Introduction to Cilium

The De-Facto Networking and Security Platform for Kubernetes

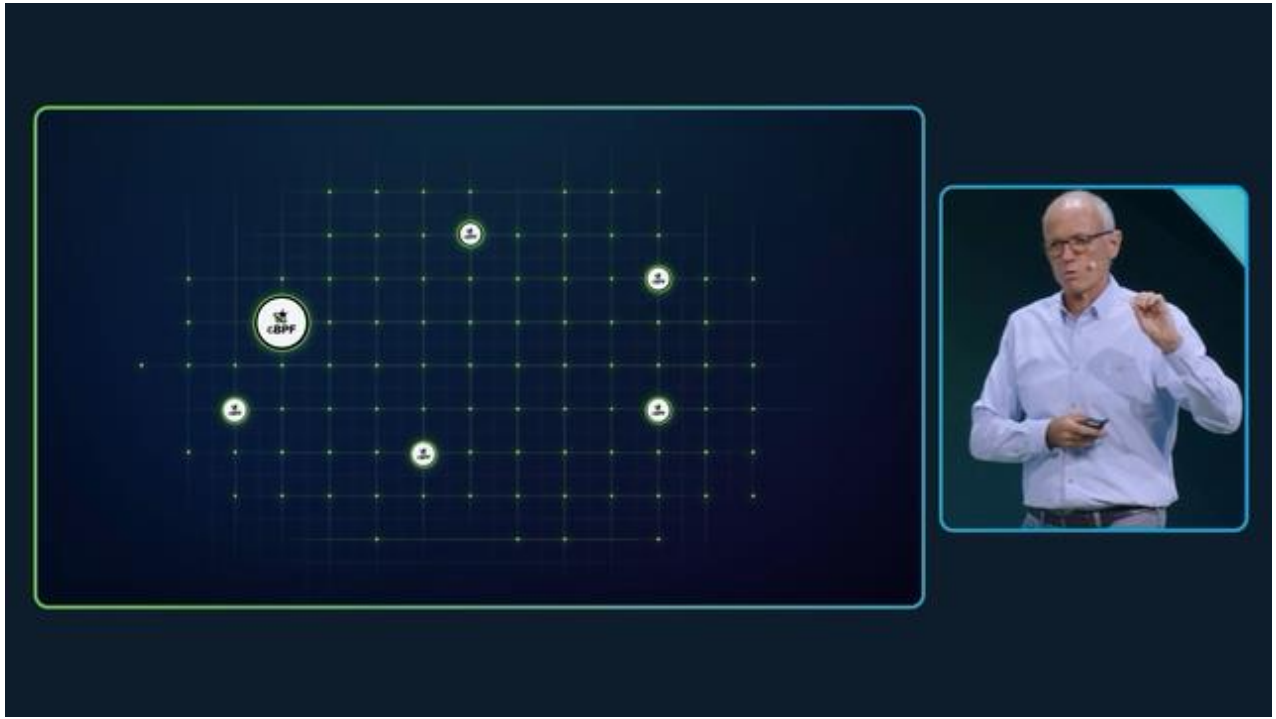
Nico Vibert & Raphaël Pinson
Technical Marketing Engineers, Isovalent at Cisco

BRKCLD-2696

Agenda

- Introduction
- Why are we here?
- What is eBPF ? *Why* eBPF?
- Kubernetes Networking Model
- Cilium Use Cases and Demos
- Conclusion

A message from Tom



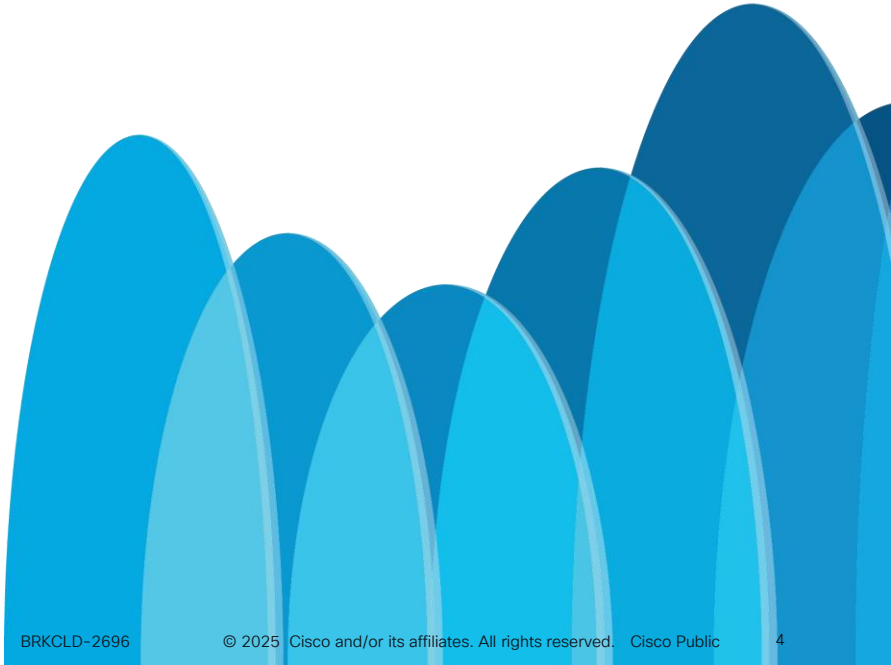
“eBPF is the future of networking.”

Tom Gillis

SVP/GM

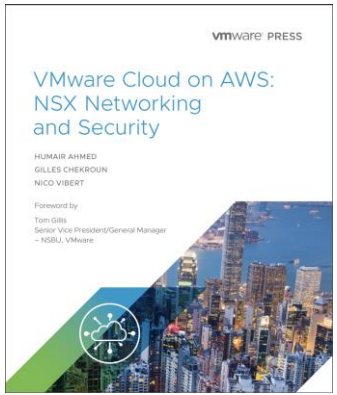
Security, Data Center, Internet & Cloud Infrastructure

Cisco



Hello, I'm Nico 👋

- Senior Staff Technical Marketing Engineer at Isovalent
- CCIE #22990 (expired… 😅)



"Cilium Up and Running"

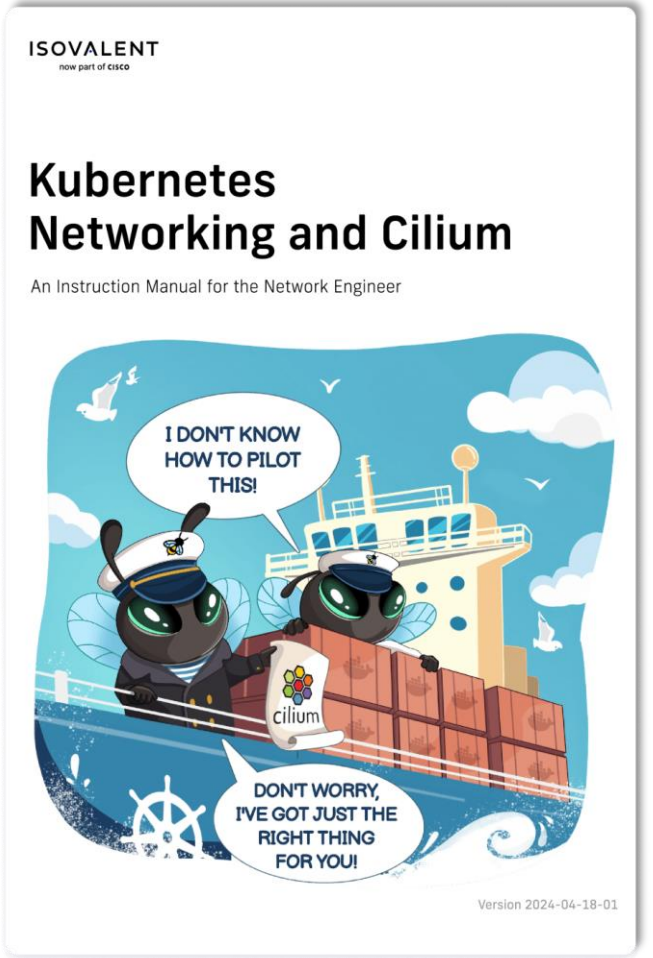
O'Reilly 2026

Kubernetes Networking eBook

- Written by a network engineer for network engineers
- Instruction manual focused on Kubernetes Networking and the de facto Kubernetes networking layer, Cilium.
- Over 5,000 downloads already!

Download the eBook here:

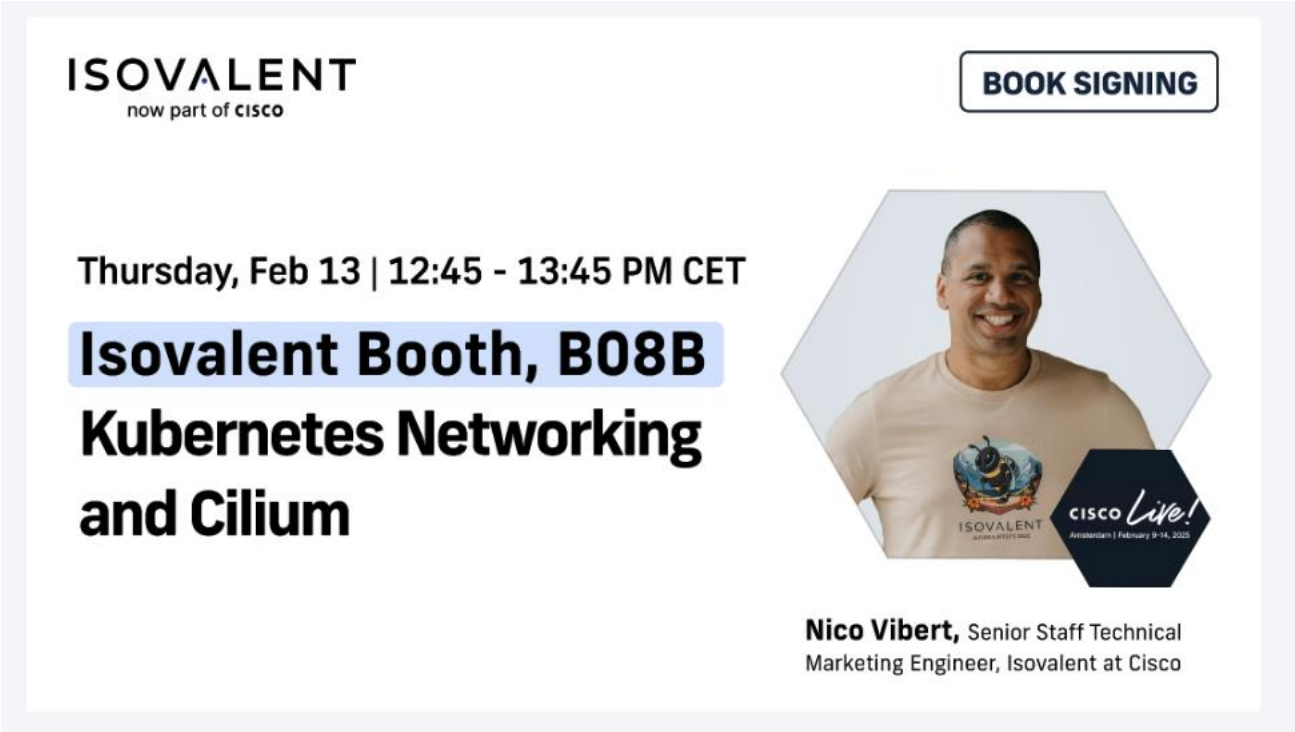
https://cs.co/k8s-for-network-engineers



Kubernetes Networking eBook

- Kubernetes Networking Fundamentals
- Introducing Cilium
- Where is my CLI?
- Where is my DHCP Server?
- What is a Kubernetes namespace?
- Where is my DNS Server?
- How Do Pods Talk To Each Other?
- How do I secure my cluster?
- What's identity-based security?
- Where's my Layer 7 firewall?
- Where's my Load Balancer?
- Where's my web proxy?
- How can I connect my cluster to existing networks?
- How do I connect my cluster with an external firewall?
- How do I manage and troubleshoot my network?
- How can I monitor and visualize network traffic?
- How do I start securing my cluster?
- How do I encrypt the traffic in my cluster?
- How do we connect clusters together?
- Is IPv6 supported on Kubernetes?
- Does the concept of QoS exist in Kubernetes?

