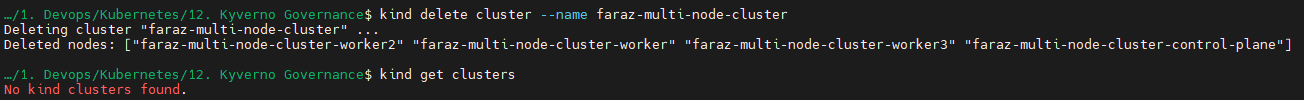
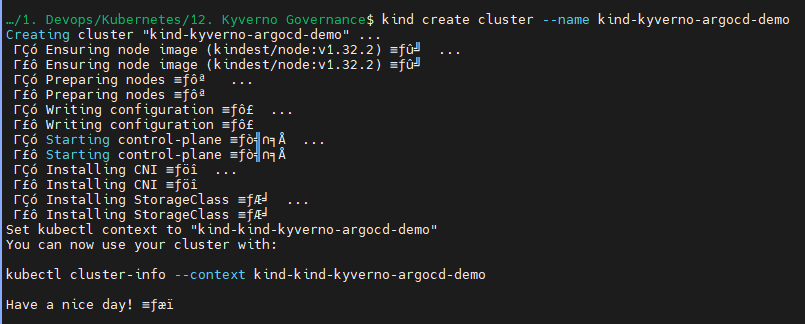
Delete the previous Kind cluster:

* kind delete cluster --name faraz-multi-node-cluster
* kind get clusters



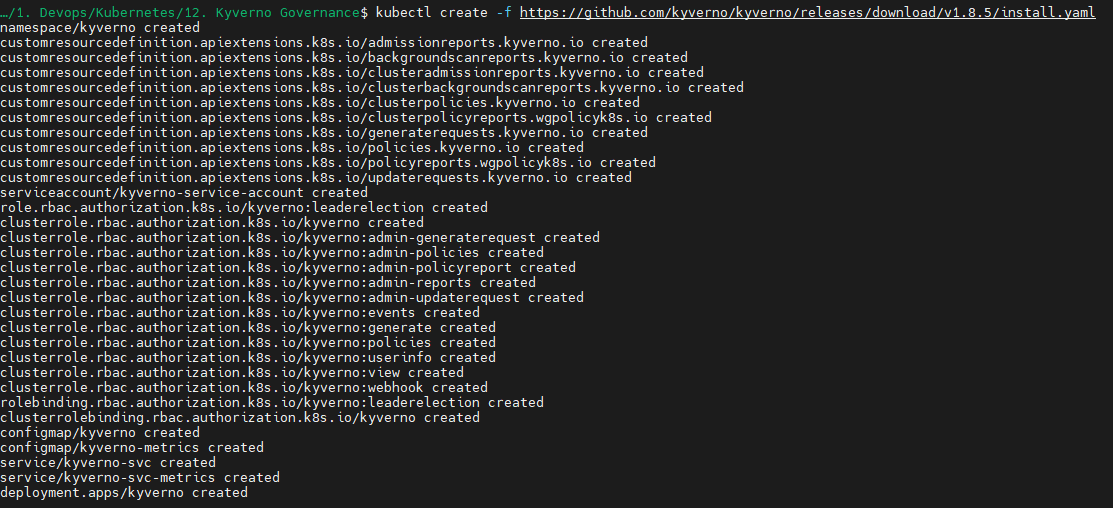
Create new Kind cluster:

* kind create cluster --name kind-kyverno-argocd-demo



Install Kyverno:

* kubectl create -f <https://github.com/kyverno/kyverno/releases/download/v1.8.5/install.yaml>

