Designing and Implementing a Microsegmentation strategy with Isovalent Enterprise Platform

Burkey (Anthony Burke)
Chief Technology Officer – APJC Field

Session ID: BRKSEC-3447

Cisco Webex App

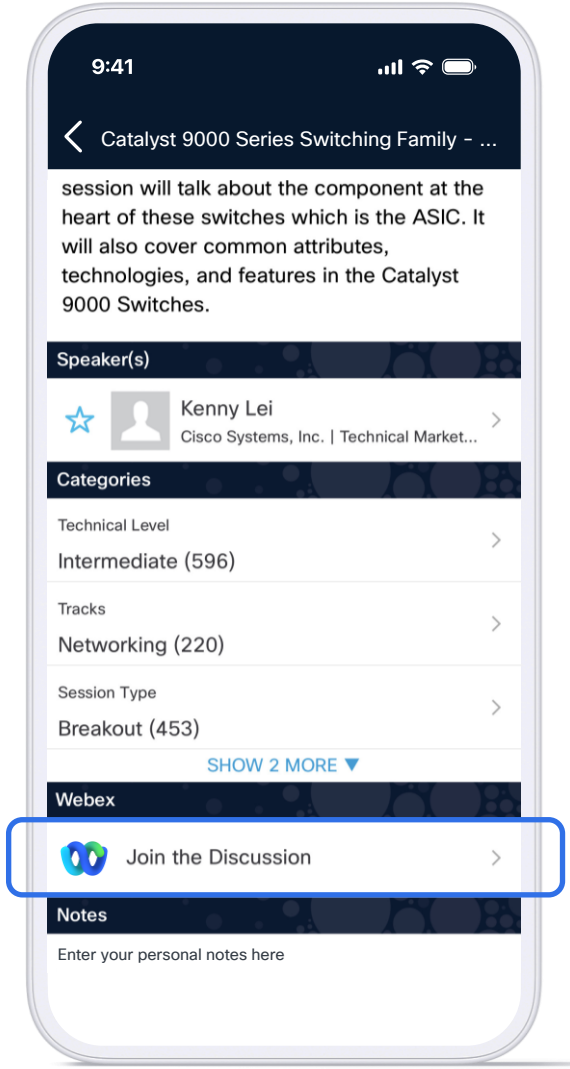
Questions?

Use Cisco Webex App to chat
with the speaker after the session

How

1. Find this session in the Cisco Live 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 14 November 2025.



https://ciscolive.ciscoevents.com/
ciscolivebot/**#BRKSEC-3447**

© 2025 Cisco and/or its affiliates. All rights reserved.

Agenda

01 Introduction

02 What is Isovalent?

03 Core Concepts

04 Design Considerations

05 Operational Consideration

06 Summary

© 2025 Cisco and/or its affiliates. All rights reserved.

Introduction

cisco Live !

Who am I?

- Burkey
- Ultra Marathon Open Water Swimming
- Board Games
- Anything Tolkien or Star Wars

What do I do?

- Chief Technology Officer – APJC Field
- Talk about anything cloud native
- Penchant for networking and security
- eBPF, Cilium, Tetragon, Linux

© 2025 Cisco and/or its affiliates. All rights reserved.

Expectations

What this session is?

- Provide a thorough understanding of enforcement types and options
- Provide guidance around segmentation design principals using Isovalent
- Demonstrate operational visibility with Timescape in provider/consumer models

What this session is not?

- Introduction to Kubernetes
- Exhaustive and comprehensive of every possible design permutation

© 2025 Cisco and/or its affiliates. All rights reserved.

cisco

Expectations

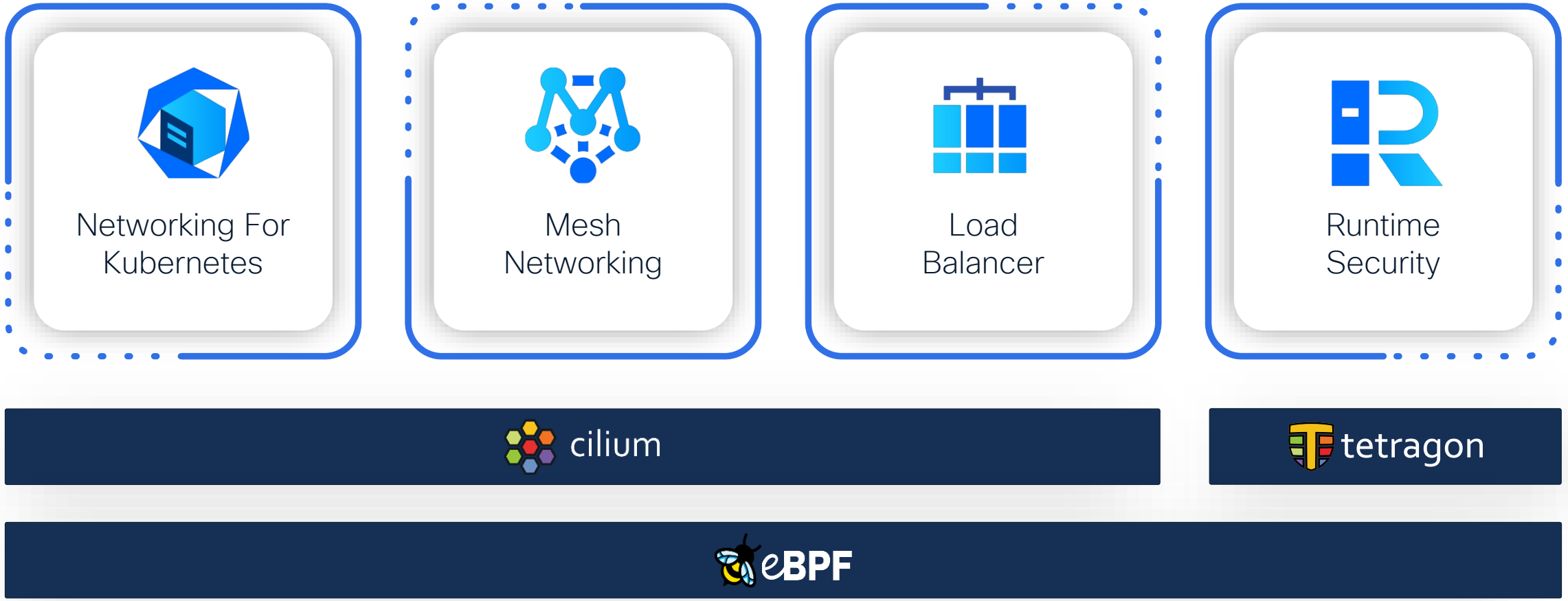
One Size Does Not Fit All

- This session provides guidance and good practice
- Requirements and Constraints will dictate what good looks like for you and your team
- The goal is to provide awareness on design principals coupled with features and functions to give you strong use-cases and tools for segmentation design

© 2025 Cisco and/or its affiliates. All rights reserved.

What is Isovalent?

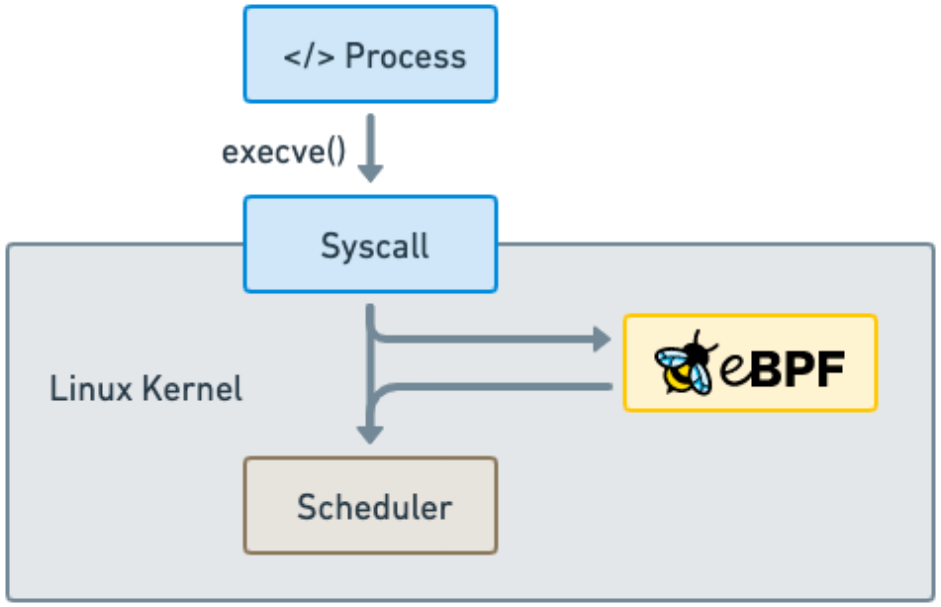
Isovalent Platform



© 2025 Cisco and/or its affiliates. All rights reserved.

eBPF Introduction

Makes the Linux kernel programmable in a secure and efficient way.



“What JavaScript is to the browser, eBPF is to the Linux Kernel”

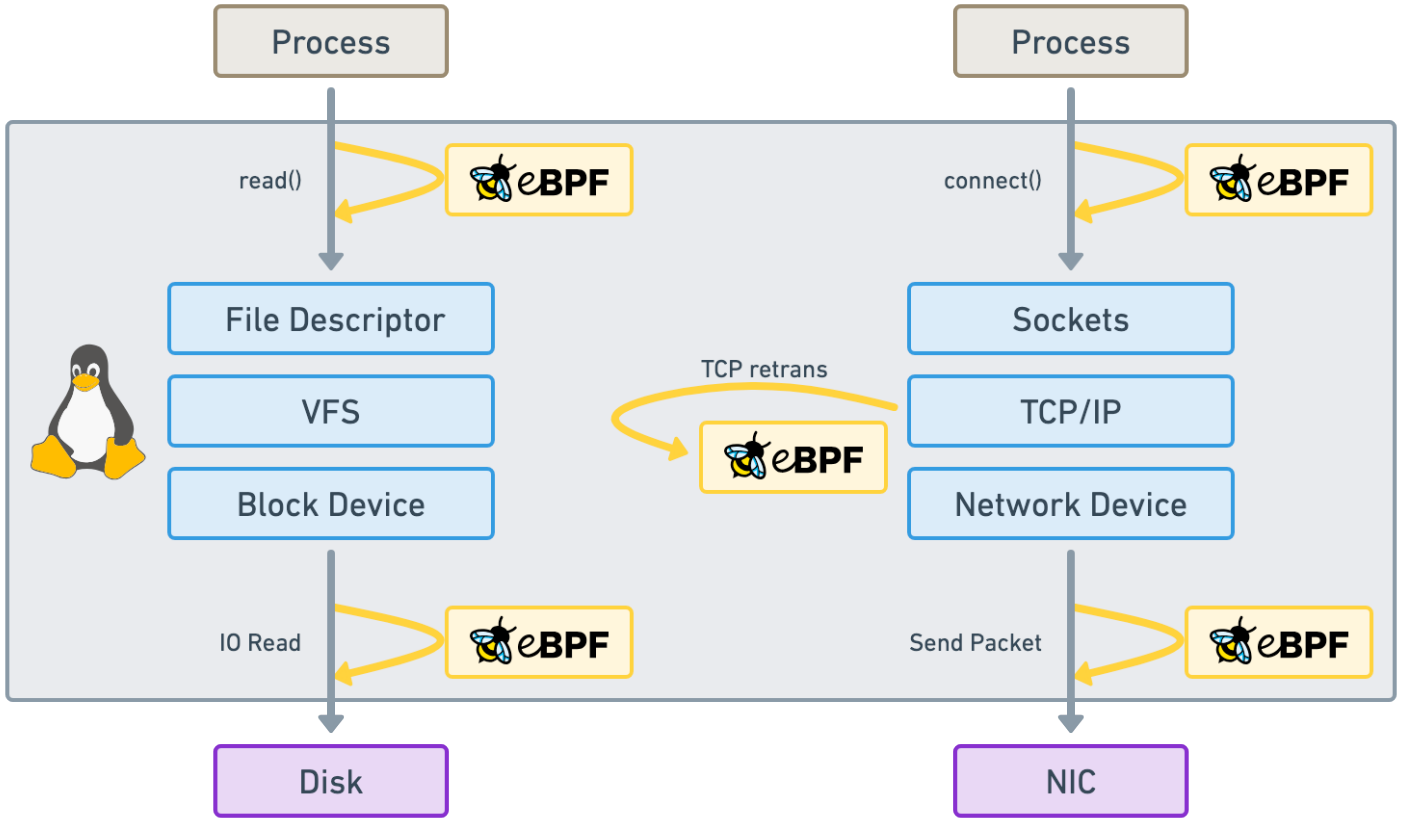
```
int syscall__ret_execve(struct pt_regs *ctx)
{
        struct comm_event event = {
                .pid = bpf_get_current_pid_tgid() >> 32,
                .type = TYPE_RETURN,
        };

        bpf_get_current_comm(&event.comm, sizeof(event.comm));
        comm_events.perf_submit(ctx, &event, sizeof(event));

        return 0;
}
```

© 2025 Cisco and/or its affiliates. All rights reserved.

Run eBPF programs on kernel events



Attachment points

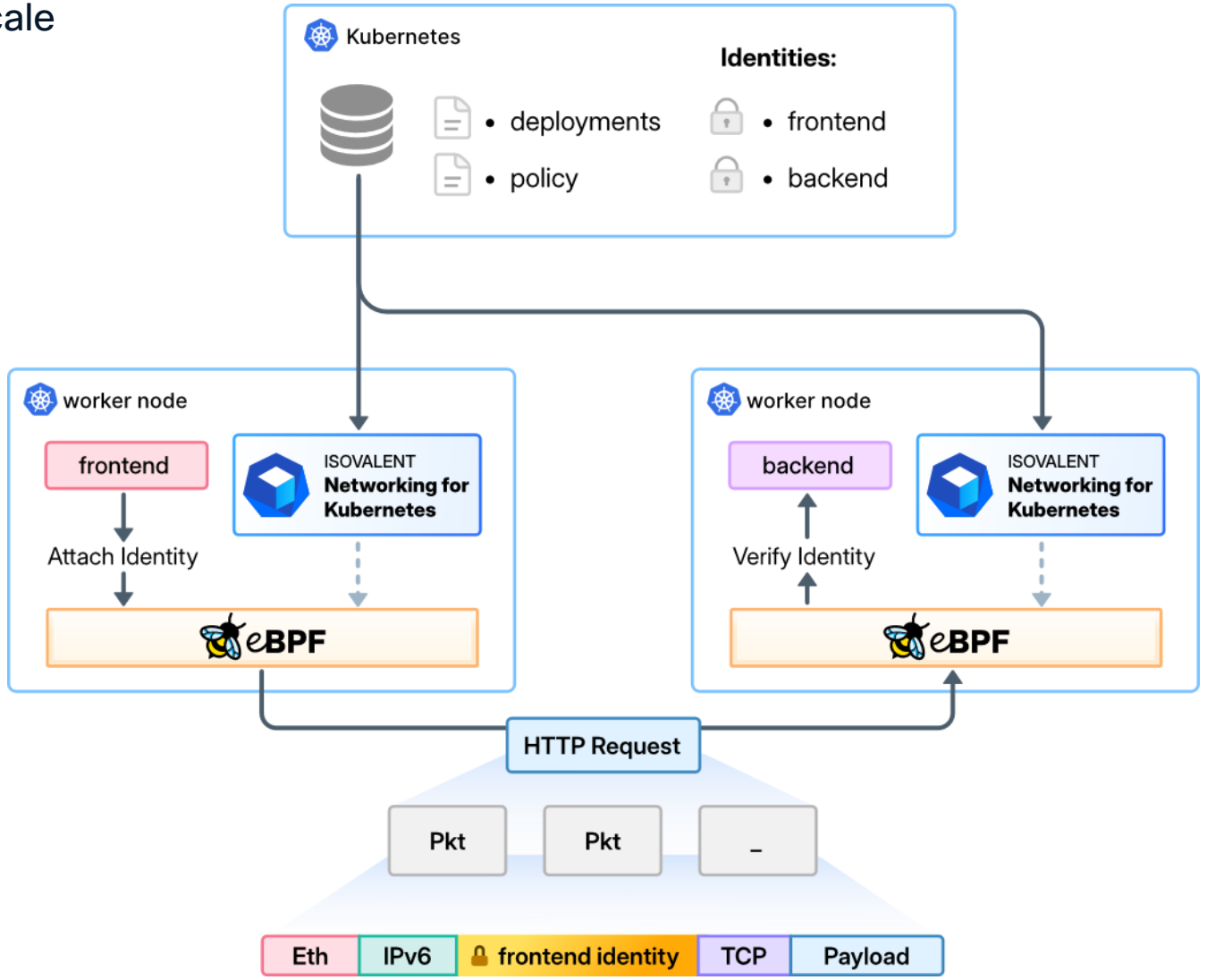
- Kernel functions (kprobes)
- Userspace functions (uprobe)
- System calls
- Tracepoints
- Sockets (data level)
- Network devices (packet level)
- Network device (DMA level) [XDP]
- …

© 2025 Cisco and/or its affiliates. All rights reserved.

Core Concepts

Identity-based Security

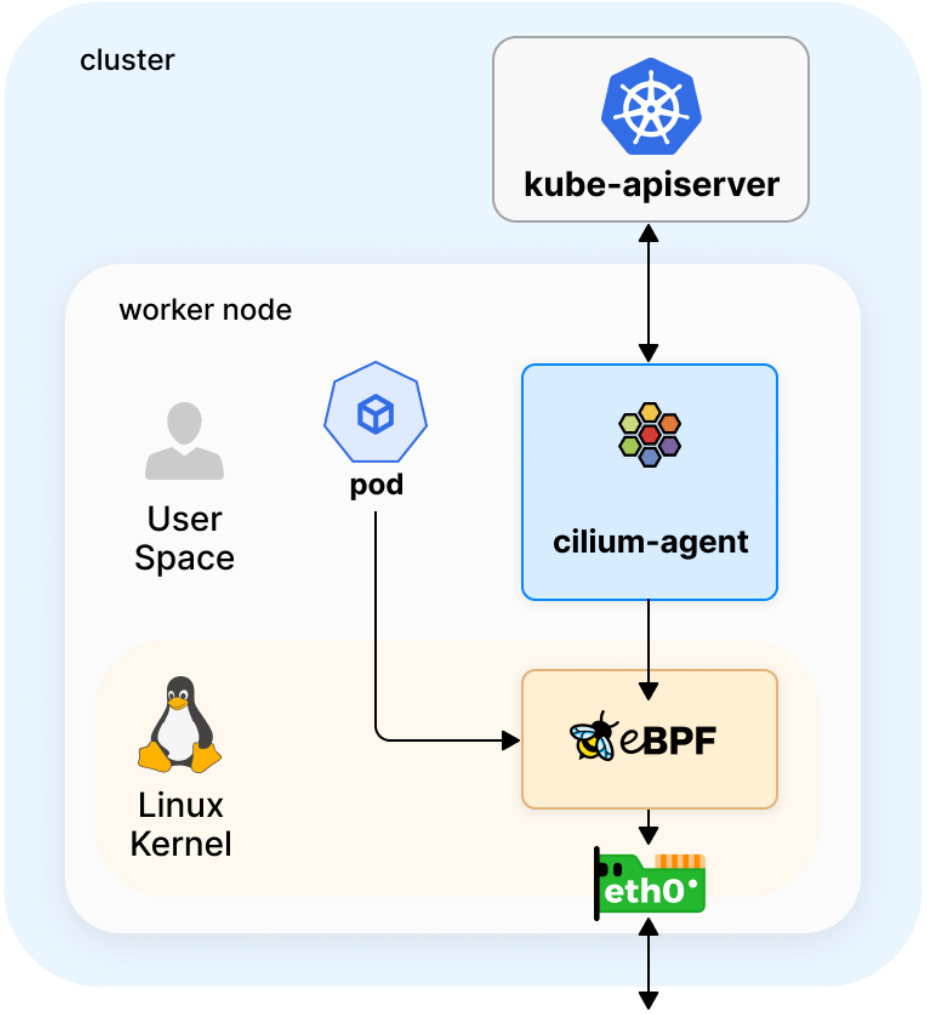
Consistent Security at Scale



© 2025 Cisco and/or its affiliates. All rights reserved.

