Jimmy Lu

Software Engineer @hyvesolutions

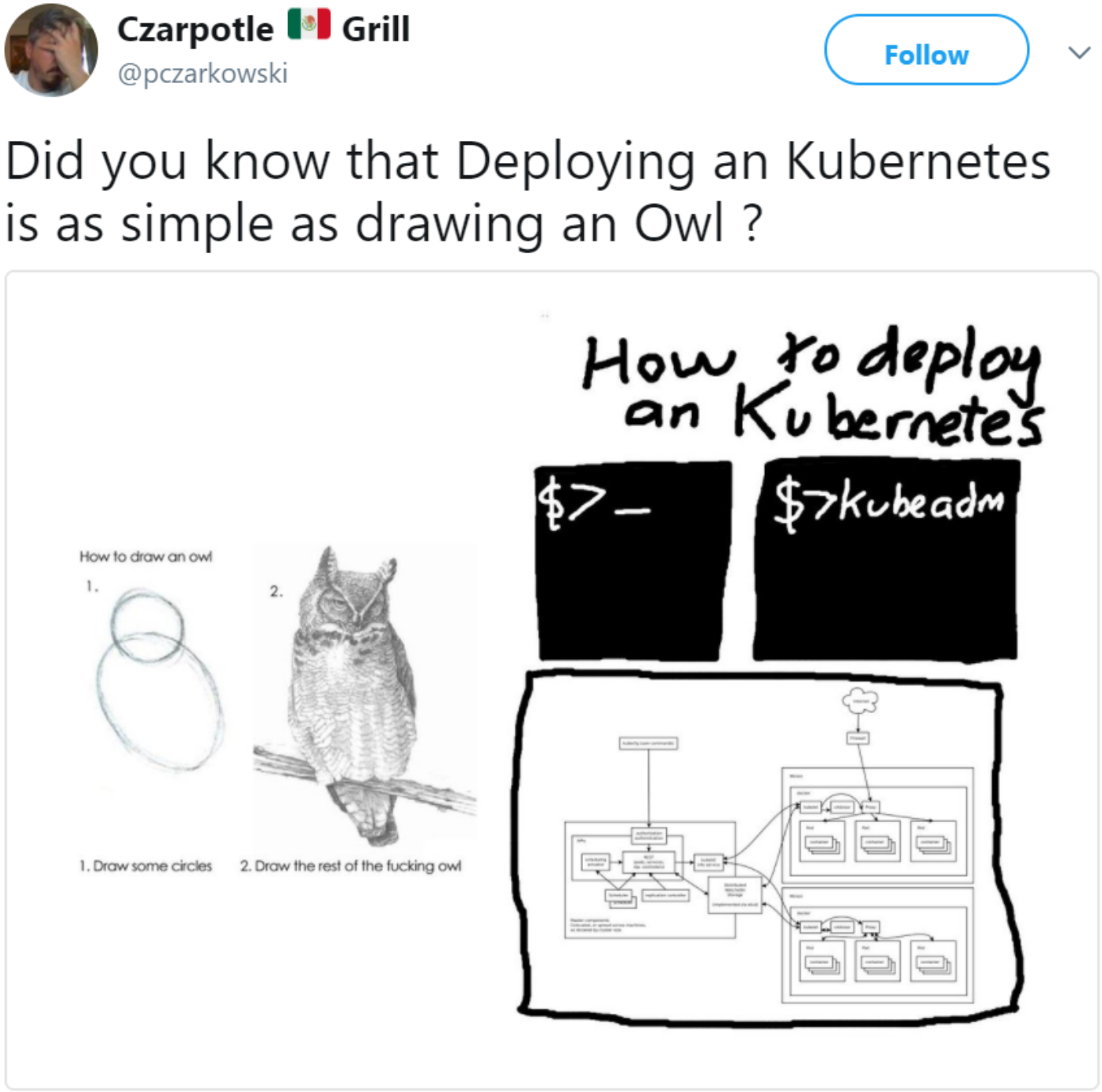
jimmylu@hyvesolutions.com

All The Troubles You Get Into When Setting up a Production-ready Kubernetes Cluster

Agenda

- Motivation
- Recap of Kubernetes Architecture
- Security
- Networking
- Miscellaneous
- High Availability

