

LAB 5 – Application Layer

Name – Sati, Ankit

Date – 10/13/2021

Section - 001

Total in points (Maximum 100 points)–

Professors Comments –

Affirmation of Independent Effort – Ankit Sati

Question 1-

1. This questions wants us to walk through the edge computing that is being developed and take a look at it from the different perspectives listed below.
 - a. NEC - Network edge compute
 - b. MEC - Multi access edge compute
 - c. Help of Robotics
 - d. Impact on mission critical systems.
 - e. Implementation and use of private networks.

Main Goal – The primary goal is to deploy everything at the end for the users so that they can reduce the time of data on the networks. We try to provide the below services at the end of the network, which makes it faster/accurate with less chances of frame drops.

- Compute Power
- Cloud space
- High Availability
- Service Space

Main Objective - Everything for the end user need to remain the same from code creation to deployment/maintenance but the end user will not know where it is being used from. It can either be on the edge of the network or the azure platforms as it has been in the past.

NEC - Network edge compute

Network Edge Compute (NEC) is the network carrier equivalent, placing the edge computing platform within their network. Instead of needing to access applications and games running in the public cloud, software providers can bring their solutions physically closer to their end-users. At AT&T's Business Summit we gave an augmented reality demonstration, working with Taqtile, and showed how to perform maintenance on an aircraft landing gear.

Prime features

- Closer to end user
- One hop communications
- Compute power at the edge.
- Deployment of services at the end.

MEC - Multi access edge compute

Through the combination of local compute resources and private mobile connectivity (private LTE), we can enable many new scenarios. For instance, in the smart factory example used earlier customers are now able to run their robotic control logic, highly available and independent of connectivity to the public cloud. MEC helps ensure that operations and any associated critical

first-stage data processing remain up and production can continue uninterrupted. Advantage of near-infinite compute and storage, the cloud is ideal for large data-intensive and computational tasks, such as machine learning jobs for predictive maintenance analytics.

Prime Features

- Combination of local compute resources and private mobile connectivity
- Communication over private network.
- Complex logic like robotics can be accessed.
- **Near infinite compute storage.**

Prime advantages of EDGE COMPUTE

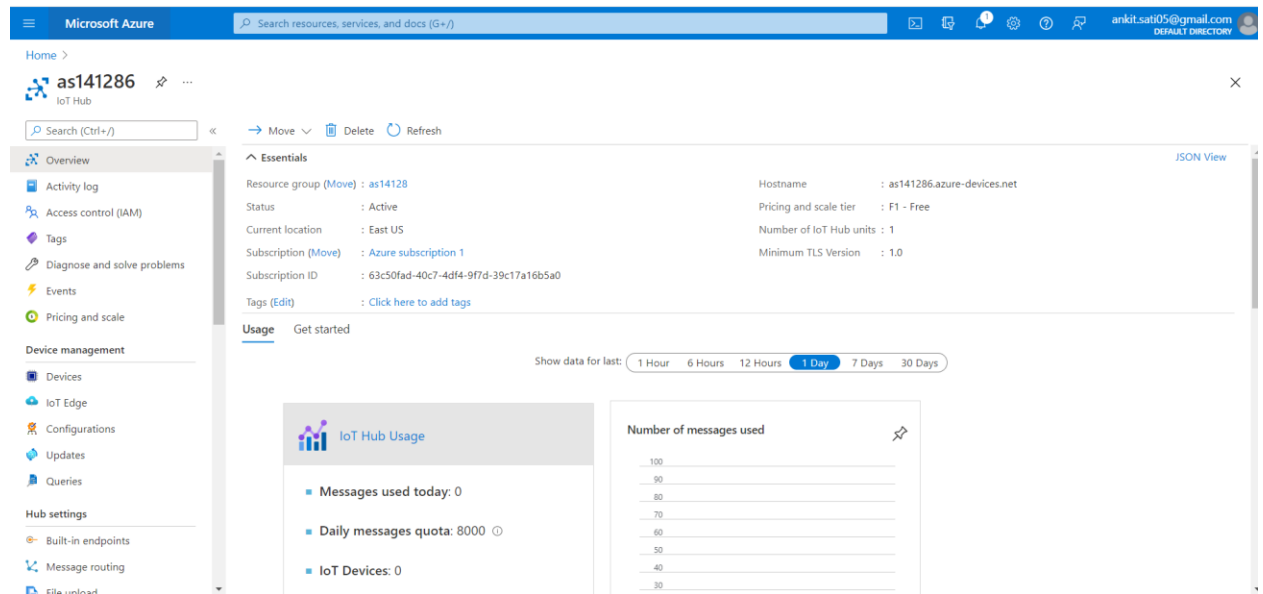
- Single hop computations
- Frames spend less time over the network.
- Reduce the chance of frame drops.
- Infinite compute storage at the end of network
- Complex logic can be deployed at the end of the network.
- Privatized network for mission critical projects.
- Very high speeds.
- High Availability of resources at the end.
- Cloud space with specific features as per demand of end user.

