Solution Brief

Intel® Ethernet Operator – Overview

Automated lifecycle management for Intel® Ethernet network interface cards.

Authors

Abdul Halim

Emma Kenny

Gershon Schatzberg

Richard Walsh

Executive Summary

This document provides a description of the Intel® Ethernet Operator (often abbreviated as IEO). IEO provides an automated means to configure and manage the lifecycle of Intel Ethernet network interface cards in a repeatable, scalable manner, thereby simplifying their deployment in a cloud environment.

This document is part of the Network Transformation Experience Kit, which is available at Network Transformation Experience Kits.

Introduction

Kubernetes Resource Operators

From a user's perspective, the primary goal of an operator is to abstract away the complexities of managing its resource through the common Kubernetes API. Kubernetes resource operators are resource-specific controllers designed to configure and manage complex resources for the automation and scaling of Kubernetes clusters. Operators are custom Kubernetes controllers that implement control loops to continuously compare the desired state of resources to their actual state and, if needs be, take action to bring them into alignment. Effectively, resource operators with resource-specific knowledge automate the entire lifecycle of the resources they manage.

Devices Supported by Intel Ethernet Operator

The Intel Ethernet Operator manages the configuration of capabilities supported in the Intel E810 series network interface cards (NICs). The Intel Ethernet Operator supports the following E810 series devices:

- Intel® Ethernet Network Adapter E810-CQDA1/CQDA2
- Intel® Ethernet Network Adapter E810-XXVDA4
- Intel® Ethernet Network Adapter E810-XXVDA2

Intel Ethernet Operator

Figure 1 shows the architecture of the Intel Ethernet Operator.

Functionality

The Intel Ethernet Operator supports:

- Update of the devices' firmware (FW) via Intel® Ethernet NVM Update Tool.
- Update of the devices' Dynamic Device Personalization (DDP) profile.
- Flow configuration of traffic handling for the devices, based on supported DDP profile.

Before deploying the operator, several dependencies (ICE driver, SR-IOV network operator, node feature discovery [NFD]) must be met. These dependencies are listed in the Prerequisites section.

Separate Custom Resources (CR) are provided for the FW/DDP update functionality and the Flow Configuration functionality. The operator constantly monitors the state of the CRs to detect any changes and acts based on the changes detected.

