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
@tlhinrichs

Ash Narkar

Software Engineer at Styra
Maintainer of OPA

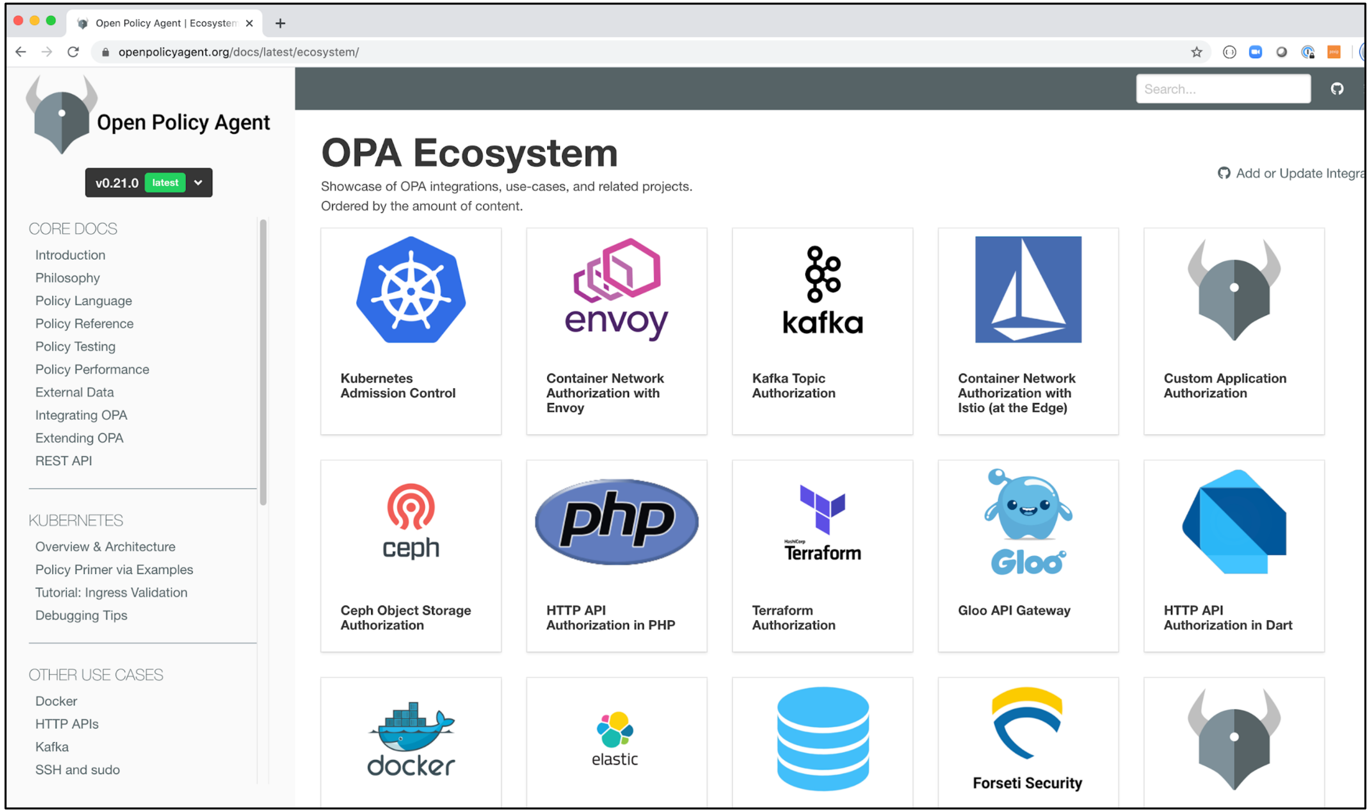
@ash on OPA
@ashtalk

openpolicyagent.org

Agenda

- OPA Overview
- API Authorization Deep Dive
- New and Future Features
- Subproject updates

OPA Integration Index



Open Policy Agent | Ecosystem

openpolicyagent.org/docs/latest/ecosystem/

Search...

Open Policy Agent

v0.21.0 latest

CORE DOCS
- Introduction
- Philosophy
- Policy Language
- Policy Reference
- Policy Testing
- Policy Performance
- 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