Examples of technologies used in field.

- Enterprise level – mission critical projects.
- Private LTE networks.
- Multiple cloud space
- Smart agriculture and services.
- Robotics in enterprise products.
- Product development.
- Resource deployment for projects.

Deploy your first IoT Edge module to a Windows device

Part 1 – Create your IOT HUB.



Part 2 – Register an IoT Edge device (Screenshot attached)

hub name - as141286

resource group as14128

Connection Key - HostName=as141286.azure-devices.net;DeviceId=myEdgeDevice;SharedAccessKey=DUQ43decn2Rqt8D5u01ZEVDZ52gq7eIK8qdiUKkwBs=

```

ing extensions without prompt.
unrecognized arguments: --hub-as141286








Examples from AI knowledge base:
https://aka.ms/cli_ref
Read more about the command in reference docs
ankit@Azure:~$ az iot hub device-identity create --device-id myEdgeDevice --edge-enabled --hub-name {hub_name}
Unable to find IoT Hub: {hub_name} in current subscription 63c50fad-40c7-4df4-9f7d-39c17a16b5a0.
ankit@Azure:~$ az iot hub device-identity create --device-id myEdgeDevice --edge-enabled --hub-name as141286
{
  "authentication": {
    "symmetricKey": {
      "primaryKey": "DUQ43decM2Rqt8D5u01ZEVDZ52gq7e1K8qdiUKkwBs=",
      "secondaryKey": "8FKd5kHwVZs41/Zq8S36UF1z79LLmJ9U/QbuA5qKx1A="
    },
    "type": "sas",
    "x509Thumbprint": {
      "primaryThumbprint": null,
      "secondaryThumbprint": null
    }
  },
  "capabilities": {
    "iotEdge": true
  },
  "cloudToDeviceMessageCount": 0,
  "connectionState": "Disconnected",
  "connectionStateUpdatedTime": "0001-01-01T00:00:00",
  "deviceId": "myEdgeDevice",
  "deviceScope": "ms-azure-iot-edge://myEdgeDevice-637725333601409849",
  "etag": "MjIzOTY3MzAy",
  "generationId": "637725333601409849",
  "lastActivityTime": "0001-01-01T00:00:00",
  "parentScopes": [],
  "status": "enabled",
  "statusReason": null,
  "statusUpdatedTime": "0001-01-01T00:00:00"
}
ankit@Azure:~$ az iot hub device-identity connection-string show --device-id myEdgeDevice --hub-name as141286
{
  "connectionString": "HostName=as141286.azure-devices.net;DeviceId=myEdgeDevice;SharedAccessKey=DUQ43decM2Rqt8D5u01ZEVDZ52gq7e1K8qdiUKkwBs="
}
ankit@Azure:~$ ^C



```



Home > All resources > as141286 >



myEdgeDevice ...



as141286



 Save  Set modules  Manage child devices  Troubleshoot  Device twin  Manage keys  Refresh



Device ID  myEdgeDevice 


Primary Key  



Secondary Key  

Primary Connection String  

Secondary Connection String  

IoT Edge Runtime Response  NA 

Enable connection to IoT Hub  ☒ Enable ☐ Disable

Parent device  No parent device 

Modules IoT Edge hub connections Deployments and Configurations

Name	Type	Specified in Deployment	Reported by Device	Runtime Status	Exit Code
\$edgeAgent	Module Identity	NA	NA	NA	NA
\$edgeHub	Module Identity	NA	NA	NA	NA

