Observability Day

EUROPE

Observability Day
EUROPE

What Is Going on Within My Network? A Subtle Introduction to Cilium Hubble

Shedrack Akintayo, Isovalent

What To Expect?

- Observability Considerations in Kubernetes
- eBPF
- What is Cilium?
- What is Cilium Hubble?
- Hubble Architecture
- eBPF + Cilium + Hubble = ❤️
- Hubble Components
- Hubble UI
- Demo
- Learning Resources

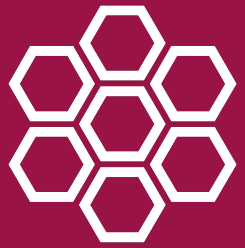
Observability Considerations in Kubernetes

- Application level Observability
  - Performance Metrics: Such as response times, transaction volumes, and error rates.
  - Logs
  - Traces

- Operations Level Observability
  - Cluster health
  - Resource utilization
  - Scaling Events

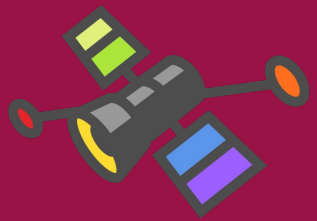
- Security Observability
  - Network traffic
  - Access logs

Observability Day
EUROPE



cilium

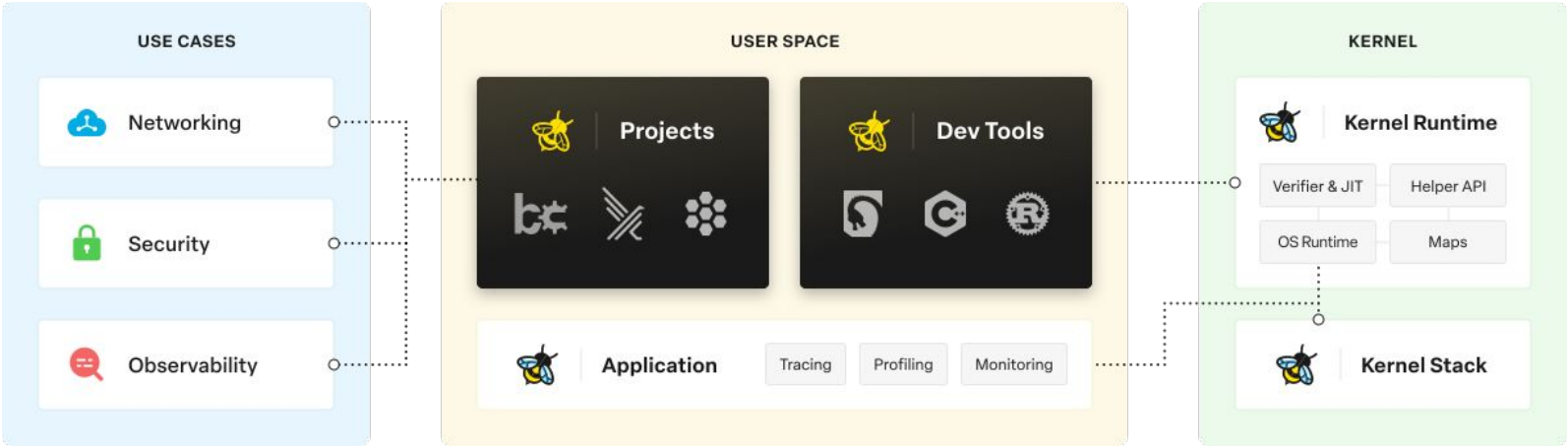
eBPF



Hubble

eBPF

eBPF is a technology used to **safely** and **efficiently** extend the capabilities of the kernel without requiring to **change kernel source code** or **load kernel modules.**



eBPF Benefits

eBPF enables the development of powerful new tools for the Cloud Native ecosystem that offers enhanced observability, efficient networking and improved performance management, revolutionizing the way cloud-native applications are built, run and operated.

- Reduced performance overhead
- Deep Visibility into an application
- Widely Available
  - Standardized across all modern Linux releases.

Examples of eBPF-based cloud native tools:

- Parca - Continuous Profiling
- Cilium - Networking
- Cilium Hubble - Observability
- Tetragon - Security

What is Cilium?

cilium

ISOVALENT

Creators of cilium and eBPF

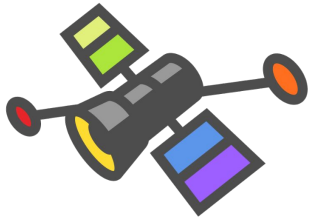
- Networking
- Security
- Observability
- Service Mesh & Ingress

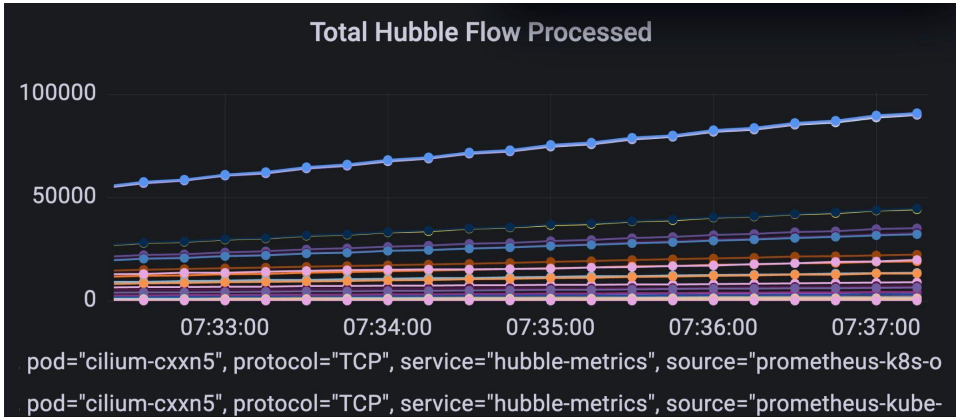
Cilium is an open source, **cloud native** solution for providing, securing, and observing network connectivity between workloads, fueled by the revolutionary Kernel technology **eBPF**

Foundation (Graduated)

Technology

What is Cilium Hubble?





