Connect FLEX-BX200AI device to your Azure IoT services

Table of Contents

Introduction

About this document

This document describes how to connect FLEX-BX200AI to Azure IoT Hub using the Azure IoT Explorer with certified device application and device models.

IoT Plug and Play certified device simplifies the process of building devices without custom device code. Using Solution builders can integrated quickly using the certified IoT Plug and Play enabled device based on Azure IoT Central as well as third-party solutions.

This getting started guide provides step by step instruction on getting the device provisioned to Azure IoT Hub using Device Provisioning Service (DPS) and using Azure IoT Explorer to interact with device's capabilities.

The FLEX-BX200 is an AI hardware ready system ideal for deep learning inference computing to help you get faster, deeper insights into your customers and your business. IEI’s FLEX-BX200 supports graphics cards, Intel FPGA acceleration cards, and Intel VPU acceleration cards, and provides additional computational power plus end-to-end solution to run your tasks more efficiently. With the NVIDIA TensorRT, QNAP QuAI, and Intel OpenVINO AI development toolkit, it can help you deploy your solutions faster than ever.

Prerequisites

You should have the following items ready before beginning the process:

For Azure IoT Central

- Azure Account
- Azure IoT Central application

For Azure IoT Hub

- Azure IoT Hub Instance
- Azure IoT Hub Device Provisioning Service
- Azure IoT Public Model Repository

Prepare the Device.

Hardware Environmental setup

- Prepare FLEX-BX200AI.
- Connect the power adapter with FLEX-BX200AI.
- Power on the FLEX-BX200AI.
- Connect to the network.

Software Environmental setup

- Download the source code from this GitHub and check the "Azure-IoTHub-general-device-main" folder.
- Install python and make sure the Python environment is ready (more than V.3.7.3).
- Install Git for Windows

Prepare IoT Hub and DPS configuration

Please refer to this tutorial to complete the following procedures :

1. Use Azure commands or Azure portal to create a Resource Group 、an Iot Hub and a Device Provisioning Service
2. To link the DPS instance to your IoT hub
3. To create your device by individual device enrollment in your DPS instance.
4. Make a note of the DPS information (DPS endpoint/Registration ID/ID Scope/Symmetric key).
5. Set the DPS information got from the former step in "\Azure-IoTHub-general-device-main\run.sh".

```
export IOTHUB_DEVICE_SECURITY_TYPE="DPS"
export IOTHUB_DEVICE_DPS_ENDPOINT=""
export IOTHUB_DEVICE_DPS_ID_SCOPE=""
export IOTHUB_DEVICE_DPS_DEVICE_ID=""
export IOTHUB_DEVICE_DPS_DEVICE_KEY=""
```

Run the sample

Under the "Azure-IoTHub-general-device-main" folder, execute run.sh by git shell (git for Windows).

Connect to Azure IoT Central

1. Create an application

Please refer to this tutorial to create a "Custom application" template.

2. Create a device template from the device catalog

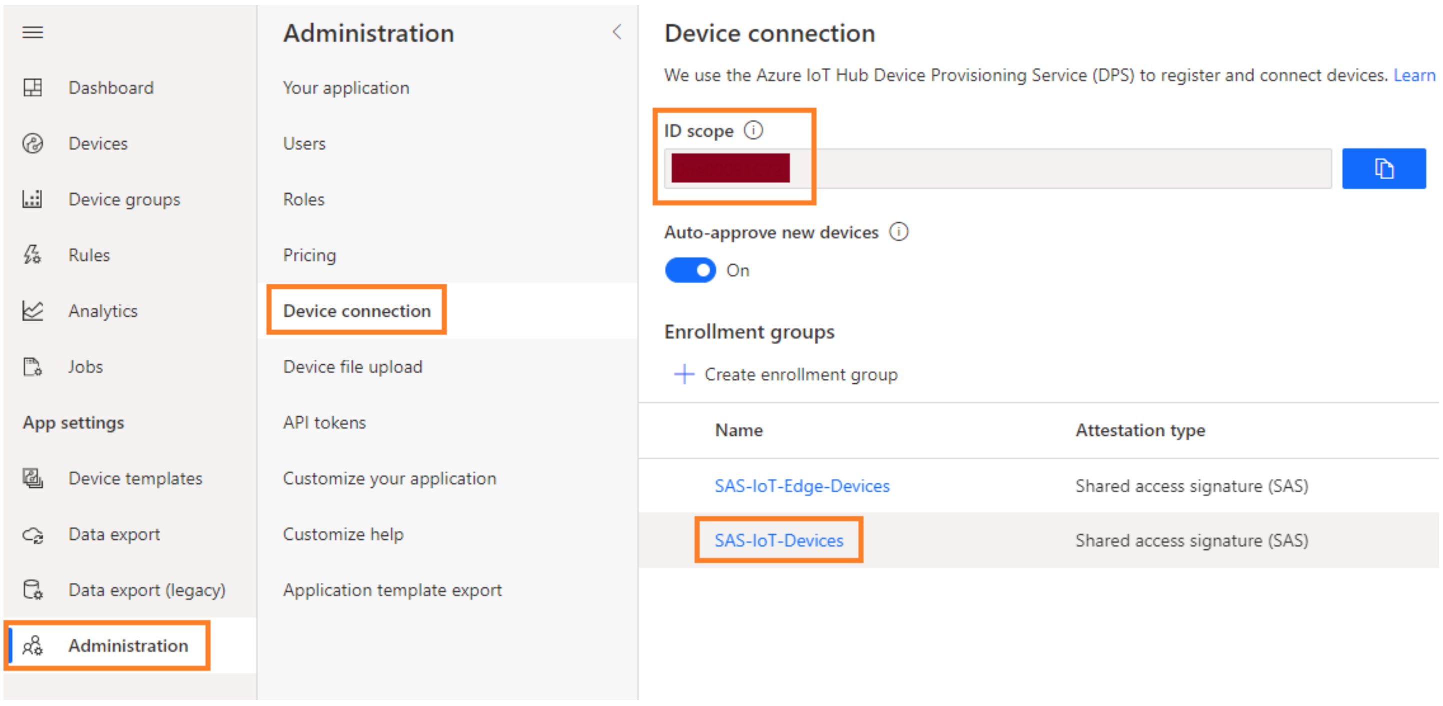
Please refer to this tutorial to create the {enter your device name here} device template.