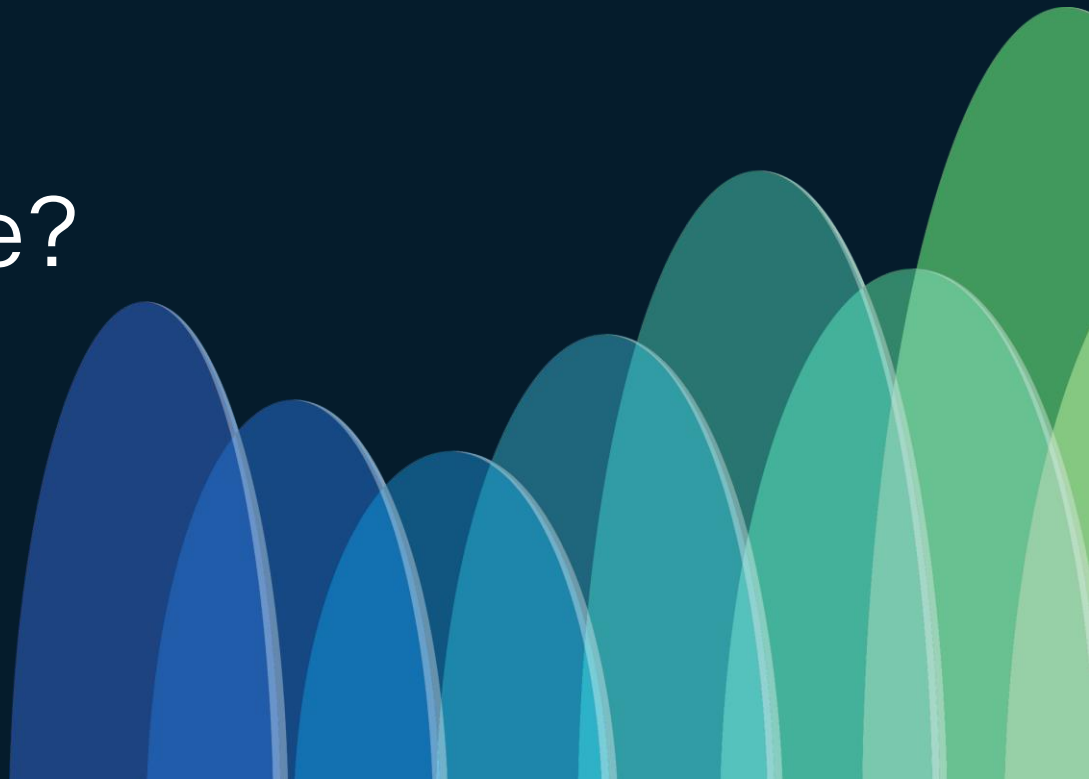
Kubernetes Networking eBook



Hello, I'm Raphaël

- Senior Technical Marketing Engineer at Isovalent
  - a.k.a. Cilium Alchemist
- CNCF Ambassador
- DevOps, Infra-As-Code, Platform Engineering

Why are we here?

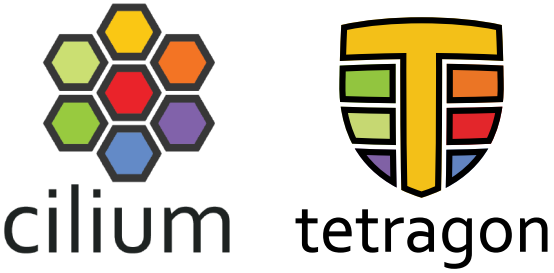


April 2024: Cisco completes the Isovalent acquisition

Cisco Completes Acquisition of Cloud Native Networking & Security Leader Isovalent

Thomas Graf | Published: Dec 21, 2023 | Updated: Apr 13, 2024 | Isovalent

Today, Cisco announced the completion of the acquisition of Isovalent. We are excited to continue our journey inside of Cisco SBG. When we started our journey seven years ago, writing the first few lines of Cilium source code, we couldn't imagine in our wildest dreams what Cilium would become. We founded Isovalent a year after starting the project, embarking on the mission of bringing the exciting eBPF technology to everyone by redefining what is possible in networking and security for the cloud native age with Cilium. While we evolved our technology and company, and built a thriving open source community, we hired a mind-blowingly talented team of amazing people and focused on building technology and products loved by customers. As we celebrate this major milestone for the team, we want to look back, but also look forward to see the exciting future ahead for Isovalent and Cisco.



- Open Source Projects
- Cilium and Tetragon use eBPF for networking and security

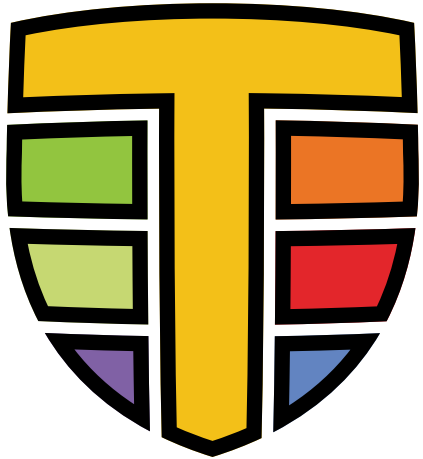
ISOVALENT

- Leading Company behind Cilium, Tetragon and eBPF
- Provides Enterprise distributions of Cilium and Tetragon

eBPF Networking and Security

cilium

= De-facto SDN for Kubernetes



tetragon

= Next-Gen Distributed Endpoint Security

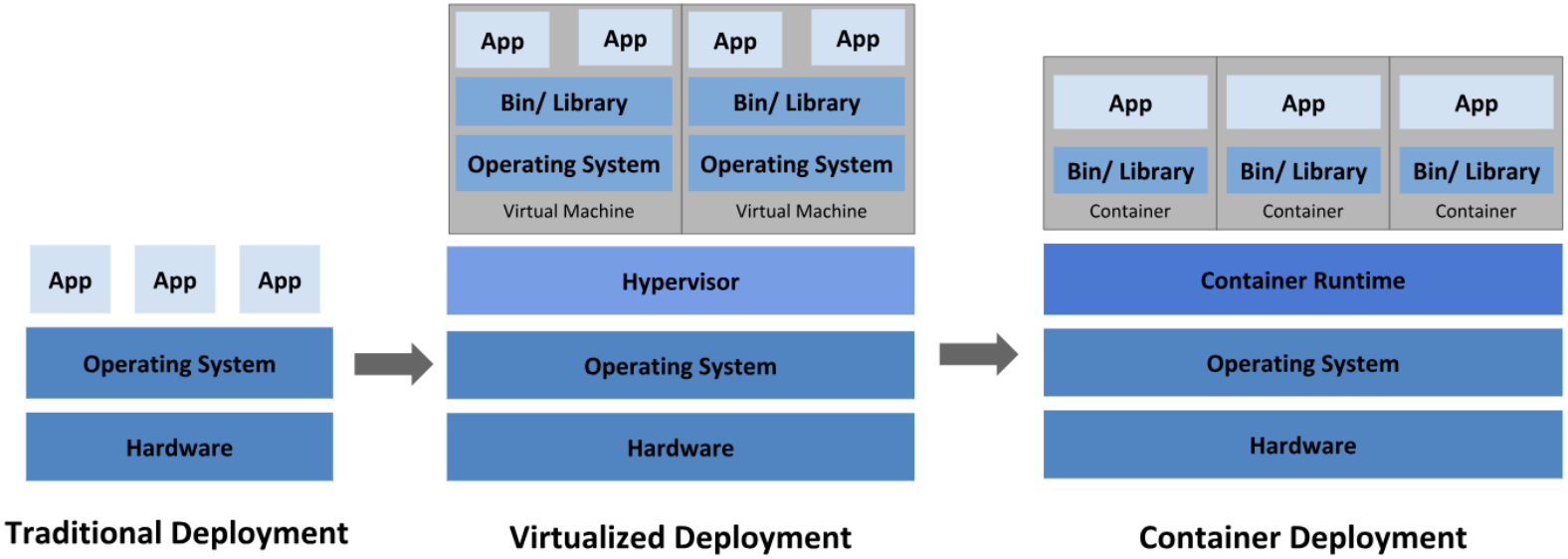
A brief history

2014-2024

Cloud Native Networking & Security

cisco Live!

While applications change deployment model…



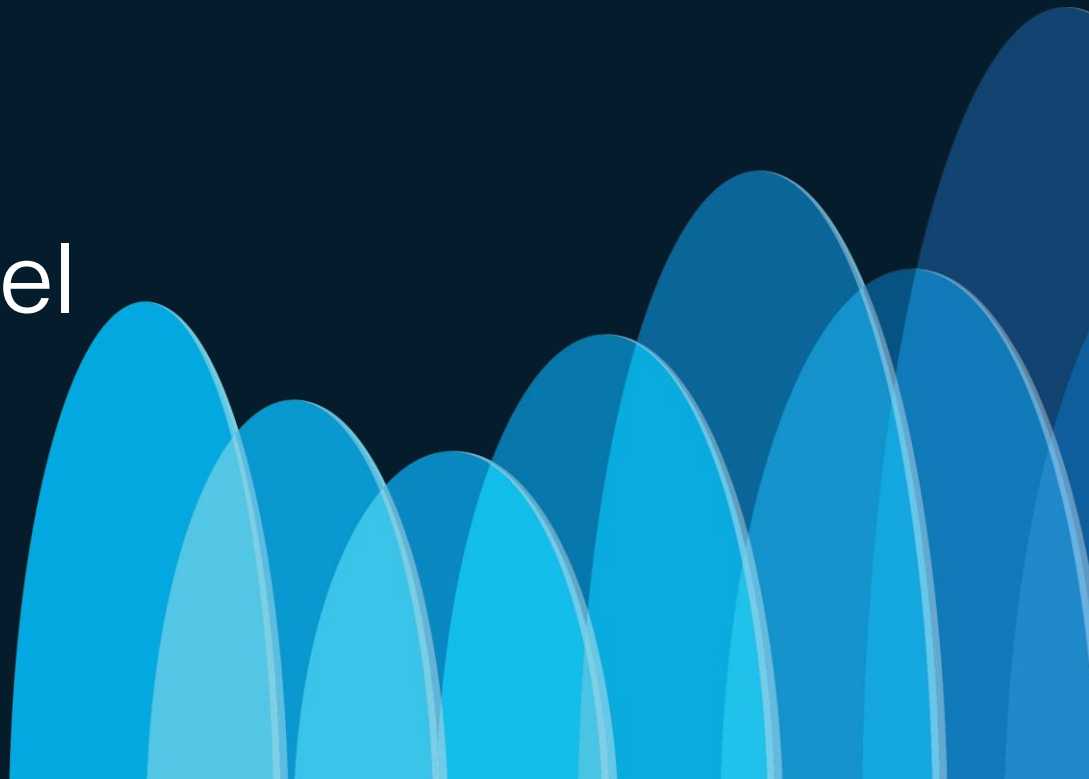
...the networking requirements remain the same

- We need our applications to
  - have accessible IP addresses
  - be able to communicate with one another
  - access the outside world (outbound access)
  - be accessible from the outside world (inbound access)
  - be secured and our data protected to meet regulatory goals
  - be globally resilient and highly available
  - be manageable and easy to troubleshoot

Traditional Networking vs Kubernetes Networking

Requirement	Traditional	Kubernetes
IP allocation	DHCP Server	
Connectivity	Switch/Router	
Network Security	FW/ACL	
Inbound Access	Load Balancer	
Encryption	MacSec / VPN	
Network Management	Ping, NetFlow, CLI, Observability Tooling	
Multi-Site	VXLAN/EVPN	

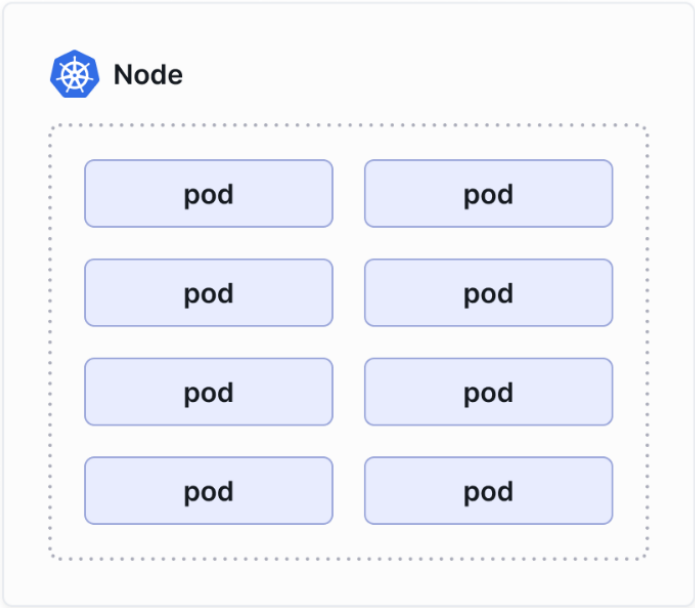
Kubernetes Networking Model



cisco Live!

Kubernetes Networking Model

- A **pod** consists of one or more containers.
- A pod can represent an entire application, a single replica of a distributed application, or an individual service in a micro-service architecture
- A pod will be running on a Kubernetes **node**.

