DualStack Networking Istio on AWS

Josh Tischer / Aspen Mesh

ASPEN MESH

IstioCon

#IstioCon

Prerequisites

To create a successful dual-stack cluster you need to be aware of the Kubernetes version its capabilities and the physical and virtual networking layers.

- Kubernetes 1.20 started allowing the simultaneous assignment of both IPv4 and IPv6 addresses to Services.
- Underlay networking - refers to the physical network infrastructure
  - Bare Metal
  - Cloud Hosting (AWS, Azure, ...)
- Overlay networking – refers to the software driven virtualized networking, in this case Kubernetes network plugin or CNI
  - CNI that support IPv6 (Calico, OVNKubernetes, EKS**)
    EKS can support either IPv4 or IPv6 as of Jan 2022, not both
- Application modifications to respond on IPv6 requests

Istio Dual Stack

Aspen Mesh has been working on dual-stack features for Istio for the past year and is working with the community to open source this work.

For the majority of our development and testing effort we chose OpenShift as our platform.

We tested (and learned a lot about) the following

- Kubernetes Clusters
  - Openshift 4.8+
  - Kind
  - KubeAdm
- Environments
  - BareMetal
  - AWS
  - Azure

OpenShift Setup on AWS

There are several ways to setup OpenShift. We are focused on the Installer Provisioned Infrastructure deployed from their installer and client tools.

- Our scripts work with awscli and the OpenShift installer to modify your AWS environment and OpenShift cluster

All scripts mentioned can be found in our github.com/aspenmesh/open-source repo.

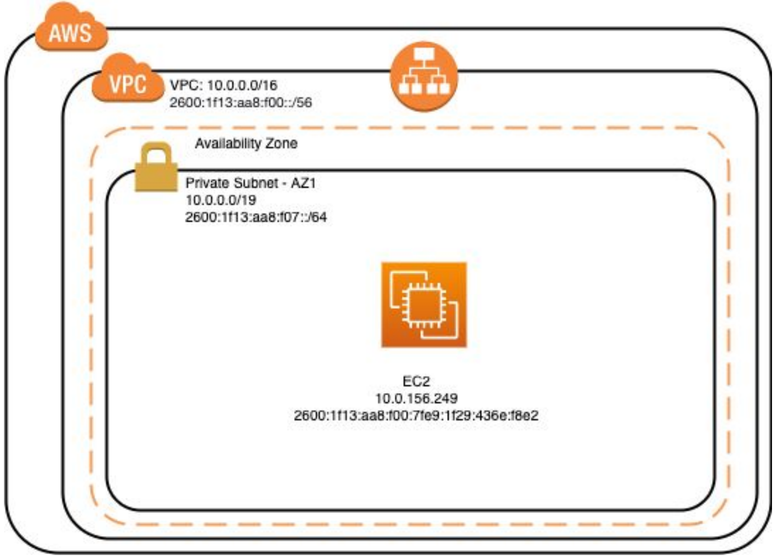
How to setup a Dual Stack OpenShift cluster in AWS

- Clone our repo: git clone git@github.com:aspenmesh/open-source
- Change your directory: cd open-source/aws-dual-stack
- Download the OpenShift Installer, Client and Pull-Secret to the _install_ directory
- ./openshift-cluster.sh <cluster-name>
  - This script creates the IPv4 AWS infrastructure and Kubernetes cluster on AWS
- ./openshift-upgrade-aws.sh <cluster-name>
  - The current installer creates an ipv4 cluster and AWS infrastructure. This scripts upgrades the underlay network in AWS for Dual Stack networking.
    ** see AWS guide
- ./openshift-upgrade-cluster.sh <cluster-name>
  - Converts the Kubernetes cluster ipv4 network add IPv6
    ** see OpenShift docs
- ./openshift-delete-cluster.sh
  - Uses metadata.json file in the _install_ directory for cluster information

Underlay (Physical) Networking on AWS

How to add IPv6 to an IPv4 network in AWS

- Associate VPC and subnets with IPv6 CIDR blocks
- Update route tables
- Update Security group rules
- Upgrade Load Balancer
- Update EC2 Instances
  - Change instance types if needed
  - Assign IPv6 Address