3. Add a device

Add a new device under FLEX-BX200AI device template. Make a note of the device ID.

4. Get connection information

- 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.



Device connection

We use the Azure IoT Hub Device Provisioning Service (DPS) to register and connect devices. Learn

ID scope

[redacted]

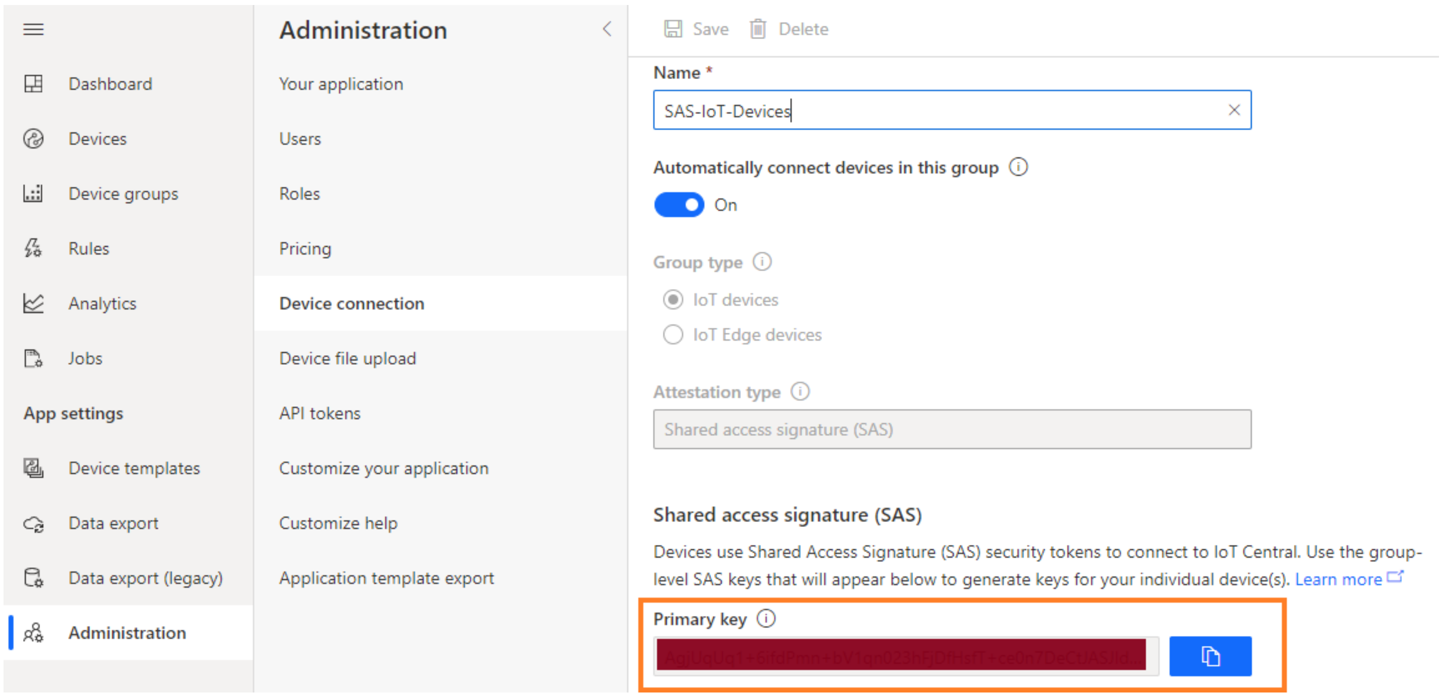
Auto-approve new devices

On

Enrollment groups

+ Create enrollment group

Name	Attestation type
SAS-IoT-Edge-Devices	Shared access signature (SAS)
SAS-IoT-Devices	Shared access signature (SAS)



Name *
SAS-IoT-Devices

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
[redacted]

Use the Cloud Shell to generate a device specific key from the group SAS key you just retrieved using the Azure CLI.

> az extension add --name azure-iot
> az iot central device compute-device-key --device-id --pk

Make a note of the generated device key, and the ID scope for this application and set it to "\Azure-IoTHub-general-device-main\run.sh".

Additional Links

Please refer to the below link for additional information for Plug and Play

- Manage cloud device messaging with Azure-IoT-Explorer
- Import the Plug and Play model
- Configure to connect to IoT Hub
- How to use IoT Explorer to interact with the device