



# AKO in GCP

Avi Technical Reference (v20.1)

Copyright © 2021

# AKO in GCP

[view online](#)

## Overview

This article explains the pod routing support in GCP for AKO.

The network configuration modes supported by AKO in GCP Full Access IaaS cloud are:

- [Inband Management](#)
- [Two-arm mode with the Same Backend Subnet](#)

In both the deployments there is a one-to-one mapping between virtual private cloud (VPC) and a Kubernetes cluster. So, in case of inband management, each cluster should have a dedicated cloud created in Avi.

In the two-arm-mode, multiple clusters can sync to the same cloud.

Refer to the [GCP Cloud Network Configuration](#) article, for more information on the network configuration modes supported for GCP IaaS cloud.

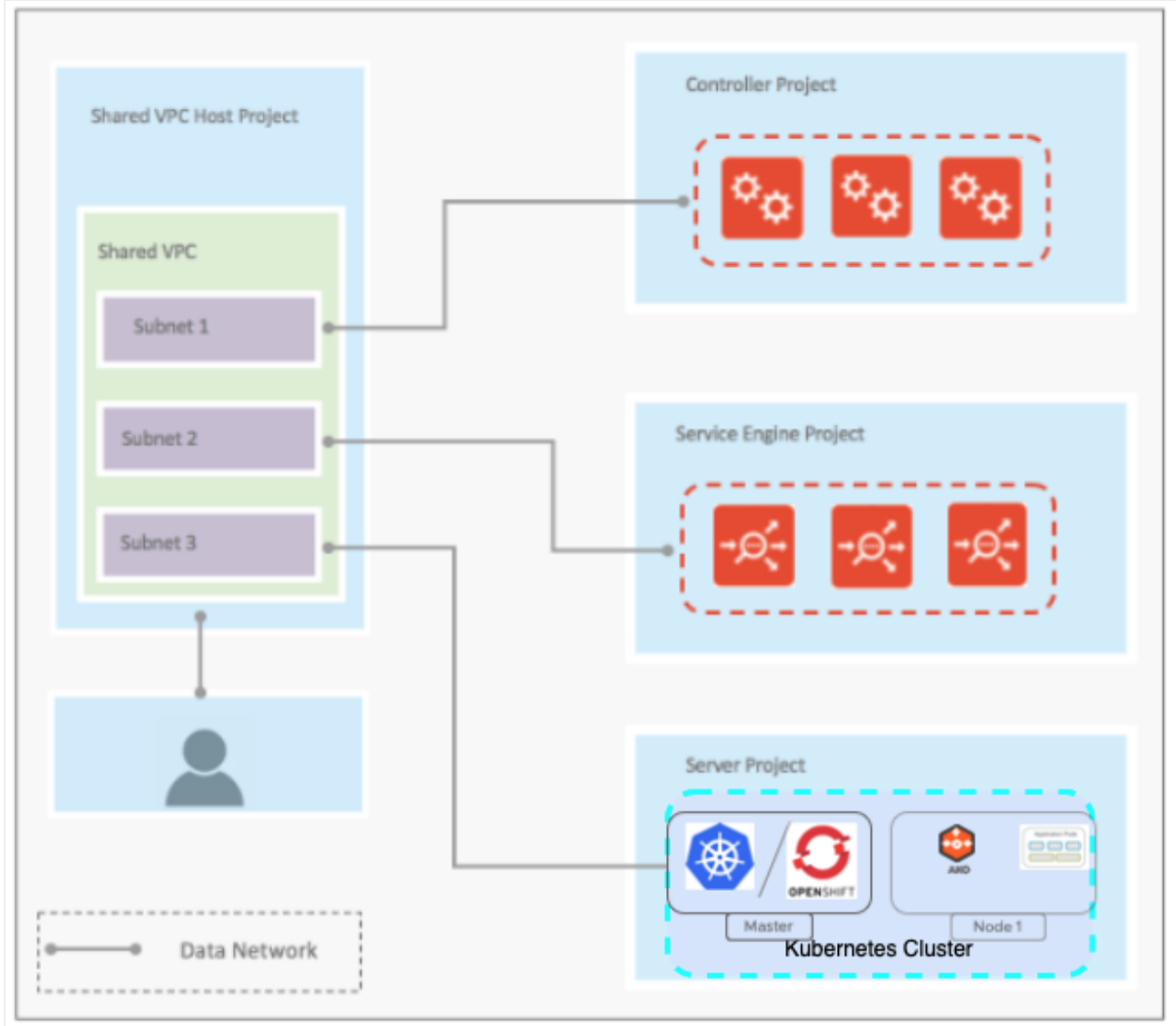
## Types of Networks

The types of networks used are as follows:

- **Frontend Data Network:** This network connects the VIP to the Service Engines. All the VIP traffic reaches Service Engines through this network.
- **Backend Data Network:** This network connects the Service Engines to the application servers. All the traffic between the Service Engines and application servers flows through this network.
- **Management Network:** This network connects the Service Engines with the Controller for all management operations.