- Observability layer for Cilium
- A fully distributed networking and security observability platform.
- Built on top of Cilium and eBPF
- **CLI** - command-line binary able to connect to either the gRPC API of Hubble Relay or the local server to retrieve flow events.
- **UI** - utilizes relay-based visibility to provide a graphical service dependency and connectivity map.
- Prometheus compatible
- Hubble datasource plugin for Grafana

What Can Hubble Do?

- Metrics collection including L7 metrics
- Logging of network flows
- Integrates with OpenTelemetry for distributed tracing.
- Visualization with Grafana

Where Can Hubble Help?

Service Dependencies & Communication Map

- What services are communicating with each other? How frequently? What does the service dependency graph look like?

- What HTTP calls are being made? What Kafka topics does a service consume from or produce to?

Where Can Hubble Help?

Network monitoring & alerting

- Is any network communication failing? Why is communication failing? Is it DNS? Is it an application or network problem? Is the communication broken on layer 4 (TCP) or layer 7 (HTTP)?

- Which services have experienced a DNS resolution problem in the last 5 minutes? Which services have experienced an interrupted TCP connection recently or have seen connections timing out? What is the rate of unanswered TCP SYN requests?

Where Can Hubble Help?

Application Monitoring

- What is the rate of 5xx or 4xx HTTP response codes for a particular service or across all clusters?

- What is the 95th and 99th percentile latency between HTTP requests and responses in my cluster? Which services are performing the worst? What is the latency between two services?

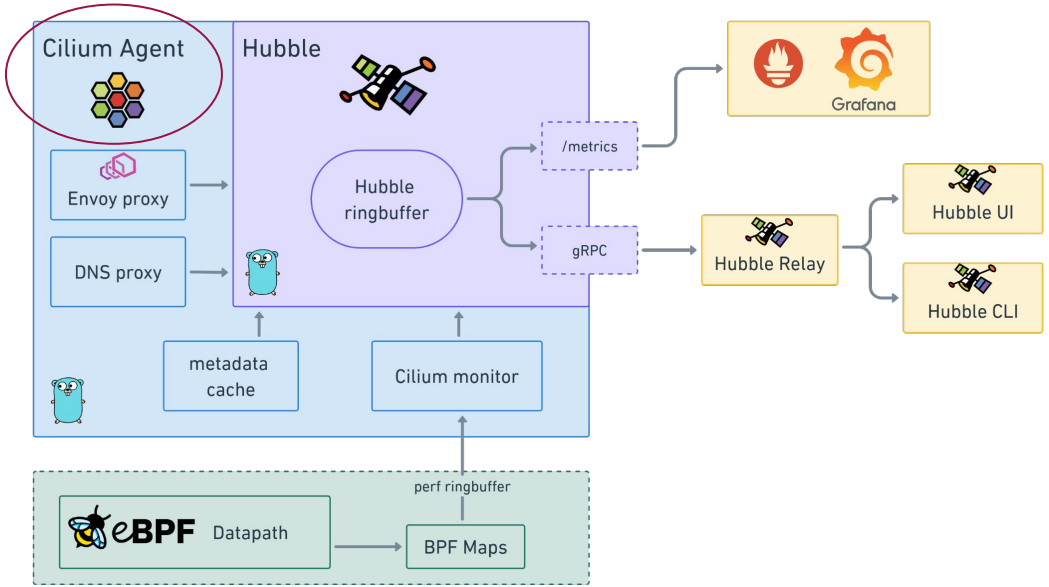
Where Can Hubble Help?

Security Observability

- Which services had connections blocked due to network policy? What services have been accessed from outside the cluster?

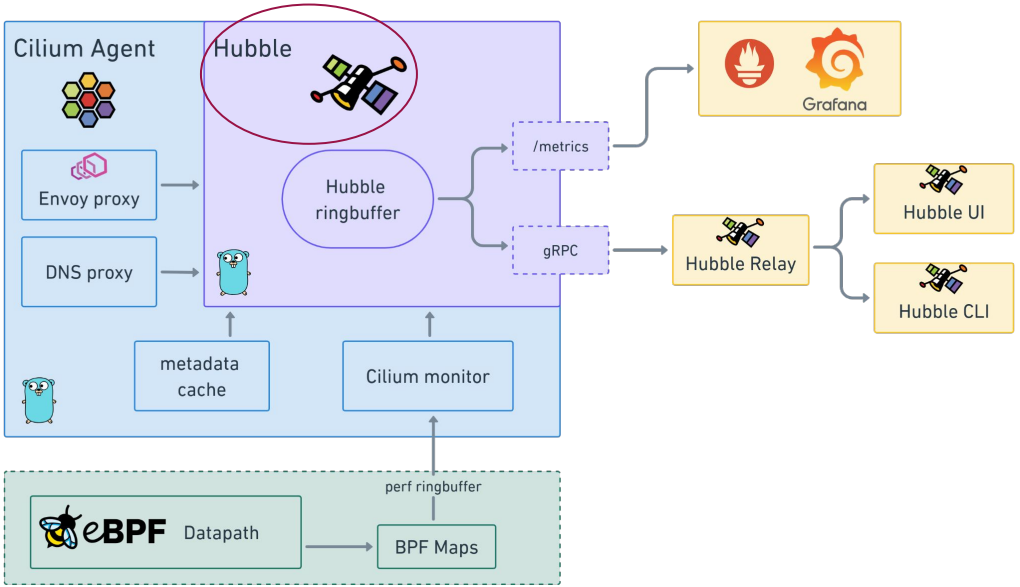
- Which services have resolved a particular DNS name?