These steps are scripted for OpenShift but could easily be tweaked to run on any VPC say for a cluster setup with KubeAdm

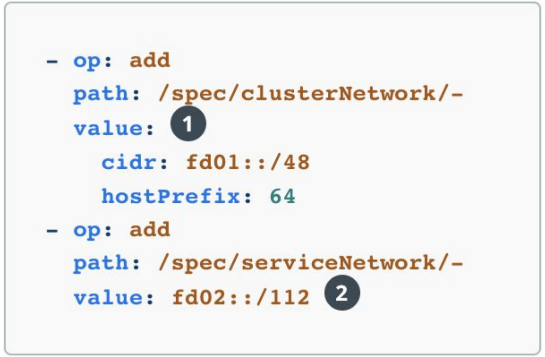
```
./openshift-upgrade-aws.sh test-openshift
```

AWS Dual-Stack Guide

Overlay - Kubernetes Networking

OpenShift

- Patch network.config.openshift.io to add IPv6 CIDR
- Wait for network changes to roll out `oc wait --for=condition=progressing=false clusteroperators/network`



```
- op: add
  path: /spec/clusterNetwork/-
  value:
    cidr: fd01::/48
    hostPrefix: 64
- op: add
  path: /spec/serviceNetwork/-
  value: fd02::/112
```

KubeAdm:

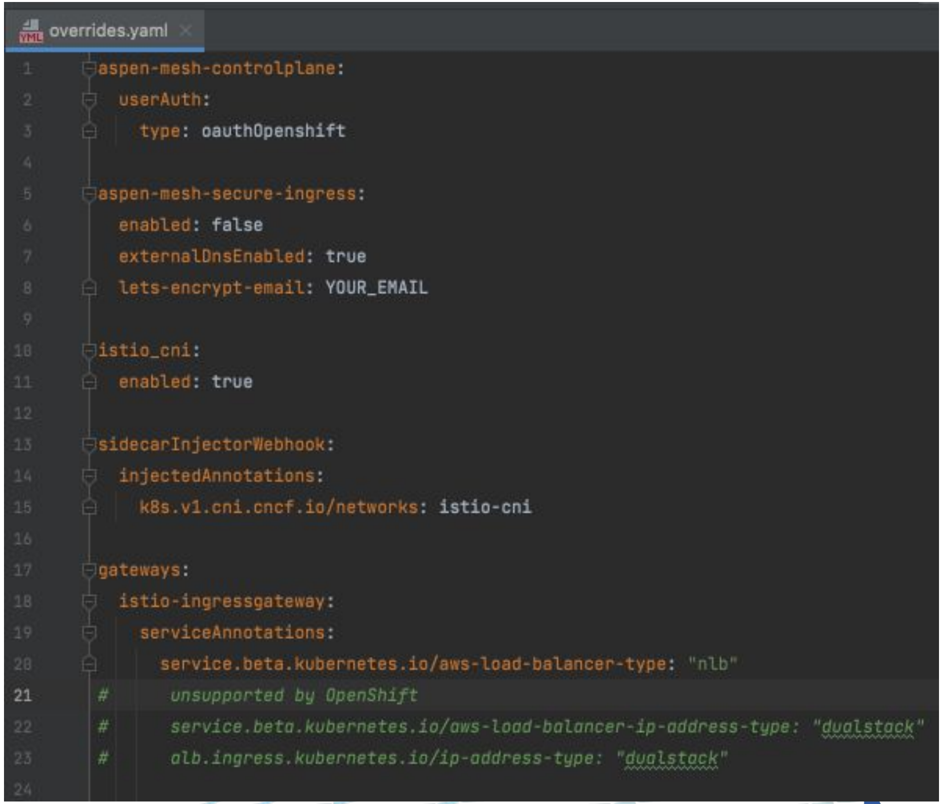
To configure IPv4/IPv6 dual-stack, set dual-stack cluster network assignments.