Part 3- Install and start the IoT Edge runtime

```
Administrator: Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/powershell

PS C:\windows\system32> $msiPath = $([io.Path]::Combine($env:TEMP, 'AzureIoTEdge.msi'))
PS C:\windows\system32> $ProgressPreference = 'SilentlyContinue'
PS C:\windows\system32> Invoke-WebRequest "https://aka.ms/AzFlowMSI" -OutFile $msiPath

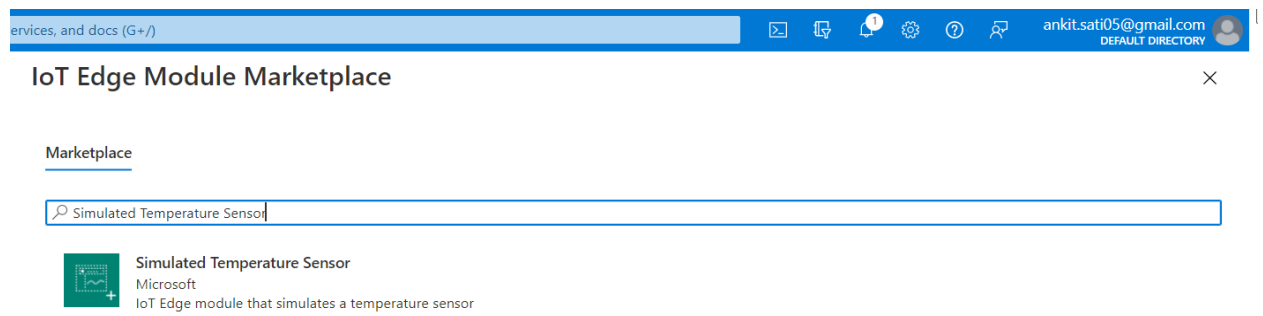
Status: 200
StatusDescription: OK
Content: {208, 207, 17, 224...}
RawContent: HTTP/1.1 200 OK
Content-Type: application/octet-stream
Date: Mon, 15 Nov 2022...
Headers: {[x-ms-blob-content-md5, Za5IX2JEE26hjmKk4AyMEw==], [Connection, keep-alive], [Accept-Ranges, bytes], [Content-Length, 562446336]...}
RawContentLength: 562446336

PS C:\windows\system32> Start-Process -Wait msixexec -ArgumentList "/I",$([io.Path]::Combine($env:TEMP, 'AzureIoTEdge.msi')),"/qn"
>>
PS C:\windows\system32> Get-ExecutionPolicy -List

Scope ExecutionPolicy
-----
MachinePolicy Undefined
UserPolicy Undefined
Process Undefined
CurrentUser Undefined
LocalMachine Restricted

PS C:\windows\system32> Set-ExecutionPolicy -ExecutionPolicy AllSigned -Force
```

Part 4- Deploy a module



Select routes

```
{
  "modulesContent": {
    "$edgeAgent": {
      "properties.desired": {
        "modules": {
          "SimulatedTemperatureSensor": {
            "settings": {
              "image": "mcr.microsoft.com/azureiotedge-simulated-temperature-sensor:1.0",
              "createOptions": ""
            },
            "type": "docker",
            "status": "running",
            "restartPolicy": "always",
            "version": "1.0"
          }
        },
        "runtime": {
          "settings": {
            "minDockerVersion": "v1.25"
          },
          "type": "docker"
        },
        "schemaVersion": "1.1",
        "systemModules": {
          "edgeAgent": {
            "settings": {
              "image": "mcr.microsoft.com/azureiotedge-agent:1.1",
              "createOptions": ""
            },
            "type": "docker"
          },
          "edgeHub": {
            "settings": {
              "image": "mcr.microsoft.com/azureiotedge-hub:1.1",
              "createOptions":
                "{\\\"HostConfig\\\":{\\\"PortBindings\\\":{\\\"443/tcp\\\":[{\\\"HostPort\\\":\\\"443\\\"}],\\\"5671/tcp\\\":[{\\\"HostPort\\\":\\\"5671\\\"}],\\\"8883/tcp\\\":[{\\\"HostPort\\\":\\\"8883\\\"}]}}}"
            },
            "type": "docker",
            "status": "running",
            "restartPolicy": "always"
          }
        }
      }
    }
  }
}
```

```

    }
  }
},
"$edgeHub": {
  "properties.desired": {
    "routes": {
      "route": "FROM /messages/* INTO $upstream",
      "SimulatedTemperatureSensorToIoTHub": "FROM
/messages/modules/SimulatedTemperatureSensor/* INTO $upstream"
    },
    "schemaVersion": "1.1",
    "storeAndForwardConfiguration": {
      "timeToLiveSecs": 7200
    }
  }
},
"SimulatedTemperatureSensor": {
  "properties.desired": {
    "SendData": true,
    "SendInterval": 5
  }
}
}
}