Hubble Architecture



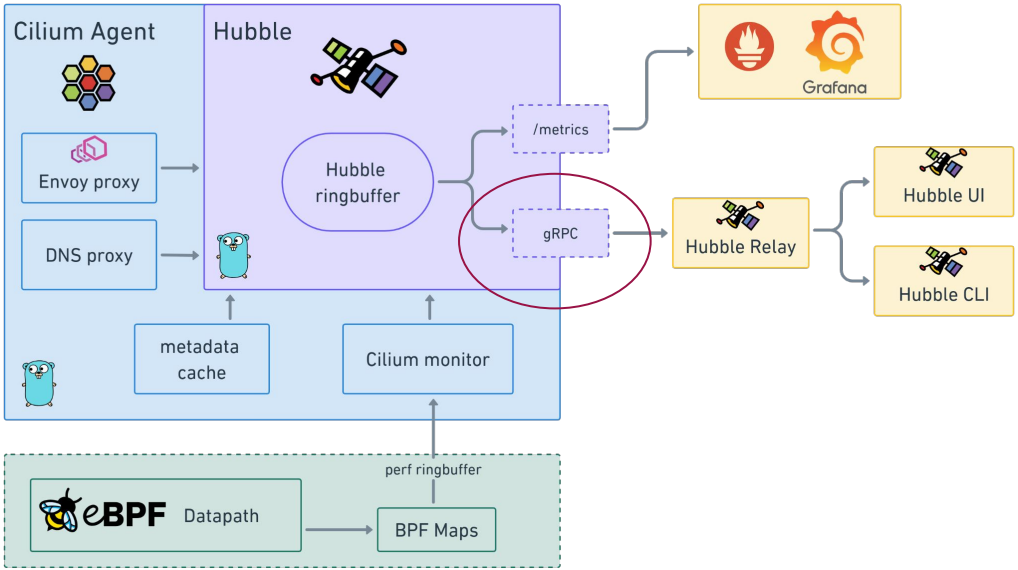
Cilium Agent – Runs the cilium-agent binary which acts as a CNI to manage connectivity, observability, and security for all CNI-managed Kubernetes pods.

Hubble Architecture



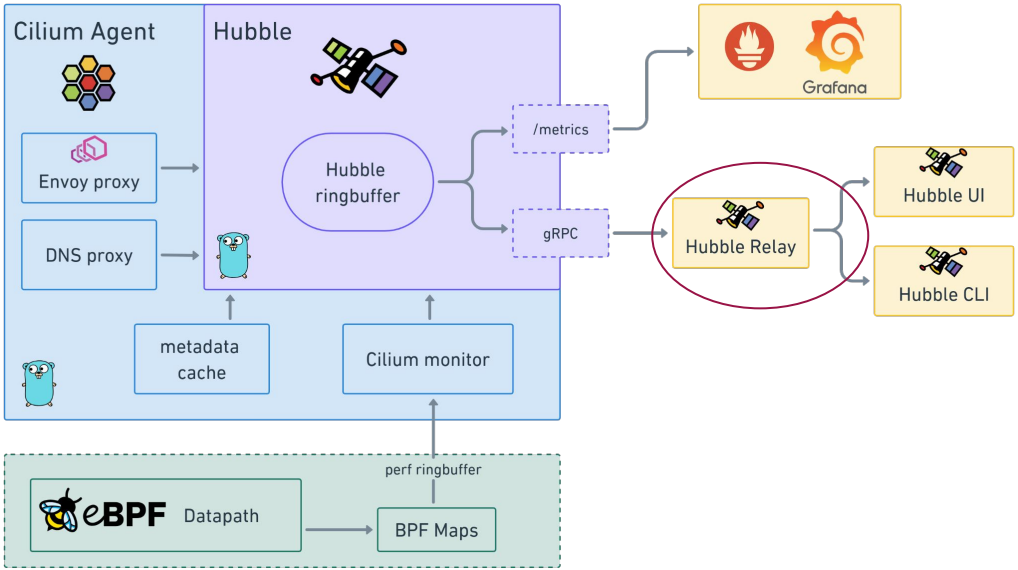
The **Hubble server** runs on each node and retrieves the eBPF-based visibility from Cilium.

Hubble Architecture



Hubble exposes **gRPC** services from the Cilium process that allows clients to receive flows and other type of data.

