

Open Policy Agent (OPA)

Unified Cloud-native Policy Control



Who Are We?



Tim Hinrichs

Co-founder & CTO at Styra
Co-creator of OPA

@tim on OPA 
@thinrichs 



Torin Sandall

Engineer at Styra
Co-creator of OPA

@tsandall on OPA 
@sometorin 

openpolicyagent.org



Agenda

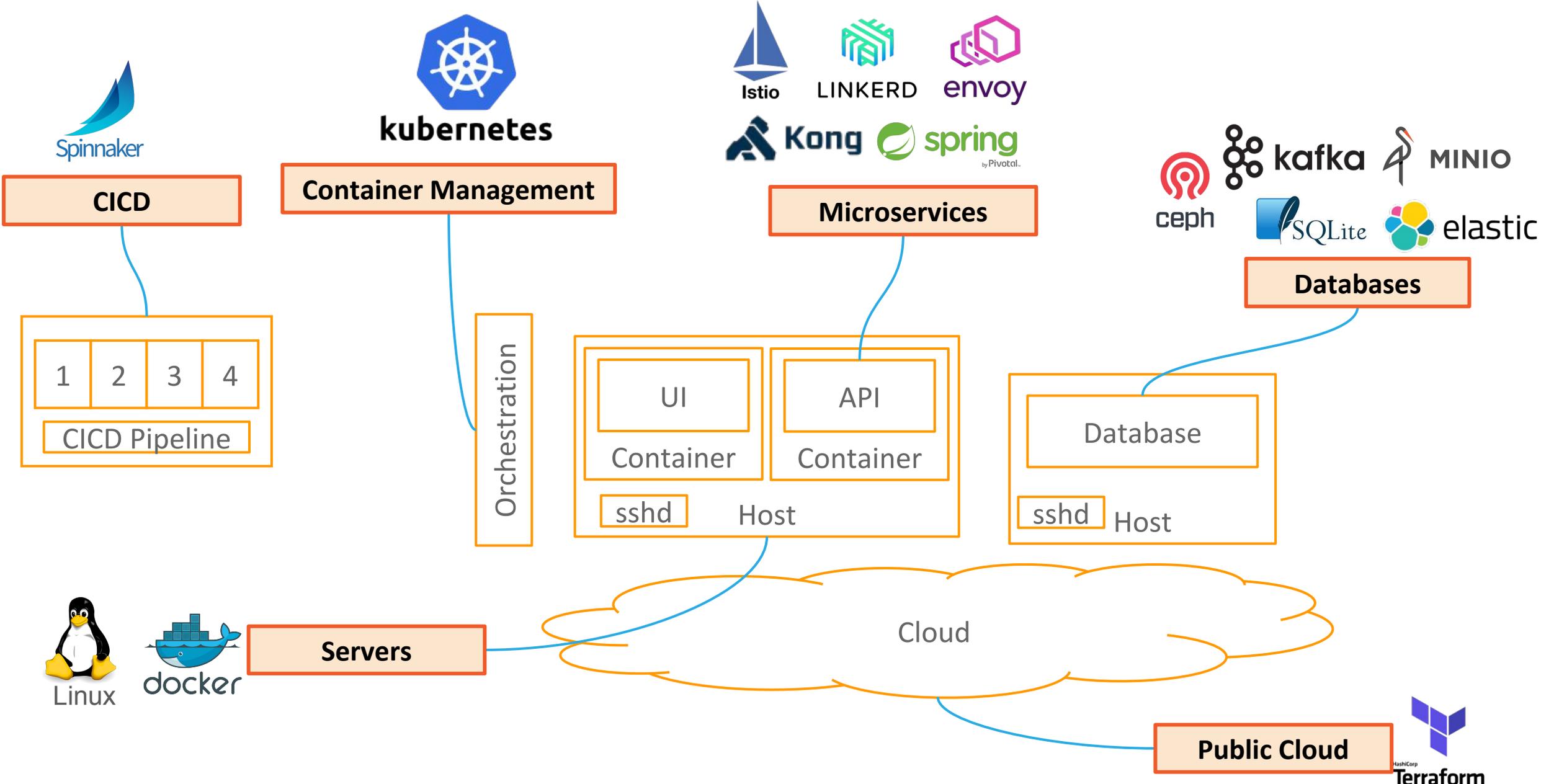
OPA Overview

Use Case Deep Dive

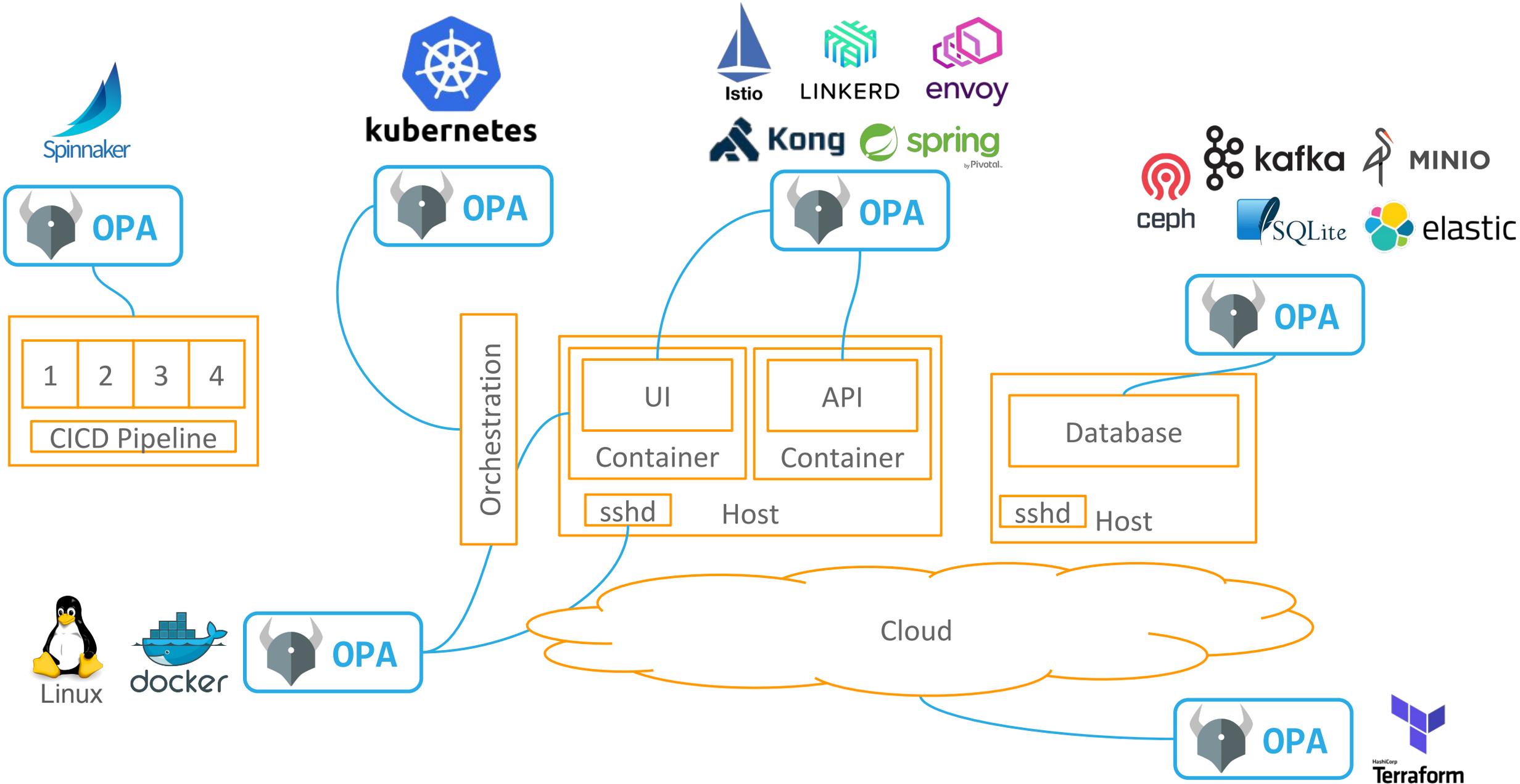
New and Future Features



Cloud-native Diversity/Dynamism Make Policy Management Challenging



OPA: Unified Policy Across the Stack



OPA Integration Index

openpolicyagent.org/docs/edge/ecosystem/

Open Policy Agent

edge pre-release

CORE DOCS

- Introduction
- Philosophy
- Policy Language
- Policy Performance
- Policy Testing
- Policy Reference
- Policy Cheatsheet
- External Data
- Integrating OPA
- Extending OPA
- REST API

KUBERNETES

- Overview & Architecture
- Policy Primer via Examples
- Tutorial: Ingress Validation
- Debugging Tips

OTHER USE CASES

- Docker
- HTTP APIs
- Kafka
- SSH and sudo
- Terraform
- Envoy

OPA Ecosystem

Showcase of OPA integrations, use-cases, and related projects.