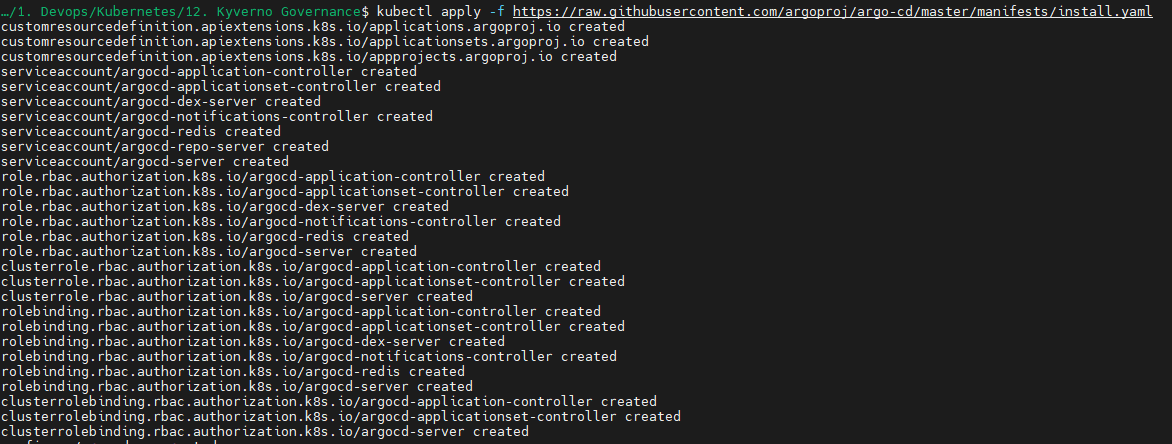


* kubectl get pods -n kyverno



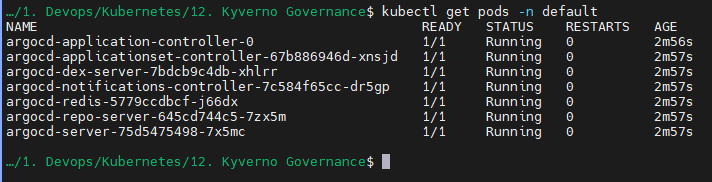
Install ArgoCD:

* kubectl apply -f <https://raw.githubusercontent.com/argoproj/argo-cd/master/manifests/install.yaml>



Validate your argocd pods are running:

* kubectl get pods -n default



Now you can get an example for require pod request limits:

<https://raw.githubusercontent.com/kyverno/policies/refs/heads/main/best-practices/require-pod-requests-limits/require-pod-requests-limits.yaml>

You can get more examples from:

<https://github.com/kyverno/policies/tree/main/best-practices>

If you use:

kubectl apply -f <https://raw.githubusercontent.com/kyverno/policies/refs/heads/main/best-practices/require-pod-requests-limits/require-pod-requests-limits.yaml>