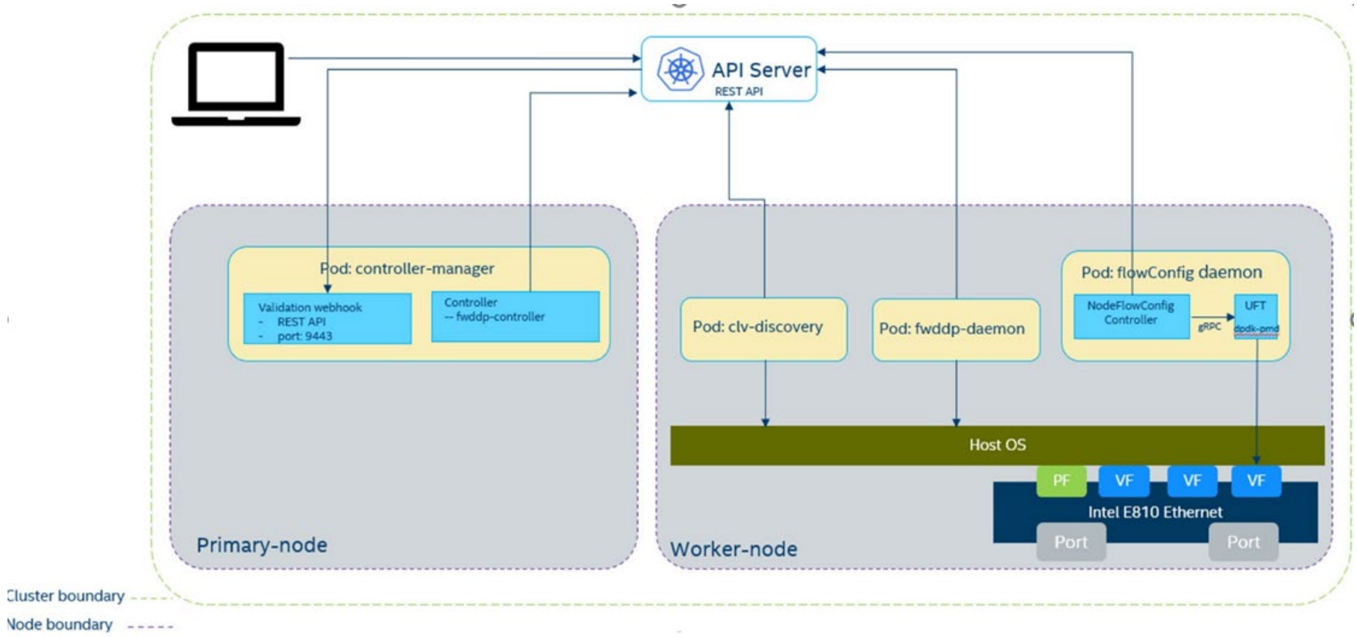

Figure 1. Intel Ethernet Operator Architecture

Controller-Manager

The controller-manager pod is the primary pod of the operator. It is responsible for the deployment of other assets, exposing the APIs, handling of the CRs, and executing the validation webhook. It contains the logic for accepting and splitting the FW/DDP CRs into node CRs and reconciling the status of each CR.

The validation webhook of the controller manager is responsible for checking each CR for invalid arguments.

Device Discovery

The CLV-discovery pod is a DaemonSet deployed on each worker node in the cluster. It is responsible for checking if supported hardware is present on the platform and labelling the node accordingly.

To get all the nodes containing the supported devices, run:

```
kubectl get EthernetNodeConfig -A
```

The result is similar to the following.

```
NAMESPACE                 NAME        UPDATE
intel-ethernet-operator   worker-1    InProgress
intel-ethernet-operator   worker-2    InProgress
```

To list supported devices found by the discovery pod run:

```
kubectl describe configmap supported-clv-devices -n intel-ethernet-operator
```

FW/DDP Daemon

The FW/DDP daemon pod is a DaemonSet deployed as part of the Intel Ethernet Operator. The DaemonSet is deployed on each node labeled with an appropriate label that indicates that a supported E810 Series NIC was detected on the platform. It is a reconcile loop that monitors the changes in each node's `EthernetNodeConfig` and acts on the changes. The logic implemented by this daemon takes care of updating the cards' NIC firmware and DDP profile. It is also responsible for draining the nodes, taking them out of commission, and rebooting when required by the update.

Firmware Update Functionality

After the Intel Ethernet Operator/daemon detects a change to a CR related to the update of the Intel® E810 NIC firmware, it tries to perform an update. The firmware for the Intel® E810 NICs is to be provided by the user in form of a tar.gz file. The user is also responsible to verify that the firmware version is compatible with the device. The user can download firmware and DDP profiles directly from Intel's download mirrors or simply use any accessible HTTP server and provide a URL for it in the CR. If the file is provided correctly and the firmware is to be updated, the Ethernet Configuration Daemon updates the Intel® E810 NICs with the NVM utility provided.

To update the NVM firmware of the Intel® E810 cards' NICs, the user must create a CR containing the information about which card should be programmed. The physical functions of the NICs are updated in logical pairs. The user needs to provide the FW URL and checksum (md5) in the CR.