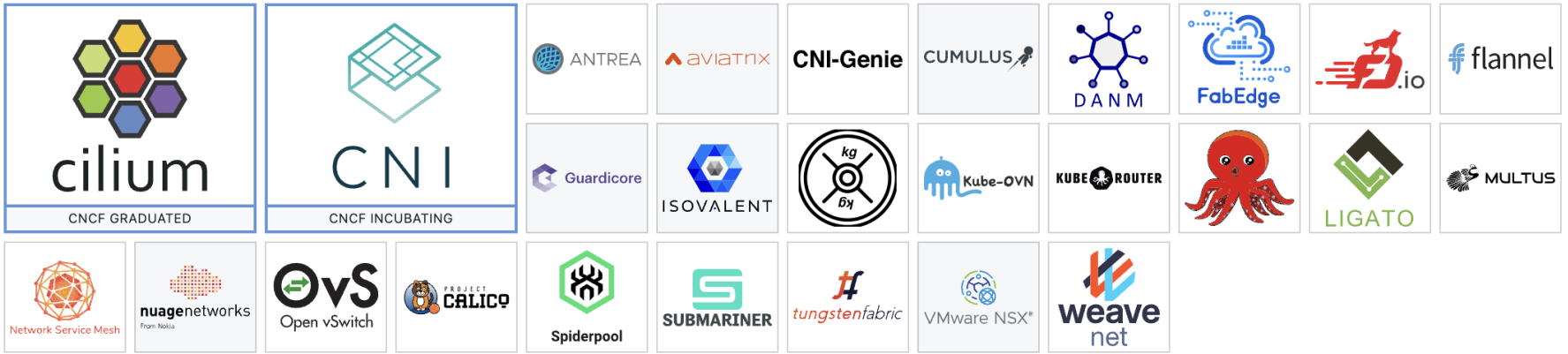


Kubernetes Networking Model

- A group of nodes makes up a Kubernetes **cluster**.
- Every pod in a cluster gets its unique cluster-wide IP address and can communicate with all other pods on any other node without using NAT.
- Pod-to-pod connectivity and IP address allocation requires a **Container Network Interface** (CNI) **plugin** to implement this model.

Kubernetes Networking Model

- Selecting a CNI for your Kubernetes deployment is like choosing a network vendor for a network refresh.
- Challenge: there's a lot of options to choose from:



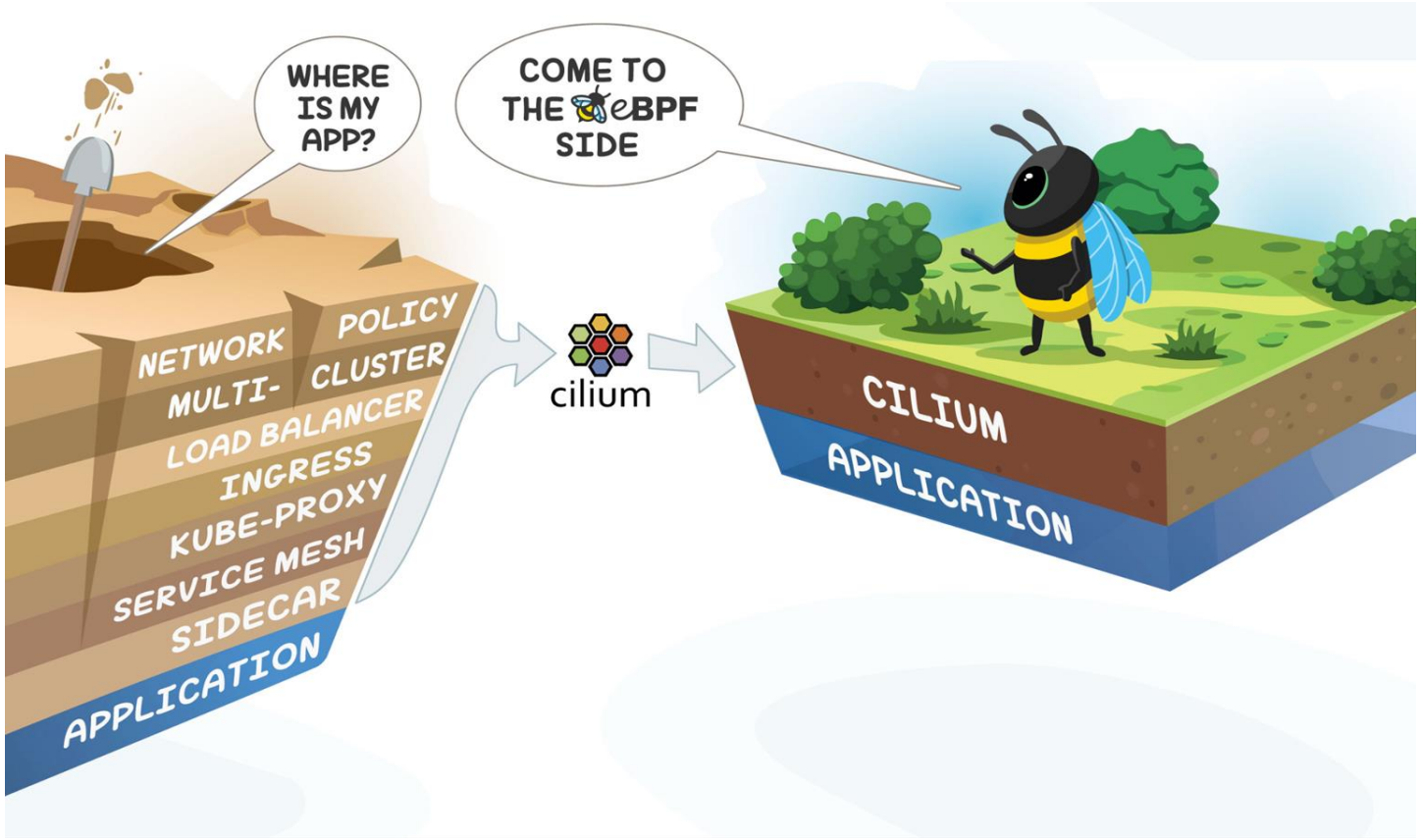
Traditional Networking vs Kubernetes Networking

Requirement	Traditional	Kubernetes
IP allocation	DHCP Server	CNI (IPAM)
Connectivity	Switch/Router	CNI
Network Security	FW/ACL	CNI (Network Policy)
Inbound Access	Load Balancer	Ingress
Encryption	MacSec / VPN	Service Mesh or CNI
Network Management	Ping, NetFlow, CLI, Observability Tooling	K8S Observability Tooling
Multi-Site	VXLAN/EVPN	Multi-Cluster Tooling

Traditional Networking vs Kubernetes Networking

Requirement	Traditional	Kubernetes
IP allocation	DHCP Server	
Connectivity	Switch/Router	
Network Security	FW/ACL	
Inbound Access	Load Balancer	
Encryption	MacSec / VPN	
Network Management	Ping, NetFlow, CLI, Observability Tooling	
Multi-Site	VXLAN/EVPN	

eBPF and Cilium to the rescue



cilium

Created by ISOVALENT

eBPF-based:

- Networking
- Security
- Observability
- Service Mesh & Ingress

Foundation

CLOUD NATIVE COMPUTING FOUNDATION

Technology

eBPF

cisco Live!

Wild Life

Building a Global Multi Cluster Gaming Infrastructure with Cilium

Adobe

What Makes a Good Multi-tenant Kubernetes Solution

Capital One

Building a Secure and Maintainable PaaS

Alibaba Cloud

Building High-Performance Cloud-Native Pod Networks

MÁSMÓVIL

Scaling a Multi-Tenant k8s Cluster in a Telco

Trip.com

First step towards cloud native networking

CENGN

Cloud Native Networking with eBPF

DigitalOcean

Managed Kubernetes: 1.5 Years of Cilium Usage at DigitalOcean

Google

Google chooses Cilium for Google Kubernetes Engine (GKE) networking

Bell

Why eBPF is changing the Telco networking space?

GitLab

Kubernetes Network Policies in Action with Cilium

aws

AWS picks Cilium for Networking & Security on EKS Anywhere

Scaleway

Scaleway uses Cilium as the default CNI for Kubernetes Kapsule

sportradar

Sportradar is using Cilium as their main CNI plugin in AWS (using kops)

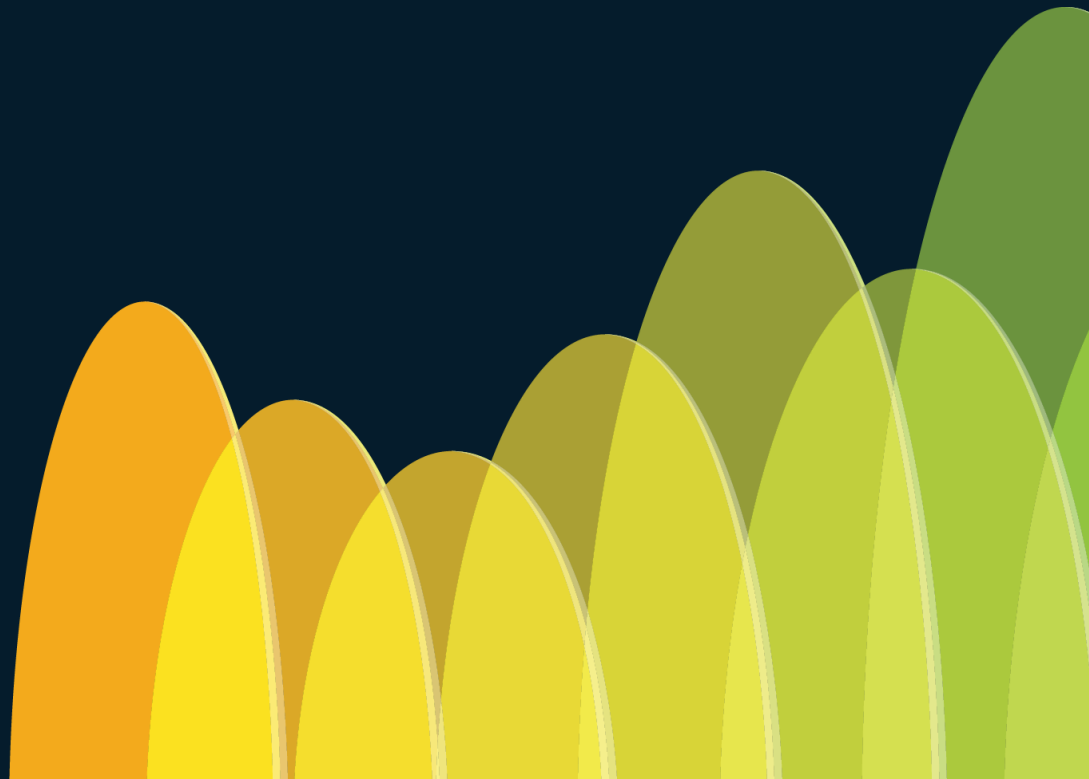
Utmost

Utmost is using Cilium in all tiers of its Kubernetes ecosystem to implement zero trust

yahoo!

Yahoo is using Cilium for L4 North-South Load Balancing for Kubernetes Services

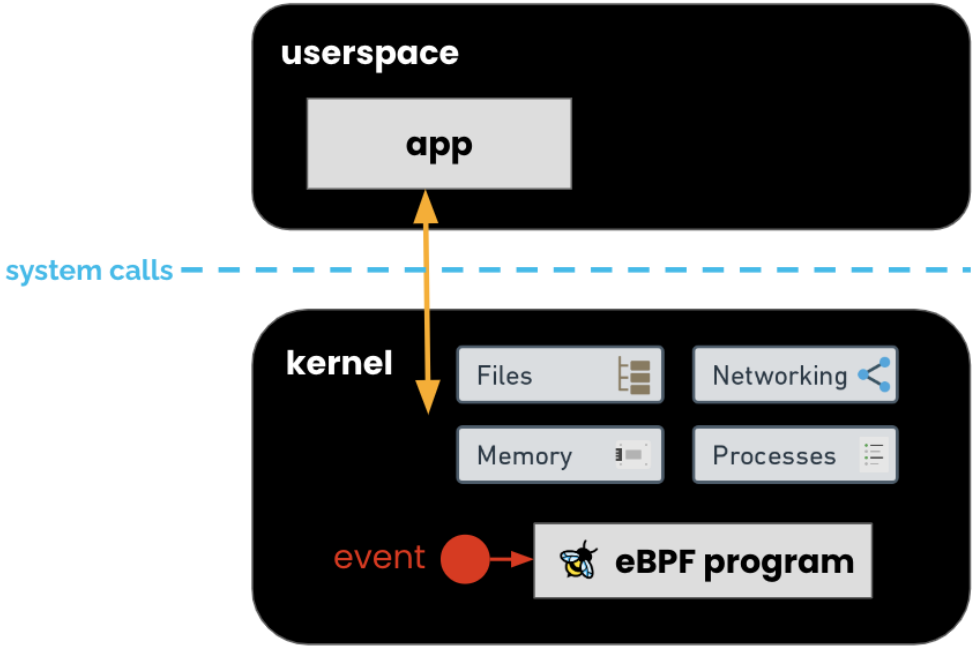
What is eBPF ?



