

Smart Data Transaction for CPS R7 Release Notes

Release Notes for the Smart Data Transaction for CPS R7 Blueprint

Summary

This is the second release of the Smart Data Transaction for CPS (Cyber-Physical Systems) Blueprint. The blueprint is focused on the following goals:

- Integrating EdgeX device-camera-go and IP camera into SDT blueprint to enable edge nodes to receive image data from Camera
- Sharing image data between edge nodes
- Analyzing image data on edge nodes, and comparing image analysis results

In addition, this release provides a basic framework for installing, configuring and monitoring the collection of edge nodes, using Kubernetes and Ansible scripting.

Release Content

The following components are new with this release:

- image-app: An application micro-service built with EdgeX Foundry for analyzing image data and comparing image analysis results
- device-camera: A device service built with EdgeX Foundry supporting communication with IP Camera

The dependencies of this release are summarized below:

- Ubuntu 20.04
- Kubernetes 1.22.9
- EdgeX Foundry release 2.1
- Docker 20.10.12
- Flannel 0.17.0 (CNI plugin 1.0.1)
- Ansible 2.11.9

There are no upstream code changes.

Upgrade Procedures

The steps to upgrade from the previous version to the current version are as follows

- Upgrade the source code to Release7, then modify the `deploy/playbook/group_vars/all/edgex.yml` file and files in `deploy/playbook/host_vars` directory according to the Installation Guide.
- Run the `edgex_stop.yml` playbook to stop the EdgeX services.
- Run the `build_images.yml` and `push_images.yml` playbooks to update the docker images in the private registry.
- Run the `edge_install.yml` playbook to update the default docker runtime of edge nodes.
- Run the `edgex_start.yml` playbook to start the Edgex services.

Release Data

Module Version Changes

All modules included with this release are version 1.1.

Document Version Changes

All documents included with this release are version 1.1.

Fixed Issues and Bugs

No existing issues and bugs are fixed in this release.

Enhancements

This release introduces support for IP cameras and sharing of static images via the synchronization application. Support for GPU acceleration of selected applications on the edge nodes has been added. A new application is introduced which analyses static images to perform shape detection.

Version Change

This is release 1.1 of this blueprint.

Deliverable

The deliverables consist of a Git repository containing all the source code for the new modules and script files for installation and operation of the blueprint, and the documents:

- Release Notes: This document
- [Blueprint Architecture Document](#)
- [Installation Guide](#)
- [Test Document](#)

Known Limitations, Issues and Workarounds

System Limitations

The following limitations are intended to be addressed in later releases:

- The new microservice image-app which is limited by the base image NVIDIA CUDA only supports arm architecture.
- Data sharing using sync-app is currently based on simple configuration variables. It should use the rules engine.
- Edge nodes cannot be added or removed dynamically without stopping the entire cluster.
- The MQTT communication channel should use TLS.

Known Issues

The following known issues exist in the EdgeX Foundry implementation and impact this blueprint (note that these issues may have been addressed in later releases of the upstream code):

- [edgex-go issue 3851](#)
- [edgex-go issue 3852](#)
- A similar issue that hides the Kong (API gateway) application's `/usr/kong/lib` directory, leading to the process crashing
- An issue with Kong's DNS processing that makes it fail to resolve the "localhost" address

These issues are all addressed by workarounds described below.

Workarounds

The `edgex-go` issues 3851 and 3852, as well as the issue hiding Kong's `lib` directory, are addressed by changes to the volume mappings of containers, found in the configuration files under `edgex/deployments`, `edgex/claims`, and `edgex/volumes`, and patches to the EdgeX Foundry `security-bootstrapper` and `security-secretstore-setup` Docker images. The patched images are generated using the `Makefile` and `Dockerfile` in the directories under `edgex` named after the affected images.

The Kong DNS name resolution problem is worked around with a patched Kong image, generated from the source under `edgex/kong`.

With these workarounds in place the blueprint can be installed and operated successfully according to the Installation Guide.

References

- [Blueprint Architecture Document](#)
- [Installation Guide](#)
- [Test Document](#)