WATER, WATER EVERYWHERE

MAKING **AUDIT** LOGS POTABLE

kate kuchin

WATER, WATER EVERYWHERE

MAKING **AUDIT** LOGS POTABLE

kate kuchin

SENIOR
SOFTWARE
ENGINEER
@ **HEPTIO**

GITHUB @k8k

TWITTER @exkuchme

INSTA @katekuchin

SENIOR
~~SOFTWARE~~
~~ENGINEER~~
~~@ HEPTIO~~

GITHUB @k8k

TWITTER @exkuchme

INSTA @katekuchin

SENIOR
MEMBER OF
TECHNICAL STAFF
@ **VMWARE**

GITHUB @k8k

TWITTER @exkuchme

INSTA @katekuchin


AMY


ASHTYN


ERIC


FABIO


HELEN


MARLON


MOYER


ME


PATRICK


ROSS


TAMIKO

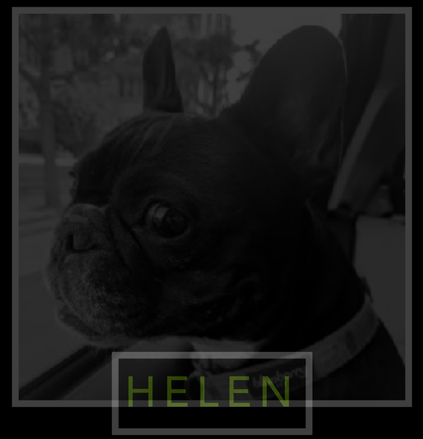

XAVIER


AMY

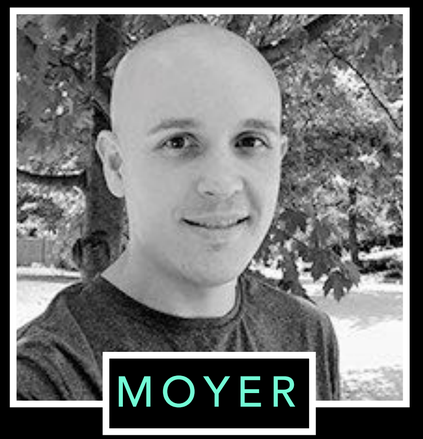

ASHTYN


ERIC


FABIO


HELEN


MARLON


MOYER


ME


PATRICK


ROSS


TAMIKO


XAVIER


AMY


ASHTYN


ERIC


FABIO


HELEN


MARLON


MOYER


ME


PATRICK


ROSS


TAMIKO


XAVIER


AMY


ASHTYN


ERIC


FABIO


HELEN


MARLON


MOYER


ME


PATRICK


ROSS


TAMIKO


XAVIER


AMY


ASHTYN


ERIC


FABIO


HELEN


MARLON


MOYER

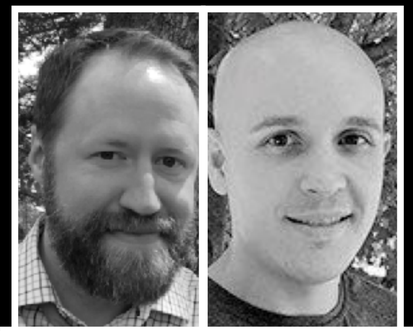

ME


PATRICK


ROSS


TAMIKO


XAVIER



LAWFUL GOOD

NEUTRAL GOOD

CHAOTIC GOOD

LAWFUL NEUTRAL

TRUE NEUTRAL

CHAOTIC NEUTRAL

LAWFUL EVIL

NEUTRAL EVIL

CHAOTIC EVIL

NO **AGENDA** NO ATTENDA

- Who am I / Yes I work for VMWare now
- What is Kubernetes Audit Logging & Why Should You Care
- Configuring Audit
- Making Sense of Audit
- Very Cool Demo 10/10
- fin.

AGENDA

- ~~Who am I / Yes I work for VMWare now~~
- **What is Kubernetes Audit Logging & Why You Should Care**
- Configuring Audit
- Making Sense of Audit
- Very Cool Demo 10/10
- fin.

WTF IS KUBERNETES AUDIT LOGGING

AND WHY SHOULD YOU CARE

AUDIT LOGS
RECORD
ALL REQUESTS
MADE **TO** THE
KUBERNETES API SERVER

EXTERNAL **COMPLIANCE**

ANOMALY DETECTION

EVENT **ATTRIBUTION**

EXTERNAL **COMPLIANCE**

ANOMALY DETECTION

EVENT ATTRIBUTION

EXTERNAL COMPLIANCE

ANOMALY DETECTION

EVENT ATTRIBUTION

EXTERNAL COMPLIANCE

ANOMALY DETECTION

EVENT **ATTRIBUTION**

OKAY, WHAT'S A LOG LOOK LIKE

SHAPE OF **AUDIT LOGS**

WHAT HAPPENED?
WHEN DID IT HAPPEN?
WHO DID IT?
WHERE WAS IT INITIATED?
WHAT DID IT HAPPEN ON?
WHERE WAS IT OBSERVED?

AUDIT EVENTS


```
{
    "metadata": {
        "creationTimestamp": "2018-11-15T20:18:13Z"
    },
    "level": "Request",
    "timestamp": "2018-11-15T20:18:13Z",
    "auditID": "277da1b5-16aa-4a1b-ae0f-4d1cdb2cc67a",
    "stage": "ResponseComplete",
    "requestURI": "/api/v1/namespaces/default/pods/populator-58f9bd8b6f-ngx8s",
    "verb": "get",
    "user": {
        "username": "system:node:ip-10-0-100-113.us-west-2.compute.internal",
        "groups": [
            "system:nodes",
            "system:authenticated"
        ]
    },
    "sourceIPs": [
        "34.217.76.189"
    ],
    "objectRef": {
        "resource": "pods",
        "namespace": "default",
        "name": "populator-58f9bd8b6f-ngx8s",
        "apiVersion": "v1"
    },
    "responseStatus": {
        "metadata": {},
        "code": 200
    },
    "requestReceivedTimestamp": "2018-11-15T20:18:13.388833Z",
    "stageTimestamp": "2018-11-15T20:18:13.389856Z"
}
```

WHAT HAPPENED

WHAT HAPPENED

```
{
   "verb": "get"
}
```

WHEN DID IT HAPPEN

WHEN DID IT HAPPEN


```
{
  "stage": "ResponseComplete",
  "metadata": {
    "creationTimestamp": “2018-11-15T20:18:13Z"
  },
  "requestReceivedTimestamp": “2018-11-15T20:18:13.388833Z",
  "stageTimestamp": “2018-11-15T20:18:13.389856Z"
}
```

WHO DID IT

WHO DID IT

```
{
  “user”: {
    "username": "system:node:ip-10-0-100-113.us- west-2.compute.internal",
    "groups": [
      "system:nodes",
      "system:authenticated"
    ]
  }
}
```

WHERE WAS IT INITIATED

WHERE WAS IT INITIATED

```
{
   "sourceIPs": [
      "34.217.76.189"
   ]
}
```

WHAT DID IT HAPPEN ON

WHAT DID IT HAPPEN ON

```
{
  “objectRef": {
    "resource": "pods",
    "namespace": "default",
    "name": "populator-58f9bd8b6f-ngx8s"
  }
}
```

WHERE WAS IT OBSERVED

WHERE WAS IT OBSERVED

Literally

WHERE WAS IT OBSERVED

Literally

always

WHERE WAS IT OBSERVED

Literally

always

the API Server

LOGS ARE CREATED AT EACH STAGE

AUDIT **STAGES**

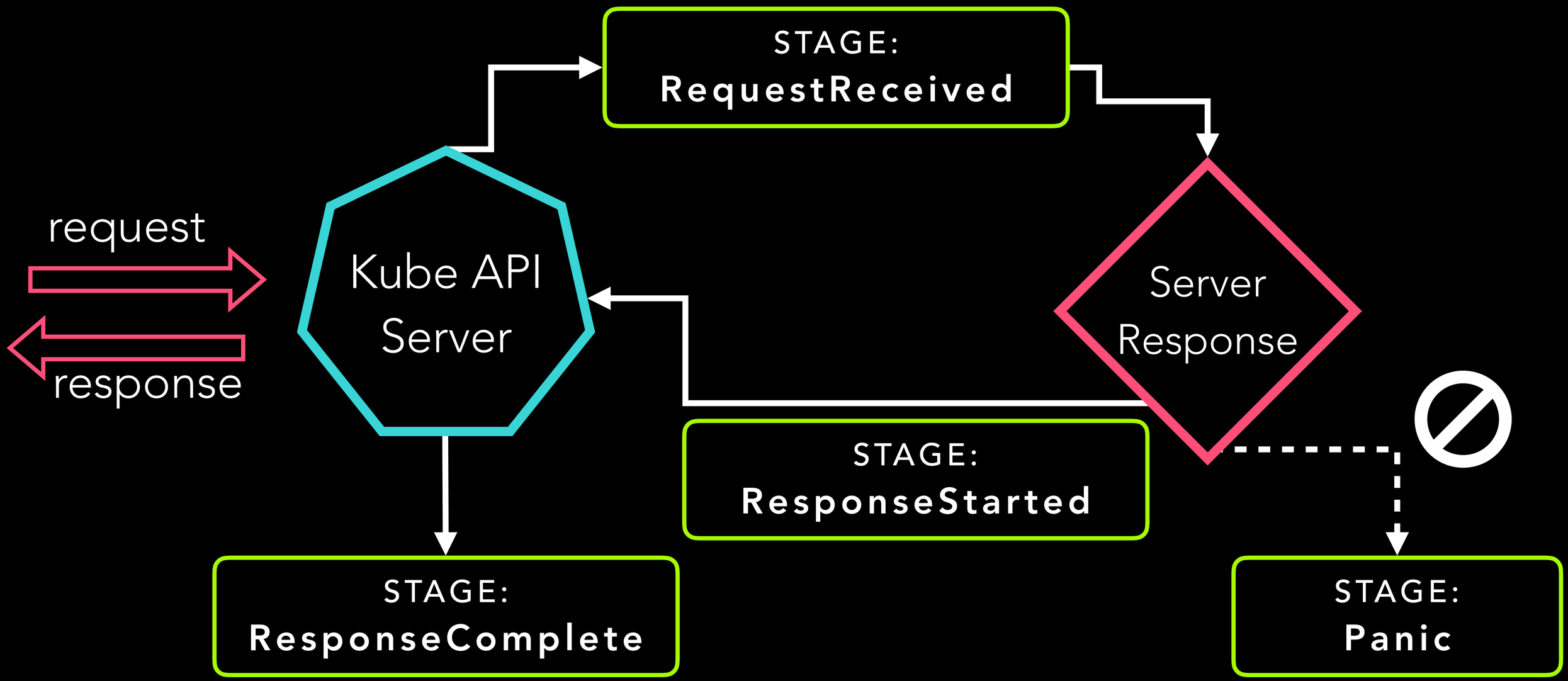
RequestReceived audit handler receives request

ResponseStarted only for requests like `watch`

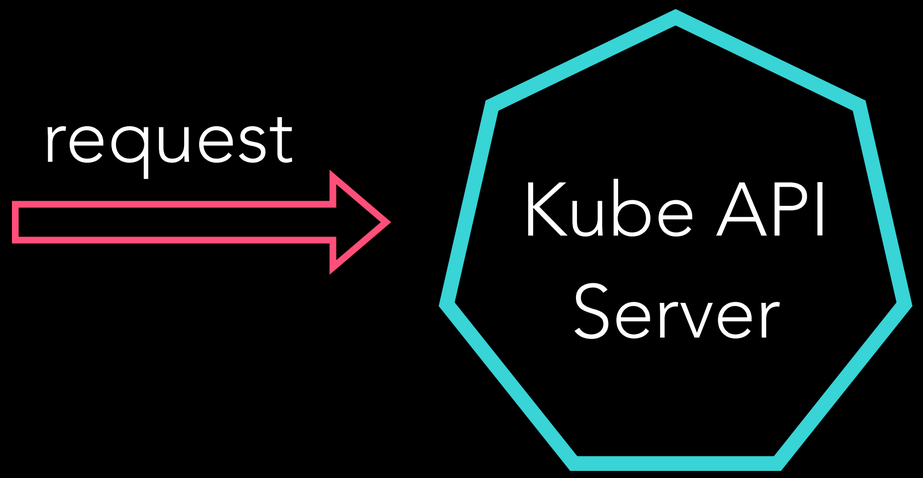
ResponseComplete no more bytes will be sent