Motivation

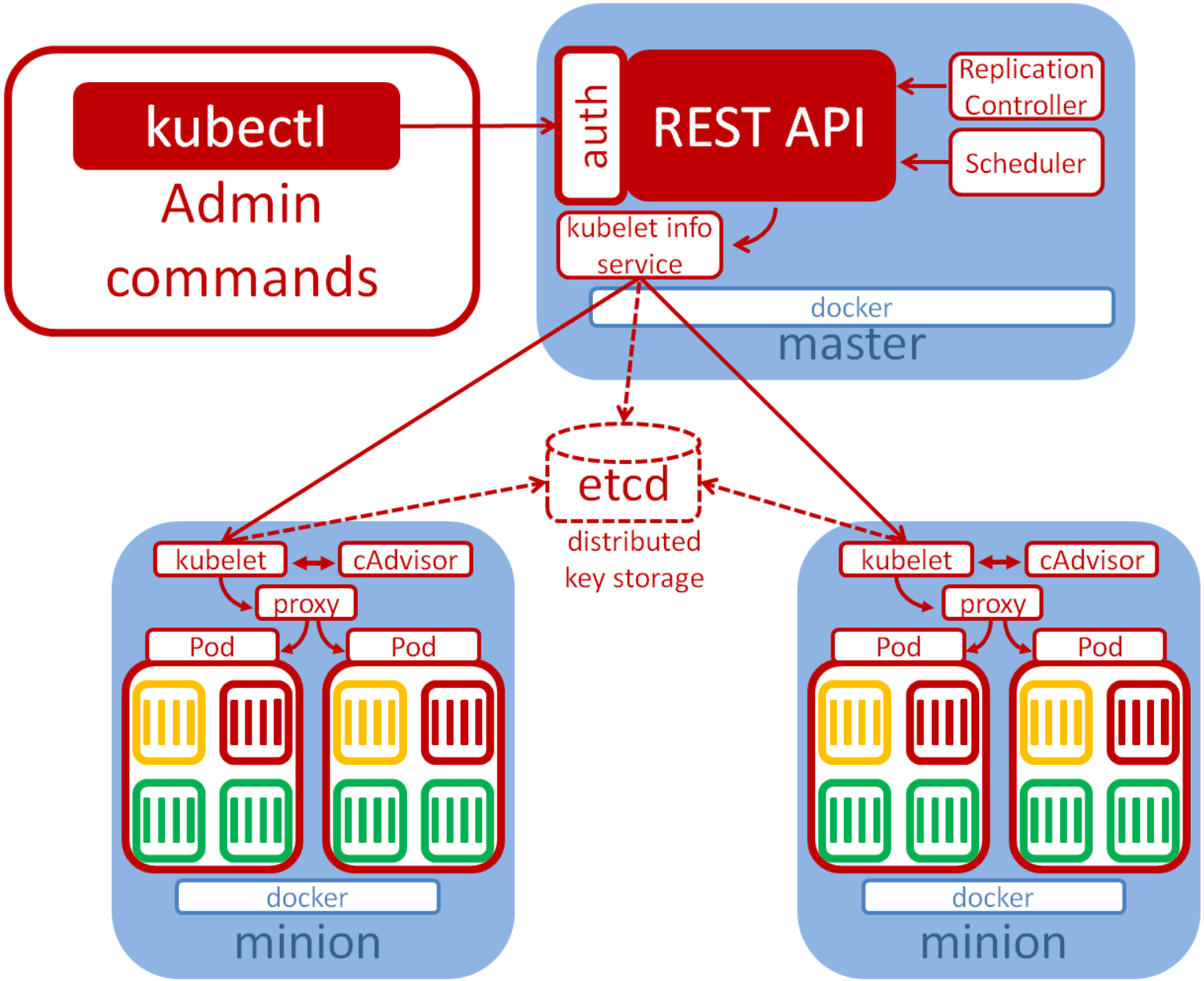

Czarpotle Grill
@pczarkowski
Follow
Did you know that Deploying an Kubernetes is as simple as drawing an Owl ?
How to draw an owl
1.
2.
1. Draw some circles
2. Draw the rest of the fucking owl
How to deploy an Kubernetes
$7 _
$7 kubeadm

Motivation

- A Million Ways of Deploying a Kubernetes Cluster – DevOpsDays 2017
  - https://goo.gl/5yHFHa
- We tried to build our own solutions – Kubewizard
  - Large Clusters
  - Configurable/Customizable
  - Easy to Use
  - Fast
  - Production Ready
- We wanna save your precious time, keeping you out of troubles

