Mohammad Saeed Pourmazar

https://github.com/MohammadSaeedPourmazar

https://gitlab.com/MohammadSaeedPourmazar

https://medium.com/@MohammadSaeedPourmazar

https://dev.to/MohammadSaeedPourmazar

https://www.youtube.com/@MohammadSaeedPourmazar

https://www.instagram.com/MohammadSaeedPourmazar

https://www.facebook.com/MohammadSaeedPourmazar

https://www.linkedin.com/in/MohammadSaeedPourmazar/

https://orcid.org/0009-0008-9383-419X

Install Rancher Using Kubernetes

To install Rancher with Kubernetes, you need to set up both Kubernetes (which Rancher will manage) and then Rancher itself. Here's a step-by-step guide to get you through the process.

Prerequisites:

** Docker: Rancher is deployed as a Docker container, so Docker must be installed on all the nodes you want to use for the Rancher installation **

** Kubernetes cluster: You need an existing Kubernetes cluster. If you don't have one, you'll need to set up Kubernetes first using kubeadm or another method **

** kubectl: This is necessary to interact with your Kubernetes cluster and deploy Rancher **

** Helm: This is the package manager for Kubernetes, used to install and manage Rancher **

1. *Install cert-manager (Optional):*

To manage SSL certificates (optional but recommended), install cert-manager first.

Install the custom resource definitions (CRDs):

```
kubectl apply -f https://github.com/cert-manager/cert-manager/releases/latest/download/cert-manager.crds.yaml
```

Install cert-manager:

```
kubectl apply -f https://github.com/cert-manager/cert-manager/releases/latest/download/cert-manager.yaml
```

Wait for the cert-manager pods to be running:

kubectl get pods -n cert-manager

** If it is pending use one of the three solutions options below: **

** Try Option 1 (if this is a test setup) and check pod status again: **

kubectl get pods -n cert-manager

** If using a multi-node cluster, add a worker node **

** If you need Cert-Manager on the control plane but want to keep the taint, use Option 3**

Solutions to Fix It:

Option 1: Allow Scheduling on the Control Plane (Temporary Fix)

If this is a single-node cluster (e.g., using kubeadm or kind for testing), you can allow scheduling workloads on the control plane by running:

kubectl taint nodes --all node-role.kubernetes.io/control-plane-

This removes the taint and lets all workloads, including Cert-Manager, be scheduled on the control plane.

Option 2: Deploy a Worker Node (Recommended for Production)

If you're in a production setup, it's best not to remove the taint but instead add worker nodes. You can check your node list:

kubectl get nodes -o wide

If you have only one node (the control plane), add a worker node to your cluster.

Option 3: Allow Cert-Manager to Run on the Control Plane

Instead of removing the taint from the node, you can modify Cert-Manager's deployment to tolerate the taint. Patch the deployment:

```
kubectl patch deployment cert-manager -n cert-manager --type='json' -p='[{"op": "add", "path": "/spec/template/spec/tolerations", "value": [{"key": "node-role.kubernetes.io/control-plane", "operator": "Exists", "effect": "NoSchedule"}]}]'
```

This allows Cert-Manager to run on the control plane node without removing the taint for other workloads.

2. Add Rancher Helm Repo:

If you haven't already added the Rancher Helm chart repository, do it now:

```
helm repo add rancher-stable https://releases.rancher.com/server-charts/stable
helm repo update
```

3. Install Rancher:

Now, install Rancher using Helm:

**** Install Rancher without LoadBalancer or NodePort (standard installation): ****

```
helm install rancher rancher-stable/rancher \
  --namespace cattle-system \
  --create-namespace \
  --set hostname=<YOUR-DOMAIN-HERE> \
  --set bootstrapPassword=admin
```

** Replace <YOUR-DOMAIN-HERE> with your actual domain IP Address **

This command will:

** Install Rancher into the cattle-system namespace **

** Set the hostname for Rancher **

** Set the bootstrap password to admin **

** Expose Rancher Using LoadBalancer (Recommended if you're using AWS or another cloud provider): **

To expose Rancher via a LoadBalancer (recommended for public access):

```
kubectl patch svc rancher -n cattle-system -p '{"spec": {"type": "LoadBalancer"}}'
```

** After a few minutes, you will get an external IP address for Rancher by running **

```
kubectl get svc -n cattle-system
```

** You can now access Rancher via the external IP or your domain **

4. Access Rancher:

Once Rancher is installed, access the UI:

Open your browser and go to:

**** https://<YOUR-DOMAIN-HERE> or the external IP from the LoadBalancer ****

Log in using the bootstrap password (admin).

Step 6: Initial Setup

You will be prompted to complete the initial setup for Rancher, including setting up a password and configuring your first cluster.

Common Issues & Fixes

LoadBalancer IP Not Assigned:

If you're not using a cloud provider that automatically assigns a LoadBalancer IP (e.g., AWS), you can manually expose Rancher using a NodePort or configure a Reverse Proxy.

Ingress Not Working:

Ensure your domain is correctly configured in DNS and pointing to your Kubernetes cluster.