**You have to install necessary tools**

Install minikube - <https://minikube.sigs.k8s.io/docs/start/>

Install kubectl - <https://kubernetes.io/docs/tasks/tools/install-kubectl-linux/>

Install Tekton cli - <https://tekton.dev/docs/cli/>

Install Docker - <https://docs.docker.com/engine/install/ubuntu/>

sudo usermod -aG docker $USER - **add your user to the Docker group**

sudo reboot

Minikube start

Or

minikube start --kubernetes-version v1.24.4

**Install Tekton & Dependency**

kubectl apply --filename \

https://storage.googleapis.com/tekton-releases/pipeline/latest/release.yaml

tkn hub install task kaniko

tkn hub install task git-clone

tkn hub install task kubernetes-actions

**Clone repository with manifest, sources and Tekton files**

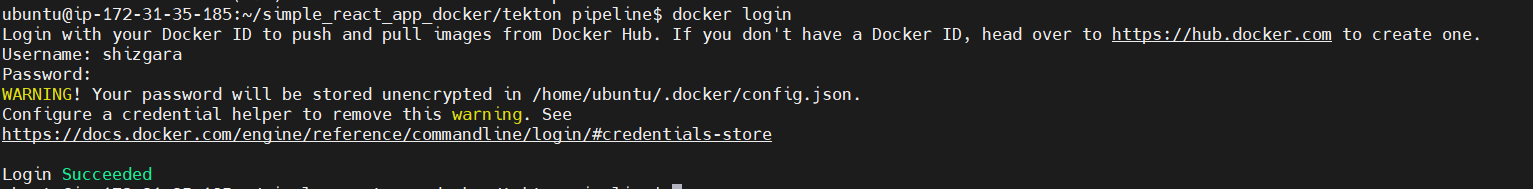
git clone <https://github.com/shizgara/simple_react_app_docker.git>

cd cd simple\_react\_app\_docker/tekton\ pipeline/ - **move to tekton files**

**Generate and apply credentials for Docker**

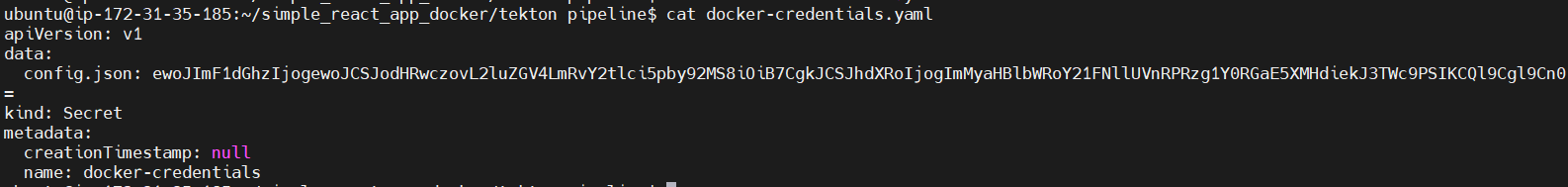
Docker login – **you have to make docker login to generate /home/shizgara/.docker/config.json**.

Enter username and password to DockerHub. When it done, you will see



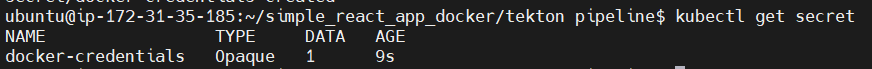
**After that you have to make secret for docker credentials. And run the output file**

kubectl create secret generic docker-credentials --from-file=/home/$USER/.docker/config.json --dry-run=client --output=yaml > docker-credentials.yaml –- > **make docker-credentials.yaml**



kubectl apply -f docker-credentials.yaml – **run credentials**

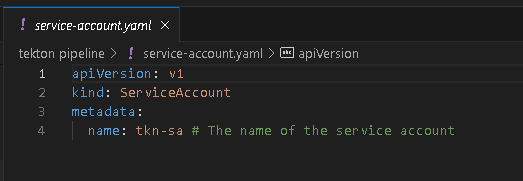
**You can check it –** kubectl get secret