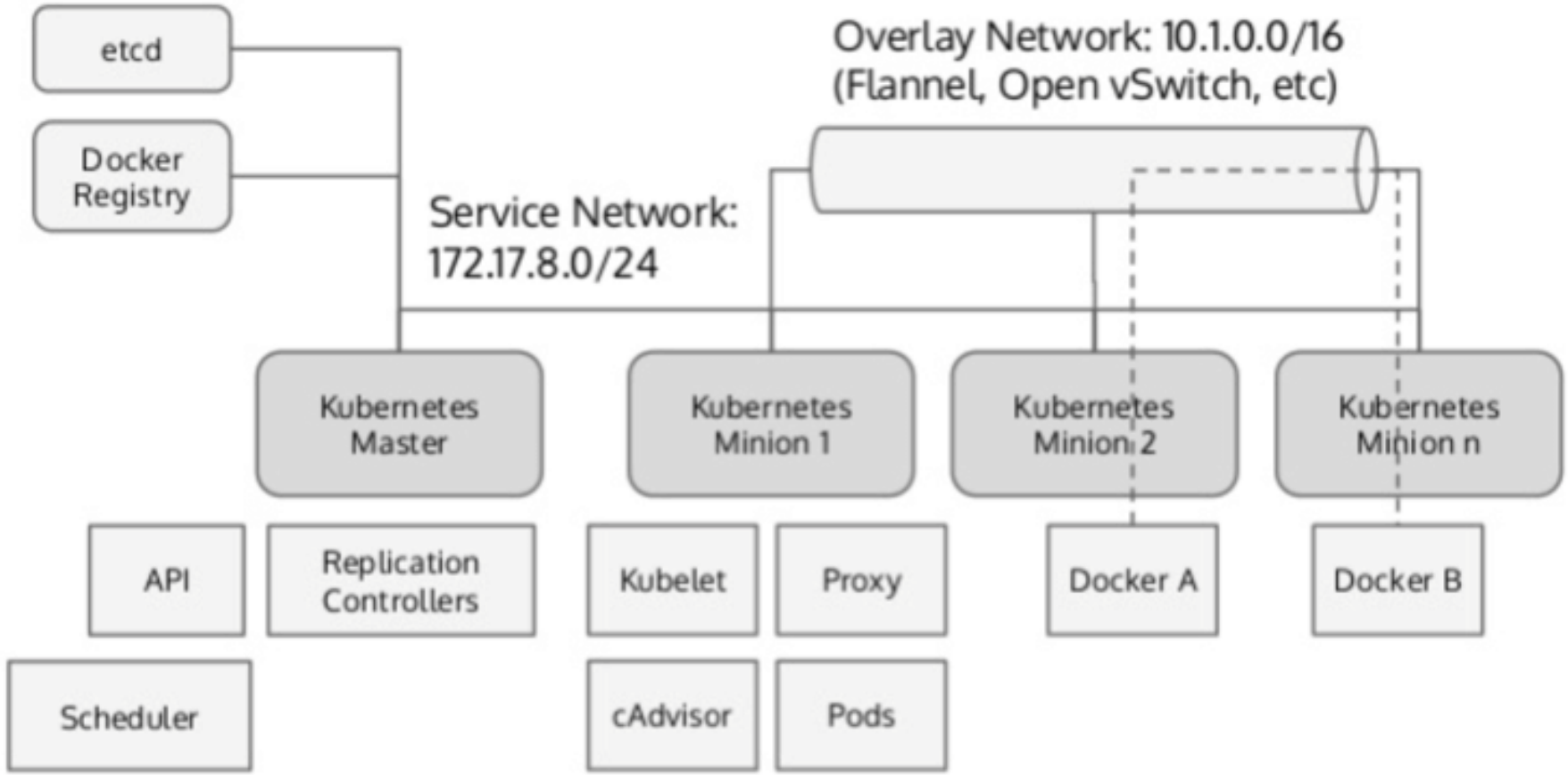
Recap of Kubernetes Architecture

Architecture Recap

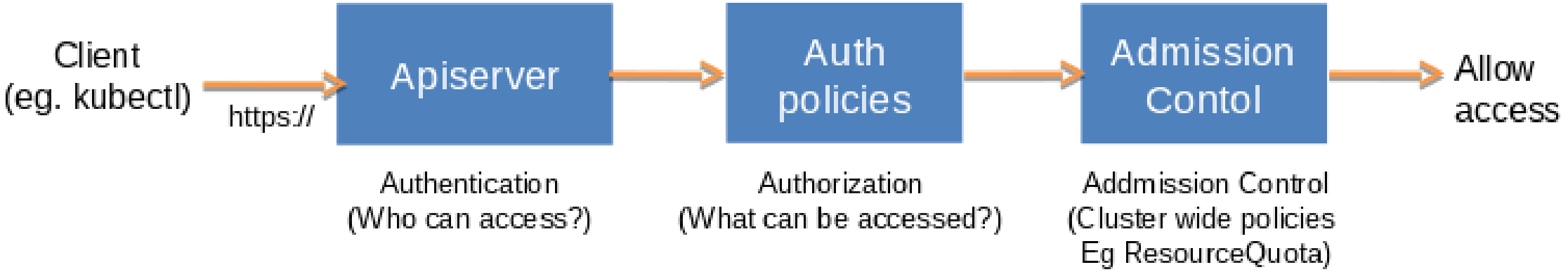


Architecture Recap – Network

Kubernetes Architecture



Architecture Recap – Authentication/Authorization



Clinical Cases

Case Study Sample

01

Symptom

02

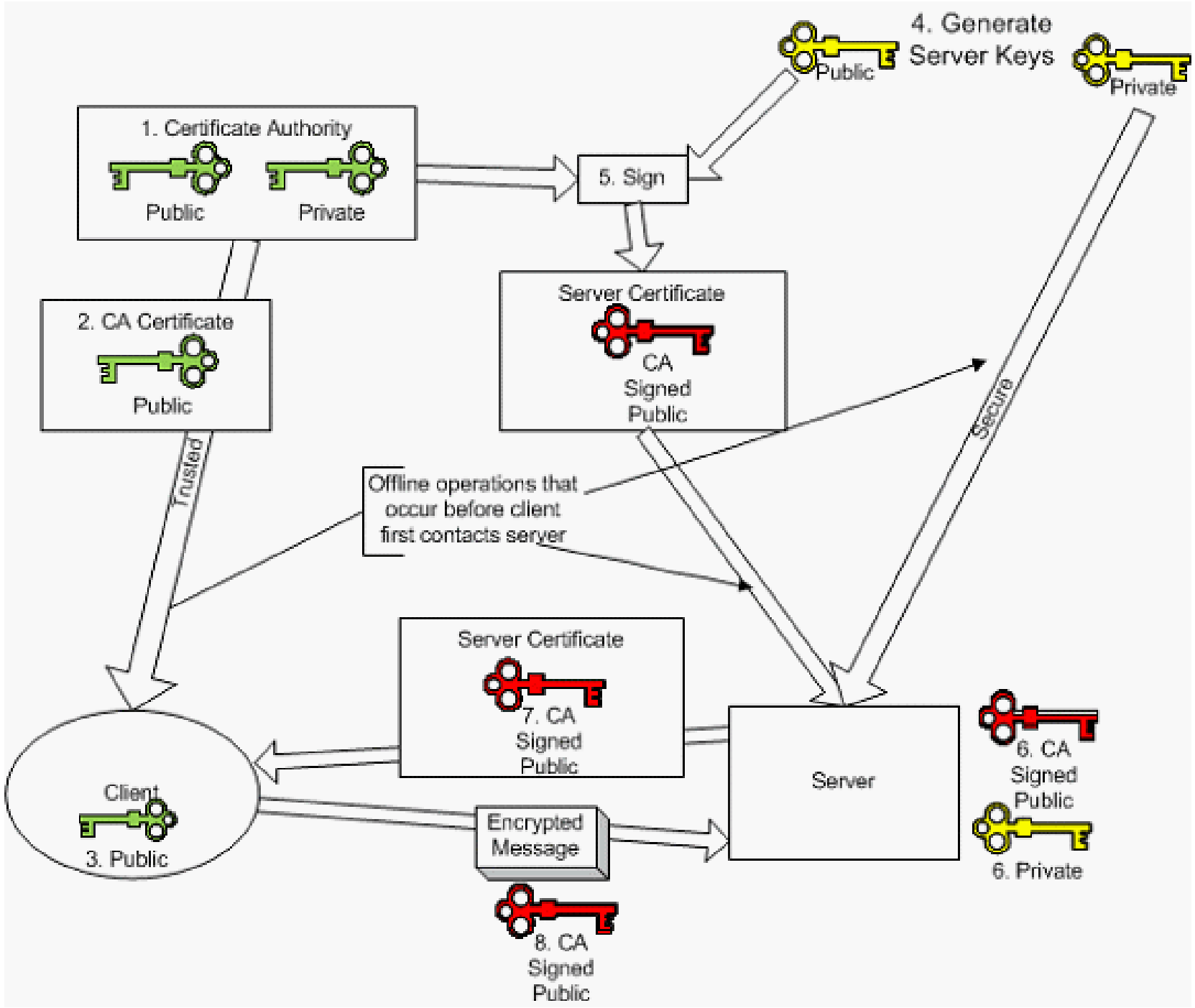
Diagnosis

03

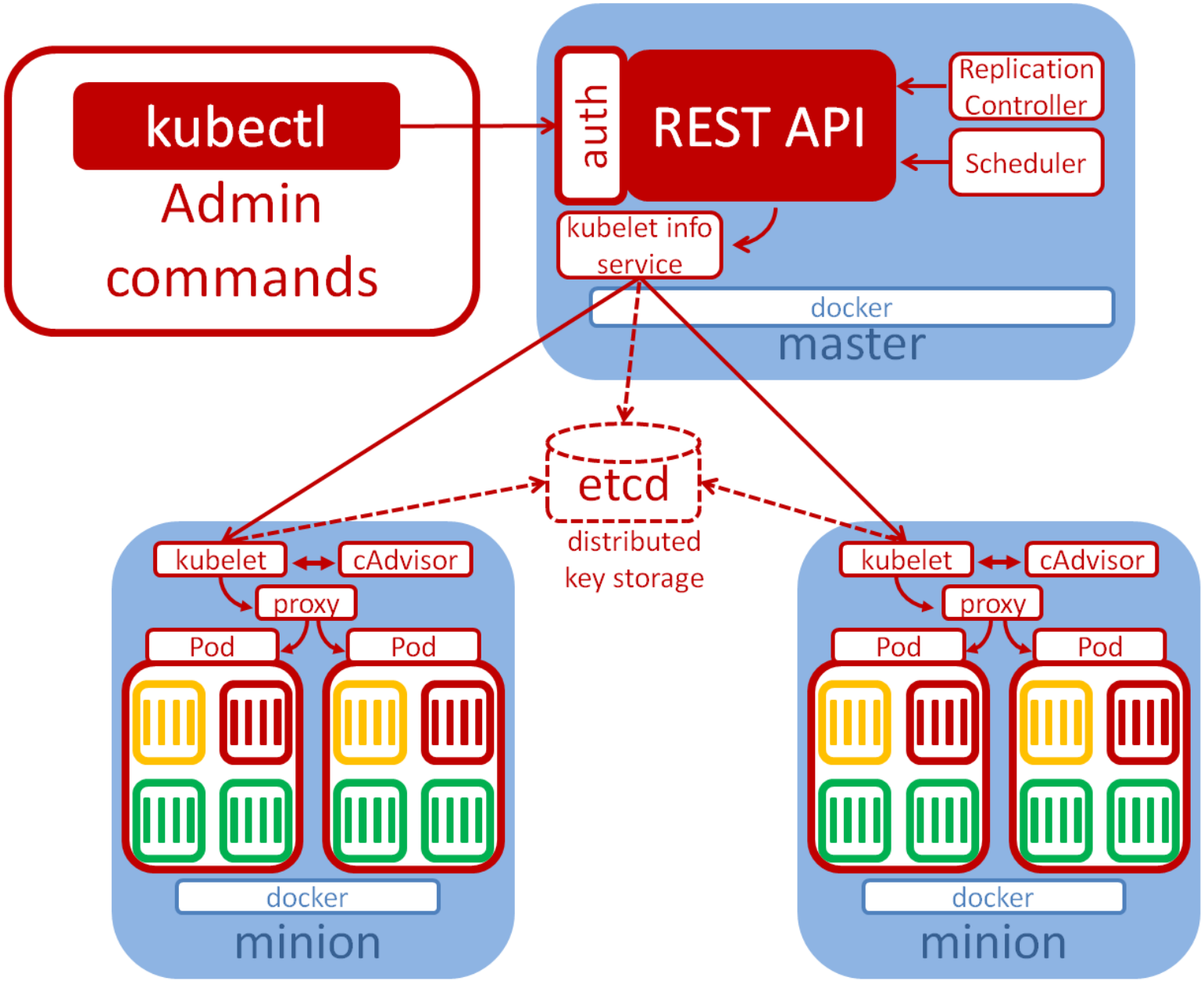
Therapy

Security

SSL/TLS Recap



Architecture Recap



Security – SSL/TLS

- **Symptom**: Unable to connect to the server: x509: certificate signed by unknown authority
- **Diagnosis**: Check the CA data in .kubeconfig file
- **Therapy**: Make your client CA data identical to the CA file assigned to the server
  - *--client-ca-file* of kube-apiserver and kubelet, *--trusted-ca-file* and *--trusted-ca-file* of etcd, and the CA of your authentication proxy

```
clusters:
- cluster:
    certificate-authority-data: LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0t……
    server: https://xxx.xxx.xxx.xxx:6443
  name: kw
contexts:
- context:
    cluster: kw
    namespace: default
    user: kw-admin
  name: kw
current-context: kw
kind: Config
preferences: {}
users:
- name: kw-admin
  user:
    client-certificate-data: LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tL……
    client-key-data: LS0tLS1CRUdJTiBSU0EgUFJJVkFURSBLRVktLS0tLQ……
```

