

Design a CI/CD pipeline for IoT Edge

6 minutes

An automated CI/CD strategy addresses two main problems.

- Creating a smoke-testing strategy for Azure IoT Edge devices.
- Integrate code from several developers as an automated process through a DevOps (continuous integration and continuous delivery) strategy.

In this unit, we'll plan the sequence of steps needed to implement a solution to this problem statement.

Benefits of a testing strategy

Designing an automated smoke-testing strategy provides many benefits.

- Deploying to virtual IoT devices is cheaper than deploying to physical IoT devices.
- A successful smoke test ensures that the program is ready for further testing.
- Automated smoke testing avoids expensive rework for more sophisticated tests
- Optimizing the time for your developer and the test teams

To implement this solution, you'll create a smoke test as a subset of your main functionality. The test should be self-scoring, automated, and able to run on virtual IoT devices. Also, the test should incorporate the principles of DevOps (continuous integration and continuous delivery) for IoT Edge devices.

Testing solution components

You'll use the following components in the overall design:

- **Azure Container Registry:** A private Docker registry service used to store published IoT Edge modules.
- **Azure IoT Hub Device Provisioning Service:** Allows for automatic provisioning of IoT devices in a secure manner.

- **Azure IoT Hub:** Service that enables us to connect, monitor, and manage IoT devices securely.
- **Azure Pipelines:** A service that automates your builds and deployments.

Test script requirements

To implement this solution, you'll create a release pipeline that deploys to QA devices. You will also smoke test the edge runtime in a containerized device. The design will run an instance of [azure-iot-edge-device-container](#), which is configured as a QA device. The tests will probe the IoT hub to ensure the QA device receives the correct deployment configuration and can run all configured modules successfully.

The script for the smoke test implements the following:

- Test for the existence of the device(s).
- Test if the device is connected to the IoT hub.
- Test if the device applied a new deployment manifest.
- Test if the modules specified in the deployment manifest are running successfully.

Test script steps

The steps involved are:

1. Create Azure resources.
2. Set up Azure DevOps Services.
3. Set up continuous integration.
4. Create a release pipeline with a smoke test.
5. Add integration tests to a release pipeline.

Module complete:

Unlock achievement