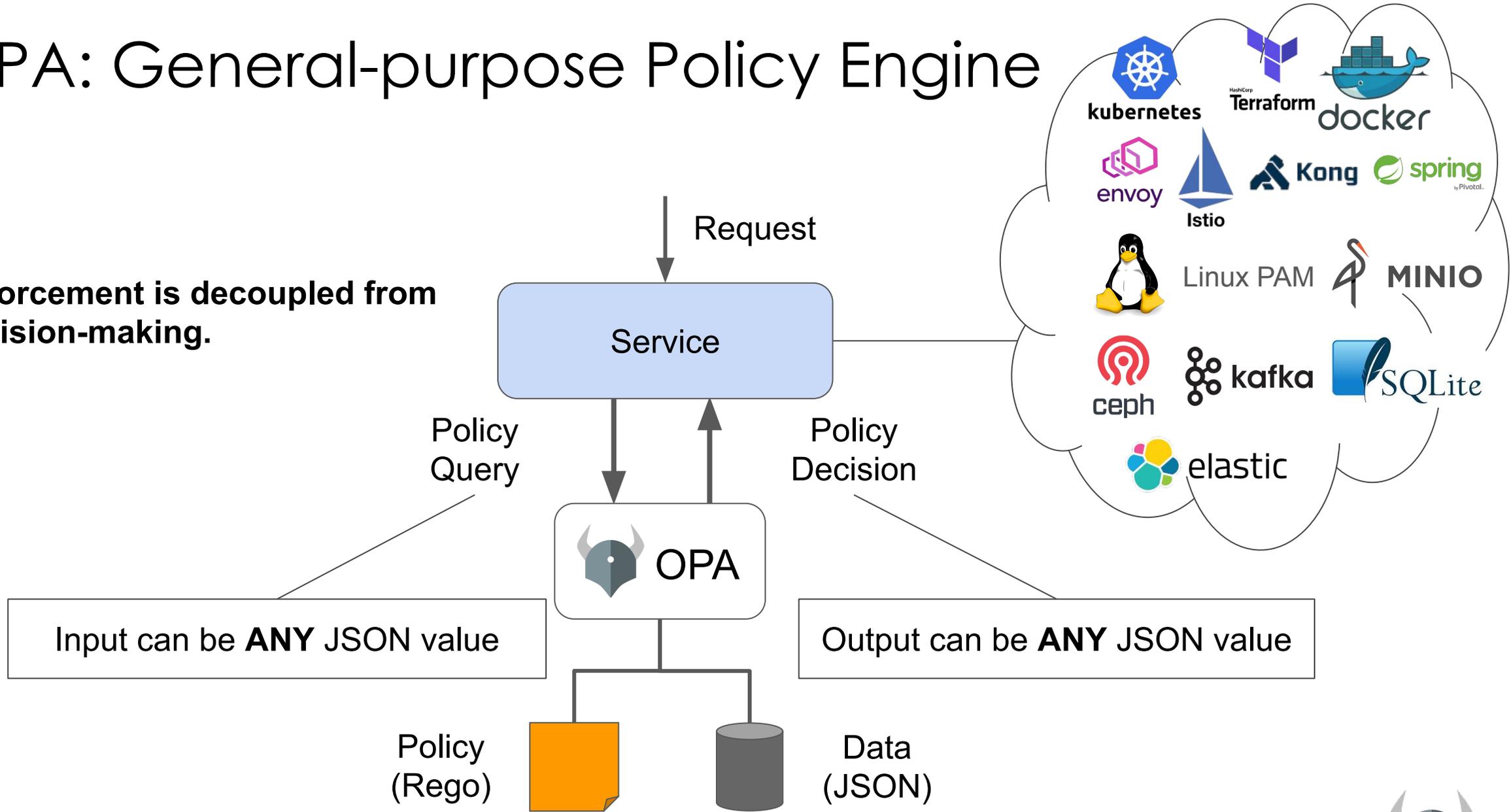
Add or Update Integration

 ceph Ceph Object Storage Authorization	 clair Kubernetes Admission Control using Vulnerability Scanning	 CLOUDFLARE Cloudflare Worker Enforcement of OPA Policies Using WASM	 Confest -- Configuration checking	 HTTP API Authorization in Dart	 docker Docker controls via OPA Policies
 elastic Elasticsearch Data Filtering	 envoy Container Network Authorization with Envoy	 IPTables	 Container Network Authorization with Istio (at the Edge)	 