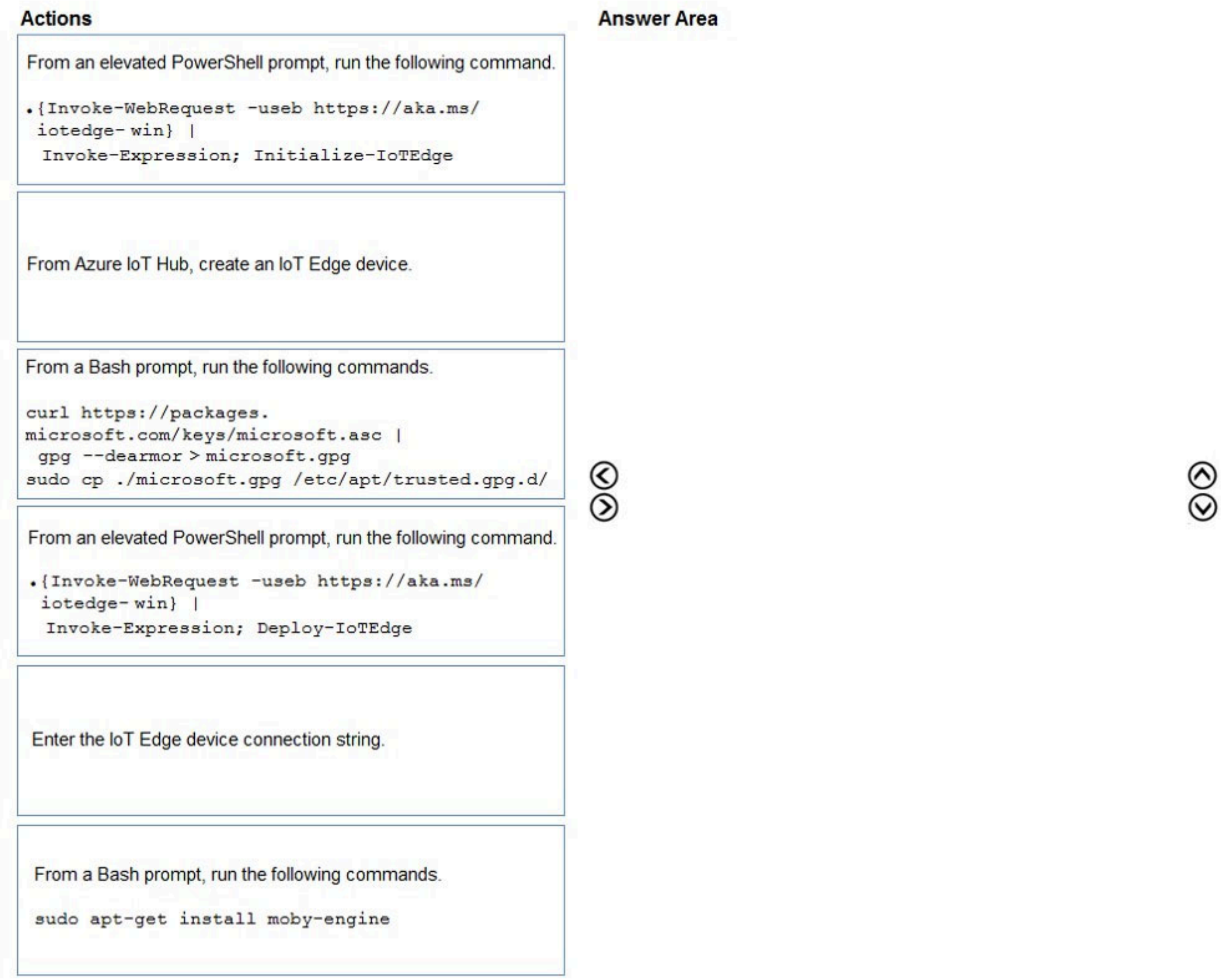
Question #5 Topic 3

DRAG DROP -

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

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

Select and Place:



Actions

From an elevated PowerShell prompt, run the following command.

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

From Azure IoT Hub, create an IoT Edge device.

From a Bash prompt, run the following commands.

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

From an elevated PowerShell prompt, run the following command.

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

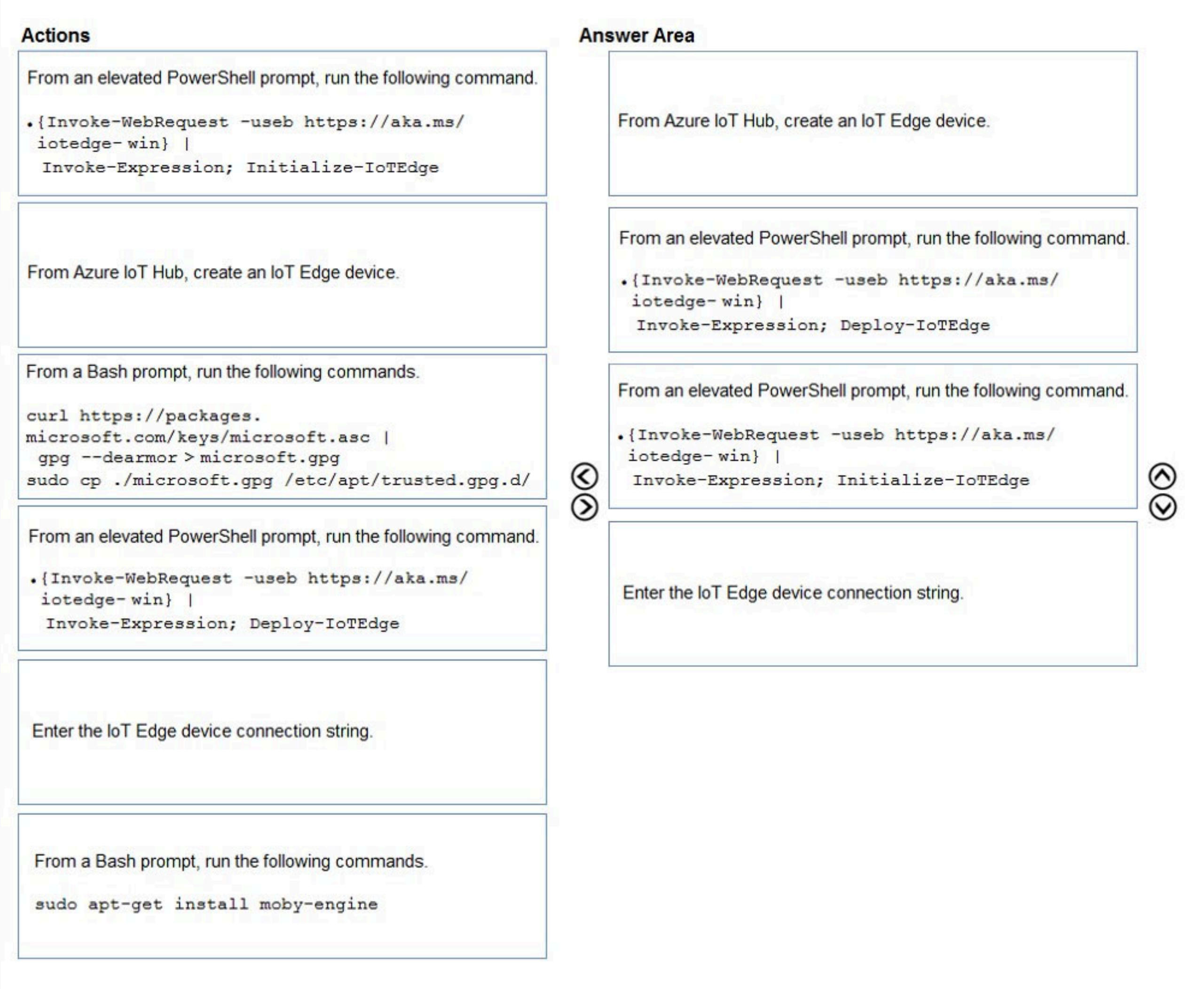
Enter the IoT Edge device connection string.

From a Bash prompt, run the following commands.

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

Answer Area

Suggested Answer:



Actions

From an elevated PowerShell prompt, run the following command.

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

From Azure IoT Hub, create an IoT Edge device.

From a Bash prompt, run the following commands.

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

From an elevated PowerShell prompt, run the following command.

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

Enter the IoT Edge device connection string.

From a Bash prompt, run the following commands.

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

Answer Area

From Azure IoT Hub, create an IoT Edge device.

From an elevated PowerShell prompt, run the following command.

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

From an elevated PowerShell prompt, run the following command.

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

Enter the IoT Edge device connection string.

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

sballmer Highly Voted 4 years, 5 months ago

Completely unnatural to first Deploy and then Initialize, but Answer are correct...

1. Create edge device in IoT HuB
2. Run Deploy Command
3. Run Initialize Command
4. Enter Edge Device Connection String

upvoted 14 times

⊟ **tita_tovenaar** Highly Voted 4 years ago

just to confirm, moby and curl options are applicable for Linux. As this is Windows, answers. are correct.

upvoted 6 times

⊟ **d0bermannn** 3 years, 5 months ago

since pwsh 7+ we can fire it on many *nix boxes)

upvoted 1 times

⊟ **liberty123** Most Recent 3 years, 3 months ago

Correct answer

upvoted 1 times

⊟ **angelsrp** 4 years, 11 months ago

Ans are correct:

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

upvoted 6 times

⊟ **angelsrp** 4 years, 11 months ago

After running the second cmlet you are prompted for the string.

upvoted 4 times

⊟ **getazusername** 4 years, 7 months ago

Thanks for the information. otherwise there would be two possible solutions. accuring your comment, you have to create the device before running the cmdlets

upvoted 3 times

Question #6 *Topic 3*

You have an Azure IoT solution that includes an Azure IoT Hub named Hub1 and an Azure IoT Edge device named Edge1. Edge1 connects to Hub1.
You need to deploy a temperature module to Edge1.
What should you do?