OPA Ecosystem

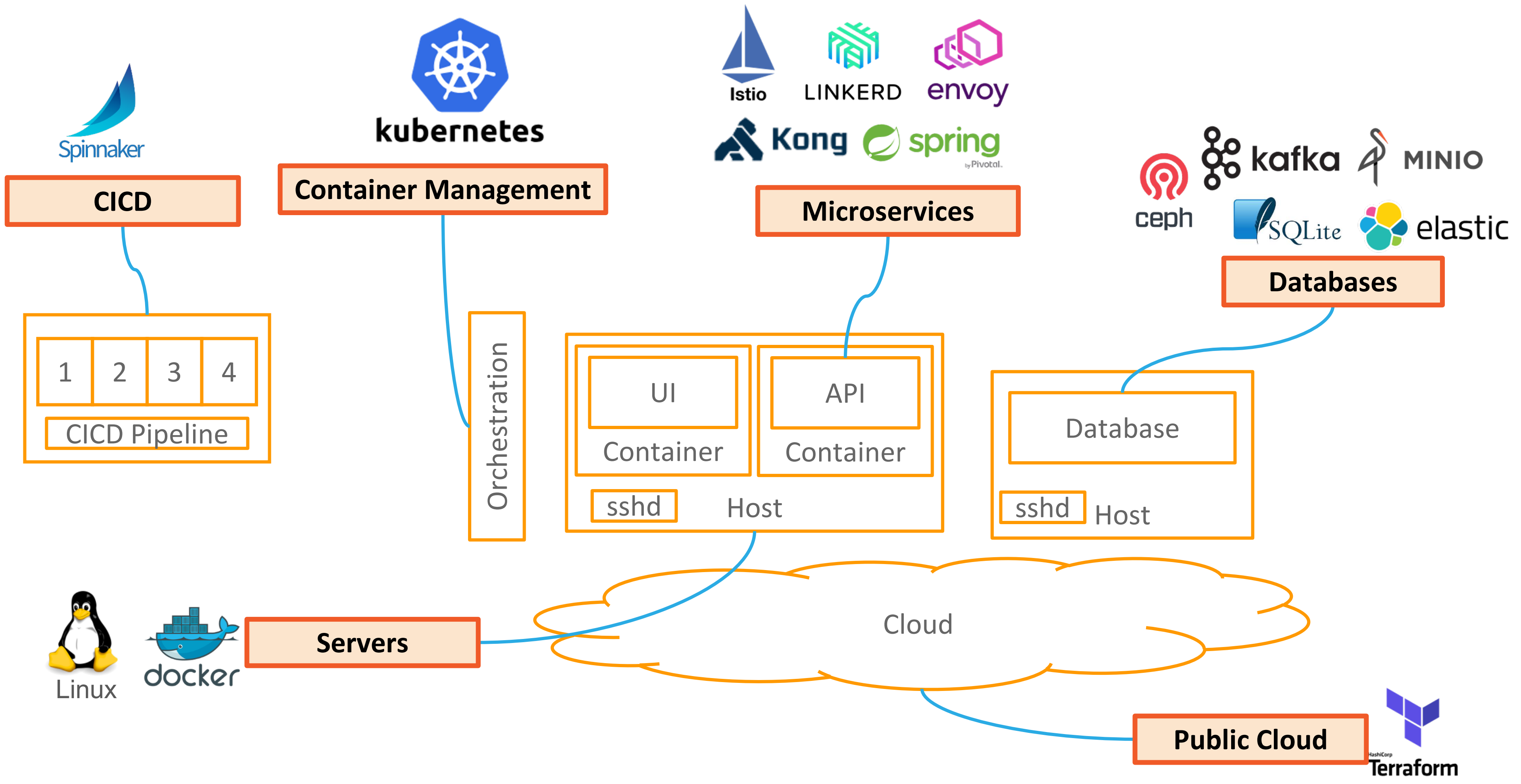
Add or Update Integra

Showcase of OPA integrations, use-cases, and related projects.
Ordered by the amount of content.

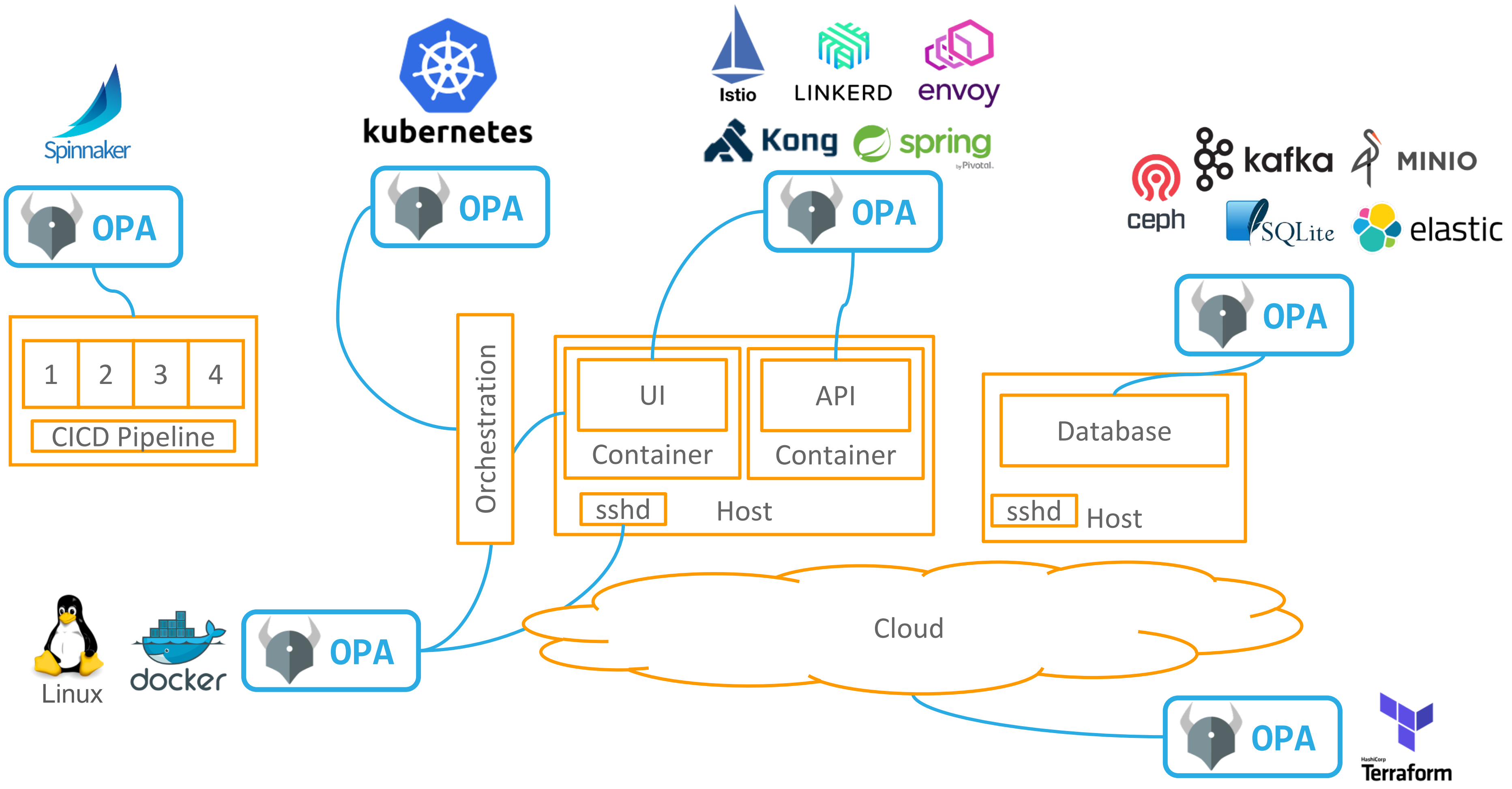
- Kubernetes Admission Control
- Container Network Authorization with Envoy
- Kafka Topic Authorization
- Container Network Authorization with Istio (at the Edge)
- Custom Application Authorization
- Ceph Object Storage Authorization
- HTTP API Authorization in PHP
- Terraform Authorization
- Gloo API Gateway
- HTTP API Authorization in Dart
- docker
- elastic
- Forseti Security

