Kubernets cronjobs and Jobs

[root@k-master yamls]# vim 2-job.yaml

apiVersion: batch/v1

kind: Job

metadata:

name: helloworld

spec:

template:

spec:

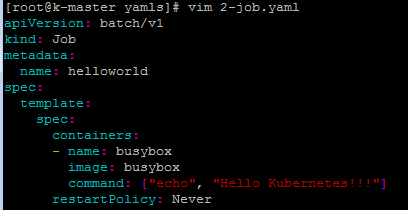
containers:

- name: busybox

image: busybox

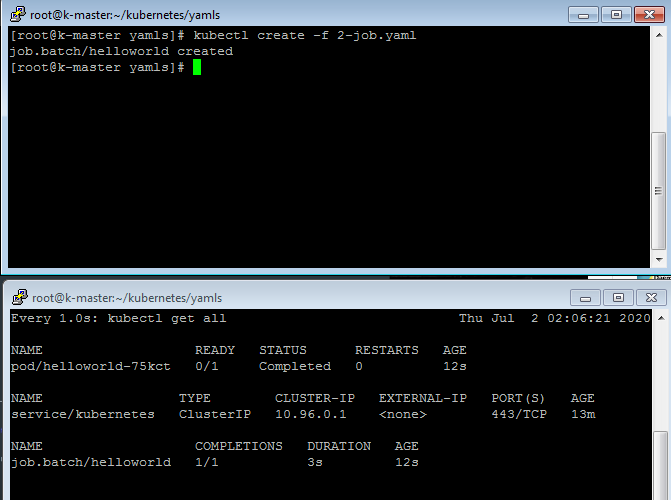
command: ["echo", "Hello Kubernetes!!!"]

restartPolicy: Never

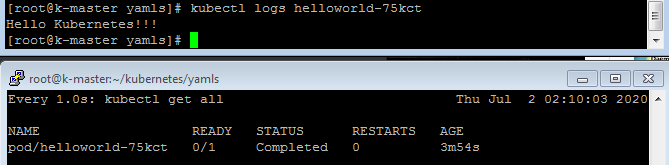


kubectl create -f 2-job.yaml

kubectl get all

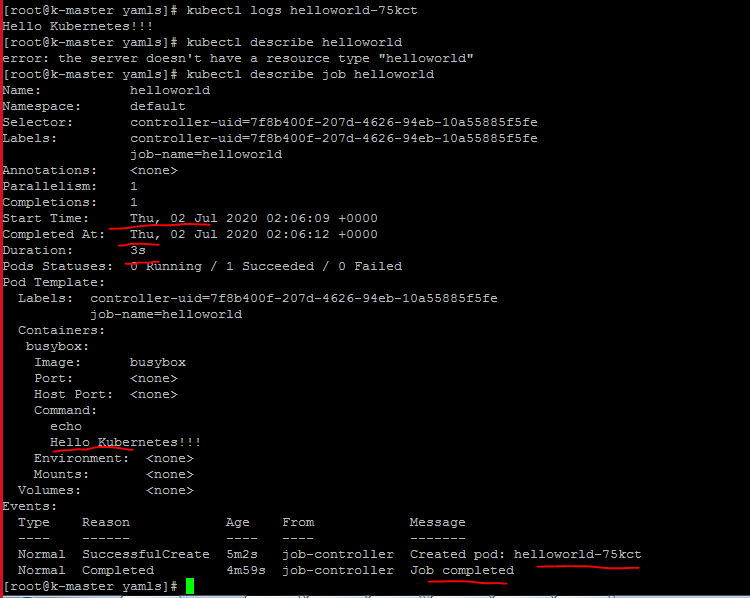


kubectl logs helloworld-75kct

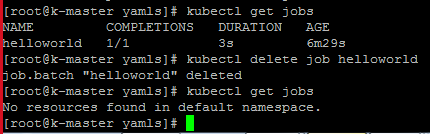


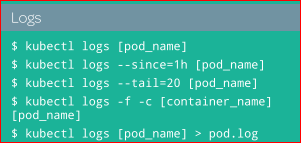
kubectl get jobs

kubectl describe job helloworld



kubectl delete job helloworld





[root@k-master yamls]# vim 2-job.yaml

apiVersion: batch/v1

kind: Job

metadata:

name: helloworld

spec:

template:

spec:

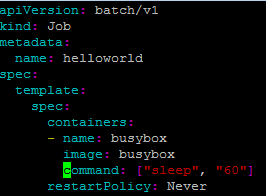
containers:

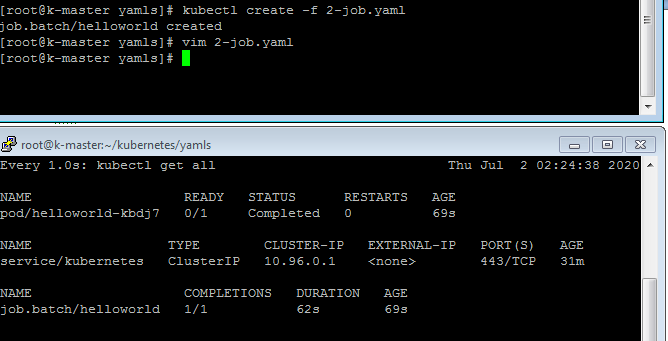
- name: busybox

image: busybox

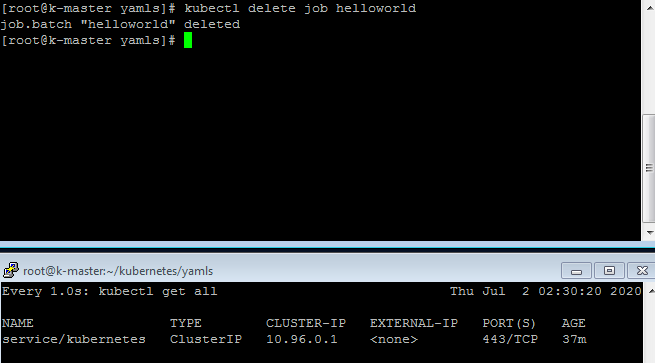
command: ["sleep", "60"]

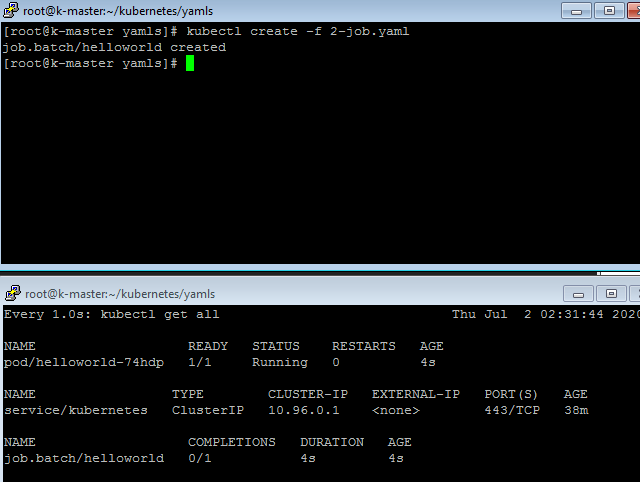
restartPolicy: Never

4

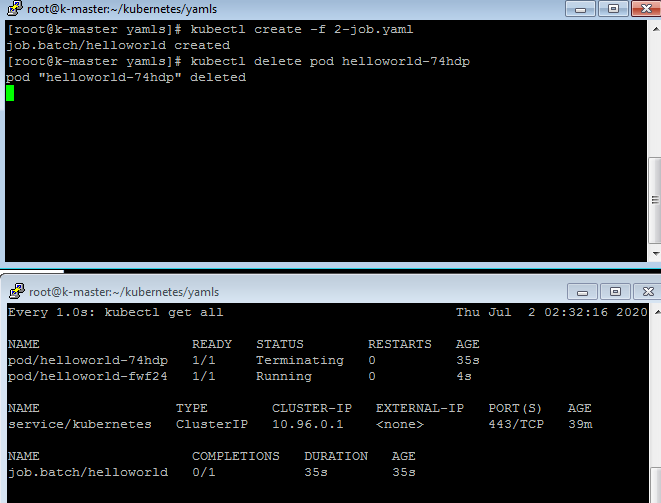


(if we remove job here, it will remove pod automatically)