Container Network Authorization with Istio (as part of Mixer)	 kafka Kafka Topic Authorization
 Kong API Gateway Authorization with	 Kubernetes Admission Control	 SSH and Sudo Authorization with	 Minio API Authorization	 OpenFaaS Serverless Functions	 Spinnaker Pipeline Policy Enforcement



OPA: General-purpose Policy Engine

Enforcement is decoupled from decision-making.



Input can be **ANY** JSON value

Output can be **ANY** JSON value

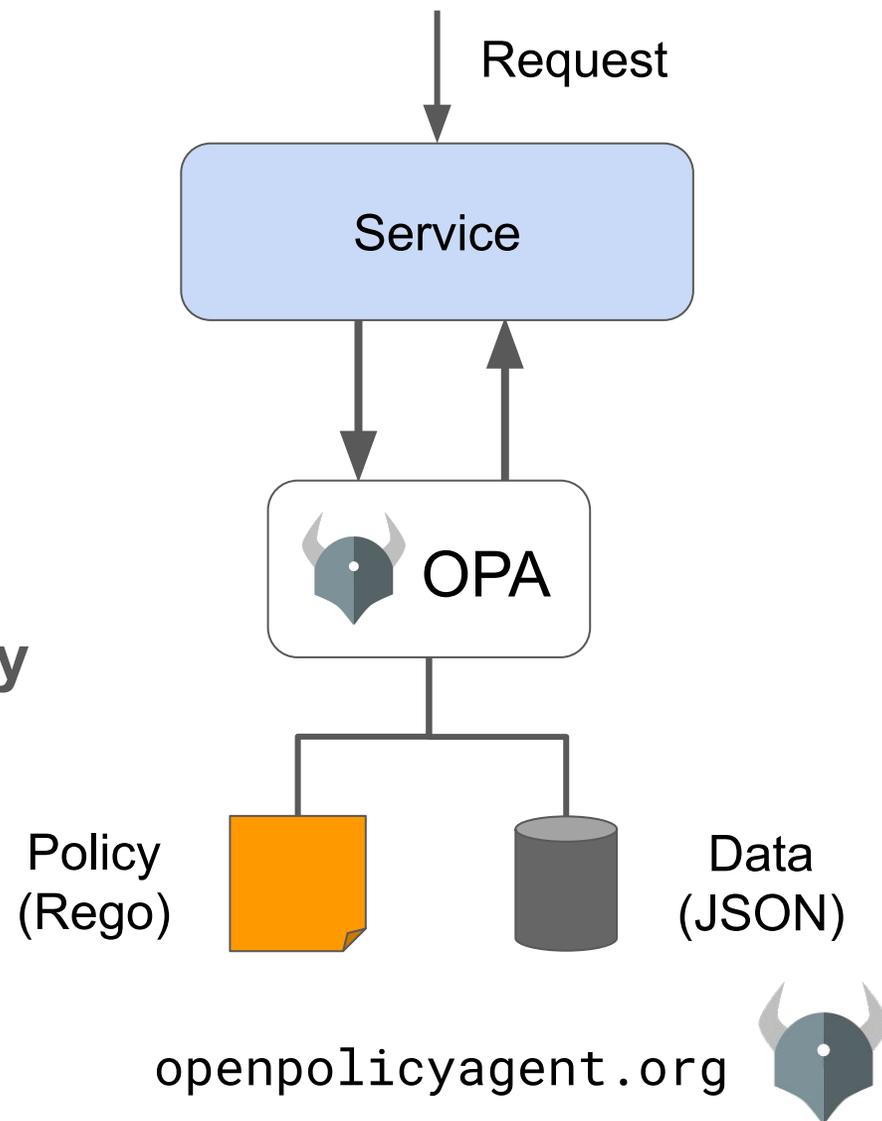
Policy (Rego)

Data (JSON)



