EKS Cluster Upgrade

🏢 Enterprise Zero Downtime Kubernetes Upgrade

🗓️ General Info

- **New version release frequency:** Every 3 months
- **Kubernetes version support:** Only latest 3 versions
  Example: If latest is `v1.33`, then supported: `v1.33`, `v1.32`, `v1.31`

✅ Pre-Upgrade Checklist

- [ ] Nodes cordoned
- [ ] Release notes reviewed (e.g., v1.30 → v1.31)
- [ ] Upgrade tested in lower environment
- [ ] Control plane and data plane versions match
- [ ] Cluster Autoscaler matches cluster version
- [ ] At least 5 available IP addresses in subnets
- [ ] Kubelet version matches control plane

📋 Pre-requisites (Details)

> **Important**

1. **Cordon Nodes** – Doesn't affect current app uptime, but new deployments are blocked
2. **Review Release Notes** – e.g., v1.30 → v1.31
3. **Cluster updates are irreversible** – No downgrade supported
4. Validate upgrade in **non-prod environment** before Production
5. **Control Plane** and **Data Plane** must be same version
6. **Cluster Autoscaler** version must match the cluster
7. Ensure **minimum 5 free IP addresses** in each subnet

8. **Kubelet version** on worker nodes must match control plane

---

🔧 Upgrade Process

🧩 Component	⏱️ Estimated Duration
🧠 Control Plane	30–40 minutes
🧱 Node Groups / Fargate	Depends on number of nodes
⚙️ Add-ons	Depends on specific Add-on

📘 Managed Control Plane – Capabilities

🔥 What does the Managed Control Plane provide?

- High Availability
- Disaster Recovery
- Security
- Auto-scaling (API Server scales with request load)

🔥 What it doesn't do:

- Cluster upgrades must be initiated and managed by the **client/user**

⚠️ Things to Keep in Mind During Upgrade Process

- The control plane upgrade (Process 1) can be triggered via a single command or UI; the provider handles this.
- **However, the data plane (nodes) is managed by the user/client.**

Depends on Node Management Strategy:

1. **Managed Node Groups (Launch Template)**
   - Node-by-node rolling upgrade
   - Optionally forceful
2. **Self-managed Nodes (Custom Launch Template, AMI, or EC2)**
   - Manually cordon and drain each node
3. **Hybrid Approach**
   - Mix of both managed and self-managed nodes

OIDC

Associate an IAM role with a Kubernetes service account to allow it to access AWS services directly.

#oidc

Hands-On Guide

Before Upgrading the Cluster

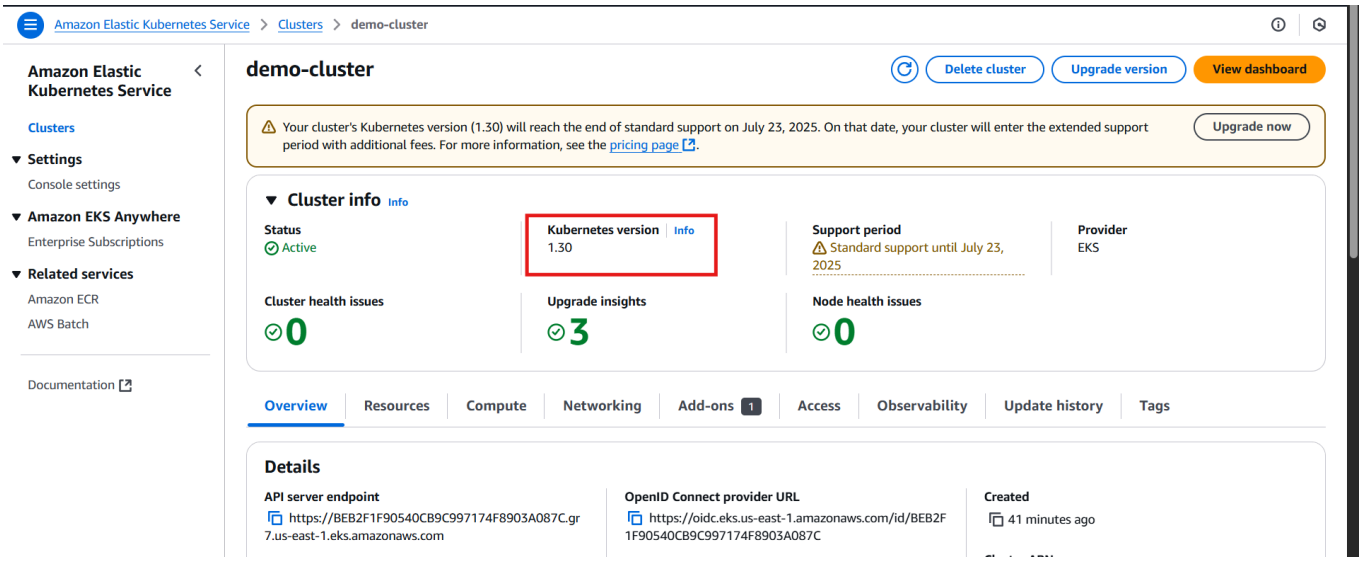
Be sure you know what controllers, networking components, and add-ons are present in the cluster.