Panic generated when a panic occurs

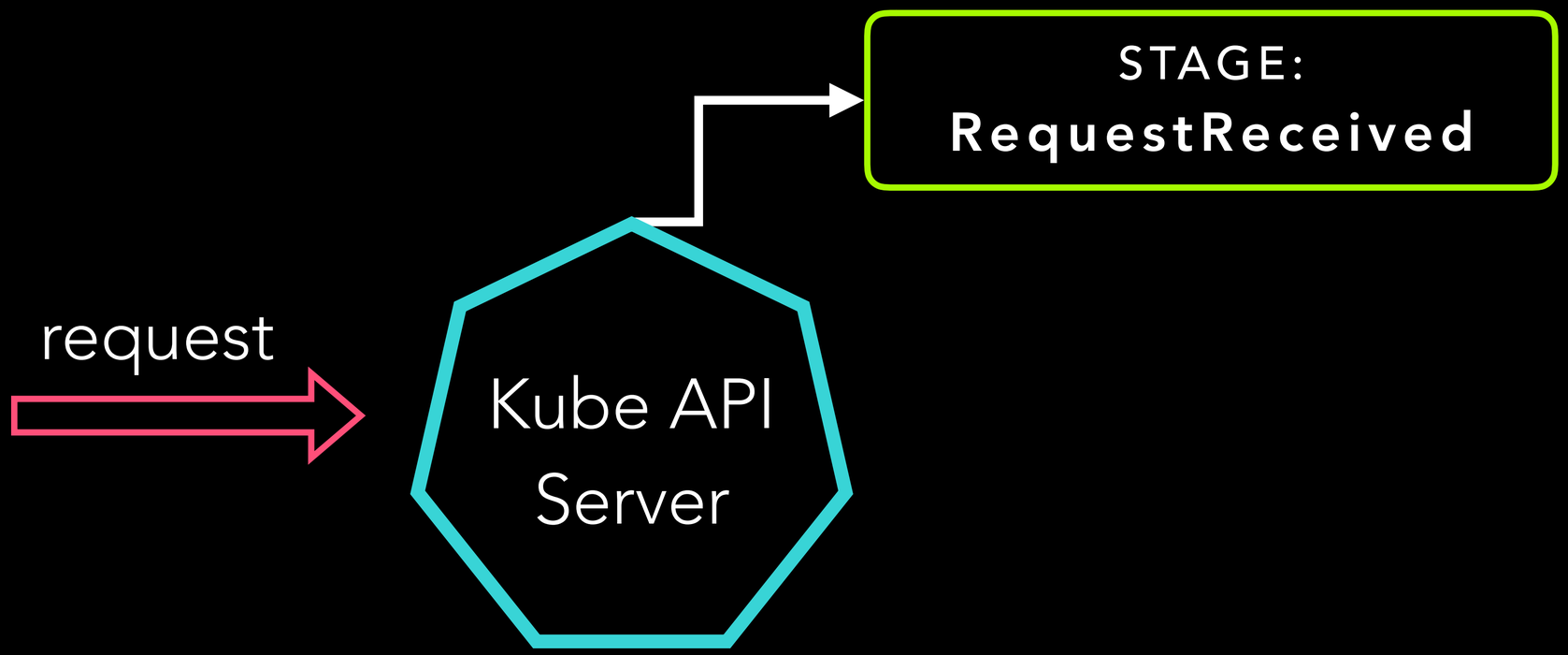
AUDIT **STAGES**



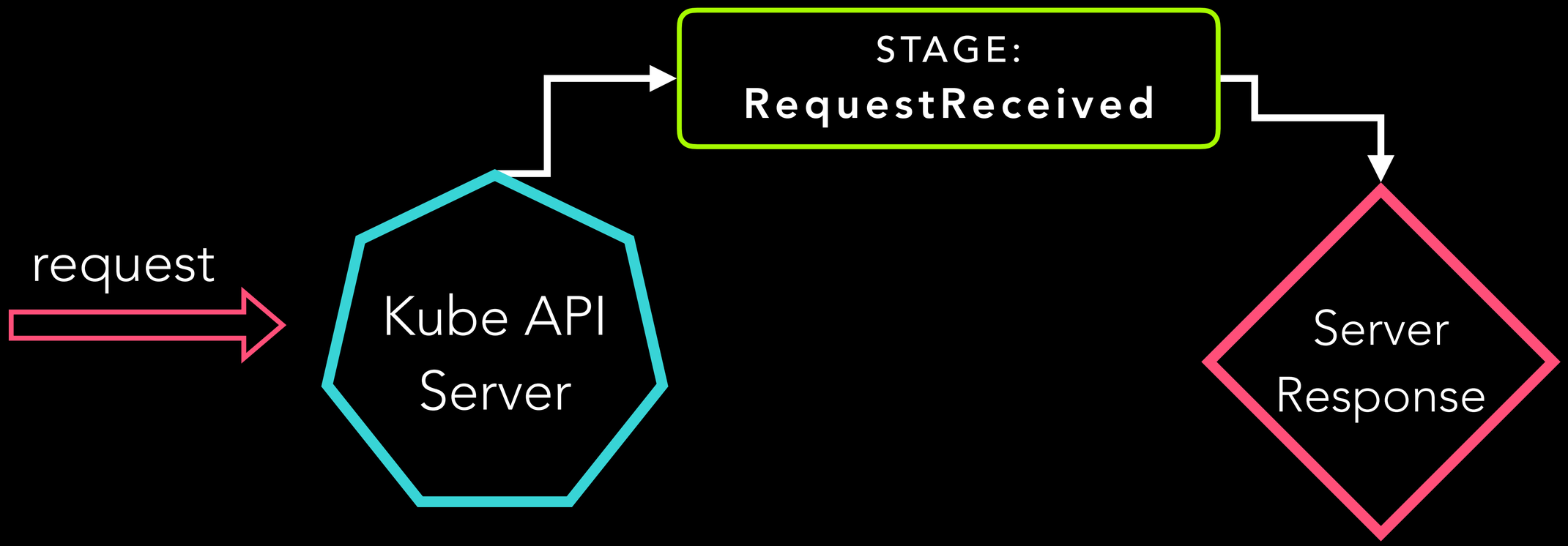
AUDIT **STAGES**



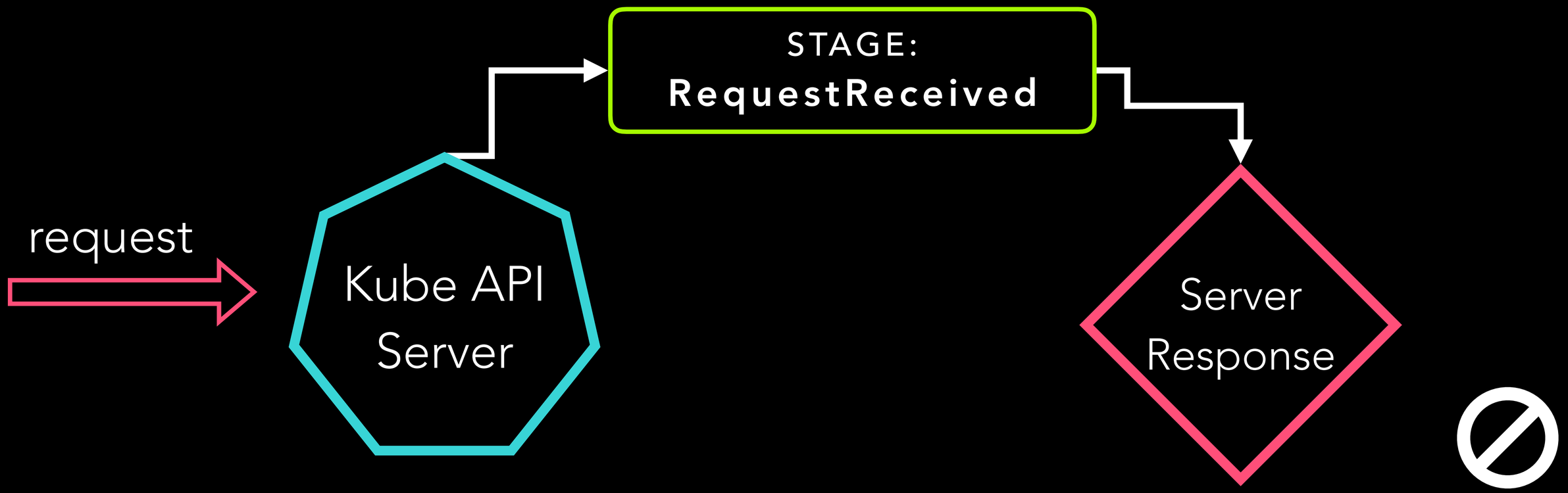
AUDIT **STAGES**



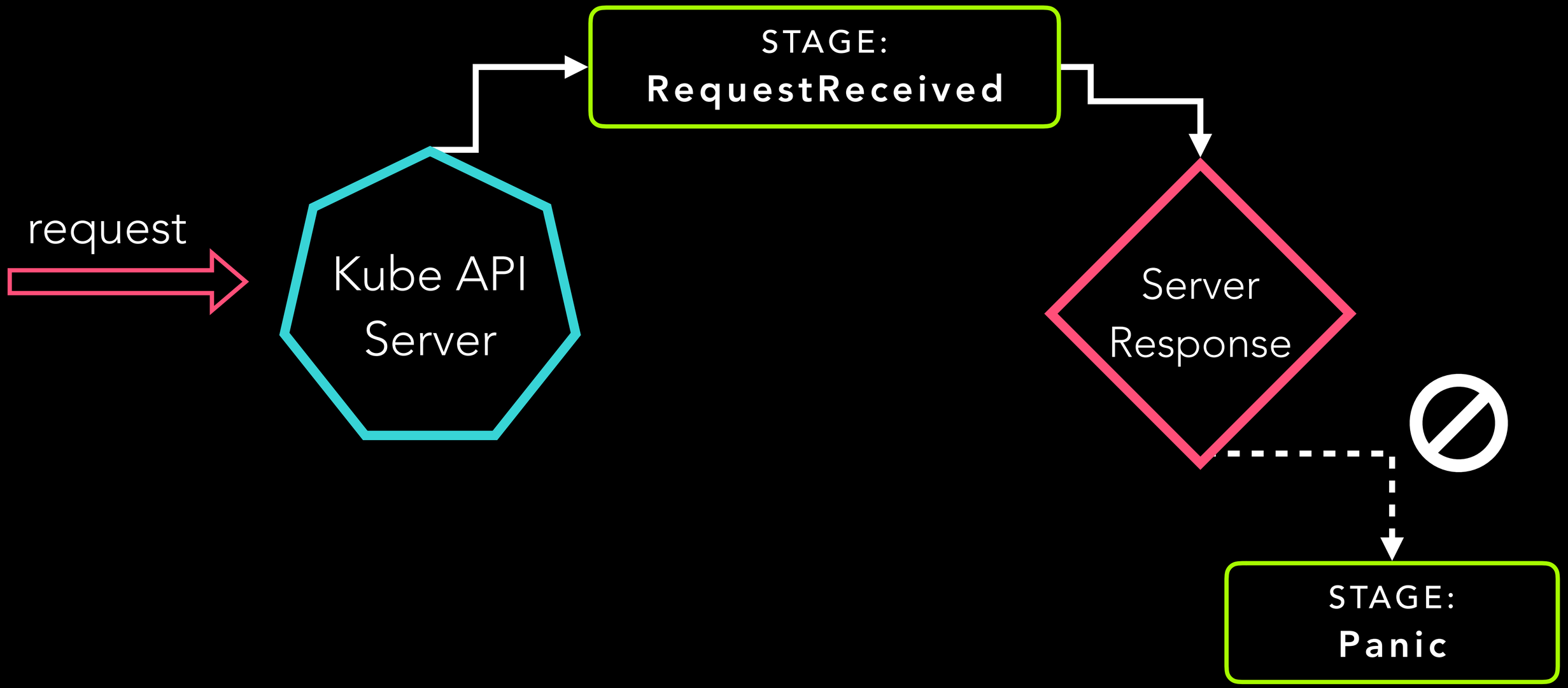
AUDIT **STAGES**



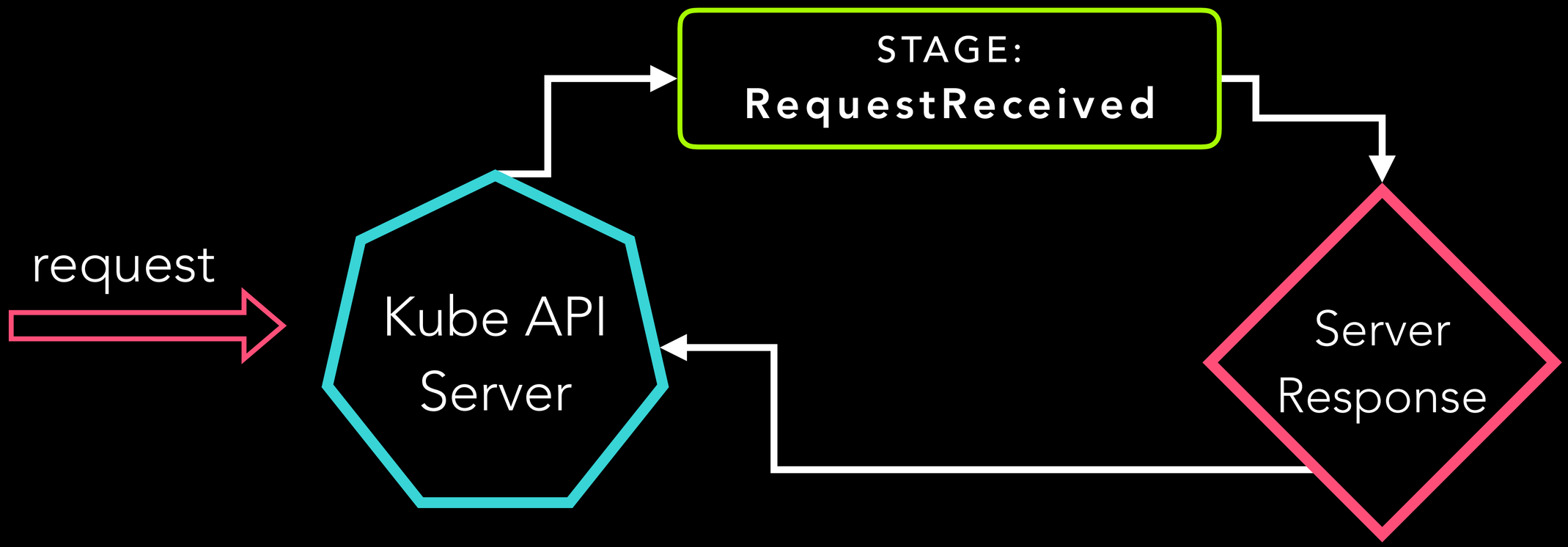
AUDIT **STAGES**



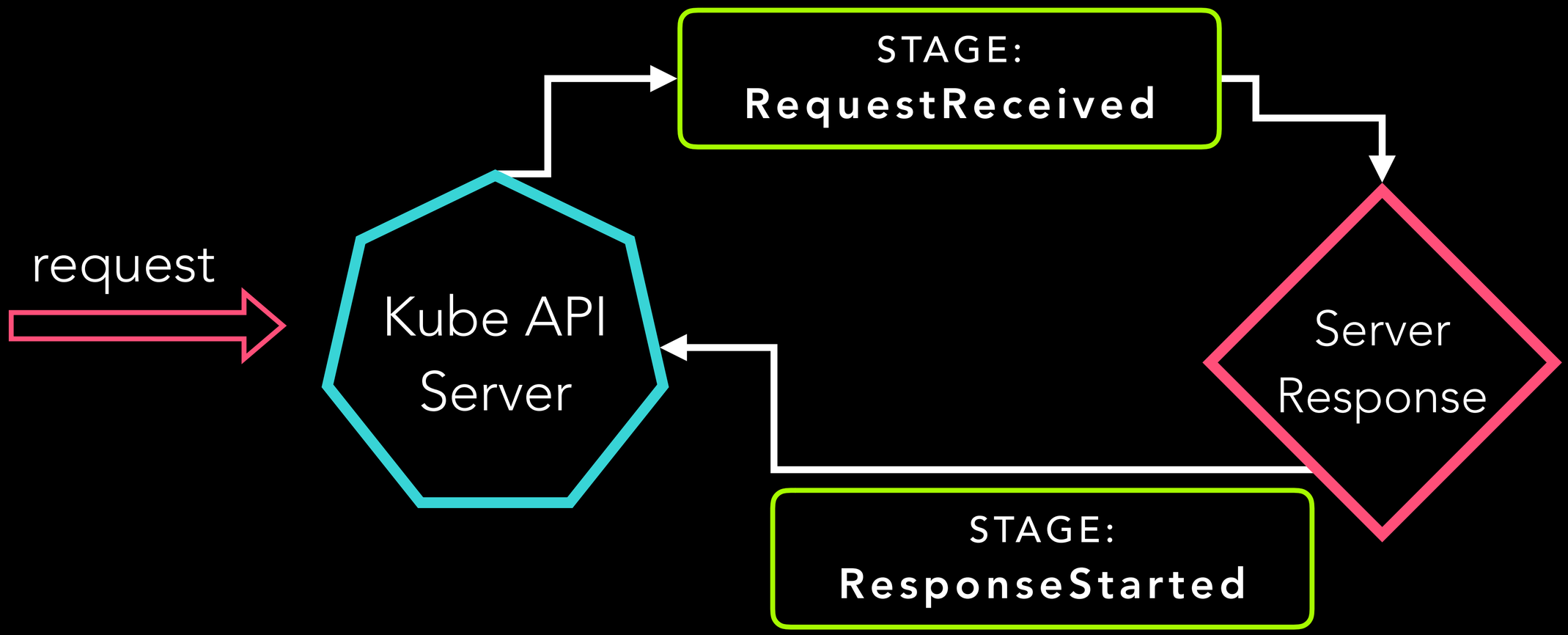
AUDIT **STAGES**



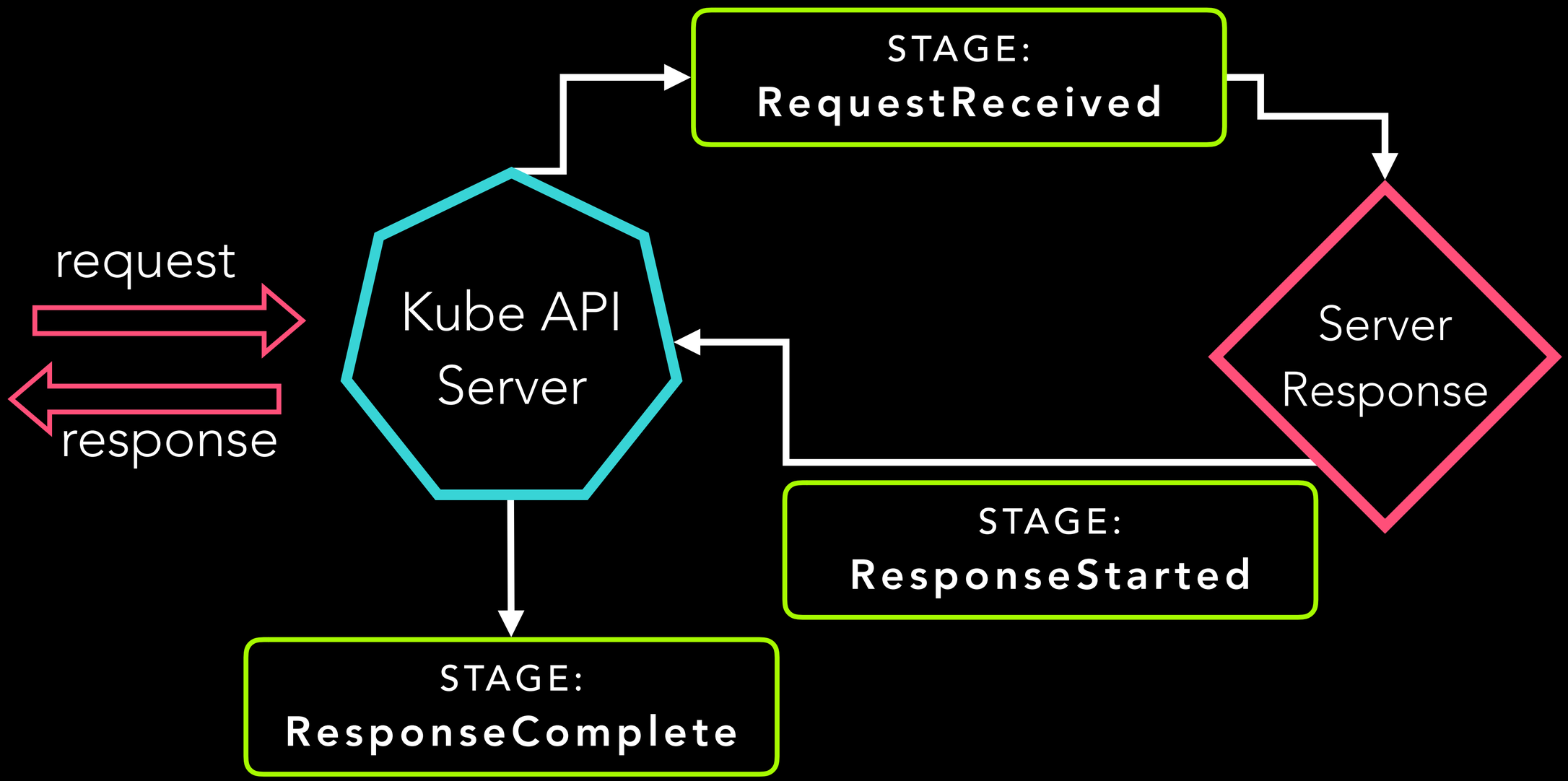
AUDIT **STAGES**



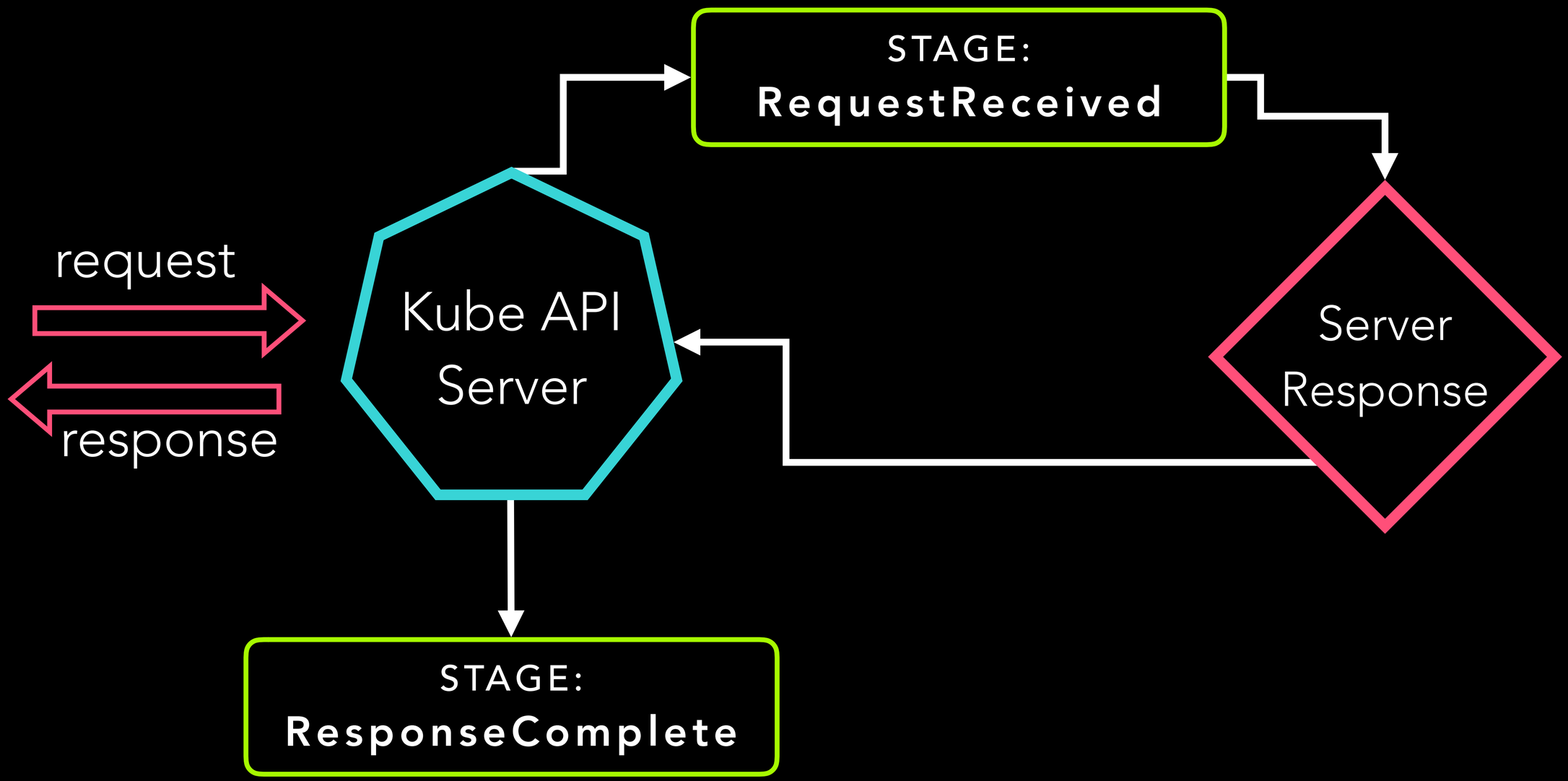
AUDIT **STAGES**



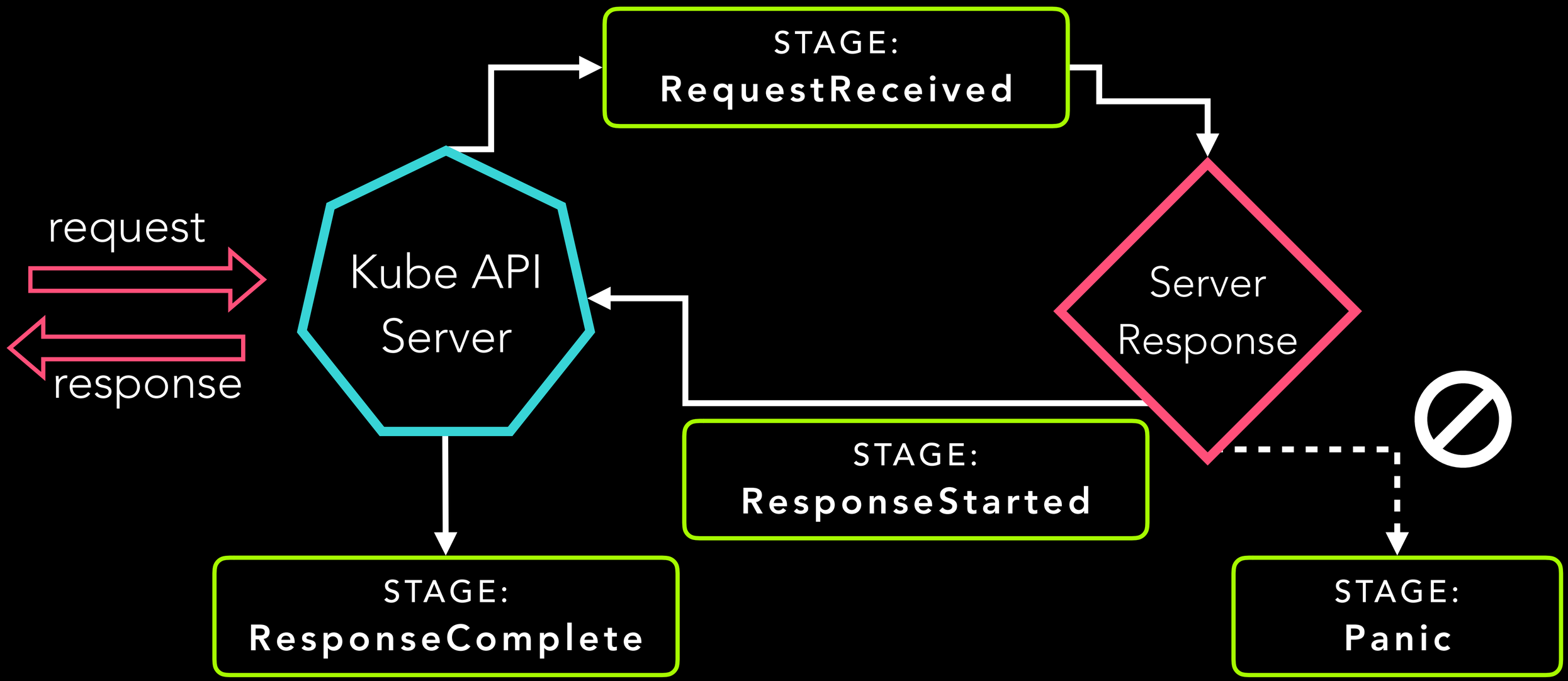
AUDIT **STAGES**



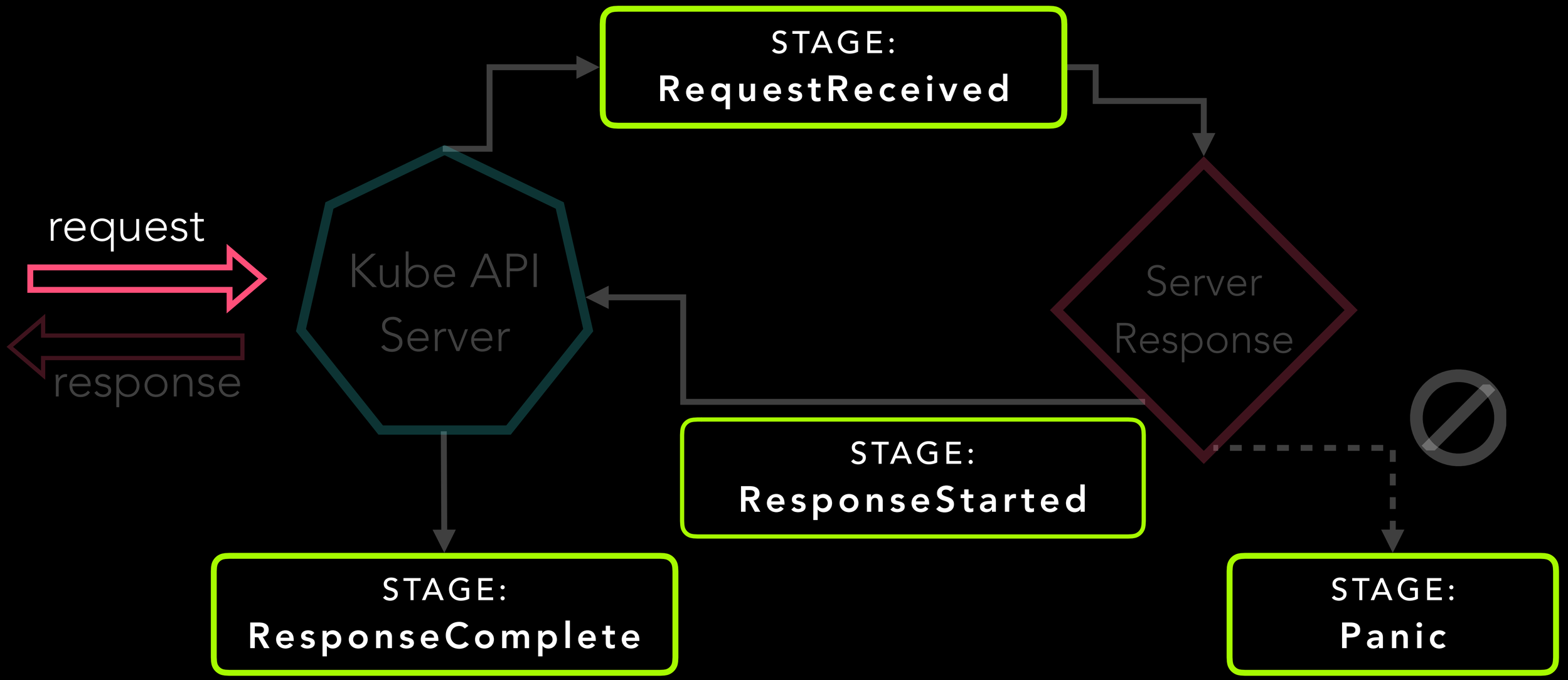
AUDIT **STAGES**



AUDIT **STAGES**



AUDIT **STAGES**



LEVEL IS DETERMINED BY POLICY

AUDITING **LEVELS**

None no requests are logged

Metadata only request metadata is logged

Request request object is logged

