



Integration of own distributed load-balancer at Kuryr for E/W traffic

BigDataStack Software Component developed by Red Hat

Description

Integration of own load-balancer through its octavia driver into Kuryr for E/W traffic.

Features

Own distributed load balancer integration at kuryr for E/W traffic. Kubernetes uses Kube-proxy to handle the services to pod traffic. This is handled through IPTables. However, Kuryr uses Octavia to create OpenStack load-balancers to back it up the services. With the default Octavia Amphora driver, there was a need for creating a load-balancer Virtual Machines (VM) (amphora) per each OpenShift/Kubernetes service. This has implications related to resource consumption (high number of VMs), control plane actions (slow, need time to create the VM), fault-tolerance (need to recreate the VM if it dies), and not performance efficient (traffic need to go through that VM which besides the extra hop in the network, could lead to network bottlenecks). By adding support for fully own base load balancers at Kuryr, we avoid all those problems as

- 1) There is no need to create VMs per service;
- 2) OAD balancing is fully based on ovs flows, which means it scales better than IPTables and it is distributed, thus avoiding the fault tolerance issue;
- 3) it is faster as it only needs to add a few flows, not to create a new VM;
- 4) thus also reducing the resource consumption;
- 5) and it performs better as it avoids both the extra hop in the network as well as the network bottleneck issues.



Integration of own distributed load-balancer at Kuryr for E/W traffic

Areas of Application

Deploying OpenShift on top of OpenStack with Kuryr CNI. All applications following Kubernetes model of exposing applications through services will benefit from this, by requiring fewer resources on the OpenStack side, as well as having a more scalable, resilience, and performance environment.

Market trends & opportunities

Make Kuryr project a more solid and appealing as a CNI for OpenShift/ Kubernetes deployments, with less resource consumption, better scalability and improved performance.

Customer benefits

When creating services, the number of resources needed for them is hugely decreased. Also, pod to service communication is improved (higher bandwidth, reduced latency).

Technological novelty

Add support for own octavia provider which is distributed by nature for E/W traffic when Kuryr CNI is used.

TRL level: 9

Find the Open Source code here:

This involves different code in different repos:

- <https://opendev.org/openstack/kuryr-kubernetes>
- <https://opendev.org/openstack/octavia>
- <https://opendev.org/openstack/neutron>
- <https://github.com/openshift/cluster-network-operator>

 bigdatastack.eu