Create EKS Cluster

```
# Create EKS Cluster

eksctl create cluster --name=demo-cluster \
                      --region=us-east-1 \
                      --zones=us-east-1a,us-east-1b \
```

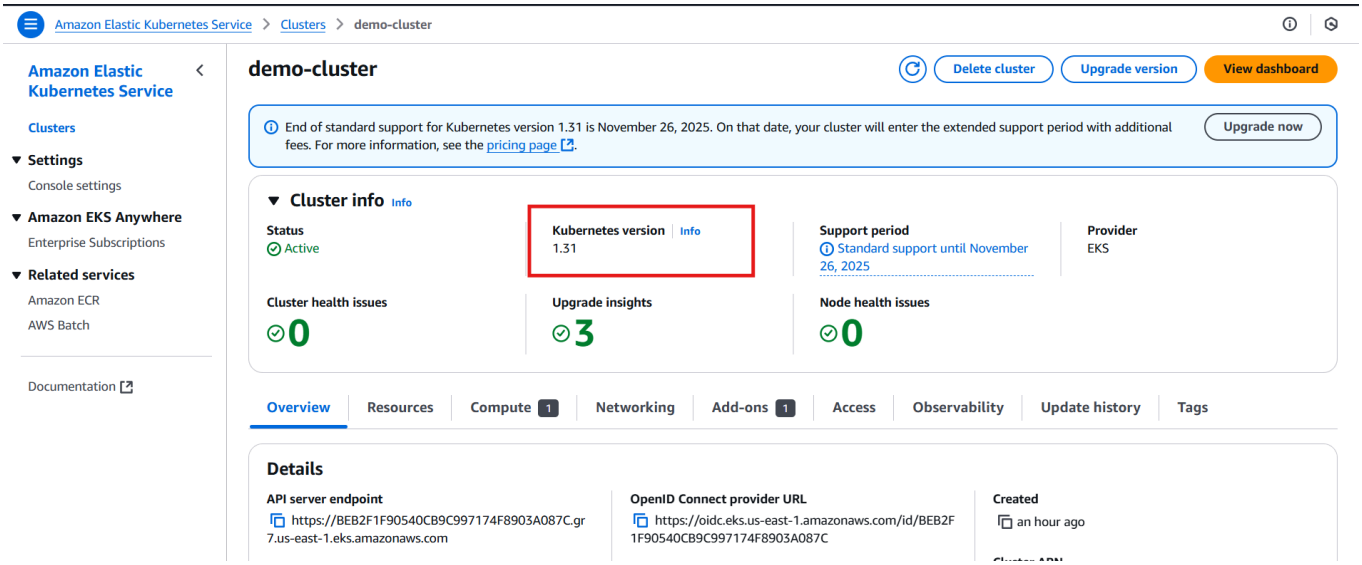
```
                        --without-nodegroup

# Enable ODIC
eksctl utils associate-iam-oidc-provider \
    --region us-east-1 \
    --cluster demo-cluster \
    --approve

# Create Nodegroup
eksctl create nodegroup --cluster=demo-cluster \
                        --region=us-east-1 \
                        --name=demo-cluster-ng-private \
                        --node-type=t3.medium \
                        --nodes-min=2 \
                        --nodes-max=3 \
                        --node-volume-size=20 \
                        --managed \
                        --asg-access \
                        --external-dns-access \
                        --full-ecr-access \
                        --appmesh-access \
                        --alb-ingress-access \
                        --node-private-networking

# Update ./kube/config file
aws eks update-kubeconfig --name demo-cluster

# Delete the EKS Cluster
eksctl delete cluster --name demo-cluster
```