RequestResponse request & response objects are logged

None no requests are logged

Metadata only request metadata is logged

Request request object is logged

RequestResponse request & response objects are logged

None no requests are logged

Metadata only request metadata is logged

Request request object is logged

RequestResponse request & response objects are logged

None no requests are logged

Metadata only request metadata is logged

Request request object is logged

RequestResponse request & response objects are logged

None no requests are logged

Metadata only request metadata is logged

Request request object is logged

RequestResponse request & response objects are logged

EXAMPLE **POLICY**

```
apiVersion: audit.k8s.io/v1
kind: Policy
omitStages:
- "RequestReceived"
rules:
- level: Metadata
  resources:
  - group: ""
    resources: ["secrets"]
```

OUTPUT


```
{
  "level": "Metadata",
  "timestamp": "2018-12-09T17:02:25Z",
  "stage": "ResponseComplete",
  "requestURI": "/api/v1/namespaces/default/secrets/api-key",
  "verb": "get",
  “user": {
    "usename": "dolores"
  },
  "sourceIPs": ["10.0.143.187"],
  "objectRef": {
    "resource": "secrets",
    "namespace": "default"
  },
  "requestReceivedTimestamp": “2018-12-09T17:02:25.399627Z",
  "stageTimestamp": "2018-12-09T17:02:25.400717Z"
}
```

AGENDA

- ~~Who am I / Yes I work for VMWare now~~
- ~~What is Kubernetes Audit Logging & Why You Should Care~~
- **Configuring Audit**
- Making Sense of Audit
- Very Cool Demo 10/10
- fin.

PROBLEM 1

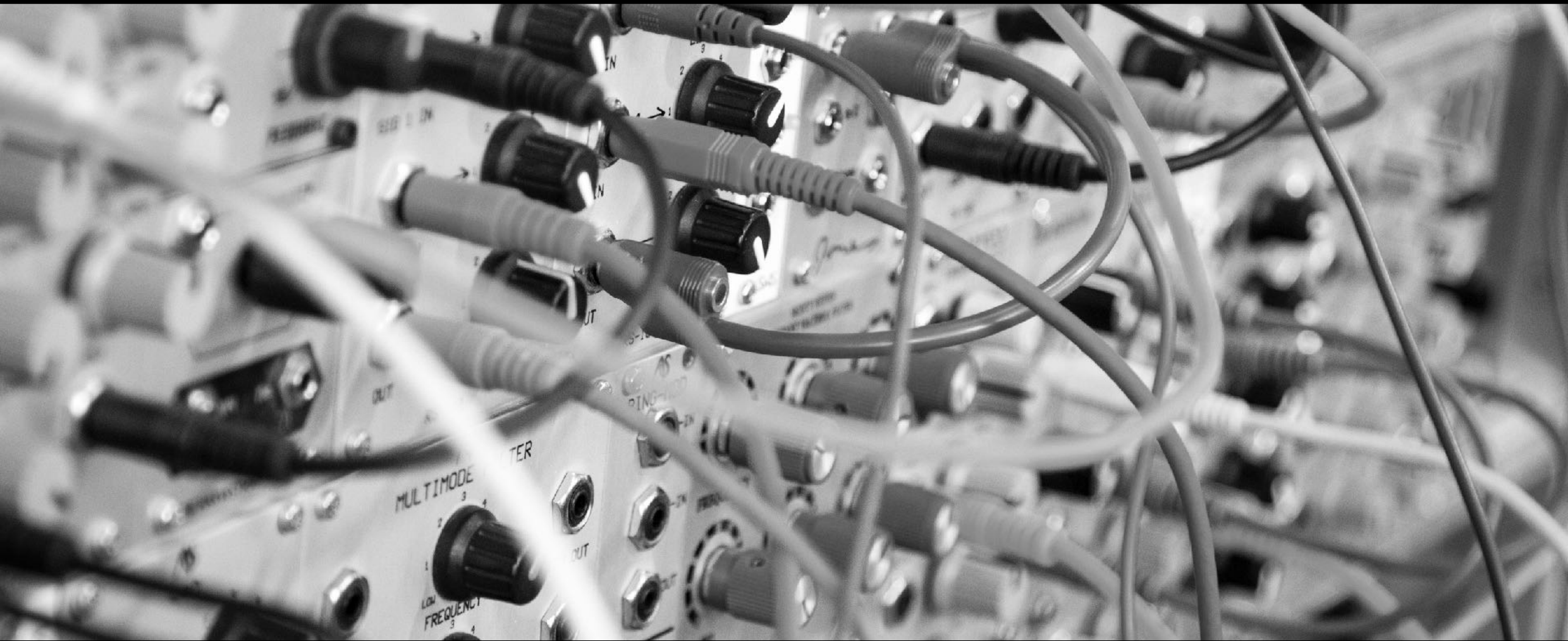
CONFIGURING AUDIT

YOU GOTTA

KNOW YOUR **INSTALLER**

GET READY TO MAKE SOME MISTAKES

CONFIGURATION IS **PRECISE**



NEW CHANGES REQUIRE YOU TO

RESTART THE API **SERVER**

soooooo...