A. From the Azure portal, navigate to Hub1 and select IoT Edge. Select Edge1, and then select Manage Child Devices. From a Bash prompt, run the following command: az iot edge set-modules --device-id Edge1 --hub-name Hub1 --content 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 deploymentMan1.json

Suggested Answer: *D*
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

⊟ **ipindado2020** Highly Voted 4 years, 7 months ago

D is ok

upvoted 10 times

⊟ **HimashuR** Most Recent 3 years, 11 months ago

correct

upvoted 4 times

Question #7 *Topic 3*

DRAG DROP -

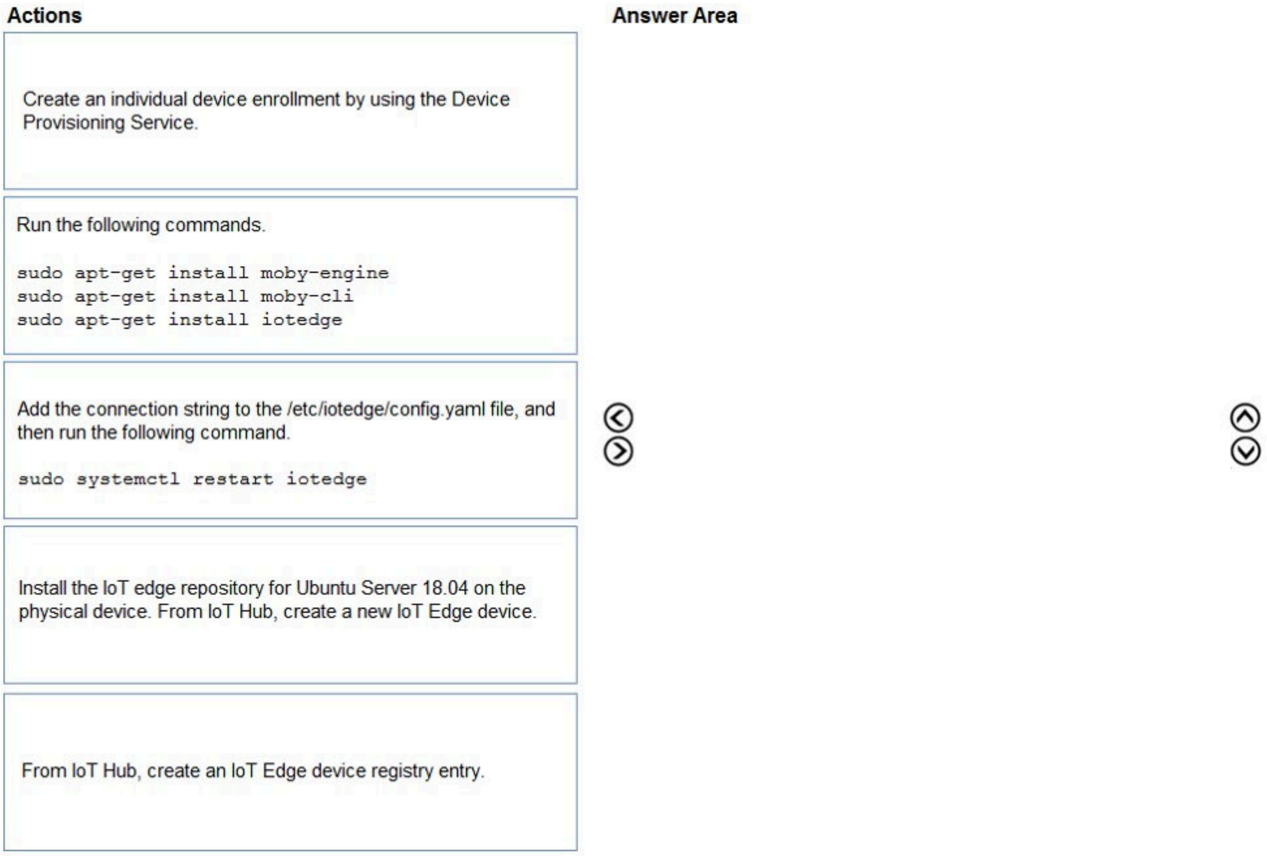
Your company is creating a new camera security system that will use Azure IoT Hub.

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

You need to configure the IoT Edge device.

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

Select and Place:



Actions

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

Run the following commands.

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

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

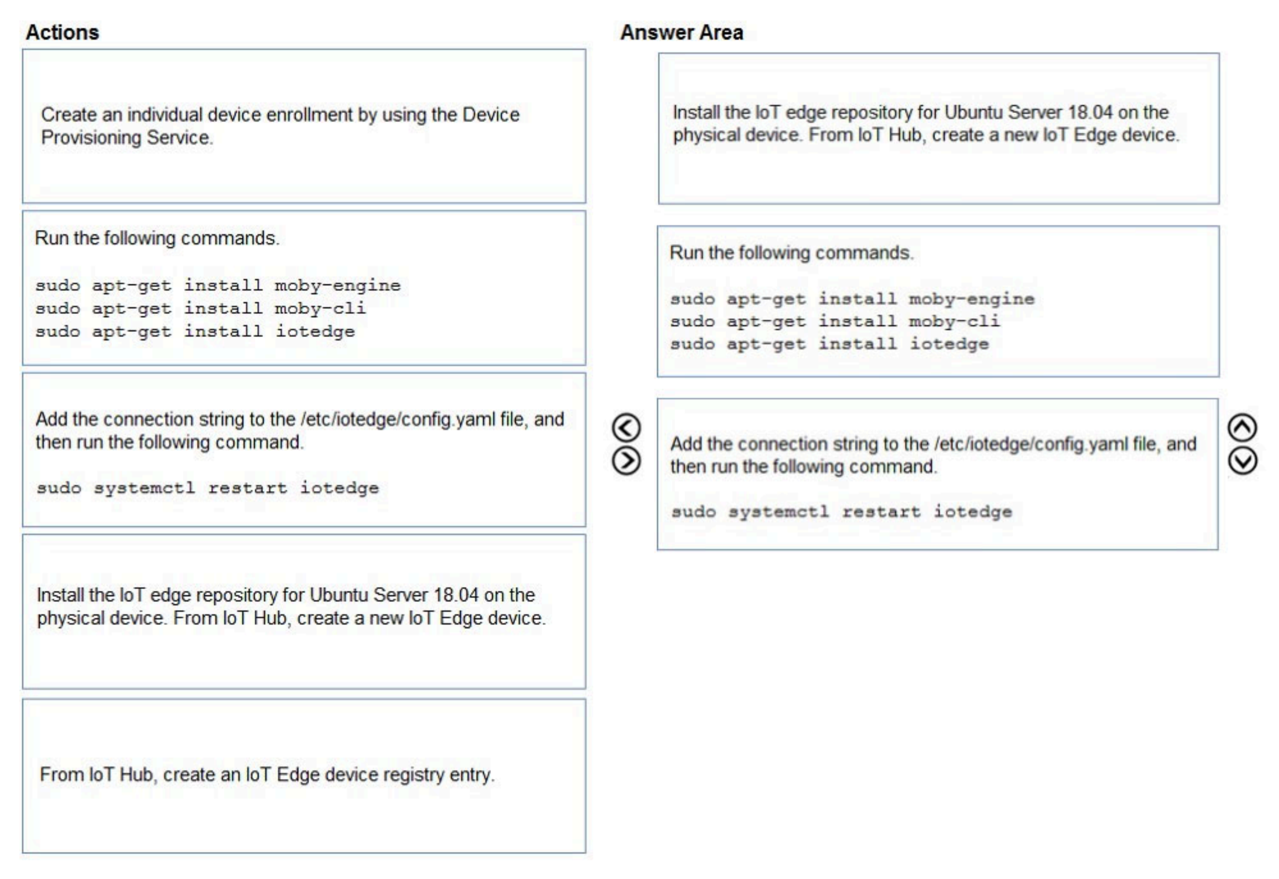
```
sudo systemctl restart iotedge
```

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

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

Answer Area

Suggested Answer:



Actions

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

Run the following commands.

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

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

```
sudo systemctl restart iotedge
```

