Maintaining, Monitoring and, Troubleshooting Kubernetes

MAINTAINING KUBERNETES CLUSTERS

Anthony E. Nocentino
ENTERPRISE ARCHITECT @ CENTINO SYSTEMS
@nocentino   www.centinosystems.com

Course Overview

Maintaining Kubernetes Clusters

Logging and Monitoring in Kubernetes Clusters

Troubleshooting Kubernetes Clusters

Summary

etcd backup and restore operations

Upgrading an existing cluster

Worker Node maintenance

High availability cluster topologies

Introducing etcd

Key value datastore

Stores cluster state data and objects

Backup and Restore

High Availability

Backing up etcd

Backup with snapshot using **`etcdctl`**

Secured and/or encrypted to protect sensitive information stored

Copied offsite as soon as possible

Schedule backups as a **`CronJob`**

Default data directory

- **`/var/lib/etcd`**
- **`hostPath`** mounted into a Pod

Getting etcdctl

Download from GitHub

Exec into an etcd Pod

Start a container

Backing up etcd with etcdctl

```
ETCDCTL_API=3 etcdctl --endpoints=https://127.0.0.1:2379 \
    --cacert=/etc/kubernetes/pki/etcd/ca.crt \
    --cert=/etc/kubernetes/pki/etcd/server.crt \
    --key=/etc/kubernetes/pki/etcd/server.key \
    snapshot save /var/lib/dat-backup.db

ETCDCTL_API=3 etcdctl --write-out=table \
    snapshot status /var/lib/dat-backup.db
```

Single Server
Pod-based etcd

Restoring etcd with etctl

Restore backup to another location

Move the original data out of the way

Stop etcd

Move the restored data to /var/lib/etcd

Kubelet will restart etcd

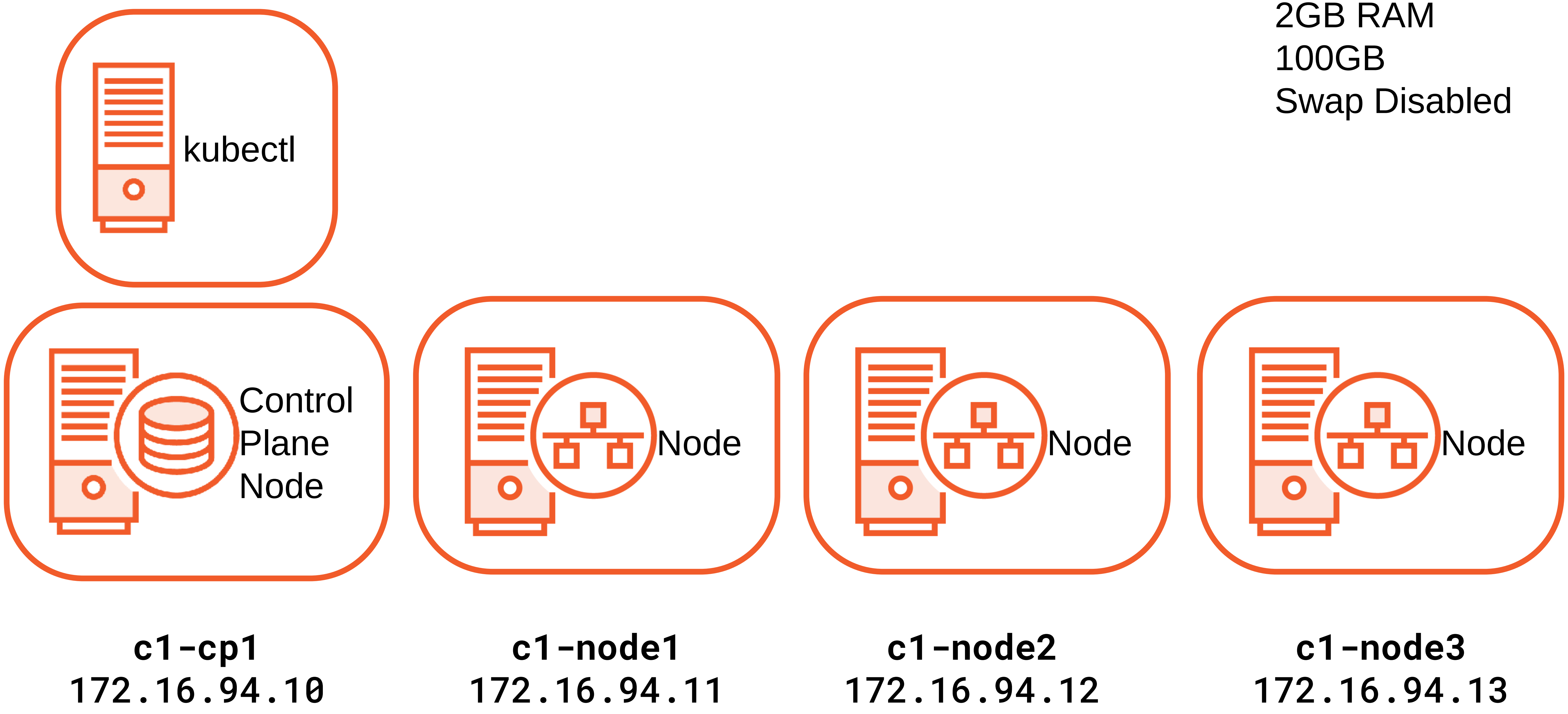
Restoring etcd with etctl

```
ETCDCTL_API=3 etcdctl snapshot restore /var/lib/dat-backup.db

mv /var/lib/etcd /var/lib/etcd.OLD

sudo crictl --runtime-endpoint
unix:///run/containerd/containerd.sock ps

sudo crictl --runtime-endpoint
unix:///run/containerd/containerd.sock stop $CONTAINER_ID

mv ./default.etcd /var/lib/etcd
```