openpolicyagent.org

Cloud-native Diversity/Dynamism Make Policy Management Challenging

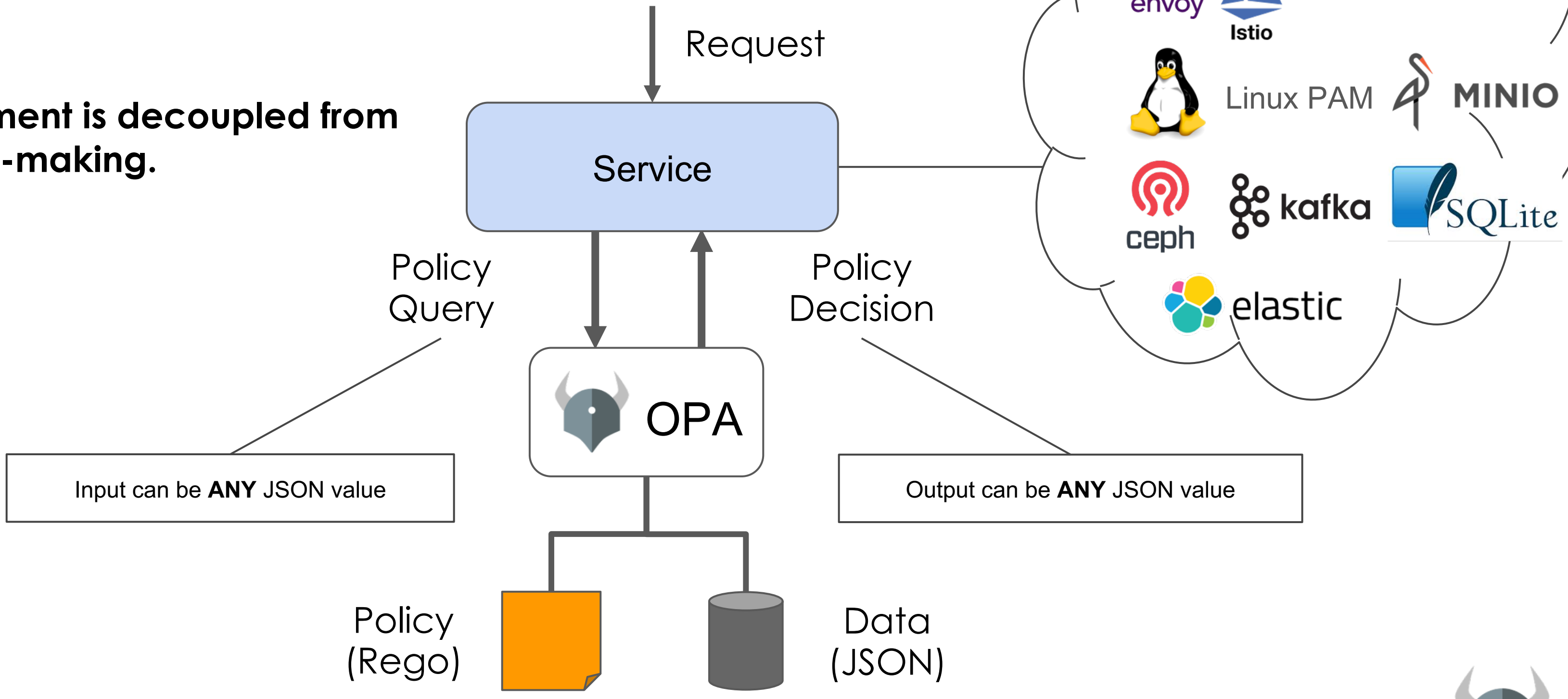


OPA: Unified Policy Across the Stack



OPA: General-purpose Policy Engine

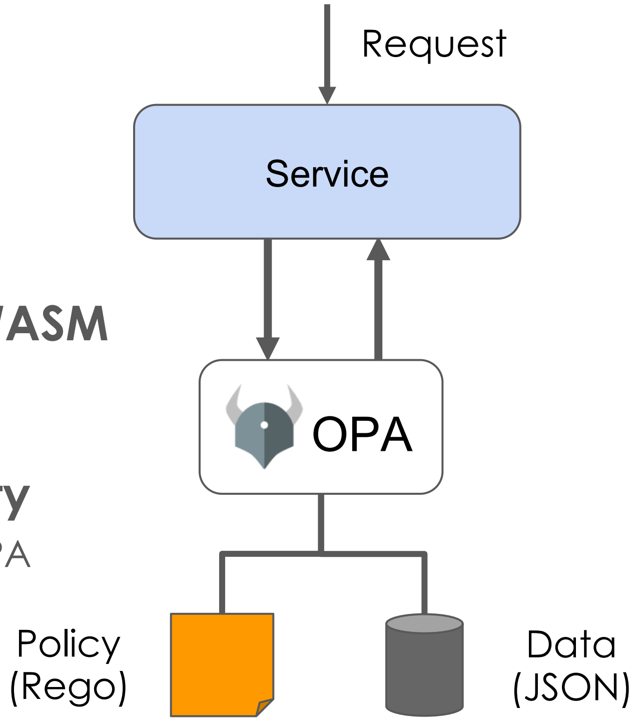
Enforcement is decoupled from decision-making.