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

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

Answer Area

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
```

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 commandsג€¦

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

not_a_robot Highly Voted 4 years, 11 months ago

Shouldn't the answer be:

1. Install the repository configuration for Ubuntu Server 18.04, and create an IoT Edge device on the IoT hub.
2. Install the container runtime.

sudo apt-get install moby-engine

sudo apt-get install moby-cli

sudo apt-get install iotedge

3. Add the connection string to config.yaml

upvoted 27 times

angelsrp 4 years, 11 months ago

this is correct

upvoted 6 times

dinesh_tng 4 years, 2 months ago

Step 3 - file name is config.toml, and command is 'sudo iotedge config apply'. These three steps are ok.

upvoted 1 times

ParaExam Highly Voted 4 years, 9 months ago

I till agree with this:

Shouldn't the answer be:

1. Install the repository configuration for Ubuntu Server 18.04, and create an IoT Edge device on the IoT hub.
2. Install the container runtime.

sudo apt-get install moby-engine

sudo apt-get install moby-cli

sudo apt-get install iotedge

3. Add the connection string to config.yaml

upvoted 11 times

certstowinirl 4 years, 9 months ago

This is correct. The official documentation tells us that we first need to install the repository configuration. We cant just 'apt-get install iotedge'.

https://docs.microsoft.com/nl-nl/azure/iot-edge/how-to-install-iot-edge-linux

upvoted 9 times

iotguy Most Recent 4 years, 10 months ago

I think steps 1 & 2 could be in either order

upvoted 3 times

noob115 4 years, 10 months ago

agreed..

upvoted 1 times

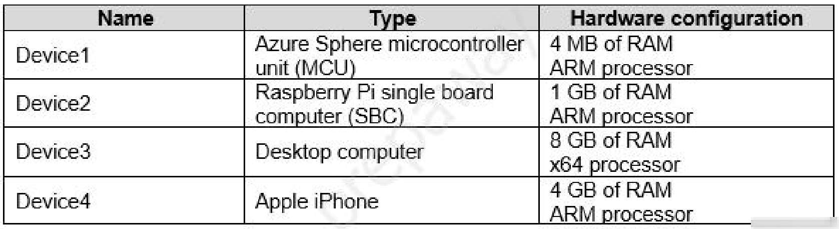
not_a_robot 4 years, 11 months ago

My bad, the original answer is correct.

upvoted 4 times

Question #8 *Topic 3*

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

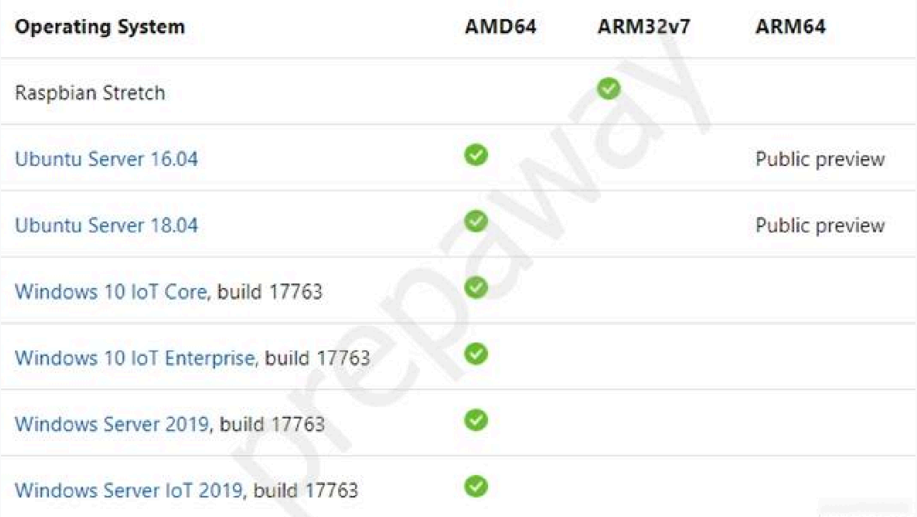
D. Device4

Suggested Answer: *BC*

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

Community vote distribution

BC (100%)

d0bermannn 3 years, 6 months ago

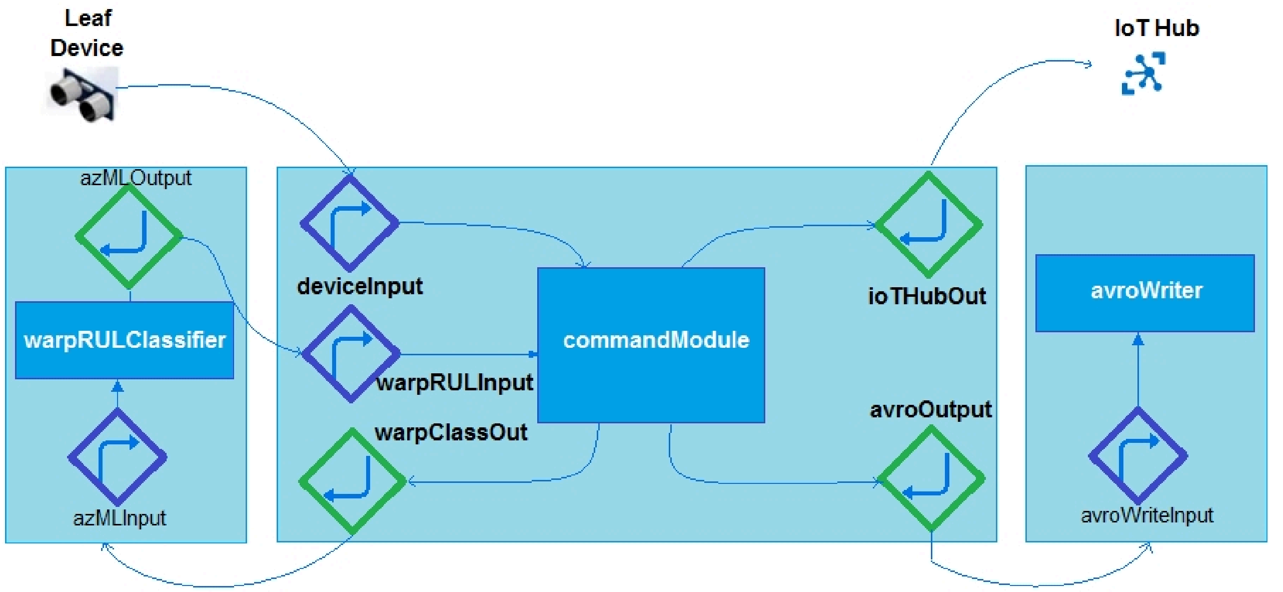
Selected Answer: BC

B&C is correct

upvoted 4 times

HOTSPOT -

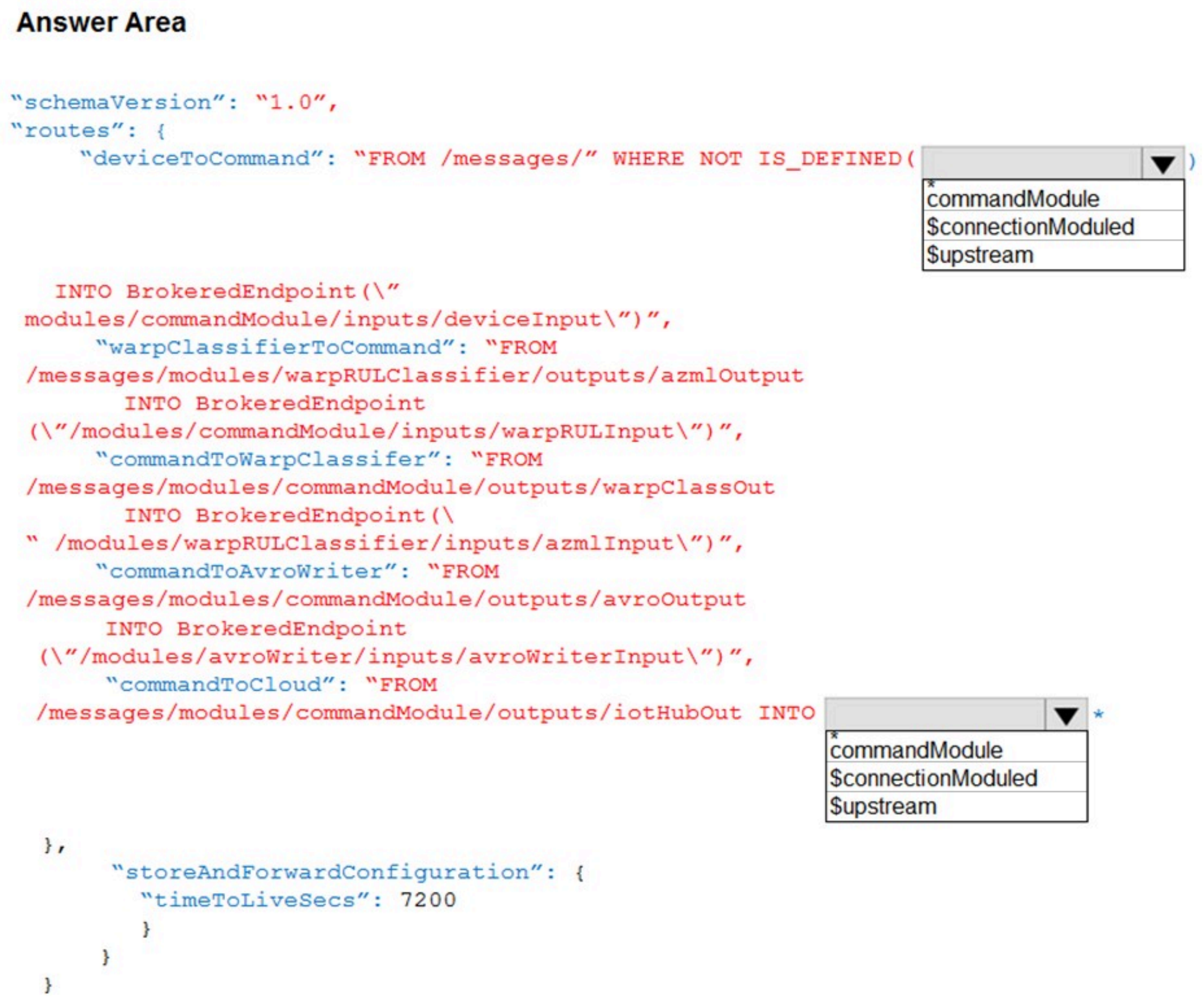
You need to configure Azure IoT Edge module routing to ensure that modules route traffic as shown in the following exhibit.



How should you complete the IoT Edge module routes? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

Hot Area:



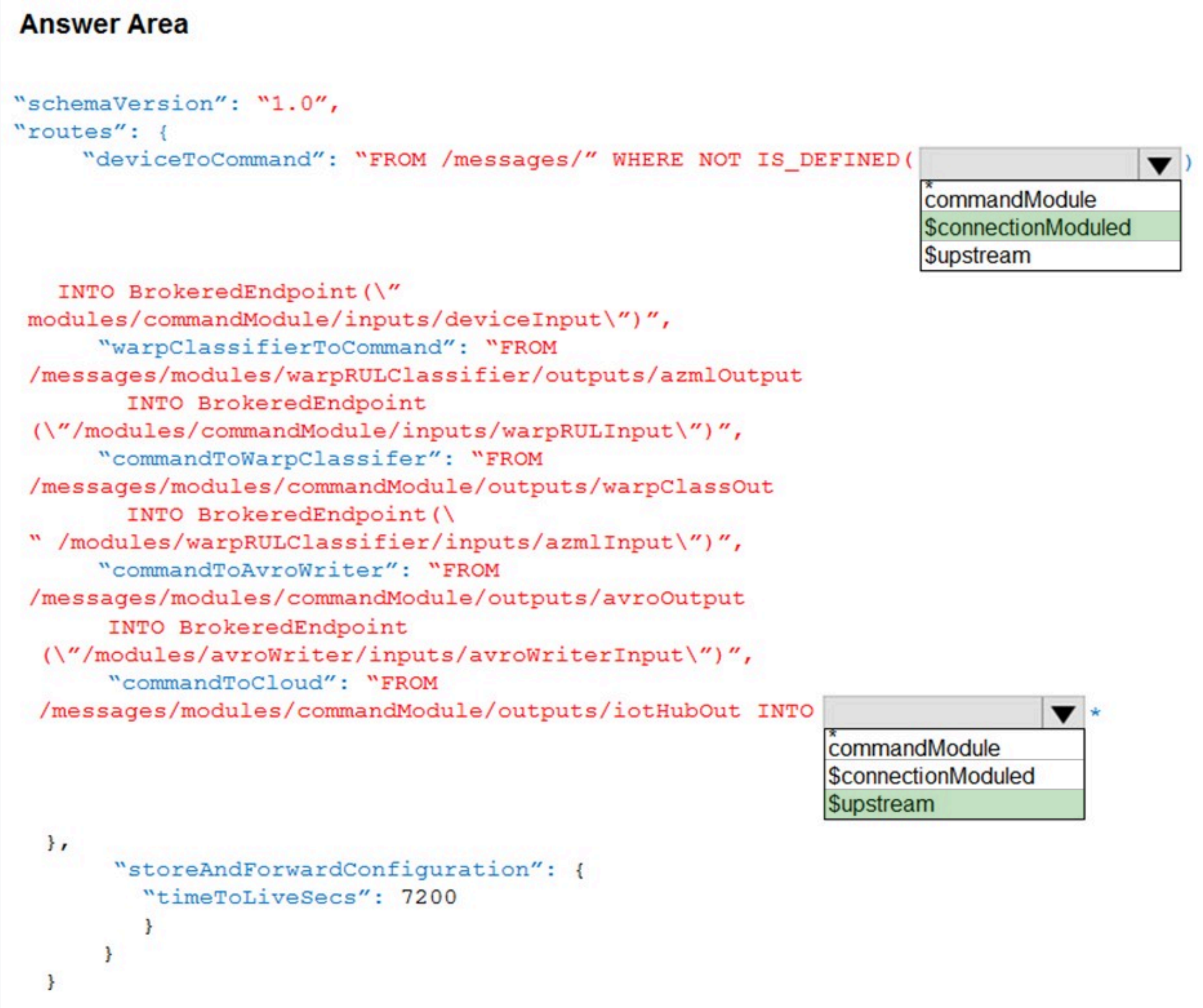
Answer Area