WHAT'S THE **PROBLEM**

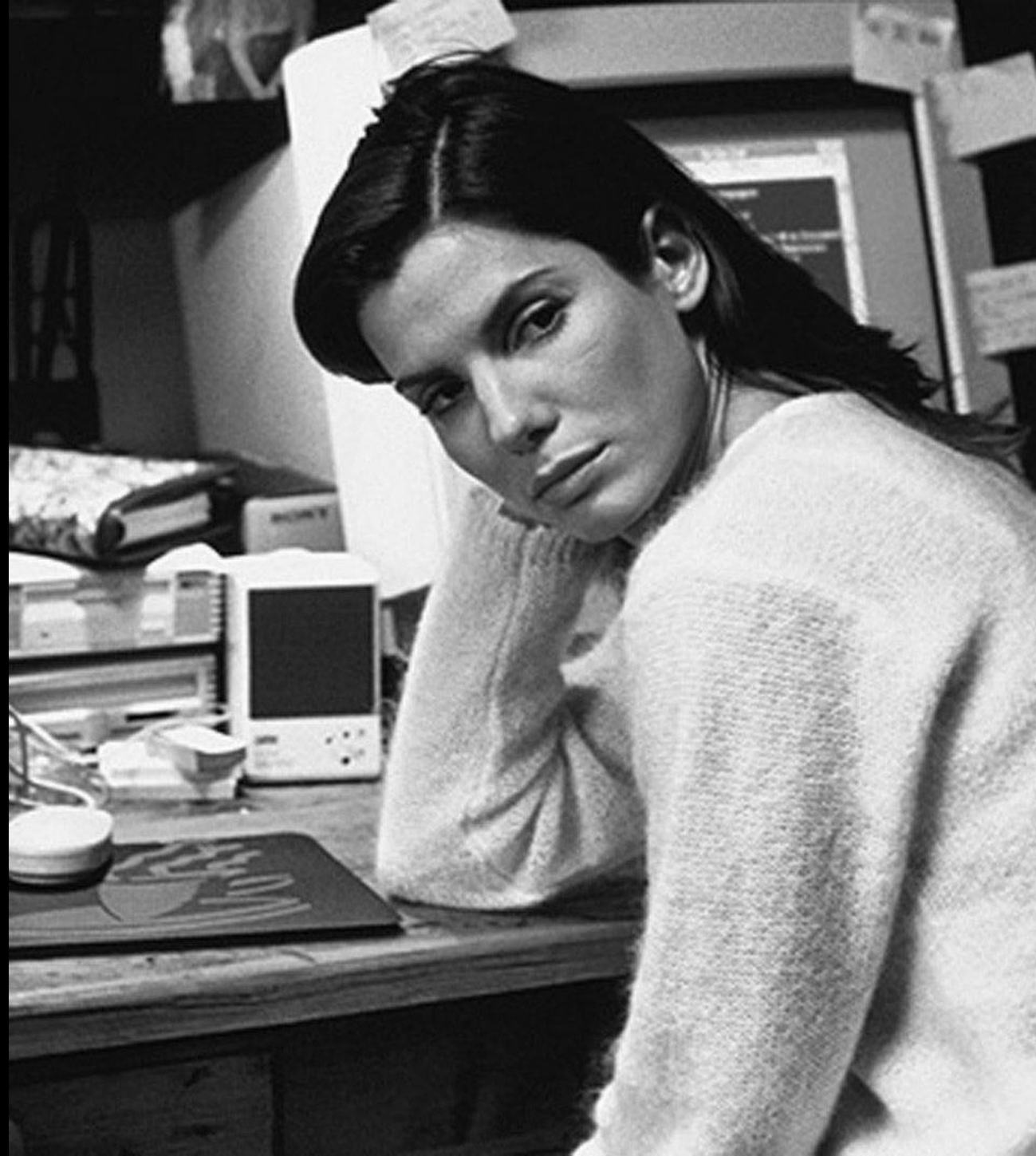
CLUSTER
OPERATORS

CLUSTER OPERATORS

- have to copy files up to the master node

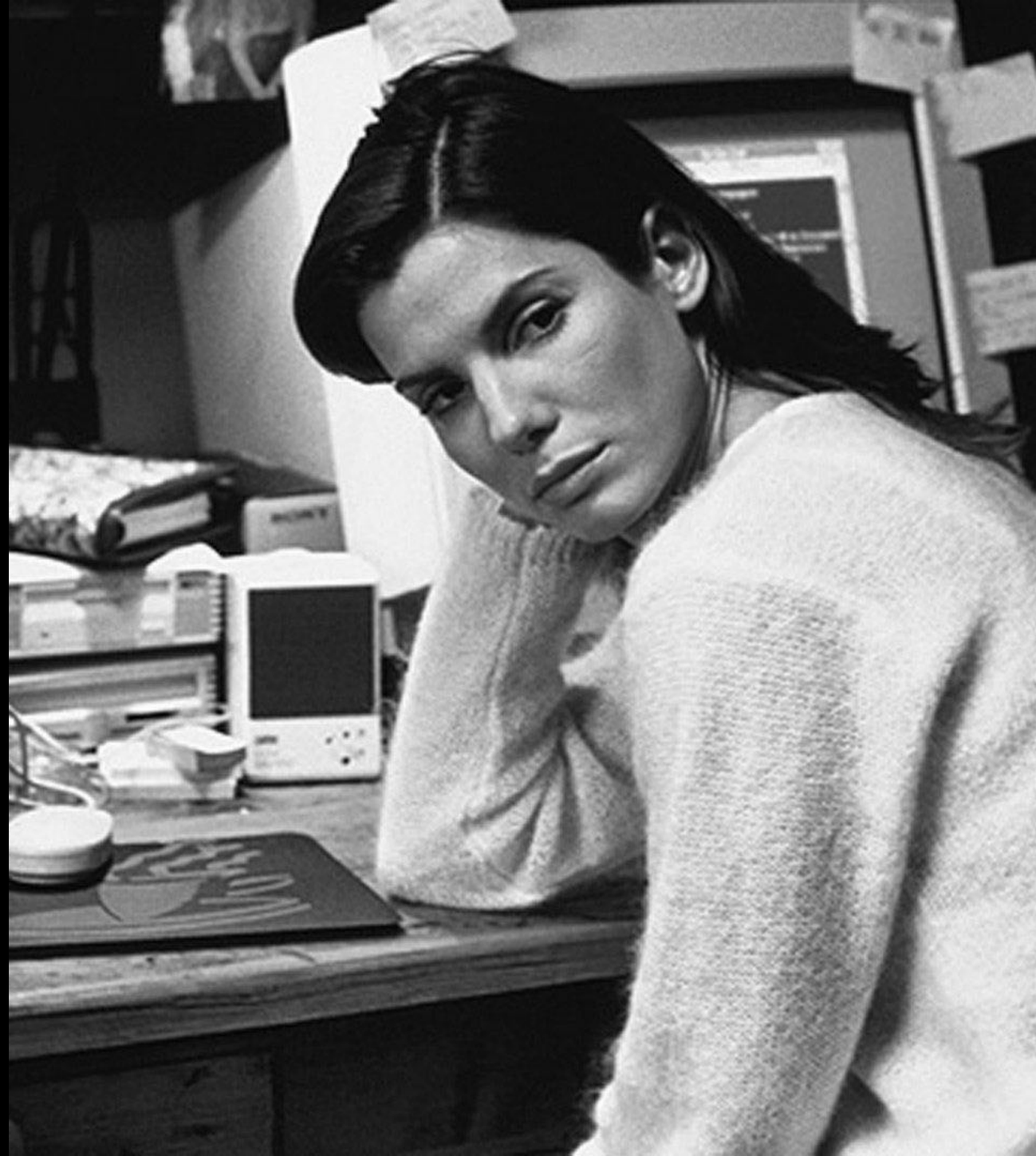
- must restart the API server on every single iteration

APP DEVS



APP DEVS

difficult to turn up audit logs when debugging an application



COOL KID

(THIRD PARTY)

APP DEVS

COOL KID

(THIRD PARTY)

APP DEVS

can’t drop in audit facilities

DYNAMIC AUDIT CONFIG

(v1.13 alpha)


WTF IS
DYNAMIC AUDIT CONFIG?

WTF IS

DYNAMIC AUDIT CONFIG?

A SET OF OBJECTS THAT ALLOW YOU TO CONFIGURE ADVANCED AUDIT FEATURES...WAIT FOR IT

DYNAMICALLY

API SERVER FLAGS

```
--audit-dynamic-configuration
--feature-gates=DynamicAuditing=true
--runtime-config=auditregistration.k8s.io/v1alpha1=true
```

API SERVER FLAGS

--audit-dynamic-configuration

--feature-gates=DynamicAuditing=true

--runtime-config=auditregistration.k8s.io/v1alpha1=true

AUDIT SINK

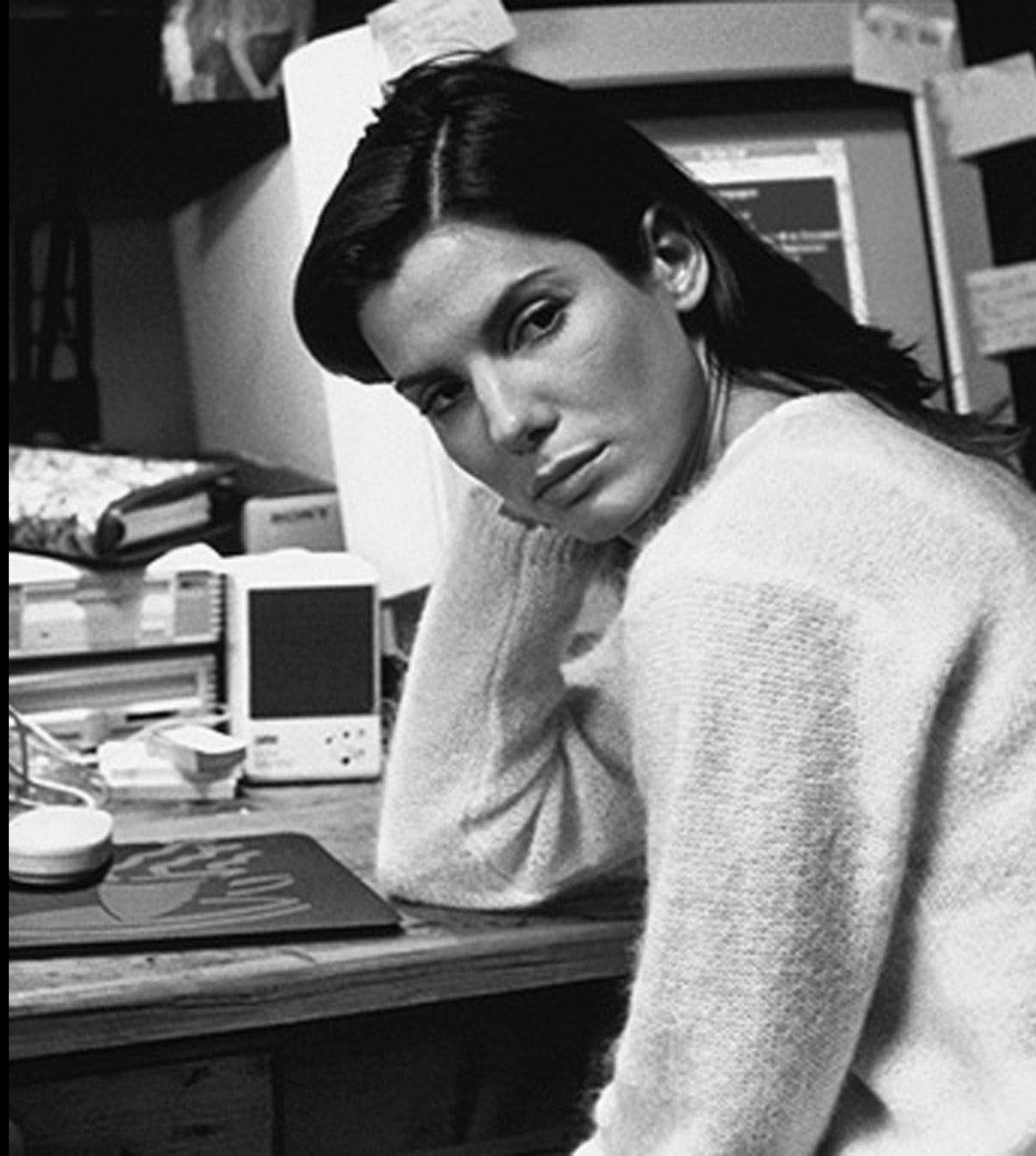
```yaml
apiVersion: auditregistration.k8s.io/v1alpha1
kind: AuditSink
metadata:
 name: mysink
spec:
 policy:
   level: Metadata
   stages:
   - ResponseComplete
 webhook:
   throttle:
     qps: 10
     burst: 15
   clientConfig:
     url: "https://audit.app"
```

CLUSTER
OPERATORS

CLUSTER OPERATORS

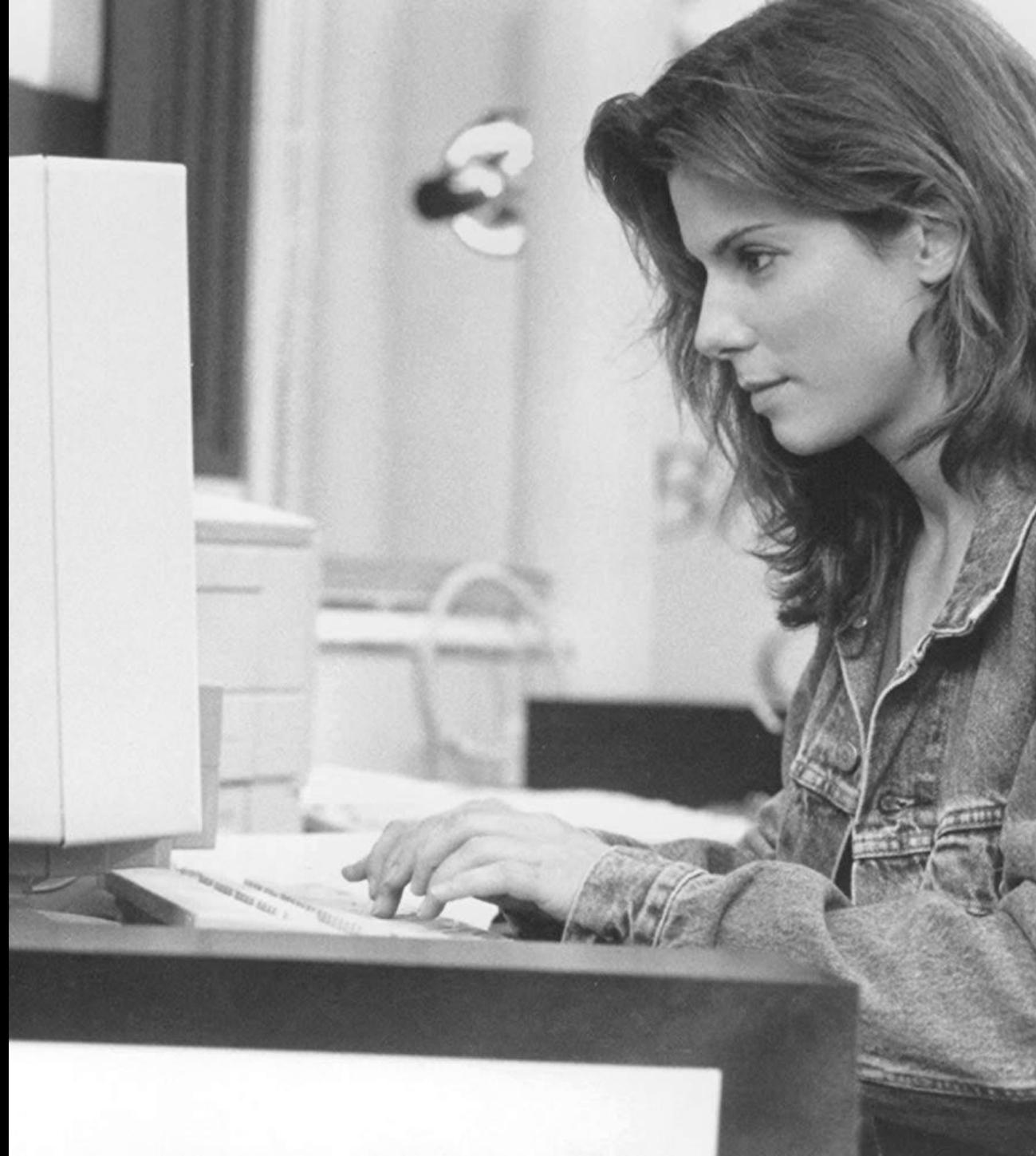
can just create or edit an **AuditSink** object to configure audit on their clusters

