Sponsored by Red Hat

Installing Pulp operator in k8s with different methods

Mike DePaulo
mikedep333

Humberto Yagi
hyagi

pulp

INTRO

- Installing minikube
- Deploying PostgreSQL
- Installing HELM
- Installing **Pulp Operator** using HELM
- Installing OLM
- Installing **Pulp Operator** using OLM catalog
- Installing **Pulp Operator** from source
  - as a k8s deployment
  - as a linux process

Sponsored by Red Hat

INSTALLING MINIKUBE

https://minikube.sigs.k8s.io/docs/start/

```
curl -LO https://storage.googleapis.com/minikube/releases/latest/minikube-linux-amd64
sudo install minikube-linux-amd64 /usr/local/bin/minikube
```

https://minikube.sigs.k8s.io/docs/drivers/podman/

```
minikube start --driver=podman
```

https://kubernetes.io/docs/tasks/access-application-cluster/ingress-minikube/#enable-the-ingress-controller

```
minikube addons enable ingress
```


pulp

INSTALLING KUBECTL

https://kubernetes.io/docs/tasks/tools/install-kubectl-linux/#install-kubectl-binary-with-curl-on-linux

```
curl -LO "https://dl.k8s.io/release/$(curl -L -s https://dl.k8s.io/release/stable.txt)/bin/linux/amd64/kubectl"
sudo install -o root -g root -m 0755 kubectl /usr/local/bin/kubectl
echo 'source <(kubectl completion bash)' >>~/.bashrc
echo 'alias k=kubectl' >>~/.bashrc
echo 'complete -o default -F __start_kubectl k' >>~/.bashrc
```

- adding *pulp.lab* in /etc/hosts to make it easier to communicate with Pulp API

```
echo "$(minikube ip) pulp.lab" |sudo tee -a /etc/hosts
```

DEPLOYING POSTGRESQL

https://github.com/pulp/pulp-operator/tree/main/config/samples

```
k create ns db
k apply -f https://raw.githubusercontent.com/pulp/pulp-operator/main/config/samples/external_db.yaml
```

WHY USE THESE TOOLS?

- Why use a tool to install pulp-operator?
  - The image is already available
  - But several resources to deploy
    - https://github.com/pulp/pulp-k8s-resources/tree/main/helm-charts/templates
  - We also want automatic / managed updates

INSTALLING HELM

https://helm.sh/docs/intro/install

```
wget https://get.helm.sh/helm-v3.13.1-linux-amd64.tar.gz
tar xf helm-v3.13.1-linux-amd64.tar.gz
sudo mv linux-amd64/helm /usr/local/bin/helm
```

INSTALLING PULP OPERATOR USING HELM CHART

https://docs.pulpproject.org/pulp_operator/install/helm/

```
k create ns pulp
k config set-context --current --namespace pulp

helm repo add pulp-operator https://github.com/pulp/pulp-k8s-resources/raw/main/helm-charts/ --force-update
helm install pulp pulp-operator/pulp-operator
```

DEPLOY PULP CR

https://github.com/pulp/pulp-operator/tree/main/config/samples

```
k apply -f https://raw.githubusercontent.com/pulp/pulp-operator/main/config/samples/pulpcon-sample.yaml
```

CLEANING UP

```
helm uninstall pulp
k delete ns pulp
```

INSTALLING OLM

https://sdk.operatorframework.io/docs/installation/#1-download-the-release-binary

```
export ARCH=$(case $(uname -m) in x86_64) echo -n amd64 ;; aarch64) echo -n arm64 ;; *) echo -n $(uname -m) ;; esac)
export OS=$(uname | awk '{print tolower($0)}')
export OPERATOR_SDK_DL_URL=https://github.com/operator-framework/operator-sdk/releases/download/v1.32.0
curl -LO ${OPERATOR_SDK_DL_URL}/operator-sdk_${OS}_${ARCH}
chmod +x operator-sdk_${OS}_${ARCH} && sudo mv operator-sdk_${OS}_${ARCH} /usr/local/bin/operator-sdk
```

https://olm.operatorframework.io/docs/getting-started/#installing-olm-in-your-cluster

```
operator-sdk olm install
```

INSTALLING PULP OPERATOR USING OLM

https://docs.pulpproject.org/pulp_operator/install/ocp/

```
kubectl create ns pulp-olm
k config set-context --current --namespace pulp-olm

kubectl apply -f-<<EOF
kind: OperatorGroup
apiVersion: operators.coreos.com/v1
metadata:
  name: pulp
  namespace: pulp-olm
spec:
  targetNamespaces:
  - pulp-olm
EOF
```