🔄 1. Update Cluster Version




```
eksctl upgrade cluster --name demo-cluster --region us-east-1 --version 1.31 --approve
```

```
Yan@Igris ~ $ eksctl upgrade cluster --name demo-cluster --region us-east-1 --version 1.31 --approve
2025-05-20 15:25:07 [ℹ]  will upgrade cluster "demo-cluster" control plane from current version "1.30" to "1.31"
2025-05-20 15:34:37 [✔]  cluster "demo-cluster" control plane has been upgraded to version "1.31"
2025-05-20 15:34:37 [ℹ]  you will need to follow the upgrade procedure for all of nodegroups and add-ons
2025-05-20 15:34:45 [ℹ]  re-building cluster stack "eksctl-demo-cluster-cluster"
2025-05-20 15:34:45 [✔]  all resources in cluster stack "eksctl-demo-cluster-cluster" are up-to-date
2025-05-20 15:34:47 [ℹ]  checking security group configuration for all nodegroups
2025-05-20 15:34:47 [ℹ]  all nodegroups have up-to-date cloudformation templates
```



🔥 Note:

You cannot skip versions during upgrades.
Example:

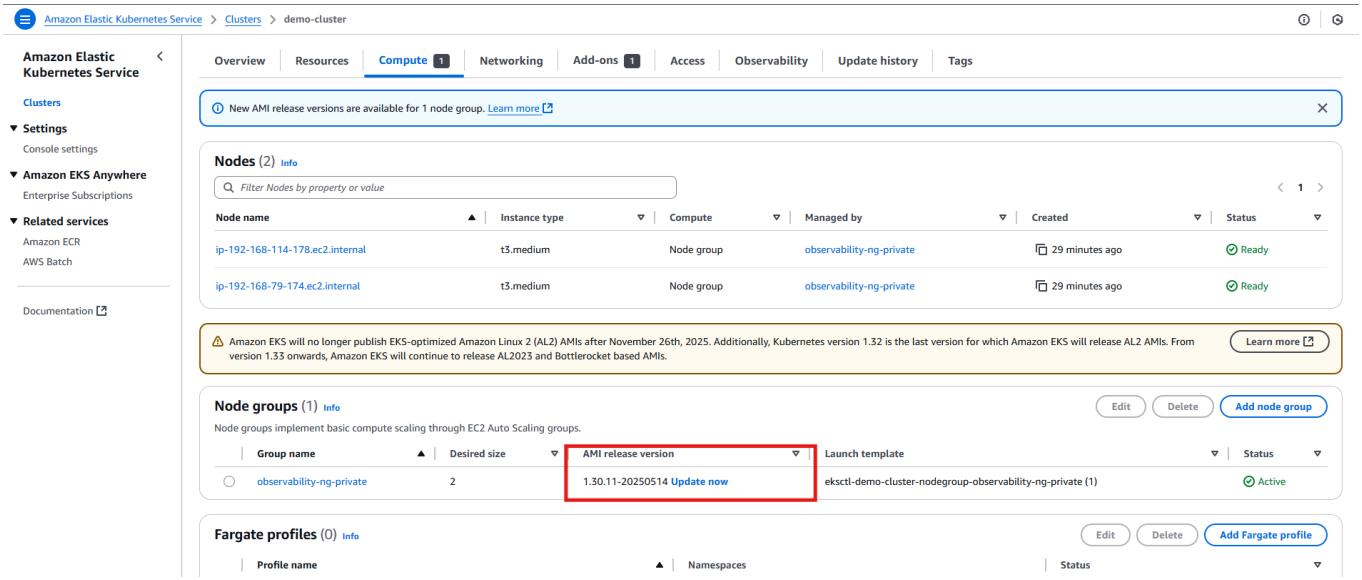
- ❌ `v1.30 → v1.33`

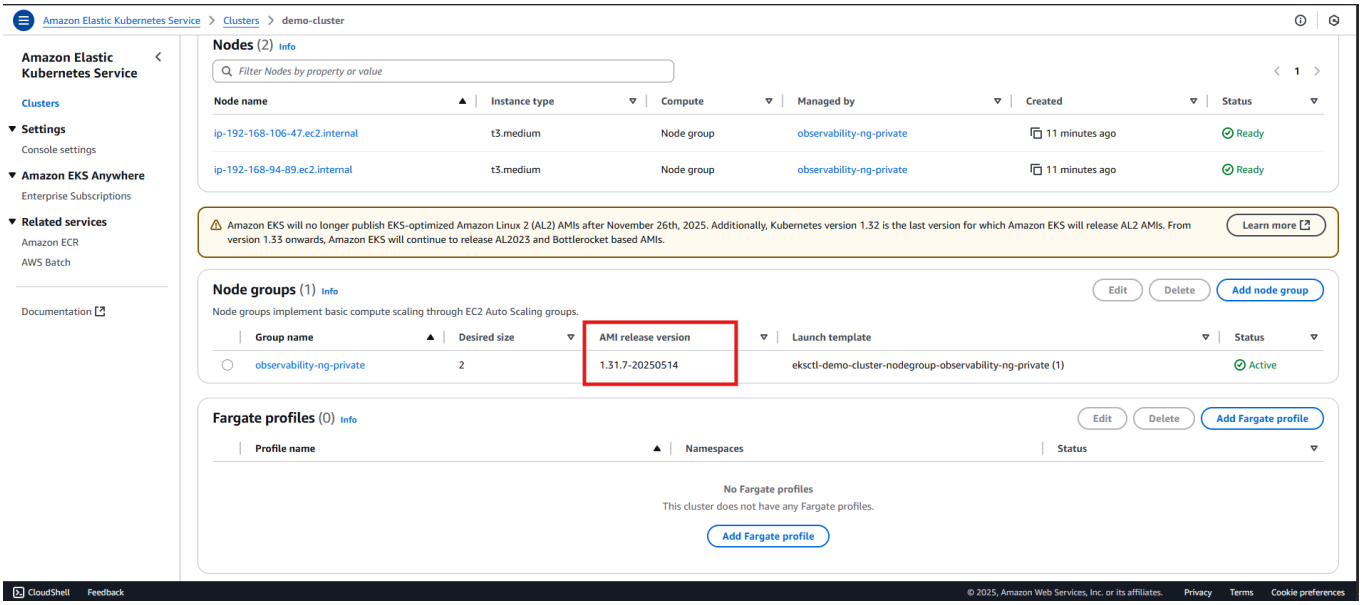
- ✅ `v1.30 → v1.31 → v1.32 → v1.33`

⚙️ 2. Node Group Update

Method 1 – Update Existing Group (via AWS UI)

- Rolling Update (zero downtime)





- Forceful Update (When - Pod Disruption Budget)

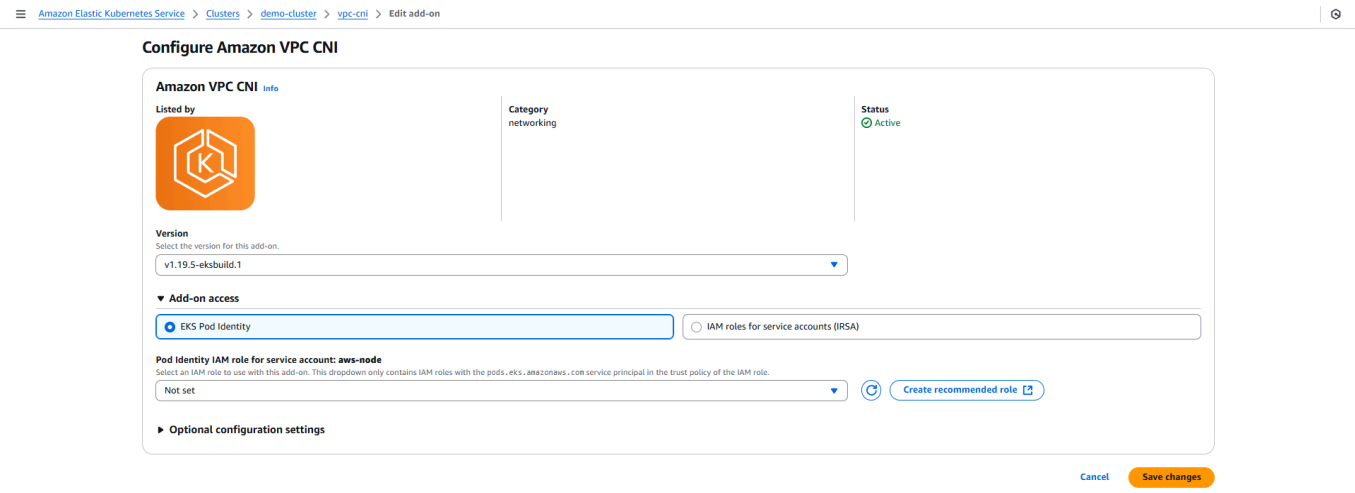
Method 2 – Add New Node Group

- More flexible but higher cost and slower

---

3. Add-Ons Update

- Can be updated via AWS Console UI



- Review compatibility with Kubernetes version