How to Enable Microsoft IoT Edge and Docker on a Secure Connector

https://campus.barracuda.com/doc/91987807/

The new enhanced container feature for LXC, Docker, and IoT Edge is currently in EA and thus free of charge until at least December 2020. Starting January 2021 we will make this new functionality available for purchase and all existing deployments will need to be licensed.

On Secure Connector version 2.0.9, you can configure container engines that allow you to easily perform Edge computing capabilities on the FSC2 such as data aggregation, normalization, and offline logic. The SC 2.0.9 container configuration supports the following options:

- **Azure IoT Edge** – Azure IoT Edge extends IoT Hub. Analyze device data locally instead of in the cloud in order to send less data to the cloud, react to events quickly, and operate offline. Installing this runtime engine on the Secure Connector allows you to manage Edge modules from your Azure portal. In addition, Microsoft provides a list of IoT Edge modules in their marketplace that you can use out of the box for various scenarios. Microsoft IoT Edge provides a cost-savings solution for your business, allowing maximum flexibility even in offline mode, as the latest state of your edge device gets automatically synchronized after reconnecting to the cloud. Barracuda SC2 running firmware version 2.0.9 is Azure IoT Edge certified.

For more Information on Azure IoT Edge, please refer to https://docs.microsoft.com/en-us/azure/iot-edge/

For getting started with Azure IoT Edge on the Secure Connector, please refer to https://catalog.azureiotsolutions.com/details?title=Barracuda-FSC2&source=null

- **Docker Engine** – Docker Engine is a client-server application designed for building and containerizing applications and managing objects such as networks, containers, images, and volumes. A built-in command line interface uses APIs to interact via scripting and commands.