**Run pipeline**

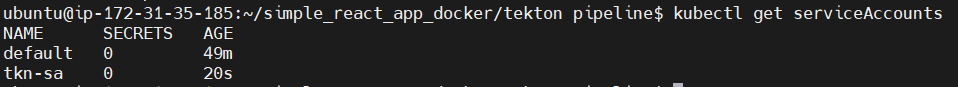
You must be in ~/simple\_react\_app\_docker/tekton pipeline. Here we have few files which to runs our pipeline

**Firstly we create the service account, which allow provides an identity for processes that run in a Pods. Here is the content of service-account.yaml**

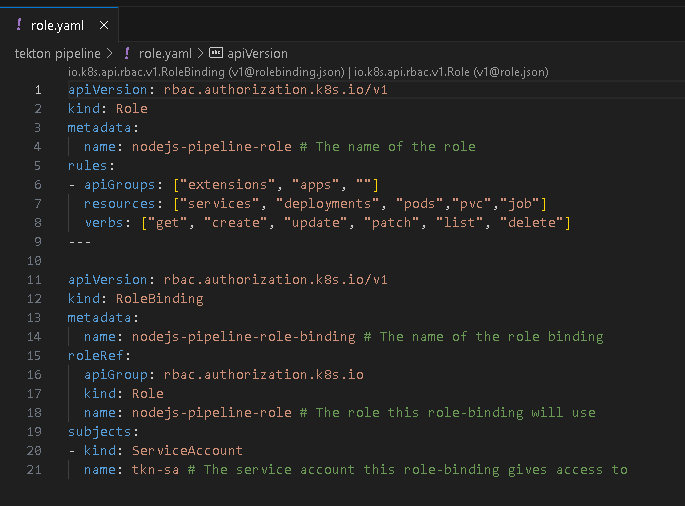
****

**Run service-account.yaml and check it**

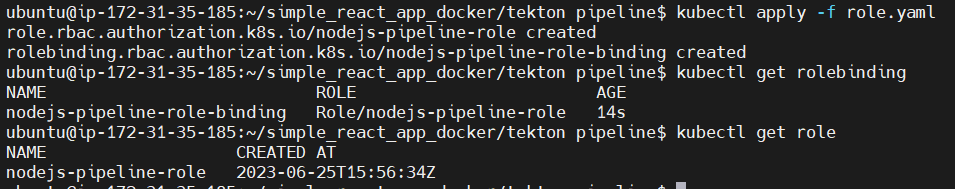
kubectl apply -f service-account.yaml

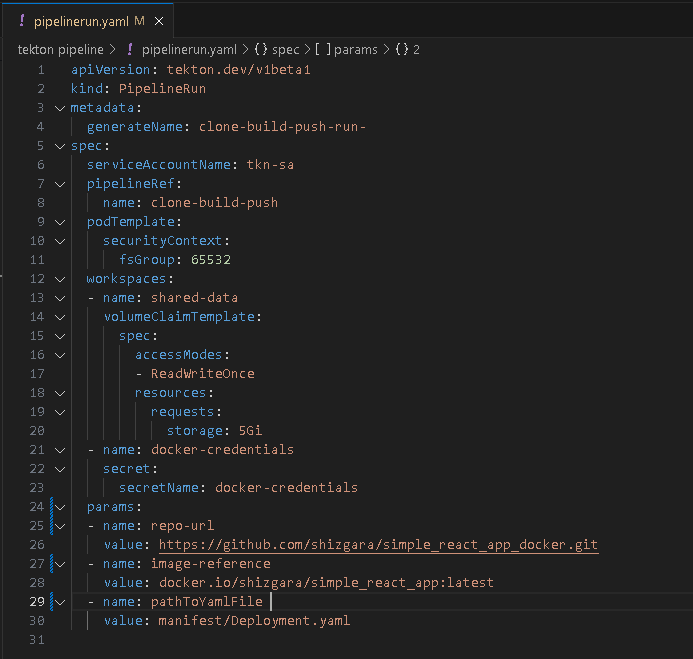
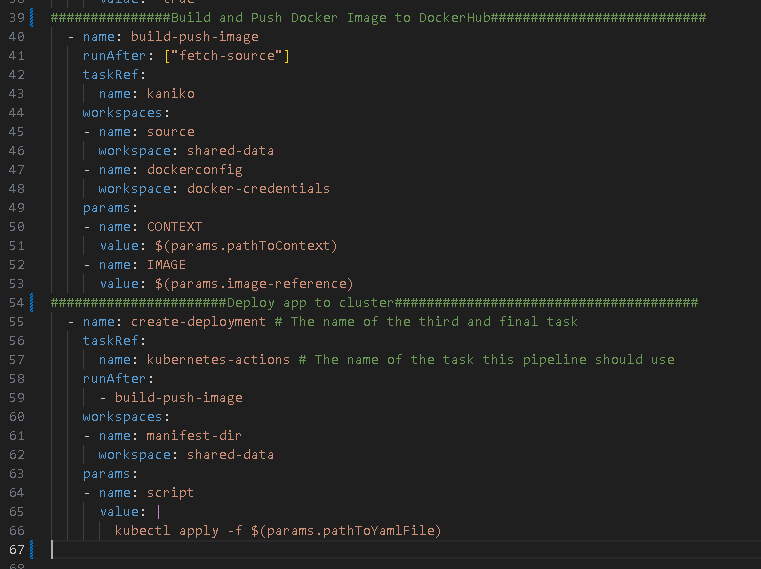
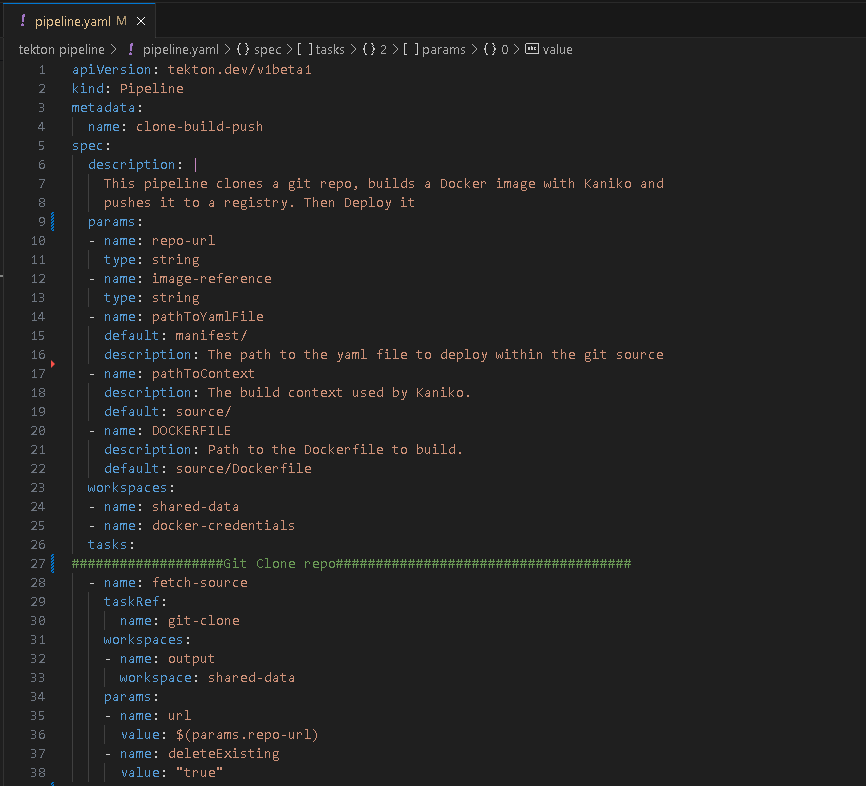