How Cilium uses eBPF

Networking and Security



Standard K8s resources:

- Deployment / Pod
- Service
- Network Policy

Cilium generated eBPF programs + maps control

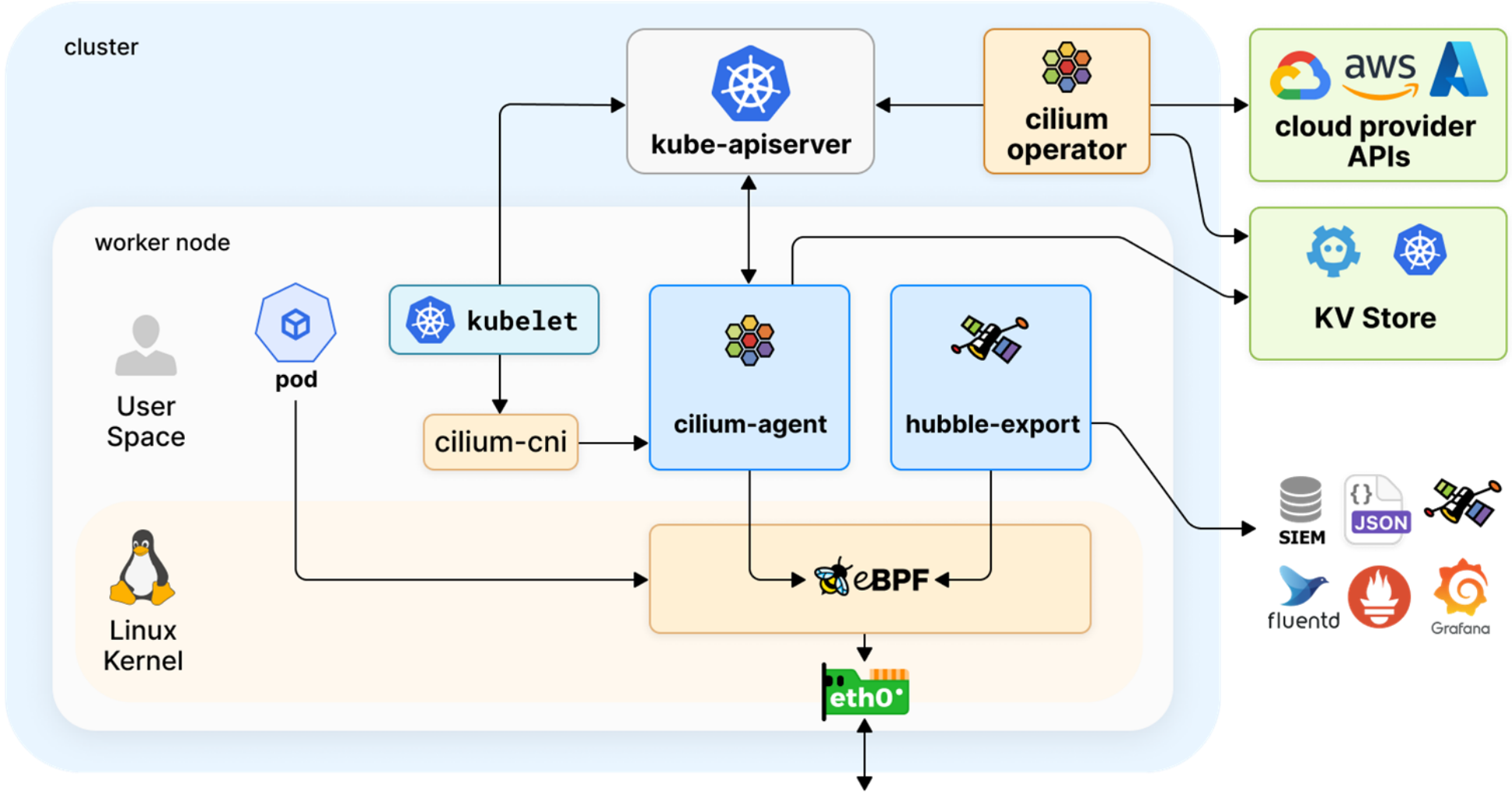
- Pod-to-pod network connectivity
- Service based load balancing
- Network observability
- Security enforcement
- Encryption

© 2025 Cisco and/or its affiliates. All rights reserved.

cisco

Cilium Architecture

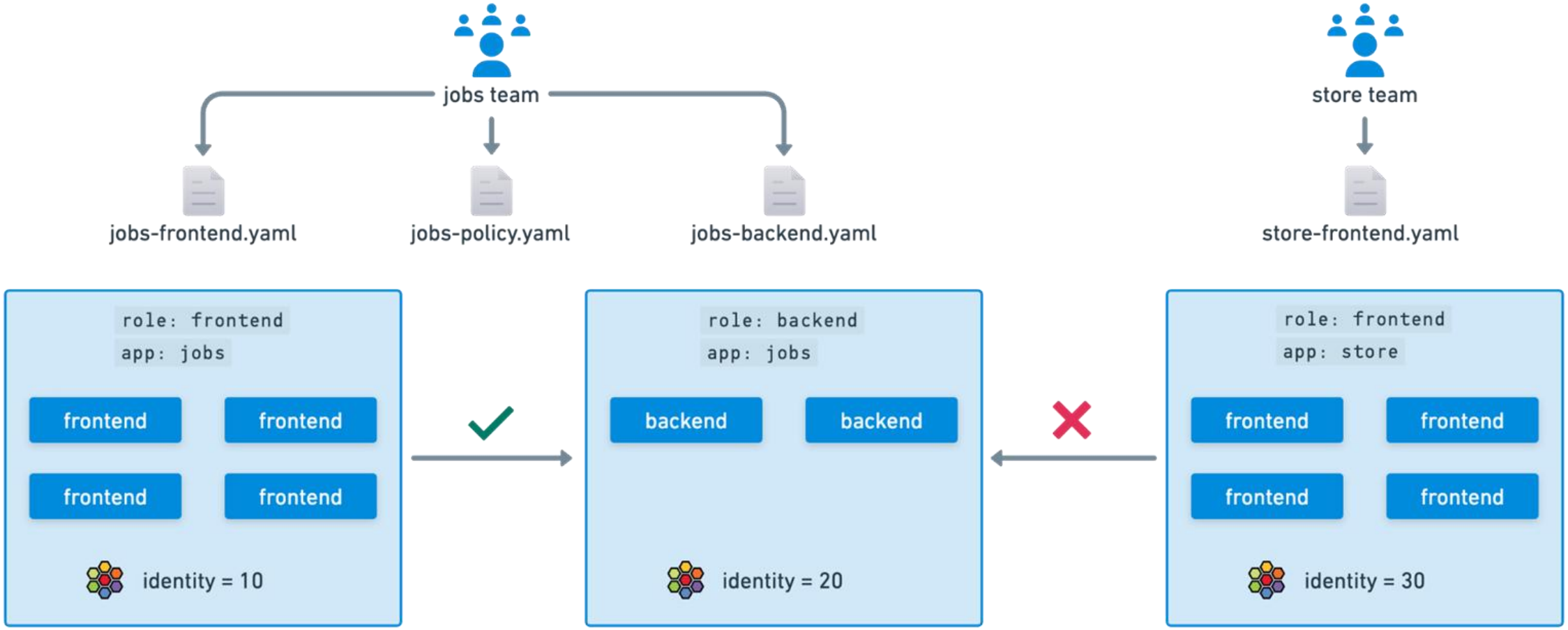
High-level view



© 2025 Cisco and/or its affiliates. All rights reserved.

Identity-based Security

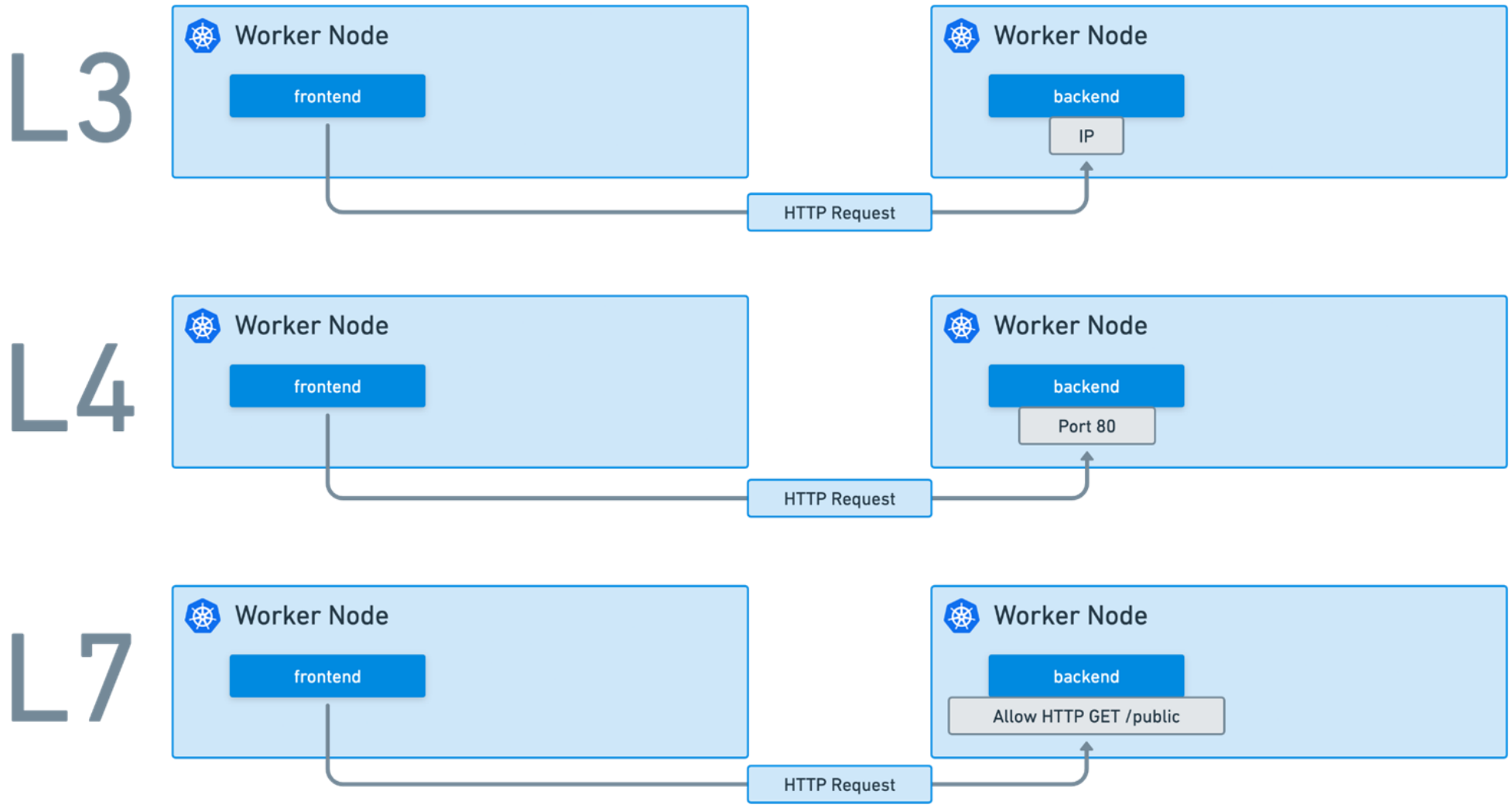
Consistent Security at Scale



© 2025 Cisco and/or its affiliates. All rights reserved.

API-aware Authorization

Layer 4 through to Layer 7



© 2025 Cisco and/or its affiliates. All rights reserved.

cisco

Capability matching

Flexibility across adoption maturity

Kubernetes

- Pod labels
- Namespace name & labels
- ServiceAccount name
- Service names
- Cluster Names

Standard K8s resources:

- FQDN and regular expression

CIDR

- CIDR blocks with exceptions

Cloud Providers

- Instance labels
- VPC/Subnet name/tags
- Security Group name

Logical Entities

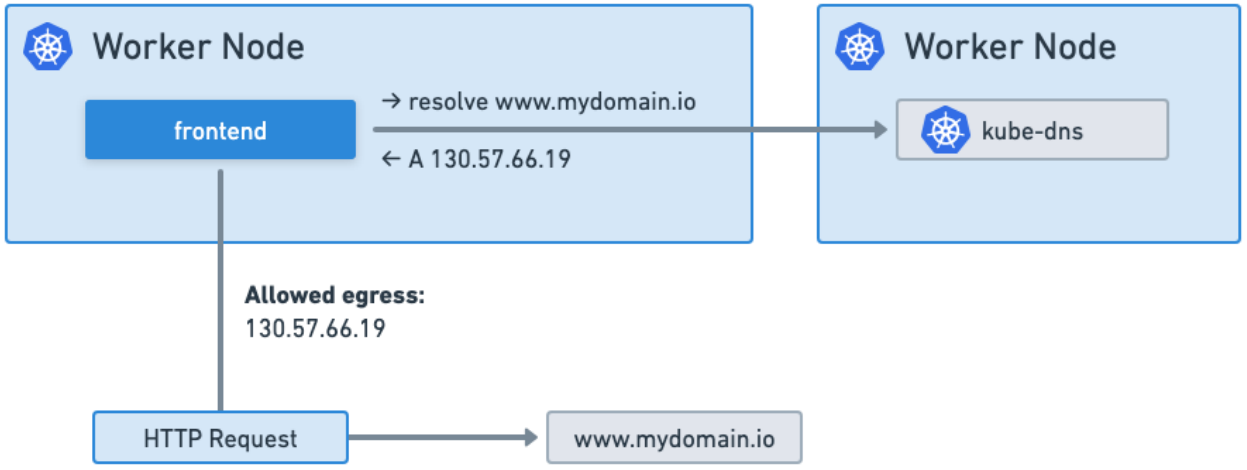
- Everything within a cluster
- Everything outside of a cluster
- Local host
- K8s API Server
- And more

© 2025 Cisco and/or its affiliates. All rights reserved.

cisco

DNS-aware Cilium Network Policies

Cilium DNS proxy

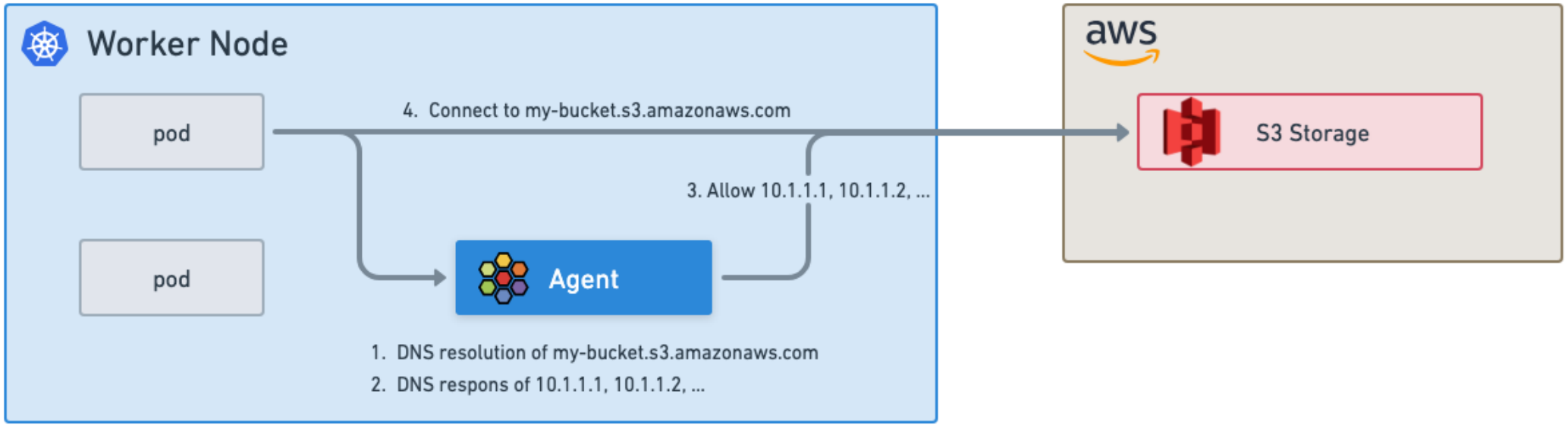



```yaml
apiVersion: "cilium.io/v2"
kind: CiliumNetworkPolicy
metadata:
  name: "frontend-egress-dns-fqdn"
spec:
  endpointSelector:
    matchLabels:
      app: frontend
  egress:
  # 1. Allow DNS queries to kube-dns
  - toEndpoints:
      - matchLabels:
          k8s-app: kube-dns
          io.kubernetes.pod.namespace: kube-system
    toPorts:
      - ports:
          - port: "53"
            protocol: UDP
          - port: "53"
            protocol: TCP
        rules:
          dns:
            - matchPattern: "*"   # Enables DNS visibility
  # 2. Allow egress to the resolved IPs of this FQDN
  - toFQDNs:
      - matchName: "www.mydomain.io"
```

© 2025 Cisco and/or its affiliates. All rights reserved.