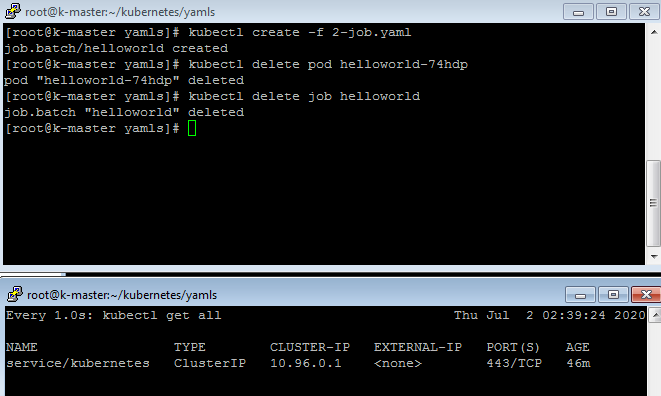


Try to remove pod



If we delete the pod, it will re-create pods until unless job is completed

(here, if we remove job, the pod is getting removed automatically)



Here we add completions for 2 times to run the job

[root@k-master yamls]# vim 2-job.yaml

apiVersion: batch/v1

kind: Job

metadata:

name: helloworld

spec:

completions: 2

template:

spec:

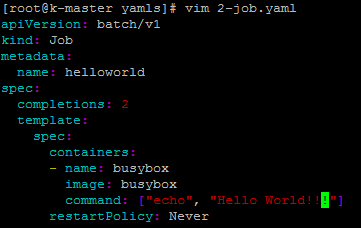
containers:

- name: busybox

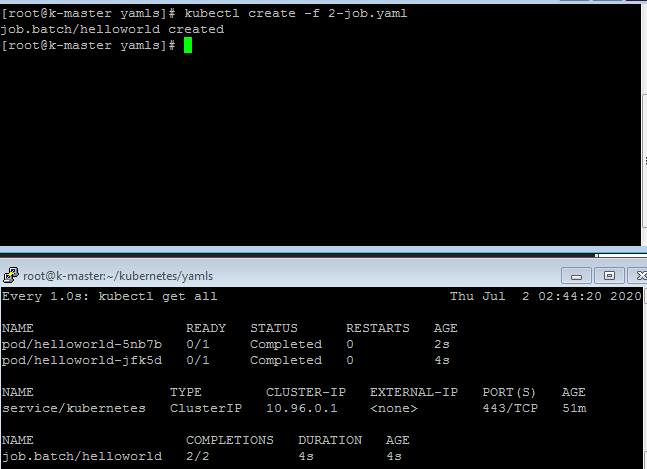
image: busybox

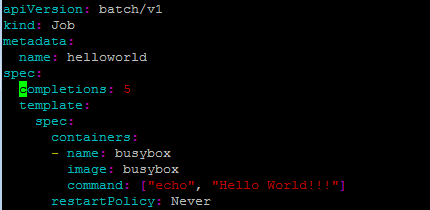
command: ["echo", "Hello World!!!"]