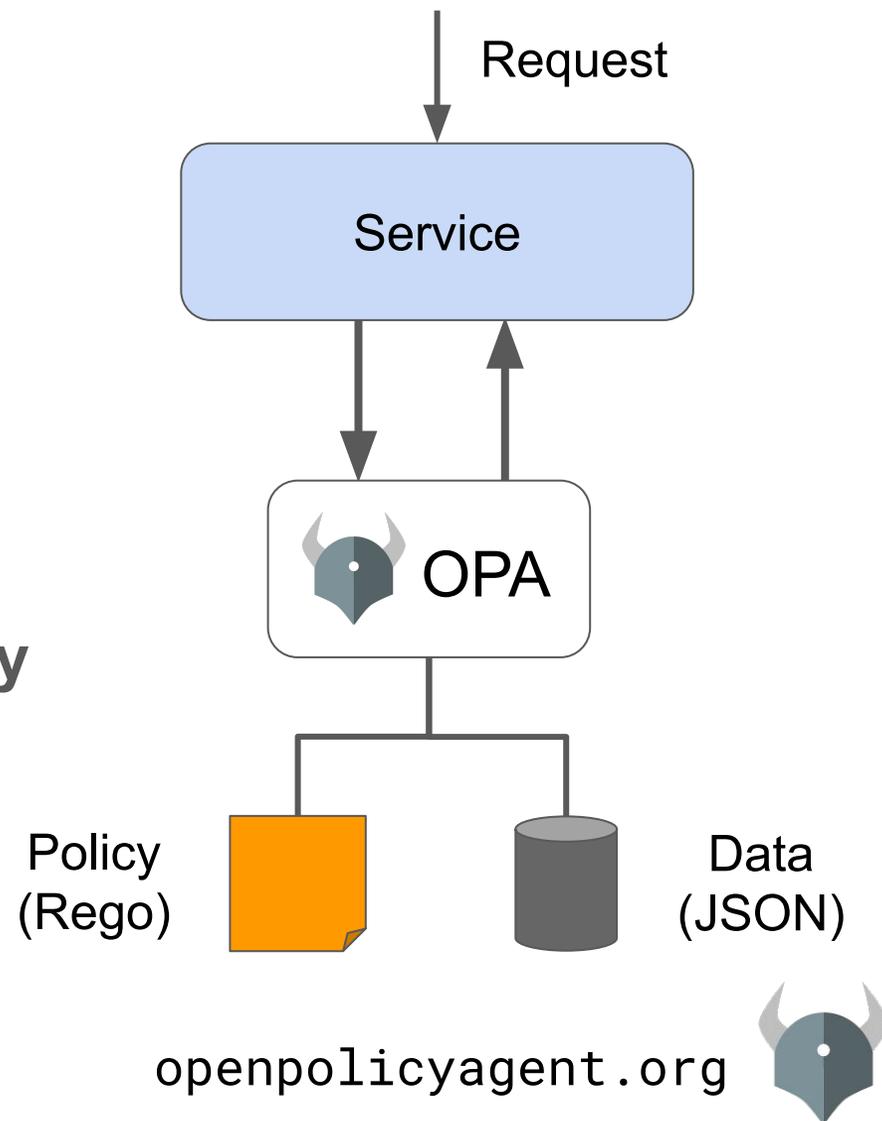
OPA: Policy-as-code

- **Declarative Policy Language (Rego)**
 - Can user X do operation Y on resource Z?
 - What invariants does workload W violate?
 - Which records should bob be allowed to see?
- **Library (Go), sidecar/host-level daemon**
 - Policy and data are kept in-memory
 - Zero decision-time dependencies
- **Management APIs for control & observability**
 - Bundle service API for sending policy & data to OPA
 - Status service API for receiving status from OPA
 - Log service API for receiving audit log from OPA
- **Tooling to build, test, and debug policy**
 - opa run, opa test, opa fmt, opa deps, opa check, etc.
 - VS Code plugin, Tracing, Profiling, etc.