Cilium DNS Proxy

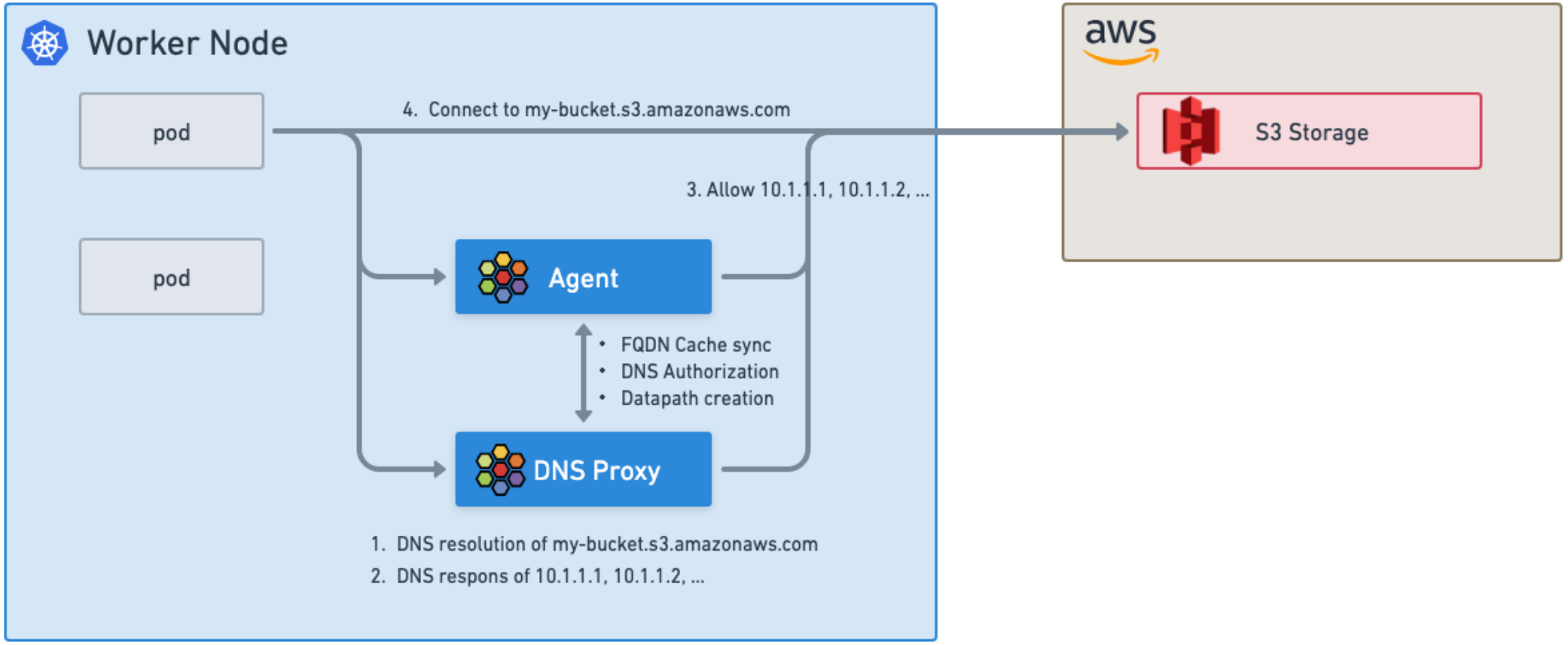
Standalone



© 2025 Cisco and/or its affiliates. All rights reserved.

Cilium DNS Proxy

High Availability

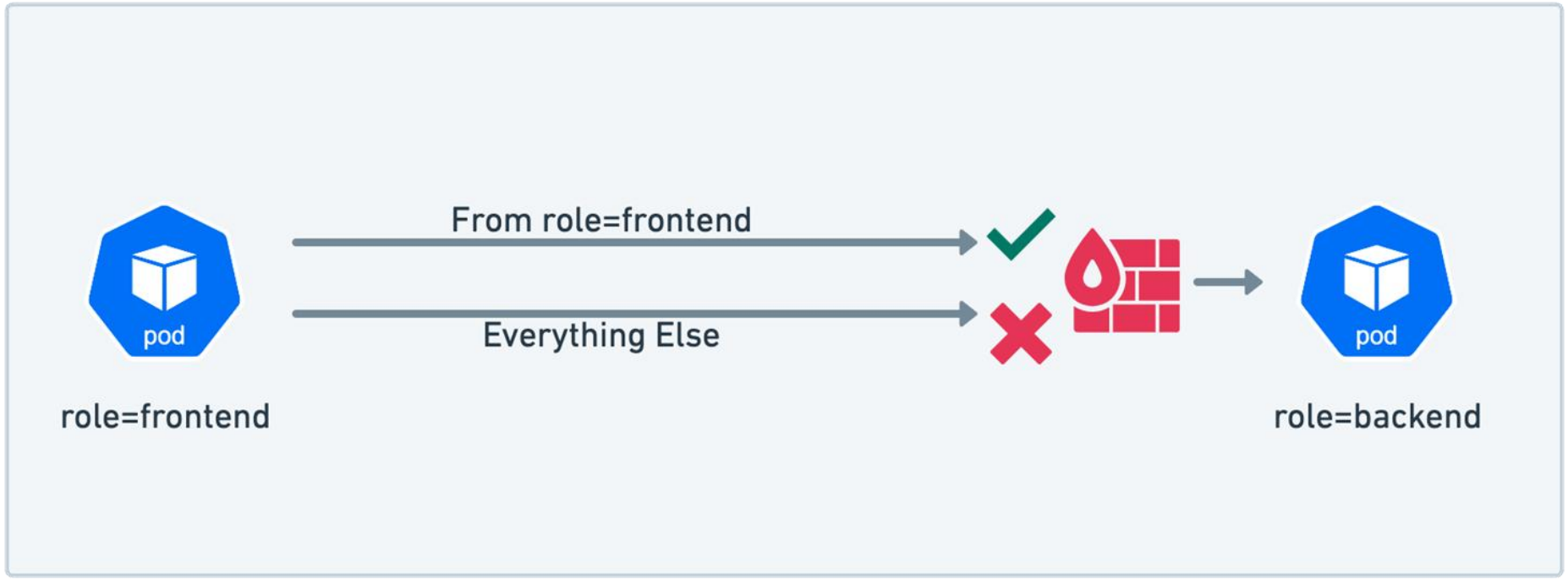


© 2025 Cisco and/or its affiliates. All rights reserved.

cisco

What is a Network Policy?

“A declarative pod-level network firewall abstractions that understands Kubernetes identity”



© 2025 Cisco and/or its affiliates. All rights reserved.

Core Concepts

Declarative Policy

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

- Allowed Connectivity is defined by using declarative Kubernetes resources
  - NetworkPolicy: built-in Kubernetes abstraction for basic policies
  - CiliumNetworkPolicy: Custom Resource Definition (CRD) that is a super-set of NetworkPolicy.
    - Includes advanced features
  - Each network policy resources defines one or more “rules”

© 2025 Cisco and/or its affiliates. All rights reserved.

cisco

Core Concepts

Endpoint Selection

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

- Each Policy exists in a namespace, except for cluster-wide network policies
- Endpoint selectors match pods within that namespace, or. Cluster-wide
- A policy matches zero or more pods in a namespace based on a label match
  - PodSelector field in NetworkPolicy
  - EndpointSelector in CiliumNetworkPolicy
- Subnet Match:
  - Pod must have labels specific in ‘matchLabels’ but may have other labels as well
- Each rule is applied to all pods that match

© 2025 Cisco and/or its affiliates. All rights reserved.

Core Concepts

Ingress & Egress Rules

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

- Ingress Rules: traffic entering a pod
- Egress Rules: traffic leaving a pod
- Policies may have any combination of ingress & egress rules

© 2025 Cisco and/or its affiliates. All rights reserved.

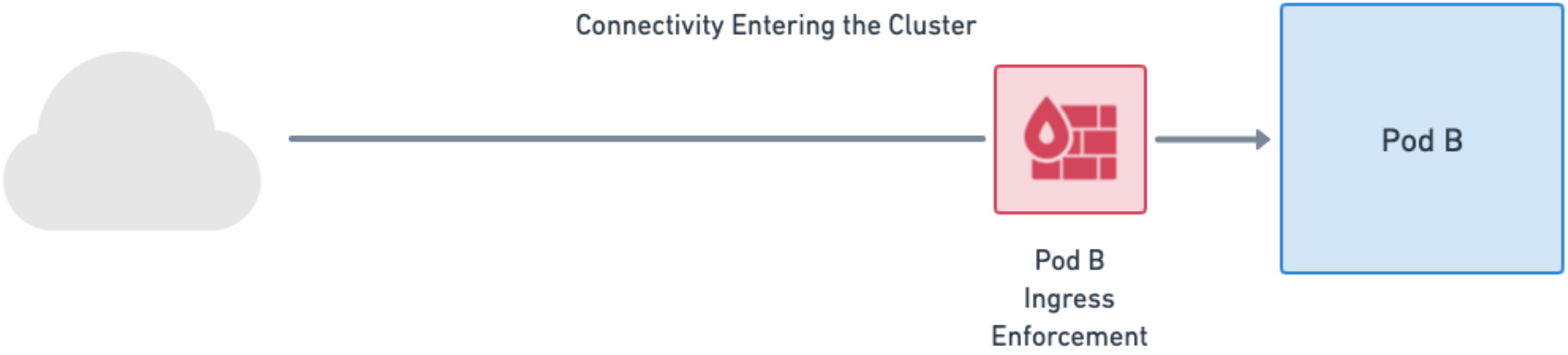
Core Concepts

Ingress & Egress Rules

© 2025 Cisco and/or its affiliates. All rights reserved.

Core Concepts

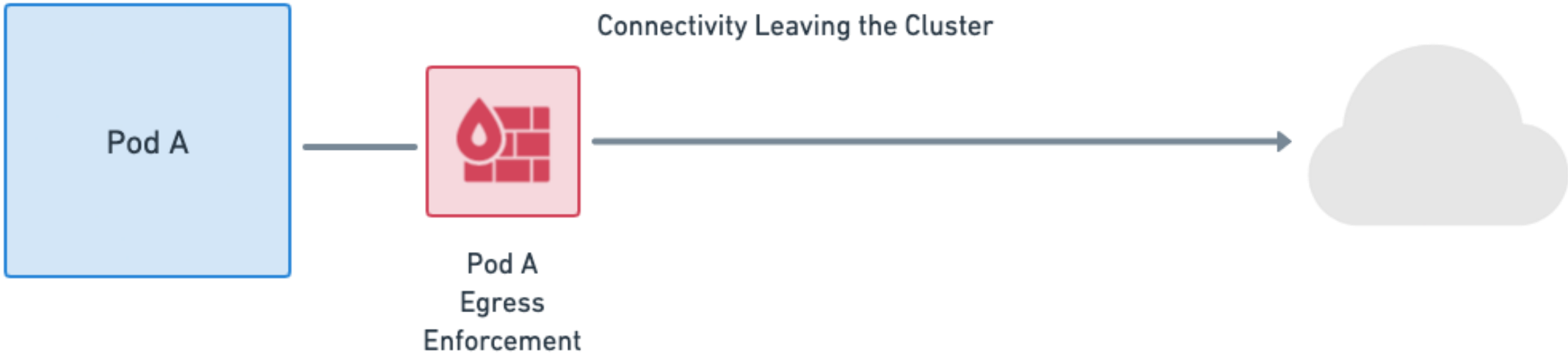
Ingress & Egress Rules



© 2025 Cisco and/or its affiliates. All rights reserved.

Core Concepts

Ingress & Egress Rules



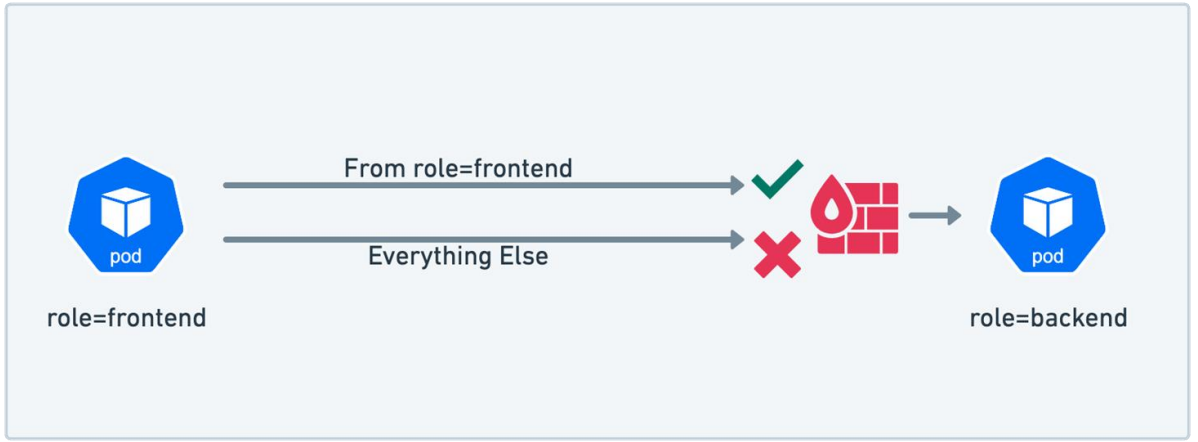
© 2025 Cisco and/or its affiliates. All rights reserved.

Core Concepts

Default Deny Model

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

- Each rule describes an allowed type of connectivity
- Total allowed connectivity is the logical OR of all rules applied to a pod
- Connectivity not explicitly allowed implicitly denied



© 2025 Cisco and/or its affiliates. All rights reserved.

Core Concepts

Automatically Allowed Connections

If a NetworkPolicy is default deny, why is traffic forwarded in a cluster with no policies defined?

- Default deny model does NOT apply:
  - On ingress for a pod if no ingress rules match the pod
  - On egress for a pod if no egress rules matches the pod

- All ingress connectivity from Kubernetes worker nodes is automatically allowed
  - Avoid unintentional impact on network liveness/readiness probes from kubelet
  - Allows connectivity from host-network pods. Host network == trusted

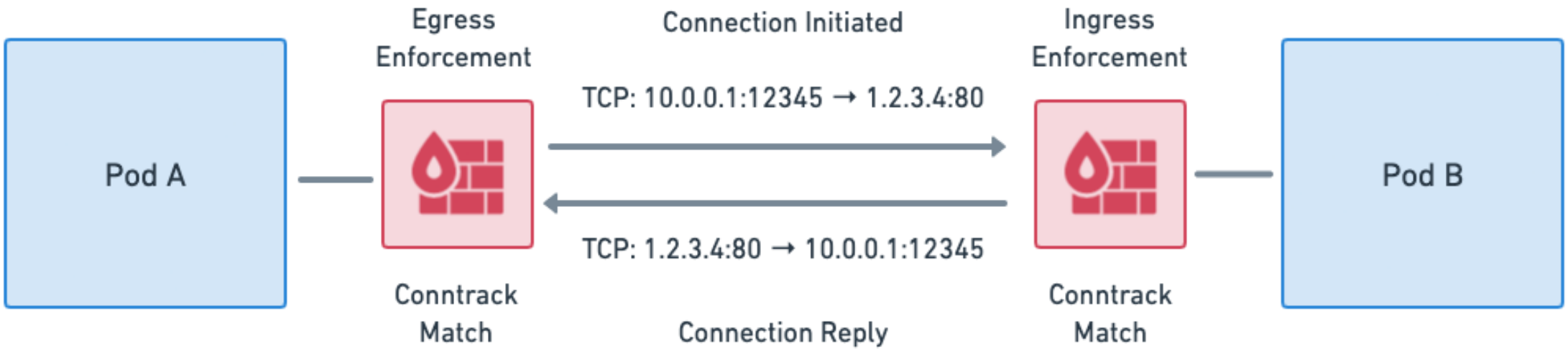
© 2025 Cisco and/or its affiliates. All rights reserved.

cisco

Core Concepts

Stateful Connection Tracking

Connection reply traffic is automatically allowed



© 2025 Cisco and/or its affiliates. All rights reserved.

Core Concepts

Policy Layers

Layer 3

- toEndpoints / fromEndpoints
- toCIDR / fromCIDR
- toCIDRset / fromCIDRset
- toFQDN / fromFQDN

Layer 4

- toPort / fromPort

Layer 7

- DNS
- HTTP, gRPC
- Kakfa, Memcached, Cassandra

© 2025 Cisco and/or its affiliates. All rights reserved.

cisco

Core Concepts

Policy Layers

Within a rule, each layer is logically ANDed with other layers

Unspecified layers are wildcards

```
ingress:
  - fromEndpoints:
    - matchLabels:
      role: frontend
```

```
ingress:
- toPorts:
  - ports:
      - port: "80"
        protocol: TCP
```

```
ingress:
- fromEndpoints:
  - matchLabels:
     role: frontend
  toPorts:
  - ports:
      - port: "80"
        protocol: TCP
```

© 2025 Cisco and/or its affiliates. All rights reserved.

cisco

Policy Definition

Multiple rules per policy

2 Rules (logical OR)

```
ingress:
- fromEndpoints:
  - matchLabels:
      role: frontend
- toPorts:
  - ports:
    - port: "80"
      protocol: TCP
```

1 Rules (logical AND)

```
ingress:
- fromEndpoints:
  - matchLabels:
      role: frontend
  toPorts:
  - ports:
    - port: "80"
      protocol: TCP
```

© 2025 Cisco and/or its affiliates. All rights reserved.

cisco

Policy Definition

Multiple rules per policy

Multi-label match

Pod must have all labels to match (logical AND)

```
spec:
  endpointSelector:
    matchLabels:
      org: empire
      class: deathstar
  egress:
...
```

Wildcards

Matches all endpoints

```
spec:
  endpointSelector: {}
  egress:
...
```

Set-based Label Selectors

More complex, more powerful matching

```
spec:
  endpointSelector:
    matchExpressions:
    - key: k8s-app
      Operator: NotIn
      Values:
      - app1
      - app2
  egress:
...
```

© 2025 Cisco and/or its affiliates. All rights reserved.

cisco

Policy Definition

toEndpoints / from Endpoints

Subset match: pod must have all labels specified in ‘matchLabels’ to be allowed by rule, but may have additional labels as well

Namespaces are the key consideration when matching use labels

Implicit Local Namespace Binding

```
...
  ingress:
  - fromEndpoints:
    - matchLabels:
        role: frontend
```

Explicit Namespace Binding

```
...
  ingress:
  - fromEndpoints:
    - matchLabels:
        role: frontend
        k8s:io.kubernetes.pod.namespace: foo
```

© 2025 Cisco and/or its affiliates. All rights reserved.

cisco

Policy Definition

toEndpoints / from Endpoints

Match all the endpoints in a local namespace

```
...
  ingress:
  - fromEndpoints:
    - matchLabels: {}
```

Wildcarding Namespace

```
...
 ingress:
 - fromEndpoints:
   - matchExpressions:
     - key: k8s:io.kubernetes.pod.namespace
       operator: Exists
     - key: app
       operator: In
       values:
       - backend
```

© 2025 Cisco and/or its affiliates. All rights reserved.

cisco

Policy Definition

toCIDR/fromCIDR and toCIDRset/fromCIDRset

NOTE: CIDR and CIDR-based policies apply ONLY to Ips that are “outside” the cluster. Do not use these to try and match on a Pod, Service, or Worker Node IP

```
...
  egress:
  - toCIDR:
    - 10.0.0.0/8
```

```
...
  egress:
  - toCIDRSet:
    - cidr: 0.0.0.0/0
      except:
      - 169.254.169.254/32
```

© 2025 Cisco and/or its affiliates. All rights reserved.

cisco

Policy Definition

toFQDN Layer 3 policies

Similar to toCIDR but translates from DNS names

```
...
  egress:
  - toFQDNs:
    - matchName: "www.domain.net"
```

Allows regex-based matching

```
...
  egress:
  - toFQDNs:
    - matchPattern: "*.company.com"
```

NOTE: Try to avoid using overlapping toFQDNs – especially in multiple policies

© 2025 Cisco and/or its affiliates. All rights reserved.

cisco

Policy Definition

toFQDNs DNS proxy requirement

```
...
   - toEndpoints:
      - matchLabels:
          io.kubernetes.pod.namespace: kube-system
          k8s-app: kube-dns
      toPorts:
        - ports:
           - port: "53"
             protocol: ANY
          rules:
            dns:
              - matchPattern: "*"
```

- toFQDNs rules can only be applied to the pods that also have a DNS proxy rule that intercepts all DNS lookups from that pod
- For security reasons, this rule should be limited to DNS servers that you trust (cluster internal e.g core-dns)

© 2025 Cisco and/or its affiliates. All rights reserved.