Hubble Architecture



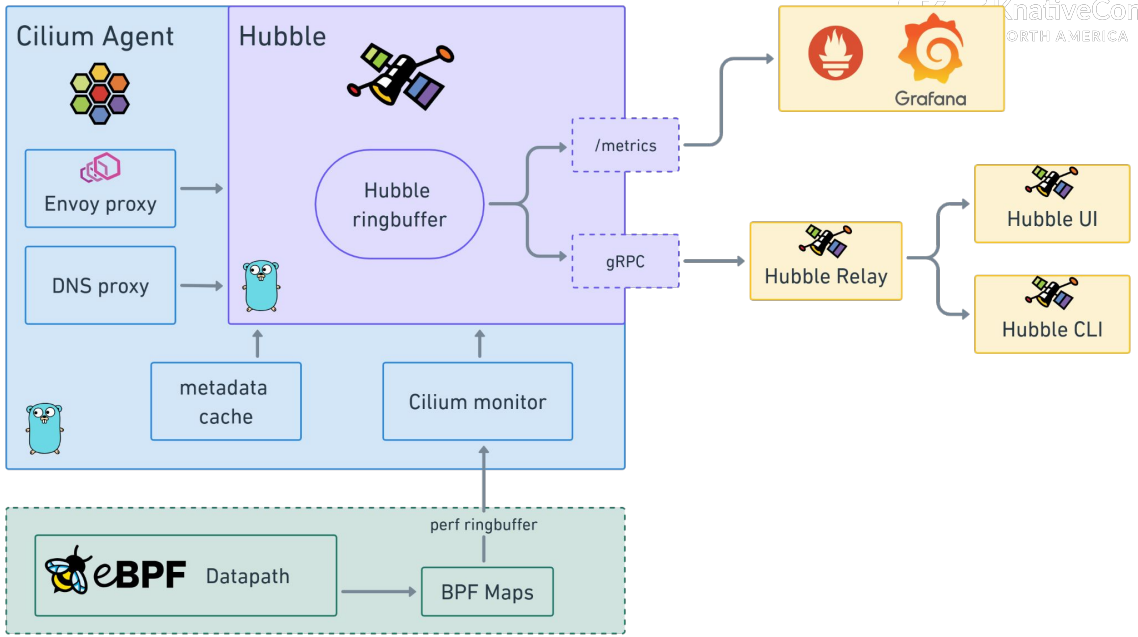
The **Hubble Relay** is there to provide full network visibility across the entire cluster – or across clusters.

