ADR-004: Calico CNI Networking

Submitters

- Luke Doyle (D00255656)
- Hafsa Moin (D00256764)

Change Log

- approved 2026-02-11

Referenced Use Case(s)

- UVote Network Security Requirements - NetworkPolicy enforcement for zero-trust service isolation within the UVote Kubernetes cluster.

Context

The UVote Kubernetes cluster requires a Container Network Interface (CNI) plugin that supports the full Kubernetes NetworkPolicy API, including both ingress and egress rules with label-based pod selection and port-level filtering. The zero-trust security model (ADR-010) depends entirely on the CNI's ability to enforce network policies. Without a CNI that supports enforcement, NetworkPolicy resources are accepted by the Kubernetes API server but have no effect, meaning all pod-to-pod traffic remains permitted regardless of what policies are defined.

Kubernetes does not enforce NetworkPolicies itself. The NetworkPolicy API defines the desired state, but enforcement is delegated to the CNI plugin. This makes CNI selection a security-critical decision rather than a purely operational one.

Calico was selected on the basis of research into CNI options for Kind-based clusters and a recommendation from the DkIT DevOps module lecturer, who identified Calico as a well-established choice for NetworkPolicy enforcement in development and production Kubernetes environments.

Proposed Design

Services and modules impacted:

Calico is installed cluster-wide and handles all pod networking and NetworkPolicy enforcement. It has no direct integration with individual UVote services but underpins the entire zero-trust security model defined in ADR-010.

Installation sequence:

```
# 1. Install Calico operator
kubectl create -f \

https://raw.githubusercontent.com/projectcalico/calico/v3.26.1/manifes
ts/tigera-operator.yaml

# 2. Install Calico custom resources
kubectl create -f \

https://raw.githubusercontent.com/projectcalico/calico/v3.26.1/manifes
ts/custom-resources.yaml

# 3. Wait for Calico to be ready
kubectl wait --for=condition=Ready pods --all -n calico-system --
timeout=300s

```

Installation is automated via the `setup_k8s_platform.py` script.

Configuration:

- Data plane: iptables (default for Kind)
- IPAM: Calico IPAM with `192.168.0.0/16` CIDR

- Encapsulation: VXLAN (required for Kind, as BGP is not available between Docker containers)
- NetworkPolicy API: Standard `networking.k8s.io/v1` only (no Calico-specific CRDs used)

DevOps impact:

The Kind cluster configuration requires `disableDefaultCNI: true` to allow Calico to take over pod networking. This is already set in `kind-config.yaml` (ADR-003). All UVote NetworkPolicy resources use the standard `networking.k8s.io/v1` API, ensuring portability to any compliant CNI.

Considerations

Cilium (not selected): Cilium is a modern CNI using eBPF for networking, security, and observability. It provides full NetworkPolicy support and adds L7-aware policies that can filter by HTTP path and headers. It was not selected for three reasons. First, it requires a kernel version of 5.4 or higher for its full eBPF feature set, which adds an environmental dependency. Second, its resource overhead is higher than Calico at approximately 400MB for system pods. Third, its Kind compatibility requires additional configuration steps that are less well-documented than Calico's. The L7 filtering capabilities are also not needed for this project, as all UVote network policies operate at L3/L4 only.

Flannel (not selected): Flannel is a simple overlay network that provides basic pod-to-pod connectivity. It was not selected because it has no NetworkPolicy support. Policies are accepted by the API server but silently ignored, meaning a default-deny policy has no effect. This is a hard disqualifier given the project's zero-trust security requirement.

Weave Net (not selected): Weave Net provides mesh networking with optional built-in traffic encryption and basic NetworkPolicy support. It was eliminated because Weaveworks, the company behind it, ceased operations in February 2024. With no active maintainer, it is not a suitable dependency for a project continuing into 2026.

Standard API usage: All UVote network policies use the standard `networking.k8s.io/v1` API rather than Calico-specific CRDs. This is an intentional design choice that keeps policies portable to any compliant CNI, including Cilium if a migration is needed in Stage 2.

Installation complexity: Calico requires disabling the default CNI and installing the operator and custom resources separately. This is a one-time setup cost and is automated in the deployment script.

Decision

Calico v3.26.1 was selected as the CNI plugin for the UVote Kubernetes cluster.

The primary driver is that Calico provides full enforcement of the Kubernetes NetworkPolicy API, including both ingress and egress rules, label-based pod selection, and port-level filtering. This is a prerequisite for the zero-trust security model in ADR-010. Of the options considered, Flannel provides no enforcement and Weave Net is unmaintained, leaving Calico and Cilium as viable candidates.

Calico was preferred over Cilium on the basis of its maturity, its well-documented compatibility with Kind clusters, and the recommendation of the DkIT DevOps module lecturer, who identified Calico as a reliable and widely-used option for this type of environment. Calico is also used by major cloud providers including Azure AKS and Amazon EKS, meaning the configuration knowledge transfers directly to production deployments.

The `calicoctl` CLI provides policy inspection, endpoint status, and IP route information, which supports troubleshooting of the 12 NetworkPolicy resources defined in ADR-010.

The performance difference between Calico's iptables-based enforcement and Cilium's eBPF is not meaningful at the scale of this project. The trade-off of slightly lower theoretical performance for simpler setup and better-

documented Kind compatibility is accepted.

A review is scheduled for the end of Stage 2 (April 2026) to evaluate whether to migrate to Cilium for eBPF performance benefits in a production environment.

Other Related ADRs

- ADR-003: Kubernetes Platform - cluster on which Calico is installed
- ADR-010: Zero-Trust Network Security - NetworkPolicy resources enforced by Calico
- ADR-012: Kind Distribution - cluster configured with `disableDefaultCNI: true`

References

- Calico Documentation
- Calico Kind Quickstart
- Kubernetes NetworkPolicy Documentation
- EdgeX Foundry ADR Template