Policy Definition

toEntities/fromEntities + Reserved Layer 3 identities

```
...
  ingress:
  - fromEntities:
    - world
```

```
...
  egress:
  - toEntities:
    - cluster
    - host
    - remote-node
```

Special reserved identities that correspond to a group of endpoints or notable entities

- **world**: all endpoints outside of the cluster
- **cluster**: all managed pod endpoints
- **host**: the local Kubernetes worker nodes
- **remote-node**: non-local Kubernetes work nodes
- **kube-apiserver**: remove nodes which have healthy backends

© 2025 Cisco and/or its affiliates. All rights reserved.

cisco

Policy Definition

toPorts

Layer 4 only policies that implicitly allow all from Layer 3 entities

```
...
  ingress:
  - toPorts:
     - ports:
         - port: "80"
           protocol: ANY
```

Layer 3 + Layer 4 policies allow only connections that match at both layers

```
...
  egress:
  - toPorts:
     - ports:
         - port: "80"
           protocol: ANY
    fromEndpoints:
      - matchLabels: {}
```

© 2025 Cisco and/or its affiliates. All rights reserved.

cisco

Policy Definition

Port Range support in Network Policies

```yaml
apiVersion: "cilium.io/v2"
kind: CiliumNetworkPolicy
metadata:
  name: "range-port-rule"
spec:
  description: "L3-L4 policy to restrict range of ports"
  endpointSelector:
    matchLabels:
      app: server
  ingress:
  - fromEndpoints:
    - matchLabels:
        app: client
    toPorts:
    - ports:
      - port: "8080"
        endPort: 8082
        protocol: TCP
```

- Allows the use of port ranges in a CiliumNetworkPolicy and a CiliumClusterNetworkPolicy
- Previously ports needed to be listed one by one even if they were contiguous.
- Cilium 1.16 added the endPort value to aid in achieving this.

© 2025 Cisco and/or its affiliates. All rights reserved.

cisco

Policy Definition

Layer 7 Policy Example (HTTP)

Layer 4 + Layer 7 policies implicitly allow traffic from Layer 3 entities

```
...
  ingress:
  - toPorts:
      - ports:
          - port: "80"
            protocol: ANY
```

Layer 3+ 4 +7 policies allow only connections that match at all layers

```
...
  egress:
  - toPorts:
      - ports:
          - port: "80"
            protocol: ANY
    fromEndpoints:
      - matchLabels: {}
```

© 2025 Cisco and/or its affiliates. All rights reserved.

cisco

Common Patterns

Default Deny Policy

```yaml
apiVersion: "cilium.io/v2"
kind: CiliumNetworkPolicy
metadata:
  name: deny-all
  namespace: my-namespace-xy
spec:
  description: Block everything
  endpointSelector: {}
  ingress:
  - {}
  egress:
  - {}
```

- Select all endpoints in a namespace, specifying only an “empty” rule for ingress & egress
- Subverts the “automatic allow” by ensuring all endpoints have at least one ingress & egress rule but doesn’t permit any traffic
- Can be installed by default in a namespace, requiring the app tam to add explicit allow exceptions

Careful: This example will even deny namespace internal and traffic towards core-dns/Kubernetes API

© 2025 Cisco and/or its affiliates. All rights reserved.

cisco

Common Patterns

Default Deny Policy – Allow within a namespace

```yaml
apiVersion: "cilium.io/v2"
kind: CiliumNetworkPolicy
metadata:
  name: allow-within-namespace
  namespace: my-namespace-xy
spec:
  description: Allow NS internal traffic, block everything else
  endpointSelector: {}
  ingress:
  - fromEndpoints:
    - {}
  egress:
  - toEndpoints:
    - {}
```

Careful: This example will deny traffic towards core-dns/Kubernetes API

© 2025 Cisco and/or its affiliates. All rights reserved.

cisco

Common Patterns

Default DNS Proxy Rule

Before applying a “default deny” it is recommended to apply a DNS proxy rule per namespace or as a CiliumClusterwideNetworkPolicy to allow and observe all DNS lookups

```
...
   - toEndpoints:
      - matchLabels:
          io.kubernetes.pod.namespace: kube-system
          k8s-app: kube-dns
      toPorts:
        - ports:
           - port: "53"
             protocol: ANY
          rules:
            dns:
              - matchPattern: "*"
```

© 2025 Cisco and/or its affiliates. All rights reserved.

cisco

Common Patterns

Layer 7 Visibility Policies

```yaml
apiVersion: cilium.io/v2
kind: CiliumNetworkPolicy
metadata:
  name: http-ingress-visibility
spec:
  endpointSelector:
    matchLabels: {}
  ingress:
    - fromEntities:
        - all
    - toPorts:
        - ports:
            - port: "80"
              protocol: "TCP"
          rules:
            http:
              - {}
```

- Provides visibility into API-layer requests even if no policy is being enforced
- For Layer 7 policy, an “empty rule” matches any request resulting in wildcard behaviour
- Visibility policies can be ingress, egress, or both ingress and egress
- **NOTE:** Will also need to have a rule to allow traffic on other ports

© 2025 Cisco and/or its affiliates. All rights reserved.

cisco

Common Policy “Gotchas”

Some good ones to remember

- Enabling egress enforcement rules but not allow DNS connectivity
- Creating toFQDN rules without corresponding DNS proxy rule
- Creating an FQDN policy for internal Kubernetes Services
- Creating rules based on the label-identity of unmanaged pods
- Creating rules to allow connections to pods based on IP/CIDR
  - Use toEndpoints instead
- Creating rules to allow connections to Kubernetes workers based on Ips
  - Use toEntities with host and remote-node instead

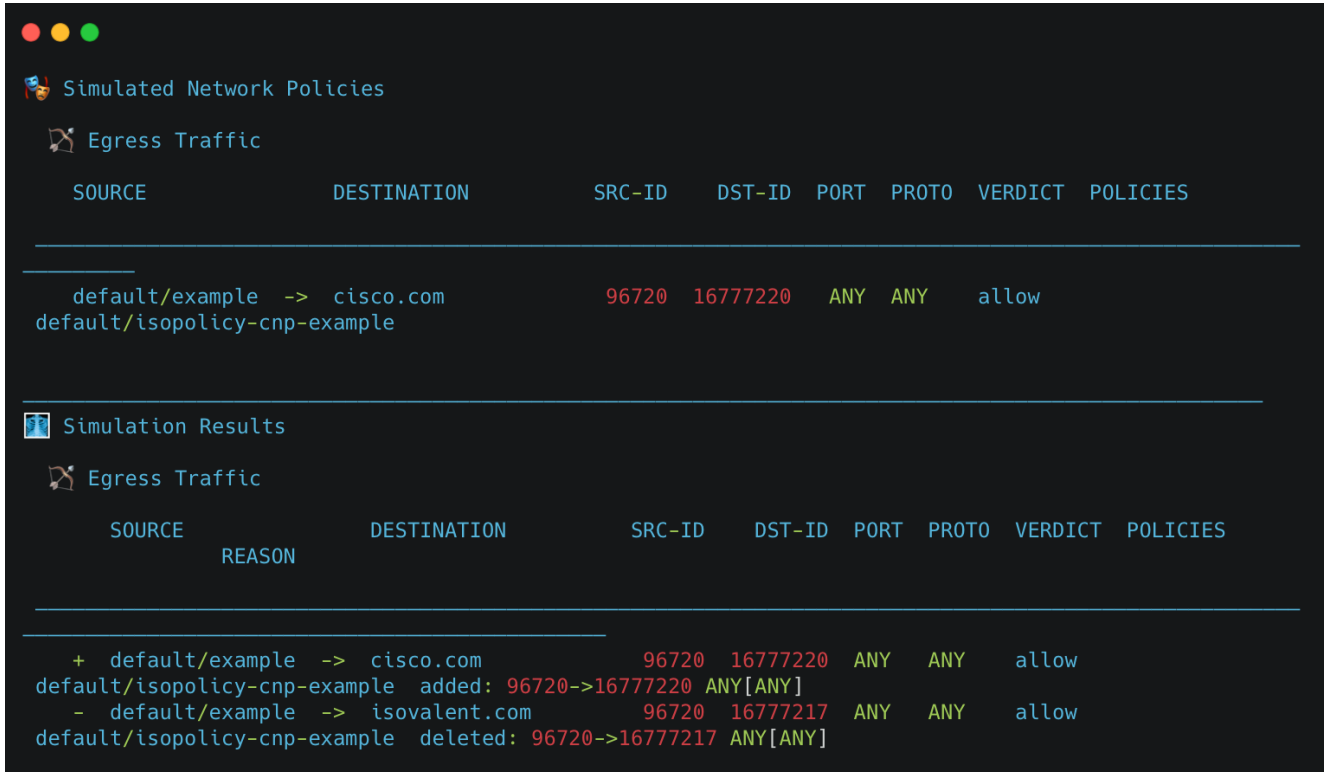
© 2025 Cisco and/or its affiliates. All rights reserved.

cisco

Isopolicy

Policy simulation and recommendation tooling

- Simplification of policy validation and management
- Static analysis mode
- Live or Historical data from Timescape
- Simulation of outcome for safe “what-if” scenarios



© 2025 Cisco and/or its affiliates. All rights reserved.

Isopolicy

```
./isopolicy simulate --include-ns default --sim-dir proposed/ --range 15m

🎭 Loading simulations...

  ◦ 🚀 default/isopolicy-cnp-example updated

────────────────────────────────────────────────────────────────────────────────────────────────────

🌊 Evaluating flows...

  > 117s   × L3_L4 [TCP] [32914] default/example (65323)                          → [  443]
isovalent.com (2)                                        (                    )
  < 117s   ✓ L3_L4 [TCP] [32914] default/example (65323)                          → [  443] cisco.com
(16777218)                                (                    )  [default/isopolicy-cnp-example]


🔥 Verdict changed for 1 flow(s). Examined (8) flow(s).

────────────────────────────────────────────────────────────────────────────────────────────────────

📊 Simulation Results

  🏹 Egress Traffic

       SOURCE                DESTINATION          SRC-ID    DST-ID  PORT  PROTO  VERDICT  POLICIES
                REASON

 ──────────────────────────────────────────────────────────────────────────────────────────────────
──────────────────────────────────────────────
    +  default/example  ->  cisco.com             65323  16777218  443   TCP    allow
 default/isopolicy-cnp-example  added: 65323->16777218 443[TCP]
    -  default/example  ->  isovalent.com         65323  16777217  443   TCP    allow
 default/isopolicy-cnp-example  deleted: 65323->16777217 443[TCP]
       default/example  ->  kube-system/coredns   65323     51148  53    UDP    allow    default/dns-
visibility
```

© 2025 Cisco and/or its affiliates. All rights reserved.

Design Considerations

Key questions when adopting segmentation

Discussions that should be had prior to design

- Where should you start?
- What policies reduce risk with the least amount of friction?
- How to troubleshoot your network policies?
- How to keep policies up to date as applications evolve?
- How to prevent application teams from simply allowing everything?
- How can security teams attest a baseline set of policies are enforced?
- How can security teams be alerted to denied connections?

© 2025 Cisco and/or its affiliates. All rights reserved.

There is no shortcuts to adopting Segmentation

Default Deny vs Risk Reduction

Default Deny Approach

- Each service-to-service communication must explicitly be allowed
- High risk of misconfiguration when enumerating across all valid connectivity profiles
- High adoption friction – application team vs platform team vs security

Risk Reduction

- Define metrics for Risk Exposure
- Focus on most sensitive workloads / namespaces first
- Leverage network observability to identify policy patterns
- Leverage observability and policy recommendation tooling within Isovalent to iterate and segment highest risk to lowest risk workloads

© 2025 Cisco and/or its affiliates. All rights reserved.

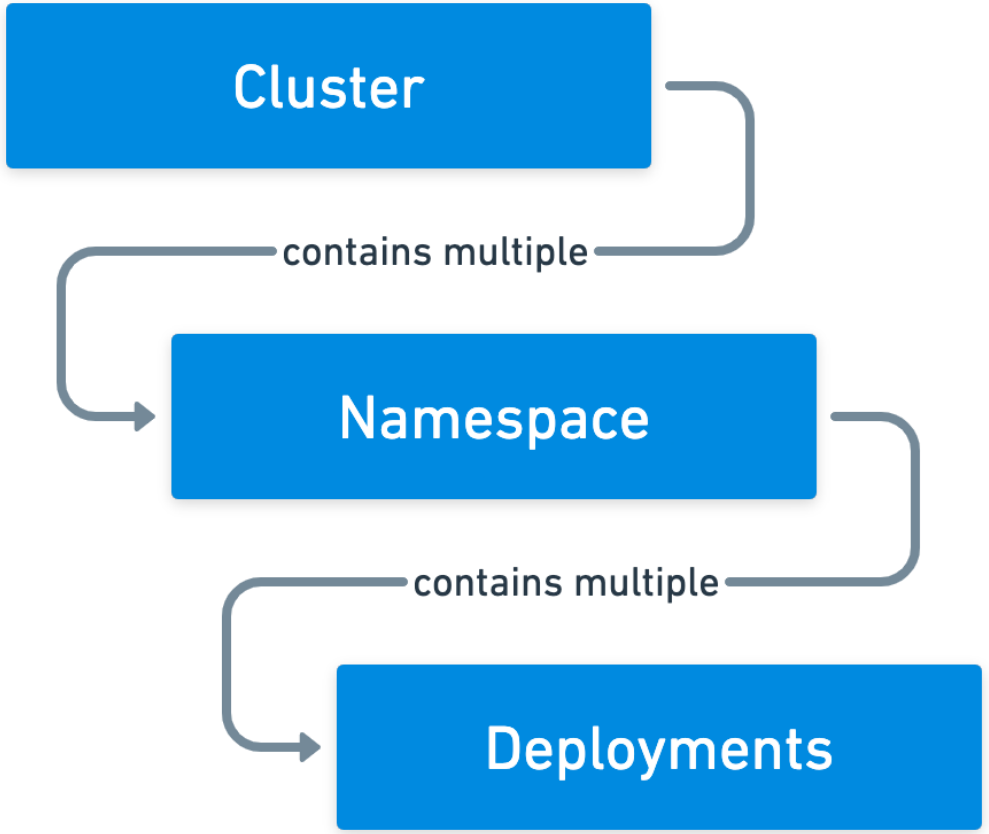
cisco

Application Teams and Multi-tenancy

Security Boundaries

Kubernetes Tenancy

- A Kubernetes cluster is often used by multiple teams with each managing one or more services
- Each team has one or more namespaces to run their applications
- Applying network policy at namespace level is a common first step for multi-tenancy
- Traditional mentality around zone segmentation policy can be used for course enforcement

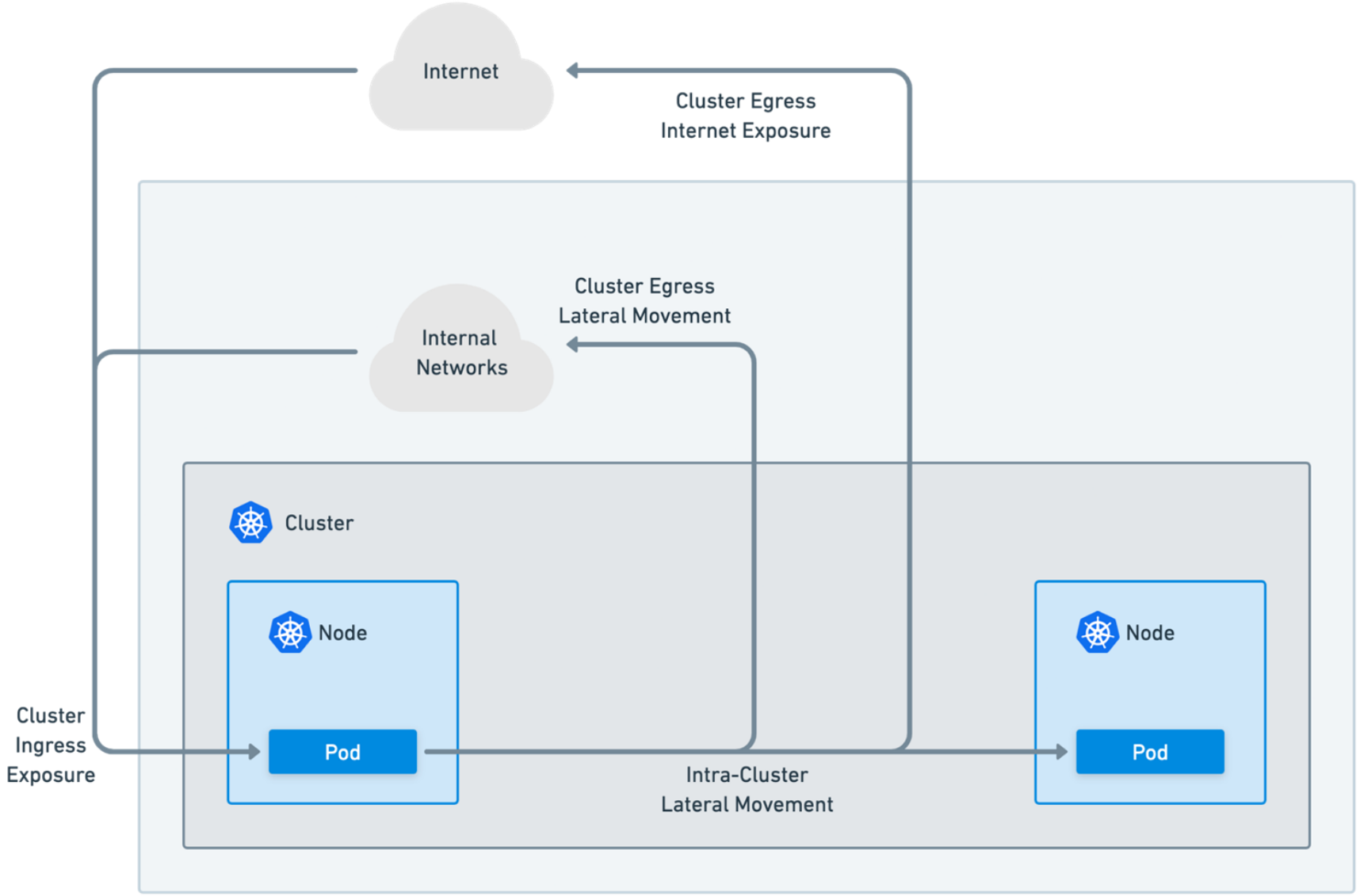


NOTE: Not always 1:1 team to namespace. Teams may be a namespace label or pod label.

