**Kubernetes in Production - Study Notes (Day 32)**

**Course: Complete DevOps Course**  
**Instructor: Abhishek**

**1. Introduction**

**Why Production Kubernetes ≠ Minikube/k3s**

* **Minikube/k3s** are for **development only** (not HA, no enterprise support).
* **Production Kubernetes** requires:
  + High Availability (HA)
  + Enterprise-grade security
  + Managed lifecycle (upgrades, scaling)

**2. Kubernetes Distributions for Production**

**What is a Kubernetes Distribution?**

A **distribution** is a customized version of Kubernetes with:  
✅ Enterprise support  
✅ Enhanced tooling (e.g., AWS EKS, Red Hat OpenShift)  
✅ Managed lifecycle (updates, patches)

**Popular Kubernetes Distributions**

| **Distribution** | **Provider** | **Type** |
| --- | --- | --- |
| **Vanilla Kubernetes** | CNCF | Open-source |
| **OpenShift** | Red Hat | Enterprise |
| **Rancher** | Rancher Labs | Managed/On-prem |
| **EKS** | AWS | Cloud-managed |
| **AKS** | Azure | Cloud-managed |
| **GKE** | GCP | Cloud-managed |

📌 **Key Insight:**

* **Minikube** → Learning
* **EKS/OpenShift** → Production

**3. How DevOps Engineers Manage Kubernetes Clusters**

**Tool: kops (Kubernetes Operations)**

* **Purpose:** Install, upgrade, and manage **production-grade Kubernetes clusters**.
* **Features:**
  + Automated cluster lifecycle management
  + Supports AWS, GCP, and other clouds
  + Stores cluster state in **S3** (for backup/recovery)

**Step-by-Step: Creating a Cluster with kops**

**Prerequisites**

1. **AWS CLI Configured**
2. aws configure # Set access key, secret, region
3. **S3 Bucket (for cluster state)**
4. aws s3api create-bucket --bucket my-kops-state-store --region us-east-1

**Create Cluster**

kops create cluster \

--name=k8s.mycompany.com \ # Custom domain

--state=s3://my-kops-state-store \

--zones=us-east-1a \

--node-count=2 \

--node-size=t3.medium \

--master-size=t3.small

**Start Cluster**

kops update cluster --name=k8s.mycompany.com --yes

**Verify Cluster**

kubectl get nodes # Check worker nodes

**4. Key Differences: Vanilla K8s vs. Managed (EKS/OpenShift)**

| **Feature** | **Vanilla Kubernetes** | **Managed (EKS/OpenShift)** |
| --- | --- | --- |
| **Installation** | Manual (kubeadm) | Automated (AWS/GCP Console) |
| **Upgrades** | Manual | One-click |
| **Support** | Community | 24/7 Enterprise Support |
| **Networking** | Manual CNI setup | Pre-configured |
| **Cost** | Free | Paid (with support) |

**5. Practical Assignment**

✅ **Task:** Set up a **production-like** cluster using kops (optional if AWS credits available).  
📌 **Alternative:** Use Minikube for local experiments.

📝 **Documentation Reference:**

* [kops Official Docs](https://kops.sigs.k8s.io/)
* [AWS EKS Guide](https://aws.amazon.com/eks/)

**6. Summary**

* **Production Kubernetes ≠ Minikube** (use **EKS/OpenShift/kops**).
* **kops** automates cluster lifecycle (create/upgrade/delete).
* **Managed K8s (EKS/AKS/GKE)** = Less ops overhead.

🚀 **Next Lesson:** Kubernetes Pods Deep Dive!

📢 **Feedback?** Comment below! 👍 **Like & Share** if this helped!

**End of Notes** 🎉

# \*\*Kubernetes in Production - Study Notes (Day 32)\*\*

\*\*Course: Complete DevOps Course\*\*

\*\*Instructor: Abhishek\*\*

---

## \*\*1. Introduction\*\*

### \*\*Why Production Kubernetes ≠ Minikube/k3s\*\*

- \*\*Minikube/k3s\*\* are for \*\*development only\*\* (not HA, no enterprise support).