INSTALLING PULP OPERATOR USING OLM

```
kubectl apply -f-<<EOF
apiVersion: operators.coreos.com/v1alpha1
kind: Subscription
metadata:
  name: pulp
  namespace: pulp-olm
spec:
  channel: beta
  installPlanApproval: Automatic
  name: pulp-operator
  source: operatorhubio-catalog
  sourceNamespace: olm
  startingCSV: pulp-operator.v1.0.0-beta.2
EOF
```

https://operatorhub.io/operator/pulp-operator

DEPLOY PULP CR

https://github.com/pulp/pulp-operator/tree/main/config/samples

```
k apply -f https://raw.githubusercontent.com/pulp/pulp-operator/main/config/samples/pulpcon-sample.yaml
```

CLEANING UP

```
k delete ns pulp-olm
k get crd -oname|grep pulp|xargs kubectl delete
k get clusterrolebinding,clusterrole -oname|grep pulp|xargs kubectl delete
```

INSTALLING GOLANG

https://go.dev/doc/install

```
wget https://go.dev/dl/go1.21.3.linux-amd64.tar.gz
sudo rm -rf /usr/local/go && sudo tar -C /usr/local -xzf go1.21.3.linux-amd64.tar.gz
echo "export PATH=$PATH:/usr/local/go/bin" >> ~/.bash_profile
source ~/.bash_profile

sudo dnf install -y make
```

INSTALLING PULP OPERATOR FROM SOURCE

AS A K8S DEPLOYMENT

```
k create ns pulp-deployment
k config set-context --current --namespace pulp-deployment

git clone https://github.com/pulp/pulp-operator.git ; cd pulp-operator
NAMESPACE=pulp-deployment make deploy
```

DEPLOY PULP CR

https://github.com/pulp/pulp-operator/tree/main/config/samples

```
k apply -f https://raw.githubusercontent.com/pulp/pulp-operator/main/config/samples/pulpcon-sample.yaml
```

CLEANING UP

```
k delete ns pulp-deployment
k get crd -oname|grep pulp|xargs kubectl delete
k get clusterrolebinding,clusterrole -oname|grep pulp|xargs kubectl delete
```

INSTALLING PULP OPERATOR FROM SOURCE

AS A LINUX PROCESS

```
k create ns pulp-process
k config set-context --current --namespace pulp-process

git clone https://github.com/pulp/pulp-operator.git ; cd pulp-operator
make install
WATCH_NAMESPACE=pulp-process go run main.go --zap-stacktrace-level=panic
```

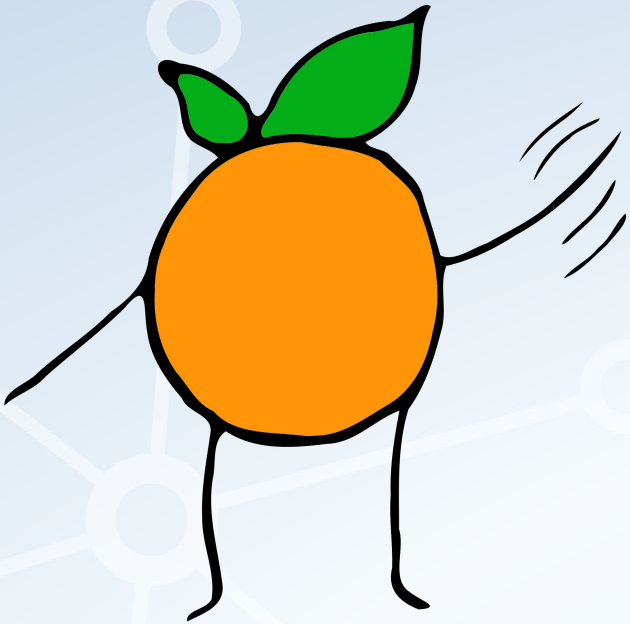
DEPLOY PULP CR

https://github.com/pulp/pulp-operator/tree/main/config/samples

```
k apply -f https://raw.githubusercontent.com/pulp/pulp-operator/main/config/samples/pulpcon-sample.yaml
```

Bonus: Deploying from OCP catalog

Sponsored by Red Hat



THANK YOU!

Questions?

pulpproject.org

pulp