```

REVIEW AND CREATE

The screenshot shows the Microsoft Azure portal interface. At the top, there's a navigation bar with the Microsoft Azure logo and a search bar. Below the navigation bar, the breadcrumb trail reads: Home > as141286 > myEdgeDevice >. The main heading is 'Module Identity Details' with a subheading 'SimulatedTemperatureSensor'. Below the heading, there are links for 'Module Identity Twin', 'Direct method', 'Delete', and 'Refresh'. A message states: 'This module identity does not employ an authentication mechanism.' Below this, the 'Module Identity Name' is displayed as 'myEdgeDevice/SimulatedTemperatureSensor'. At the bottom, there's a section titled 'Configurations' with a subheading 'Module configurations associated with this module'.

PART 5 – Viewing the device and the data

Modules IoT Edge hub connections Deployments and Configurations					
Name	Type	Specified in Deployment	Reported by Device	Runtime Status	Exit Code
\$edgeAgent	Module Identity	NA	NA	NA	NA
\$edgeHub	Module Identity	NA	NA	NA	NA
SimulatedTemperatureSensor	Module Identity	NA	NA	NA	NA

Monitor the data

NAME	STATUS	DESCRIPTION	CONFIG
SimulatedTemperatureSensor	running	Up 3 hours	mcr.microsoft.com/azureiotedge-simulated-temperature-sensor:1.0
edgeAgent	running	Up 3 hours	mcr.microsoft.com/azureiotedge-agent:1.0
edgeHub	running	Up 3 hours	mcr.microsoft.com/azureiotedge-hub:1.0

FINAL PART _ CLEANING UP THE RESOURCES

Final screenshot before deletion

Microsoft Azure Search resources, services, and docs (G+)

Azure services

- Create a resource
- Resource groups
- All resources
- Quickstart Center
- Virtual machines
- App Services
- Storage accounts
- SQL databases
- Azure Cosmos DB
- More services

Recent resources

Name	Type	Last Viewed
as141286	IoT Hub	11 minutes ago
as14128	Resource group	26 minutes ago

Navigate

- Subscriptions
- Resource groups
- All resources
- Dashboard

Tools

- Microsoft Learn Learn Azure with free online training from Microsoft
- Azure Monitor Monitor your apps and infrastructure
- Security Center Secure your apps and infrastructure
- Cost Management Analyze and optimize your cloud spend for free

Post Deletion

Home >

Resource not found



Resource not found

[Get support](#) [Perform self-diagnostics](#)

da72b6da062e466ab15d65d358039f13

/subscriptions/63c50fad-40c7-4df4-9f7d-39c17a16b5a...

Extension	Content
HubsExtension	ResourceMenuBlade

Error code
404

Details
The resource was not found, it may have been deleted. If this was launched from a pinned tile on the dashboard, it should be removed.

Resource ID: /subscriptions/63c50fad-40c7-4df4-9f7d-39c17a16b5a0/resourcegroups/as14128/providers/Microsoft.Devices/IotHubs/as141286

Status Code: 404

Status Message: Resource group 'as14128' could not be found.

Activity log snippet

Notifications



[More events in the activity log →](#)

[Dismiss all](#)

✓ Deleted resource group as14128



Deleted resource group as14128

2 minutes ago

✓ Set Modules



Successfully updated IoT Edge settings for device myEdgeDevice.