© 2025 Cisco and/or its affiliates. All rights reserved.

Kubernetes Network Risk Exposure

Exposure Types



© 2025 Cisco and/or its affiliates. All rights reserved.

Application Teams and Multi-tenancy

Exposure Types

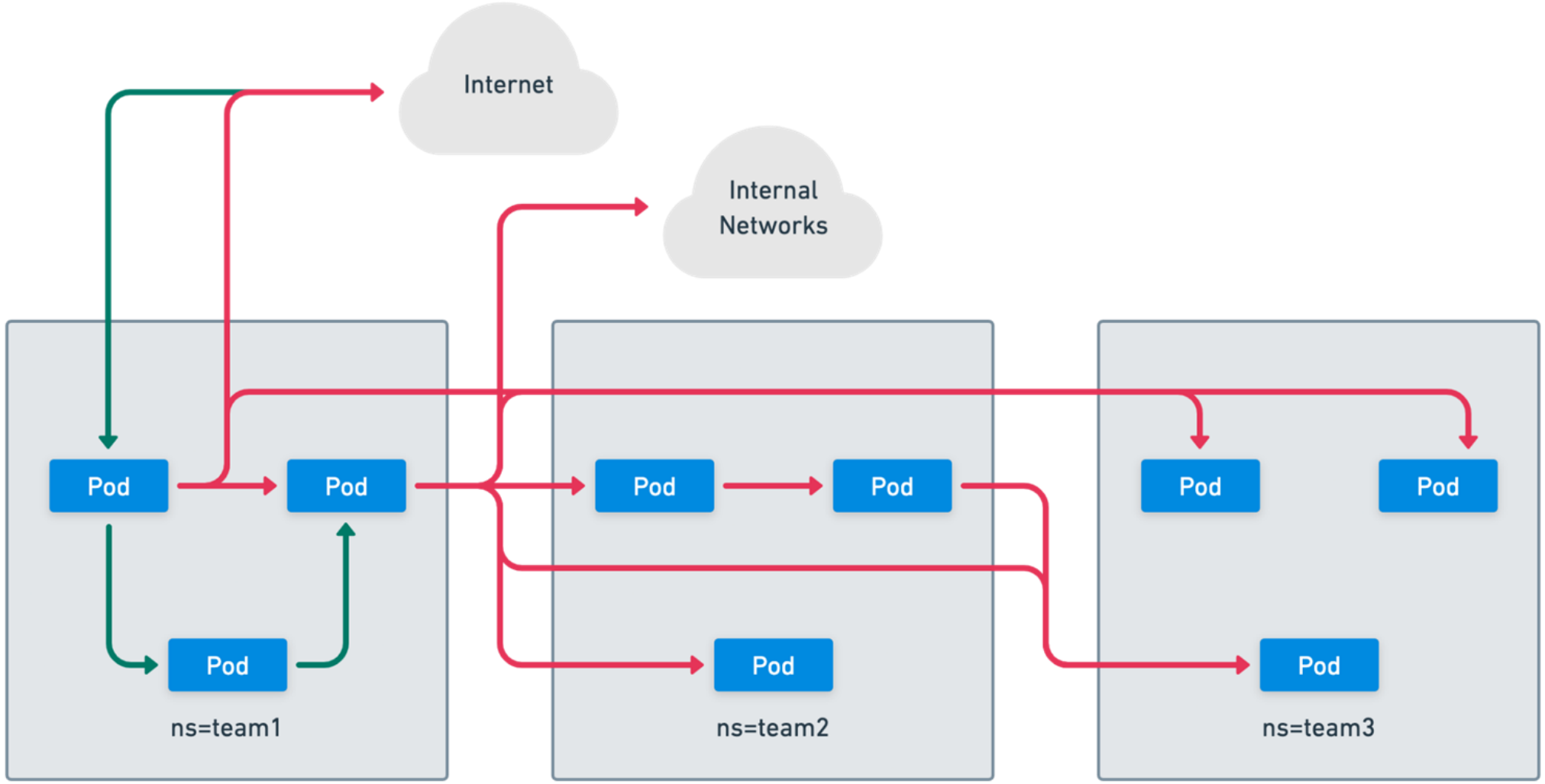
Exposure Type	Example
Cluster Ingress	• Service Exposed to Internet • Service Exposed to External Network • Overlay Routing: Internal service un-unintentionally exposed as a type:LoadBalancer or NodePort rather than ClusterIP • Direct Routing: All pods are reachable
Intra-Cluster Lateral	By default any pod to pod connectivity is allowed
Cluster Egress Lateral	Egress connections to other VMs/bare-metal services without corporate network, as well as cloud meta data, Kubernetes work nodes and API servers
Cluster Egress to Internet	Access to internet for destinations for data exfiltration, access to command & control servers (C2), crypto mining, etc

© 2025 Cisco and/or its affiliates. All rights reserved.

cisco

Kubernetes Network Risk Reduction

Wide Blast Radius

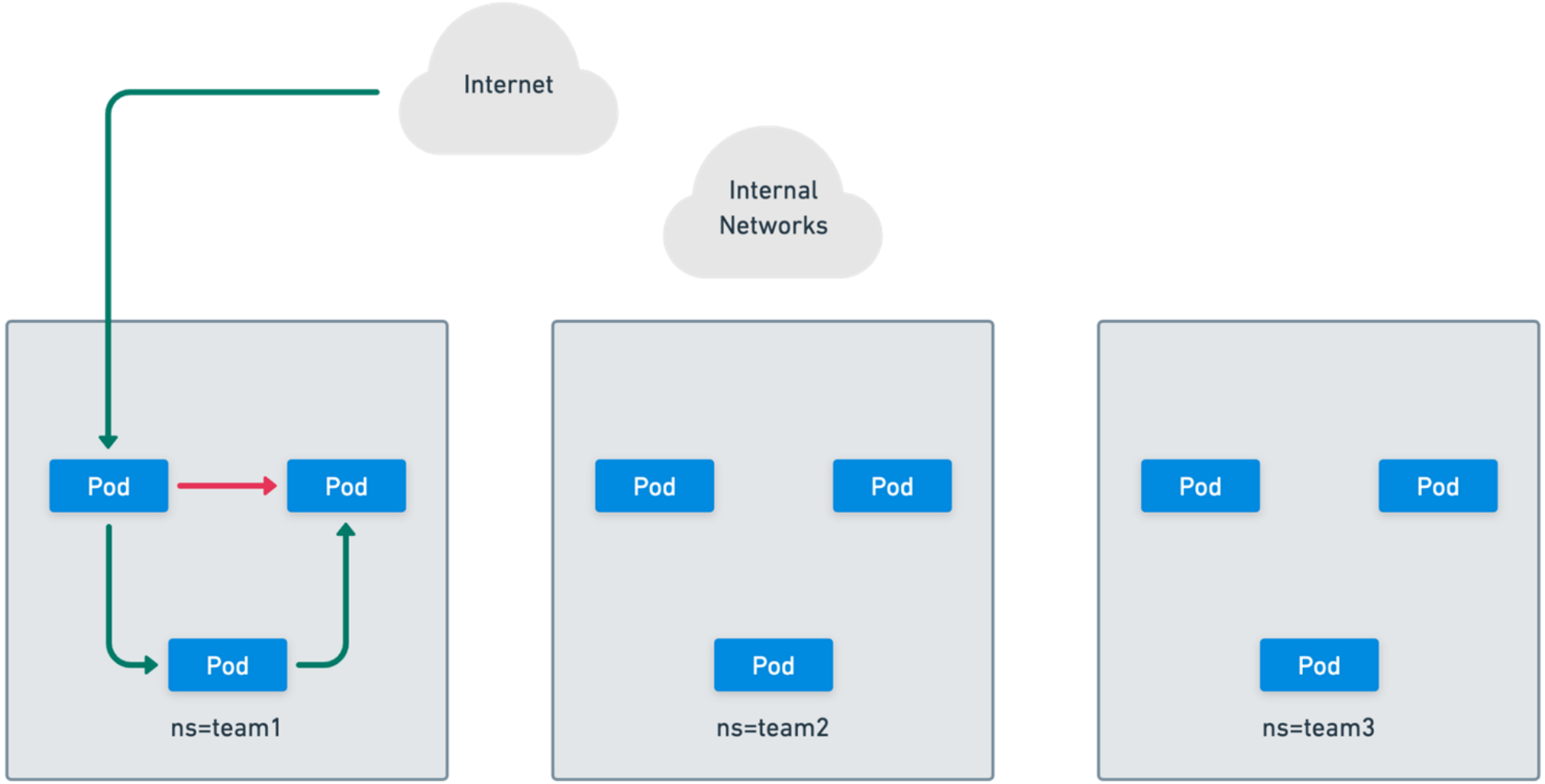


© 2025 Cisco and/or its affiliates. All rights reserved.

cisco

Kubernetes Network Risk Reduction

Limited Blast Radius



© 2025 Cisco and/or its affiliates. All rights reserved.

cisco

Measuring Risk

Quantifying Unused Access

Measuring Progress

- With rich network observability the “mismatch” between what is allowed by coarse-grained network policy (or lack of policy) is the focus of minimisation

Example

- Kube-dns being exposed to all workloads is intended and not a risk but a private tenant service being reachable by all pods
- A frontend service on port 443 that is supposed to be public should be reachable within and outside the cluster. The redis server that it uses to cache database results should only be visible to the frontend

© 2025 Cisco and/or its affiliates. All rights reserved.

cisco

Measuring Risk

Prioritisation

Concrete measurements to guide

- # of services reachable via Cluster Ingress with no matching traffic
- # of services reachable from other namespaces with no matching traffic
- # of services with access to External Networks or Internet with no matching connections

© 2025 Cisco and/or its affiliates. All rights reserved.

cisco

Coarse-Grained Per-Namespace

Reduce rule explosion

Concrete measurements to guide

- Allow all ingress/egress communications within a namespace
- Use Timescape Observability data to identify and permit
  - Which (service & port) within a namespace are public facing
    - Allow from cluster-only
    - Allow from “world” (loadBalancer/nodePort) or ingress namespace
  - Egress access to
    - Other services (optional, limit ports)
    - Other services in the external private network (optional, limit ports
    - Other services on the internet (optional, limit ports)
- Coarse-grained nature of policies mean that policies only change when an appl changes a core part of its communication pattern

© 2025 Cisco and/or its affiliates. All rights reserved.

cisco

Baseline vs Per-Namespace

Two approaches

Global Baseline Policies

- Default deny ingress/egress
- Allow all ingress/egress within a namespace
- Egress to “public services”
  - Cluster-wide shared services (e.g kube-dns, prometheus)
  - External shared CIDR/DNS (e.g food and water)
- Often implemented with CiliumClusterwideNetworkPolicies

Per-Namespace Policies

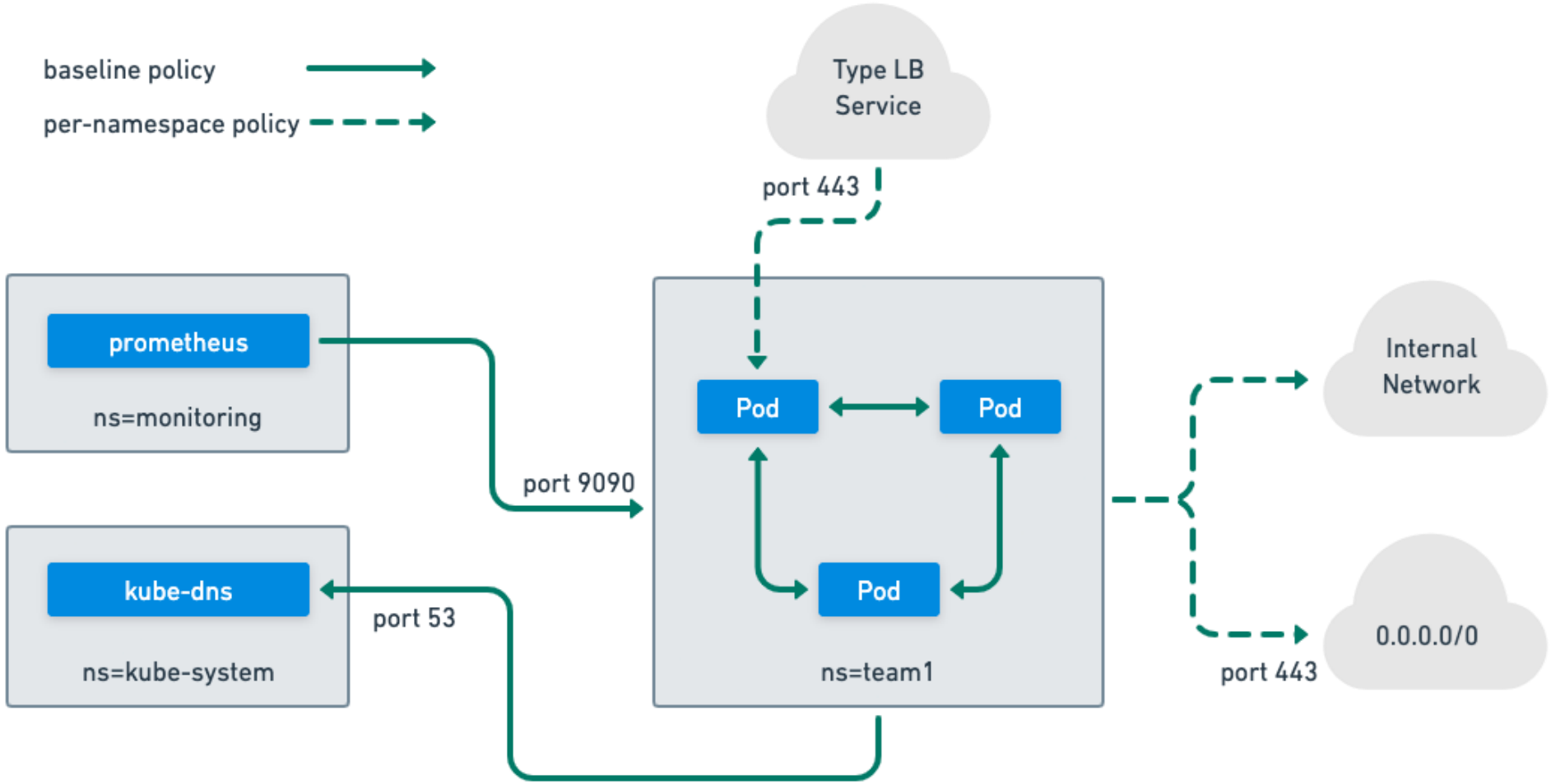
- Per-service ingress (limited by port)
  - Exposed within cluster via ClusterIP service
  - Exposed externally via ingress or LoadBalancer/NodePort
- Namespace-specific egress
  - No access
  - Egress to specific CIDR/DNS + port
  - Unrestricted access on specific ports

© 2025 Cisco and/or its affiliates. All rights reserved.

cisco

Kubernetes Network Baseline

Course-grained per namespace policy



© 2025 Cisco and/or its affiliates. All rights reserved.

cisco

Transitioning from Coarse to Fine-Grained

Move between policy types

Consider the following:

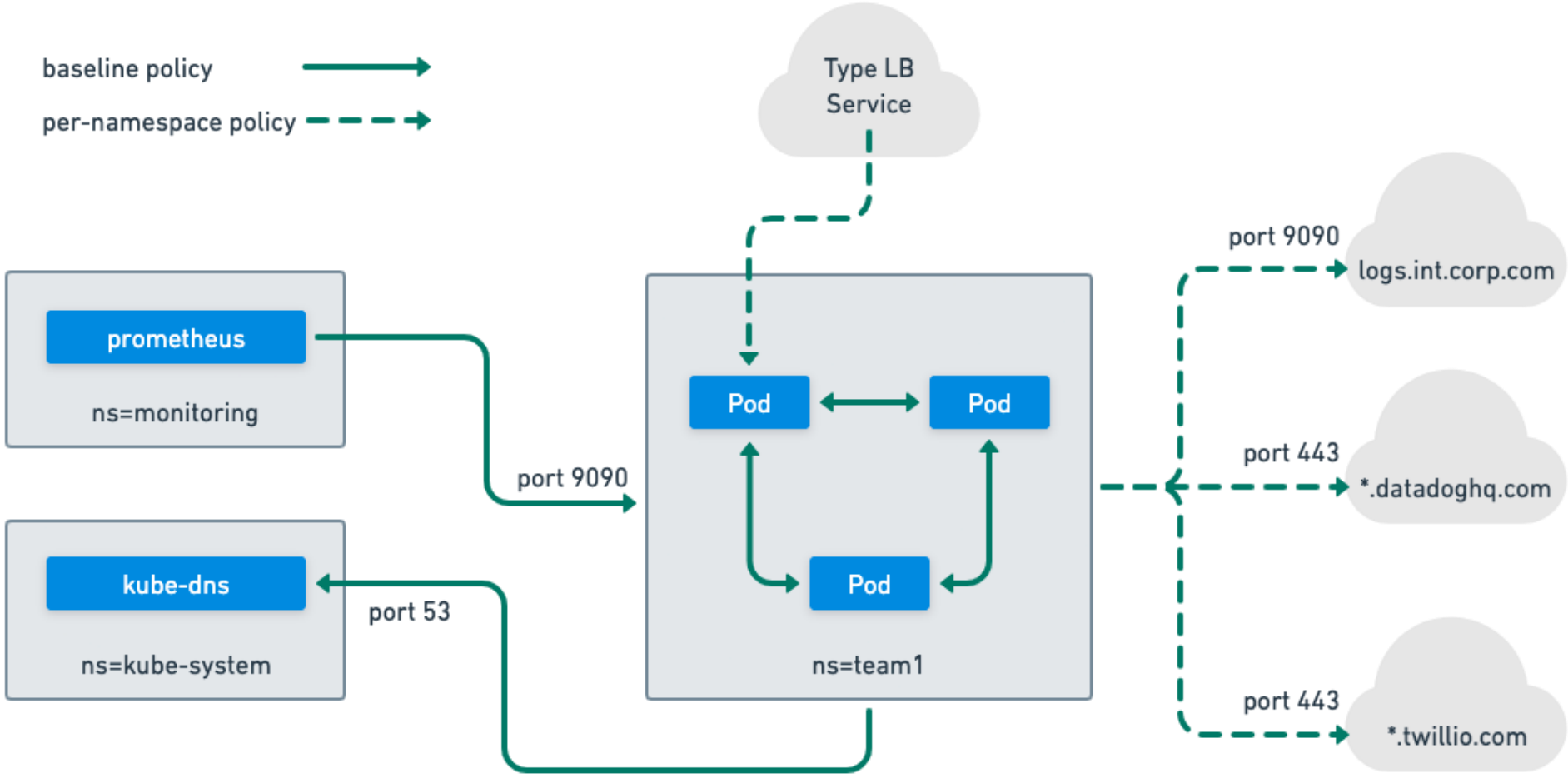
- Prioritise namespaces based on:
  - Most sensitive applications
  - Measurement of exposure vs used connectivity
- Transition of egress outside the cluster
  - Access to all private network becomes Access to specific FQDNs or small CIDRS (with ports)
  - Access to Internet becomes Access to specific FQDNs or small CIDRS (with ports)
- Transitions within the cluster:
  - Shift rules that allow all to/from cluster to allowing to/from specific namespaces or services (zone based)