- \*\*Production Kubernetes\*\* requires:

- High Availability (HA)

- Enterprise-grade security

- Managed lifecycle (upgrades, scaling)

---

## \*\*2. Kubernetes Distributions for Production\*\*

### \*\*What is a Kubernetes Distribution?\*\*

A \*\*distribution\*\* is a customized version of Kubernetes with:

✅ Enterprise support

✅ Enhanced tooling (e.g., AWS EKS, Red Hat OpenShift)

✅ Managed lifecycle (updates, patches)

### \*\*Popular Kubernetes Distributions\*\*

| \*\*Distribution\*\* | \*\*Provider\*\* | \*\*Type\*\* |

|------------------------|---------------|-------------------|

| \*\*Vanilla Kubernetes\*\* | CNCF | Open-source |

| \*\*OpenShift\*\* | Red Hat | Enterprise |

| \*\*Rancher\*\* | Rancher Labs | Managed/On-prem |

| \*\*EKS\*\* | AWS | Cloud-managed |

| \*\*AKS\*\* | Azure | Cloud-managed |

| \*\*GKE\*\* | GCP | Cloud-managed |

📌 \*\*Key Insight:\*\*

- \*\*Minikube\*\* → Learning

- \*\*EKS/OpenShift\*\* → Production

---

## \*\*3. How DevOps Engineers Manage Kubernetes Clusters\*\*

### \*\*Tool: `kops` (Kubernetes Operations)\*\*

- \*\*Purpose:\*\* Install, upgrade, and manage \*\*production-grade Kubernetes clusters\*\*.

- \*\*Features:\*\*

- Automated cluster lifecycle management

- Supports AWS, GCP, and other clouds

- Stores cluster state in \*\*S3\*\* (for backup/recovery)

### \*\*Step-by-Step: Creating a Cluster with `kops`\*\*

#### \*\*Prerequisites\*\*

1. \*\*AWS CLI Configured\*\*

```sh

aws configure # Set access key, secret, region

```

2. \*\*S3 Bucket (for cluster state)\*\*

```sh

aws s3api create-bucket --bucket my-kops-state-store --region us-east-1

```

#### \*\*Create Cluster\*\*

```sh

kops create cluster \

--name=k8s.mycompany.com \ # Custom domain

--state=s3://my-kops-state-store \

--zones=us-east-1a \

--node-count=2 \

--node-size=t3.medium \

--master-size=t3.small

```

#### \*\*Start Cluster\*\*

```sh

kops update cluster --name=k8s.mycompany.com --yes

```

#### \*\*Verify Cluster\*\*

```sh

kubectl get nodes # Check worker nodes

```

---

## \*\*4. Key Differences: Vanilla K8s vs. Managed (EKS/OpenShift)\*\*

| \*\*Feature\*\* | \*\*Vanilla Kubernetes\*\* | \*\*Managed (EKS/OpenShift)\*\* |

|----------------------|-----------------------|----------------------------|

| \*\*Installation\*\* | Manual (`kubeadm`) | Automated (AWS/GCP Console) |

| \*\*Upgrades\*\* | Manual | One-click |

| \*\*Support\*\* | Community | 24/7 Enterprise Support |

| \*\*Networking\*\* | Manual CNI setup | Pre-configured |

| \*\*Cost\*\* | Free | Paid (with support) |

---

## \*\*5. Practical Assignment\*\*

✅ \*\*Task:\*\* Set up a \*\*production-like\*\* cluster using `kops` (optional if AWS credits available).

📌 \*\*Alternative:\*\* Use Minikube for local experiments.

📝 \*\*Documentation Reference:\*\*

- [`kops` Official Docs](https://kops.sigs.k8s.io/)

- [AWS EKS Guide](https://aws.amazon.com/eks/)

---

## \*\*6. Summary\*\*

- \*\*Production Kubernetes ≠ Minikube\*\* (use \*\*EKS/OpenShift/kops\*\*).

- \*\*`kops`\*\* automates cluster lifecycle (create/upgrade/delete).

- \*\*Managed K8s (EKS/AKS/GKE)\*\* = Less ops overhead.

🚀 \*\*Next Lesson:\*\* Kubernetes Pods Deep Dive!

📢 \*\*Feedback?\*\* Comment below! 👍 \*\*Like & Share\*\* if this helped!

---

\*\*End of Notes\*\* 🎉