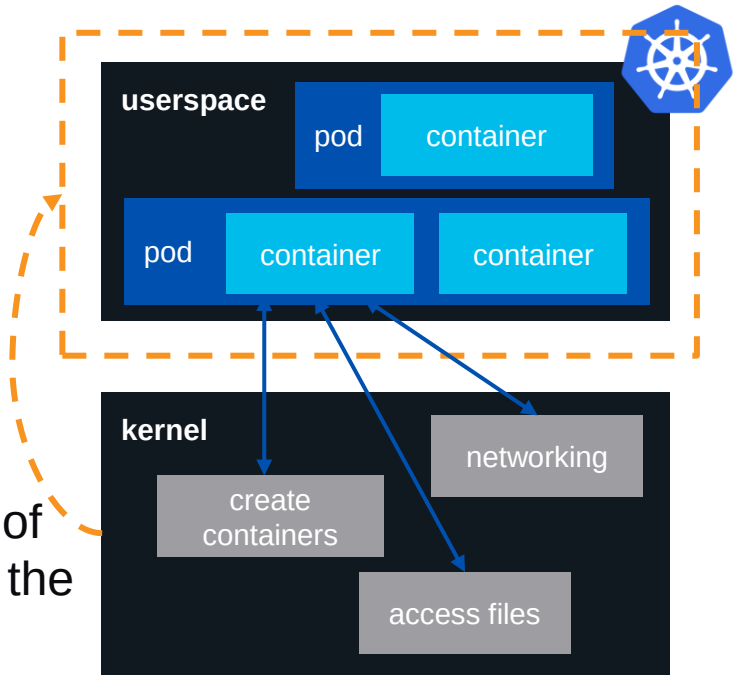
What is eBPF?

- Makes the kernel programmable
- Allows bespoke, dynamic changes to kernel behavior
- Enables high performance, low overhead infrastructure tools
  - Networking
  - Observability
  - Security

Run custom code in the kernel



eBPF and Kubernetes



Kernel aware of everything on the host

Applications run in containers in **pods**

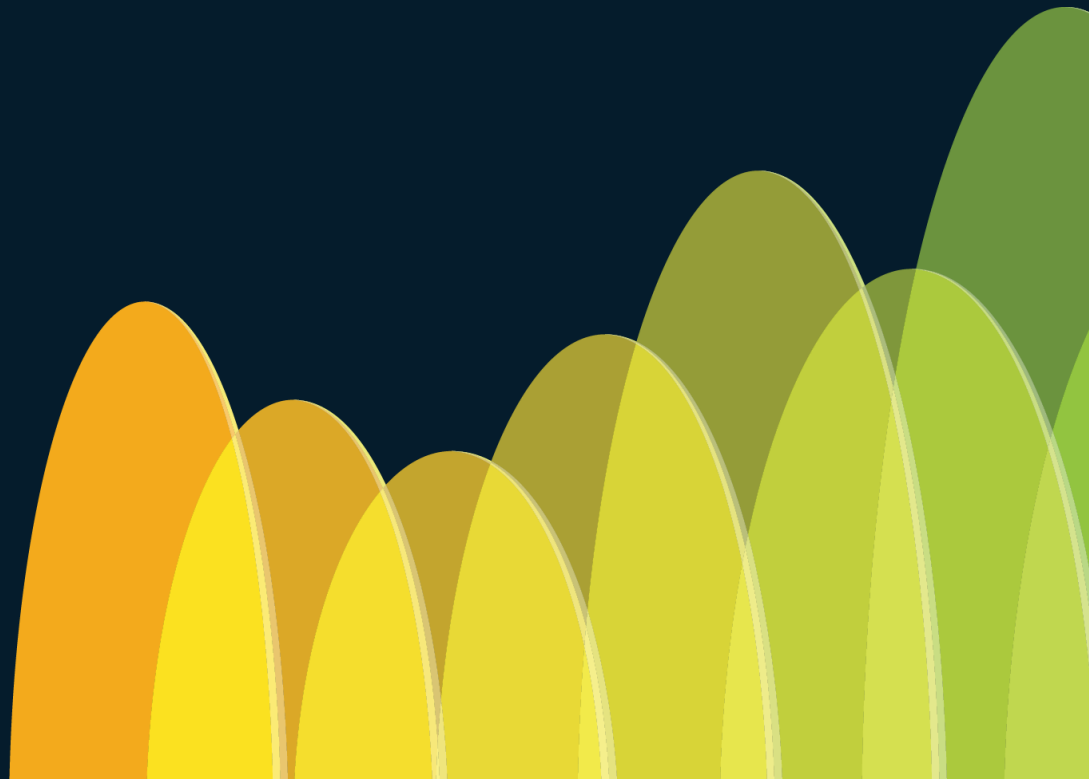
Pods are distributed across (virtual) machines (**nodes**) in a cluster

One kernel per node

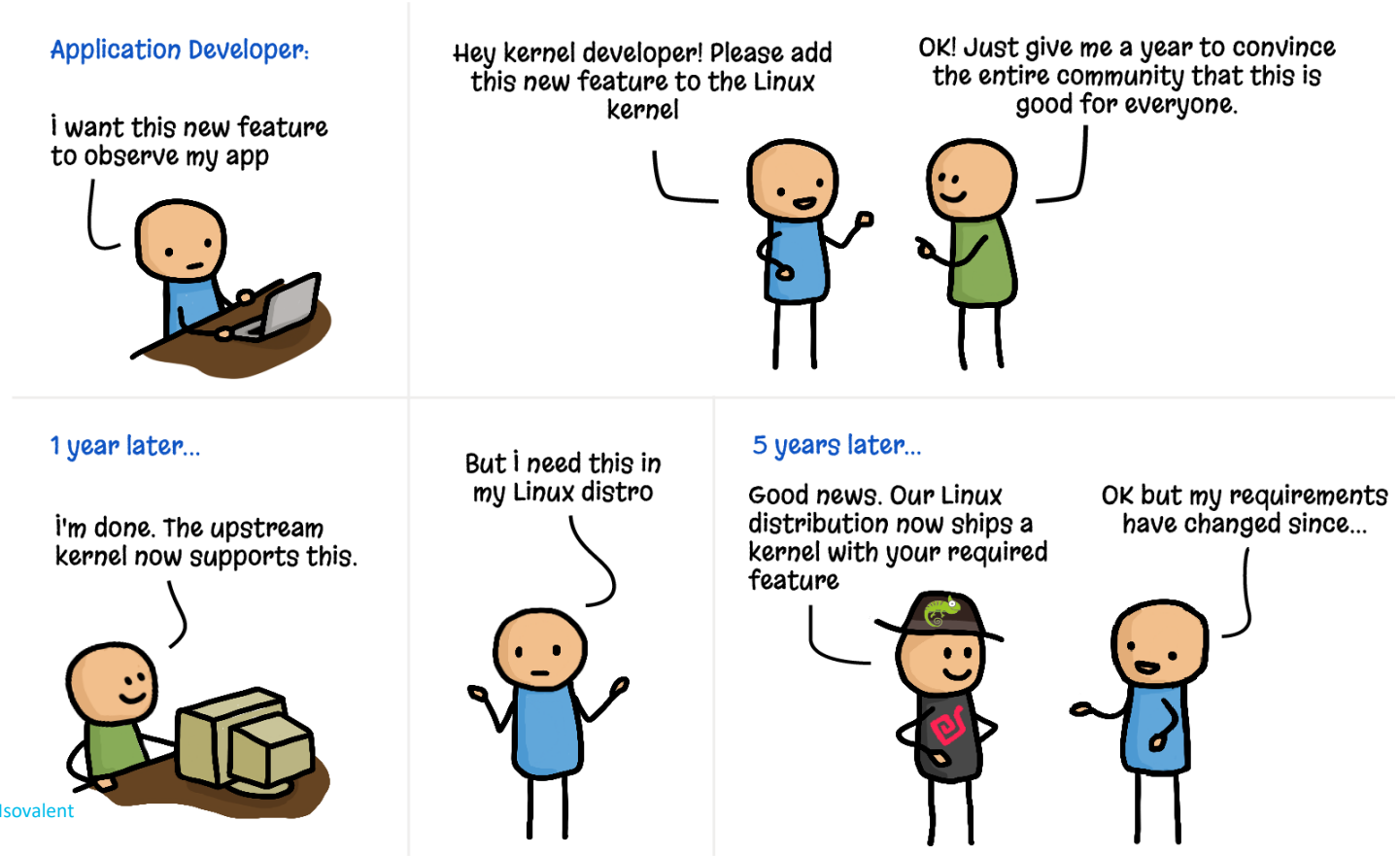
eBPF for network engineers

- Remember the Catalyst 6500?
  - Initially just used for switching packets
  - Insert modules to support additional use cases such as load balancing, security, wireless, etc…
  - eBPF programs are alike service modules
- eBPF programs are similar to:
  - virtual network function (VNF) concept in Telco networks
  - service insertion for Cisco ACI users.
- Note - you don't *have to* know eBPF: just like you don't *have to* know how ASICs are programmed.

Why eBPF ?