openpolicyagent.org

OPA: Policy-as-code

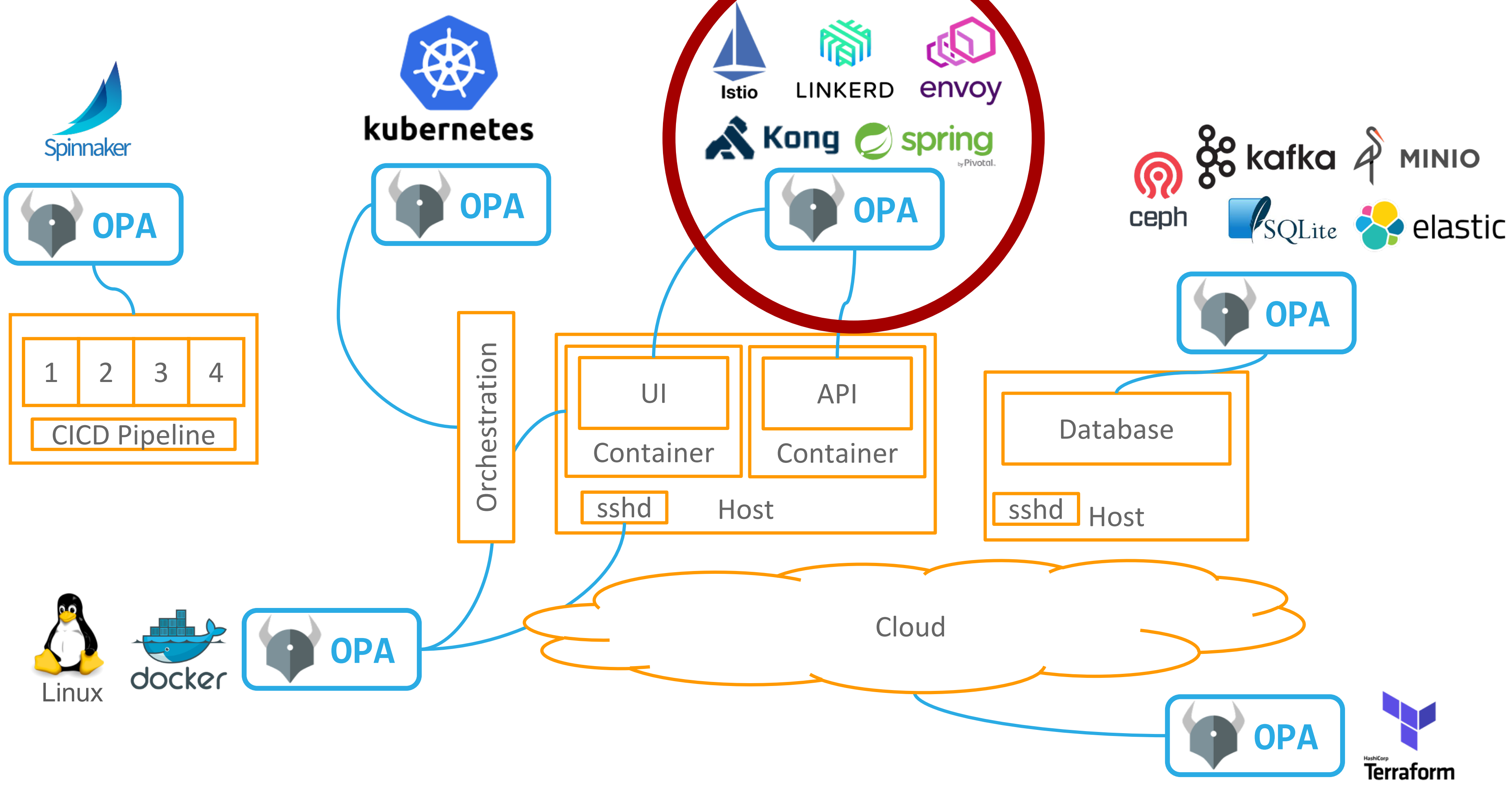
- **Declarative Policy Language (Rego)**
  - Can user X do operation Y on resource Z?
  - What invariants does workload W violate?
  - Which records should bob be allowed to see?
- **Library (Go), sidecar/host-level daemon, WASM**
  - Policy and data are kept in-memory
  - Zero decision-time dependencies
- **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.