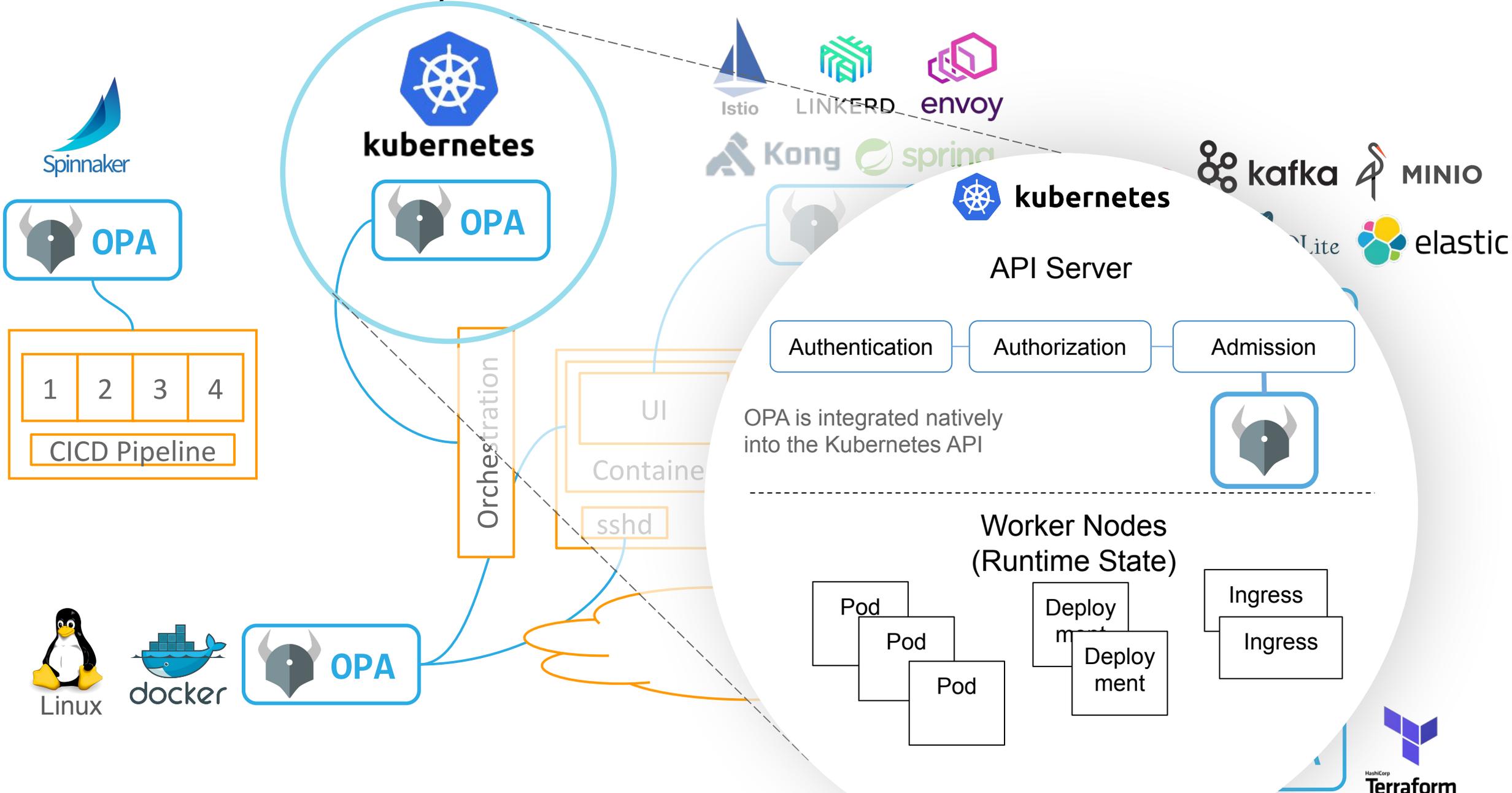
OPA: Policy-as-code

- **Declarative Policy Language (Rego)**
 - Can user X do operation Y on resource Z?
 - What invariants does workload W violate?
 - Which records should bob be allowed to see?
- **Library (Go), sidecar/host-level daemon**
 - Policy and data are kept in-memory
 - Zero decision-time dependencies
- **Management APIs for control & observability**
 - Bundle service API for sending policy & data to OPA
 - Status service API for receiving status from OPA
 - Log service API for receiving audit log from OPA
- **Tooling to build, test, and debug policy**
 - opa run, opa test, opa fmt, opa deps, opa check, etc.
 - VS Code plugin, Tracing, Profiling, etc.