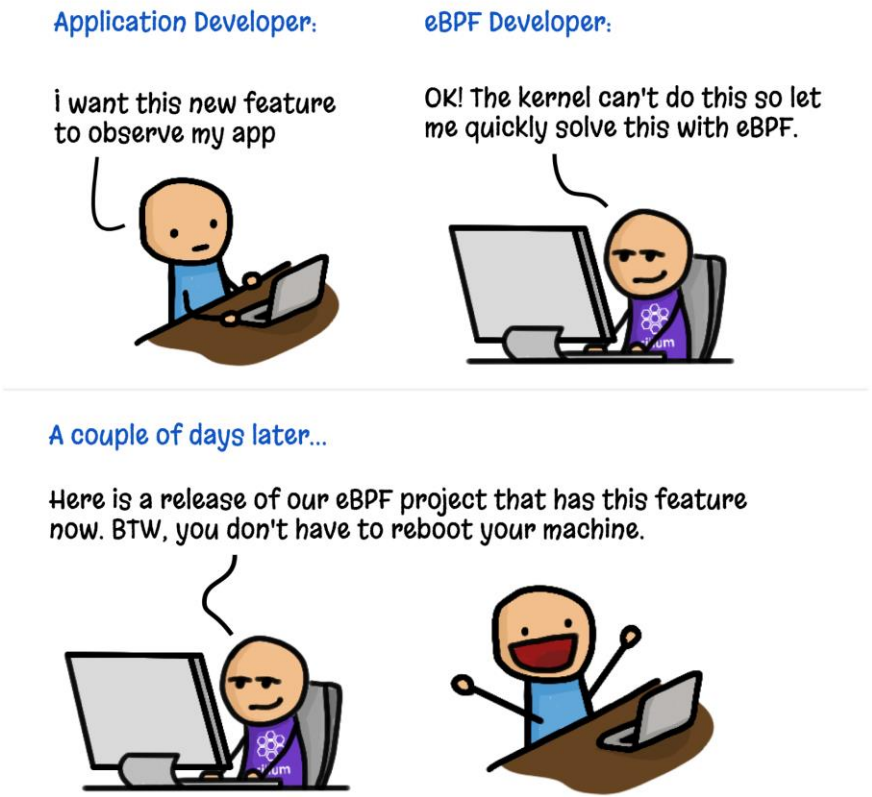


Without eBPF



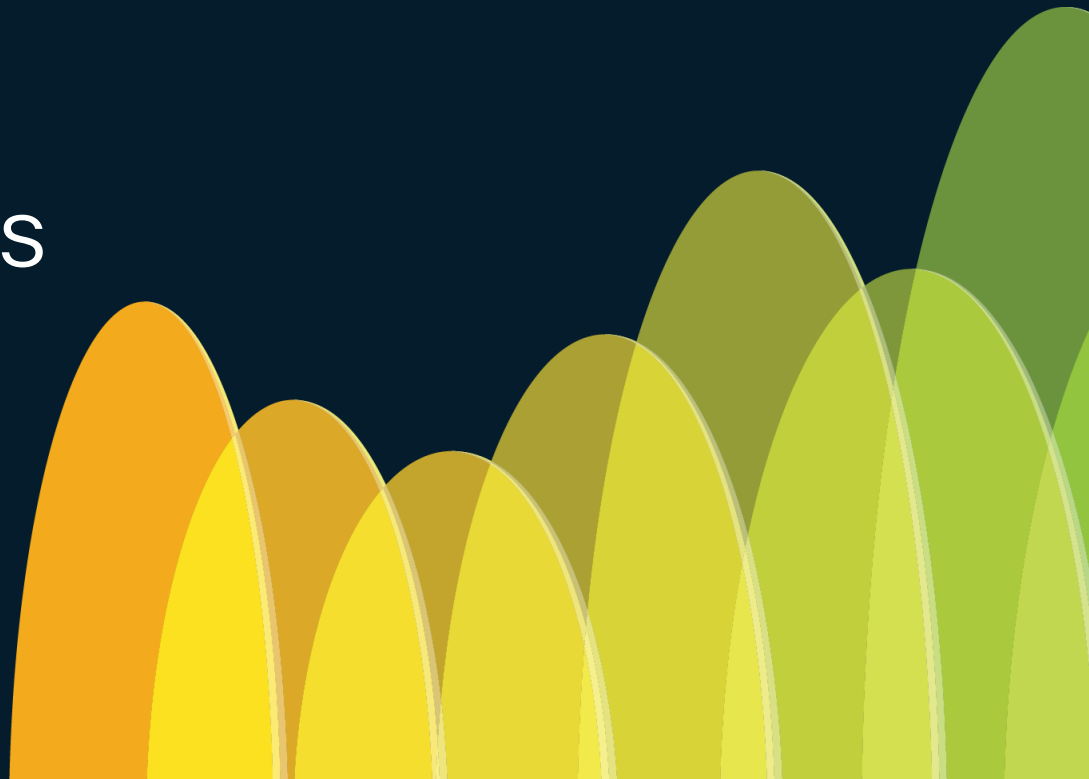
Cartoon by Vadim Shchekoldin, Isovalent

With eBPF



Cartoon by Vadim Shchekoldin, Isovalent

Cilium Use Cases

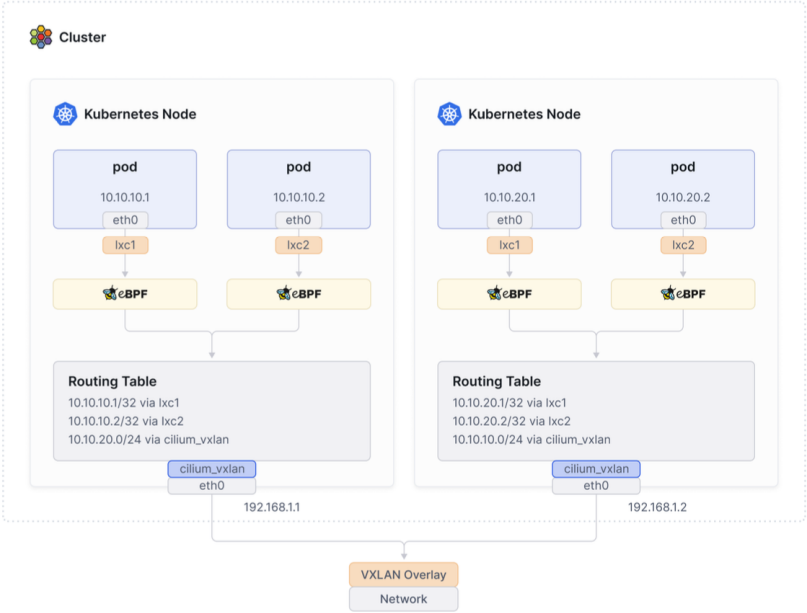


Cilium for Networking

cisco Live!

Cilium as a Fabric

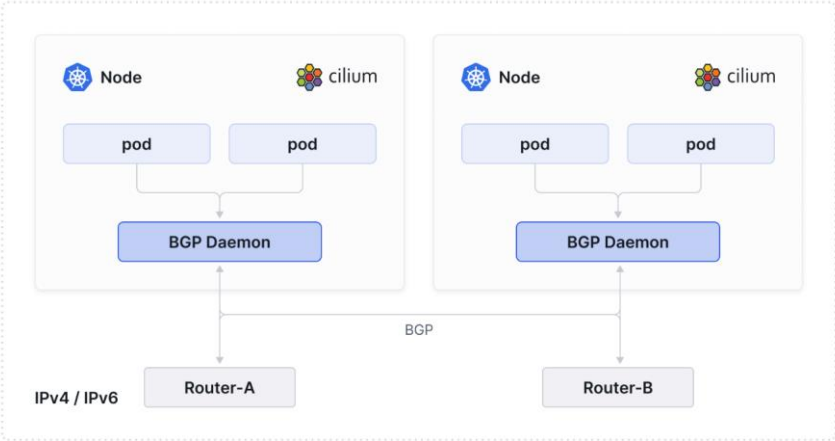
- Provide connectivity between pods located in different nodes or between different clusters
- Builds a network overlay (VXLAN/Geneve) between nodes or leverages underlying network
- Also provides IPAM (==DHCP) functions



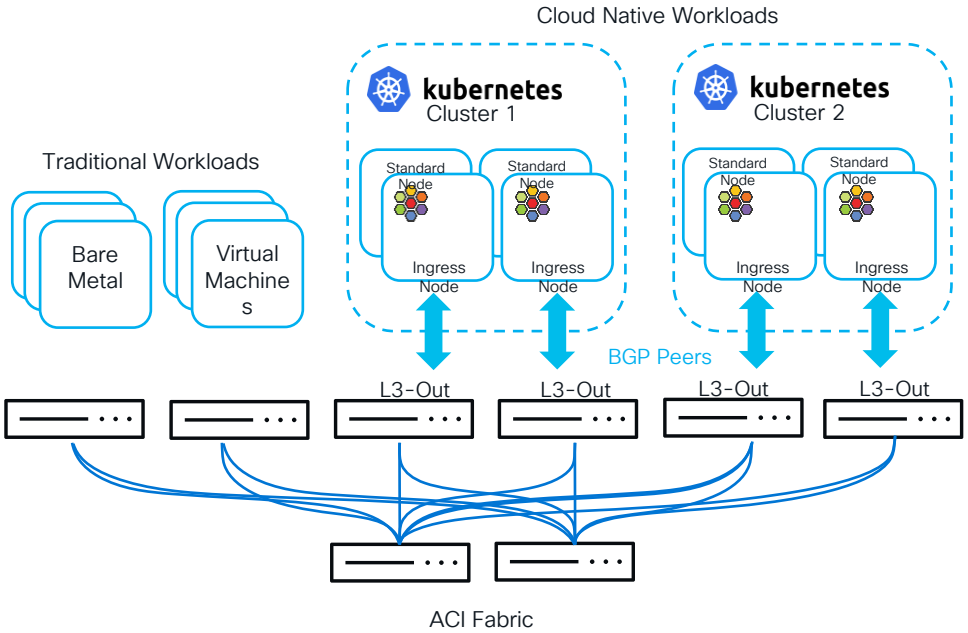
Encapsulation Routing Mode

Cilium as a BGP Router

- Connect back to the core network (ACI/NX-OS for example) to advertise cluster IP addresses
- Enable Kubernetes apps to be accessible to workloads outside the cluster
- Cilium natively support common BGP features (Custom timers, Graceful Restart, MD5 auth, BFD, etc...)



ACI and Cilium – K8s peering over BGP



Cilium and ACI Connected

Seamless routing between Kubernetes/Cilium and ACI

Cilium & ACI Full Stack Automation

Ready Solutions such as Argo and Flux *(CX Cloud Platform Automation)*

Multi-Cloud & Multi-Data Centre

Multicloud connectivity for cloud native and traditional workloads

Zero Trust & Microsegmentation

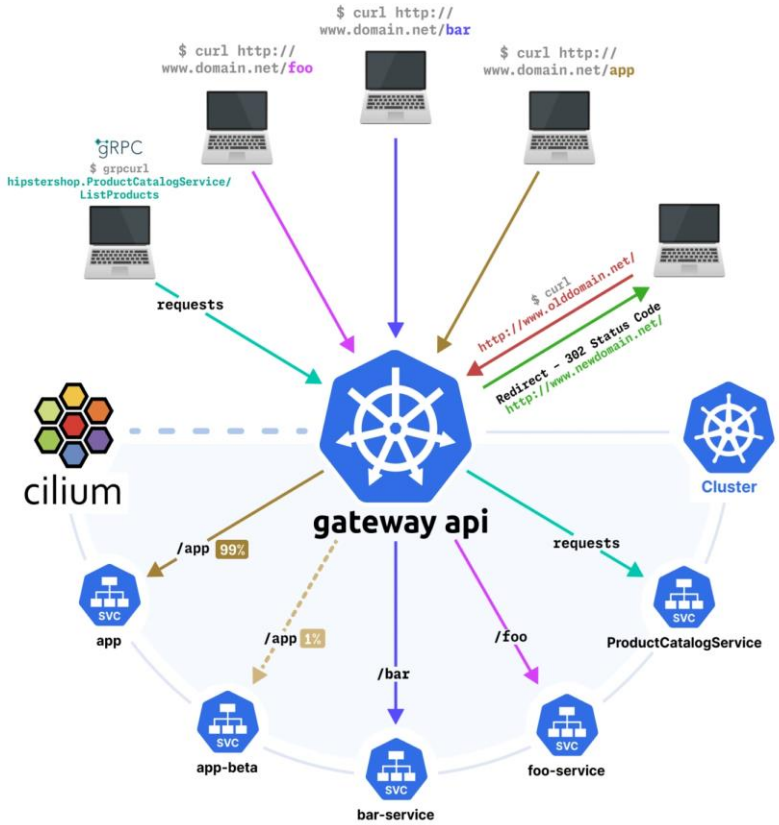
Identity based eBPF powered security enforcement at source

Observability

Full stack observability supporting Prometheus metrics and SIEM integration with Splunk

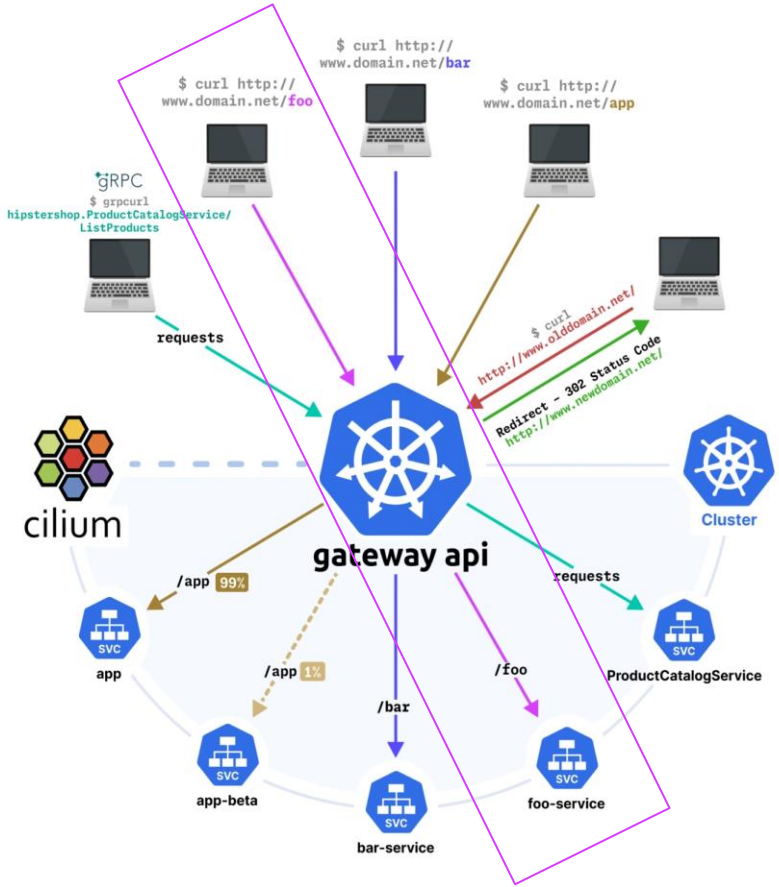
Cilium as a L7 Load Balan

- Cilium natively supports Ingress Controller and its successor Gateway AP
- Comparable to external-facing load-balancer
- Supports multitude of use cases:
  - HTTP routing
  - HTTP traffic splitting and load-balancing
  - HTTP request and response header rewrite
  - HTTP redirect and path rewrites
  - HTTP mirroring
  - SSL Offload



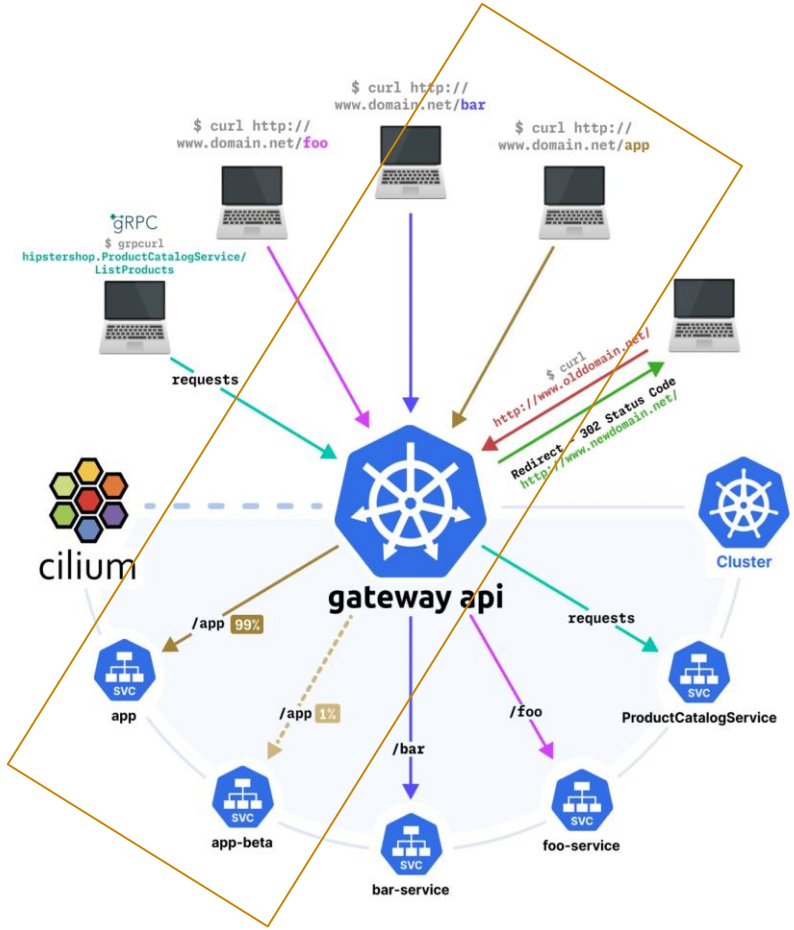
L7-Aware Routing

```yaml
apiVersion: gateway.networking.k8s.io/v1
kind: HTTPRoute
metadata:
  name: foo-app-route
spec:
  parentRefs:
  - name: my-cilium-gateway
    namespace: default
  rules:
  - matches:
    - path:
        type: PathPrefix
        value: /foo
    backendRefs:
    - name: foo-service
      port: 9080
```