eBPF + Cilium + Hubble = ❤️

1. Cilium Attaches eBPF programs to all Kubernetes pods

2. Cilium Subscribes to the Kubernetes API for updates on resources.

3. Cilium Converts Kubernetes resource information into BPF maps for eBPF datapath access.

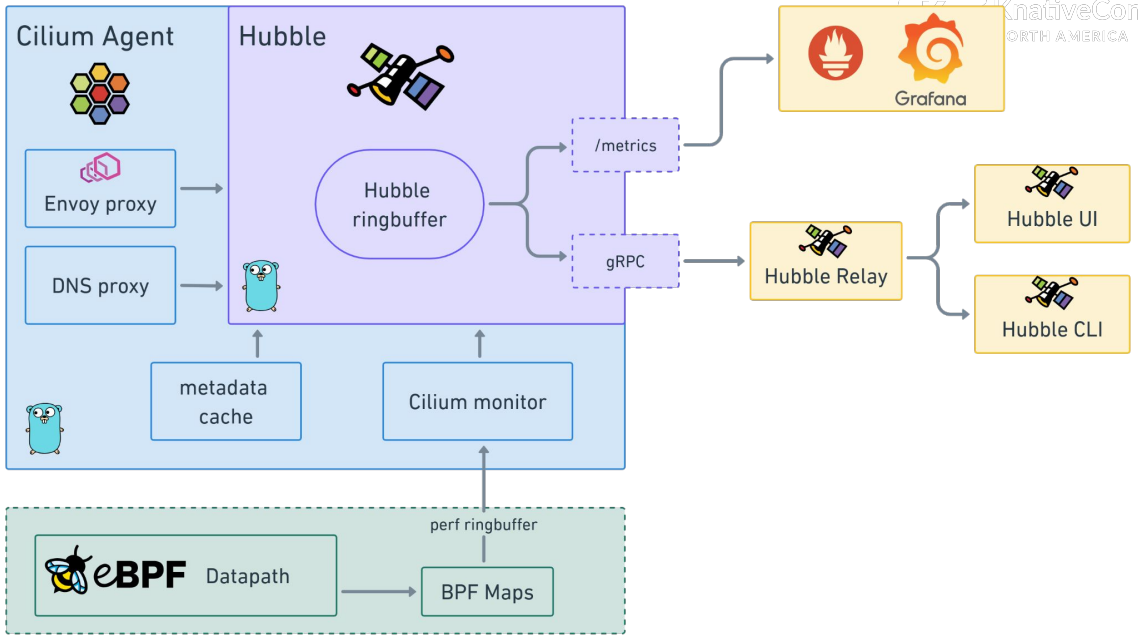


eBPF + Cilium + Hubble = ❤️

4. The eBPF datapath Inspects incoming connections to the pods.

5. The eBPF datapath emits trace and policy events, pushing these to a BPF map (acting as a ring buffer).

6. A Hubble instance, running inside the Cilium agent, reads from the BPF map.

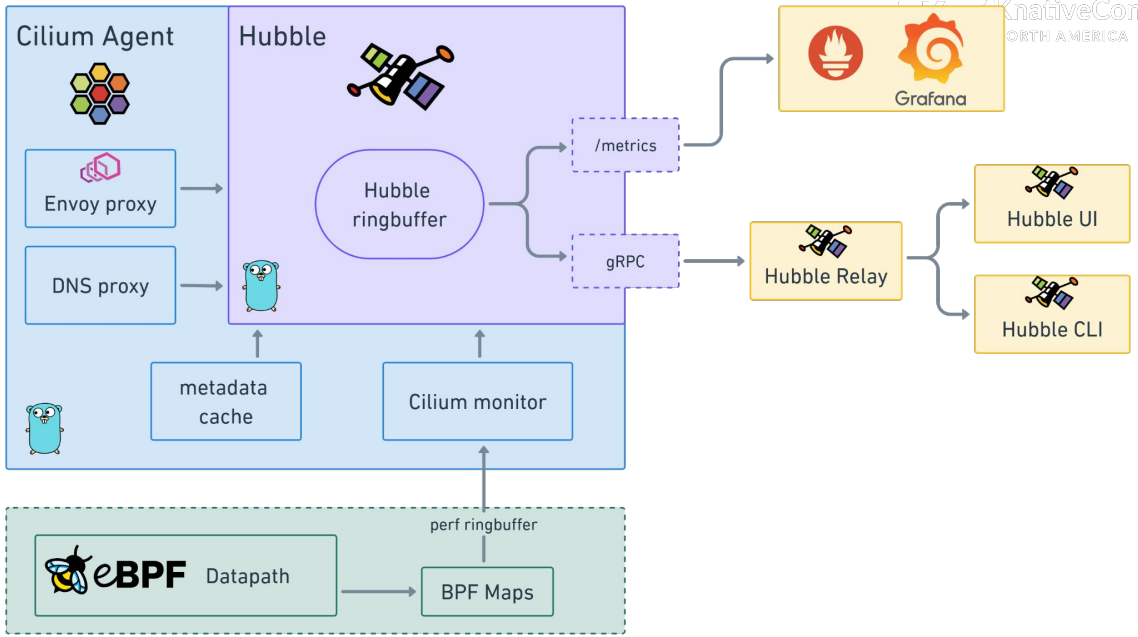


eBPF + Cilium + Hubble = ❤️

7. The Hubble Instance collects networking events, storing them in a historical buffer.

8. Hubble exposes the collected data through gRPC and metrics.

9. This data is accessible by other Hubble components, like the UI and Prometheus, for monitoring and analysis.



Hubble Components

Hubble CLI

- Detailed Flow Visibility
- Extensive Filtering
- JSON Output

Hubble UI

- Service Dependency maps
- Flow Display and Filtering
- Network Policy Viewer

Hubble Metrics

Built-in Metrics for operations and application monitoring

Hubble CLI

Observability Day EUROPE

```
root@server:~# hubble observe
Mar 12 09:15:55.790: 10.0.2.9:33312 (remote-node) <> monitoring/prometheus-k8s-operator-6865d4b8c9-fpd6b:10250 (ID:58913) to-overlay FORWARDED (TCP Flags: ACK, PSH)
Mar 12 09:15:55.799: 10.0.2.9:33326 (remote-node) <> monitoring/prometheus-k8s-operator-6865d4b8c9-fpd6b:10250 (ID:58913) to-overlay FORWARDED (TCP Flags: ACK, PSH)
Mar 12 09:15:55.799: 10.0.2.9:33326 (remote-node) <> monitoring/prometheus-k8s-operator-6865d4b8c9-fpd6b:10250 (ID:58913) to-overlay FORWARDED (TCP Flags: ACK, PSH)
Mar 12 09:15:55.800: 10.0.2.9:33326 (remote-node) <> monitoring/prometheus-k8s-operator-6865d4b8c9-fpd6b:10250 (ID:58913) to-overlay FORWARDED (TCP Flags: ACK, PSH)
Mar 12 09:15:55.838: 10.0.2.9:33312 (remote-node) <> monitoring/prometheus-k8s-operator-6865d4b8c9-fpd6b:10250 (ID:58913) to-overlay FORWARDED (TCP Flags: ACK)
Mar 12 09:15:55.854: 10.0.2.9:33326 (remote-node) <> monitoring/prometheus-k8s-operator-6865d4b8c9-fpd6b:10250 (ID:58913) to-overlay FORWARDED (TCP Flags: ACK)
Mar 12 09:15:58.057: 10.0.2.9:34888 (world) <> kube-system/hubble-ui-5dfbdddc49-766sz:8081 (ID:5302) to-overlay FORWARDED (TCP Flags: ACK)
Mar 12 09:15:58.553: 10.0.2.9:34888 (world) <> kube-system/hubble-ui-5dfbdddc49-766sz:8081 (ID:5302) to-overlay FORWARDED (TCP Flags: ACK)
Mar 12 09:15:58.553: 10.0.2.9:34888 (world) <> kube-system/hubble-ui-5dfbdddc49-766sz:8081 (ID:5302) to-overlay FORWARDED (TCP Flags: ACK)
Mar 12 09:15:58.553: 10.0.2.9:34888 (world) <> kube-system/hubble-ui-5dfbdddc49-766sz:8081 (ID:5302) to-overlay FORWARDED (TCP Flags: ACK)
