Cron Job in K8s

Contents

[Index 1](#_Toc35933492)

[Kubernetes Introduction 1](#_Toc35933493)

[Important terms in K8s 1](#_Toc35933494)

[Cron Job 2](#_Toc35933495)

[Creating a GKE 2](#_Toc35933496)

[Creating a Cron Job 2](#_Toc35933497)

[create the cron job 2](#_Toc35933498)

[Creating directly 3](#_Toc35933499)

[Get the status 3](#_Toc35933500)

[Watch the Job 3](#_Toc35933501)

[References 3](#_Toc35933502)

# Kubernetes Introduction

Platform for managing containerized workloads and services, that facilitates both declarative configuration and automation

## Important terms in K8s

***Node***: Smallest unit of computing **hardware**

**Cluster**: Nodes pool

**Containers**: Self Contained execution environment (Pre build image) to deploy in K8s.

Unit of replication.

***Pods***: Doesn’t run containers directly; it wraps one or more containers into a higher-level structure

**Deployment**: Pods are managed by one more layer of abstraction. If pod dies, the deployment will automatically re-create it.

**Ingress**: Channel for communication to outside.

**Persistent Volume**: File system that can be mounted to the cluster- not associated with any node.

# Cron Job

cron is a Linux utility which schedules a command or script on your server to run automatically at a specified time and date.

\* \* \* \* \*

# ┌───────────── minute (0 - 59)

# │ ┌───────────── hour (0 - 23)

# │ │ ┌───────────── day of the month (1 - 31)

# │ │ │ ┌───────────── month (1 - 12)

# │ │ │ │ ┌───────────── day of the week (0 - 6) (Sunday to Saturday;

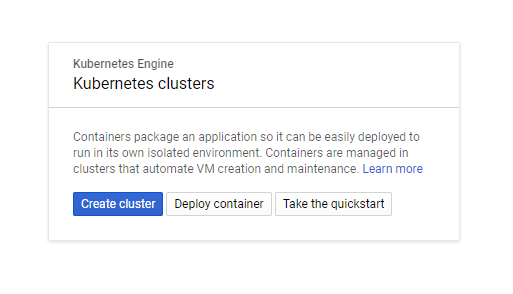
# │ │ │ │ │ 7 is also Sunday on some systems)