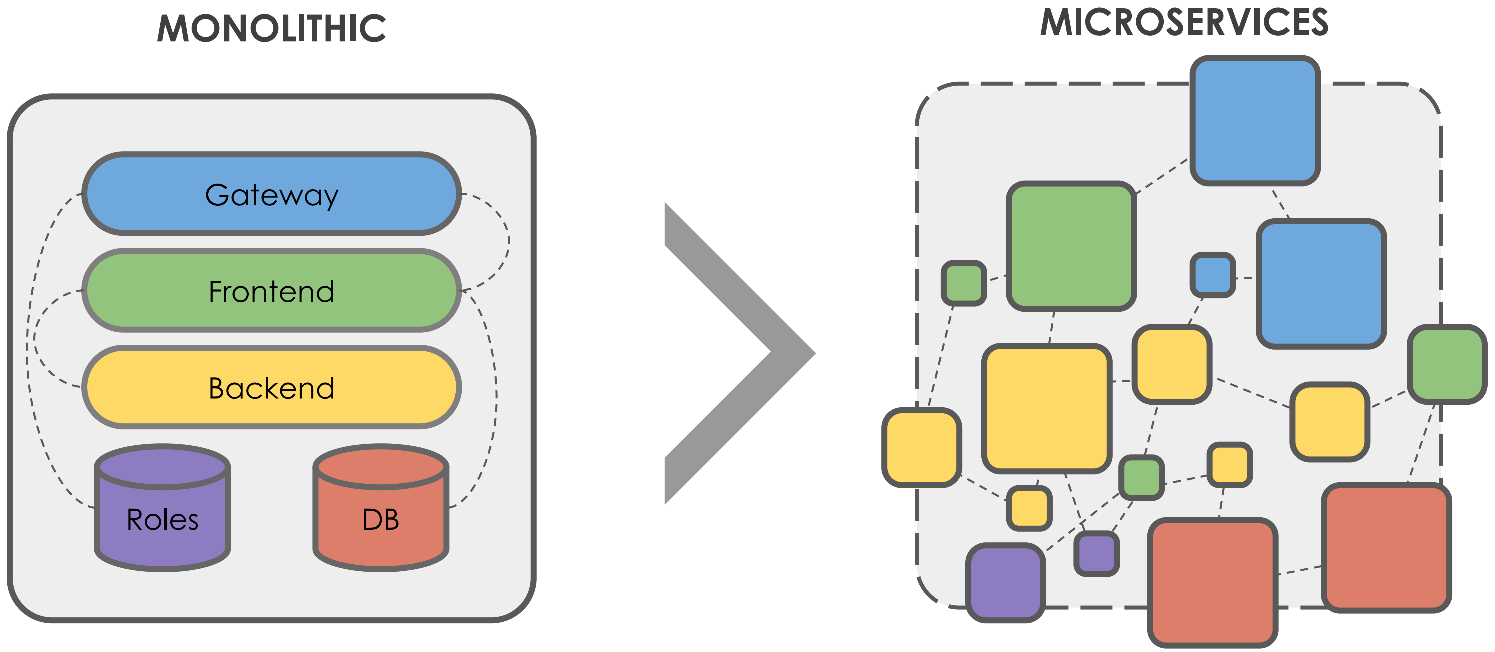
openpolicyagent.org

API Authorization Deep Dive

OPA: Unified Policy Across the Stack

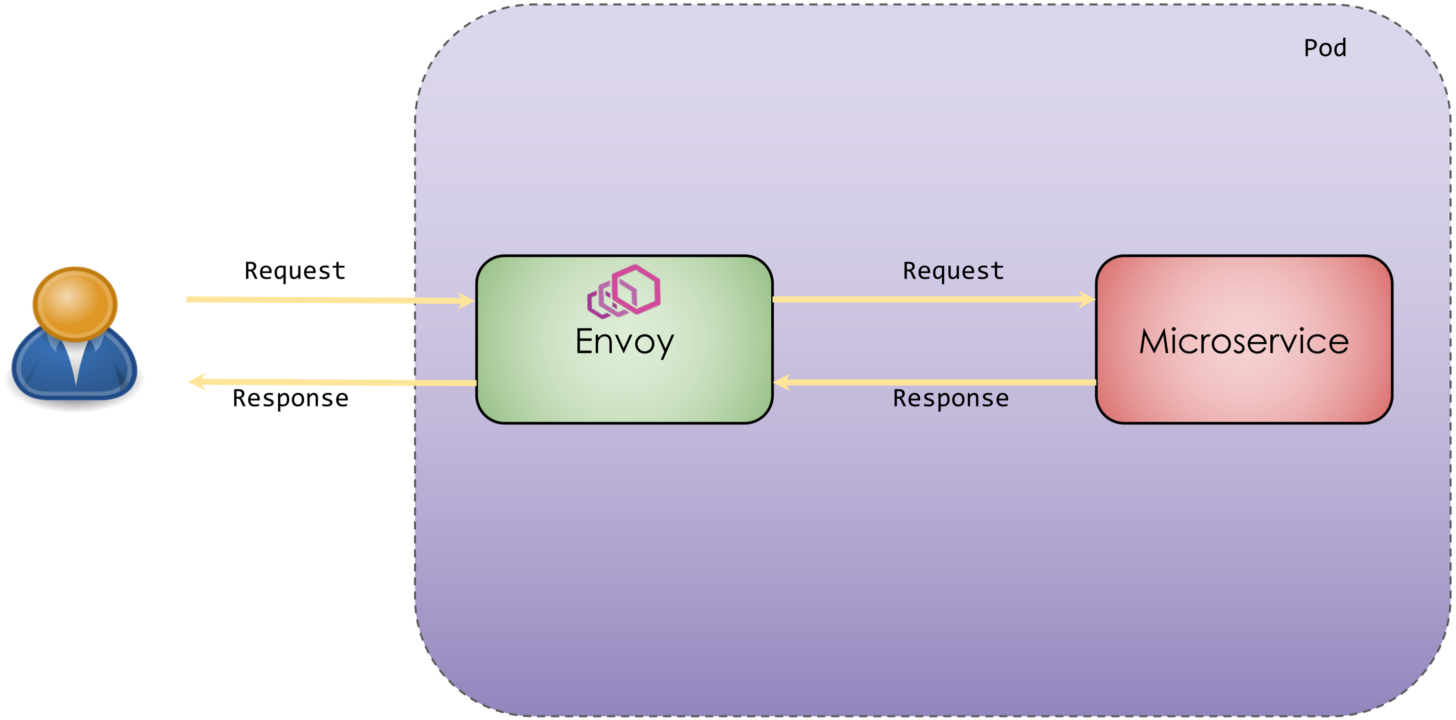


API Authorization with Microservices

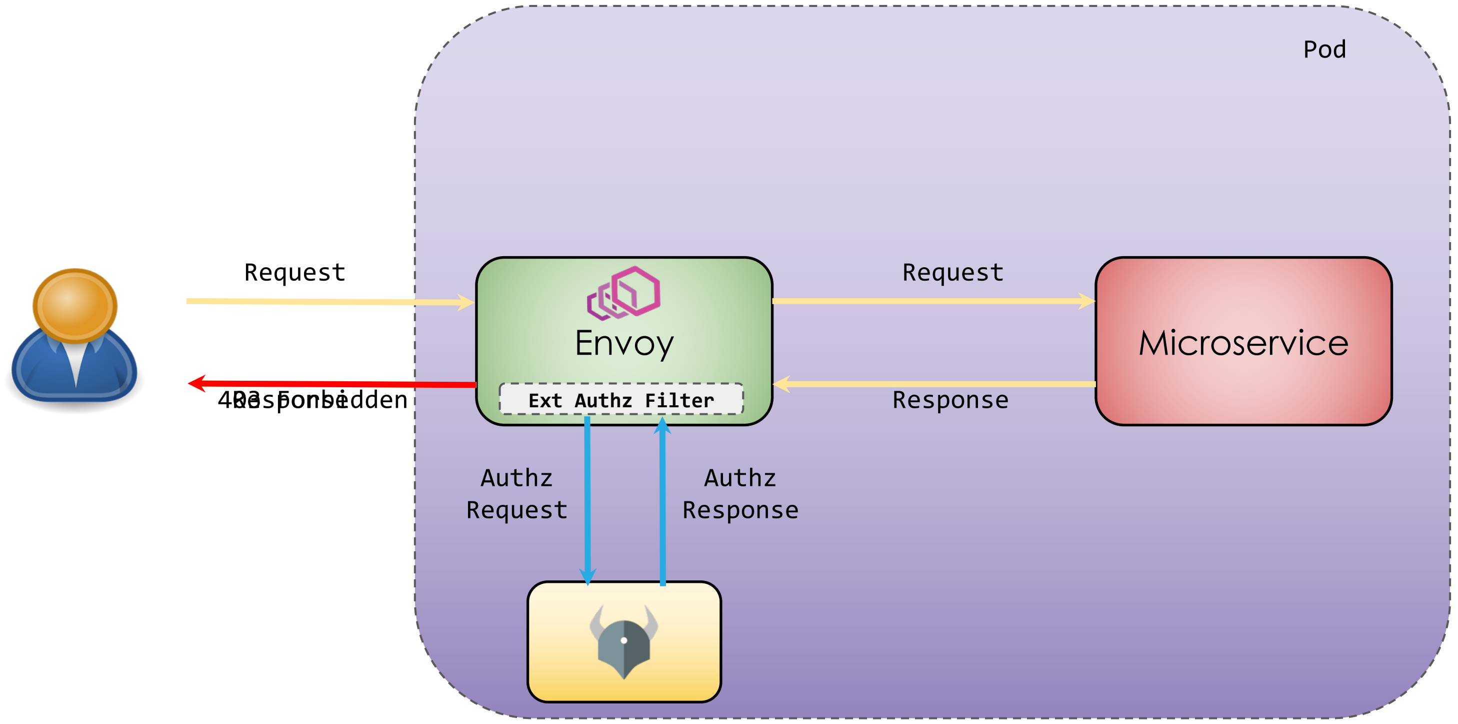


- Availability and Performance
- Collaboration
- Security

Request Flow - Envoy



Request Flow - OPA & Envoy



New and Future Features

New: Web Assembly Update

- WebAssembly (abbreviated Wasm) is a binary instruction format for a stack-based virtual machine
- Compile Rego to Wasm and package into a Bundle

```
# build a WASM bundle
opa build policy.rego -e example/allow -t wasm
```

- Built-in Function coverage
  - arithmetic
  - set
  - array
  - type checking
  - string (except sprintf)

New: Benchmarking Tool

- `opa bench` and `opa test --bench` commands for benchmarking policy evaluation
- Sample Usage:

```
# benchmark a single query
$ opa bench --data rbac.rego 'data.rbac.allow'

# benchmark unit tests
$ opa test -v --bench ./rbac.rego ./rbac_test.rego
```

Sample Output:

```
+-------------------------------------------+------------+
| samples                                   |      27295 |
| ns/op                                     |      45032 |
| B/op                                      |      20977 |
| allocs/op                                 |        382 |
| histogram_timer_rego_query_eval_ns_stddev |      25568 |
| histogram_timer_rego_query_eval_ns_99.9%  |     335906 |
| histogram_timer_rego_query_eval_ns_99.99% |     336493 |