- kube-apiserver:
  --service-cluster-ip-range=<IPv4 CIDR>,<IPv6 CIDR>
- kube-controller-manager:
  --cluster-cidr=<IPv4 CIDR>,<IPv6 CIDR>
  --service-cluster-ip-range=<IPv4 CIDR>,<IPv6 CIDR>
  --node-cidr-mask-size-ipv4|--node-cidr-mask-size-ipv6 defaults to /24 for IPv4 and /64 for IPv6
- kube-proxy:
  --cluster-cidr=<IPv4 CIDR>,<IPv6 CIDR>
- kubelet:
  Manually assign .status.addresses for a node

#IstioCon

Install Aspen Mesh

- Sign up for an account https://aspenmesh.io/invite/
- Visit https://my.aspenmesh.io/
- Follow the documentation
- Install 1.11.8-am2
  (Istio + Dual Stack features)
  ** sample overrides for OpenShift

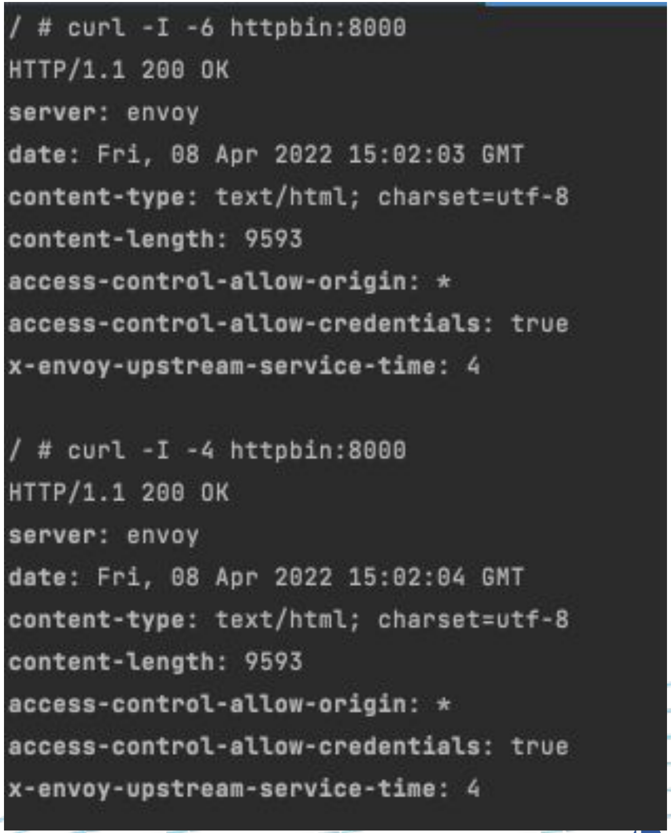


```yaml
aspen-mesh-controlplane:
  userAuth:
    type: oauthOpenshift

aspen-mesh-secure-ingress:
  enabled: false
  externalDnsEnabled: true
  lets-encrypt-email: YOUR_EMAIL

istio_cni:
  enabled: true

sidecarInjectorWebhook:
  injectedAnnotations:
    k8s.v1.cni.cncf.io/networks: istio-cni

gateways:
  istio-ingressgateway:
    serviceAnnotations:
      service.beta.kubernetes.io/aws-load-balancer-type: "nlb"
#     unsupported by OpenShift
#     service.beta.kubernetes.io/aws-load-balancer-ip-address-type: "dualstack"
#     alb.ingress.kubernetes.io/ip-address-type: "dualstack"