Canary Testing

```yaml
apiVersion: gateway.networking.k8s.io/v1
kind: HTTPRoute
metadata:
  name: load-balancing-route
spec:
  parentRefs:
  - name: my-cilium-gateway
  rules:
  - matches:
    - path:
        type: PathPrefix
        value: /app
    backendRefs:
    - kind: Service
      name: app
      port: 8080
      weight: 99
    - kind: Service
      name: app-beta
      port: 8090
      weight: 1
```

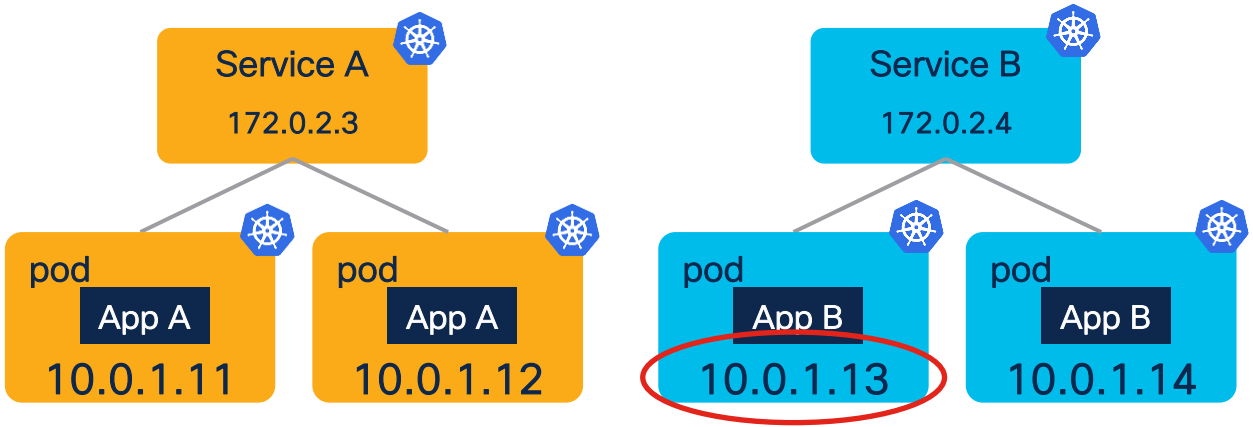


Demo

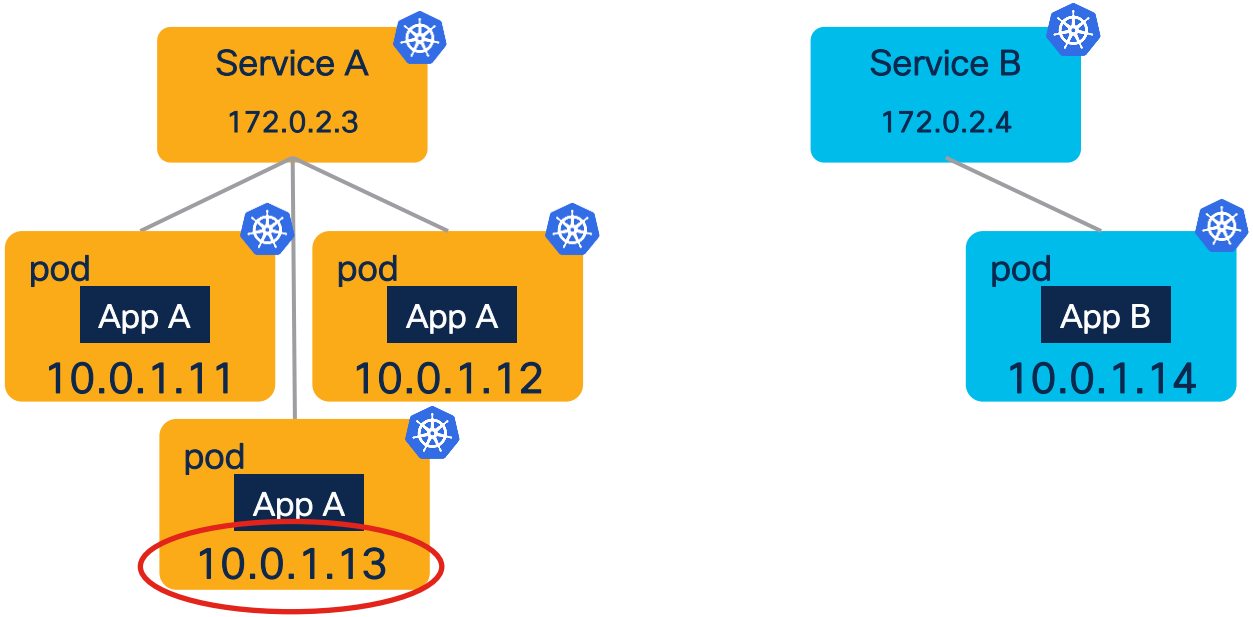
Cilium for Security

cisco Live!

Containers / pods are ephemeral



Containers / pods are ephemeral



What's different in Cloud Native?

- Pods come and go dynamically e.g. scaling
- IP addresses can be reused for different pods
- Resources in a cluster can be divided into **namespaces**
- Namespaces should not be considered not security boundaries
- IP addresses names are not meaningful enough and cannot be used as security identities.
- Instead, we need Kubernetes identities

Network Policy 101 (1/2)

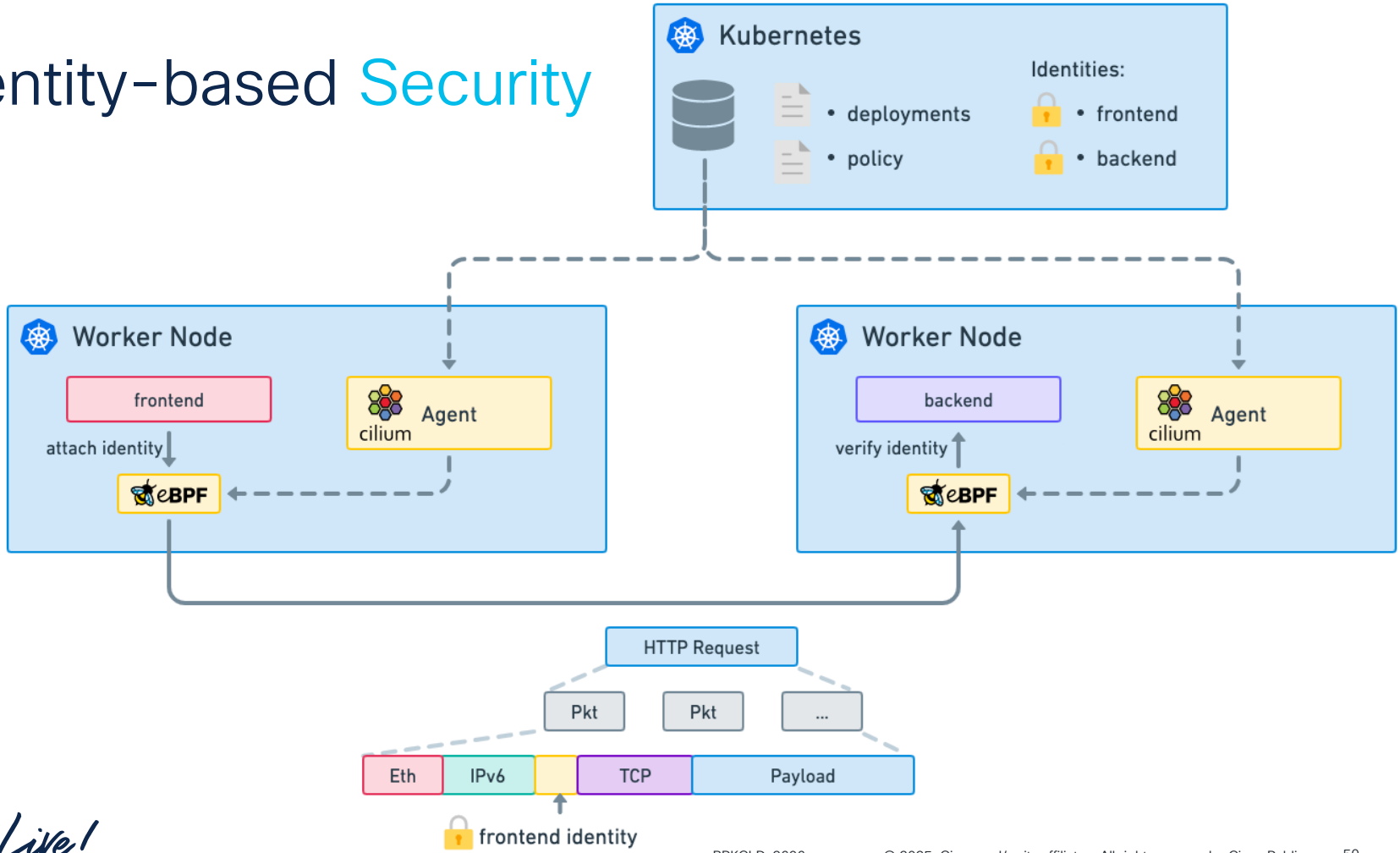
- Define which flows are allowed within a cluster
- The equivalent of ACLs in Kubernetes
- Kubernetes Network Policies are basic L3/L4 policies
- Cilium Network Policies are more granular and support:
  - L3/L4
  - L7
  - FQDN-based
- Network Policies rely on identities rather than IP addresses

Network Policy 101 (1/2)

- Business logic is described in labels
- Labels are a well-structured method to describe what a Pod does
- Labels are derived to create identities
- Security rules are enforced between identities

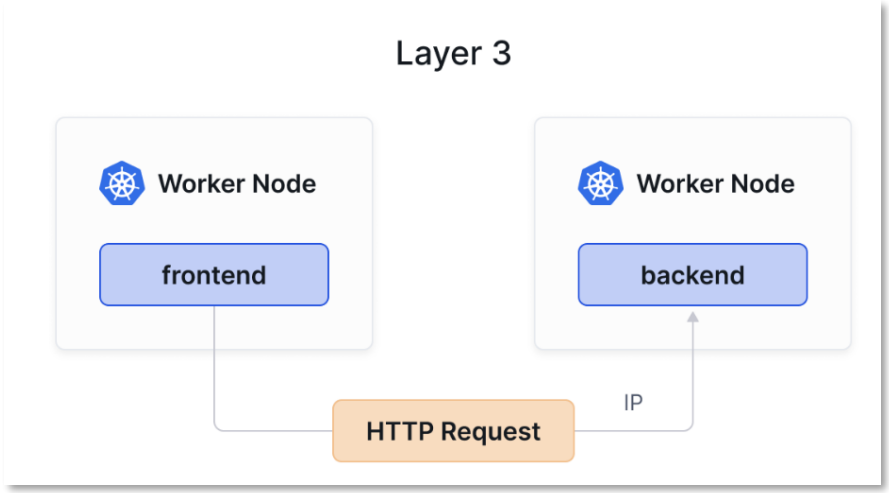
```yaml
apiVersion: v1
kind: Pod
metadata:
  name: sample-pod
  labels:
    app: myapp
    tier: frontend
    environment: production
spec:
  containers:
  - name: nginx-container
    image: nginx:latest
    ports:
    - containerPort: 80
```