Hostnames set
Host file on each

Lab Environment

Ubuntu 18.0.4
VMware Fusion VMs
2vCPU
2GB RAM
100GB
Swap Disabled



kubectl

Control Plane Node

Node

Node

Node

c1-cp1
172.16.94.10

c1-node1
172.16.94.11

c1-node2
172.16.94.12

c1-node3
172.16.94.13

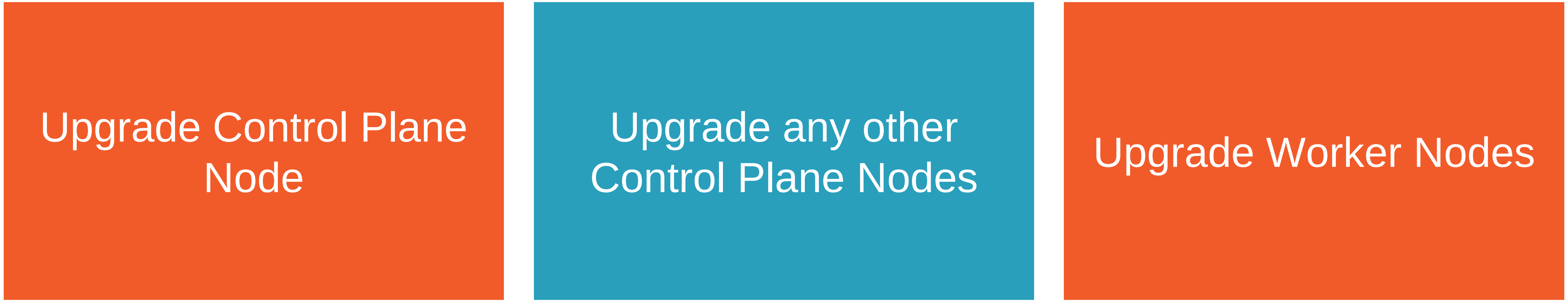
Demo

Investigating etcd and its configuration

Backing up etcd with etcdctl

Restoring etcd with etcdctl

Cluster Upgrade Process Overview



Upgrade Control Plane Node

Upgrade any other Control Plane Nodes

Upgrade Worker Nodes

Upgrading kubeadm-based Clusters

Static Pod based Control Plane

You can only upgrade minor versions

1.17 -> 1.18

1.16 X 1.18

Read the Release Notes

https://kubernetes.io/docs/setup/release/notes/

Cluster Upgrade Process - Control Plane

Update kubeadm package

Drain the Control Plane/Master Node

kubeadm upgrade plan

kubeadm upgrade apply

Uncordon the Control Plane/Master Node

Update kubelet and kubectl

kubeadm upgrade node

Cluster Upgrade Process - Control Plane

```
sudo apt-mark unhold kubeadm
sudo apt-get update
sudo apt-cache policy kubeadm
sudo apt-get install kubeadm=$TARGET_VERSION
sudo apt-mark hold kubeadm

kubectl drain c1-cp1 --ignore-daemonsets

sudo kubeadm upgrade plan

sudo kubeadm upgrade apply v$TARGET_VERSION

kubectl uncordon c1-cp1
```

Cluster Upgrade Process - Control Plane

```
sudo apt-mark unhold kubelet kubectl
sudo apt-get update
sudo apt-get install -y kubelet=$TARGET_VERSION kubectl=$TARGET_VERSION
sudo apt-mark hold kubelet kubectl
```