© 2025 Cisco and/or its affiliates. All rights reserved.

cisco

Kubernetes Network Baseline

Fine-grained per namespace policy

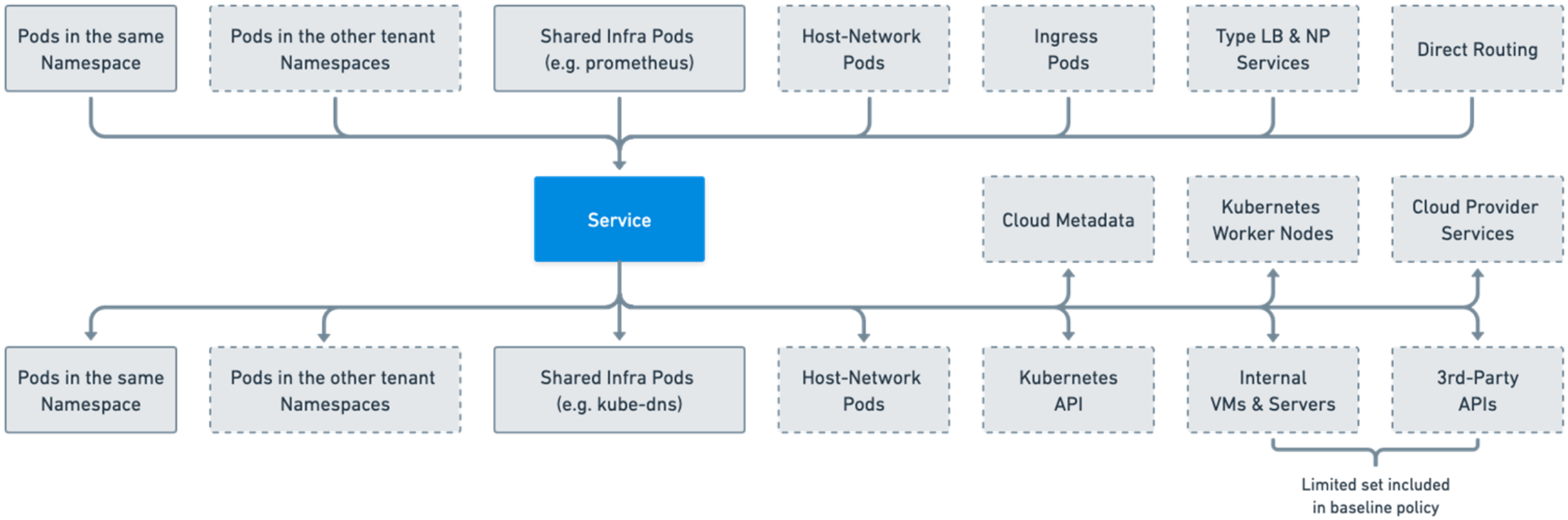


© 2025 Cisco and/or its affiliates. All rights reserved.

cisco

Policy Sources and Destinations to Consider

Example Enterprise traffic flows



© 2025 Cisco and/or its affiliates. All rights reserved.

cisco

Delegating Network Policy Creation

Who manages the policy

Consider the following:

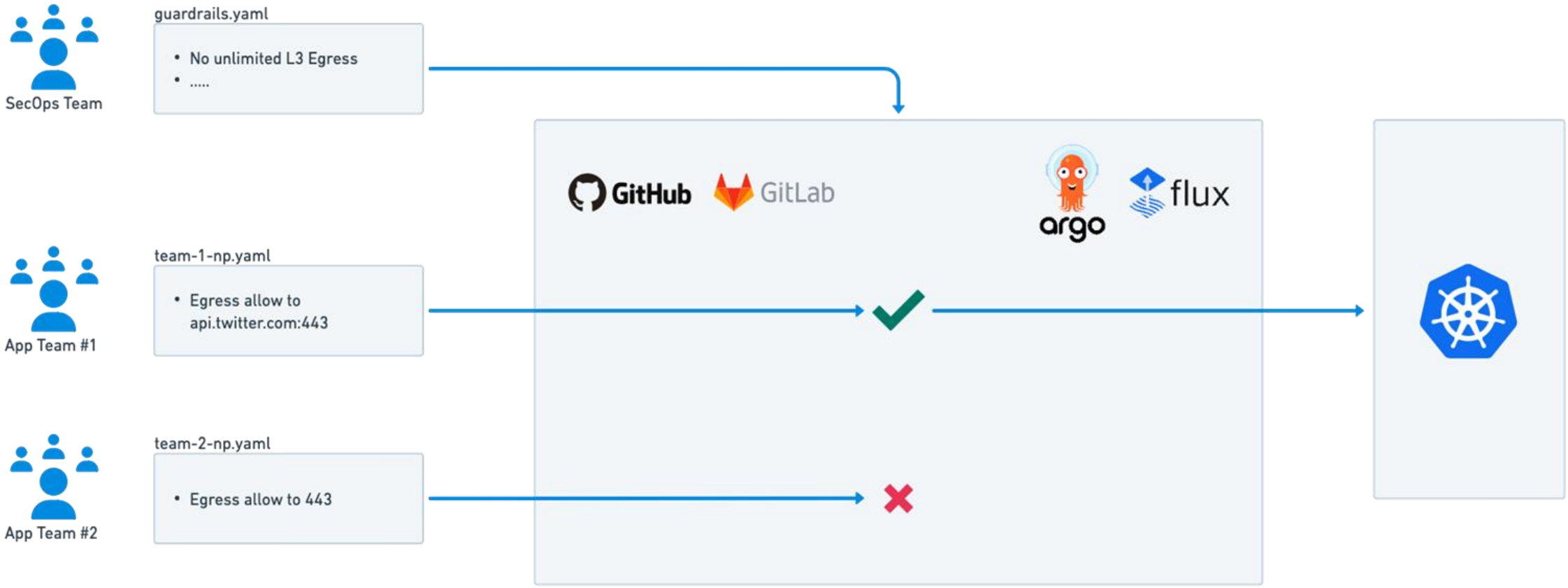
- Prioritise namespaces based on:
  - Most sensitive applications
  - Measurement of exposure vs used connectivity
- Transition of egress outside the cluster
  - Access to all private network becomes Access to specific FQDNs or small CIDRS (with ports)
  - Access to Internet becomes Access to specific FQDNs or small CIDRS (with ports)
- Transitions within the cluster:
  - Shift rules that allow all to/from cluster to allowing to/from specific namespaces or services (zone based)

© 2025 Cisco and/or its affiliates. All rights reserved.

cisco

Network Policy Guardrails

Control with application self-service management



© 2025 Cisco and/or its affiliates. All rights reserved.

GitOps-based Network Policy

Approvals and Guardrails

Consider the following:

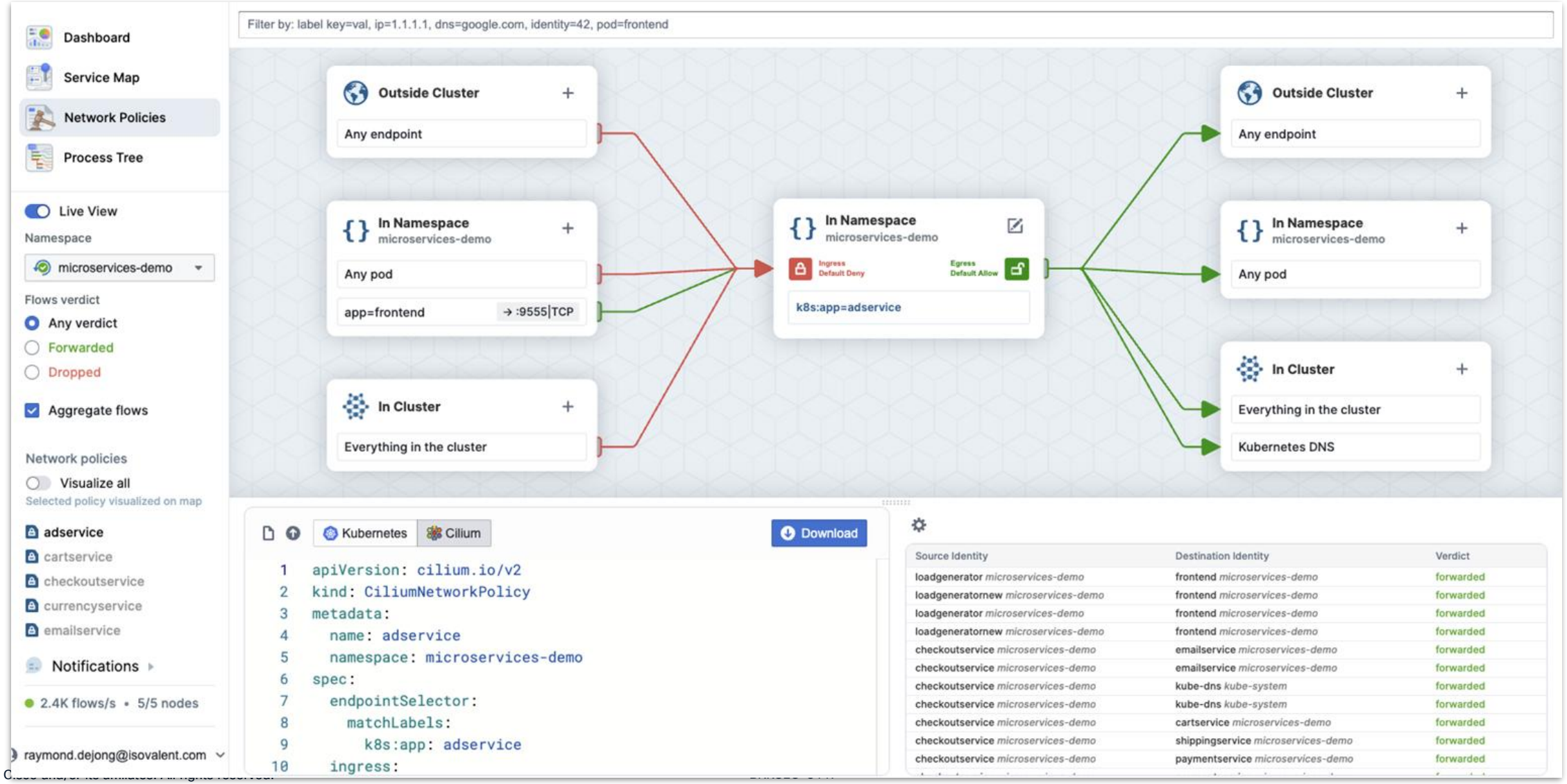
- Example of “direct access with validation”
  - Policies are created via preferred method
  - All policies pushed via GitOps-based workflow
  - Declarative guardrails created by SecOps ensures acceptable rules (no 0.0.0.0/0)
  - Polices with violations require exception approval or rejected (after PR reviews, etc)
- Three patters usually arise
  - Require egress destination to match a list of “approved” DNS domains or CIDR’s
  - Prevent egress access to specific sensitive destination or set of destinations
    - Cloud metadata services
    - Access to Kubernetes worker nodes
  - Prevent wild “wildcarding”
    - Large CIDR domains (beyond /24)
    - FQDN wildcards

© 2025 Cisco and/or its affiliates. All rights reserved.

cisco

Timescape Network Policy Editor

Visualise policy edits and changes within your environment



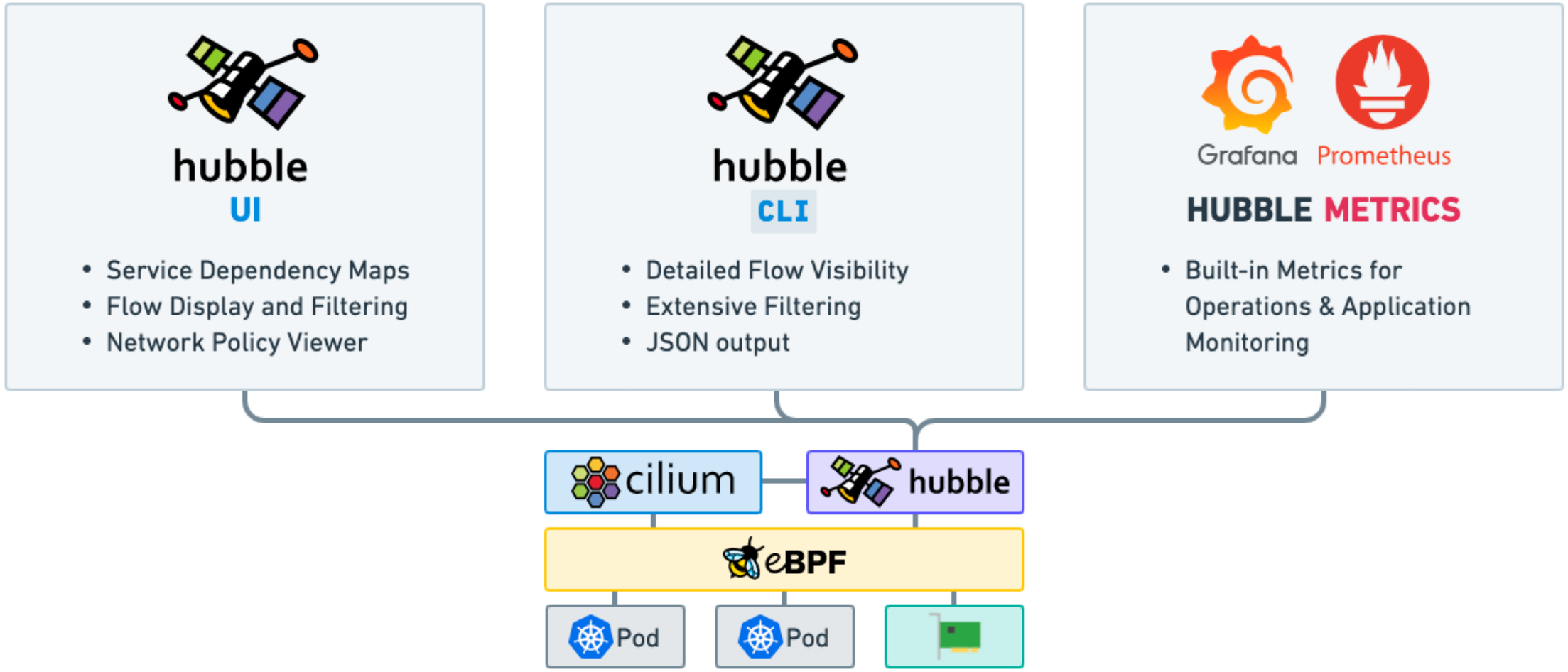
© 2025 Cisco and/or its affiliates. All rights reserved.

Operational Considerations

cisco Live !

Timescape Overview

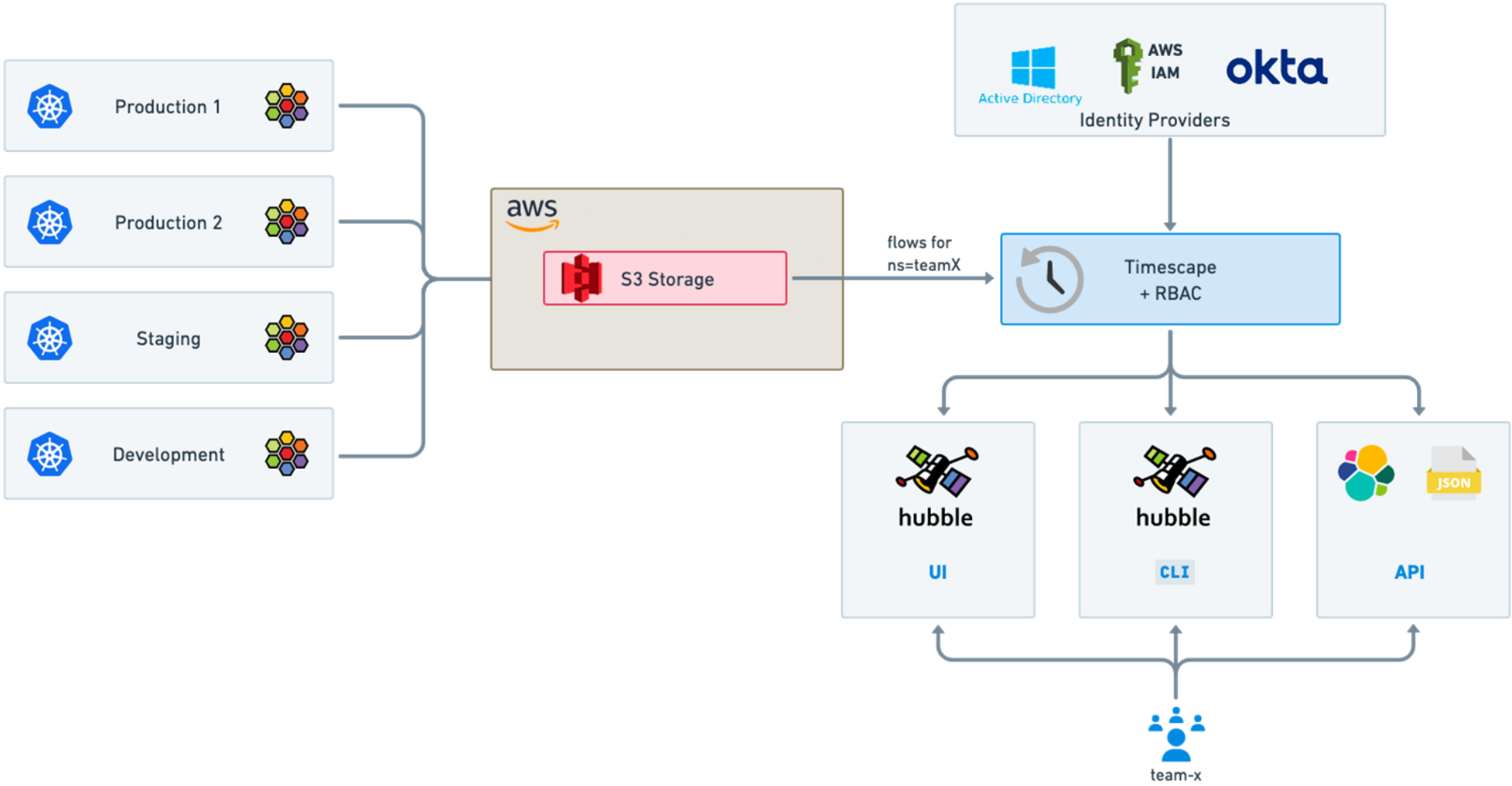
The power of eBPF Network Visibility



© 2025 Cisco and/or its affiliates. All rights reserved.