Identity-based Security



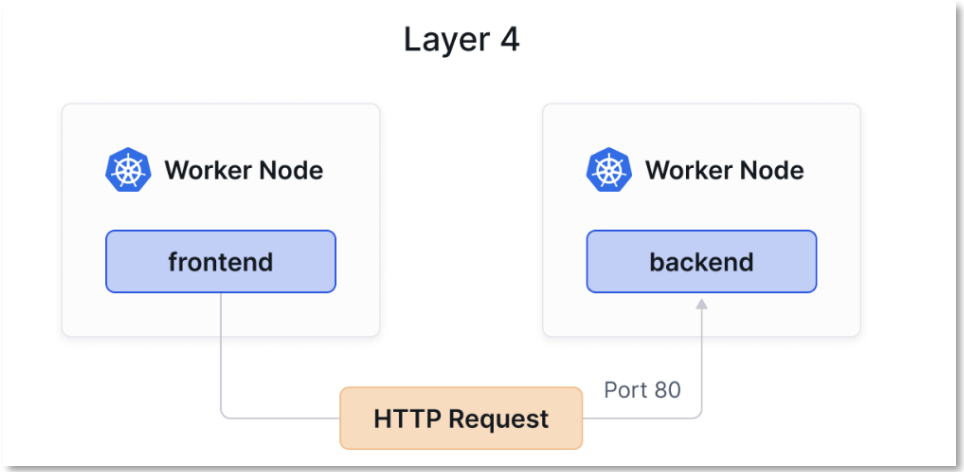
L3 Network Policies

```yaml
apiVersion: "cilium.io/v2"
kind: CiliumNetworkPolicy
metadata:
  name: "l3-rule"
spec:
  endpointSelector:
    matchLabels:
      role: backend
  ingress:
  - fromEndpoints:
    - matchLabels:
        role: frontend
```



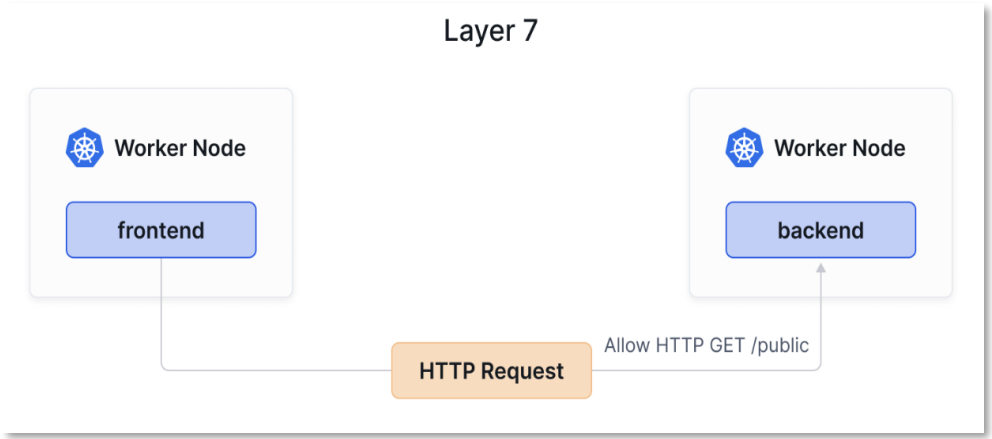
L4 Network Policies

```yaml
apiVersion: "cilium.io/v2"
kind: CiliumNetworkPolicy
metadata:
  name: "l4-rule"
spec:
  endpointSelector:
    matchLabels:
      role: frontend
  egress:
    - toPorts:
      - ports:
        - port: "80"
          protocol: TCP
```

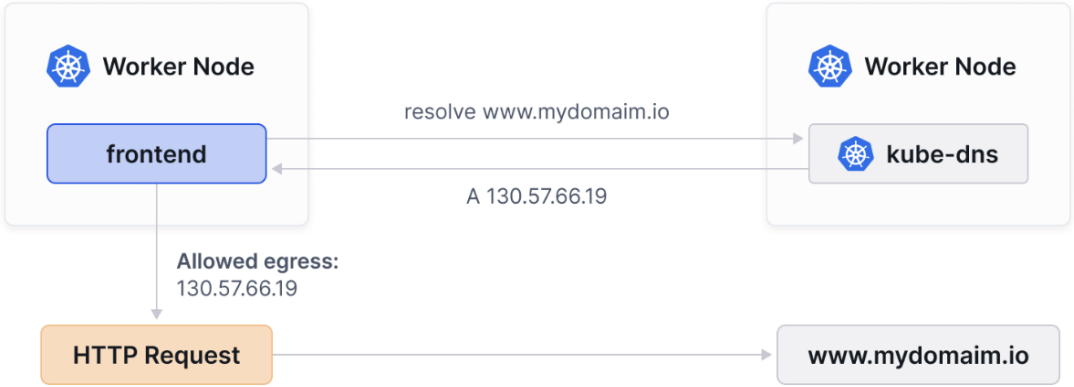


L7 Network Policies

```yaml
apiVersion: "cilium.io/v2"
kind: CiliumNetworkPolicy
metadata:
  name: "l7-rule"
spec:
  endpointSelector:
    matchLabels:
      role: frontend
  egress:
    - toPorts:
      - ports:
        - port: "80"
          protocol: TCP
        rules:
          http:
          - method: "GET"
            path: "/public"
```



DNS-aware Cilium Network Policy




```yaml
apiVersion: "cilium.io/v2"
kind: CiliumNetworkPolicy
[...]
specs:
- endpointSelector:
  matchLabels:
    app: frontend
  egress:
  - toFQDNs:
    - matchName: "*.mydomain.io"
    toPorts:
    - ports:
      - port: "443"
        protocol: TCP
```

Anatomy of a Network Policy

Who the policy applies to

In which traffic direction does this policy apply to? (**egress**: traffic **from** the pods this policy applies to)

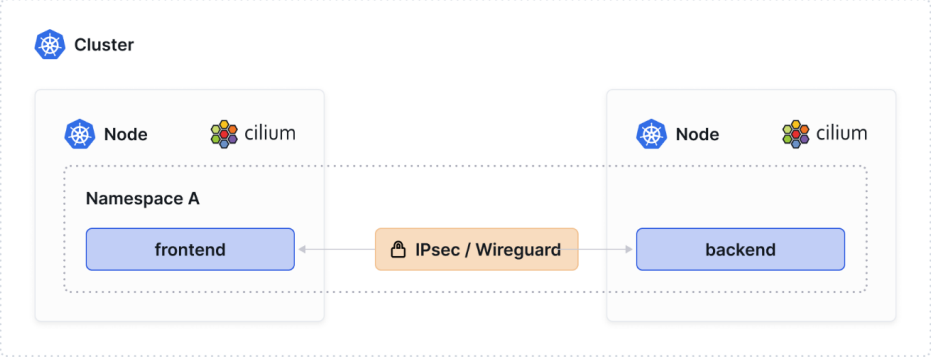
To which destinations (domains or endpoints

Over which ports

```yaml
apiVersion: cilium.io/v2
kind: CiliumNetworkPolicy
metadata:
  name: egress-policy
  namespace: endor
spec:
  endpointSelector:
    matchLabels:
      class: tiefighter
      org: empire
  egress:
  - toFQDNs:
    - matchName: disney.com
    toPorts:
    - ports:
      - port: "443"
  - toFQDNs:
    - matchName: swapi.dev
    toPorts:
    - ports:
      - port: "443"
    toEndpoints:
    - matchLabels:
        class: deathstar
        org: empire
    toPorts:
    - ports:
      - port: "80"
```

Cilium for encryption

- Cilium can natively encrypt traffic with either:
  - IPsec
  - WireGuard
- IPsec provides more customization but requires more management than WireGuard.
- Great option when hardware encryption offload like MacSec are not supported.



Demo

Cilium for Observability

cisco Live!

Cilium for observability

```
$ kubectl get pods
NAME                                READY   STATUS    RESTARTS   AGE
tiefighter                  1/1     Running   0          2m34s
xwing                       1/1     Running   0          2m34s
deathstar-5b7489bc84-crlxh  1/1     Running   0          2m34s
deathstar-5b7489bc84-j7qwq  1/1     Running   0          2m34s

$ hubble observe --follow -l class=xwing
# DNS lookup to coredns
default/xwing:41391 (ID:16092) -> kube-system/coredns-66bff467f8-28dgp:53 (ID:453) to-proxy FORWARDED (UDP)
kube-system/coredns-66bff467f8-28dgp:53 (ID:453) -> default/xwing:41391 (ID:16092) to-endpoint FORWARDED (UDP)
# ...
# Successful HTTPS request to www.disney.com
default/xwing:37836 (ID:16092) -> www.disney.com:443 (world) to-stack FORWARDED (TCP Flags: SYN)
www.disney.com:443 (world) -> default/xwing:37836 (ID:16092) to-endpoint FORWARDED (TCP Flags: SYN, ACK)
www.disney.com:443 (world) -> default/xwing:37836 (ID:16092) to-endpoint FORWARDED (TCP Flags: ACK, FIN)
default/xwing:37836 (ID:16092) -> www.disney.com:443 (world) to-stack FORWARDED (TCP Flags: RST)
# ...
# Blocked HTTP request to deathstar backend
default/xwing:49610 (ID:16092) -> default/deathstar:80 (ID:16081) Policy denied DROPPED (TCP Flags: SYN)
```

Flow Metadata

- Ethernet headers
- IP & ICMP headers
- UDP/TCP ports, TCP flags
- HTTP, DNS, Kafka, ...

Kubernetes

- Pod names and labels
- Service names
- Worker node names

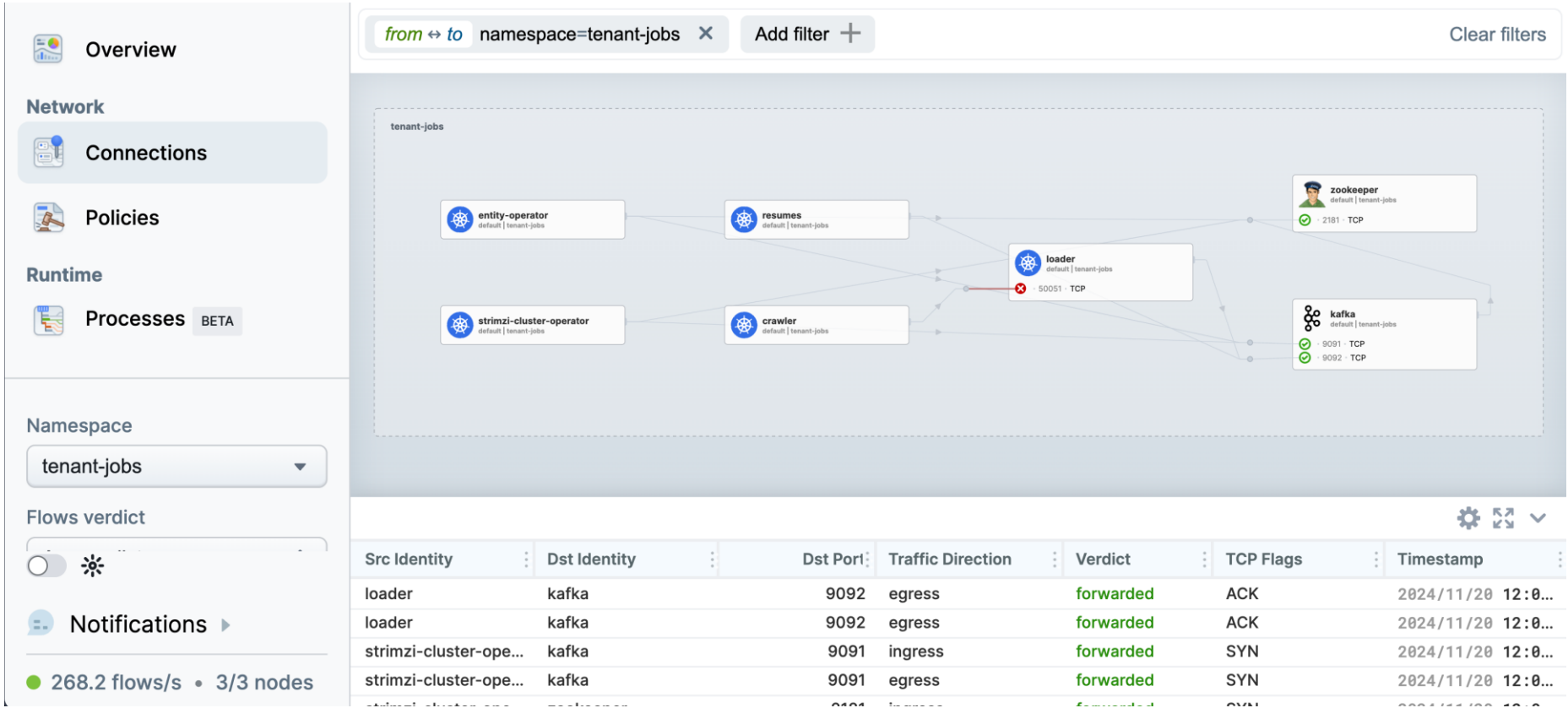
DNS

- FQDN for source and destination

Cilium

- Security identities and endpoints
- Drop reasons
- Policy verdict matches

Cilium for observability



Demo

Cilium for Performance

cisco Live!