13 minutes ago

✓ Deployment succeeded



Deployment 'as141286-1114192541' to resource group 'as14128' was successful.

[Go to resource](#)

[Pin to dashboard](#)

53 minutes ago

EXTRA CREDIT

Tutorial 1 - Monitor IoT Edge devices

Step 1 - Create a Log Analytics workspace

Workspace ID : b2050b98-4b47-4ee2-8a19-7c6d7b6f65b1

Delete

Essentials

Resource group (Move) : newgroup

Status : Active

Location : East US

Subscription (Move) : Azure subscription 1

Subscription ID : 63c50fad-40c7-4df4-9f7d-39c17a16b5a0

Tags (Edit) : Click here to add tags

Workspace Name : as14128

Workspace ID : b2050b98-4b47-4ee2-8a19-7c6d7b6f65b1

Pricing tier : Pay-as-you-go

Access control mode : Use resource or workspace permissions

Operational issues : OK

Get started with Log Analytics

Step 2 - Create a Log Analytics workspace

ID - b7f18d8e-271b-442d-8714-ee5d59e0159e

as14128
IoT Hub

Resource ID

Security settings

Identity

Support + troubleshooting

Resource health

System-assigned User-assigned

A system-assigned managed identity enables Azure resources to authenticate to cloud services (e.g. Azure Key Vault) without storing credentials in code. Once enabled, all necessary permissions can be granted via Azure role-based-access-control. The lifecycle of this type of managed identity is tied to the lifecycle of this resource. Additionally, each resource (e.g. Virtual Machine) can only have one system-assigned managed identity. [Learn more about Managed identities.](#)

Save Discard Refresh Feedback

Status

On

Off

This resource is registered with Azure Active Directory. You can control its access to services like Azure Resource Manager, Azure Key Vault, etc. [Learn more](#)

Object ID

b7f18d8e-271b-442d-8714-ee5d59e0159e

Permissions

Azure role assignments

Step 3 - Deploy the metrics collector module

Dashboard > as14128 >

as14128

as14128

Save Set modules Manage child devices Troubleshoot Device twin Manage keys Refresh