```
"schemaVersion": "1.0",
"routes": {
     "deviceToCommand": "FROM /messages/" WHERE NOT IS_DEFINED( [▼] )
                                                               *
                                                               commandModule
                                                               $connectionModuled
                                                               $upstream

   INTO BrokeredEndpoint(\"
 modules/commandModule/inputs/deviceInput\")",
      "warpClassifierToCommand": "FROM
 /messages/modules/warpRULClassifier/outputs/azmlOutput
        INTO BrokeredEndpoint
 (\"/modules/commandModule/inputs/warpRULInput\")",
      "commandToWarpClassifer": "FROM
 /messages/modules/commandModule/outputs/warpClassOut
        INTO BrokeredEndpoint(\
 " /modules/warpRULClassifier/inputs/azmlInput\")",
      "commandToAvroWriter": "FROM
 /messages/modules/commandModule/outputs/avroOutput
       INTO BrokeredEndpoint
  (\"/modules/avroWriter/inputs/avroWriterInput\")",
       "commandToCloud": "FROM
  /messages/modules/commandModule/outputs/iotHubOut INTO [▼] *
                                                          *
                                                          commandModule
                                                          $connectionModuled
                                                          $upstream

   },
        "storeAndForwardConfiguration": {
          "timeToLiveSecs": 7200
          }
       }
   }
```

Suggested Answer:



Answer Area

```
“schemaVersion”: “1.0”,
“routes”: {
     “deviceToCommand”: “FROM /messages/” WHERE NOT IS_DEFINED(              )
                                                                  commandModule
                                                                  $connectionModuled
                                                                  $upstream

  INTO BrokeredEndpoint(\”
 modules/commandModule/inputs/deviceInput\”)”,
      “warpClassifierToCommand”: “FROM
 /messages/modules/warpRULClassifier/outputs/azmlOutput
         INTO BrokeredEndpoint
 (\”/modules/commandModule/inputs/warpRULInput\”)”,
      “commandToWarpClassifer”: “FROM
 /messages/modules/commandModule/outputs/warpClassOut
         INTO BrokeredEndpoint(\
 “ /modules/warpRULClassifier/inputs/azmlInput\”)”,
      “commandToAvroWriter”: “FROM
 /messages/modules/commandModule/outputs/avroOutput
        INTO BrokeredEndpoint
  (\”/modules/avroWriter/inputs/avroWriterInput\”)”,
       “commandToCloud”: “FROM
  /messages/modules/commandModule/outputs/iotHubOut INTO              *
                                                           commandModule
                                                           $connectionModuled
                                                           $upstream

   },
        “storeAndForwardConfiguration”: {
          “timeToLiveSecs”: 7200
          }
       }
   }
```

Box 1: $connectionModuled -

Add a route that tells the edge hub to route any message received by the IoT Edge device that was not sent by an IoT Edge module.

Box 2: $upstream -

Send messages to $upstream, which passes the messages to the connected IoT Hub.

Reference:

https://docs.microsoft.com/en-us/azure/iot-edge/tutorial-machine-learning-edge-06-custom-modules

d0bermannn 3 years, 6 months ago

Provided answer is correct, but in question there is some typo

must be connectionModuleID not connectionModuled

upvoted 4 times

Question #10 *Topic 3*

DRAG DROP -

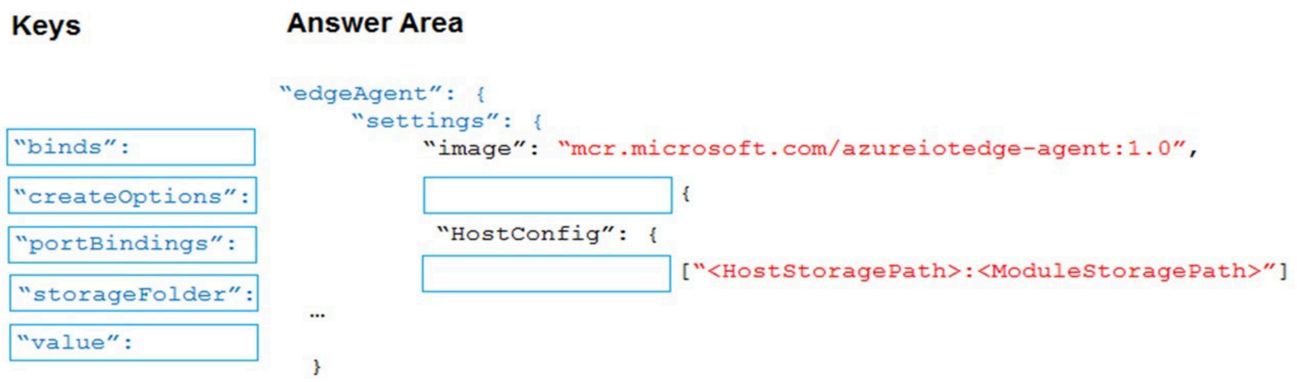
You have an Azure IoT Edge device named Edge1.

You need to configure the module container to link the module storage to the host storage.

How should you configure the deployment manifest? To answer, drag the appropriate keys to the correct targets. Each key may be used once, more than once, or not at all. You may need to drag the split bar between panes or scroll to view content.

NOTE: Each correct selection is worth one point.

Select and Place:



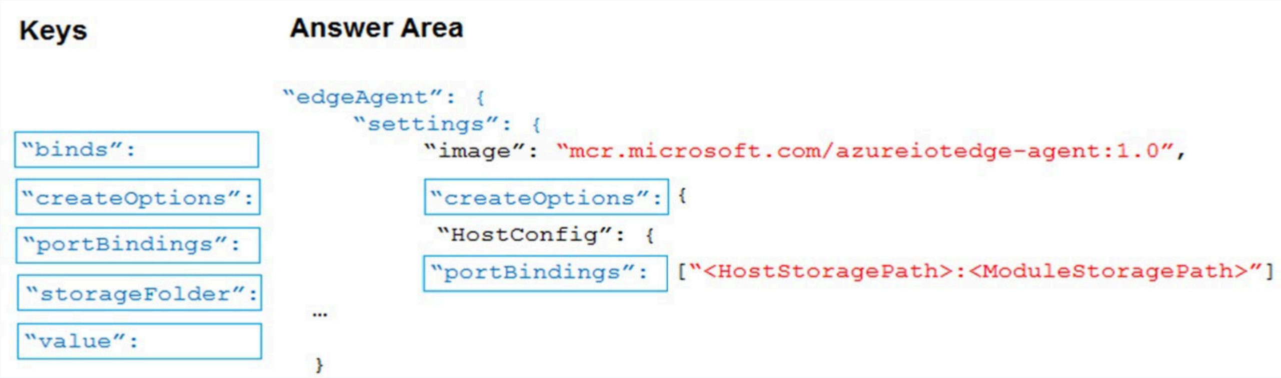
Keys

- “binds”:
- “createOptions”:
- “portBindings”:
- “storageFolder”:
- “value”:

Answer Area

```
“edgeAgent”: {
     “settings”: {
          “image”: “mcr.microsoft.com/azureiotedge-agent:1.0”,
          [                ] {
           “HostConfig”: {
          [                ] [“<HostStoragePath>:<ModuleStoragePath>”]
  ...
  }
```

Suggested Answer:



Keys

- “binds”:
- “createOptions”:
- “portBindings”:
- “storageFolder”:
- “value”:

Answer Area

```
“edgeAgent”: {
     “settings”: {
          “image”: “mcr.microsoft.com/azureiotedge-agent:1.0”,
          [“createOptions”:] {
           “HostConfig”: {
          [“portBindings”: ] [“<HostStoragePath>:<ModuleStoragePath>”]
  ...
  }
```

Box 1: createOptions -

Every module has a settings property that contains the module image, an address for the container image in a container registry, and any createOptions to configure the image on startup.

Box 2: portbindings -

Use the PortBindings setting in the HostConfig group of the Docker container create options to map the exposed port in the module to a port on the host device.

For example, if you exposed port 8080 inside the module and want to map that to port 80 of the host device, the create options in the template.json file would look like the following example:

```
"createOptions": {
"HostConfig": {
"PortBindings": {
"8080/tcp": [
{
"HostPort": "80"
}
]
}
}
}
```

Reference:

https://docs.microsoft.com/en-us/azure/iot-edge/how-to-use-create-options

kohmaksimka Highly Voted 3 years, 5 months ago

Answer is incorrect, should be:
createOptions & binds

No portmapping is done in the following code but accessing the device local storage instead of container temporary storage:
https://docs.microsoft.com/en-us/azure/iot-edge/how-to-access-host-storage-from-module?view=iotedge-2020-11
upvoted 20 times

⊟ **Ouss7** 2 years, 11 months ago
yeah you are right thank you
upvoted 1 times

⊟ **[Removed]** 3 years, 1 month ago
Yeah, you are right, it is path binding, not port.
upvoted 1 times

⊟ **d0bermannn** Most Recent 3 years, 6 months ago
looks correct as we see link provided
upvoted 1 times

Question #11 *Topic 3*

You are developing an Azure IoT solution for a shipping company. The company's ships will have sensors used for predictive maintenance. Some sensor devices will be MQTT-capable, and others will use Modbus.

Each ship has an internet connection that is available only when the ship is docked.

You create an Azure IoT hub.

You need to implement an IoT solution that uses Azure IoT Edge.

What should you do?

A. Configure an IoT Edge gateway. Deploy an IoT Edge Modbus module. From the Azure portal, create IoT devices and add connection strings to the devices.

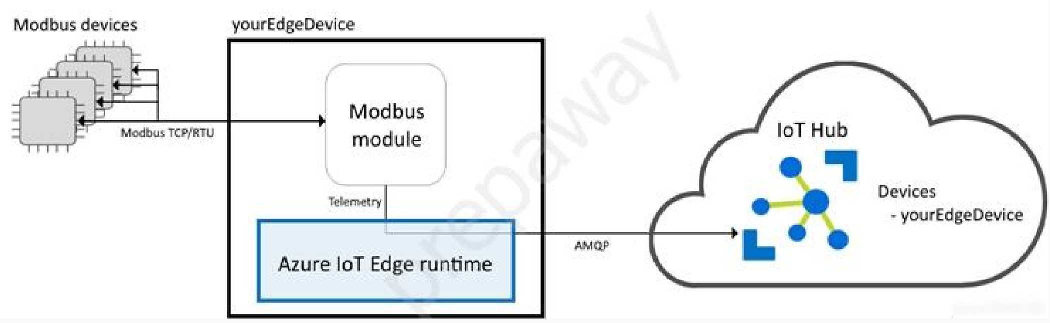
B. Add the MQTT devices to the IoT hub and configure an IoT Edge gateway. From the IoT Edge gateway device, assign the MQTT devices as child devices of the gateway. Use the File upload feature of IoT Hub when internet connectivity is available.

C. Add the MQTT devices to the IoT hub, configure an IoT Edge gateway, and set Enable connection to IoT Hub to Disable. From the IoT Edge gateway device, assign the MQTT devices as child devices of the gateway. Deploy the IoT Edge Modbus module.

D. Add the MQTT devices to the IoT hub and configure an IoT Edge gateway. From the IoT Edge gateway device, assign the MQTT devices as child devices of the gateway. Deploy an IoT Edge Modbus module.

Suggested Answer: *C*

Note: If you want to connect IoT devices that use Modbus TCP or RTU protocols to an Azure IoT hub, you can use an IoT Edge device as a gateway. The gateway device reads data from your Modbus devices, then communicates that data to the cloud using a supported protocol.



Incorrect Answers:

A: The MQTT devices should be added to the IoT hub.

B: Need to use an IoT Edge Modbus module.

Reference:

https://docs.microsoft.com/en-us/azure/iot-edge/deploy-modbus-gateway

Community vote distribution

D (100%)

⊟ d0bermannn Highly Voted 3 years, 6 months ago

why not D? disconnected state sounds too pessimisic)

upvoted 6 times

⊟ j_c_000 Most Recent 2 years, 1 month ago

Selected Answer: D

D! Can't be disabled, because still needs to connect when in port and transmit stored telemetry

upvoted 2 times

⊟ LaggAt 2 years, 5 months ago

Selected Answer: D

I agree with d0bermannn, why should we set disconnected state?

upvoted 1 times

Question #12 *Topic 3*

HOTSPOT -

You have an Azure subscription that contains an Azure IoT hub, an Azure IoT Edge gateway, and 1,000 leaf devices. The leaf devices use a custom communication protocol that is NOT supported by the IoT hub.

You need to configure the gateway to meet the following requirements:

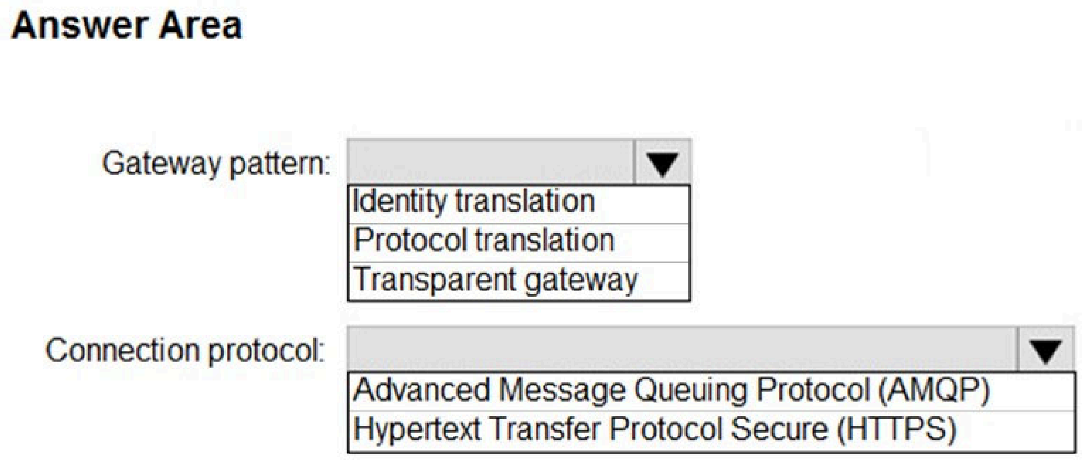
✑ Minimize the number of connections between the gateway and the IoT hub.

✑ Support addressing cloud-to-device messages to individual leaf devices.

How should you configure the gateway? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

Hot Area:

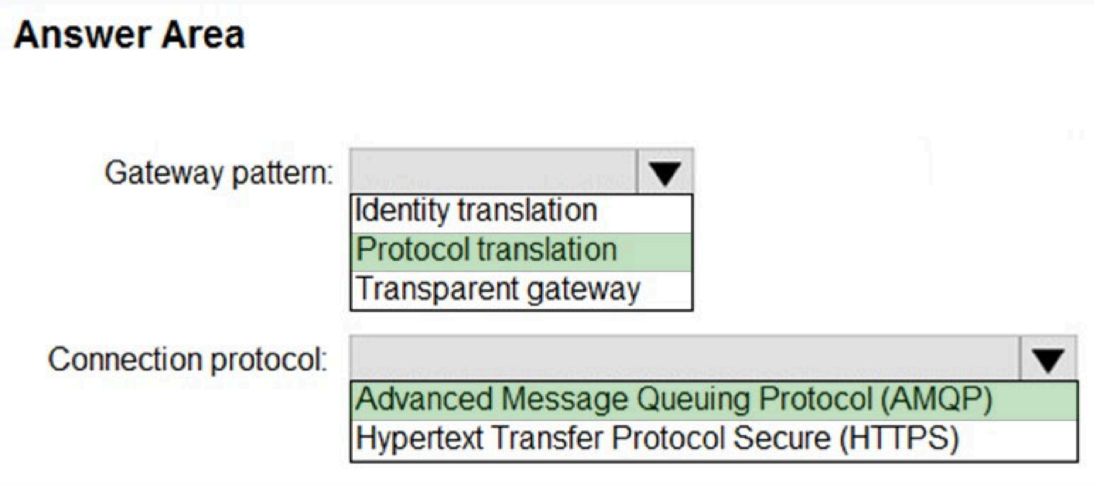


Answer Area

Gateway pattern: ▼
- Identity translation
- Protocol translation
- Transparent gateway

Connection protocol: ▼
- Advanced Message Queuing Protocol (AMQP)
- Hypertext Transfer Protocol Secure (HTTPS)

Suggested Answer:



Answer Area

Gateway pattern: ▼
- Identity translation
- [✓ Protocol translation]
- Transparent gateway

Connection protocol: ▼
- [✓ Advanced Message Queuing Protocol (AMQP)]
- Hypertext Transfer Protocol Secure (HTTPS)

Box 1: Protocol translation -

In the protocol translation gateway pattern, only the IoT Edge gateway has an identity with IoT Hub. The translation module receives messages from downstream devices, translates them into a supported protocol, and then the IoT Edge device sends the messages on behalf of the downstream devices.

Box 2: Advanced MessageQueuing Protocol (AMQP)

Connection multiplexing - All devices connecting to IoT Hub through an IoT Edge gateway can use the same underlying connection. This multiplexing capability requires that the IoT Edge gateway uses AMQP as its upstream protocol.

Reference:

https://docs.microsoft.com/en-us/azure/iot-edge/iot-edge-as-gateway

⊟ **kohmaksimka** Highly Voted 3 years, 5 months ago

Should be:

Identity translation: Support addressing cloud-to-device messages (https://docs.microsoft.com/en-us/azure/iot-edge/iot-edge-as-gateway?view=iotedge-2020-11#device-capabilities-behind-translation-gateways). Identity translation is built upon protocol translation

AMQP: Minimize number of connections

upvoted 17 times

⊟ **j_c_000** Most Recent 2 years, 1 month ago

Identity translation. "The cloud can address each connected device individually" in this case.

upvoted 1 times

⊟ **IMARRA** 2 years, 11 months ago

how to 'Minimize the number of connections between the gateway and the IoT hub'?

upvoted 1 times

⊟ **lah123** 2 years, 7 months ago

by using AMQP

upvoted 1 times

⊟ **liberty123** 3 years, 3 months ago

Identity translation

AMQP

upvoted 3 times

⊟ **d0bermannn** 3 years, 6 months ago

looks ok like amqp over nat)

upvoted 1 times

⊟ **d0bermannn** 3 years, 5 months ago

but for the cloud can address each connected device individually in box1 must be Identity translation

upvoted 1 times

⊟ **TechieBloke** 3 years, 5 months ago

Wrong!

Look at the link:

https://docs.microsoft.com/en-us/azure/iot-edge/iot-edge-as-gateway?view=iotedge-2020-11

With identity translation there are many logical connections. Given ans correct.

upvoted 1 times

⊟ **d0bermannn** 3 years, 5 months ago

No way, must be id.translation.

In question we have reqs. ' Support addressing cloud-to-device messages to individual leaf devices' and nice link about edge gw said that

- protocol translation supports only 'The cloud can only address the gateway device'

- id. translation supports 'The cloud can address each connected device individually', i.e. exactly the reqs.

upvoted 4 times

Question #13 *Topic 3*

You have an Azure IoT Edge module named SampleModule that runs on a device named Device1.
You make changes to the code of SampleModule by using Microsoft Visual Studio Code.
You need to push the code to the container registry and then deploy the module to Device1.
Which two actions should you perform from Visual Studio Code? Each correct answer presents part of the solution.
NOTE: Each correct selection is worth one point.

A. Build and push the SampleModule code to the registry.

B. Create a deployment for a single device.

C. Generate a deployment manifest.

D. Build an IoT Edge solution.

E. Generate a shared access signature (SAS) token for Device1.

Suggested Answer: *BC*
C: Configure a deployment manifest. A deployment manifest is a JSON document that describes which modules to deploy, how data flows between the modules, and desired properties of the module twins.
B: You deploy modules to your device by applying the deployment manifest that you configured with the module information.
Reference:
https://docs.microsoft.com/en-us/azure/iot-edge/how-to-deploy-modules-vscode

Community vote distribution
AB (65%) BC (35%)

kohmaksimka Highly Voted 3 years, 5 months ago
Selected Answer: AB
Would say AB
A: Build and push the SampleModule code to registry - as this is requested
B: Create a deployment for single device - to deploy the module to the device

I would not take C as the module already runs on the device. So no need to generate a (a.k.a. create a new) deployment manifest
upvoted 6 times

kohmaksimka 3 years, 5 months ago
So nothing really changes in the existing manifest (maybe change module version but that's it)
upvoted 4 times

RajeevP26 3 years, 5 months ago
Agreed with A & B
https://docs.microsoft.com/en-us/azure/iot-edge/tutorial-develop-for-windows?view=iotedge-2018-06
upvoted 2 times

j_c_000 Most Recent 2 years, 1 month ago
Selected Answer: AB
Correct answer HAS to include "A", the question plainly says we need to push our changes.

The link below is not great. It only creates a deployment for a microsoft sample module, not a custom code module, so doesn't directly answer this question.
https://learn.microsoft.com/en-us/azure/iot-edge/how-to-deploy-modules-vscode?view=iotedge-1.4
upvoted 2 times

superPashtet 2 years, 9 months ago
Selected Answer: AB
Agreed with A & B
upvoted 3 times

LightningKeven 2 years, 11 months ago
Selected Answer: AB

Agreed with A & B

upvoted 2 times

nqthien041292 3 years, 3 months ago

Selected Answer: BC

Vote BC for description related to IoT

upvoted 1 times

liberty123 3 years, 3 months ago

Selected Answer: BC

- Configure a deployment manifest
- Create Deployment for Single Device.

https://docs.microsoft.com/en-us/azure/iot-edge/how-to-deploy-modules-vscode?view=iotedge-2020-11

upvoted 4 times

d0bermannn 3 years, 6 months ago

Selected Answer: BC

B&C looks ok as link provided

upvoted 2 times

HOTSPOT -

You have an Azure subscription that contains an Azure IoT hub and two IoT devices named Device1 and Device2.

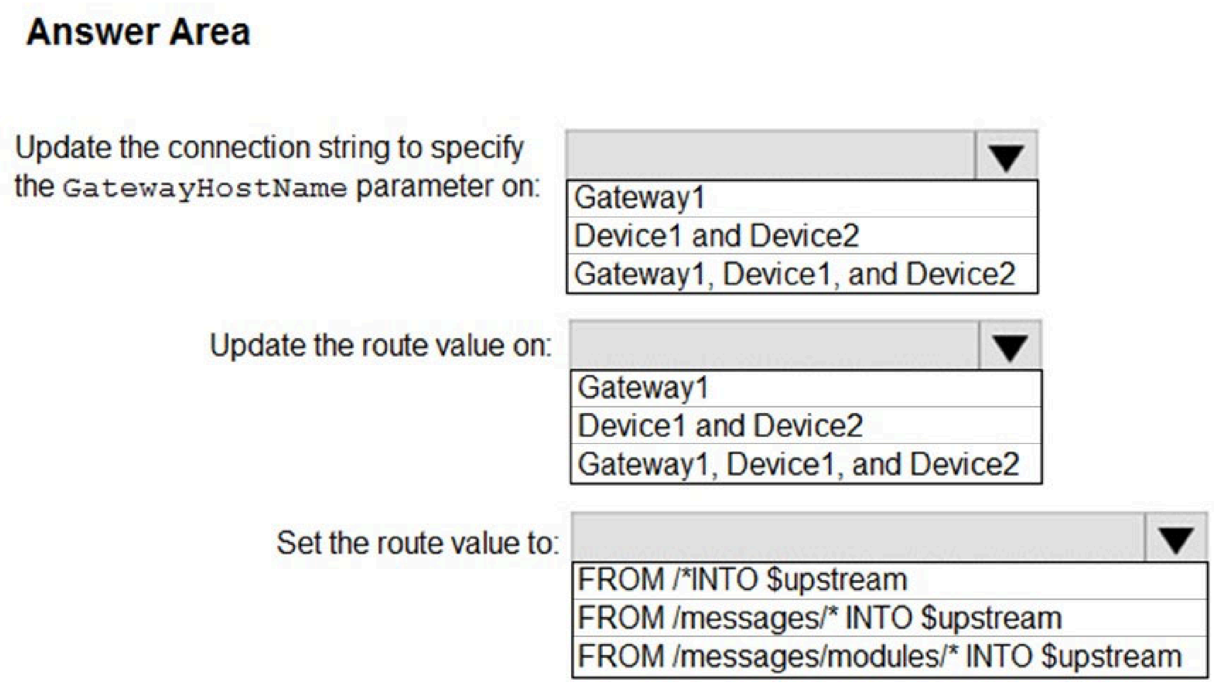
You plan to deploy an Azure IoT Edge gateway device named Gateway1.

You need to ensure that all device-to-cloud messages and twin change notifications from Device1 and Device2 to the IoT hub are routed by using Gateway1.

What tasks should you perform to configure the devices? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

Hot Area:



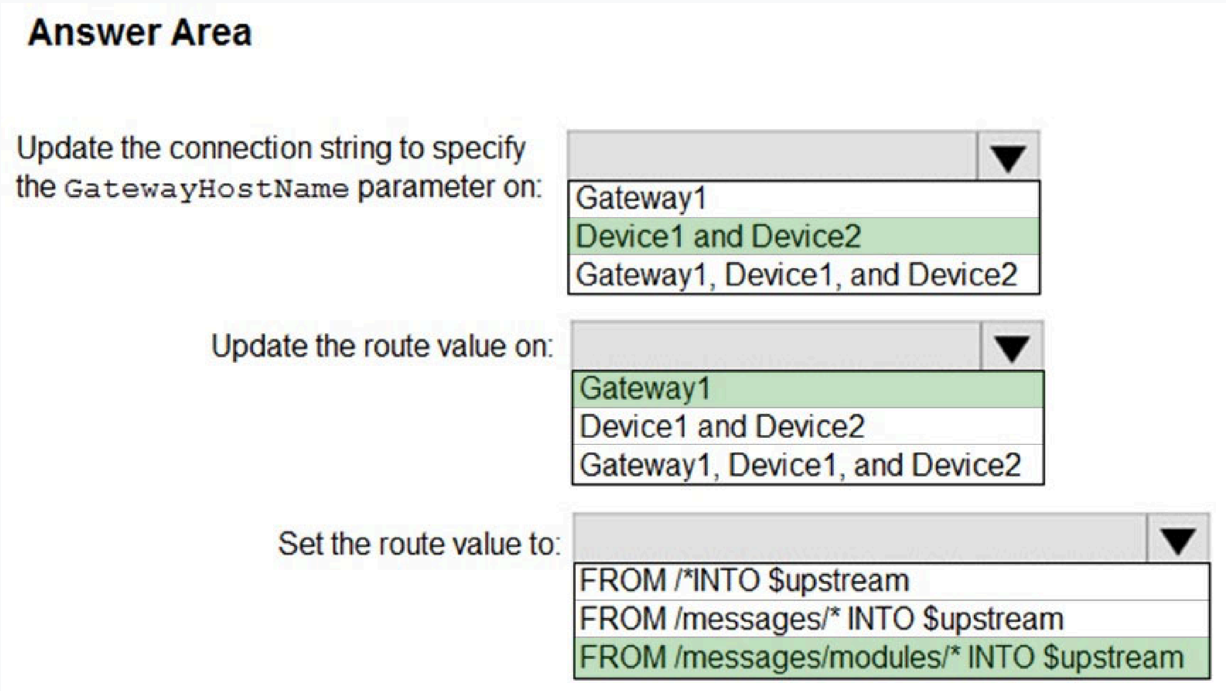
Answer Area

Update the connection string to specify the `GatewayHostName` parameter on:
- Gateway1
- Device1 and Device2
- Gateway1, Device1, and Device2

Update the route value on:
- Gateway1
- Device1 and Device2
- Gateway1, Device1, and Device2

Set the route value to:
- FROM /*INTO $upstream
- FROM /messages/* INTO $upstream
- FROM /messages/modules/* INTO $upstream

Suggested Answer:



Answer Area

Update the connection string to specify the `GatewayHostName` parameter on:
- Gateway1
- [✓ Device1 and Device2]
- Gateway1, Device1, and Device2

Update the route value on:
- [✓ Gateway1]
- Device1 and Device2
- Gateway1, Device1, and Device2

Set the route value to:
- FROM /*INTO $upstream
- FROM /messages/* INTO $upstream
- [✓ FROM /messages/modules/* INTO $upstream]

Box 1: Device1 and Device2 -

Connection strings for downstream devices need the following components:

✑ The gateway device that the device connects through. Provide the hostname value from the IoT Edge gateway device's config file:

GatewayHostName=

{gateway hostname}

Box 2: Gateway1 -

To deploy the IoT Edge hub module and configure it with routes to handle incoming messages from downstream devices, follow these steps:
1. In the Azure portal, navigate to your IoT hub.
2. Go to IoT Edge and select your IoT Edge device that you want to use as a gateway.
3. Select Set Modules.
4. On the Modules page, you can add any modules you want to deploy to the gateway device.
5. Select Next: Routes.
6. On the Routes page, make sure that there is a route to handle messages coming from downstream devices. For example:
A route that sends all messages, whether from a module or from a downstream device, to IoT Hub:

Name: allMessagesToHub -
Value: FROM /messages/* INTO $upstream
Box 3: FROM /messages/* INTO $upstream
Reference:
https://docs.microsoft.com/en-us/azure/iot-edge/how-to-authenticate-downstream-device

kohmaksimka Highly Voted 👍 3 years, 5 months ago