| histogram_timer_rego_query_eval_ns_mean   |      40355 |
| histogram_timer_rego_query_eval_ns_median |      35846 |
| histogram_timer_rego_query_eval_ns_99%    |     133936 |
| histogram_timer_rego_query_eval_ns_90%    |      44780 |
| histogram_timer_rego_query_eval_ns_95%    |      50815 |
| histogram_timer_rego_query_eval_ns_min    |      31284 |
| histogram_timer_rego_query_eval_ns_max    |     336493 |
| histogram_timer_rego_query_eval_ns_75%    |      38254 |
| histogram_timer_rego_query_eval_ns_count  |      27295 |
+-------------------------------------------+------------+
```

New: Decision log mutation

- Decision logs contain policy input, decision, query etc.
- Input as well as policy decision may hold sensitive data
  - For example, JWT passed as input to OPA
- OPA now supports updating and adding information to decision logs

```
package system.log

# always upsert, no conditions in rule body
mask[{"op": "upsert", "path": "/input/password", "value": x}] {
  x := "**REDACTED**"
}
```

```
{
  "decision_id": "b4638167-7fcb-4bc7-9e80-31f5f87cb738",
  "masked": [
    "/input/password"
  ],
  "input": {
    "name": "bob",
    "resource": "user",
    "password": "**REDACTED**"
  },
------------------------- 8< -------------------------
  "path": "system/main",
  "requested_by": "127.0.0.1:36412",
  "result": true,
  "timestamp": "2019-06-03T20:07:16.939402185Z"
}
```

- Thanks to Domingo Kiser at Frontdoor for implementing this feature !