Mar 12 09:15:58.553: 10.0.2.9:34888 (world) <> kube-system/hubble-ui-5dfbdddc49-766sz:8081 (ID:5302) to-overlay FORWARDED (TCP Flags: ACK)
Mar 12 09:15:58.554: 10.0.2.9:34888 (world) <> kube-system/hubble-ui-5dfbdddc49-766sz:8081 (ID:5302) to-overlay FORWARDED (TCP Flags: ACK)
Mar 12 09:16:01.111: 10.0.2.9:37074 (world) <> kube-system/hubble-ui-5dfbdddc49-766sz:8081 (ID:5302) to-overlay FORWARDED (TCP Flags: ACK)
Mar 12 09:16:01.730: tenant-jobs/jobs-app-kafka-0:36006 (ID:47789) -> tenant-jobs/jobs-app-zookeeper-0:2181 (ID:54269) to-overlay FORWARDED (TCP Flags: ACK, PSH)
Mar 12 09:16:01.731: tenant-jobs/jobs-app-kafka-0:36006 (ID:47789) <- tenant-jobs/jobs-app-zookeeper-0:2181 (ID:54269) to-endpoint FORWARDED (TCP Flags: ACK, PSH)
Mar 12 09:16:01.862: 10.0.1.118:53884 (host) -> tenant-jobs/jobs-app-entity-operator-79864b44c-27sfz:8081 (ID:16319) to-endpoint FORWARDED (TCP Flags: ACK)
Mar 12 09:16:01.990: tenant-jobs/jobs-app-entity-operator-79864b44c-27sfz:40978 (ID:16319) -> tenant-jobs/jobs-app-kafka-0:9091 (ID:47789) to-overlay FORWARDED (TCP Flags: ACK, PSH)
Mar 12 09:16:01.990: tenant-jobs/jobs-app-entity-operator-79864b44c-27sfz:40978 (ID:16319) -> tenant-jobs/jobs-app-kafka-0:9091 (ID:47789) to-endpoint FORWARDED (TCP Flags: ACK, PSH)
Mar 12 09:16:01.992: tenant-jobs/jobs-app-entity-operator-79864b44c-27sfz:40978 (ID:16319) <- tenant-jobs/jobs-app-kafka-0:9091 (ID:47789) to-overlay FORWARDED (TCP Flags: ACK, PSH)
Mar 12 09:16:01.992: tenant-jobs/jobs-app-entity-operator-79864b44c-27sfz:40978 (ID:16319) <- tenant-jobs/jobs-app-kafka-0:9091 (ID:47789) to-endpoint FORWARDED (TCP Flags: ACK, PSH)
Mar 12 09:16:02.026: kube-system/hubble-ui-5dfbdddc49-766sz:36454 (ID:5302) <- kube-system/hubble-relay-5469f799bf-282jx:80 (ID:20003) to-endpoint FORWARDED (TCP Flags: ACK)
Mar 12 09:16:02.026: kube-system/hubble-ui-5dfbdddc49-766sz:36454 (ID:5302) -> kube-system/hubble-relay-5469f799bf-282jx:4245 (ID:20003) to-endpoint FORWARDED (TCP Flags: ACK)
Mar 12 09:16:02.113: monitoring/prometheus-prometheus-k8s-prometheus-0:57132 (ID:50306) -> 172.18.0.4:9965 (kube-apiserver) to-stack FORWARDED (TCP Flags: ACK, PSH)
Mar 12 09:16:02.115: monitoring/prometheus-prometheus-k8s-prometheus-0:57132 (ID:50306) <- 172.18.0.4:9965 (kube-apiserver) to-endpoint FORWARDED (TCP Flags: ACK, PSH)
Mar 12 09:16:02.278: kube-system/hubble-ui-5dfbdddc49-766sz:56570 (ID:5302) <- kube-system/hubble-relay-5469f799bf-282jx:80 (ID:20003) to-endpoint FORWARDED (TCP Flags: ACK)
Mar 12 09:16:02.278: kube-system/hubble-ui-5dfbdddc49-766sz:56570 (ID:5302) -> kube-system/hubble-relay-5469f799bf-282jx:4245 (ID:20003) to-endpoint FORWARDED (TCP Flags: ACK)
```