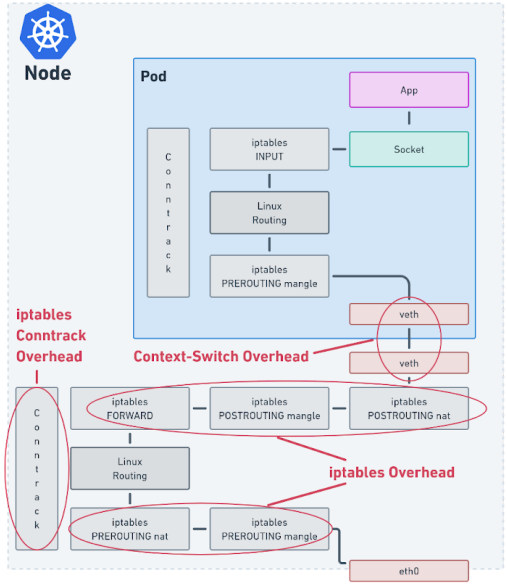
Cilium for High Performance Networking

AI workloads have strict demands from the network:

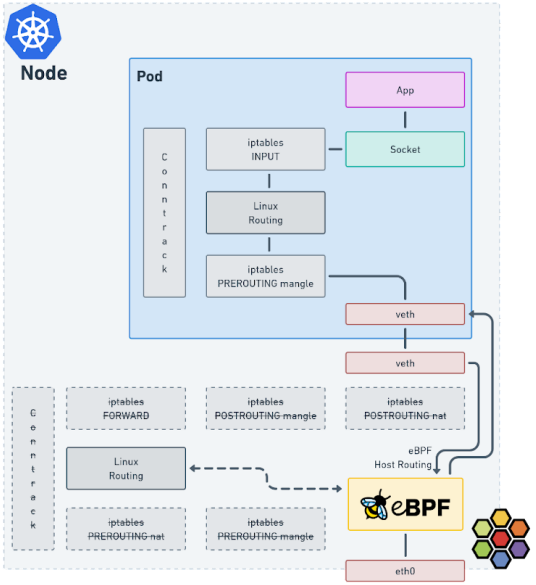
- Low latency and high bandwidth
- Scalable and reliable networking
- Secured APIs
- Encryption at speed

Liberating Kubernetes from iptables

eBPF-based Kube-Proxy Replacement



Standard Container Networking



Cilium eBPF Container Networking

Liberating Kubernetes from iptables

eBPF-based Kube-Proxy Replacement

```
root@aks-nodepool1-20100607-vmss000000:/# iptables-save | grep -c KUBE-SEP
432
root@aks-nodepool1-20100607-vmss000000:/# iptables-save | grep -c KUBE-SVC
423
```

Without eBPF

Liberating Kubernetes from iptables

eBPF-based Kube-Proxy Replacement

```
root@aks-nodepool1-33954605-vmss000000:/# iptables-save | grep -c KUBE-SVC
0
root@aks-nodepool1-33954605-vmss000000:/# iptables-save | grep -c KUBE-SEP
0
```

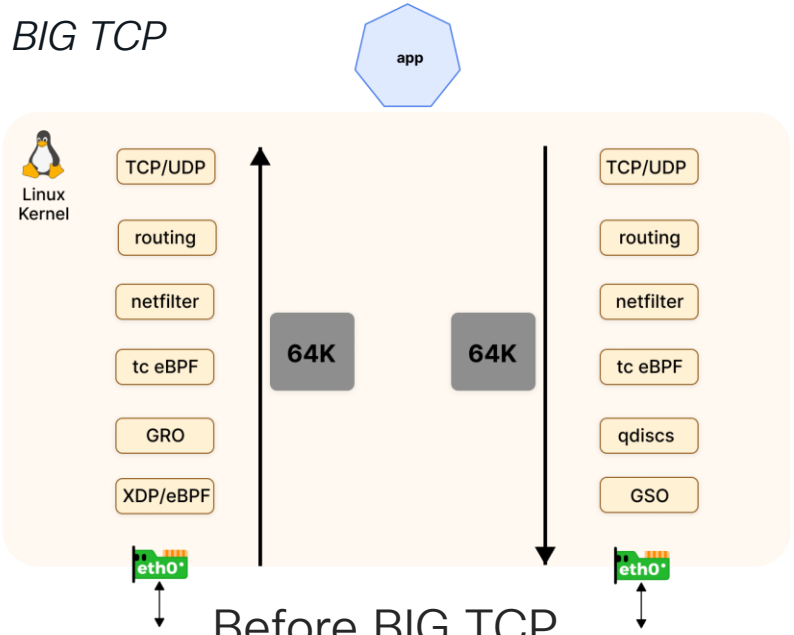
With eBPF

Cilium for innovative networking

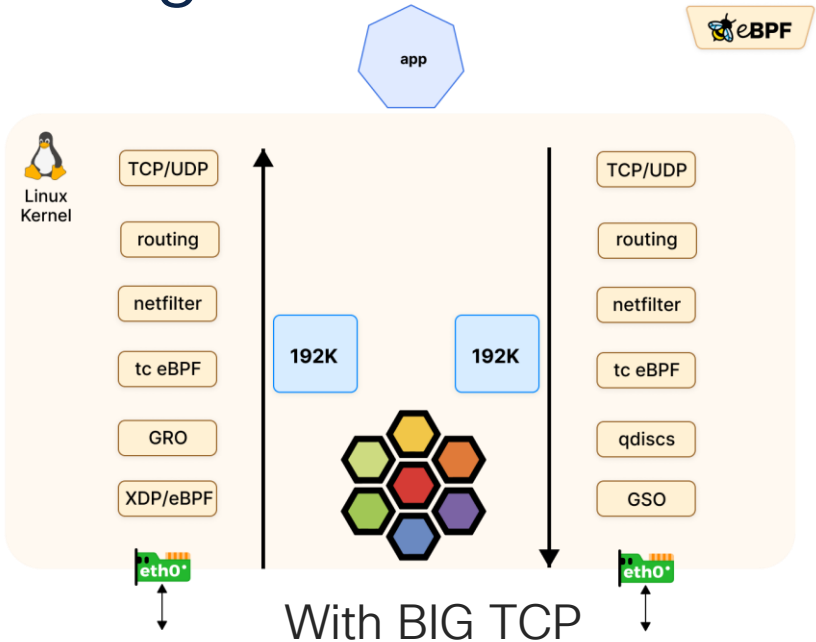
- 100G networks mean 8,000,000 packets per second to process (assuming 1,538 Bytes MTU)
- Leaves 120 nanoseconds per packet to process!
- We need bigger packets
- We need BIG TCP

Cilium for innovative networking

BIG TCP



Before BIG TCP



With BIG TCP

Cilium for innovative networking

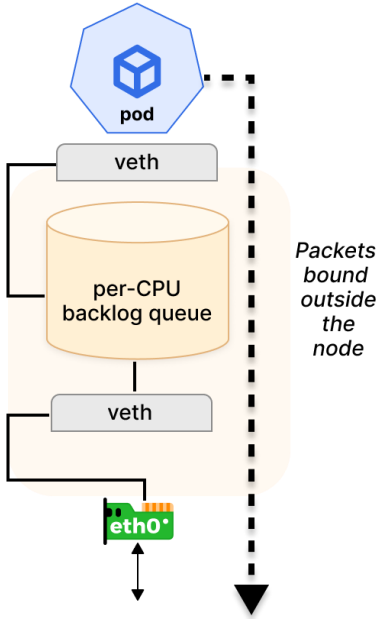
netkit

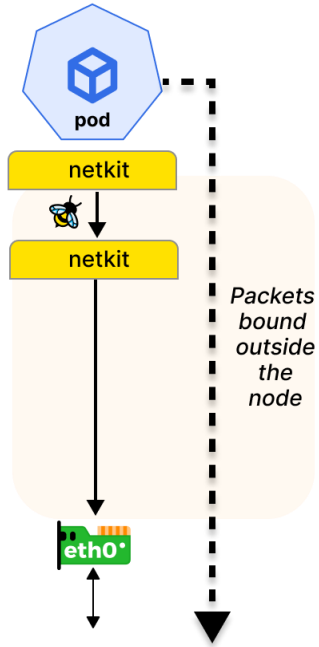
- Most CNIs attach Kubernetes Pod to the node it's hosted on by a virtual ethernet device (veth).
- veth was introduced over 15 years ago
- veth comes with performance penalties

Cilium for innovative networking

netkit

With veth





With netkit

Follow-up

- Watch the eBPF documentary

- Download "Kubernetes Networking for Network Engineer" eBook

- Prepare for the Cilium Certified Associate exam
- Take the Isovalent Labs

Get Ready for the CCA Exam

“The CCNA for Kubernetes”

Core Areas:

- Architecture (20%)
- Network Policy (18%)
- Service Mesh (16%)
- Network Observability (10%)
- Installation and Configuration (10%)
- Cluster Mesh (10%)
- eBPF (10%)
- BGP and External Networking (6%)

Webex App

Questions?

Use the Webex app to chat with the speaker after the session

How

1. Find this session in the Cisco Events mobile app
2. Click “Join the Discussion”
3. Install the Webex app or go directly to the Webex space
4. Enter messages/questions in the Webex space

Webex spaces will be moderated by the speaker until February 28, 2025.

Fill Out Your Session Surveys

Participants who fill out a minimum of 4 session surveys and the overall event survey will get a unique Cisco Live t-shirt.

(from 11:30 on Thursday, while supplies last)

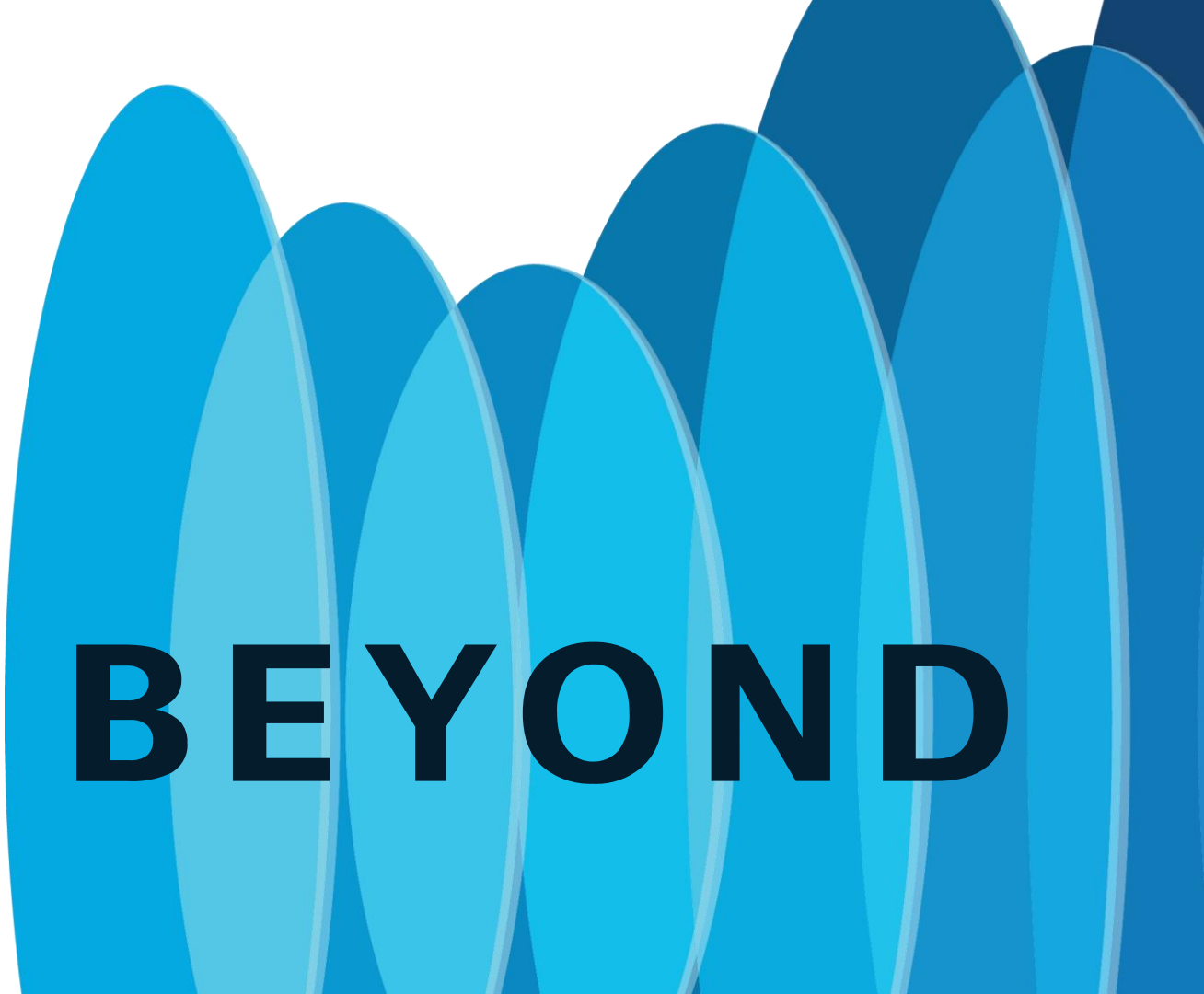
All surveys can be taken in the Cisco Events mobile app or by logging in to the Session Catalog and clicking the ‘Participant Dashboard’

Content Catalog

Thank you

cisco Live!

cisco Live!



GO BEYOND