**Secondly we run role.yaml, which create role and bind it with service account created in previous step**

****

kubectl apply -f role.yaml



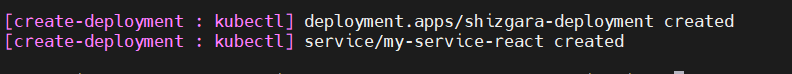
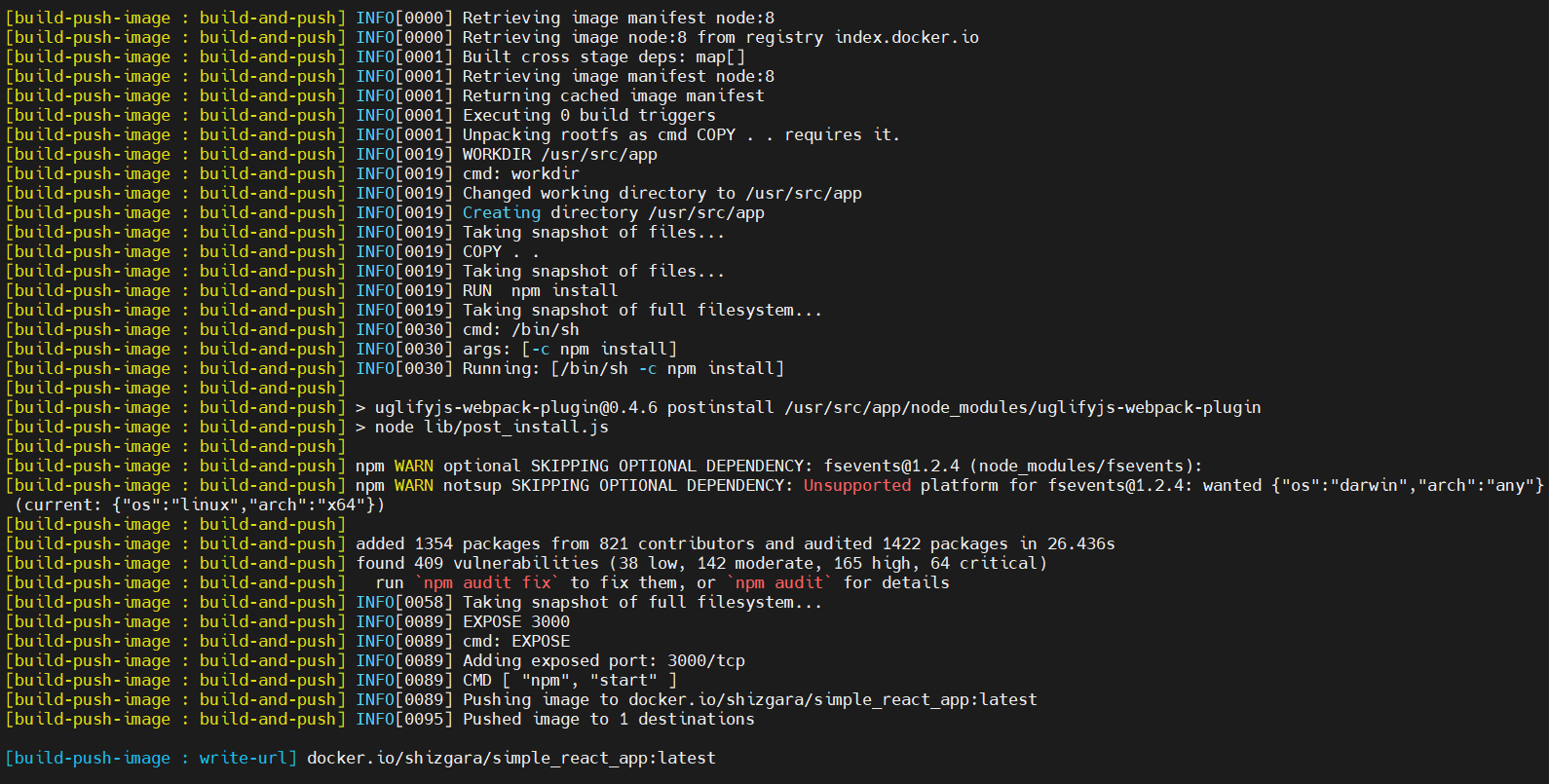
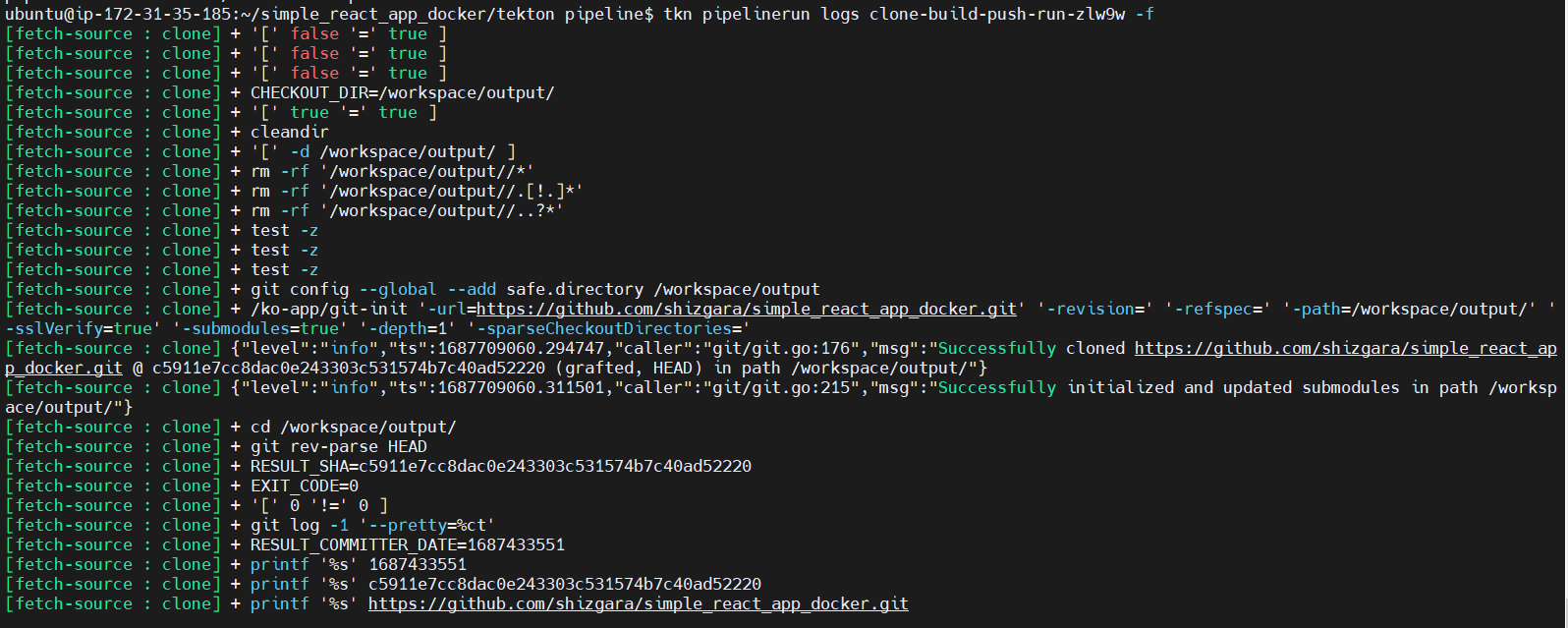
**After that we can apply pipeline with tasks and run it with pipelinerun**