Cluster Upgrade Process - Worker Nodes

Update kubeadm

Drain the Node

kubeadm upgrade node

Update kubelet and kubectl

Uncordon Node

Cluster Upgrade Process - Worker Node

```
kubectl drain c1-node1 ––ignore-daemonsets

sudo apt-mark unhold kubeadm
sudo apt-get update
sudo apt-get install -y kubeadm=$TARGET_VERSION
sudo apt-mark hold kubeadm

sudo kubeadm upgrade node

sudo apt-mark unhold kubelet kubectl
sudo apt-get update
sudo apt-get install -y kubelet=$TARGET_VERSION kubectl=$TARGET_VERSION
sudo apt-mark hold kubelet kubectl

kubectl uncordon c1-node1
```

Demo

Upgrading an existing cluster

- Control Plane
- Worker Nodes

Worker Node Maintenance

- OS Updates and hardware upgrades
- Drain/Cordon the Node
  - `kubectl drain NODE_NAME`
- Marks the Node Unschedulable
- Gracefully terminates Pods
- Reboot the Node
  - Pod Eviction Timeout
- Keep resources in mind…memory and CPU

HA Cluster Architecture Overview

Keeping
applications online

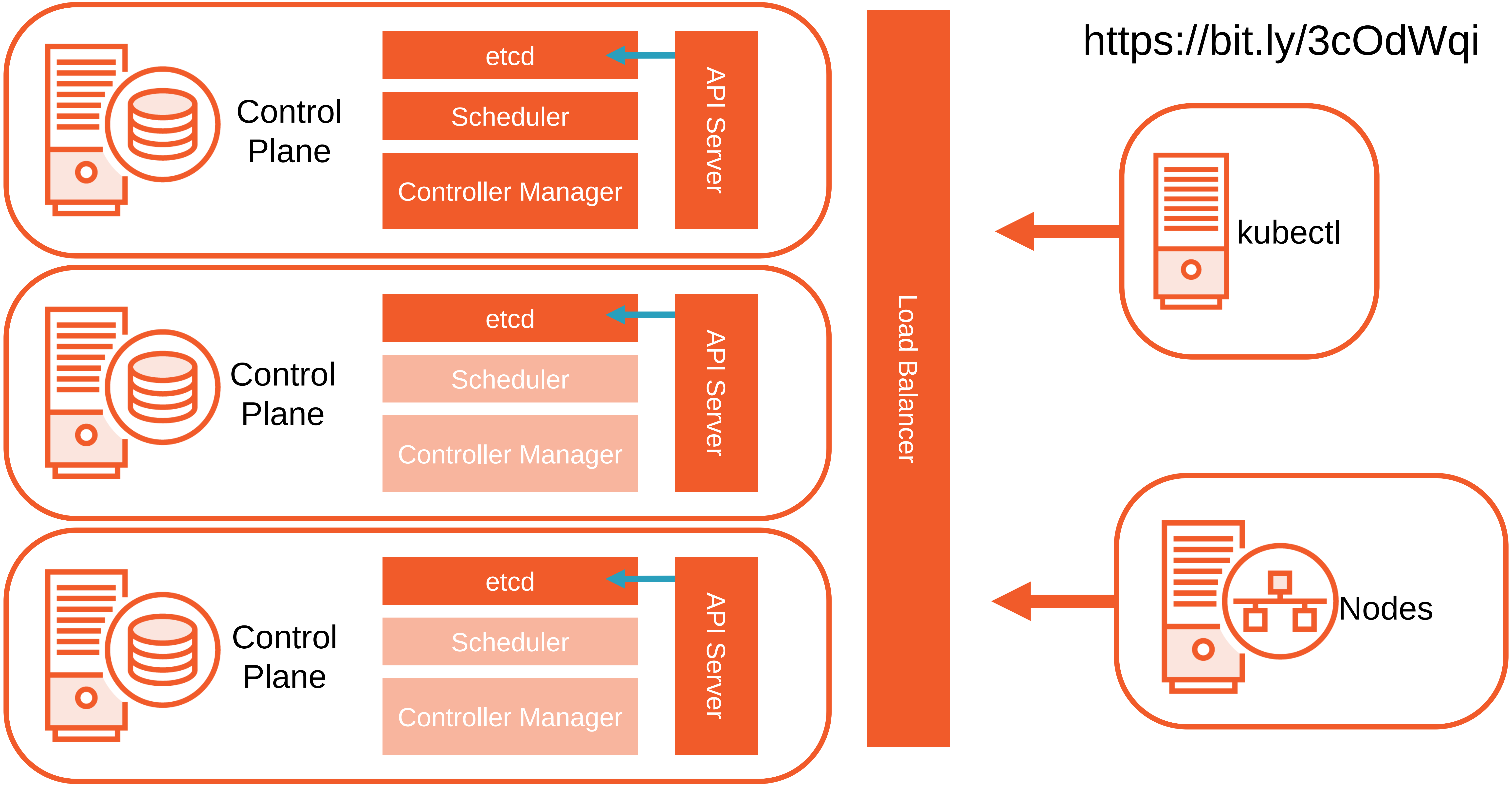
API Server

etcd

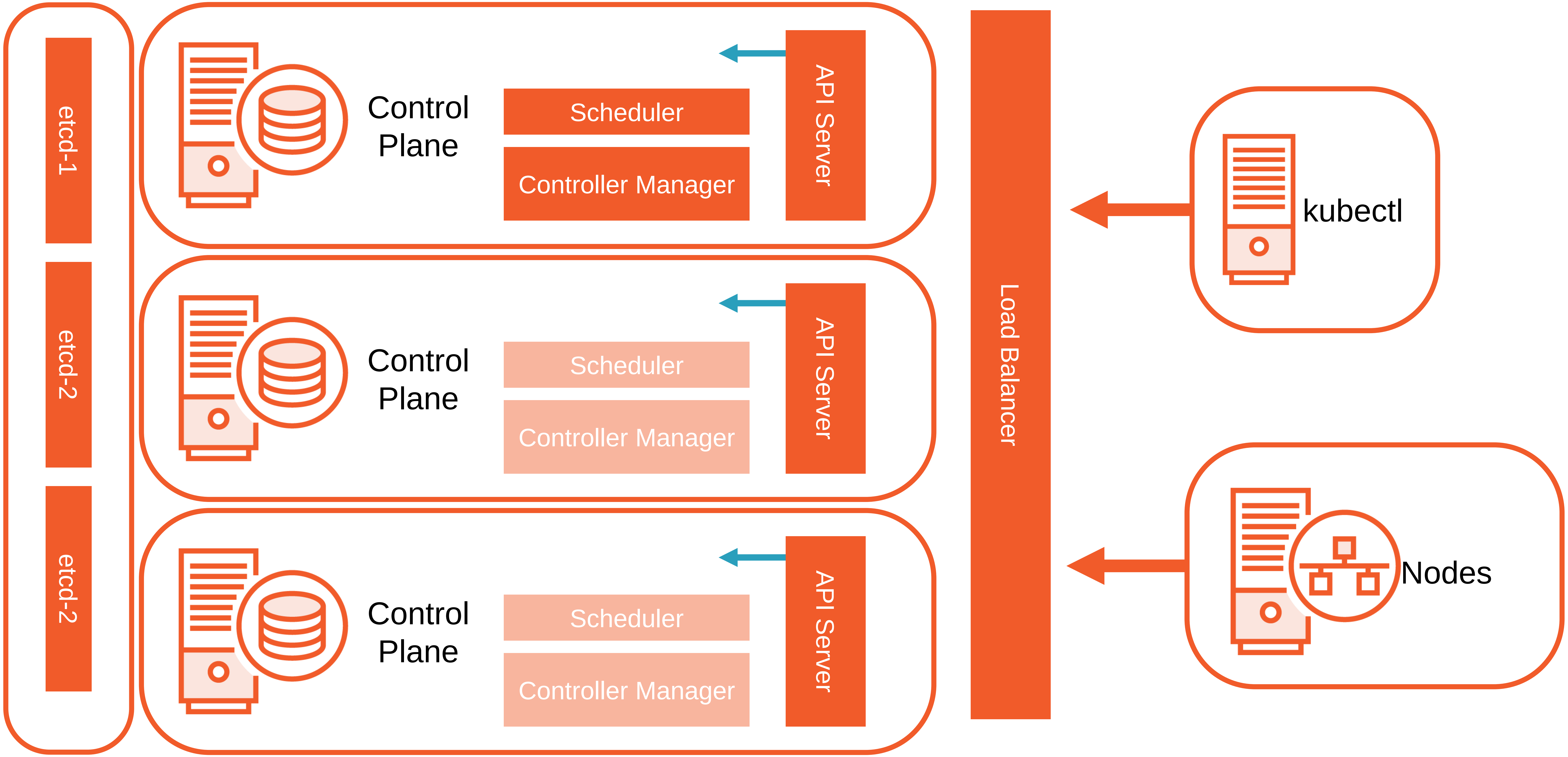
Running multiple
Control Plane
Nodes

HA Cluster Topology - Stacked etcd

https://bit.ly/3cOdWqi



HA Cluster Topology - External etcd



Resources for Building High Availability Clusters

Cluster Topologies

https://bit.ly/3cOdWqi

Building an HA Cluster with kubeadm

https://bit.ly/37dyMOL

Building an HA etcd cluster

https://bit.ly/3dOrRxH

Review

etcd backup and restore operations

Upgrading an existing cluster

Worker Node maintenance

High availability cluster topologies

Up Next:

Logging and Monitoring in Kubernetes Clusters