```

Validate Pod Networking

`./test-dual-stack.sh`
Installs a sleep pod and httpbin to dual-stack namespace and configures an istio gateway and virtual service

```
sleeppod=$(oc get pods --no-headers -o
custom-columns=":metadata.name"
--selector=app=sleep )
oc exec -it $sleeppod sh

curl -I -6 httpbin:8000
curl -I -4 httpbin:8000
```



```
/ # curl -I -6 httpbin:8000
HTTP/1.1 200 OK
server: envoy
date: Fri, 08 Apr 2022 15:02:03 GMT
content-type: text/html; charset=utf-8
content-length: 9593
access-control-allow-origin: *
access-control-allow-credentials: true
x-envoy-upstream-service-time: 4

/ # curl -I -4 httpbin:8000
HTTP/1.1 200 OK
server: envoy
date: Fri, 08 Apr 2022 15:02:04 GMT
content-type: text/html; charset=utf-8
content-length: 9593
access-control-allow-origin: *
access-control-allow-credentials: true
x-envoy-upstream-service-time: 4
```

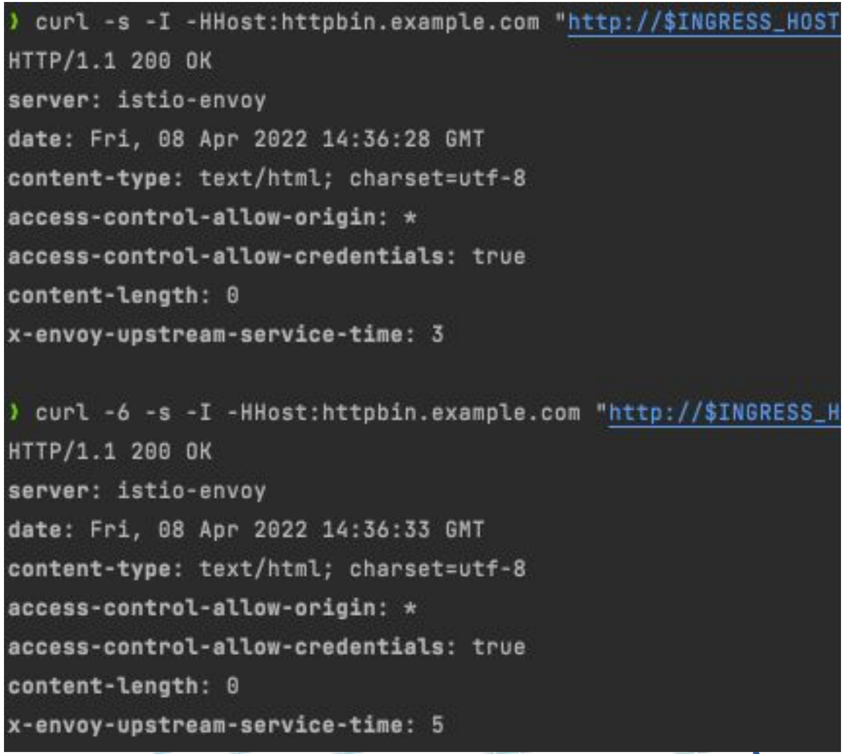
Validate Public Access

export INGRESS_PORT=$(kubectl -n istio-system get service istio-ingressgateway -o jsonpath='{.spec.ports[?(@.name=="http2")].port}’)

export INGRESS_HOST=$(kubectl -n istio-system get service istio-ingressgateway -o jsonpath='{.status.loadBalancer.ingress[0].hostname}’)

echo $INGRESS_HOST:$INGRESS_PORT
a92bf429b…..elb.us-west-2.amazonaws.com:80

Also check nslookup -–type=aaaa for ipv6 dns

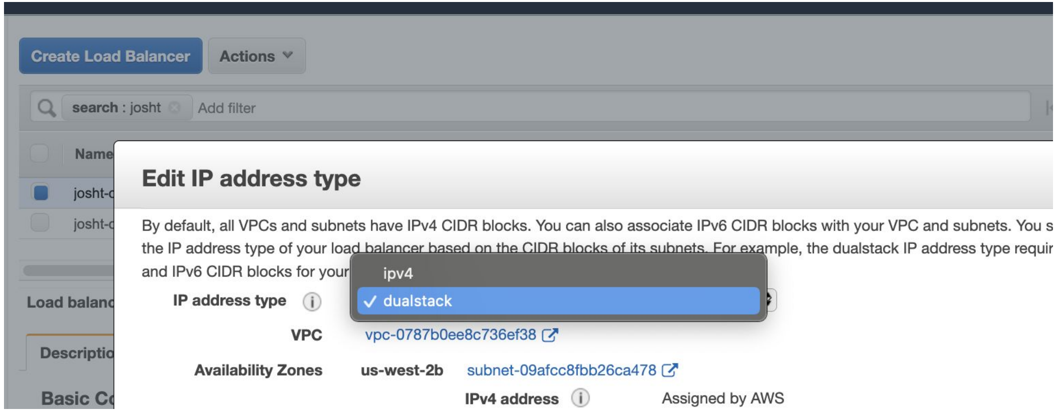


```
❯ curl -s -I -HHost:httpbin.example.com "http://$INGRESS_HOST
HTTP/1.1 200 OK
server: istio-envoy
date: Fri, 08 Apr 2022 14:36:28 GMT
content-type: text/html; charset=utf-8
access-control-allow-origin: *
access-control-allow-credentials: true
content-length: 0
x-envoy-upstream-service-time: 3

❯ curl -6 -s -I -HHost:httpbin.example.com "http://$INGRESS_H
HTTP/1.1 200 OK
server: istio-envoy
date: Fri, 08 Apr 2022 14:36:33 GMT
content-type: text/html; charset=utf-8
access-control-allow-origin: *
access-control-allow-credentials: true
content-length: 0
x-envoy-upstream-service-time: 5
```

Istio Ingress-Gateway on OpenShift issues

OpenShift creates a Classic LB by default, which does not support dualstack ☹️

Istio overrides can change this to an NLB, but the dual stack mode is not yet supported and has to be manually upgraded.



Edit IP address type

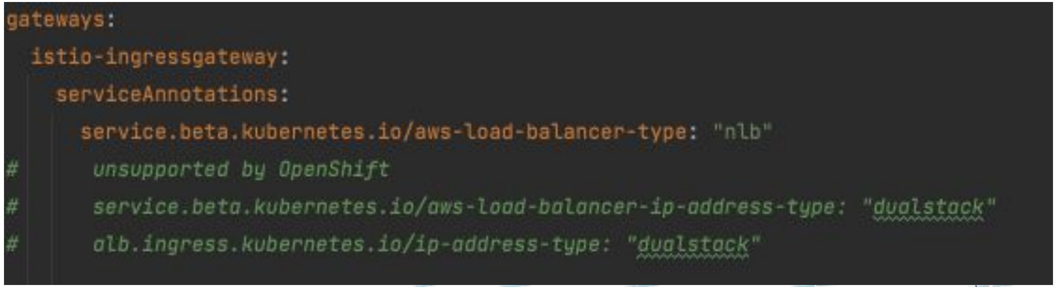
By default, all VPCs and subnets have IPv4 CIDR blocks. You can also associate IPv6 CIDR blocks with your VPC and subnets. You s
the IP address type of your load balancer based on the CIDR blocks of its subnets. For example, the dualstack IP address type requir
and IPv6 CIDR blocks for your

IP address type: ipv4 / ✓ dualstack

VPC: vpc-0787b0ee8c736ef38

Availability Zones: us-west-2b subnet-09afcc8fbb26ca478

IPv4 address: Assigned by AWS



```
gateways:
  istio-ingressgateway:
    serviceAnnotations:
      service.beta.kubernetes.io/aws-load-balancer-type: "nlb"
#       unsupported by OpenShift
#       service.beta.kubernetes.io/aws-load-balancer-ip-address-type: "dualstack"
#       alb.ingress.kubernetes.io/ip-address-type: "dualstack"
```

Open Source Istio

- Aspen Mesh and Intel are working on implementing dual-stack capability in Istio
    - We proposed an RFC based on our design, however, we are currently evaluating our next steps based on feedback
        - We are intending to be making changes in Envoy, but we are still in the (re-)design phase
    - We are committed to getting dual-stack implemented as it's important to us and our customers

Thank you!

@Josh Tischer (Aspen Mesh)
https://aspenmesh.io