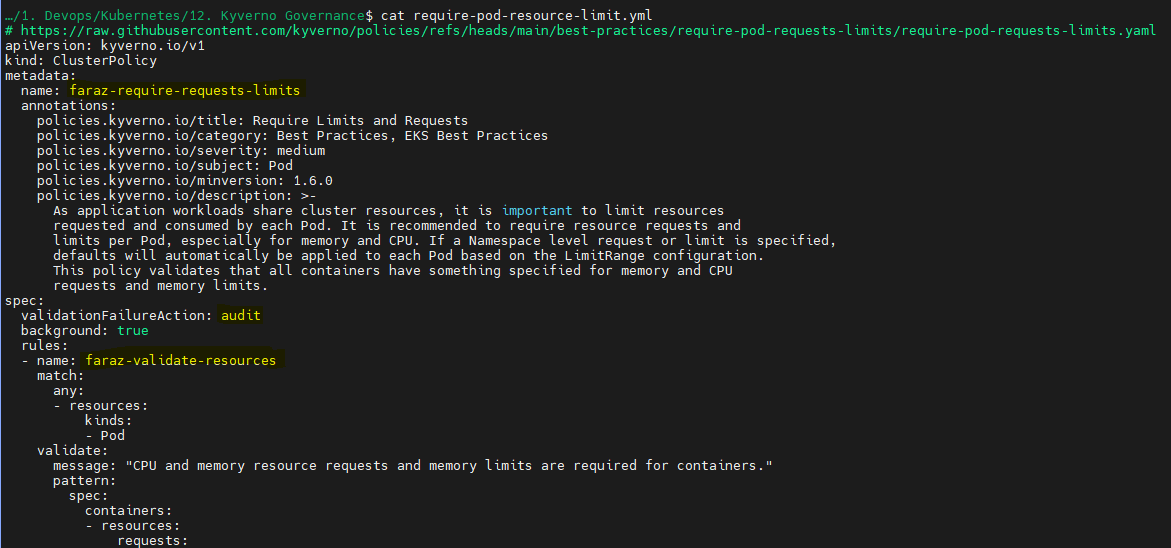


You may get error

You can solve this by copy pasting from the raw github example provided by kyverno:



Use small a instead of capital A for Audit



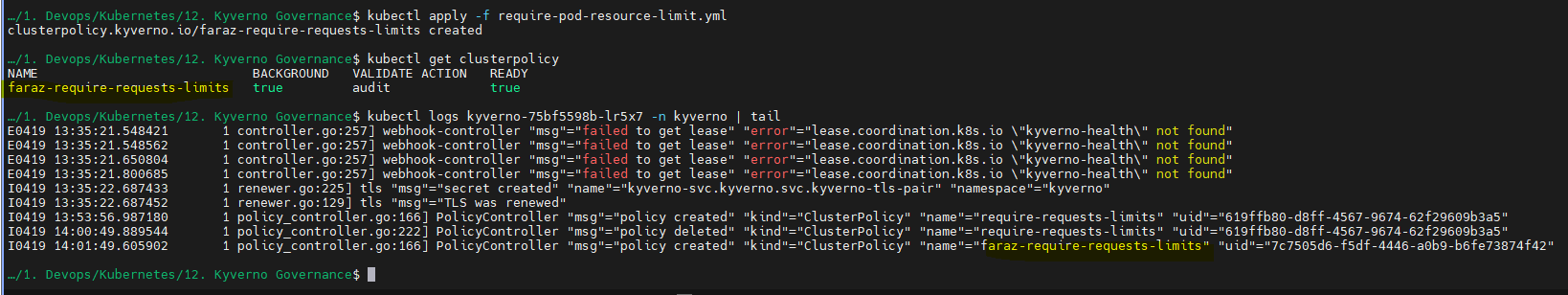


Then apply the cluster policy:

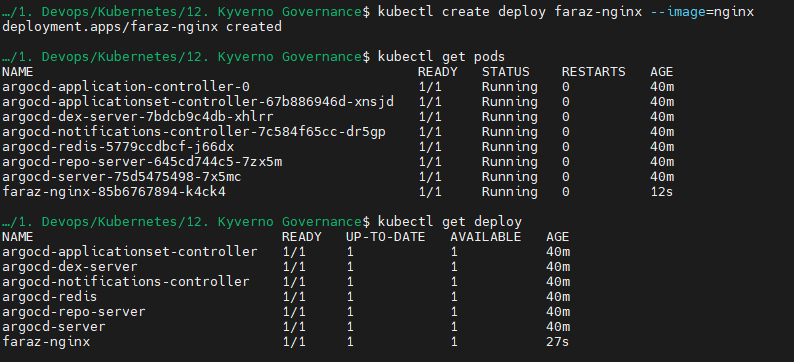
* kubectl apply -f require-pod-resource-limit.yml

Now you will be able to see that Kyverno policy reads the policy which we just created:

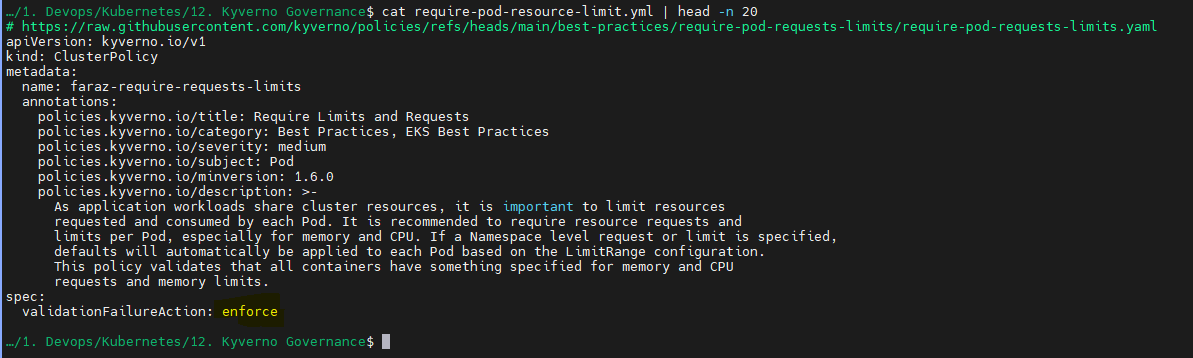
* kubectl get clusterpolicy
* kubectl logs kyverno-75bf5598b-lr5x7 -n kyverno | tail



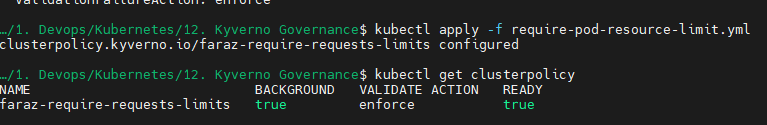
You may see that the nginx deployment and pods were created:



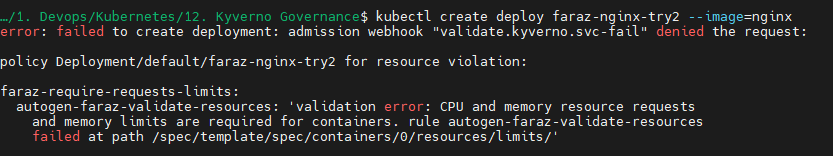
This is due to the fact that we need to enable enforce the validation failure action:



Apply again:

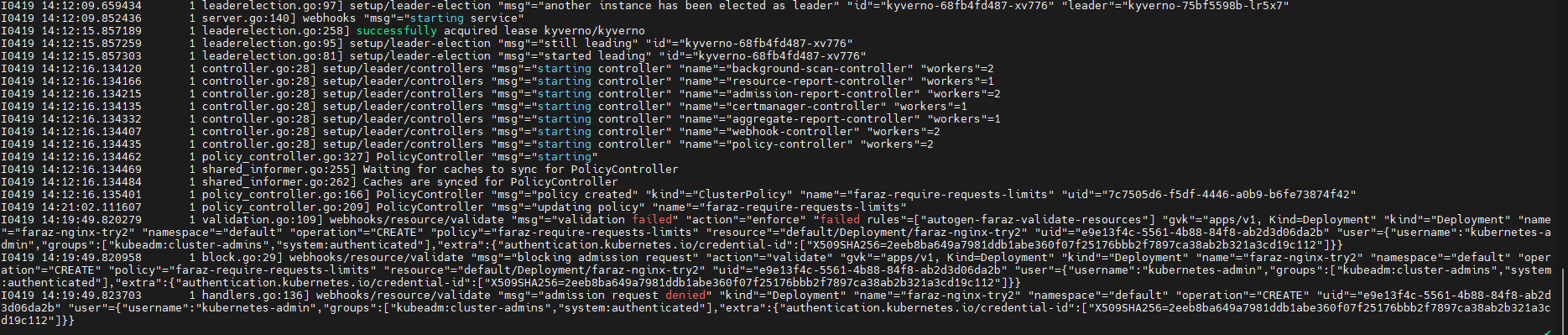


Now you may see the below error:



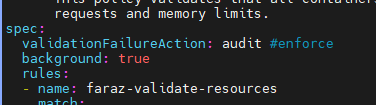
You can also check the logs:

* kubectl logs kyverno-68fb4fd487-xv776 -n kyverno



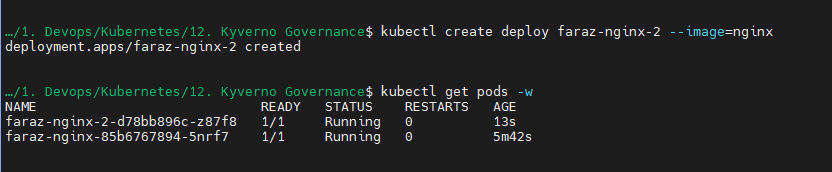
--------------------------------------------------------------------------------------------------------------------------------------

If you want to use audit and see which pods are created which don’t follow the cluster policy, you can change the validation failure action to audit:

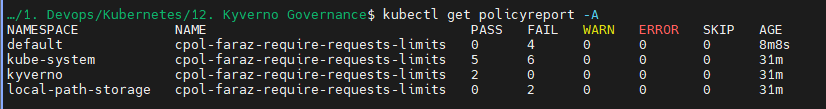


Create deployment:



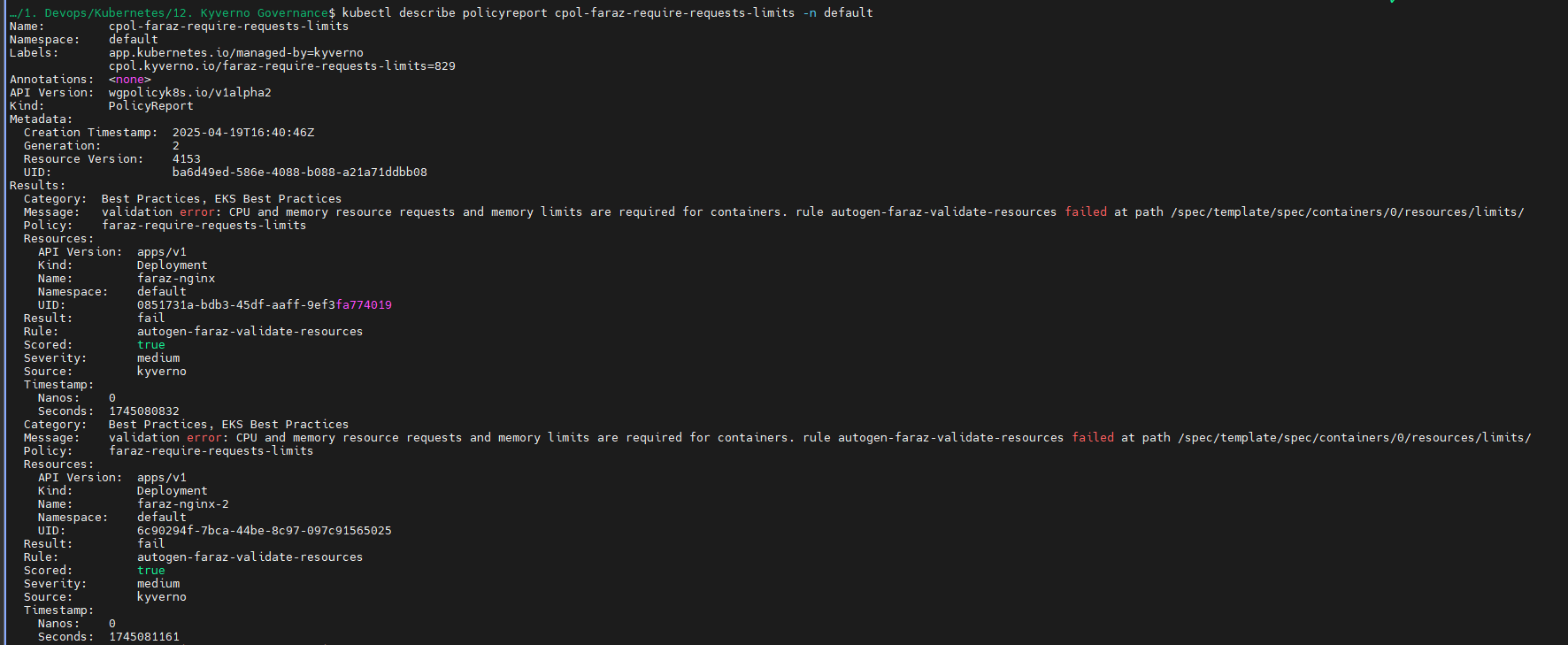


* kubectl get policyreport -A

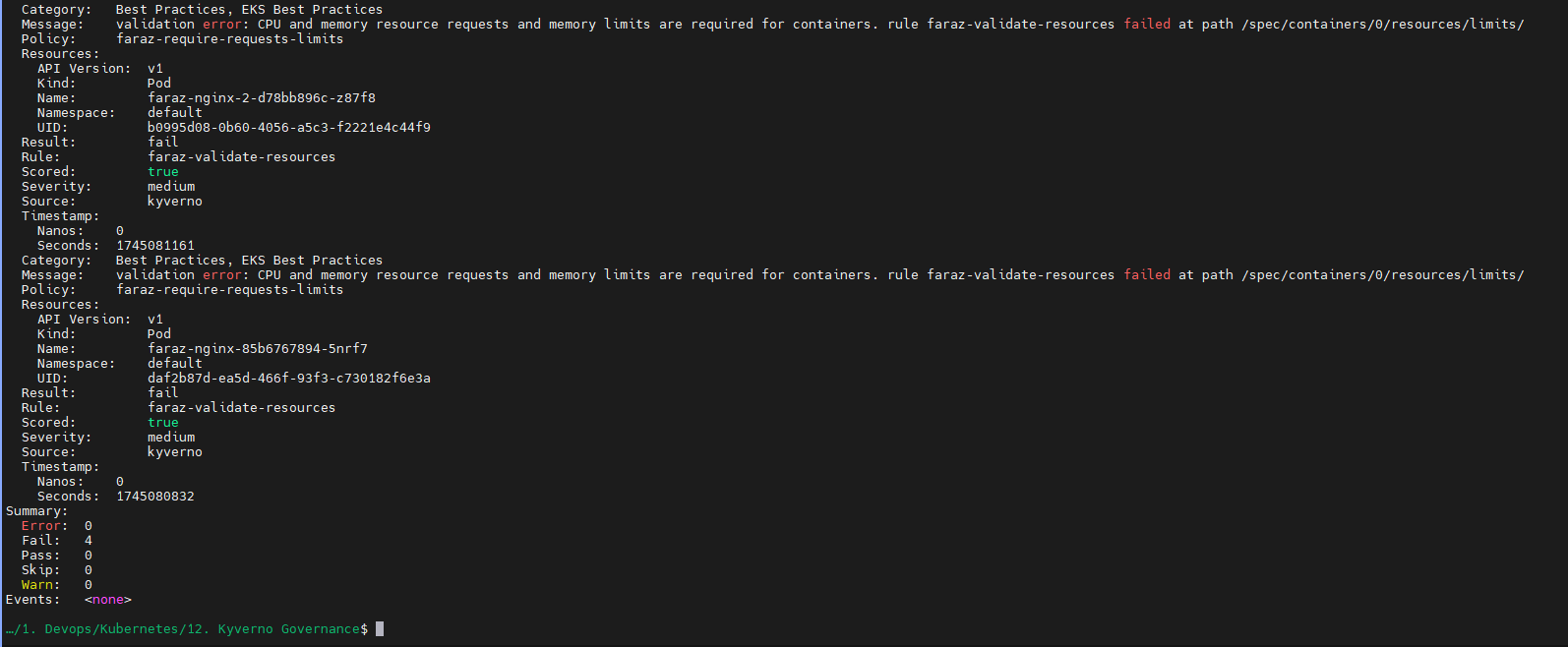


* kubectl describe policyreport cpol-faraz-require-requests-limits -n default

For deployment:



For pods:



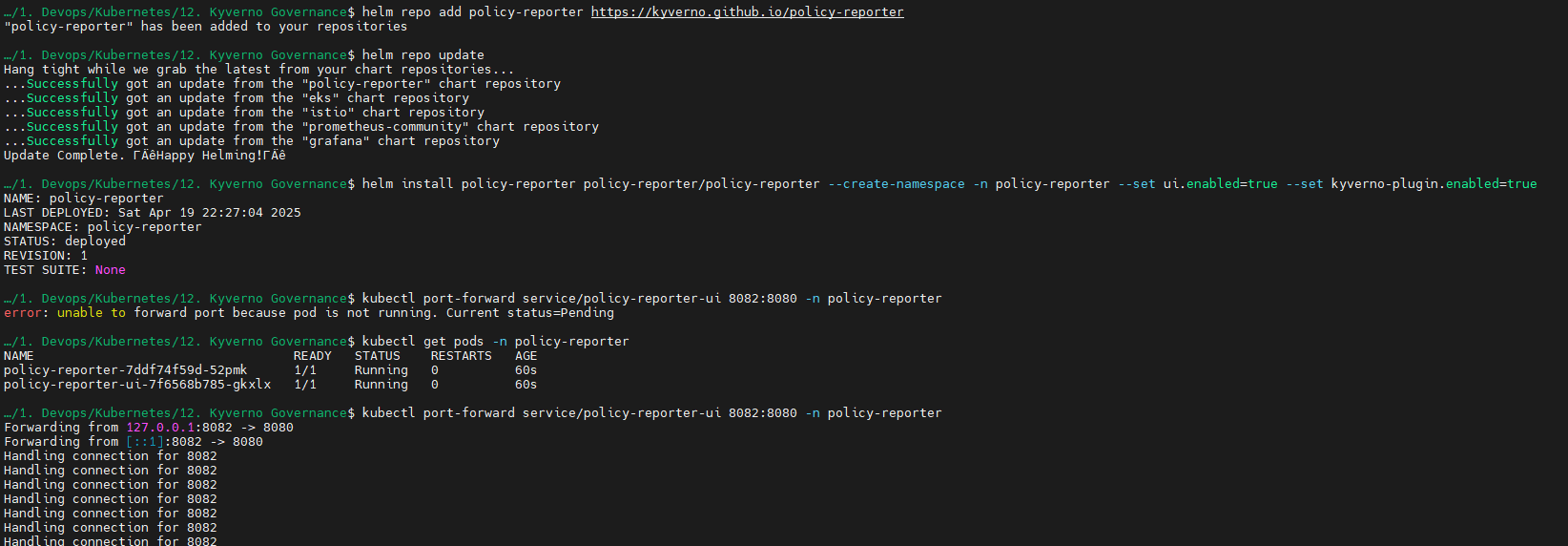
Enforce flag doesn’t allow you to create resources with the defined cluster policy while audit just tracks the failed cluster policy.

