Kubernetes Cluster Upgrade to a Specific Version using kubeadm

This guide explains how to upgrade a Kubernetes cluster created with kubeadm to a specific version (for example, v1.29.2) on RPM-based Linux systems such as CentOS, RHEL, Rocky Linux, or AlmaLinux.

Step 1: Check Current Kubernetes Version

---

kubectl version --short

- Shows Kubernetes client and server versions.

kubeadm version

- Displays current kubeadm version.

Step 2: Drain the Node (Control Plane)

---

kubectl drain <node-name> --ignore-daemonsets

- Prepares the node for upgrade by evicting running pods.

- --ignore-daemonsets allows DaemonSet pods to continue running.

Step 3: Upgrade kubeadm to the Target Version

---

sudo yum install -y kubeadm-1.29.2 --disableexcludes=kubernetes

- Installs the specific kubeadm version.

- --disableexcludes ensures the package comes from the Kubernetes repo.

Step 4: Verify the New kubeadm Version

---

kubeadm version

- Confirms the installed kubeadm version.

Step 5: Review the Upgrade Plan (Master Node Only)

---

sudo kubeadm upgrade plan

- Shows available versions and upgrade instructions.

Step 6: Upgrade the Control Plane (Master Node Only)

---

sudo kubeadm upgrade apply v1.29.2

- Upgrades control plane components to the target version.

Step 7: Upgrade kubelet and kubectl (All Nodes)

---

sudo yum install -y kubelet-1.29.2 kubectl-1.29.2 --disableexcludes=kubernetes

- Installs kubelet and kubectl of the specified version.

Step 8: Restart kubelet (All Nodes)

---

sudo systemctl daemon-reexec

sudo systemctl restart kubelet

- Applies the new kubelet version.

Step 9: Make the Node Schedulable Again

---

kubectl uncordon <node-name>

- Makes the node available again for scheduling pods.

Step 10: Upgrade Worker Nodes

Repeat Steps 2-9 on each worker node.

Final Step: Check Cluster Status

---

kubectl get nodes

- Verifies all nodes are in the Ready state and upgraded.