Timescape Overview

Troubleshoot policies



© 2025 Cisco and/or its affiliates. All rights reserved.

Troubleshooting Network Policy

Who manages the policy

Troubleshooting Data from Timescape

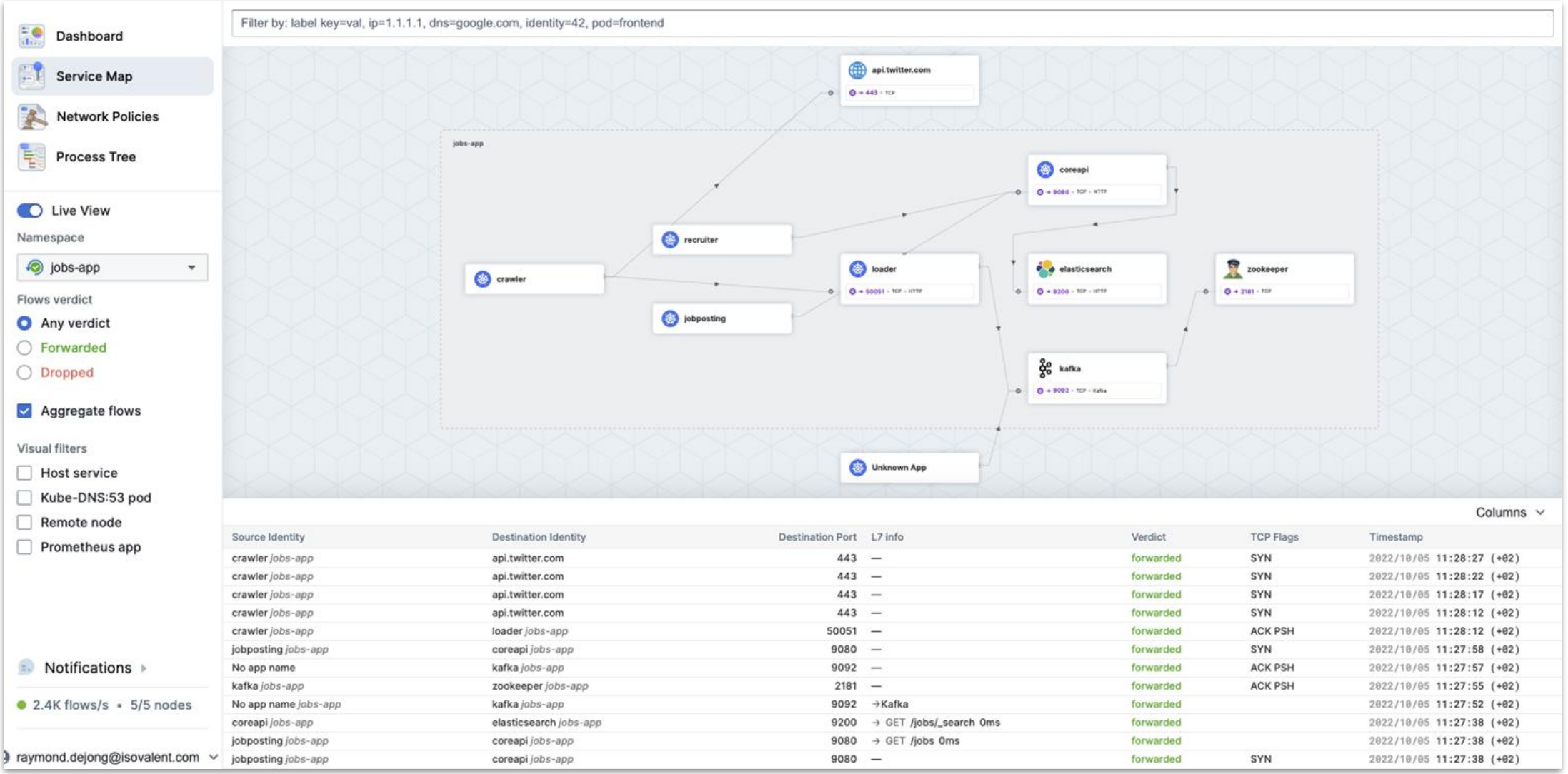
- Identity labels of source
- Identity labels of destination
- Policy verdict
- “allowed by” policy match (or lack of match)
- Current set of policy enforced on egress and ingress
- Date/Time
  - Policy drops
  - Changes to policy rules
  - Container start times correlated with services
  - Node issues

© 2025 Cisco and/or its affiliates. All rights reserved.

cisco

Timescape Service Map

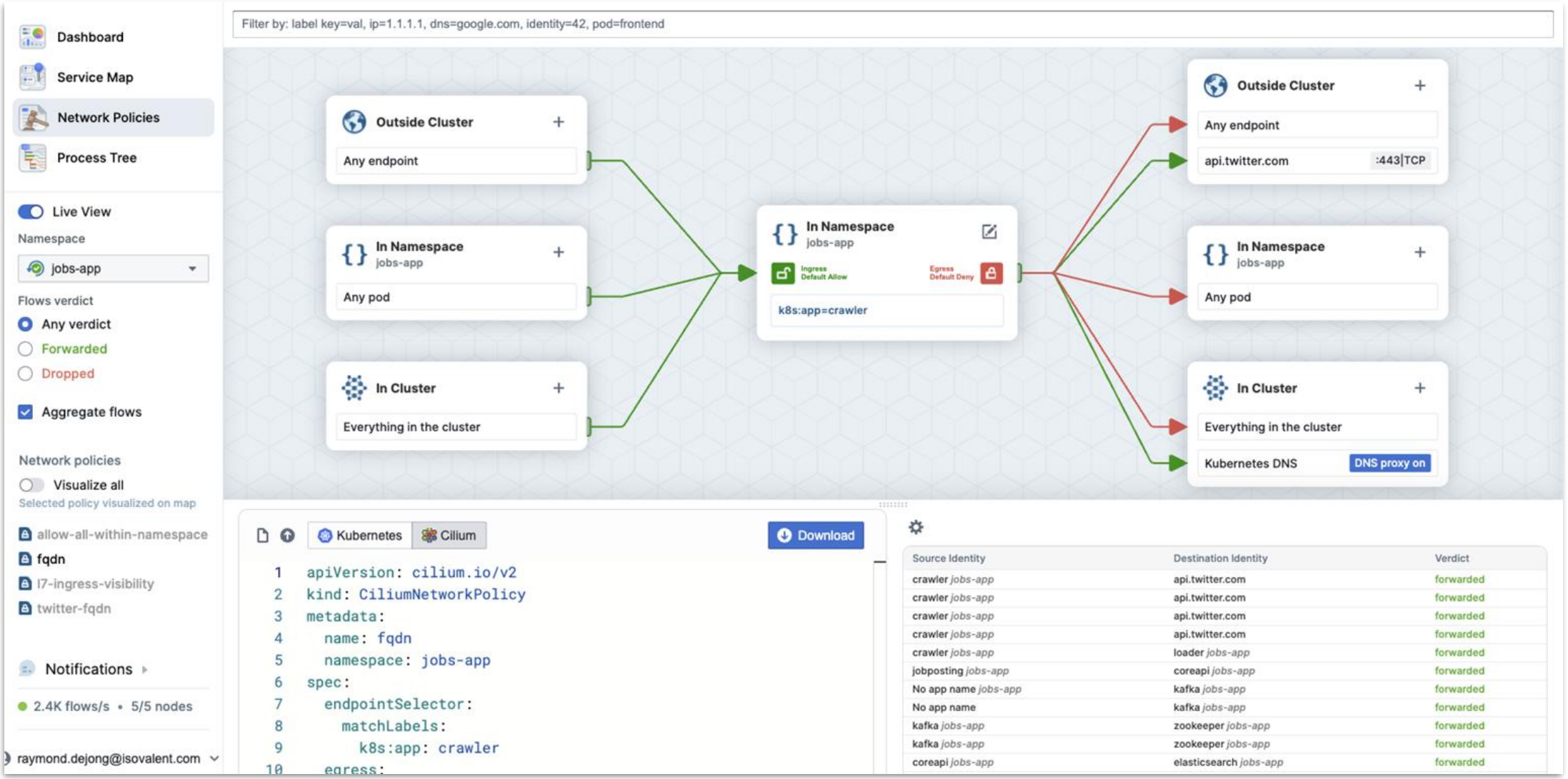
See what is communicating automatically



© 2025 Cisco and/or its affiliates. All rights reserved.

Timescape Overview

Data driven policy editing

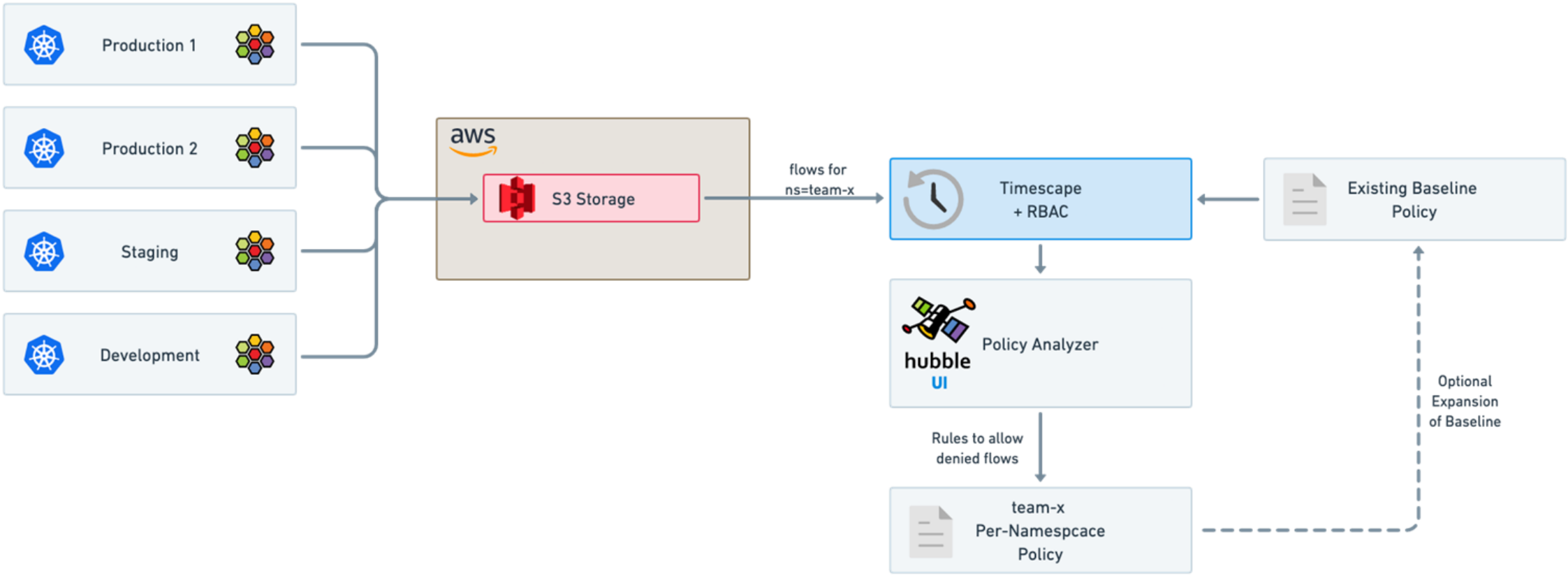


© 2025 Cisco and/or its affiliates. All rights reserved.

cisco

Timescape Overview

Policy Analyser to provide recommendations

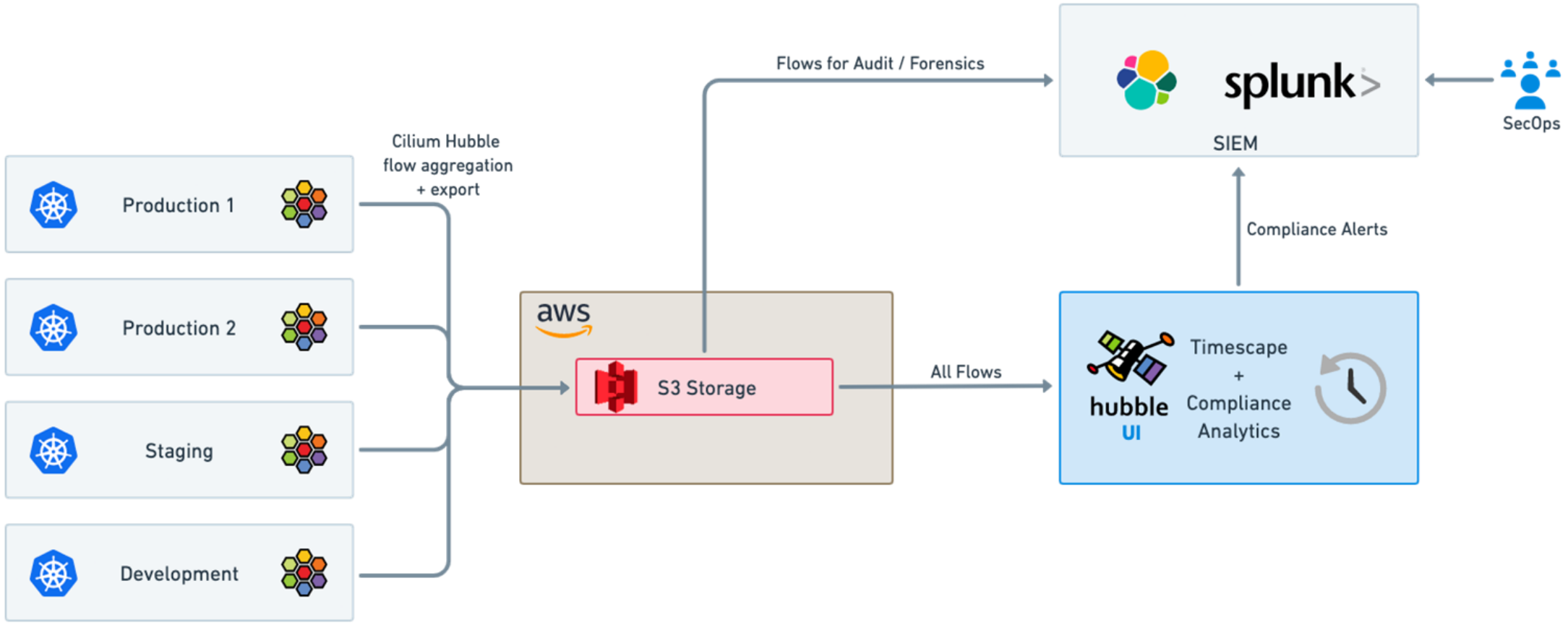


© 2025 Cisco and/or its affiliates. All rights reserved.

cisco

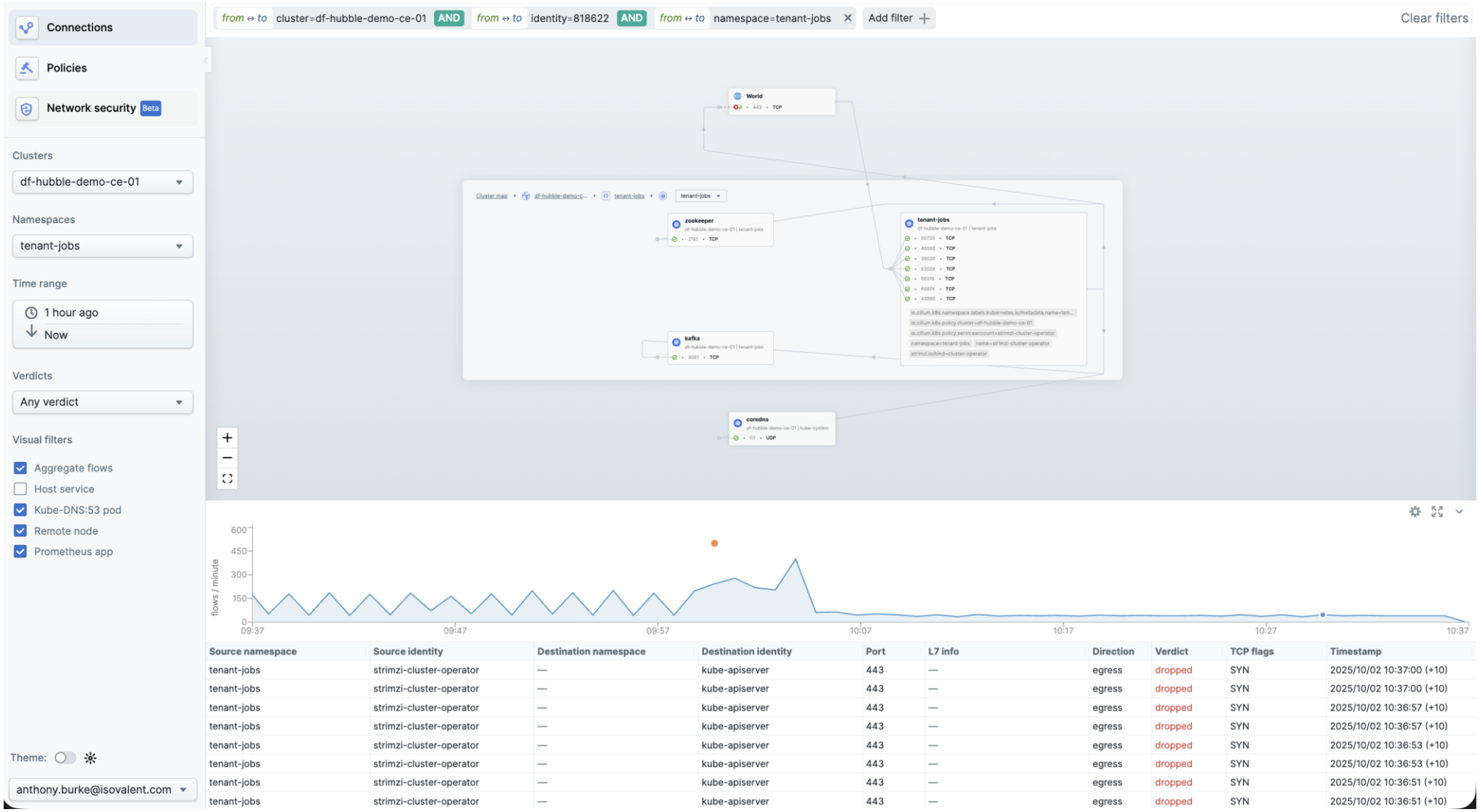
Timescape for Security Compliance

Validate with SIEM integration



© 2025 Cisco and/or its affiliates. All rights reserved.