APP DEVS



APP DEVS

can easily update log levels when they're debugging



COOL KID

(THIRD PARTY)

APP DEVS

COOL KID (THIRD PARTY) **APP DEVS**

can drop in third party audit facilities to their heart's content

ENABLING **DYNAMIC** AUDIT ON A **KIND*** **CLUSTER**

*kubernetes in docker

CREATE A **KIND CLUSTER**

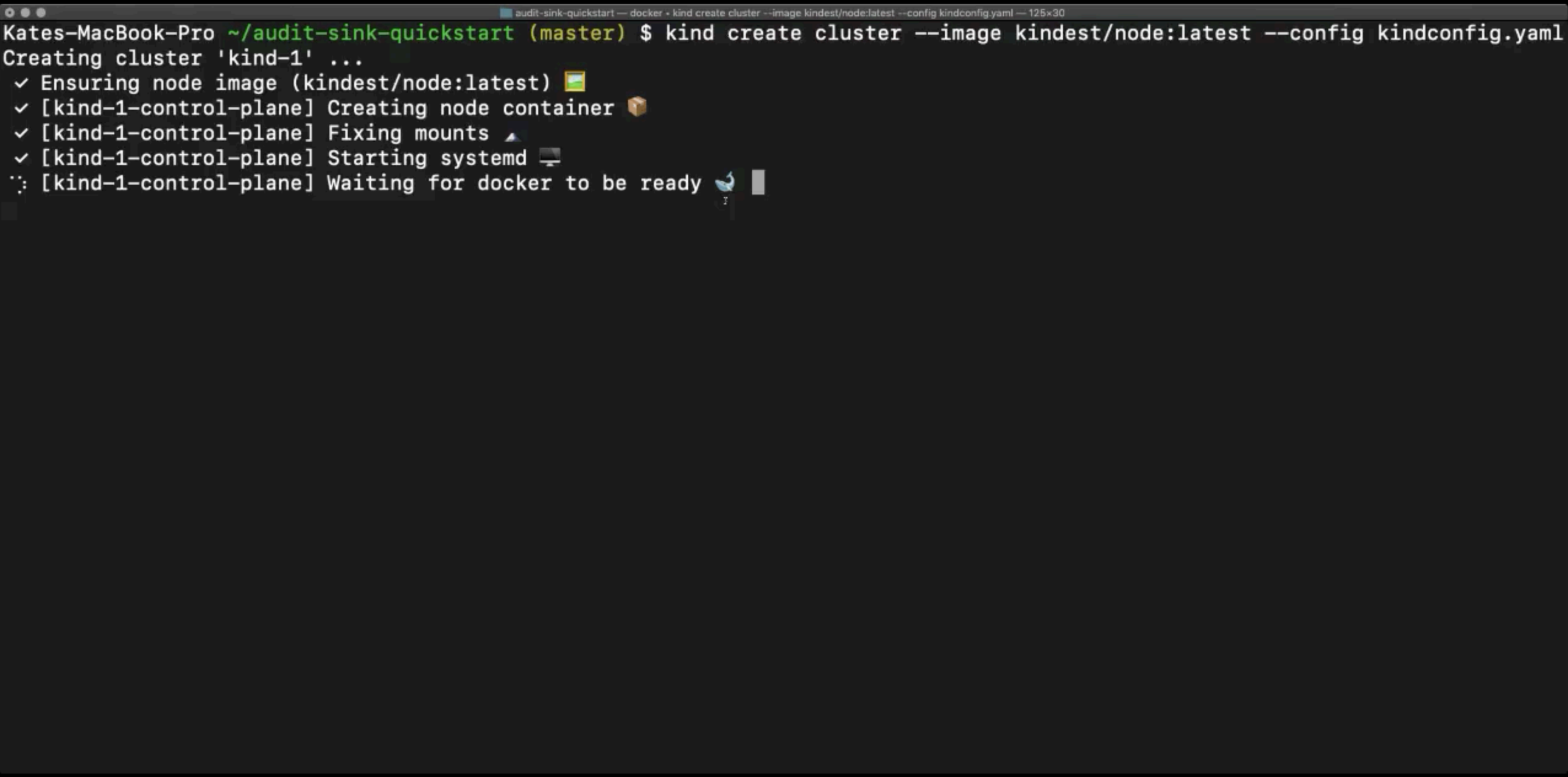
```
kind build node-image \
  && kind create cluster \
  --image kindest/node:latest \
  --config kindconfig.yaml
```

CREATE A **KIND CLUSTER**

```
kind build node-image \
  && kind create cluster \
  --image kindest/node:latest \
  --config kindconfig.yaml
```

KINDCONFIG.YAML

```yaml
apiVersion: kind.sigs.k8s.io/v1alpha1
kind: Config
kubeadmConfigPatches:
- |
  apiVersion: kubeadm.k8s.io/v1beta1
  kind: ClusterConfiguration
  apiServer:
    extraArgs:
      audit-dynamic-configuration: "true"
      feature-gates: DynamicAuditing=true
      runtime-config: auditregistration.k8s.io/v1alpha1=true
```


audit-sink-quickstart — docker • kind create cluster --image kindest/node:latest --config kindconfig.yaml — 125×30
Kates-MacBook-Pro ~/audit-sink-quickstart (master) $ kind create cluster --image kindest/node:latest --config kindconfig.yaml
Creating cluster 'kind-1' ...
✓ Ensuring node image (kindest/node:latest)
✓ [kind-1-control-plane] Creating node container
✓ [kind-1-control-plane] Fixing mounts
✓ [kind-1-control-plane] Starting systemd
[kind-1-control-plane] Waiting for docker to be ready

```
audit-sink-quickstart — -bash — 125×30
Kates-MacBook-Pro ~/audit-sink-quickstart (master) $ kind create cluster --image kindest/node:latest --config kindconfig.yaml
Creating cluster 'kind-1' ...
 ✓ Ensuring node image (kindest/node:latest) 🖼
 ✓ [kind-1-control-plane] Creating node container 📦
 ✓ [kind-1-control-plane] Fixing mounts 🗻
 ✓ [kind-1-control-plane] Starting systemd 🖥
 ✓ [kind-1-control-plane] Waiting for docker to be ready 🐋
 ✓ [kind-1-control-plane] Starting Kubernetes (this may take a minute) ☸
Cluster creation complete. You can now use the cluster with:

export KUBECONFIG="$(kind get kubeconfig-path --name="1")"
kubectl cluster-info
Kates-MacBook-Pro ~/audit-sink-quickstart (master) $
```

THIS IS STILL IN **ALPHA**

THIS IS STILL IN **ALPHA**

- increase in **CPU/ Memory**
- a **cluster admin** level privilege
- does **not support auth**
- **policy is scoped down** for now will be handled by a proxy

AGENDA

- ~~Who am I / Yes I work for VMWare now~~
- ~~What is Kubernetes Audit Logging & Why Should You Care~~
- ~~Configuring Audit~~
- **Making Sense of Audit**
- Very Cool Demo 10/10
- fin.

PROBLEM 2

MAKING **SENSE** OF AUDIT

GOBS OF **DATA**

GOBS OF
DATA

LIKE
SO MUCH
DATA

```
kubectl apply -f deployment.yaml
```

USER EVENTS

```
kubectl apply -f deployment.yaml
```

GET

CREATE

LIST

SYSTEM EVENTS

kubectl apply -f deployment.yaml

GET	CREATE	LIST	UPDATE	GET
UPDATE	CREATE	GET	UPDATE	GET
CREATE	UPDATE	UPDATE	GET	GET
UPDATE	CREATE	UPDATE	UPDATE	GET
CREATE	CREATE	CREATE	UPDATE	GET
GET	UPDATE	GET	LIST	UPDATE