New: Additional features

Partial Evaluation enhancements

data.acls is known

```
allow = true {
  acl := data.acls[_]
  input.action == acl.action
  input.resource == acl.resource
  input.user == acl.user
}
```

input is unknown

Enhanced subcommand: opa build

```
# build an OPA bundle out of the current directory
opa build -b .

# build an OPA bundle and optimize for example/allow
opa build -b . -e example/allow -O 1

# build a WASM bundle
opa build policy.rego -e example/allow -t wasm
```

New Parser for Rego

```
name                               old time/op    new time/op    delta
ParseModuleRulesBase/1-16            210µs ± 1%       4µs ± 1%  -98.02%  (p=0.008 n=5+5)
ParseModuleRulesBase/10-16          1.39ms ± 1%    0.03ms ± 0%  -97.93%  (p=0.008 n=5+5)
ParseModuleRulesBase/100-16         13.5ms ± 1%     0.3ms ± 1%  -97.93%  (p=0.008 n=5+5)
ParseModuleRulesBase/1000-16         148ms ± 5%       3ms ± 6%  -97.77%  (p=0.008 n=5+5)
ParseS                                                                              5+5)
ParseS                                                                              5+5)
ParseS                                                                              5+5)
ParseS                                                                              5+5)
ParseS                                                                              5+5)
ParseStatementSimpleArray/1000-16   42.0ms ± 3%     0.5ms ± 4%  -98.70%  (p=0.008 n=5+5)
ParseStatementNestedObjects/1x1-16   233µs ± 6%       4µs ± 3%  -98.49%  (p=0.008 n=5+5)
ParseStatementNestedObjects/5x1-16   514µs ± 0%       9µs ± 4%  -98.33%  (p=0.008 n=5+5)
ParseStatementNestedObjects/10x1-16  911µs ± 5%      14µs ± 5%  -98.46%  (p=0.008 n=5+5)
ParseStatementNestedObjects/1x5-16  4.24ms ± 1%    0.01ms ± 1%  -99.82%  (p=0.016 n=4+5)
ParseStatementNestedObjects/1x10-16  138ms ± 1%       0ms ± 1%  -99.99%  (p=0.008 n=5+5)
ParseStatementNestedObjects/5x5-16   714ms ± 0%       5ms ± 5%  -99.26%  (p=0.016 n=4+5)
```

100% faster across nearly all benchmarks.
Arbitrarily faster in contrived cases.

Optimization for Group-by idioms

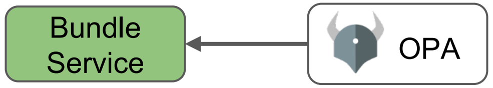
```
exposed_ports_by_interface := {intf: ports |
  some i
  intf := input.exposed[i].interface
  ports := [port |
    some j
    input.exposed[j].interface == intf
    port := input.exposed[j].port
  ]
}
```

$O(n^2) \rightarrow O(n)$

openpolicyagent.org

Upcoming features

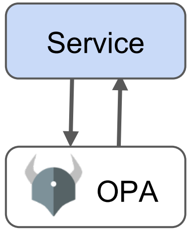
Digital signatures for bundle downloads



1. Download signed bundle
2. Verify integrity of bundle
3. Activate bundle

Ash Narkar (Styra), Ashish Tripathi (ANZ)

Always-on Tracing for Explanations



```
allow    → true
is_admin → false
is_get   → true
...
```

Patrick East (Styra)

IntelliJ plugin for OPA

- Evaluating queries
- Tracing execution
- Profiling performance
- Building bundles

Frankie Cerkvenik (Styra), Vincent Gramer (Indep), Asad Ali (Styra), Anders Eknert (Bisnode)

MongoDB integration

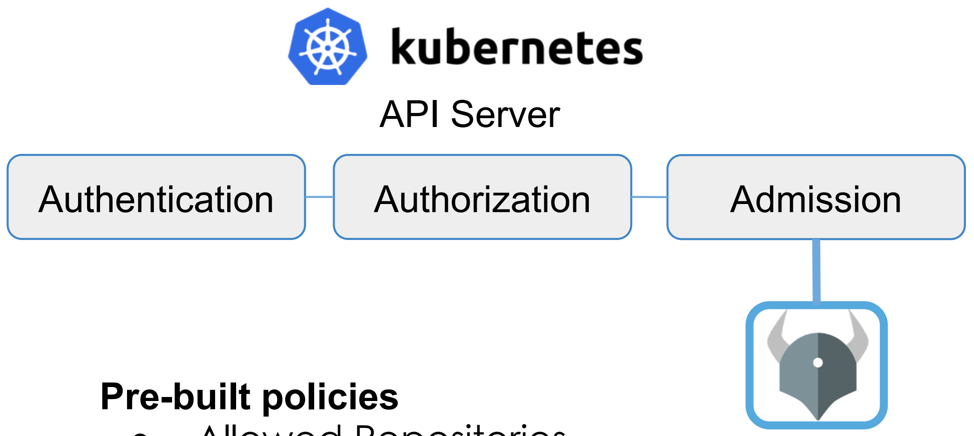
- Translating Rego to MongoDB query
- Support for basic relational operations like ==, !=, >, <

Vineeth Pothulapati (AquaSecurity via CommunityBridge), Ash Narkar (Styra)

Subproject Updates

Gatekeeper Update

Gatekeeper deploys OPA as an admission controller for Kubernetes.