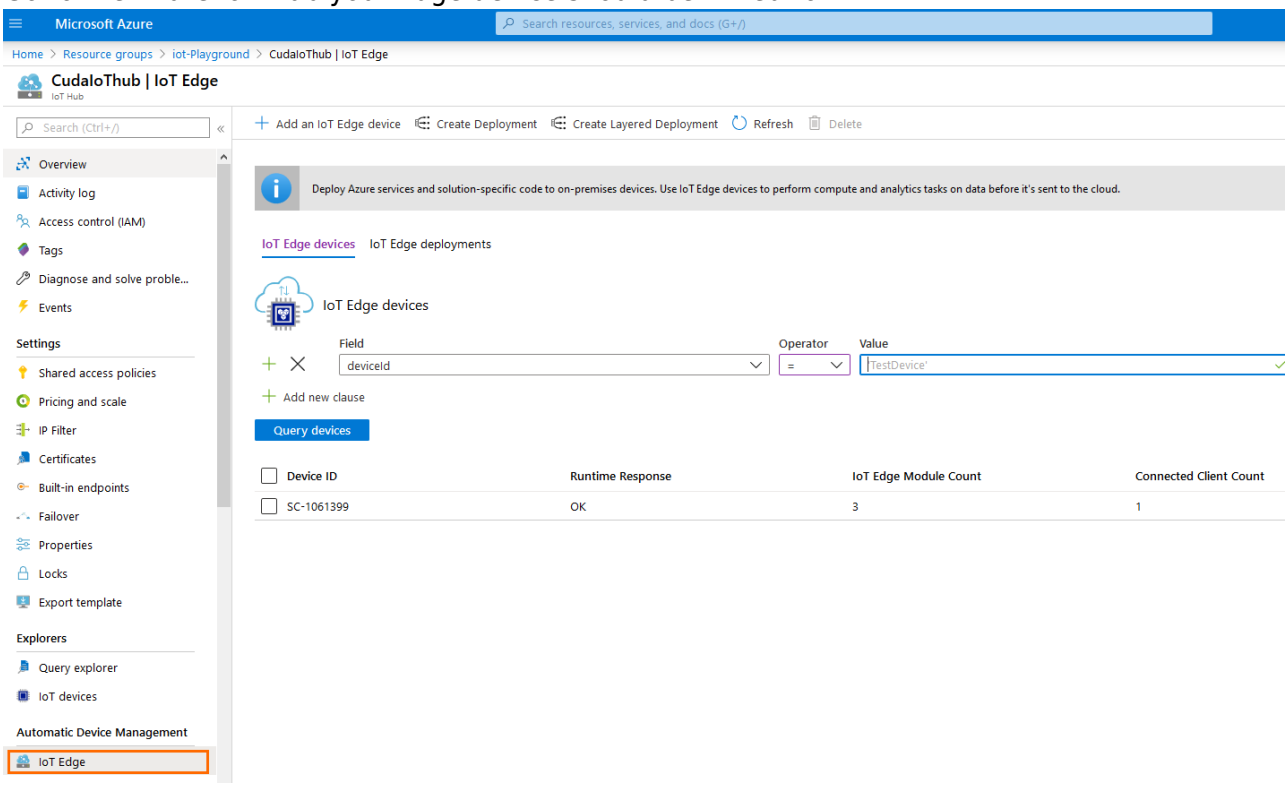
Configure Azure IoT Edge

Configure Azure IoT Edge as container engine on a Secure Connector.

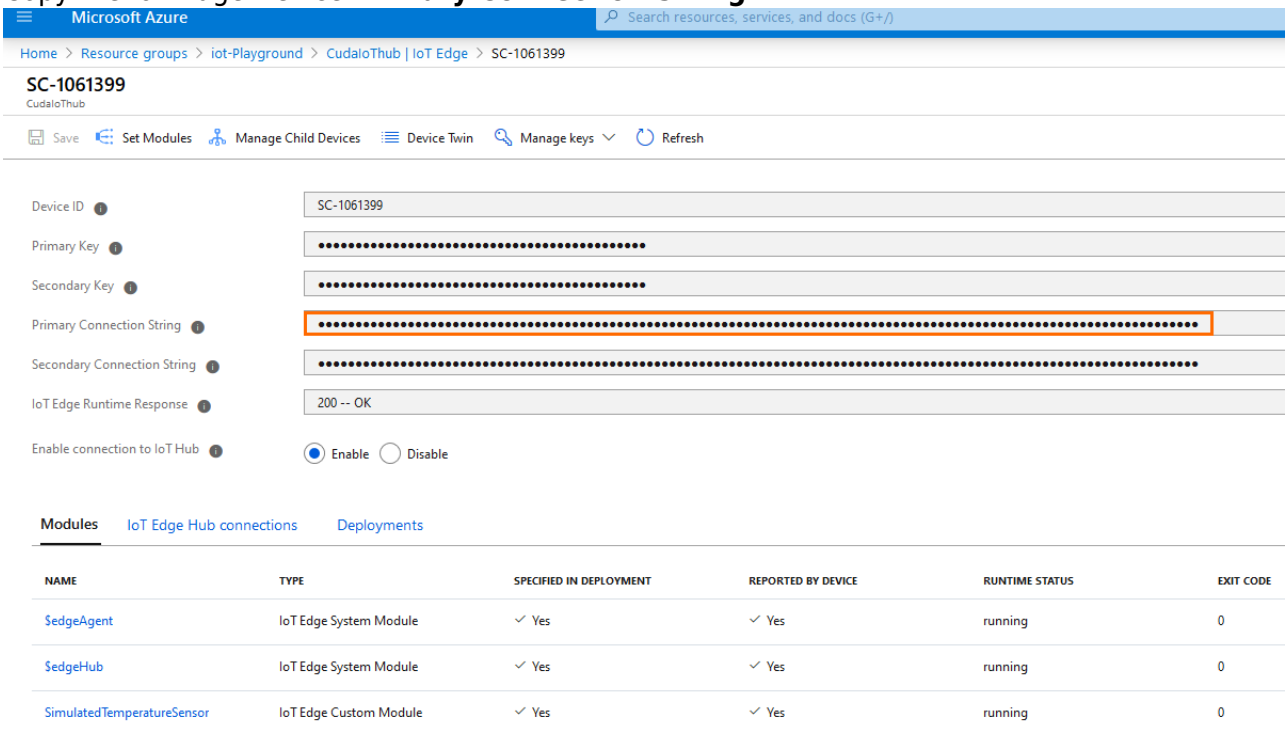
Before You Begin

Configuring Azure IoT Edge requires the IoT Edge Device primary connection string. To access your device on Azure IoT Edge, do the following:

1. Log into the Azure portal: https://portal.azure.com
2. Go to the Azure IoT Hub your Edge device should be linked to:



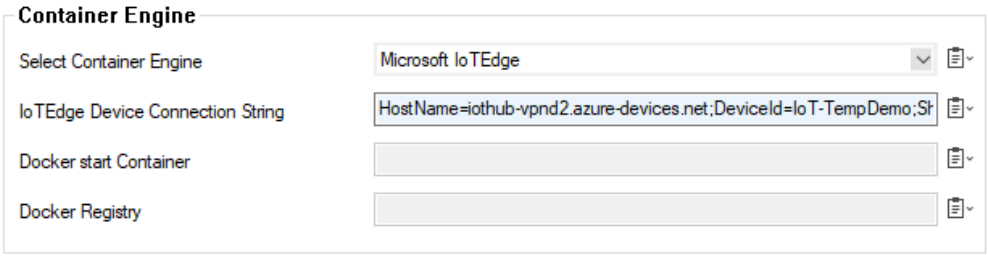
3. Copy the IoT Edge Device **Primary Connection String**.



Configure the Container Engine

Configure Microsoft IoT Edge on the Secure Connector.

1. On the Control Center, go to ***your cluster*** **> Cluster Settings > Secure Connector Editor**.
2. Click **Lock**.
3. Double-click to edit the device or Secure Connector.
4. Make sure the container is enabled and a root password is set. For more information, see Secure Connector Container.
5. Select **Microsoft IoT Edge** as the **Container Engine**.
6. Paste the **IoT Edge Device Connection String**.



7. Click **OK**.
8. Click **Send Changes** and **Activate**.

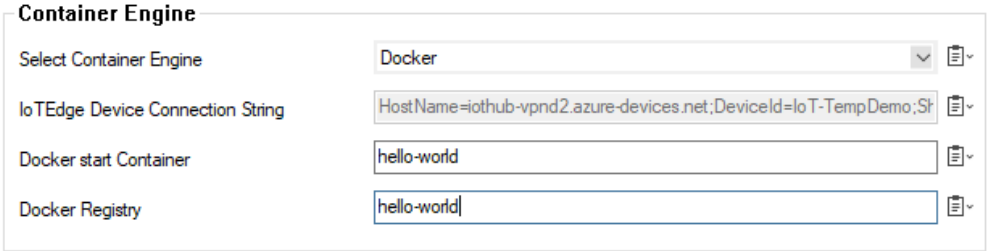
It will take a couple of minutes until the runtime is installed and configured.

The activity is logged on the Secure Connector in **/var/log/changevirtualizationengine.log**. To verify successful installation, log into the container via SSH and run the **iotedge check** command.

Configure Docker

Configure Microsoft Docker as container engine on a Secure Connector.

1. On the Control Center, go to ***your cluster*** **> Cluster Settings > Secure Connector Editor**.
2. Click **Lock**.
3. Double-click to edit the device or Secure Connector template.
4. Make sure the container is enabled and a root password is set. For more information, see Secure Connector Container.
5. Select **Docker** as the **Container Engine**. Provide the information for start container as well as for the Docker Registry the container must be pulled from.



6. Click **OK**.

7. Click **Send Changes** and **Activate**.

It will take a couple of minutes until the runtime is installed and configured.

The activity is logged on the Secure Connector in **/var/log/changevirtualizationengine.log**. To verify successful installation, log into the container via SSH and run the **docker ps** command.

Next Step

Create a Secure Connector firewall management rule to allow SSH access into the **container** zone. For more information, see Secure Connector Container.

Figures

1. container_az_01.png
2. container_az_02.png
3. container_az_03.png
4. container_doc_01.png