Use Case Deep Dive

OPA: Unified Policy Across the Stack



Kubernetes Policy Example

[OPA Playground \(with flat policy\)](#)
[OPA Playground \(with rich policy\)](#)
[OPA Playground \(no policy\)](#)

JSON/YAML from Kubernetes

```
apiVersion: admission.k8s.io/v1beta1
kind: AdmissionReview
request:
  kind:
    group: ''
    kind: Pod
    version: v1
  namespace: opa
  object:
    metadata:
      labels:
        app: nginx
        name: nginx
        namespace: opa
    spec:
      containers:
        - image: nginx
          imagePullPolicy: Always
          name: nginx
          volumeMounts:
            - mountPath: "/var/run/serviceaccount"
              name: default-token-tm9v8
              readOnly: true
  operation: CREATE
```

OPA Policy: All images come from a trusted registry

```
package kubernetes.admission

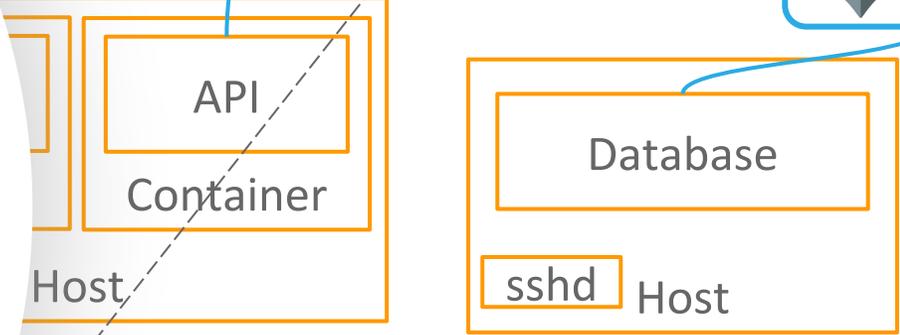
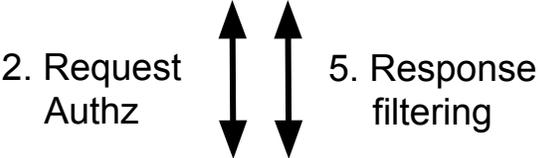
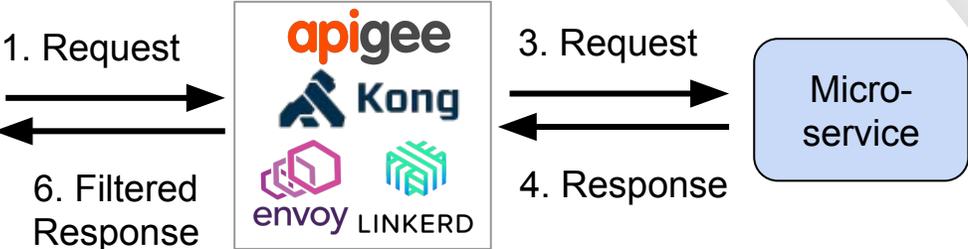
deny[msg] {
  input.request.kind.kind == "Pod"
  some i
  image := input.request.object.spec.containers[i].image
  not startswith(image, "hooli.com")
  msg := sprintf("image comes from bad registry: %v", [image])
}
```

openpolicyagent.org



OPA: Unified Policy Across the Stack

Network Proxy Integration



Envoy Policy Example

[OPA Playground \(with policy\)](#)

[OPA Playground \(no policy\)](#)

JSON/YAML from Envoy

```
parsed_path: ["api", "v1", "products"]
attributes:
  source:
    address:
      Address:
        SocketAddress:
          address: "172.17.0.10"
          PortSpecifier:
            PortValue: 36472
    destination:
      address:
        Address:
          SocketAddress:
            address: "172.17.0.17"
            PortSpecifier:
              PortValue: 9080
  request:
    http:
      id: 13359530607844510314
      method: GET
      headers: ...
      path: "/api/v1/products"
      host: "192.168.99.100:31380"
      protocol: "HTTP/1.1"
```

OPA Policy: Allow all GET and some PUT

```
package envoy.authz

# everyone can read everything
permit {
  input.attributes.request.http.method == "GET"
}

# writes dependent on source
permit {
  input.attributes.request.http.method == "PUT"
  input.parsed_path = ["v1", "deployment", x]
  src := input.attributes.source.address.Address.SocketAddress.address
  net.cidr_contains("172.28.0.0/16", src)
}
```

openpolicyagent.org



New and Future Features

Future: Automated Complexity Analysis

```
package kubernetes.admission
```

```
deny[msg] {  
  input.request.kind.kind == "Pod"  
  some i  
  image := input.request.object.spec.containers[i].image  
  not startswith(image, "hooli.com")  
  msg := sprintf("image comes from bad registry: %v", [image])  
}
```

Complexity: $O(\text{input})$

Before: Application owners, K8s admins, microservice integrators can delegate the authoring of policy to other people.