Flow visibility provides visibility into flow information on the network and application protocol level. This enables visibility into individual TCP connections, DNS queries, HTTP requests, Kafka communication, and much more.

Hubble CLI

Observability Day EUROPE

```
root@server:~# hubble observe -n tenant-jobs --from-label=app=resumes
Mar 12 16:39:00.639: tenant-jobs/resumes-7c9d8fc76b-zvnqr:52084 (ID:5772) -> tenant-jobs/coreapi-58b9f5f67d-n54fm:9080 (ID:11699) to-endpoint FORWARDED (TCP Flags: ACK, PSH)
Mar 12 16:39:00.662: tenant-jobs/resumes-7c9d8fc76b-zvnqr:52084 (ID:5772) -> tenant-jobs/coreapi-58b9f5f67d-n54fm:9080 (ID:11699) to-endpoint FORWARDED (TCP Flags: ACK, FIN)
Mar 12 16:39:03.087: tenant-jobs/resumes-7c9d8fc76b-zvnqr:49754 (ID:5772) -> tenant-jobs/jobs-app-kafka-0:9092 (ID:36767) to-endpoint FORWARDED (TCP Flags: ACK)
Mar 12 16:39:04.998: tenant-jobs/resumes-7c9d8fc76b-zvnqr:47764 (ID:5772) -> tenant-jobs/jobs-app-kafka-0:9092 (ID:36767) to-endpoint FORWARDED (TCP Flags: ACK)
Mar 12 16:39:05.000: tenant-jobs/resumes-7c9d8fc76b-zvnqr:44033 (ID:5772) -> kube-system/coredns-5d78c9869d-4xtqz:53 (ID:10455) to-endpoint FORWARDED (UDP)
Mar 12 16:39:05.001: tenant-jobs/resumes-7c9d8fc76b-zvnqr:32784 (ID:5772) -> tenant-jobs/coreapi-58b9f5f67d-n54fm:9080 (ID:11699) to-endpoint FORWARDED (TCP Flags: SYN)
Mar 12 16:39:05.001: tenant-jobs/resumes-7c9d8fc76b-zvnqr:32784 (ID:5772) -> tenant-jobs/coreapi-58b9f5f67d-n54fm:9080 (ID:11699) to-endpoint FORWARDED (TCP Flags: ACK)
Mar 12 16:39:05.002: tenant-jobs/resumes-7c9d8fc76b-zvnqr:32784 (ID:5772) -> tenant-jobs/coreapi-58b9f5f67d-n54fm:9080 (ID:11699) to-endpoint FORWARDED (TCP Flags: ACK, PSH)
Mar 12 16:39:05.304: tenant-jobs/resumes-7c9d8fc76b-zvnqr:49766 (ID:5772) -> tenant-jobs/jobs-app-kafka-0:9092 (ID:36767) to-endpoint FORWARDED (TCP Flags: ACK, PSH)
Mar 12 16:39:05.304: tenant-jobs/resumes-7c9d8fc76b-zvnqr:32784 (ID:5772) -> tenant-jobs/coreapi-58b9f5f67d-n54fm:9080 (ID:11699) to-endpoint FORWARDED (TCP Flags: ACK, FIN)
Mar 12 16:39:06.244: tenant-jobs/resumes-7c9d8fc76b-zvnqr:59122 (ID:5772) -> kube-system/coredns-5d78c9869d-m7rn5:53 (ID:10455) to-endpoint FORWARDED (UDP)
Mar 12 16:39:06.245: tenant-jobs/resumes-7c9d8fc76b-zvnqr:32788 (ID:5772) -> tenant-jobs/coreapi-58b9f5f67d-n54fm:9080 (ID:11699) to-endpoint FORWARDED (TCP Flags: SYN)
Mar 12 16:39:06.245: tenant-jobs/resumes-7c9d8fc76b-zvnqr:32788 (ID:5772) -> tenant-jobs/coreapi-58b9f5f67d-n54fm:9080 (ID:11699) to-endpoint FORWARDED (TCP Flags: ACK)
Mar 12 16:39:06.246: tenant-jobs/resumes-7c9d8fc76b-zvnqr:32788 (ID:5772) -> tenant-jobs/coreapi-58b9f5f67d-n54fm:9080 (ID:11699) to-endpoint FORWARDED (TCP Flags: ACK, PSH)
Mar 12 16:39:06.270: tenant-jobs/resumes-7c9d8fc76b-zvnqr:32788 (ID:5772) -> tenant-jobs/coreapi-58b9f5f67d-n54fm:9080 (ID:11699) to-endpoint FORWARDED (TCP Flags: ACK, FIN)
Mar 12 16:39:07.911: tenant-jobs/resumes-7c9d8fc76b-zvnqr:48685 (ID:5772) -> kube-system/coredns-5d78c9869d-m7rn5:53 (ID:10455) to-endpoint FORWARDED (UDP)
Mar 12 16:39:07.912: tenant-jobs/resumes-7c9d8fc76b-zvnqr:32792 (ID:5772) -> tenant-jobs/coreapi-58b9f5f67d-n54fm:9080 (ID:11699) to-endpoint FORWARDED (TCP Flags: SYN)
Mar 12 16:39:07.912: tenant-jobs/resumes-7c9d8fc76b-zvnqr:32792 (ID:5772) -> tenant-jobs/coreapi-58b9f5f67d-n54fm:9080 (ID:11699) to-endpoint FORWARDED (TCP Flags: ACK)
Mar 12 16:39:07.913: tenant-jobs/resumes-7c9d8fc76b-zvnqr:32792 (ID:5772) -> tenant-jobs/coreapi-58b9f5f67d-n54fm:9080 (ID:11699) to-endpoint FORWARDED (TCP Flags: ACK, PSH)
Mar 12 16:39:07.935: tenant-jobs/resumes-7c9d8fc76b-zvnqr:32792 (ID:5772) -> tenant-jobs/coreapi-58b9f5f67d-n54fm:9080 (ID:11699) to-endpoint FORWARDED (TCP Flags: ACK, FIN)
```

With the Hubble CLI you can carry out extensive filtering.

You can use filters to only capture traffic based on criteria such as: traffic to or from a specific pod, port, service, namespace, label, IP address.

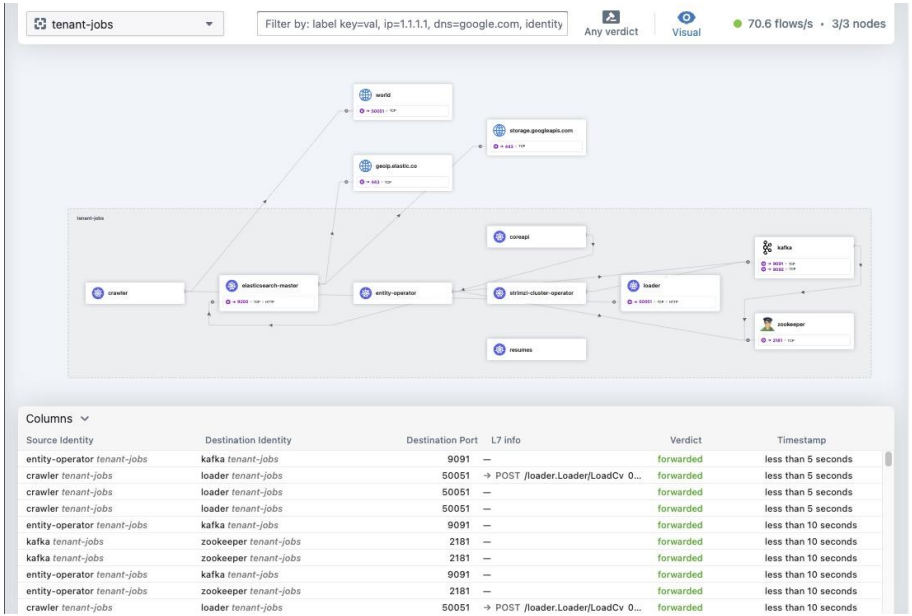
You can also use some of the HTTP like HTTP methods or status code.