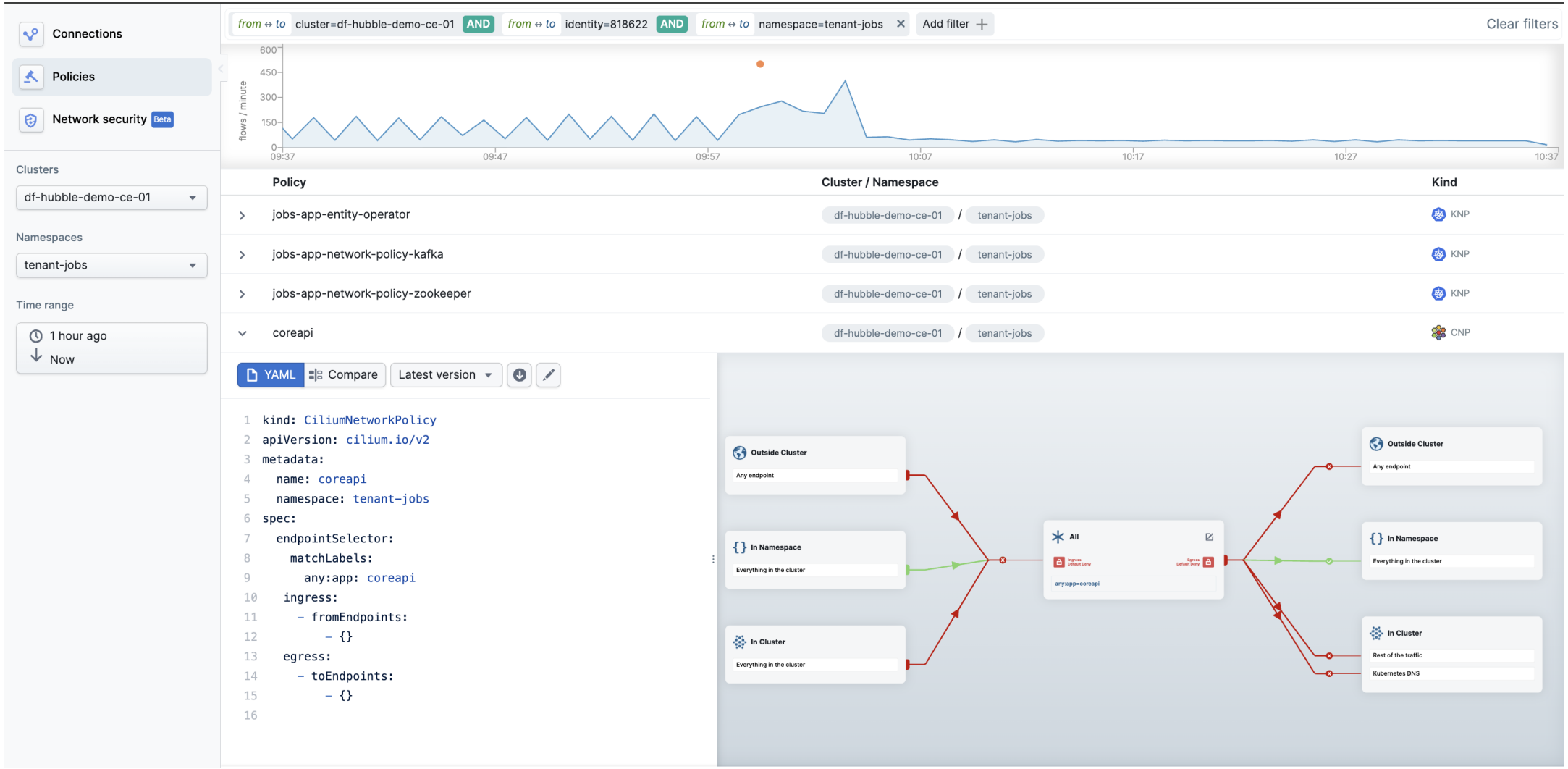
Service Map



Source namespace	Source identity	Destination namespace	Destination identity	Port	L7 info	Direction	Verdict	TCP flags	Timestamp
tenant-jobs	strimzi-cluster-operator	—	kube-apiserver	443	—	egress	dropped	SYN	2025/10/02 10:37:00 (+10)
tenant-jobs	strimzi-cluster-operator	—	kube-apiserver	443	—	egress	dropped	SYN	2025/10/02 10:37:00 (+10)
tenant-jobs	strimzi-cluster-operator	—	kube-apiserver	443	—	egress	dropped	SYN	2025/10/02 10:36:57 (+10)
tenant-jobs	strimzi-cluster-operator	—	kube-apiserver	443	—	egress	dropped	SYN	2025/10/02 10:36:57 (+10)
tenant-jobs	strimzi-cluster-operator	—	kube-apiserver	443	—	egress	dropped	SYN	2025/10/02 10:36:53 (+10)
tenant-jobs	strimzi-cluster-operator	—	kube-apiserver	443	—	egress	dropped	SYN	2025/10/02 10:36:53 (+10)
tenant-jobs	strimzi-cluster-operator	—	kube-apiserver	443	—	egress	dropped	SYN	2025/10/02 10:36:51 (+10)
tenant-jobs	strimzi-cluster-operator	—	kube-apiserver	443	—	egress	dropped	SYN	2025/10/02 10:36:51 (+10)

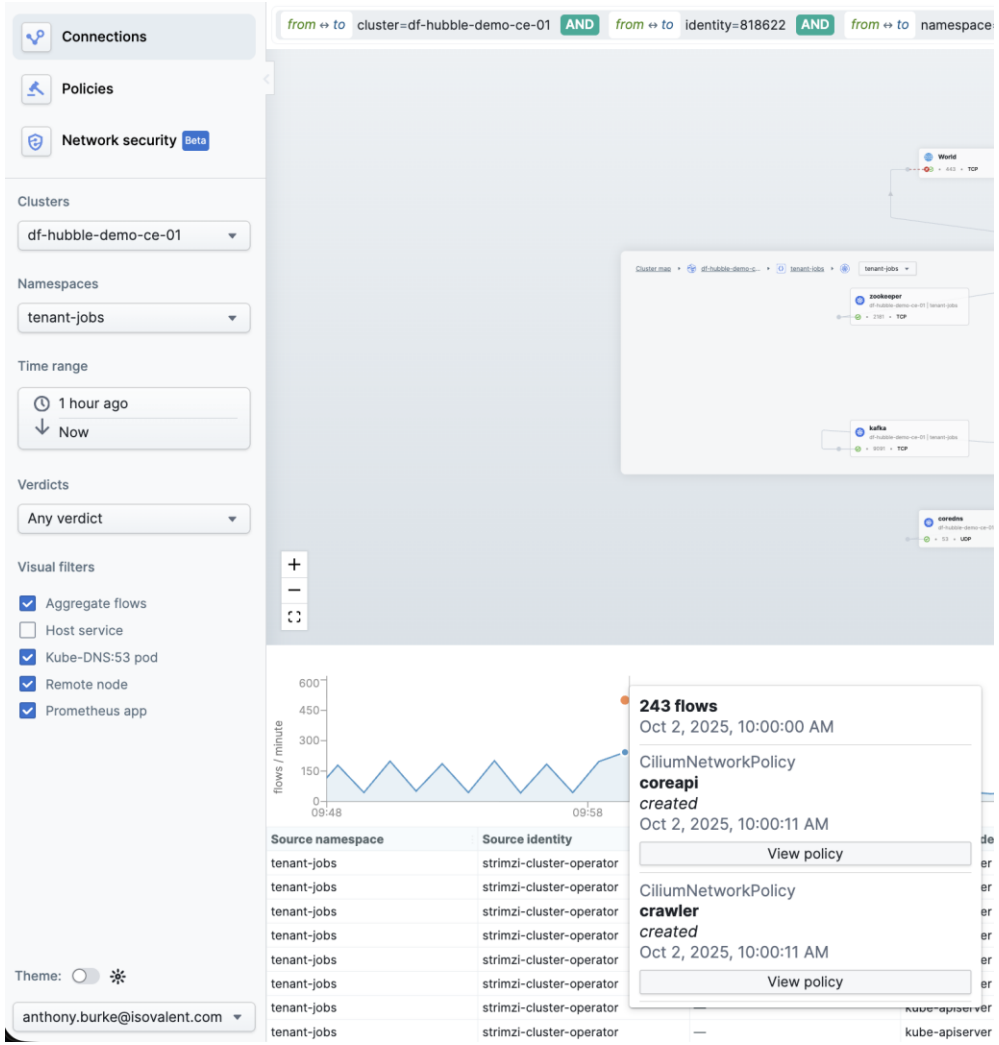
© 2025 Cisco and/or its affiliates. All rights reserved.

Policies per Namespace



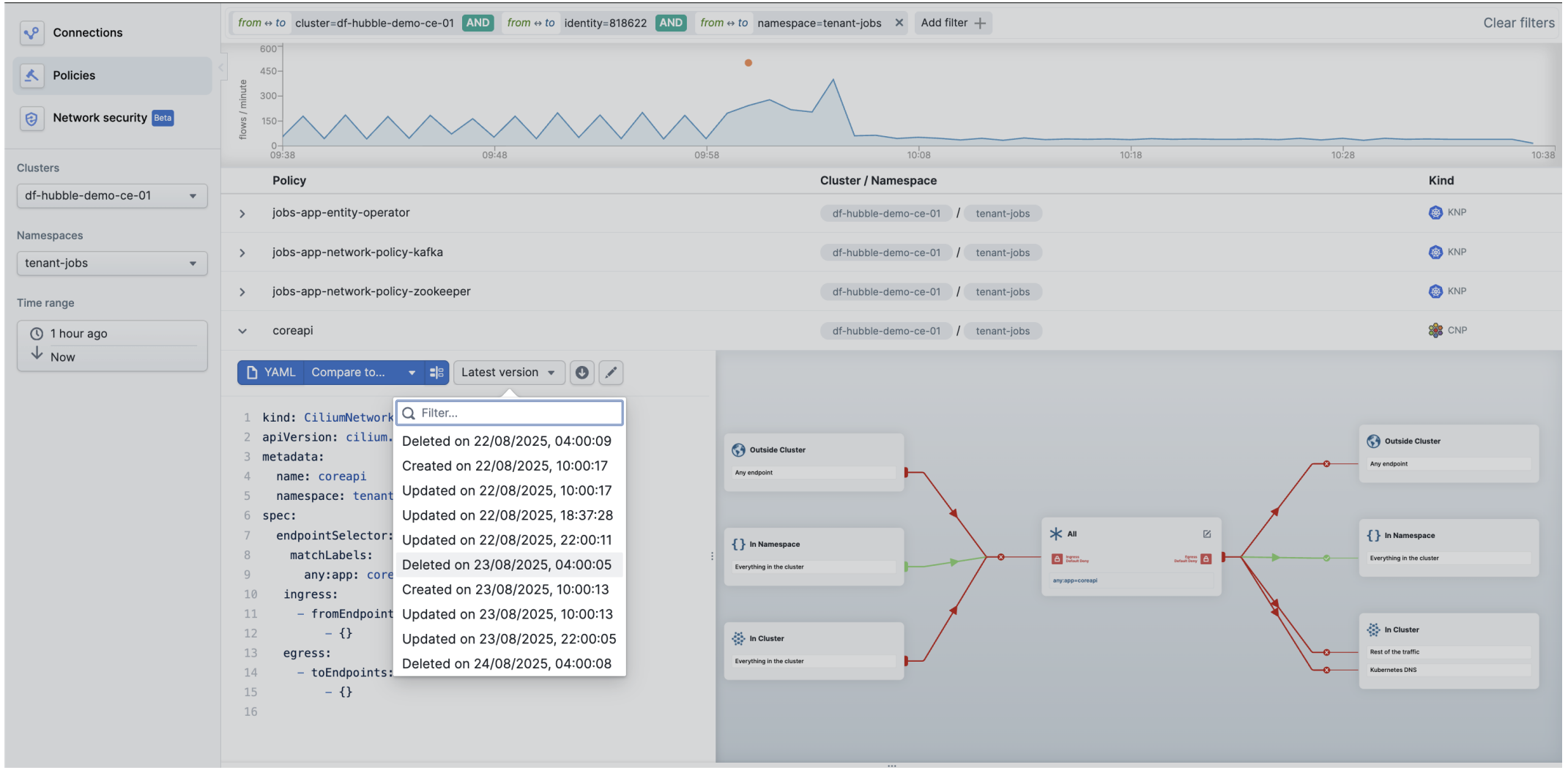
© 2025 Cisco and/or its affiliates. All rights reserved.

Timescape shows policy change...



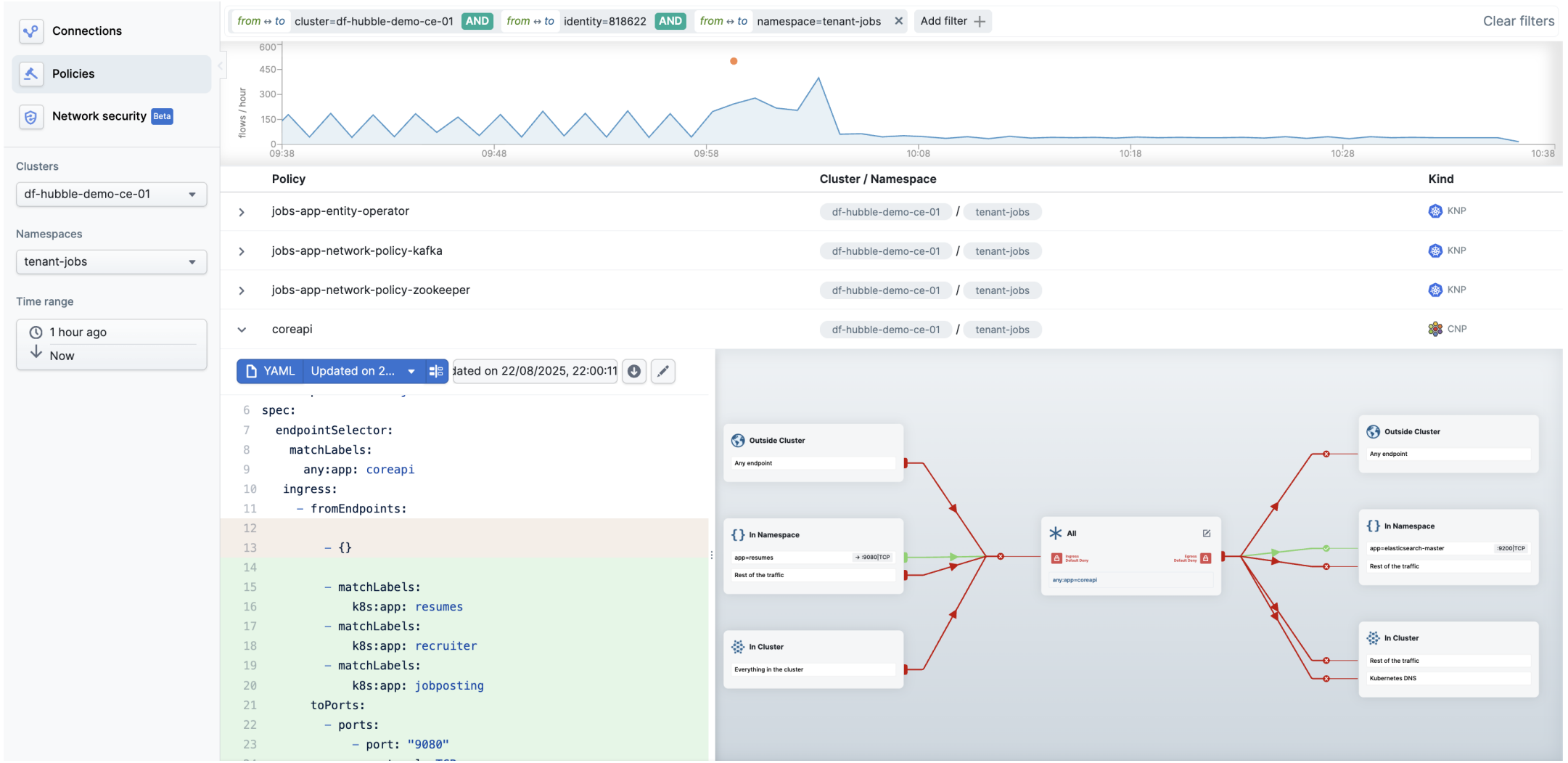
© 2025 Cisco and/or its affiliates. All rights reserved.

... and allows comparison to previous policies



© 2025 Cisco and/or its affiliates. All rights reserved.

Including a differential between current and prior



© 2025 Cisco and/or its affiliates. All rights reserved.

Summary

Summary

Isovalent provides

- Provide a thorough understanding of enforcement types and options
- Provide guidance around segmentation design principals using Isovalent
- Demonstrate operational visibility with Timescape in provider/consumer models

Microsegmentation starts with

- Understanding risk profile of applications and stack-rack accordingly
- Using observability from Timescape to as a starting point for SecOps and AppTeams and as a data driven reference
- Leverage identity and logical objects to segment workloads with finer namespace rules

© 2025 Cisco and/or its affiliates. All rights reserved.

Complete your session evaluations

Complete a minimum of 4 session surveys and the Overall Event Survey to claim a Cisco Live T-Shirt.

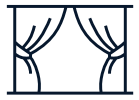
Earn up to 800 points by completing all surveys and climb the Cisco Live Challenge leaderboard.

Level up and earn exclusive prizes!

Complete your surveys in the Cisco Live Events app.

© 2025 Cisco and/or its affiliates. All rights reserved.

Continue your education



Visit the Cisco Stand for related demos

Book your one-on-one Meet the Expert meeting

Attend the interactive education with Capture the Flag, and Walk-in Labs

Visit the On-Demand Library for more sessions at www.CiscoLive.com/on-demand

© 2025 Cisco and/or its affiliates. All rights reserved.

Thank you

cisco Live !

CISCO