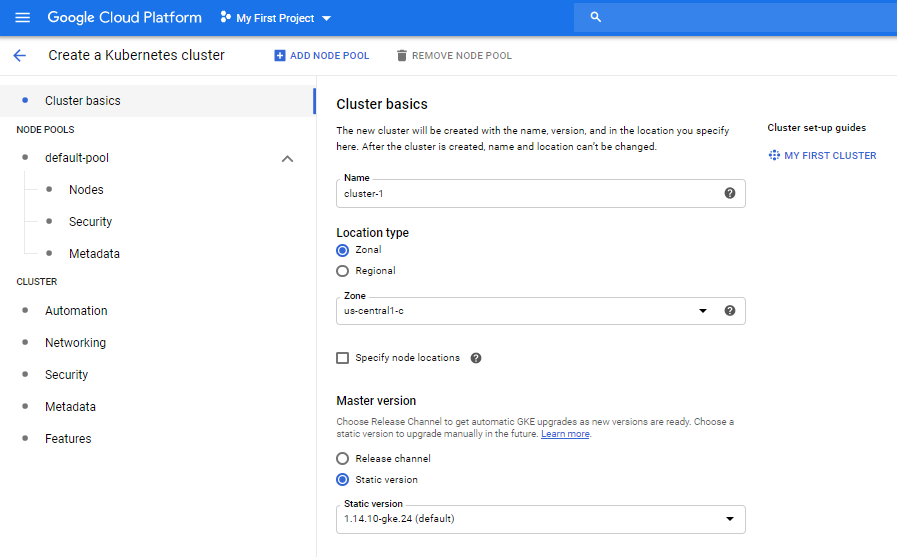
# │ │ │ │ │

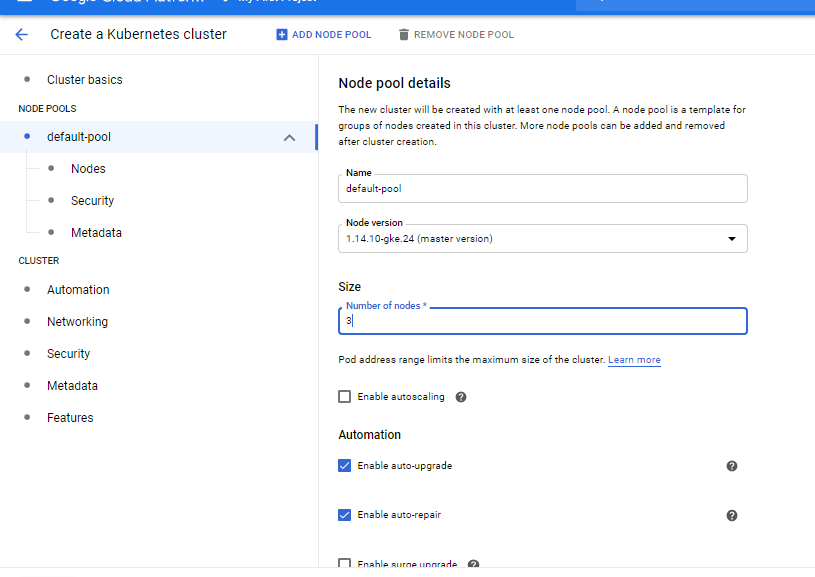
# │ │ │ │ │

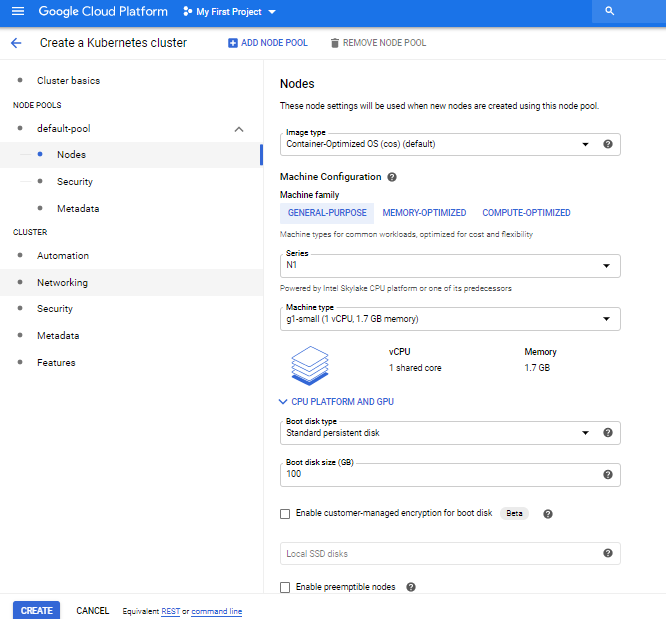
# \* \* \* \* \* command to execute

# Creating a GKE cluster









Setting to the Project

gcloud config set project **project-id**

Setting the Zone

gcloud config set compute/zone us-west1-a

Create Cluster

gcloud container clusters get-credentials **cluster-name**

Deployment

kubectl create deployment hello-server --image=gcr.io/google-samples/hello-app:1.0

Exposing the port

kubectl expose deployment hello-server --type LoadBalancer \

--port 80 --target-port 8080

kubectl get pods

Checking the external IP

kubectl get service hello-server

http://**external-ip**/

Delete the service

kubectl delete service hello-server

Delete the cluster

gcloud container clusters delete **cluster-name**

# Creating a Cron Job

Config file

apiVersion: batch/v1beta1

kind: CronJob

metadata:

name: hello

spec:

schedule: "\*/1 \* \* \* \*"

jobTemplate:

spec:

template:

spec:

containers:

- name: hello

image: busybox

args:

- /bin/sh

- -c

- date; echo Hello from the Kubernetes cluster

restartPolicy: OnFailure

## create the cron job

kubectl create -f <https://github.com/t2run/learning/blob/master/K8s/CronJob/cronjob.yml>

## Creating directly

kubectl run hello --schedule="\*/1 \* \* \* \*" --restart=OnFailure --image=busybox -- /bin/sh -c "date; echo Hello from the Kubernetes cluster"

## Get the status

kubectl get cronjob hello

## Watch the Job

kubectl get jobs --watch

# References

1. <https://kubernetes.io/docs/concepts/>
2. <https://crontab.guru/examples.html>
3. <https://kubernetes.io/docs/tasks/job/automated-tasks-with-cron-jobs/>
4. <https://kubernetes.io/docs/concepts/workloads/controllers/jobs-run-to-completion/#writing-a-job-spec>
5. <https://cloud.google.com/kubernetes-engine/docs/how-to/creating-a-cluster>