You can also create policy reporter dashboard:

* <https://github.com/kyverno/policy-reporter#readme>

Installation:

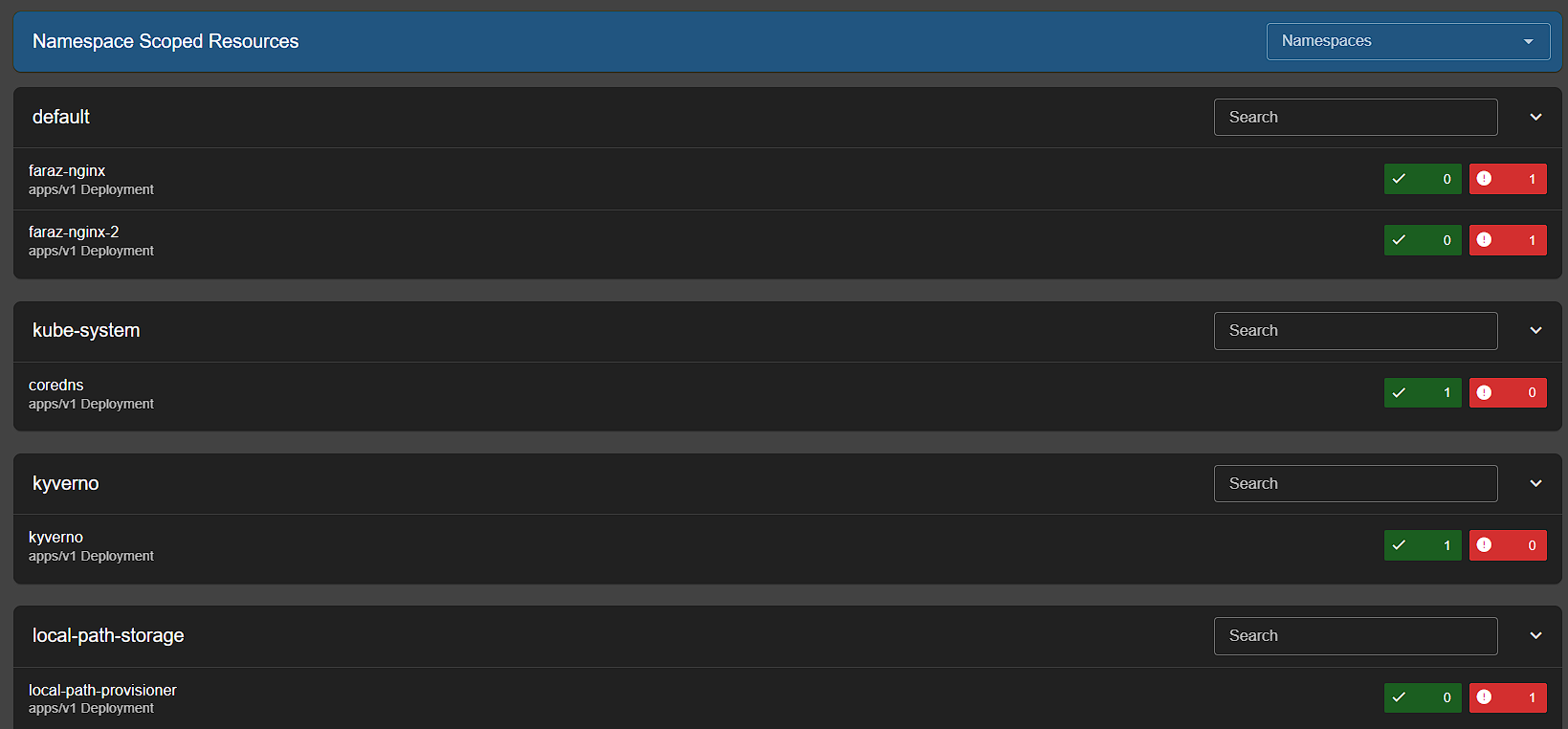
* helm repo add policy-reporter <https://kyverno.github.io/policy-reporter>
* helm repo update
* helm install policy-reporter policy-reporter/policy-reporter --create-namespace -n policy-reporter --set ui.enabled=true --set kyverno-plugin.enabled=true
* kubectl get pods -n policy-reporter
* kubectl port-forward service/policy-reporter-ui 8082:8080 -n policy-reporter



* Open <http://localhost:8082/> in your browser.

Now you can check at deployment level:





Similarly for pods:

