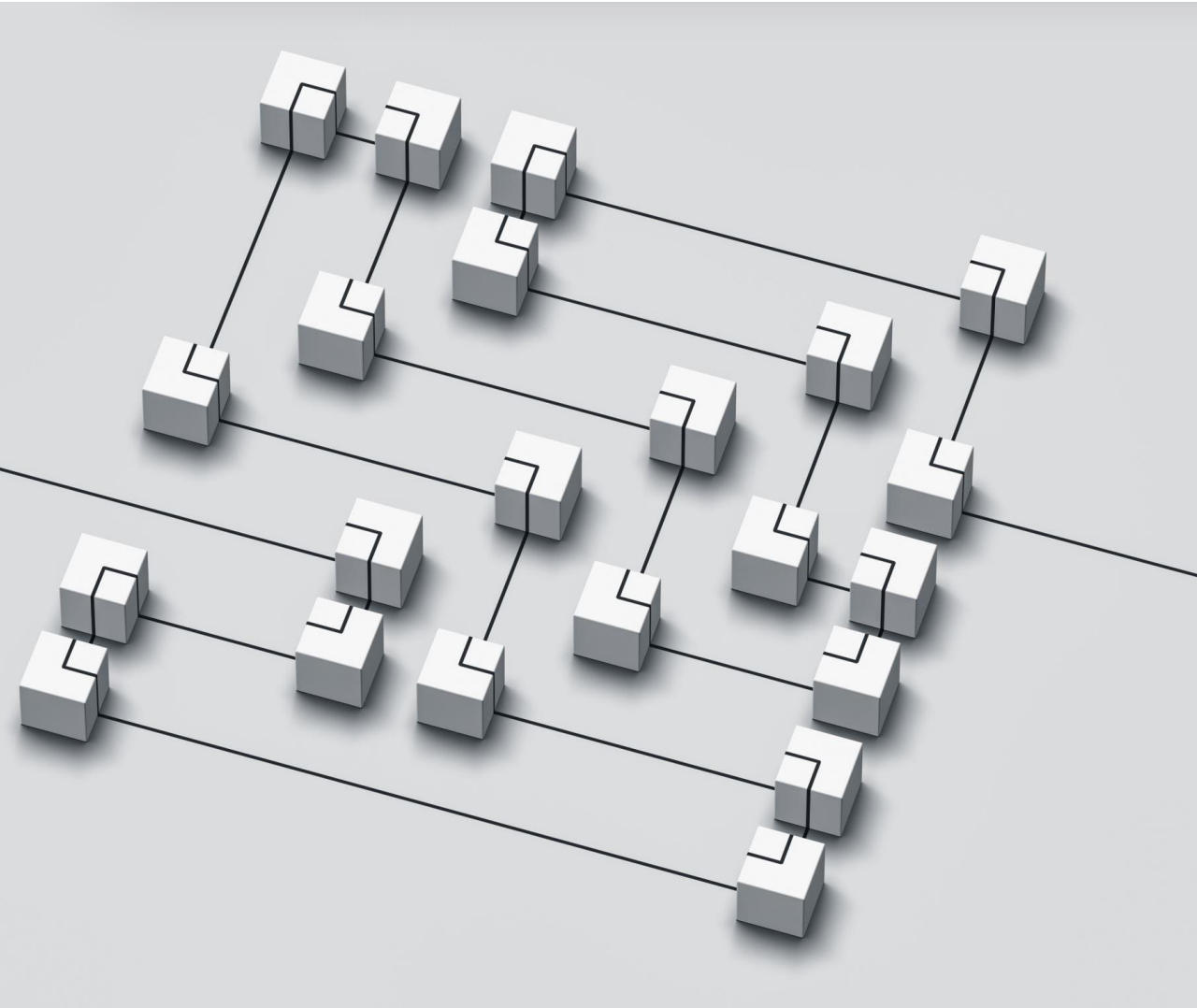


Multi Cluster Network Connectivity

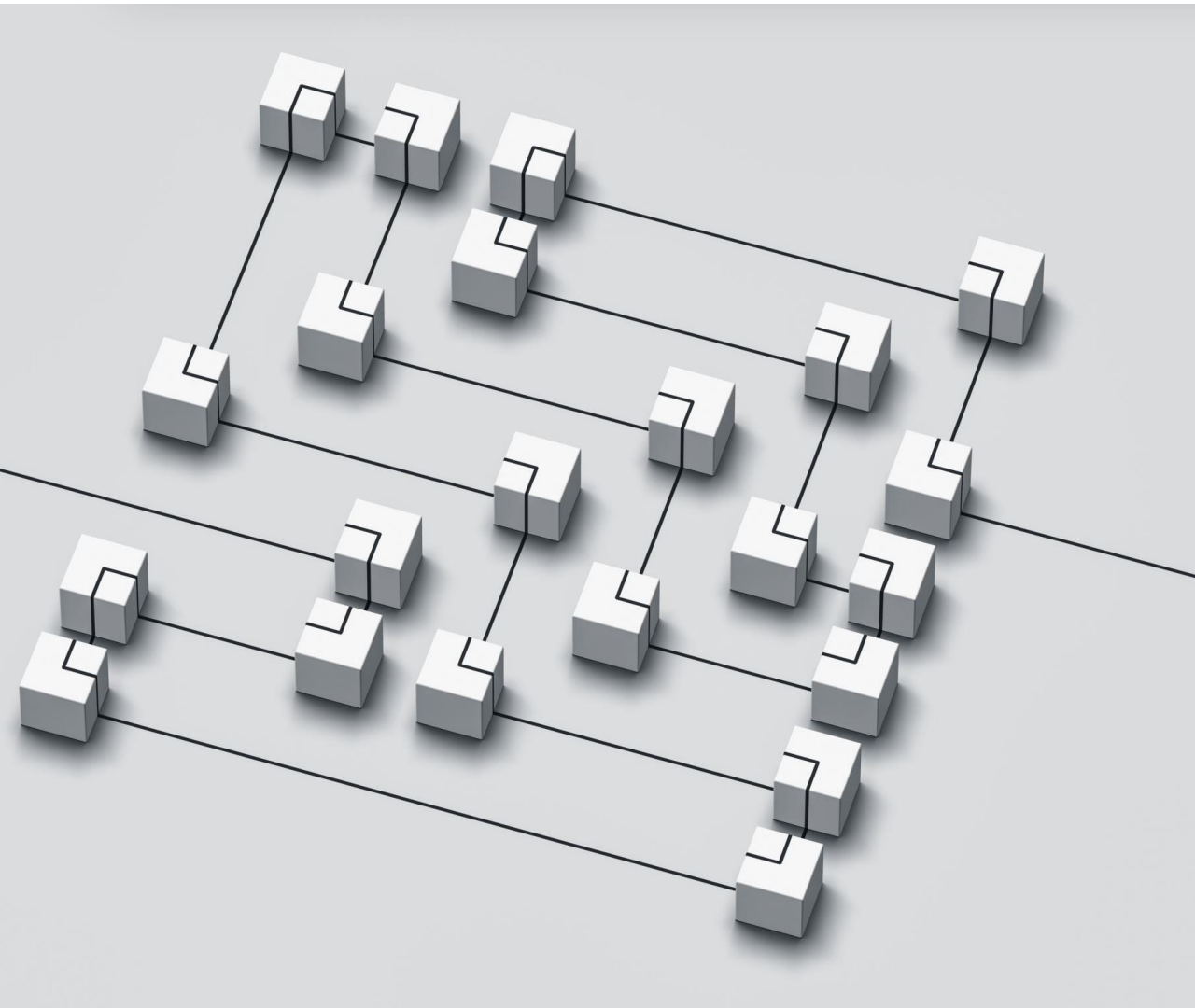
Calsoft Pvt Ltd | Vinod Borole, Senior Software Architect

Background



- Enterprises today often deploy applications across multiple Kubernetes clusters, which could be hosted on various cloud providers or on-premises environments. Managing and connecting these dispersed clusters securely and efficiently is crucial for maintaining a reliable and scalable application infrastructure.

Problem Statement

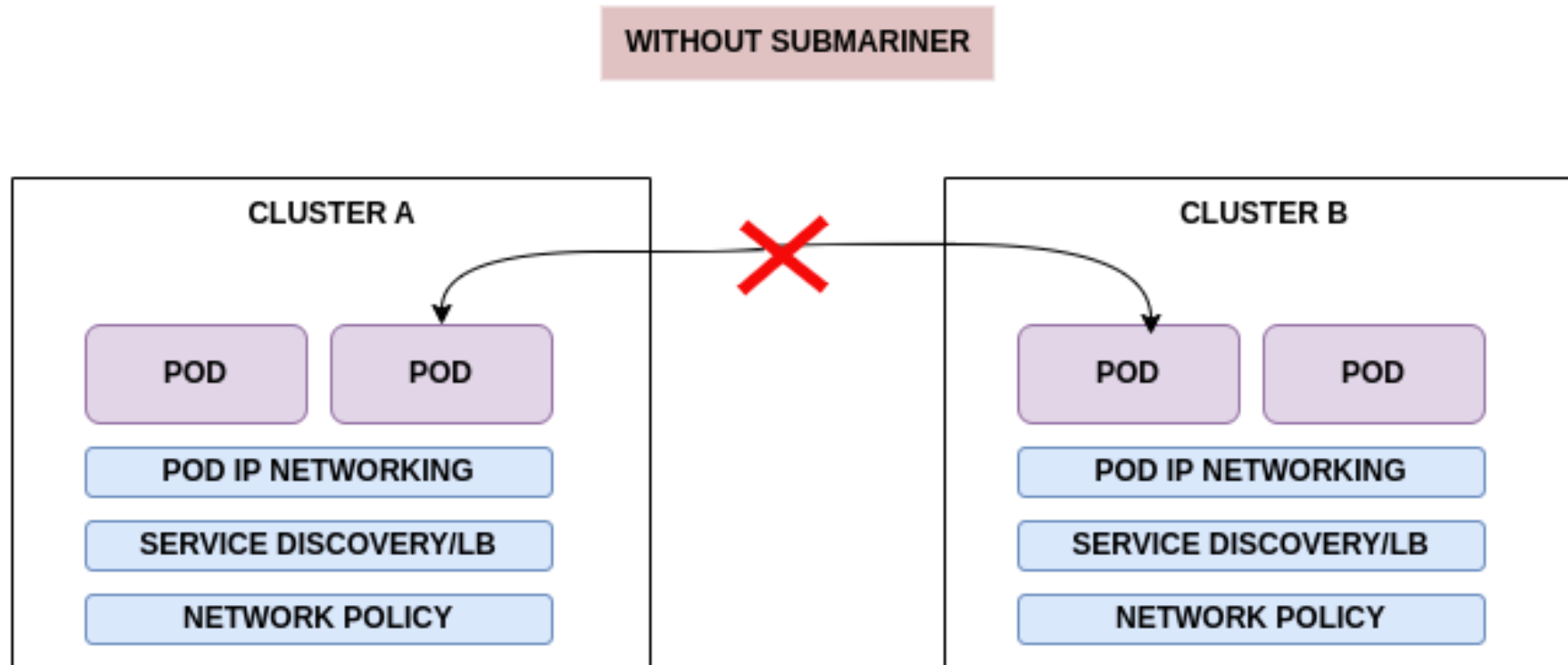


- Design and implement a solution for connecting and managing multiple Kubernetes clusters located in different environments, including public clouds (e.g., AWS, GCP, Azure) and on-premises data centers.
- The solution should provide seamless communication between clusters while ensuring security, scalability, and minimal latency.

Key Requirements

Requirements	Description
Inter-Cluster Communication	Enable communication between pods and services deployed across different Kubernetes clusters
Network Security	Implement secure networking practices, including encryption, authentication, and authorization, to protect data in transit between clusters.
Scalability	Ensure the solution can scale to support a growing number of clusters and workloads efficiently.
Resource Discovery	Provide mechanisms for discovering and accessing resources (e.g., services, pods) across clusters without manual configuration
Centralized Management	Offer a centralized management interface for monitoring, configuring, and troubleshooting the connected clusters
Cross-Cloud Compatibility	Ensure the solution is compatible with major cloud providers and can connect clusters regardless of their hosting environment.

Multi Cluster Architecture



Submariner



Open Source License: This software is distributed under the Apache 2.0 License.



Multi-Cluster Kubernetes Networking Tool: This tool enables communication between pods across multiple Kubernetes clusters, irrespective of where they are present geographically.



Direct Pod Communication: Pods can directly communicate with each other across Kubernetes clusters.

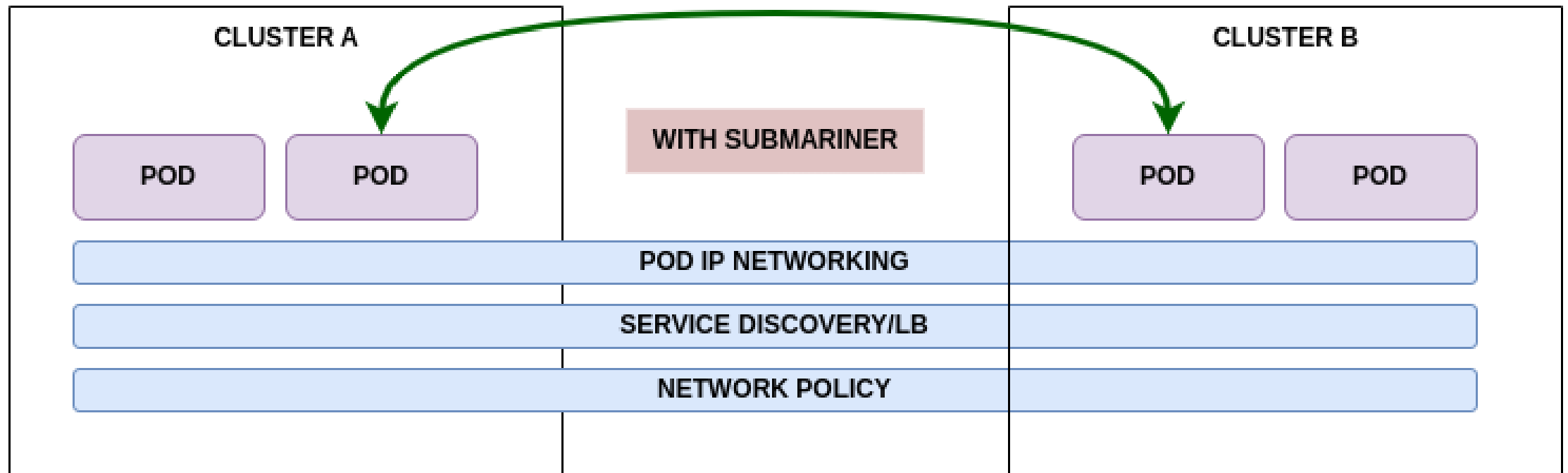


Security Features: It ensures security by establishing an IPsec tunnel between clusters.



CNI Agnostic: This tool operates at a layer that is independent of your network provider, making it compatible with various Container Network Interfaces (CNIs).

Multi Cluster Architecture with Submariner



Submariner Deployment



Gateway Engine: manages the secure tunnels to other clusters.



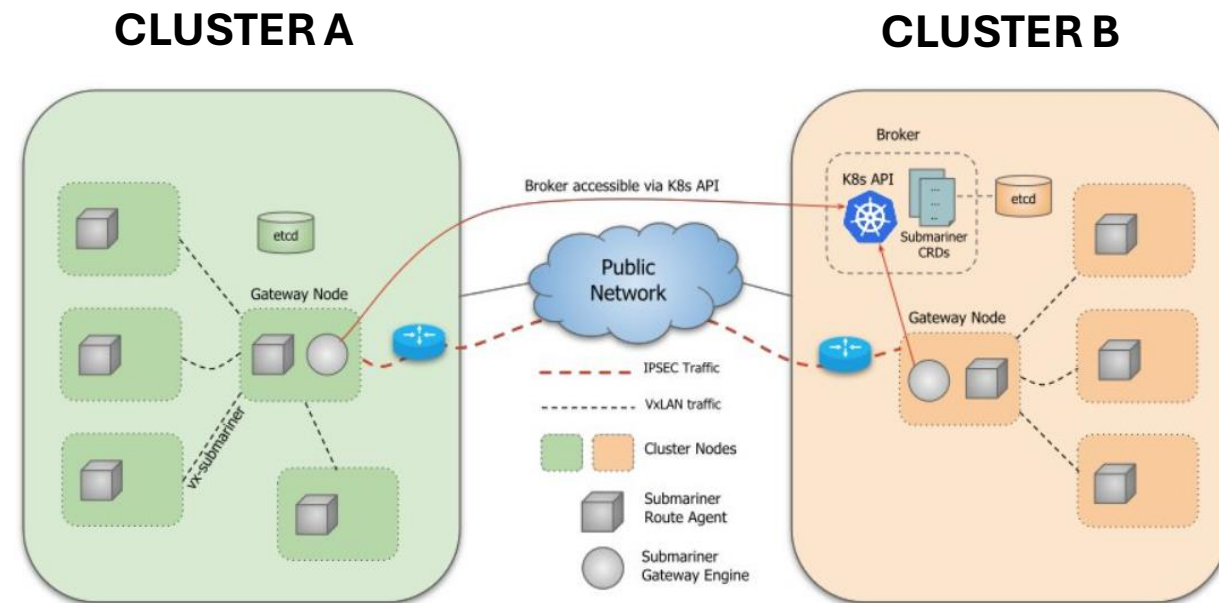
Route Agent: routes cross-cluster traffic from nodes to the active Gateway Engine.



Broker: facilitates the exchange of metadata between Gateway Engines enabling them to discover one another.



Service Discovery: provides DNS discovery of Services across clusters.



Submariner Features

Requirements	Submariner
Inter-Cluster Communication	Supports with the help of gateway nodes and route agents
Network Security	IPSec implementation using Libreswan or Wireguard supported
Scalability	Supports as gateway nodes can be deployed in HA mode
Resource Discovery	Using Subctl CLI it can detect nodes and deploy the route agents
Centralized Management	Submariner can be deployed as an independent cluster from where it can manage multiple
Cross-Cloud Compatibility	Supports all cloud providers along with Google GKE and Rancher

References

- Submariner <https://submariner.io/>
- Demo Source code Github: <https://github.com/Calsoft-Pvt-Ltd/submariner-demo>

Thank you

Vinod Borole | Senior Software Architect
Calsoft Pvt Ltd.