Hubble CLI


```
root@server:~# hubble observe --verdict DROPPED -o json | jq
{
  "flow": {
    "time": "2024-03-12T15:45:23.401879239Z",
    "uuid": "3d60a80d-dd86-4891-89e0-419b46da7c04",
    "verdict": "DROPPED",
    "drop_reason": 139,
    "ethernet": {
      "source": "8a:aa:1f:77:4a:df",
      "destination": "33:33:00:00:00:02"
    },
    "IP": {
      "source": "fe80::88aa:1fff:fe77:4adf",
      "destination": "ff02::2",
      "ipVersion": "IPv6"
    },
    "l4": {
      "ICMPv6": {
        "type": 133
      }
    },
    "source": {
      "identity": 1515,
      "labels": [
        "k8s:io.cilium.k8s.namespace.labels.kubernetes.io/metadata.name=tenant-jobs",
        "k8s:io.cilium.k8s.namespace.labels.name=tenant-jobs",
        "k8s:io.cilium.k8s.policy.cluster=default",
        "k8s:io.cilium.k8s.policy.serviceaccount=strimzi-cluster-operator",
        "k8s:io.kubernetes.pod.namespace=tenant-jobs",
        "k8s:name=strimzi-cluster-operator",
        "k8s:strimzi.io/kind=cluster-operator"
      ]
    },
    "destination": {
      "labels": [
        "reserved:unknown"
      ]
    },
    "Type": "L3_L4",
    "node_name": "kind-worker2",
    "event_type": {
      "type": 1,
      "sub_type": 139
    },
    "traffic_direction": "INGRESS",
```

If you wish, you can also render your network flow logs in JSON by adding the `*-o json*` flag to the `*hubble observe*` command

Hubble UI



- All your network flow traffic in one dashboard
- Visualization with Service Maps
- Filter network traffic by verdict
- Easily switch between namespaces in your cluster

Hubble Metrics

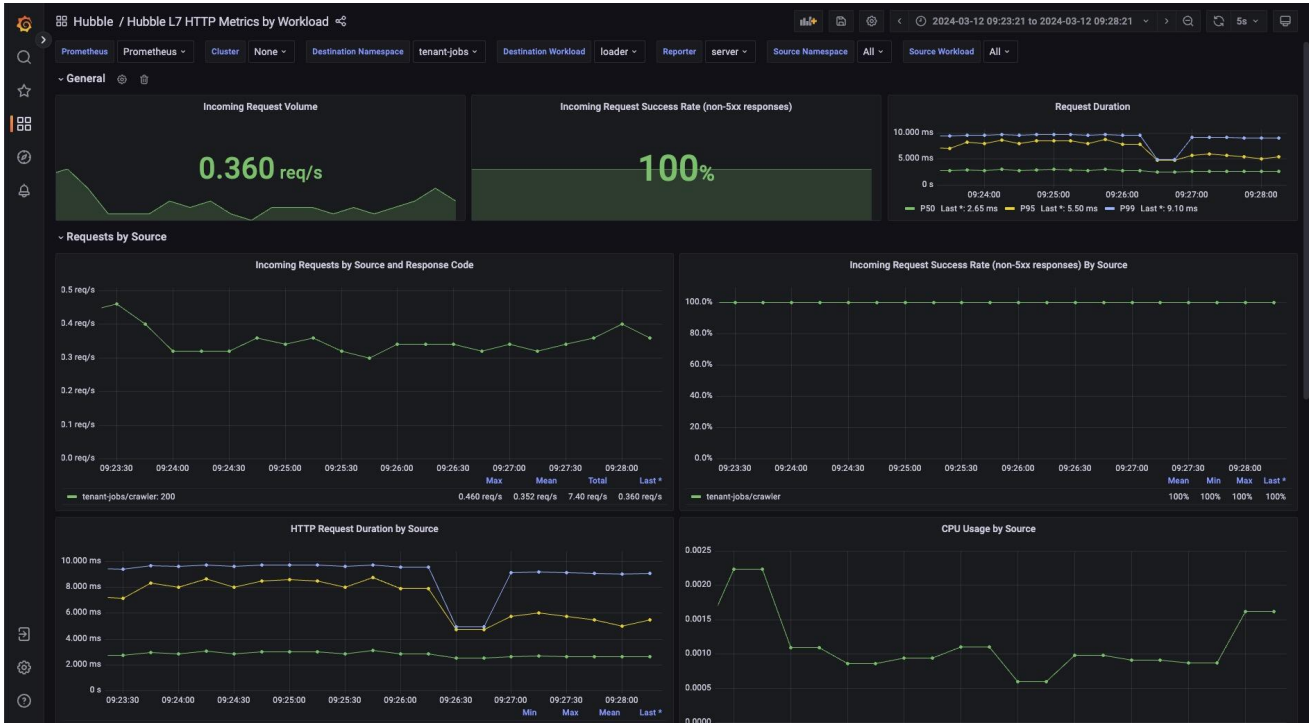
- Hubble exposes a metrics endpoint which can be scraped
  - Prometheus

- Built-in metrics
  - HTTP
  - TCP
  - UDP
  - DNS
  - ICMP

- Visualization with Grafana

- Customizable and Extendable
  - Write your own metrics :)

Hubble Metrics



A Grafana server with a data source pointing to Prometheus and L7 HTTP metrics from Hubble

Hubble exposes Prometheus metrics

Observability Day
EUROPE

DEMO

https://isogo.to/cilium-hubble-demo

Learning Resources

- Official introduction to Cilium and Hubble
- Official Documentation
- Cilium Labs
- eCHO show
- “Are These Things Talking to Each Other?” - Observing Kubernetes Networking... - Anna Kapuścińska
- Hubble - eBPF Based Observability for Kubernetes - Sebastian Wicki, Isovalent

Observability Day
EUROPE

THANK YOU :)