Device ID

as14128

Primary Key

Secondary Key

Primary Connection String

Secondary Connection String

IoT Edge Runtime Response

NA

Enable connection to IoT Hub

☒ Enable ☐ Disable

Parent device

No parent device

Modules

IoT Edge hub connections

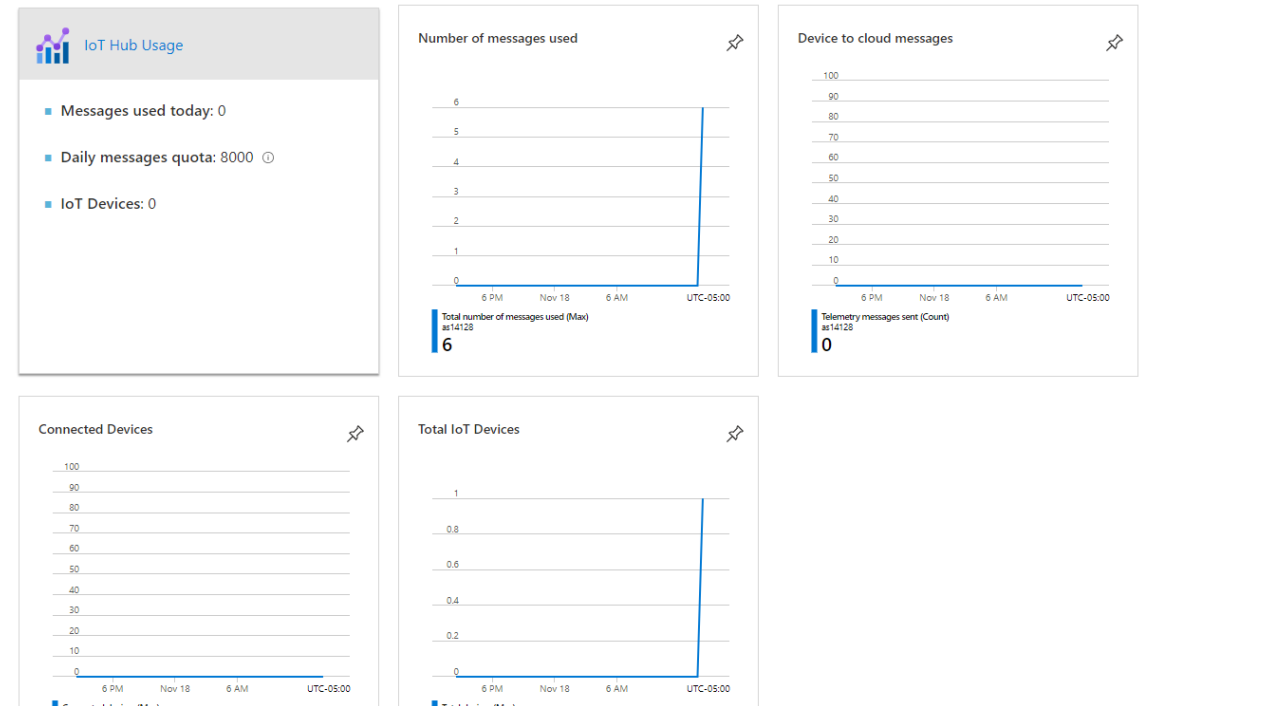
Deployments and Configurations

Client ID

Status

There are no entries for this device.