Correct answer:
BOX 1: Device1 and Device2: GatewayHostName is configured on leaf devices

BOX2: Gateway1 - routing is defined in the edgeHub module

BOX3: FROM /* INTO $upstream to send both device-to-cloud messages and twin updates to cloud
https://docs.microsoft.com/en-us/azure/iot-edge/module-composition?view=iotedge-2020-11#source

upvoted 19 times

liberty123 3 years, 3 months ago

Thanks you, I agree with you: BOX3: FROM /* INTO $upstream

upvoted 3 times

RajeevP26 Most Recent ⊙ 3 years, 5 months ago

The FROM /* part of the message route will match all device-to-cloud messages or twin change notifications from any module or leaf device. Then, the INTO $upstream tells the route to send those messages to the Azure IoT Hub.

upvoted 2 times

d0bermannn 3 years, 6 months ago

linl provided is not reliable fo question
box 1&2 is ok, like routing on L3)
box 3 must be Box 3: FROM /messages/* INTO $upstream

upvoted 2 times

kohmaksimka 3 years, 5 months ago

Incorrect, should be FROM /* INTO $upstream as /messages/* only relates to device-to-cloud messages

upvoted 1 times

Question #15 *Topic 3*

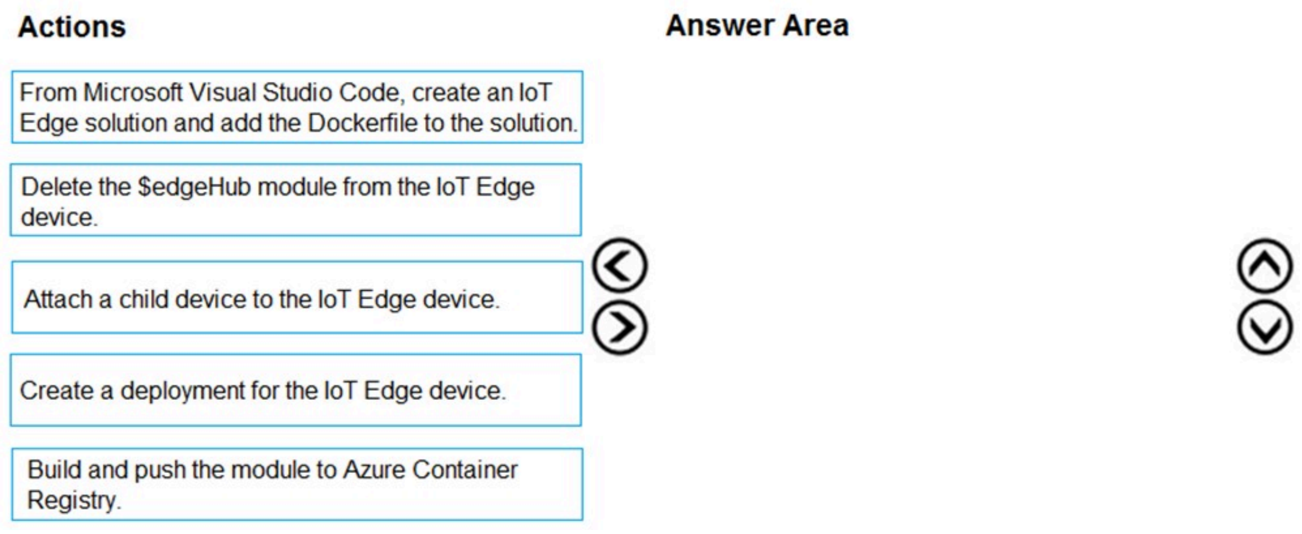
DRAG DROP -

Your company develops a custom module and exports the module as a Linux Dockerfile.

You need to deploy the module to an Azure IoT Edge device that runs Ubuntu Server 18.04.

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

Select and Place:



Actions

From Microsoft Visual Studio Code, create an IoT Edge solution and add the Dockerfile to the solution.

Delete the $edgeHub module from the IoT Edge device.

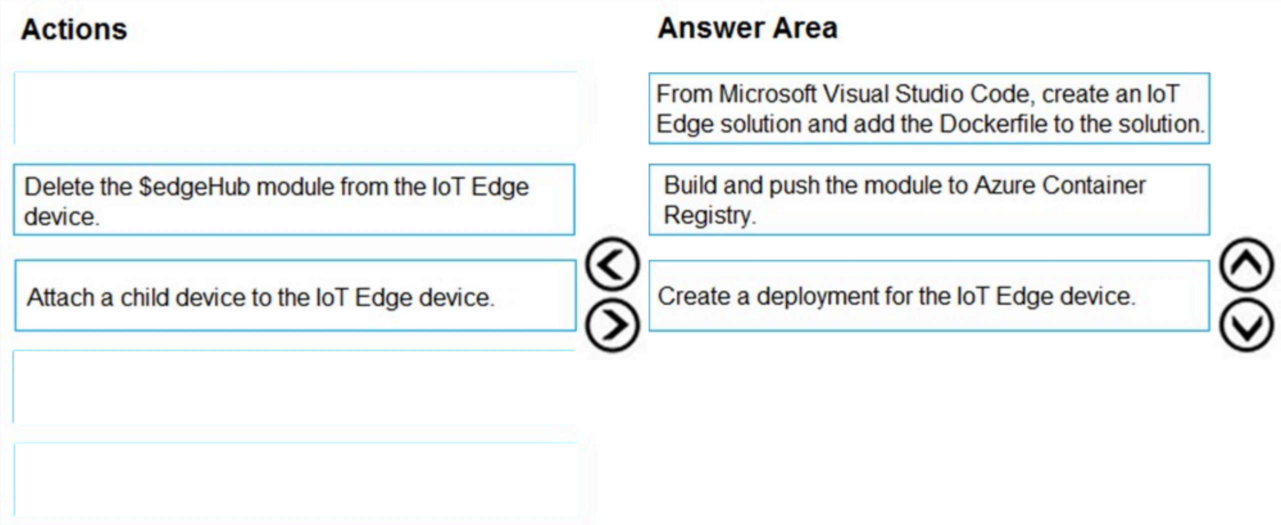
Attach a child device to the IoT Edge device.

Create a deployment for the IoT Edge device.

Build and push the module to Azure Container Registry.

Answer Area

Suggested Answer:



Actions

Delete the $edgeHub module from the IoT Edge device.

Attach a child device to the IoT Edge device.

Answer Area

From Microsoft Visual Studio Code, create an IoT Edge solution and add the Dockerfile to the solution.

Build and push the module to Azure Container Registry.

Create a deployment for the IoT Edge device.

Step 1: From Microsoft Visual Studio Code,ג€¦

The Azure IoT Tools extension provides project templates for all supported IoT Edge module languages in Visual Studio Code. These templates have all the files and code that you need to deploy a working module to test IoT Edge, or give you a starting point to customize the template with your own business logic.

Step 2: Build and push the module to Azure Container Registry

Build and push your solution. Review the module code and the deployment. Then build the SampleModule container image and push it to your container registry.

Step 3: Create a deployment for the IoT Edge device.

Verify that the built container images are stored in your container registry, then deploy the modules to the device.

Reference:

https://docs.microsoft.com/en-us/azure/iot-edge/tutorial-develop-for-linux?view=iotedge-2020-11

d0bermannn Highly Voted 3 years, 5 months ago

looks correct as we see link provided

upvoted 5 times

Question #16 *Topic 3*

Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution.
After you answer a question in this section, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.
You are developing a custom Azure IoT Edge module.
The module needs to identify the device ID of the local device.
Solution: You configure the module to read the IOTEDGE_DEVICEID environment variable.
Does this meet the goal?

A. Yes

B. No

Suggested Answer: *B*
The Azure ID of the current device is available on the IOTEDGE_DEVICEID environment variable.
Instead read the device ID of the device twin.
Note: Device twins are JSON documents that store device state information including metadata, configurations, and conditions. Azure IoT Hub maintains a device twin for each device that you connect to IoT Hub.
Device identity properties. The root of the device twin JSON document contains the read-only properties from the corresponding device identity stored in the identity registry.
Reference:
https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-devguide-device-twins

Community vote distribution

A (100%)

kohmaksimka Highly Voted 3 years, 5 months ago

Selected Answer: A

Should be YES.
IOTEDGE_DEVICEID is an environment variable that holds the device ID of the local device.
You can check this through the command: sudo docker inspect <MODULENAME> on the edge device.
+ also stated here https://github.com/Azure/iotedge/issues/633

upvoted 7 times

j_c_000 Most Recent 2 years, 1 month ago

Selected Answer: A

Should be YES, this is exactly what you're supposed to do.

upvoted 2 times

data12345 2 years, 7 months ago

Should be yes - source: https://github.com/MicrosoftDocs/azure-docs/issues/18767.

upvoted 1 times

Question #18 *Topic 3*

Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution.
After you answer a question in this section, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.
You are developing a custom Azure IoT Edge module.
The module needs to identify the device ID of the local device.
Solution: You configure the module to read the device ID of the device twin.
Does this meet the goal?

A. Yes

B. No

Suggested Answer: *A*
Device twins are JSON documents that store device state information including metadata, configurations, and conditions. Azure IoT Hub maintains a device twin for each device that you connect to IoT Hub.
Device identity properties. The root of the device twin JSON document contains the read-only properties from the corresponding device identity stored in the identity registry.
Reference:
https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-devguide-device-twins

theScriptingEngineer 3 years, 3 months ago
As far as I know, the full device twin is only available in the cloud for back-end apps. The IoT Edge only gets the desired properties part of the device twin
upvoted 3 times

trahd 2 years, 2 months ago
I can confirm your statement in testing with C#.
From a code snippet