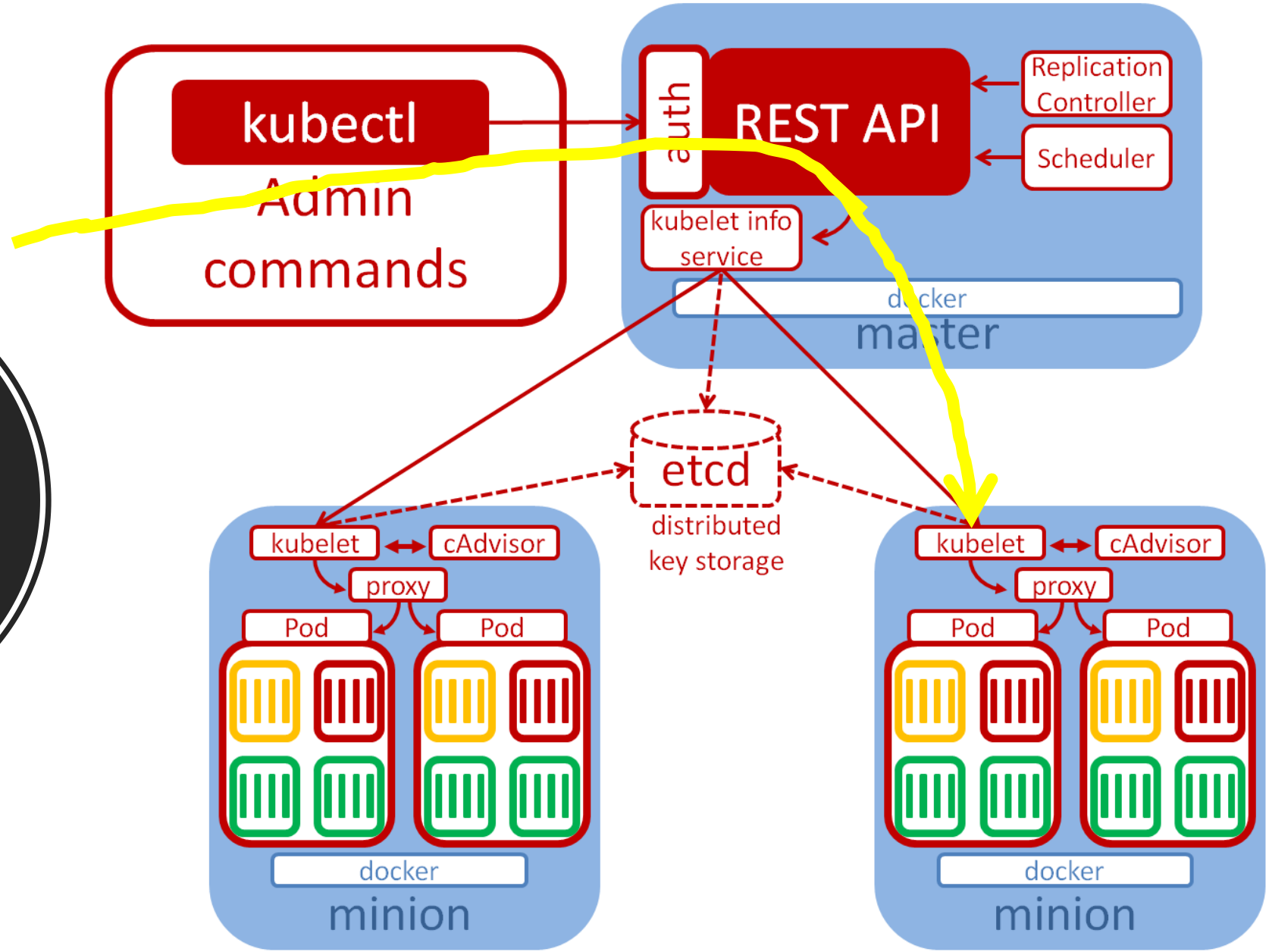
Security – SSL/TLS

- **Symptom**: You try to get the logs via '*kubectl logs $(pod_name)*' but encounter x509: certificate signed by unknown authority

- **Diagnosis**: Check if kubelet client CA assigned to kube-apiserver is correct

- **Therapy**: Make your kubelet client CA assigned to kube-apiserver via --kubelet-certificate-authority matches what is assigned to kubelets

Architecture Recap



Security – SSL/TLS

- **Symptom**: Unable to connect to the server: x509: certificate is valid for 10.240.0.4, 10.240.0.5, 35.194.148.244
- **Diagnosis**: Check IPs and domains in the **hosts** part of the certificate request
- **Therapy**: Make sure all the IPs and domains are included in your certificate request file when generating the server certificates

```
{
  "CN": "kube-apiserver",
  "hosts": [
    "kw-master-001",
    "kw-master-002",
    "10.240.0.4",
    "10.240.0.5",
    "35.194.148.244",
    "35.201.222.64",
    "35.201.171.127",
    "10.96.0.1",
    "127.0.0.1",
    "kubernetes",
    "kubernetes.default",
    "kubernetes.default.svc",
    "kubernetes.default.svc.cluster.local"
  ],
  "key": {
    "algo": "rsa",
    "size": 2048
  },
  "names": [{}]
}
```

hostname

internal ip

external ip

load balancer

cluster ip

Kube-DNS domain

Security – Authentication

- **X509 Client Certs**
- Static Token File
- **Bootstrap Tokens**
- Static Password File
- **Service Account Tokens**
- OpenID Connect Tokens
- Webhook Token Authentication
- **Authenticating Proxy**
- Keystone Password
- Anonymous requests
- https://kubernetes.io/docs/admin/authentication/

Security – Authentication