kubectl apply -f pipeline.yaml

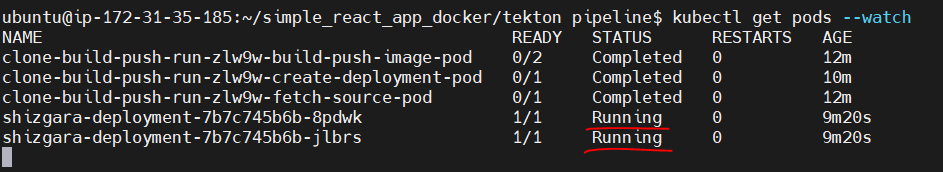
kubectl create -f pipelinerun.yaml

**We can see the logs of pipelinerun**

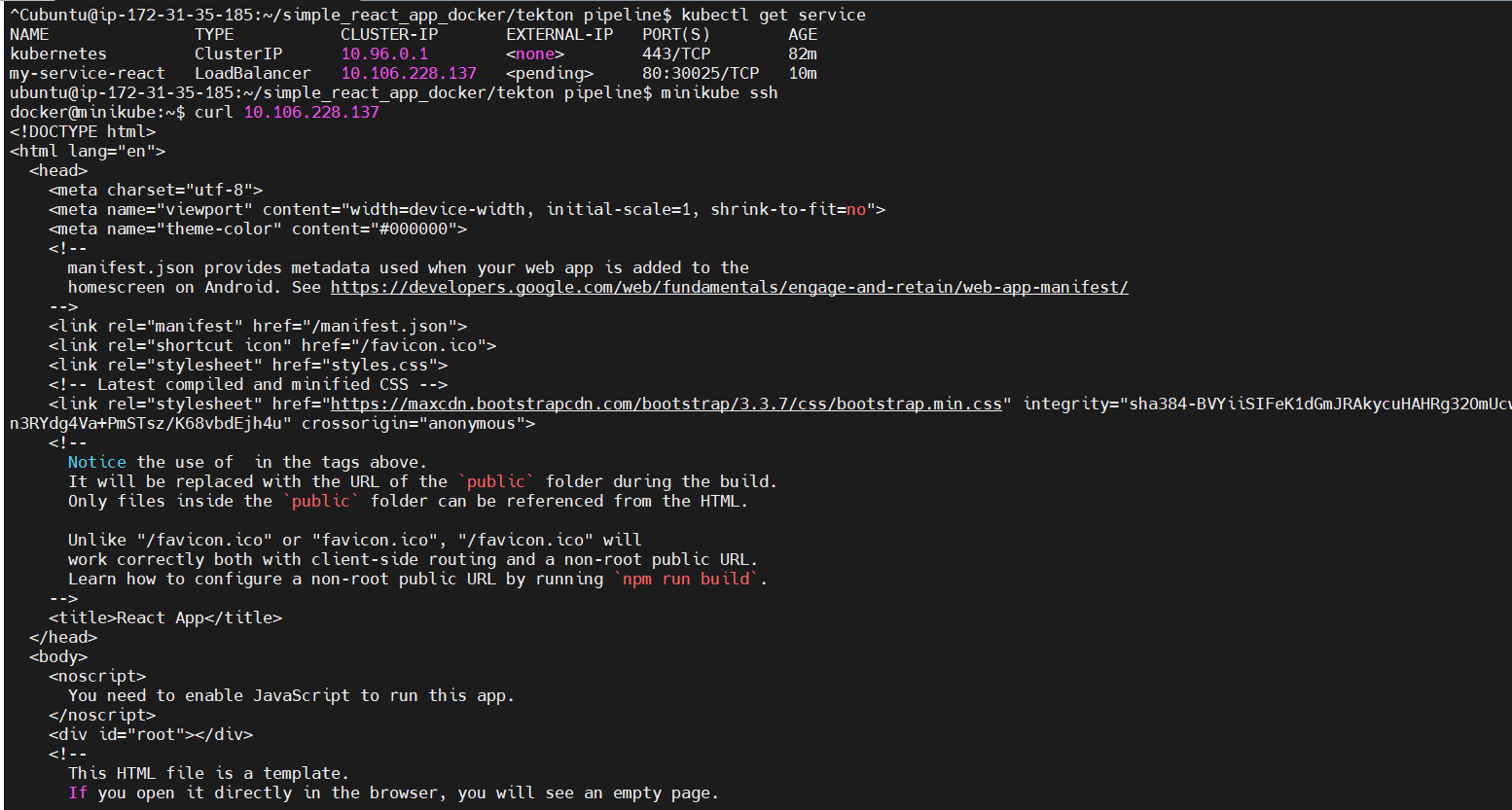
tkn pipelinerun logs <name created pipilinerun> -f



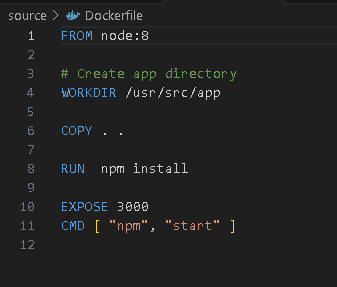
kubectl get pods –watch



**Now we check our app. Does it run?**

****

**Dockerfile for image**

****

**Manifest file, for deployment**

Here we run 2 pods with our app. App running on port 3000. So Load balancer map tcp requests from 80 to 3000 port

****

**Additional commands**

tkn pipeline delete –all

tkn pipelinerun delete --all

kubectl get rolebinding

kubectl get role

kubectl get serviceAccounts

kubectl get secret

kubectl get nodes -o wide

**Links**

<https://tekton.dev/docs/getting-started/tasks/>

<https://adamtheautomator.com/tekton-kubernetes/>

<https://earthly.dev/blog/building-k8s-tekton/>

<https://developer.ibm.com/tutorials/build-and-deploy-a-docker-image-on-kubernetes-using-tekton-pipelines/>

<https://hub.tekton.dev/>

<https://github.com/tektoncd/cli/blob/main/README.md>