```
...
s_deviceClient = DeviceClient.CreateFromConnectionString(s_connectionString, Microsoft.Azure.Devices.Client.TransportType.Mqtt_Tcp_Only);
var twin = await s_deviceClient.GetTwinAsync();
Console.WriteLine(twin.ToJson( Formatting.Indented));
...
```

Got the following output:

```
{
"deviceId": null,
"etag": null,
"version": null,
"properties": {
"desired": {
"FPS": 45,
"$version": 5
},
"reported": {
"$version": 1
}
}
}
```

Other Notes:
SDK: Microsoft.Azure.Devices.Client version 1.41.3.0
.NET 7.02
Tested on non-IoT Edge device but I don't expect that to make a difference.
upvoted 1 times

Question #20 *Topic 3*

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

After you answer a question in this 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 and an Azure IoT Edge device.

You plan to deploy 10 Bluetooth sensors. The sensors do not support MQTT, AMQP, or HTTPS.

You need to ensure that all the sensors appear in the IoT hub as a single device.

Solution: You configure the sensors to connect directly to the IoT hub.

Does this meet the goal?

A. Yes

B. No

Suggested Answer: *B*

Instead use a translation gateway.

Note: In the protocol translation gateway pattern, only the IoT Edge gateway has an identity with IoT Hub. The translation module receives messages from downstream devices, translates them into a supported protocol, and then the IoT Edge device sends the messages on behalf of the downstream devices. All information looks like it is coming from one device, the gateway.

Reference:

https://docs.microsoft.com/en-us/azure/iot-edge/iot-edge-as-gateway

Community vote distribution

B (100%)

kohmaksimka 3 years, 5 months ago

Selected Answer: B

Correct

upvoted 4 times

Question #21 *Topic 3*

You plan to develop modules for an Azure IoT Edge solution.

You need to recommend a development tool that supports the following:

• Node.js
• Module templates
• Development on Linux workstations

What should you recommend?

A. the Azure IoT explorer

B. Microsoft Visual Studio

C. Microsoft Visual Studio Online

D. Microsoft Visual Studio Code

Suggested Answer: *C*

Community vote distribution

D (100%)

⊟ **j_c_000** **Highly Voted** 2 years, 1 month ago

Selected Answer: D

VSCode supports everything listed here.

https://learn.microsoft.com/en-us/azure/iot-edge/tutorial-develop-for-linux

upvoted 6 times

Question #22 *Topic 3*

You have an Azure subscription that contains an Azure IoT hub and two Azure IoT Edge devices named Device1 and Device2.

You need to ensure that the IoT hub only accepts connections from Device1 and Device2.

What should you configure?

A. Azure API Management

B. Azure Active Directory (Azure AD) Identity Protection

C. Azure Defender for IoT

D. an IP filter

Suggested Answer: *B*

Community vote distribution

D (100%)

xixi4den Highly Voted 2 years, 1 month ago

Selected Answer: D

Should be D

upvoted 6 times

ChaiD Most Recent 1 year, 11 months ago

Yes it's D, on the test there's no confusion (the site is legit, pay attention to these comments, the test was 54 questions of which I had seen 53 before, scored 865). This comment applies to any other questions in here with this option as a choice.

upvoted 1 times

hotwheelsinsf 2 years, 2 months ago

I thought this answer should be Private Endpoint as per another question on this test.

upvoted 1 times

Yameo 2 years ago

As I get it there are two possible correct answers, that's why only one of them is present in each answer set.

upvoted 1 times

Question #23 *Topic 3*

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

After you answer a question in this 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 and an Azure IoT Edge device.

You plan to deploy 10 Bluetooth sensors. The sensors do not support MQTT, AMQP, or HTTPS.

You need to ensure that all the sensors appear in the IoT hub as a single device.

Solution: You configure the IoT Edge device as an IoT Edge identity translation gateway. You configure the sensors to connect to the device.

Does this meet the goal?

A. Yes

B. No

Suggested Answer: *A*

Community vote distribution

B (100%)

Yameo 2 years ago

Selected Answer: B

Should be Protocol translation, not Identity one

upvoted 1 times

j_c_000 2 years, 1 month ago

Selected Answer: B

Should be "B", "NO". The question requires "You need to ensure that all the sensors appear in the IoT hub as a single device." If you do identity translation, they will show up as multiple devices. Need protocol translation.

upvoted 2 times

xixi4den 2 years, 1 month ago

Selected Answer: B

Should be Protocol translation, not Identity one

upvoted 1 times

hotwheelsinsf 2 years, 2 months ago

A. identity translation gateway is what you need to configure here so the individual devices can connect and also translate the protocol coming from these devices

upvoted 1 times

slafcemafce 2 years, 2 months ago

Is should be B. No. Protocol translation is needed instead of identity translation.

https://learn.microsoft.com/en-us/azure/iot-edge/iot-edge-as-gateway?view=iotedge-1.4

upvoted 2 times

Question #24 *Topic 3*

You have an Azure IoT Edge module named SampleModule that runs on a device named Device1.

You make changes to the code of SampleModule by using Microsoft Visual Studio Code.

You need to push the code to the container registry and then deploy the module to Device1.

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

NOTE: Each correct selection is worth one point.

A. Build and push the SampleModule code to the registry.

B. Create a deployment for a single device.

C. Upload to Azure Storage.

D. Build an IoT Edge solution.

E. Generate a shared access signature (SAS) token for Device1.

Suggested Answer: *BD*

Community vote distribution

AB (100%)

j_c_000 Highly Voted 👍 2 years, 1 month ago

Selected Answer: AB

Should be AB here. You already HAVE a solution, because you're "making changes". And of course, you have to push your code to the repository or edge devices won't get it.

upvoted 6 times

hotwheelsinsf Most Recent 2 years, 3 months ago

Question #13 in topic 3 what gives??

upvoted 1 times

hotwheelsinsf 2 years, 3 months ago

This question was asked earlier and had a different answer

upvoted 2 times

Question #27 *Topic 3*

You need to update the IoT Edge runtime by using rolling tags.

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 IoT Edge device, remove the edgeHub and edgeAgent container images.

B. Modify the systemModules section of the deployment manifest JSON file.

C. On the IoT Edge device, update the security daemon.

D. Add an update tag to the IoT Edge device twin.

Suggested Answer: *AC*

smudo1965 1 year, 11 months ago

Following this link https://learn.microsoft.com/en-us/azure/iot-edge/how-to-update-iot-edge?view=iotedge-1.4&tabs=linux#understand-iot-edge-tags given answers are correct

upvoted 1 times

Dilpreet23 1 year, 11 months ago

Should it be B & D?

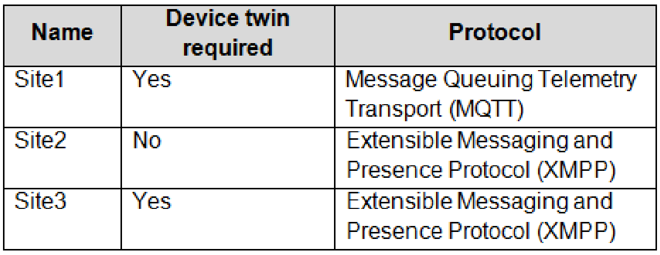
upvoted 1 times

Question #28 *Topic 3*

DRAG DROP

-

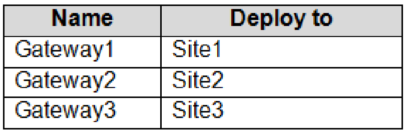
You have sites that contain IoT devices as shown in the following table.



Name	Device twin required	Protocol
Site1	Yes	Message Queuing Telemetry Transport (MQTT)
Site2	No	Extensible Messaging and Presence Protocol (XMPP)
Site3	Yes	Extensible Messaging and Presence Protocol (XMPP)

You have an Azure subscription.

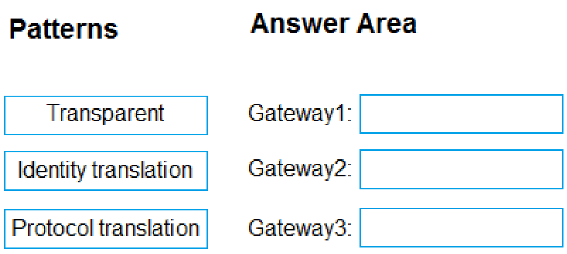
You need to create the Azure IoT Edge devices shown in the following table.



Name	Deploy to
Gateway1	Site1
Gateway2	Site2
Gateway3	Site3

Which type of gateway pattern should you use for each IoT Edge device? To answer, drag the appropriate gateway pattern types to the correct devices. Each pattern type 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.



Patterns

Transparent

Identity translation

Protocol translation

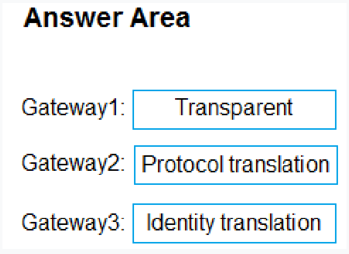
Answer Area

Gateway1:

Gateway2:

Gateway3:

Suggested Answer:



Answer Area

Gateway1: Transparent

Gateway2: Protocol translation

Gateway3: Identity translation

slafcemafce 2 years, 2 months ago

Correct answers

upvoted 3 times