Kubernetes Tips & Commands!

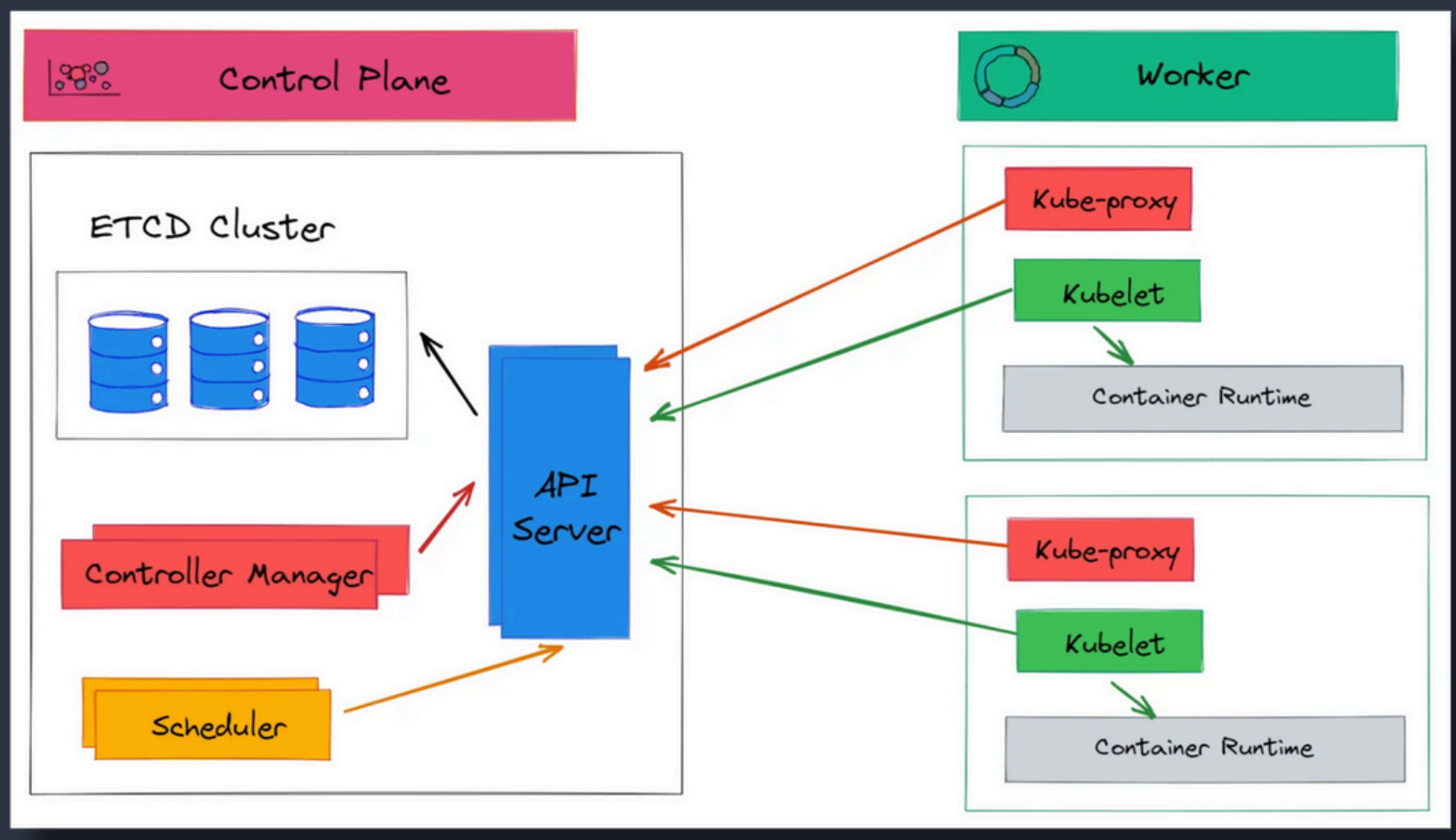
Keep Swiping →

/sahil-saxenadx

01

Tip #1: Understand Kubernetes Architecture

Familiarize yourself with the core components of Kubernetes: Nodes, Pods, Services, and Controllers.



- Nodes: Worker machines where containers run.
- Pods: Smallest deployable units, which can contain one or more containers.
- Services: Stable endpoints to access Pods.
- Controllers: Ensure desired state of Pods (e.g., Deployments, StatefulSets).

02

Tip #2: Use Kubectl for Cluster Management

Kubectl is the command-line tool for interacting with Kubernetes clusters.

```
# List all nodes in the cluster
kubectl get nodes

# List all Pods in the default namespace
kubectl get pods

# Describe a specific Pod
kubectl describe pod <pod_name>

# Apply a configuration from a file
kubectl apply -f <file.yaml>

# Delete a resource by name
kubectl delete <resource_type> <resource_name>
```

03

Tip #3: Deploy Applications with YAML Files

Define your applications declaratively with YAML files.

```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: myapp-deployment
spec:
  replicas: 3
  selector:
    matchLabels:
      app: myapp
  template:
    metadata:
      labels:
        app: myapp
    spec:
      containers:
      - name: myapp
        image: myapp:latest
        ports:
        - containerPort: 80
```

04

Tip #4: Use Namespaces for Better Organization

Namespaces help organize and manage resources in a cluster.

```
# Create a new namespace
kubectl create namespace mynamespace

# List all namespaces
kubectl get namespaces

# Run a Pod in a specific namespace
kubectl run mypod --image=myimage --namespace=mynamespace
```

05

Tip #5: Monitor and Debug with Ease

Quickly troubleshoot and monitor your Kubernetes cluster.

```
# View logs of a specific Pod
kubectl logs <pod_name>

# Access a running container in a Pod
kubectl exec -it <pod_name> -- /bin/bash

# View resource usage of nodes
kubectl top nodes

# View resource usage of Pods
kubectl top pods
```

06

Tip #6: Scale Applications Efficiently

Easily scale your applications to handle varying loads.

```
# Scale a Deployment to a specific number of replicas
kubectl scale deployment myapp-deployment --replicas=5
```

07

Tip #7: Secure Your Cluster

Implement security best practices to protect your cluster.

```
# Example of a NetworkPolicy to restrict traffic
apiVersion: networking.k8s.io/v1
kind: NetworkPolicy
metadata:
  name: allow-specific
  namespace: default
spec:
  podSelector:
    matchLabels:
      role: db
  ingress:
  - from:
    - podSelector:
        matchLabels:
          role: frontend
    ports:
    - protocol: TCP
      port: 3306
```

FOLLOW FOR MORE DEVOPS TIPS AND INSIGHTS! 🚀