New Features

- Metrics (e.g. violations, performance, ...)
- Semantic logging (logging is now JSON)
- CNCF Security review (success :))
- Standalone audit controller
- Pod Security Policies added to library
- HA support for webhook
- Stable constraint/template format
- Namespace Exclusion for constraints

Pre-built policies

- Allowed Repositories
- Container limits
- Container resource ratios
- Required labels
- Required probes
- HTTPS only
- Unique ingress hosts
- Unique service selector
- Pod Security Policies

openpolicyagent.org

Welcome Conftest as a new OPA project!

Conftest uses OPA to provide a user experience optimised for developers wanting to test all kinds of configuration files.

www.conftest.dev

```
$ conftest test deployment.yaml
FAIL - deployment.yaml - Containers must not run as root
FAIL - deployment.yaml - Deployments are not allowed

2 tests, 0 passed, 0 warnings, 2 failures
```

Inputs

Conftest parses lots of config formats into a structure OPA can act on.

- YAML
- JSON
- INI
- TOML
- HOCON
- HCL
- HCL1
- CUE
- Dockerfile
- EDN
- VCL
- XML

Outputs

Output results in various formats to make developer tools integration easier.

- User friendly
- JSON
- TAP
- JUnit XML

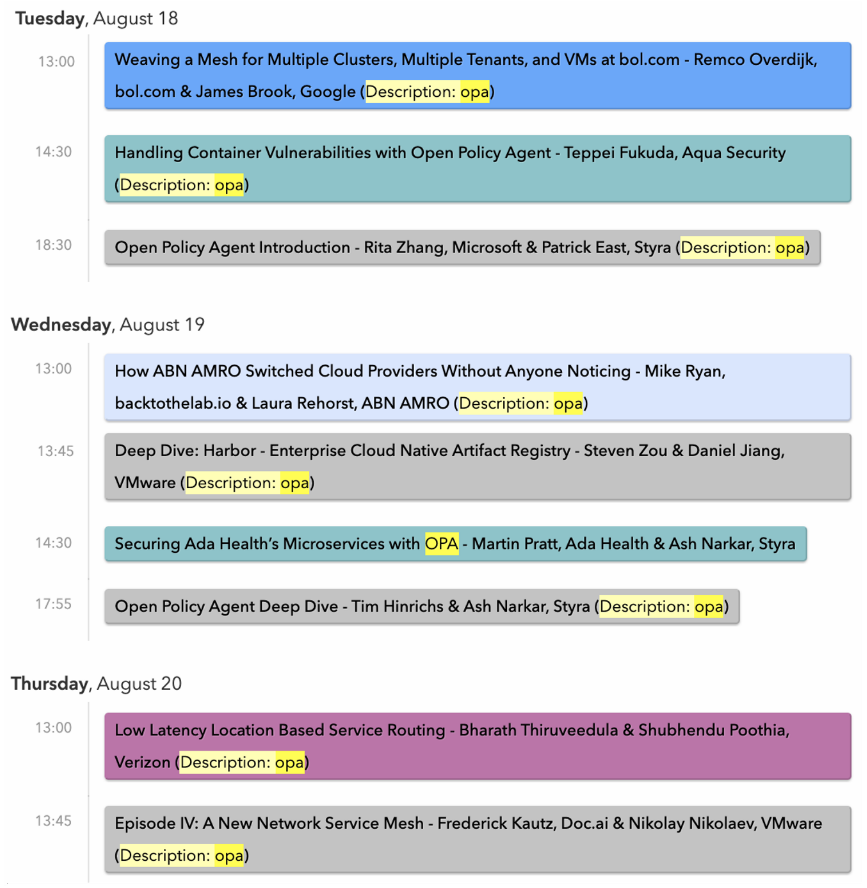
Integrations

Conftest has out-of-the-box integrations with popular CI/CD tooling.

circleci TEKTON GitHub Actions

openpolicyagent.org

Kubecon EU 2020 OPA talks



Tuesday, August 18

13:00 Weaving a Mesh for Multiple Clusters, Multiple Tenants, and VMs at bol.com - Remco Overdijk, bol.com & James Brook, Google (Description: opa)

14:30 Handling Container Vulnerabilities with Open Policy Agent - Teppei Fukuda, Aqua Security (Description: opa)

18:30 Open Policy Agent Introduction - Rita Zhang, Microsoft & Patrick East, Styra (Description: opa)

Wednesday, August 19

13:00 How ABN AMRO Switched Cloud Providers Without Anyone Noticing - Mike Ryan, backtothelab.io & Laura Rehorst, ABN AMRO (Description: opa)

13:45 Deep Dive: Harbor - Enterprise Cloud Native Artifact Registry - Steven Zou & Daniel Jiang, VMware (Description: opa)

14:30 Securing Ada Health's Microservices with OPA - Martin Pratt, Ada Health & Ash Narkar, Styra

17:55 Open Policy Agent Deep Dive - Tim Hinrichs & Ash Narkar, Styra (Description: opa)

Thursday, August 20

13:00 Low Latency Location Based Service Routing - Bharath Thiruveedula & Shubhendu Poothia, Verizon (Description: opa)

13:45 Episode IV: A New Network Service Mesh - Frederick Kautz, Doc.ai & Nikolay Nikolaev, VMware (Description: opa)

Q&A

Ash Narkar

Engineer at Styra
Maintainer of OPA

@ash on OPA
@ashtalk

Tim Hinrichs

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

@tim on OPA
@tlhinrichs

OPA Playground

Envoy Policy Example

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