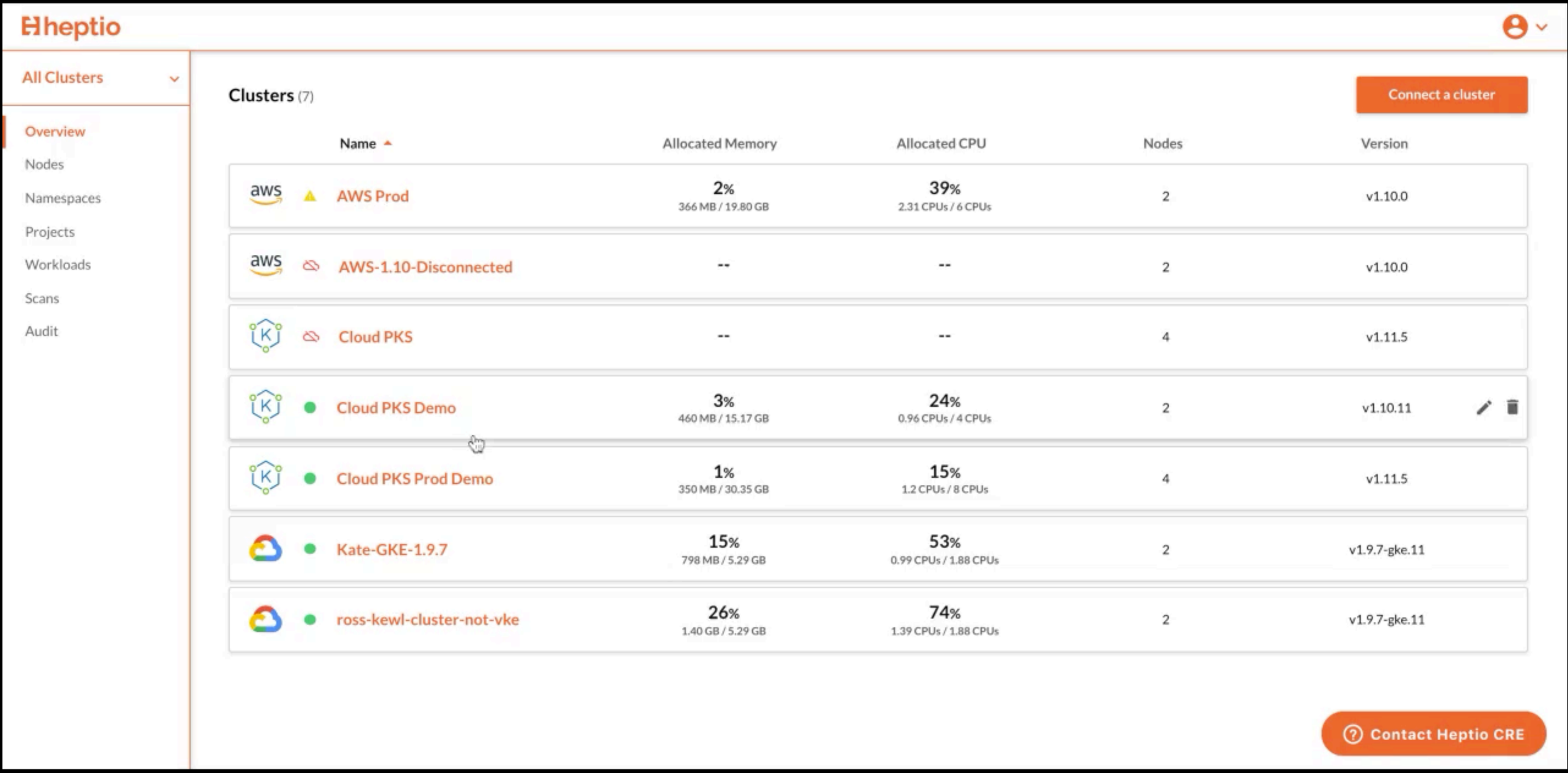
60,000 EVENTS PER **HOUR**

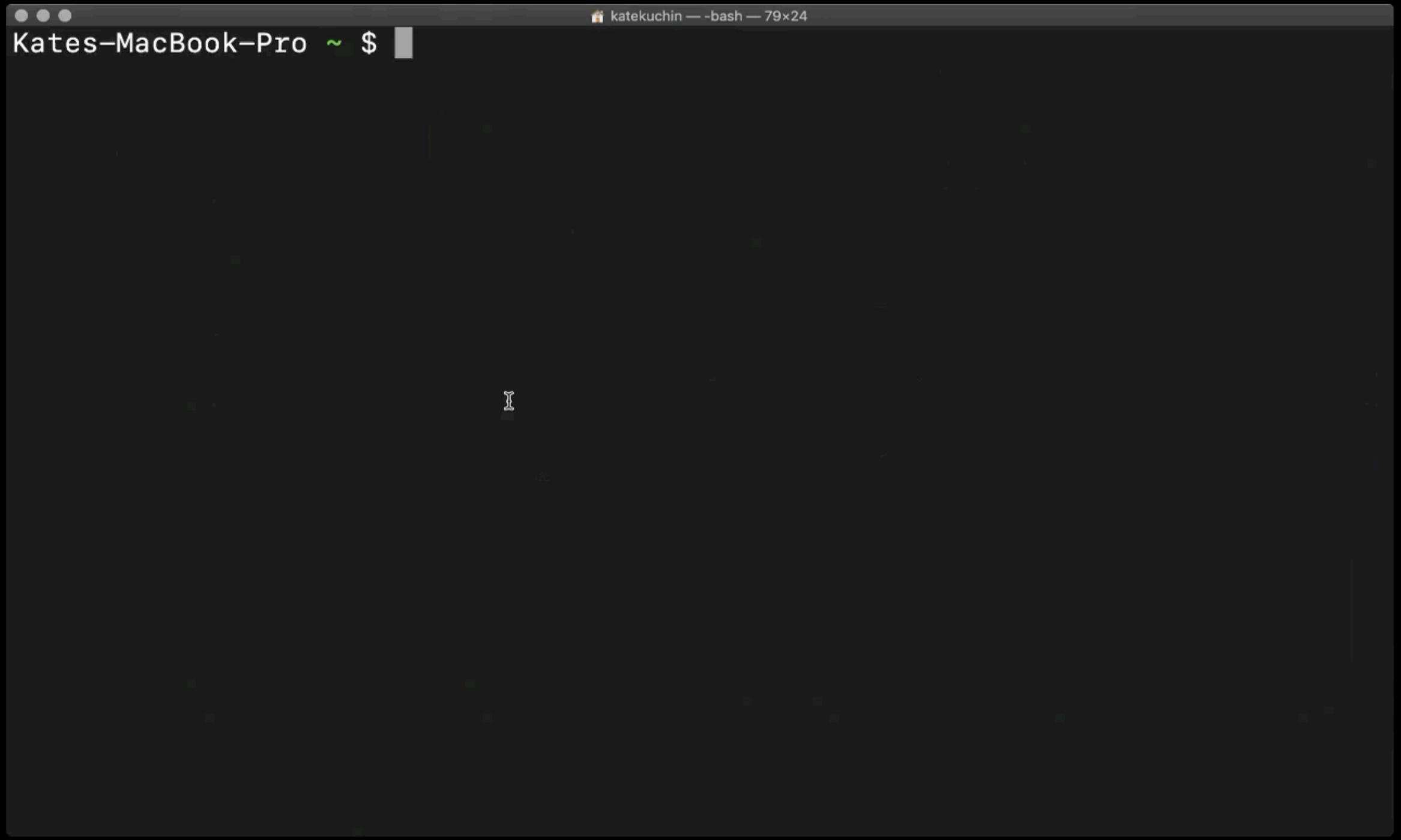
1.5 **MILLION** EVENTS PER **DAY**

10 **MILLION** EVENTS PER **WEEK**

43 **MILLION** EVENTS PER **MONTH**

DEMO


heptio
All Clusters
Overview
Nodes
Namespaces
Projects
Workloads
Scans
Audit
Clusters (7)
Connect a cluster
Name
Allocated Memory
Allocated CPU
Nodes
Version
AWS Prod
2%
366 MB / 19.80 GB
39%
2.31 CPUs / 6 CPUs
2
v1.10.0
AWS-1.10-Disconnected
--
--
2
v1.10.0
Cloud PKS
--
--
4
v1.11.5
Cloud PKS Demo
3%
460 MB / 15.17 GB
24%
0.96 CPUs / 4 CPUs
2
v1.10.11
Cloud PKS Prod Demo
1%
350 MB / 30.35 GB
15%
1.2 CPUs / 8 CPUs
4
v1.11.5
Kate-GKE-1.9.7
15%
798 MB / 5.29 GB
53%
0.99 CPUs / 1.88 CPUs
2
v1.9.7-gke.11
ross-kewl-cluster-not-vke
26%
1.40 GB / 5.29 GB
74%
1.39 CPUs / 1.88 CPUs
2
v1.9.7-gke.11
Contact Heptio CRE


katekuchin — -bash — 79×24
Kates-MacBook-Pro ~ $

heptio

AWS Prod

- Overview
- Nodes
- Namespaces
- Projects
- Workloads
- Scans
- Audit

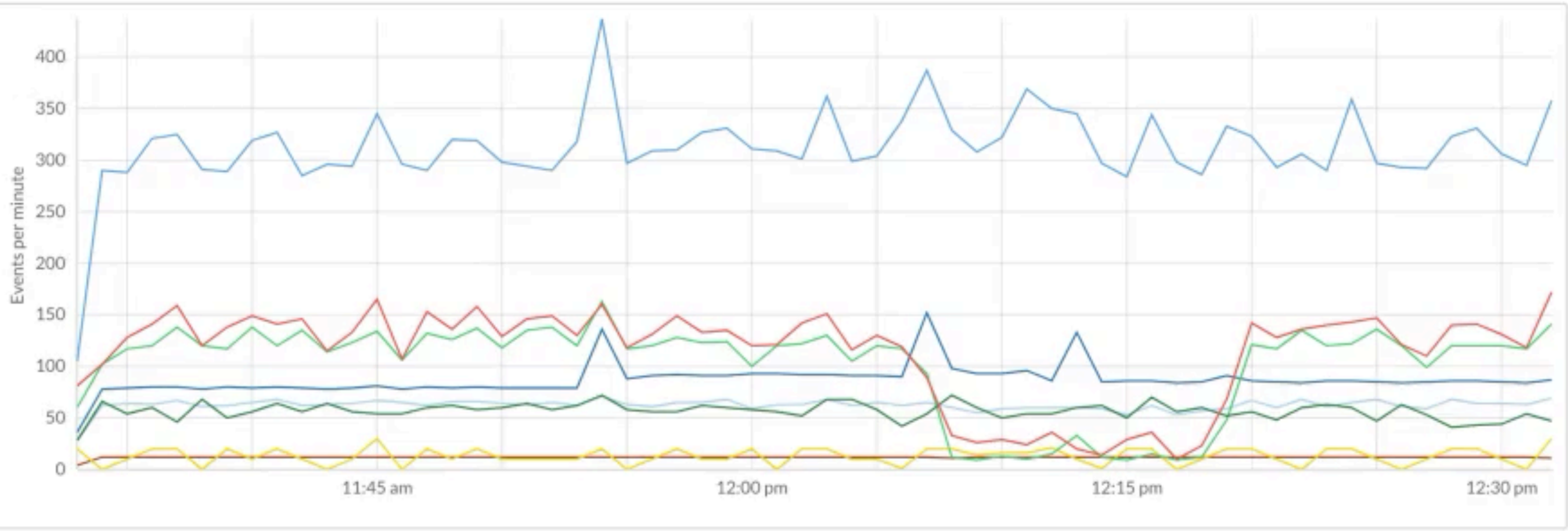
aws AWS Prod

Master	Location	Total Cores	Total Memory	Nodes	Pods
v1.10.0	us-west-2	6 CPUs	19.80 GB	2	126

Namespace | Resource | Name | Verb | User | Source IP | Failures only | System events | 1h 6h 1d 1w 1m Custom

Showing **all** events for the last **hour** (including system events)

- **38,847** Events
- **715** Patches
- **18,863** Gets
- **5,269** Updates
- **3,763** Lists
- **769** Deletions
- **6,056** Creations
- **3,412** Watches
- 6,796 Failures



Items per page: 100 1-100 of 10,000 (38,847 events total)

‹ 1 2 3 ... 100 ›

Time	Resource	Name	Verb	Namespace	User
2018-12-08T20:33:21Z	pods	heptio-updater-20181128-5678b75	get	heptio-hq	system:node:ip-10-0-100-113.us-west-2.com... 34.217.76.189

Contact Heptio CRE

AGENDA

- ~~Who am I / Yes I work for VMWare now~~
- ~~What is Kubernetes Audit Logging~~
- ~~Configuring Audit~~
- ~~Making Sense of Audit~~
- ~~Very Cool Demo 10/10~~
- **fin.**

THANK YOU

find me at the heptio booth or **@exkuchme** on twitter if you wanna talk audit or just say hi