Step 4 - Explore the fleet view and health snapshot workbooks

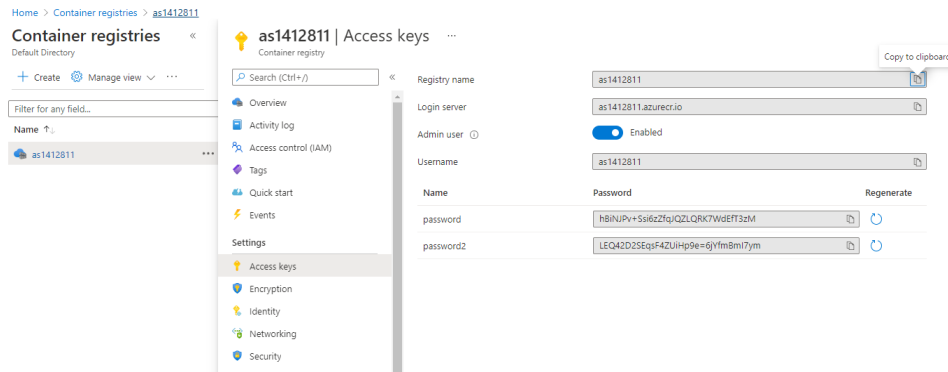


Tutorial 2 - Develop IoT Edge modules using Windows containers

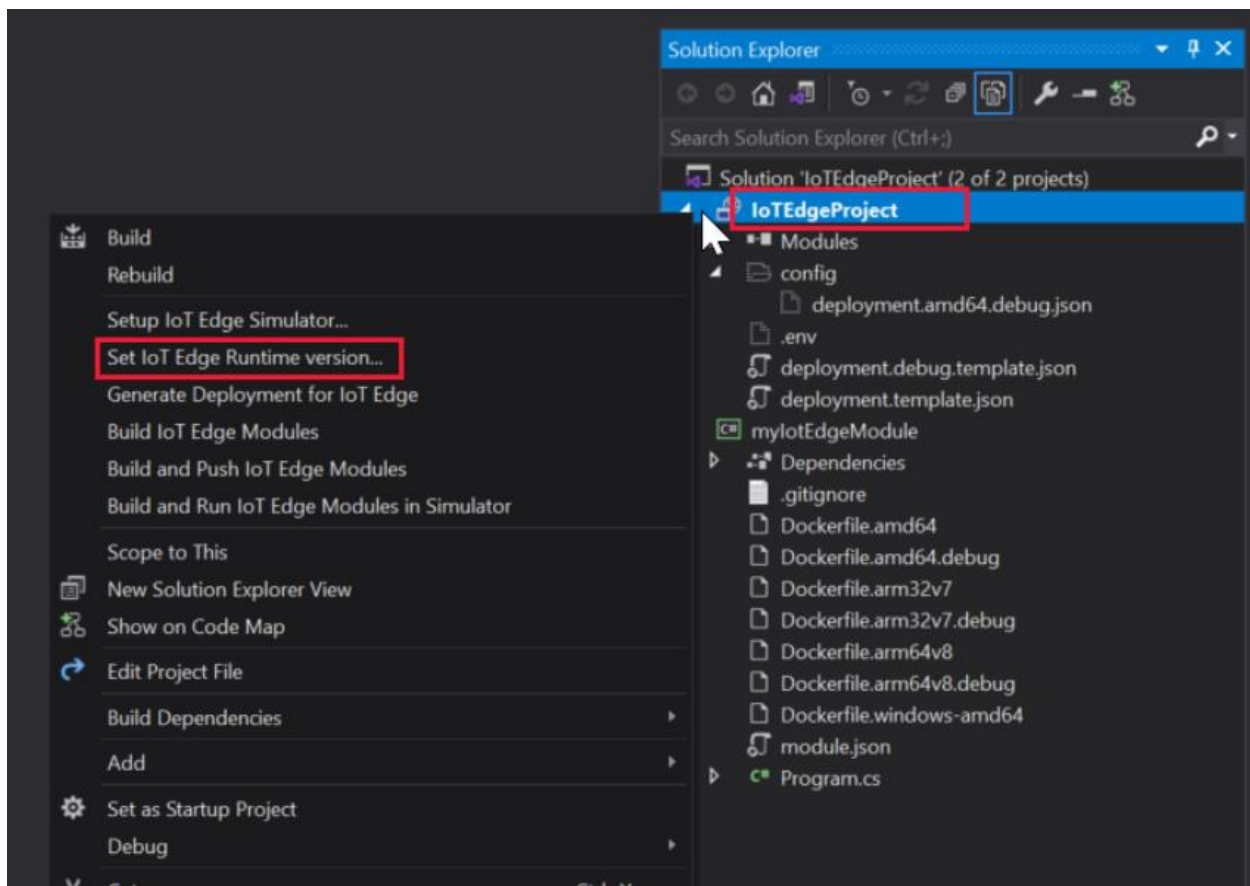
Step 1 - Set up VS

Installing environment and repos.