- **Symptom**: tls: failed to find any PEM data in certificate (or key) input
- **Diagnosis**: Check the certificate and key data in .kubeconfig file
- **Therapy**: Make sure the certificate and key data are correctly signed by the CA you created

```
clusters:
- cluster:
    certificate-authority-data: LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0t……
    server: https://xxx.xxx.xxx.xxx:6443
  name: kw
contexts:
- context:
    cluster: kw
    namespace: default
    user: kw-admin
  name: kw
current-context: kw
kind: Config
preferences: {}
users:
- name: kw-admin
  user:
    client-certificate-data: LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tL……
    client-key-data: LS0tLS1CRUdJTiBSU0EgUFJJVkFURSBLRVktLS0tLQ……
```

Security – RBAC Authorization

- **Symptom**: Error from server (Forbidden): User "kubernetes-admin" cannot list nodes at the cluster scope. (get nodes)
- **Diagnosis**: Check clusterrole, clusterrolebinding, role, rolebinding, and the **user:group** values in the kubeconfig, token, or http-headers
- **Therapy**: Create corresponding roles and rolebindings for the users, groups, or service accounts

certificate request

```
{
  "CN": "system:node:kw-etcd-001",
  "names": [
    {
      "O": "system:nodes"
    }
  ]
}
```

authentication token

2915baa1f710cbada00aad86706ded28,kubelet-bootstrap,10001,"system:kubelet-bootstrap"

configmap

```
 clusters:
    - cluster:
        certificate-authority:
/var/run/secrets/kubernetes.io/serviceaccount/ca.crt
        server: https://10.96.0.1:443
      name: default
    contexts:
    ……
    users:
    - name: default
      user:
        tokenFile: /var/run/secrets/kubernetes.io/serviceaccount/token
```