Dynamic Device Personalization (DDP)

After the operator/daemon detects a change to a CR related to the update of the Intel® E810 DDP profile, it tries to perform an update. The DDP profile for the Intel® E810 NICs is to be provided by the user. The user is also responsible to verify that the DDP version is compatible with the device. If the DDP is to be updated, the Ethernet Configuration Daemon can update the DDP profile of Intel® E810 NICs by retrieving it from Intel's download mirrors, or by placing the profile in the correct location on the host. To update the DDP profile of the Intel® E810 NIC, the user must create a CR containing the information about which card should be programmed. All the physical functions of the NICs are updated for each NIC.

Flow Configuration

The Flow Configuration pod DaemonSet uses a CRD `FlowConfigNodeAgentDeployment` provided by the Intel Ethernet Operator after it is up and running. The required device config function (DCF) virtual function (VF) pools and their network attachment definitions are created with SR-IOV network operator APIs. It is deployed on each node that exposes a DCF VF pool as an extended node resource. It is a reconcile loop that monitors the changes in each node's CR and acts on the changes. The logic implemented into this daemon takes care of updating the cards' NIC traffic flow configuration. It consists of two components: Node Flow Config controller container and Unified Flow Tool (UFT) container.

Node Flow Configuration Controller

The Node Flow Configuration Controller monitors flow rules changes via a node-specific CRD `NodeFlowConfig` that matches the node name. After the operator/daemon detects a change to this CR related to the Intel® E810 flow configuration, it tries to create/delete rules via UFT over an internal gRPC call.

Unified Flow Tool

When a Flow Config change is required, the Node Flow Configuration Controller communicates with the UFT container that is running a DPDK-based DCF application. The UFT application accepts an input with the configuration and proceeds to program the device using a trusted VF created for this device. It is responsibility of the user to provide the trusted VFs as an allocatable K8s resource; see the Prerequisites section.

Prerequisites

The Intel Ethernet Operator has several prerequisites that must be met for complete functionality.

SR-IOV

To allow the Flow Configuration feature to compose the flow rules for the network card's traffic, the deployment must use a trusted virtual function (VF) from each physical function (PF). Usually it is the first VF (VF0) for each PF that has trust mode enabled and then bound to the `vfio-pci` driver. This VF pool must be created by the user and be allocatable as a Kubernetes resource. This VF pool is used exclusively by the UFT container and is not used by any application container.

For user applications, additional VF pools should be created separately as needed. The SR-IOV networking operator is automatically installed as a dependency of part of the IEO installation.

One way of creating and providing this trusted VF and application VFs is to configure it through the SR-IOV network operator. The configuration and creation of the trusted VFs and application is out of scope of the Intel Ethernet Operator and is the user's responsibility.

Hardware Validation Environment

- Intel® Ethernet Network Adapter E810-CDQA2
- 3rd Gen Intel® Xeon® Scalable processor

Summary

The Intel Ethernet Operator is a functional tool to manage the Intel® E810 NICs FW update and DDP profile, as well as the programming of the NICs' VFs flow configuration autonomously in a cloud native environment based on configurable user input.

Terminology

Table 1. Terminology

ABBREVIATION	DESCRIPTION
CR	Custom Resource
CRD	Custom Resource Definition
DCF	Device Config Function
DDP	Dynamic Device Personalization (DDP)
DPDK	Data Plane Development Kit
FW	Firmware
IEO	Intel® Ethernet Operator
NFD	Node Feature Discovery
NIC	Network Interface Card
PF	Physical Function
RPC	Remote Procedure Call
UFT	Unified Flow Tool
VF	Virtual Function

References

Table 2. References

TITLE	SOURCE
Intel® Ethernet Operator documentation	https://github.com/intel/intel-ethernet-operator/blob/main/docs/intel-ethernet-operator.md

Document Revision History

REVISION	DATE	DESCRIPTION
001	July 2022	Initial release.

Performance varies by use, configuration and other factors. Learn more on the Performance Index site.

No product or component can be absolutely secure.

Intel technologies may require enabled hardware, software or service activation.

Intel disclaims all express and implied warranties, including without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade.

0722/DN/WIPRO/PDF 737064-001US