### ### Inband Management

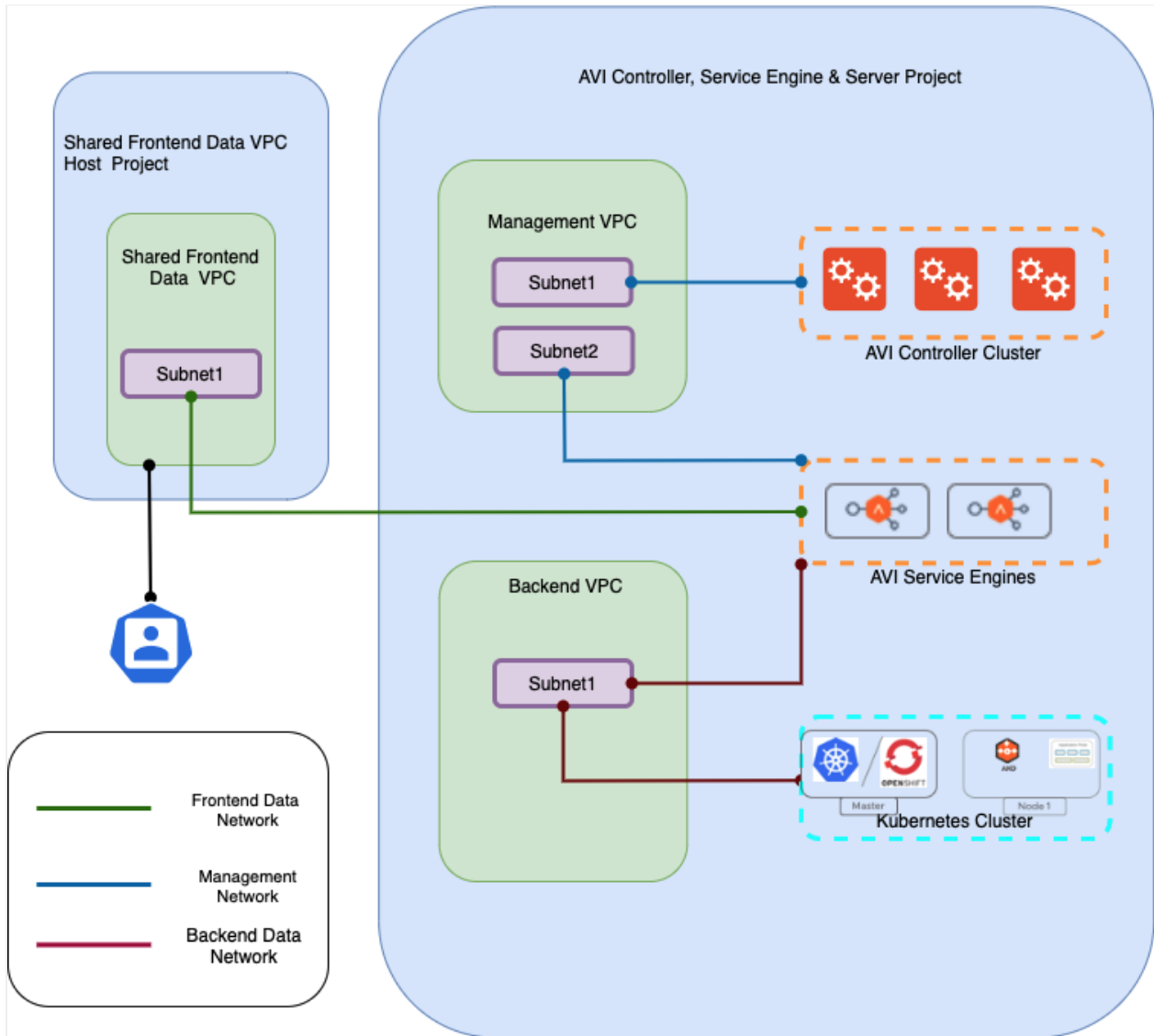
The Service Engines will be connected to only one VPC subnet.



There is no network isolation between frontend and backend data and management traffic as both will go through same VPC subnet.

### Two-Arm Mode with the Same Backend Subnet

In the two-arm mode, the Service Engines are connected to three VPC subnets, one each for the frontend data traffic, management traffic and backend data traffic.

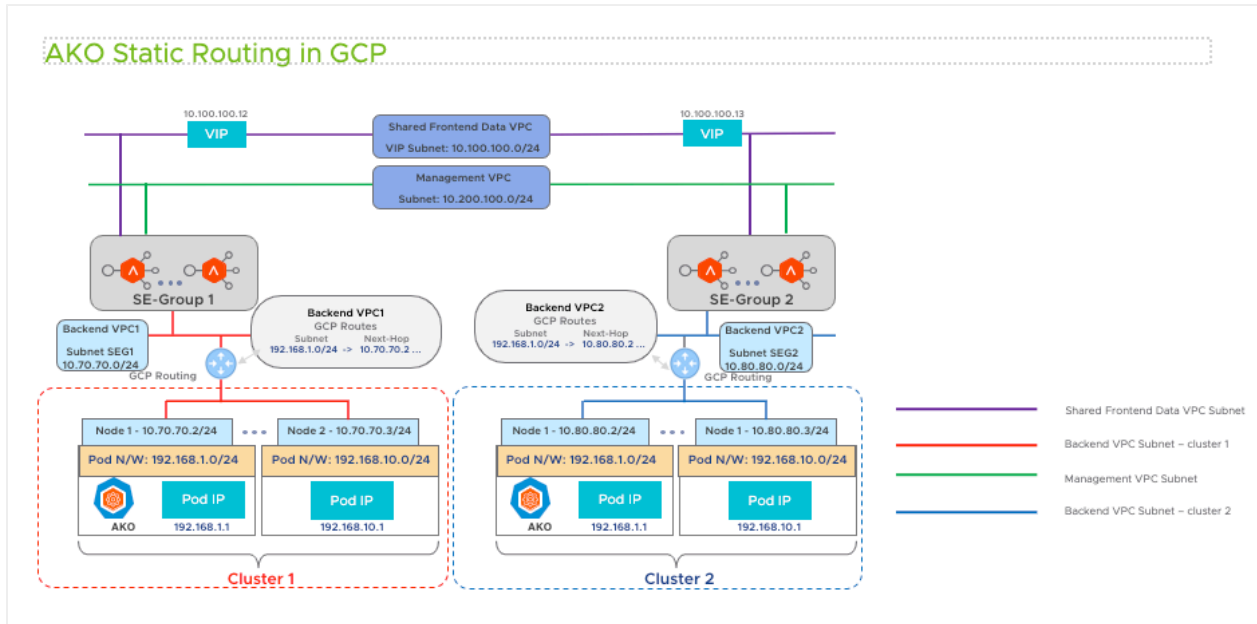


- The two-arm mode provides isolation between the management, the frontend data and the backend data networks.
- The first interface of the SE will be connected to the frontend data network. GCP supports shared VPC only on first NIC, therefore the data VPC can be a shared VPC.
- The second interface of the SE will be connected to the management network. This interface cannot be connected to a shared VPC because GCP allows shared VPC only on first NIC.
- The third interface of the SE will be connected to the Kubernetes cluster where AKO is running.
- In this mode, the backend network of the SE is the same as the one where the Kubernetes or OpenShift nodes are connected.

**Routing in GCP Two-Arm Mode with the same Backend Subnet**

When multiple clusters are syncing to the same cloud the pod CIDRs can overlap. Currently, in AKO for each cluster, SE Group is created.

In addition to Service Engine Group each cluster should be in unique VPC in GCP cloud. This config is supplied in the Service Engine Group during the cloud setup by the admin.



AKO configures the routes for `cluster1` and `cluster2` in the GCP routing tables for `vpc1` and `vpc2` respectively. Since there is unique mapping for cluster and SE Group the overlapping pod CIDR's routes will be in their respective VPC's.

## AKO in Azure (Day 0 Preparation)

The Day 0 preparation checklist required to set up AKO in GCP is as listed below:

### Operations on GCP Side

- Ensure that the Kubernetes / OpenShift clusters running in their own dedicated VPC.
- Ensure that the Kubernetes and OpenShift cluster node virtual machines should have IP forwarding enabled in the GCP's virtual machines.
- Create a dedicated backend VPC for each of the clusters.

### Operations on Avi Side

- Create a GCP cloud in Avi.  
Note: Skip this step if the IaaS cloud is already created.
- Create a Service Engine group for each cluster.
- Override the backend VPC subnet in each of the Service Engine group.
  - If there are two clusters, `cluster1` and `cluster2`,
    - Run `configure serviceenginegroup cluster1-seg`
    - Override the `gcp_config` in the SE Group

```
configure serviceenginegroup Default-Group
```

```
|-----|-----| |gcp_config| | |
backend_data_vpc_network_name dev-multivnet-1|dev-multivnet-1|<br> |
```

```
backend_data_vpc_subnet_name |mv-subnet-1 |<br>  
|-----|-----|
```

### Document Revision History

| Date                 | Change Summary                                      |
|----------------------|---|
| December 18,<br>2020 | Published the pod routing support in GCP for<br>AKO |