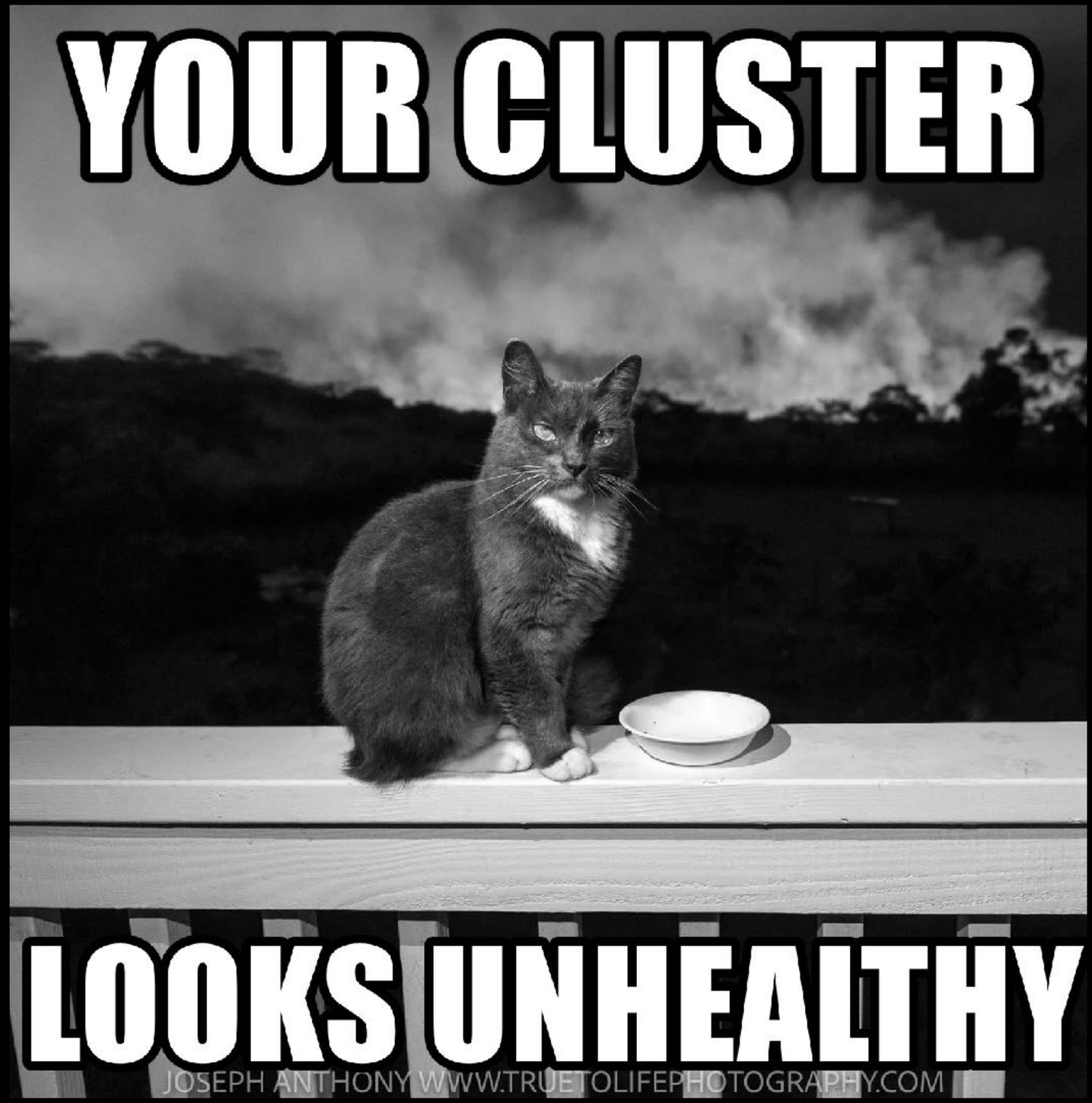


PHOTO BY RAOUL DROOG ON UNSPLASH

PHOTO CREDIT FOX SEARCHLIGHT

PHOTO BY JEAN WIMMERLIN ON UNSPLASH

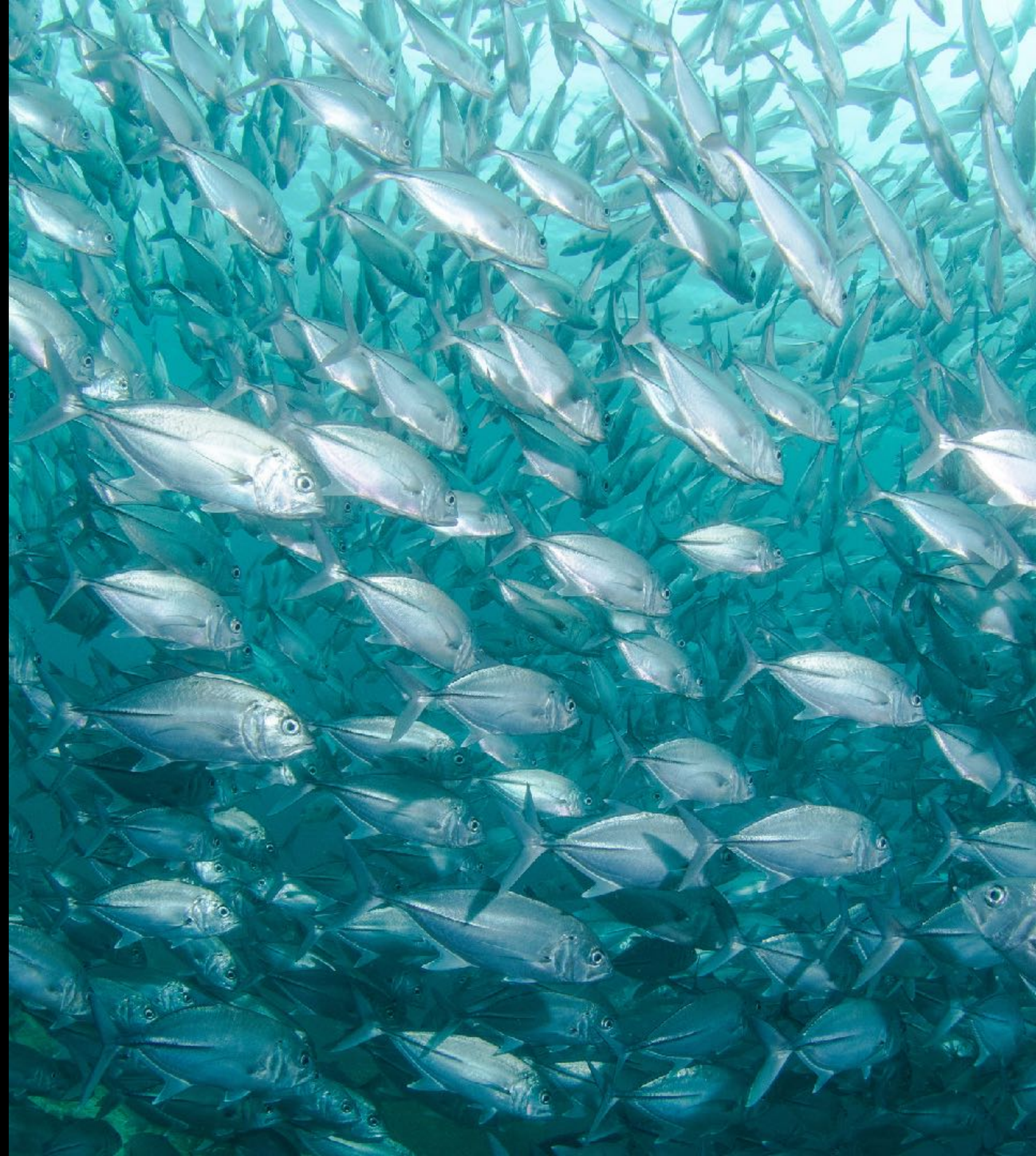


PHOTO BY MATT ARTZ ON UNSPLASH

PHOTO BY RAWPIXEL ON UNSPLASH

PHOTO BY JOHN PRATT ON UNSPLASH

PHOTO BY CYRUS LOPES ON UNSPLASH

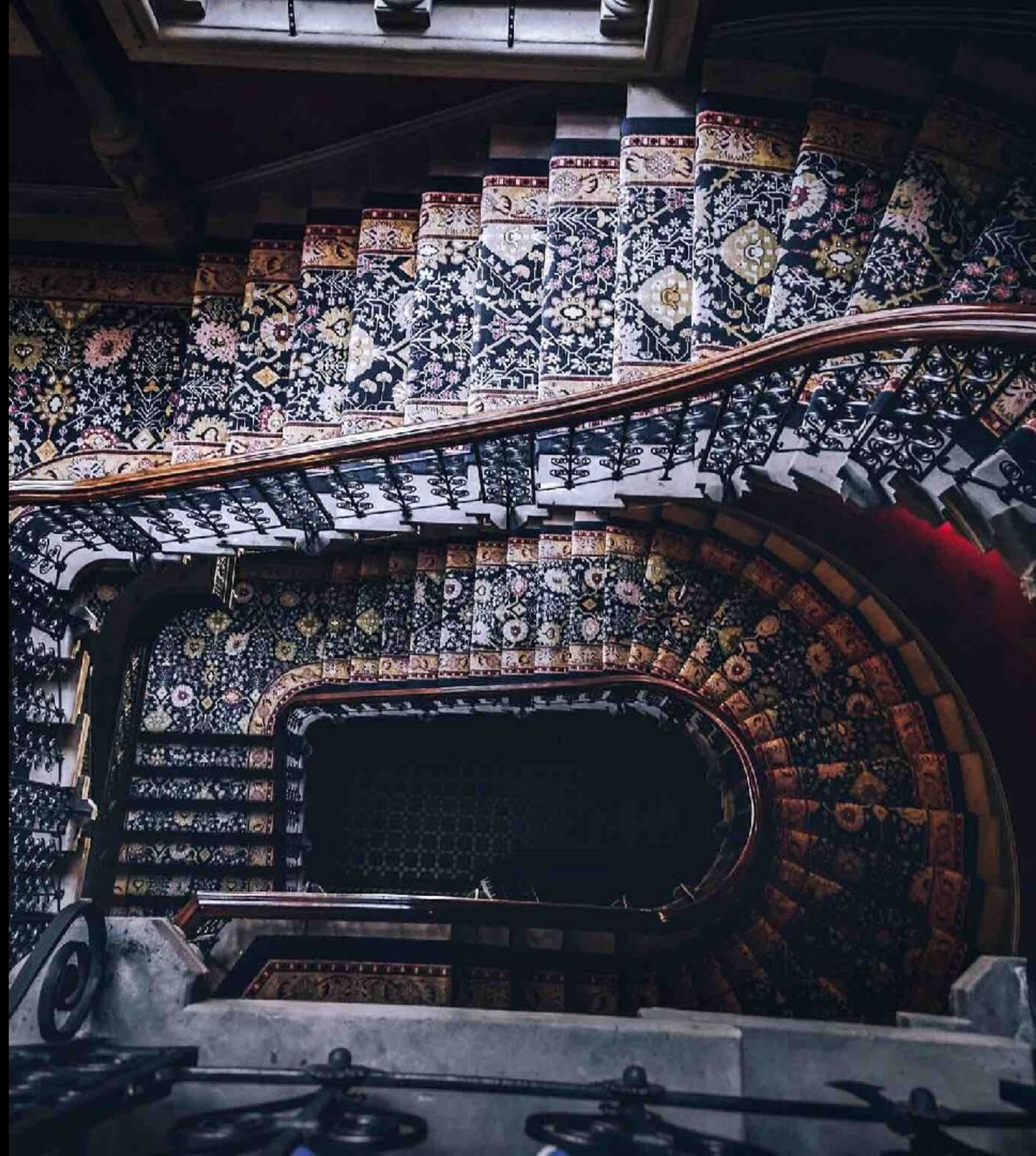


PHOTO BY NIKITA KOSTRYKIN ON UNSPLASH

PHOTO BY JOHN CARLISLE ON UNSPLASH

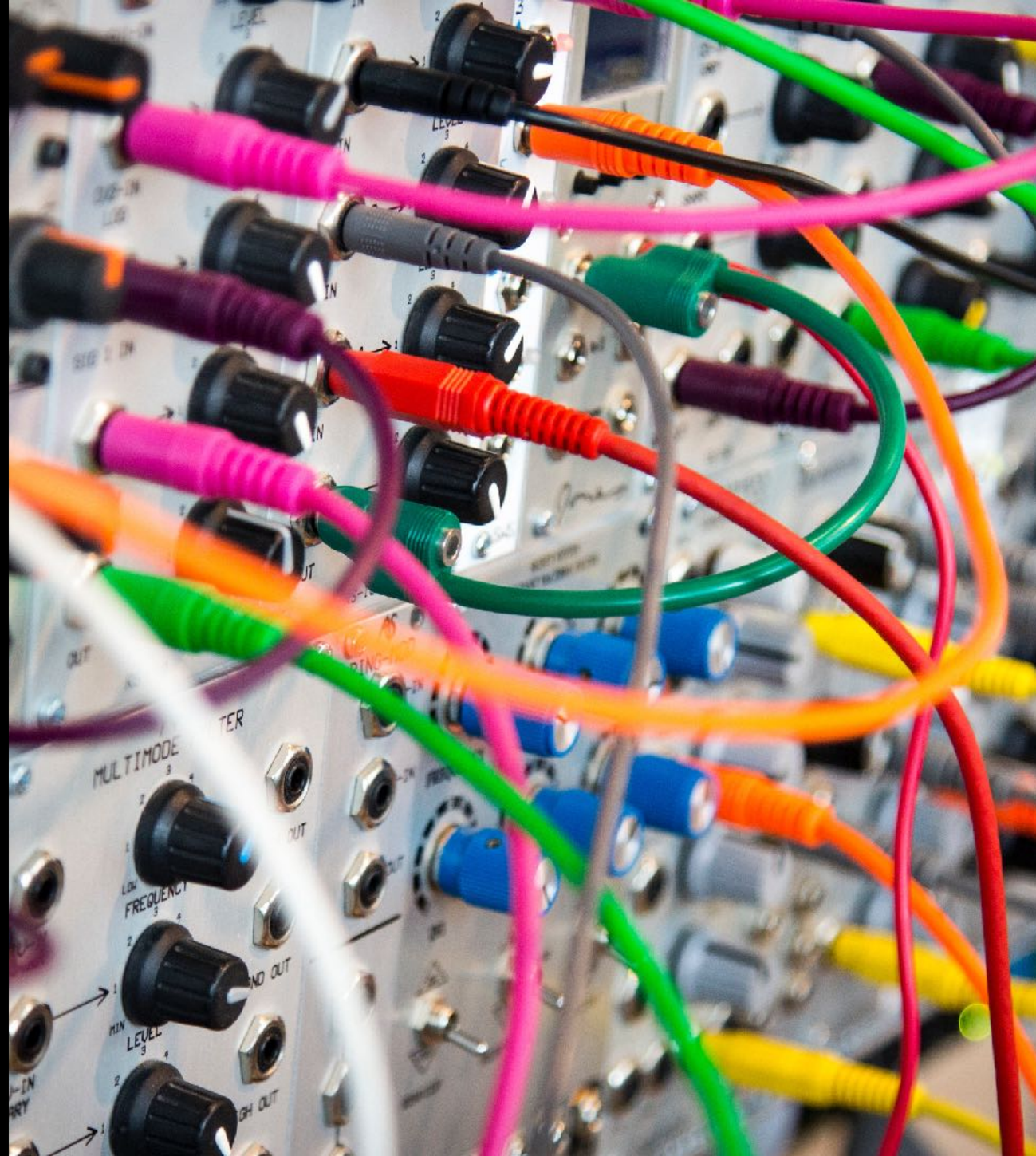


PHOTO BY YOMEX OWO ON UNSPLASH

PHOTO BY SAMUEL ZELLER ON UNSPLASH

PHOTO BY TODD QUACKENBUSH ON UNSPLASH

PHOTO BY TYLER NIX ON UNSPLASH

PHOTO BY LACHLAN DONALD ON UNSPLASH

PHOTO BY BRIANNA SANTELLAN ON UNSPLASH

PHOTO BY MICAH WILLIAMS ON UNSPLASH

PHOTO BY FILIP MROZ ON UNSPLASH

AGENDA

- ~~Who am I / Yes I work for VMWare now~~
- ~~What is Kubernetes Audit Logging~~
- ~~Configuring Audit~~
- ~~Making Sense of Audit~~
- **Very Cool Demo 10/10**
- fin.