**🔐 What are Network Policies in Kubernetes?**

Network Policies are rules in Kubernetes that **control who can talk to whom** over the network. By default, **all pods can talk to each other**. But once you define a NetworkPolicy, it **blocks everything not explicitly allowed**—kind of like a firewall for your pods.

**🚦Ingress vs. Egress in Network Policies**

* **Ingress** = Incoming traffic **to a pod** (who can connect to it).
* **Egress** = Outgoing traffic **from a pod** (who it can connect to).

In a typical app:

* The **frontend** talks to the **backend**.
* The **backend** talks to the **database**.
* We might want to **block the frontend from talking directly to the database**.

**🛡️ Scenario: Frontend, Backend, Database**

Let’s say you have 3 components:

* **frontend** (e.g. UI pod)
* **backend** (business logic pod)
* **db** (database pod)

The desired access:

* Frontend ⟶ Backend ✅
* Backend ⟶ Database ✅
* Frontend ⟶ Database ❌ (should be blocked)

**🎯 How Do We Implement This?**

We use **labels** and **selectors** in Network Policies.

Example:

1. Label each pod with role=frontend, role=backend, role=db.
2. Then create a NetworkPolicy for the **database** that **only allows ingress from pods with role=backend**.

apiVersion: networking.k8s.io/v1

kind: NetworkPolicy

metadata:

name: allow-backend-to-db

spec:

podSelector:

matchLabels:

role: db

ingress:

- from:

- podSelector:

matchLabels:

role: backend

This blocks frontend access to the database because it doesn't match the rule.

**🧩 What CNI Plugins Support Network Policies?**

Kubernetes itself doesn’t enforce Network Policies. That’s the job of the **CNI (Container Network Interface)** plugin.

* ✅ **Supports NetworkPolicy**:
  + **Calico** (most powerful and flexible)
  + **Cilium**
  + **Weave**
  + **Kube-router**
* ❌ **Does NOT Support NetworkPolicy**:
  + **Flannel** (by default, unless combined with another plugin like Calico for policy support)

**✅ Final Summary**

* Network Policies let you control access between pods.
* You can block frontend from accessing the database using podSelectors.
* You must use a CNI plugin like Calico that supports network policies.
* Always test policies carefully, as once they’re in place, all traffic is denied **unless explicitly allowed**.