Step -2 – Create a container registry



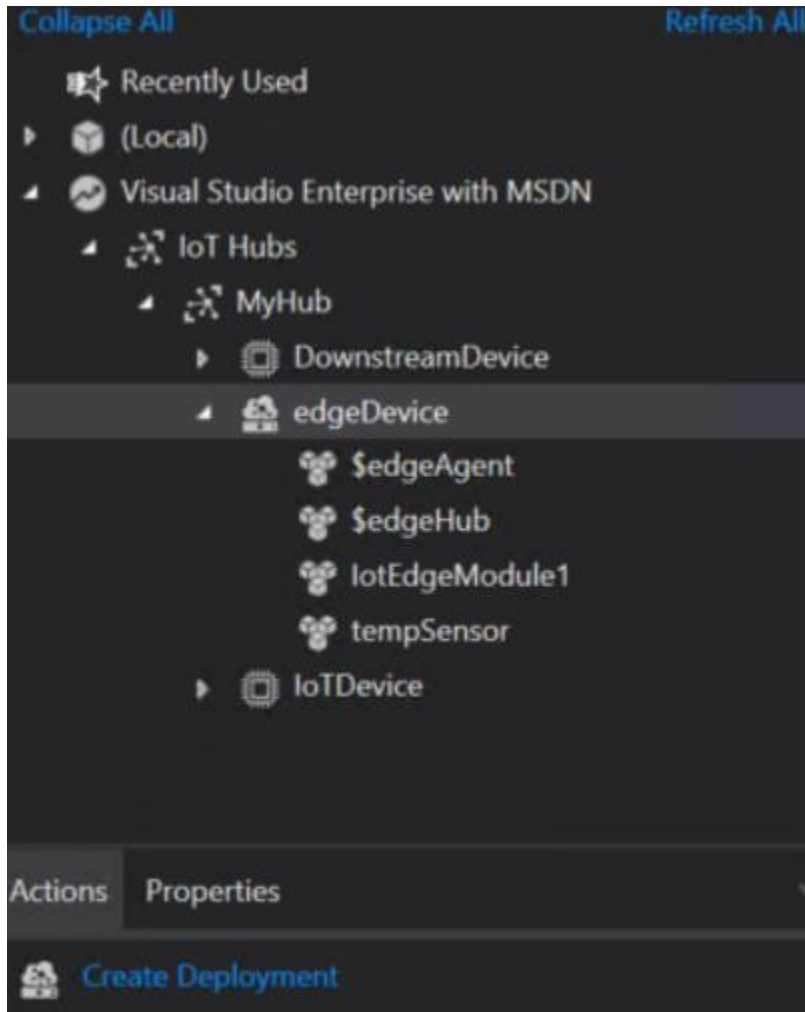
Step 3 – Edge Runtime



Step 4 - Provide your registry credentials to the IoT Edge agent

```
"registryCredentials": {  
  "<registry name>": {  
    "username": "$CONTAINER_REGISTRY_USERNAME_<registry name>",  
    "password": "$CONTAINER_REGISTRY_PASSWORD_<registry name>",  
    "address": "<registry name>.azurecr.io"  
  }  
}
```

Step 5 - Changes on Device



STEP 6 –

Clean up resource

- Resources – Done
- Modules – Done
- Devices done

Tutorial 3 – Custom code

Step 1 - Set up Java

Installing environment and repos.

Step 2 – Code applet

```
4 import java.util.Map;
5
6 import javax.json.Json;
7 import javax.json.JsonObject;
8 import javax.json.JsonReader;
9
10 import com.microsoft.azure.sdk.iot.device.DeviceTwin.Pair;
11 import com.microsoft.azure.sdk.iot.device.DeviceTwin.Property;
12 import com.microsoft.azure.sdk.iot.device.DeviceTwin.TwinPropertyCallback;
13
14 private static final String TEMP_THRESHOLD = "TemperatureThreshold";
15 private static AtomicLong tempThreshold = new AtomicLong(25);
16
17 protected static class MessageCallbackMqtt implements MessageCallback {
18     private int counter = 0;
19     @Override
20     public IoTHubMessageResult execute(Message msg, Object context) {
21         this.counter += 1;
22
23         String msgString = new String(msg.getBytes(), Message.DEFAULT_IOTHUB_MESSAGE_CHARSET);
24         System.out.println(
25             String.format("Received message %d: %s",
26                 this.counter, msgString));
27         if (context instanceof ModuleClient) {
28             try (JsonReader jsonReader = Json.createReader(new StringReader(msgString))) {
29                 final JsonObject msgObject = jsonReader.readObject();
30                 double temperature = msgObject.getJsonObject("machine").getJsonNumber("temperature").doubleValue();
31                 long threshold = App.tempThreshold.get();
32                 if (temperature >= threshold) {
33                     ModuleClient client = (ModuleClient) context;
34                     System.out.println(
35                         String.format("Temperature above threshold %d. Sending message: %s",
36                             threshold, msgString));
37                     client.sendEventAsync(msg, eventCallback, msg, App.OUTPUT_NAME);
38                 }
39             }
40         }
41     }
42 }
```

Step 3 – Push the custom modules

Step 4 - Deploy modules to device

All resources
Azure subscription 1

Refresh

as14128	IoT Hub
as14128	Log Analytics works...
cs210032001a644cba6	Storage account

Azure getting started made easy!

Launch an app of your choice on Azure in a few quick steps

Create DevOps Starter

Quickstarts + tutorials

Step 5 - Edit the module twin

```
import com.microsoft.azure.sdk.iot.device.DeviceTwin.TwinPropertyCallback;

private static final String TEMP_THRESHOLD = "TemperatureThreshold";
private static AtomicLong tempThreshold = new AtomicLong(5);

protected static class MessageCallbackMqtt implements MessageCallback {
    private int counter = 0;
```

Step 6 – Delete all IOT edge modules.

- Resources – Done
- Modules – Done
- Devices done