Security – RBAC Authorization

- Affects to all the components connecting to apiserver
- **Symptom**:
  - Failed to list *v1.Pod, *v1.Node, etc. in the logs of all the components that talk to apisrever
  - kube-proxy – requests to the service cannot be proxied to the endpoints
  - kube-dns – domain name cannot be resolved
  - kubelet – nodes cannot join the cluster
  - overlay network – cannot assign IPs to pods thus no traffic to the pods
  - kube-controller-manager – primary features of Kubenretes are malfunctioned

Networking

Networking

- **Symptom**: Nodes are in NotReady state when ‘kubectl get nodes’
- **Diagnosis**: Verify if overlay networks work
- **Therapy**: Install CNI and CNI-plugins and make sure they work as expected

Networking

- **Symptom**: All traffic between pods, or between nodes and pods are dropped
- **Diagnosis**: Look at rules of iptables and routing tables
- **Therapy**: Allow packet forward or downgrade to docker v1.12.x

$ sudo iptables-save

-A INPUT -j KUBE-FIREWALL

-A FORWARD -j DOCKER-ISOLATION

-A FORWARD -o docker0 -j DOCKER

-A FORWARD -o docker0 -m conntrack --ctstate RELATED,ESTABLISHED -j ACCEPT

-A FORWARD -i docker0 ! -o docker0 -j DROP

-A FORWARD -i docker0 -o docker0 -j DROP

-A OUTPUT -j KUBE-FIREWALL

-A DOCKER-ISOLATION -j RETURN

Miscellaneous

Miscellaneous

- **Symptom**: Kubernetes components are signal to stop periodically
- **Diagnosis**: Check the configuration of liveness probe and readiness probe.
- **Therapy**: Make sure the host, port, scheme match the health-check targets. Also, make sure your applications are in the good states

kube-controller-manager.yaml

```
livenessProbe:
    failureThreshold: 8
    httpGet:
      host: 127.0.0.1     # default to pod's IP
      path: /healthz
      port: 10252
      scheme: HTTP
    initialDelaySeconds: 15
    timeoutSeconds: 15
```

Miscellaneous

- **Symptom**: Cannot create cloud load balancer or dynamic persistent volumes automatically
- **Diagnosis**: Look at *--clud-provider* argument of kube-apiserver, kube-controller-manager, and kubelet
- **Therapy**: Enable cloud integration by giving correct values to the *--cloud-provider* argument

Miscellaneous

- **Symptom**: TLS certificate request cannot work with static pods, --run_once does not help solving the issue because it’s broken.

- **Therapy**: Either applies TLS certificate request or static pods, not both

High Availability

High Availability

- **Symptom**: controller.go:290] Resetting endpoints for master service "kubernetes" to...'
- **Diagnosis**: Look at the *--apiserver-count* argument to see if it matches the actual number of apiservers
- **Therapy**: Correct the value of the --apisrever-count argument
  - https://stackoverflow.com/questions/36337431/kubernetes-newer-api-server-shows-errors-resetting-endpoints-for-master-service

High Availability

- **Symptom**: attempting to acquire leader lease... keep showing in the logs of kube-controller-managers and kube-schedulers
- **Diagnosis**: Check to see if '**successfully acquired lease…**' appears in one of the logs of kube-controller-managers and kube-schedulers
- **Therapy**: No action needed

High Availability

- Use monit or systemctl to watch over kubelet and docker
- Let the health check of external load balancer hits against insecure port of kube-apiserver *(--insecure-port*, *--insecure-bind-address*)
- Load balancer may sometimes may aggravate the issues. E.g. some apiservers are in the good status, some are not.
- etcd, kube-apiserver, kube-controller-manager, kube-scheduler, kubelet should all set to be high available

Kubewizard

Summary

- Setting up a distributed system is never easy, especially the complex system like Kubernetes
- Some suggestions
  - Be patient
  - Step-by-step, reduce the number of control factors to a minimum
  - Start from a small cluster, then to a HA cluster, then a large cluster
  - kubectl logs, kubectl describe, systemctl status, journalctl -xe, docker logs, minikube, and kubeadm, Kubernetes-the-hard-way are your good friends
  - RTFM (Read the Documents)

Q&A