After: Admins can also dictate how rich the policies are that those other people can write.

```
package envoy.authz
```

```
# everyone can read everything  
permit {  
  input.attributes.request.http.method == "GET"  
}  
  
# writes dependent on source  
permit {  
  input.attributes.request.http.method == "PUT"  
  input.parsed_path = ["v1", "deployment", x]  
  src := input.attributes.source.address.Address.SocketAddress.address  
  net.cidr_contains("172.28.0.0/16", src)  
}
```

Complexity: $O(1)$

What's New?

- WebAssembly compiler
- Demo: Partial Evaluation Improvements
 - With keyword
 - Comprehensions

OPA Summit



[Overview](#) [Agenda](#) [Sponsors](#)

Agenda

- 11:30am ○ Lunch & Coffee
- Brief introductions
- OPA in Practice: From Angular to OPA in Chef Automate [Session Abstract](#)
- OPA at Scale: How Pinterest Manages Policy Distribution [Session Abstract](#)
- TripAdvisor: Building a Testing Framework for Integrating OPA into K8s [Session Abstract](#)
- Deploying OPA at Atlassian [Session Abstract](#)
- High Performance Rego at Scale with Fugue [Session Abstract](#)
- Policy Enabled Kubernetes and CICD [Session Abstract](#)
- Additional end-user sessions TBA
- 4:55pm - 5:00pm ○ Wrap-up and prep for evening reception
- 6:00pm ○ Nearby evening reception for drinks and games

Kubecon San Diego

Tuesday, November 19

- 10:55am [OPA Introduction & Community Update](#) - Rita Zhang, Microsoft & Patrick East, Styra
- 11:20am [Meet the Maintainer: OPA, Ash Narkar & Patrick East](#)
- 11:50am [Applying Policy Throughout The Application Lifecycle with Open Policy Agent](#) - Gareth Rushgrove, Snyk (Description: opa)
- 2:25pm [Managing Helm Deployments with Gitops at CERN](#) - Ricardo Rocha, CERN (Description: opa)
- [Enforcing Automatic mTLS With Linkerd and OPA Gatekeeper](#) - Ivan Sim, Buoyant & Rita Zhang, Microsoft
- 3:20pm [Meet the Maintainer: OPA, Torin Sandall](#)
- [Walls Within Walls: What if Your Attacker Knows Parkour?](#) - Tim Allclair & Greg Castle, Google (Description: opa)

Wednesday, November 20

- 11:20am [Meet the Maintainer: OPA, Ash Narkar & Patrick East](#)
- 11:50am [From Brownfield to Greenfield: Istio Service Mesh Journey at Freddie Mac](#) - Shiram Rajagopalan, Tetrate & Lixun Qi, Freddie Mac (Speakers: opa)
- 12:20pm [Meet the Maintainer: OPA, Torin Sandall](#)
- 1:20pm [Meet the Maintainer: OPA, Rita Zhang & Max Symthe](#)
- 5:20pm [OPA Deep Dive](#) - Tim Hinrichs, Styra & Torin Sandall, Styra

Thursday, November 21

- 10:55am [How Yelp Moved Security From the App to the Mesh with Envoy and OPA](#) - Daniel Popescu, Yelp & Ben Plotnick, Cruise
- 11:20am [Meet the Maintainer: OPA, Ash Narkar & Patrick East](#)
- 12:20pm [Meet the Maintainer: OPA, Torin Sandall](#)
- 1:20pm [Meet the Maintainer: OPA, Rita Zhang & Max Symthe](#)
- 3:20pm [Building a Medical AI with Kubernetes and Kubeflow](#) - Jeremie Vallee, Babylon Health (Description: opa)
- 4:25pm [Enforcing Service Mesh Structure using OPA Gatekeeper](#) - Sandeep Parikh, Google
- 5:20pm [Kubernetes Policy Enforcement Using OPA At Goldman Sachs](#) - Miguel Uzcategui, Goldman Sachs & Tim Hinrichs, Styra

Q&A

Torin Sandall

Engineer at Styra
Co-creator of OPA



@tsandall on OPA
@sometorin

Tim Hinrichs

Co-founder & CTO at Styra
Co-creator of OPA



@tim on OPA
@thinrichs