restartPolicy: Never

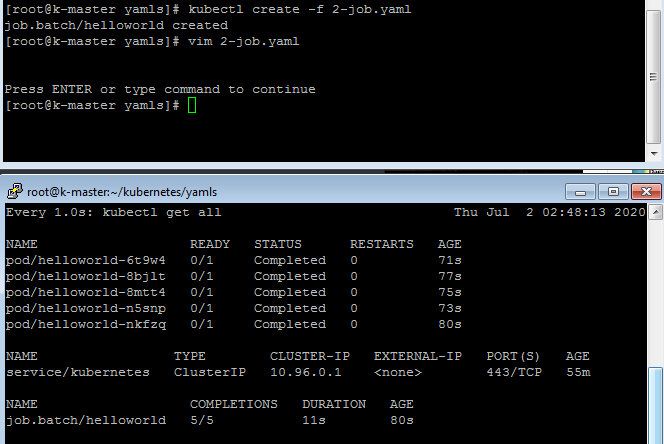


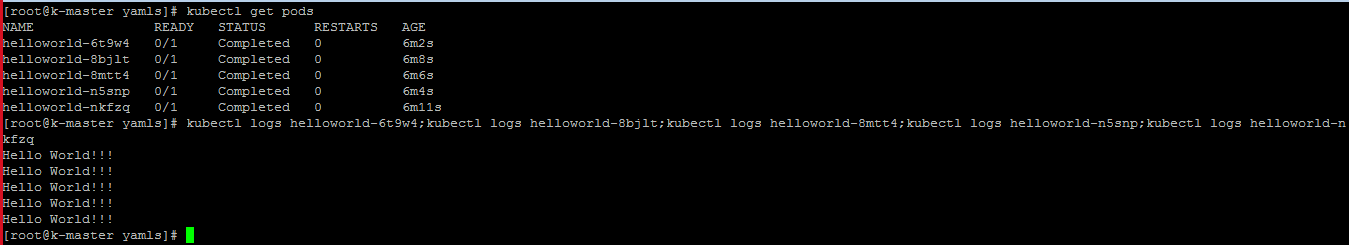
(here, you can it run 2times the same job)





(To complete 5 times job,it create 5 pods and get delete automatically once job comlpeted)





[root@k-master yamls]# kubectl logs helloworld-6t9w4 --since=1h

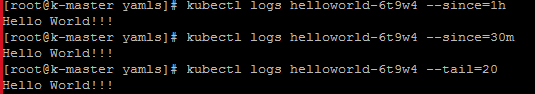
Hello World!!!

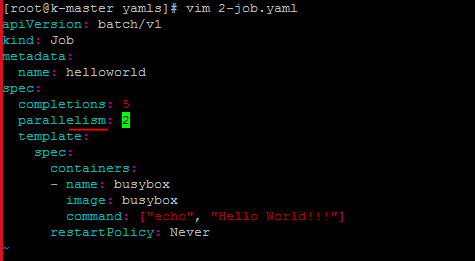
[root@k-master yamls]# kubectl logs helloworld-6t9w4 --since=30m

Hello World!!!

[root@k-master yamls]# kubectl logs helloworld-6t9w4 --tail=20

Hello World!!!





[root@k-master yamls]# vim 2-job.yaml

apiVersion: batch/v1

kind: Job

metadata:

name: helloworld

spec:

completions: 5

parallelism: 2

template:

spec:

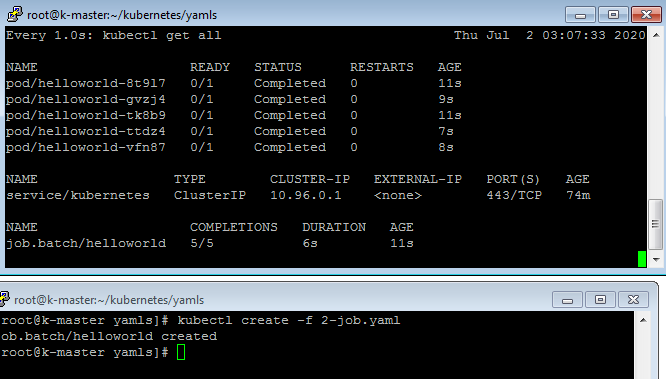
containers:

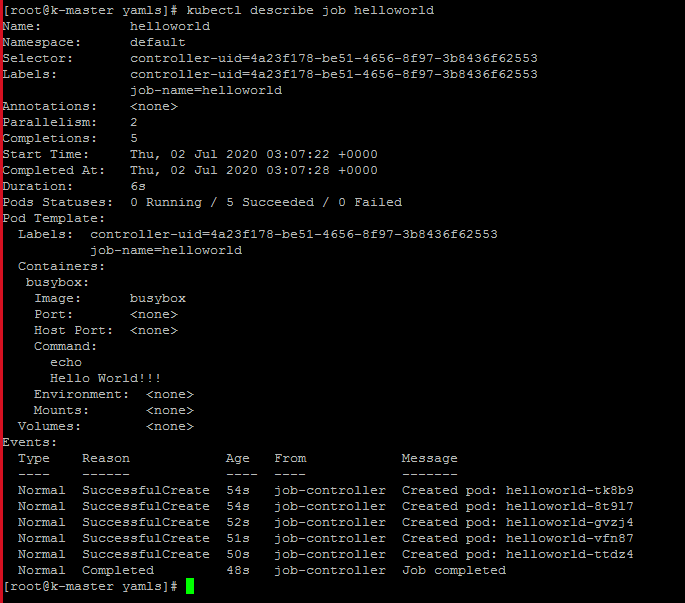
- name: busybox

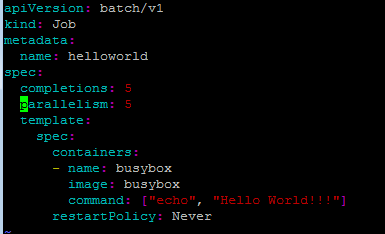
image: busybox

command: ["echo", "Hello World!!!"]

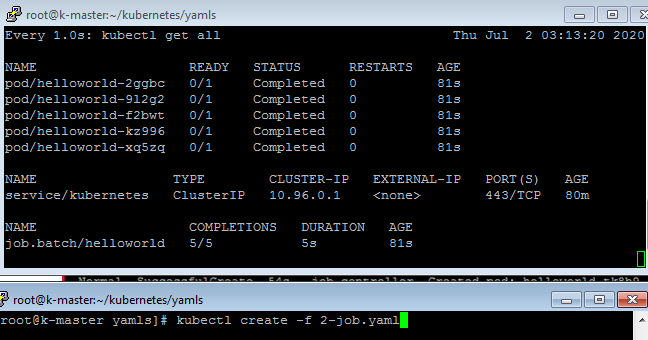
restartPolicy: Never







(pod will create bundle wise to complete thejobs)



(here, we are going to run job to list /sahoo dir,as sahoo dis isnot exist, it will fail, so pod will re-try to create to complete the job, so it will tr-try to create the pods again and again in every 120 sec)

apiVersion: batch/v1

kind: Job

metadata:

name: helloworld

spec:

template:

spec:

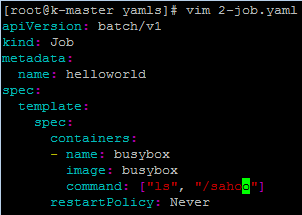
containers:

- name: busybox

image: busybox

command: ["ls", "/sahoo"]

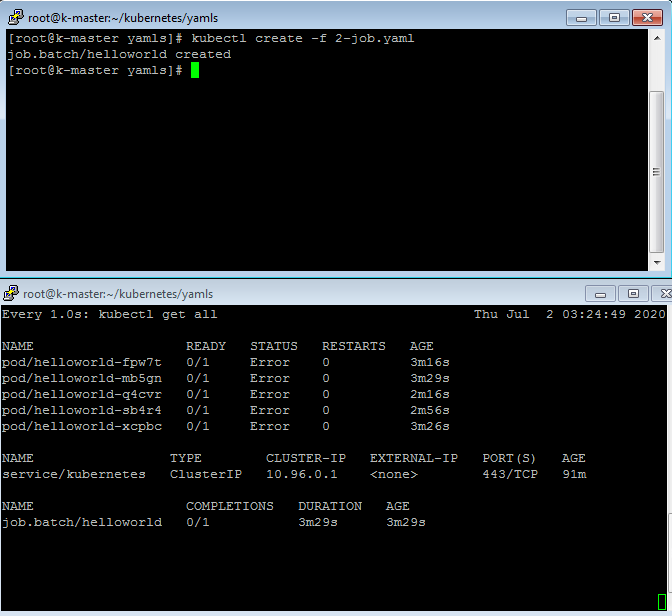
restartPolicy: Never



Lets delete old job first

kubectl delete job helloworld

Create same job again



Here, we are adding backoffLimit

[root@k-master yamls]# vim 2-job.yaml

apiVersion: batch/v1

kind: Job

metadata:

name: helloworld

spec:

backoffLimit: 1

template:

spec:

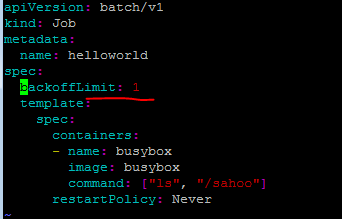
containers:

- name: busybox

image: busybox

command: ["ls", "/sahoo"]

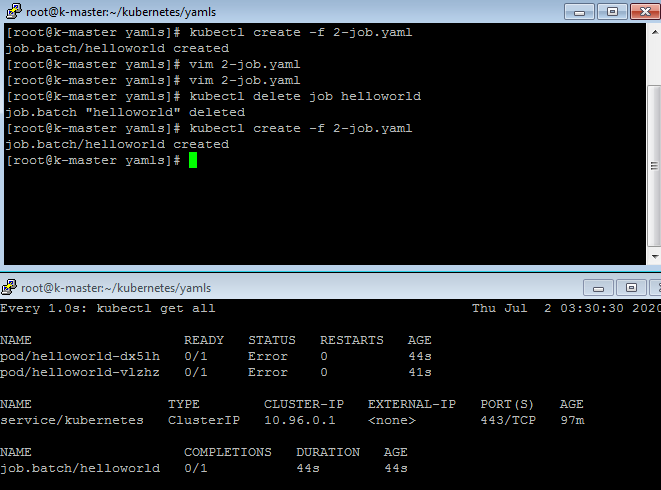
restartPolicy: Never



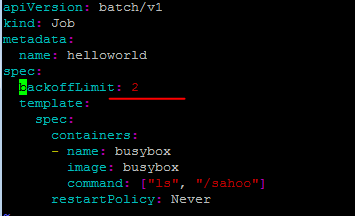
Lets remove existing job.

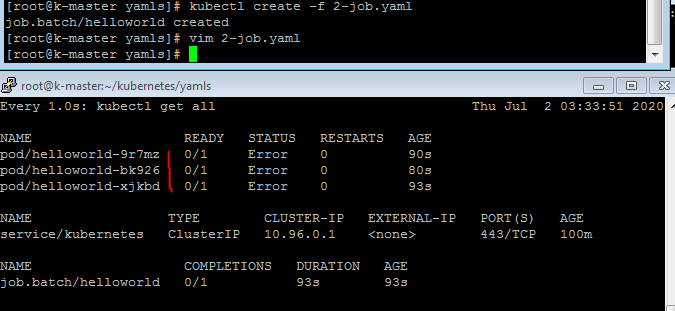
We areadding backupoffLimit here, because it will re-create the pod

You can see here, as the job getting fail. It will try to create the pod as per backupoffLimit n+1=2



Lets change the backuplimit 2, so it will try for 3 times to create pod





(here, we are adding **activeDeadlineSeconds,** so that after 10 secon, the pod will terminate after 10 second automatically)

apiVersion: batch/v1

kind: Job

metadata:

name: helloworld

spec:

activeDeadlineSeconds: 10

template:

spec:

containers:

- name: busybox

image: busybox

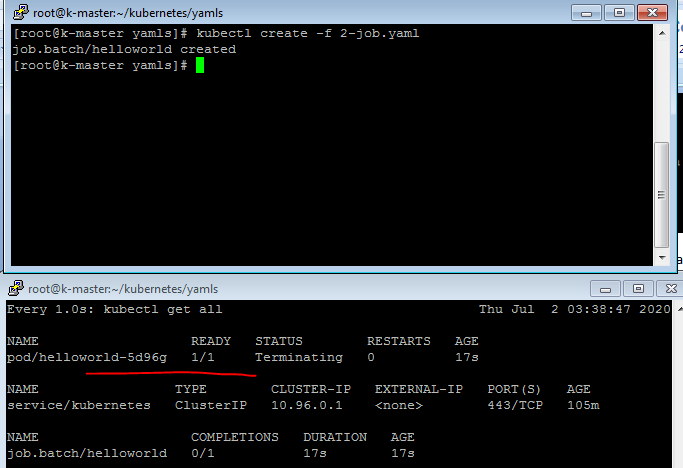
command: ["sleep", "60"]

restartPolicy: Never

Lets remove old job first

kubectl delete job helloworld

(the pod get terminate automatic after 10 sec)



**How to schedule the job**

[root@k-master yamls]# vim 2-cronjob.yaml

apiVersion: batch/v1beta1

kind: CronJob

metadata:

name: helloworld-cron

spec:

schedule: "\* \* \* \* \*"

jobTemplate:

spec:

template:

spec:

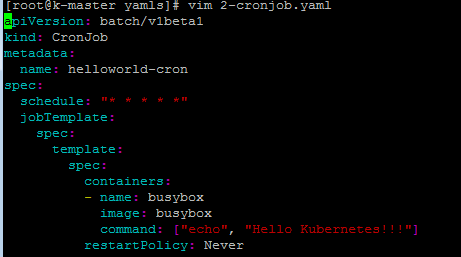
containers:

- name: busybox

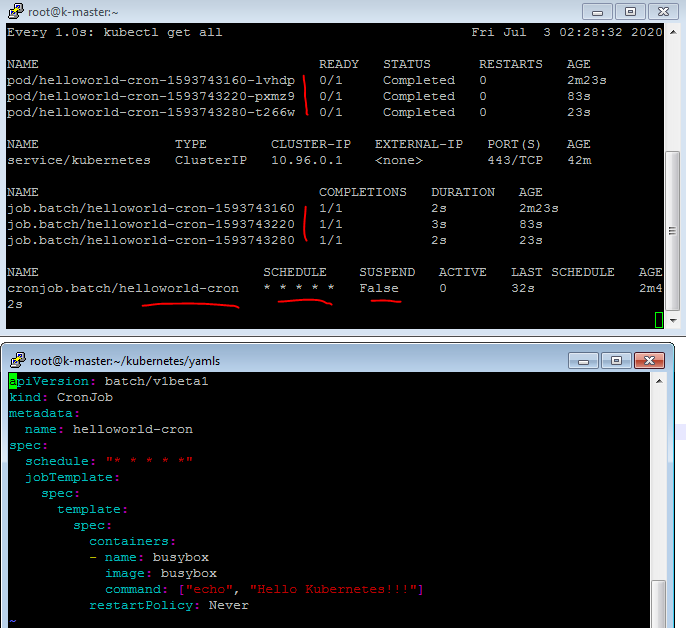
image: busybox

command: ["echo", "Hello Kubernetes!!!"]

restartPolicy: Never



The job will run on every minute like below



How to change successful and failed job history limit

[root@k-master yamls]# vim 2-cronjob.yaml

apiVersion: batch/v1beta1

kind: CronJob

metadata:

name: helloworld-cron

spec:

schedule: "\* \* \* \* \*"

successfulJobsHistoryLimit: 0

failedJobsHistoryLimit: 0

jobTemplate:

spec:

template:

spec:

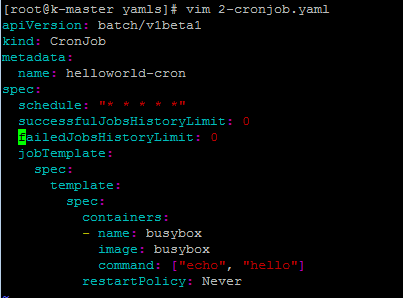
containers:

- name: busybox

image: busybox

command: ["echo", "hello"]

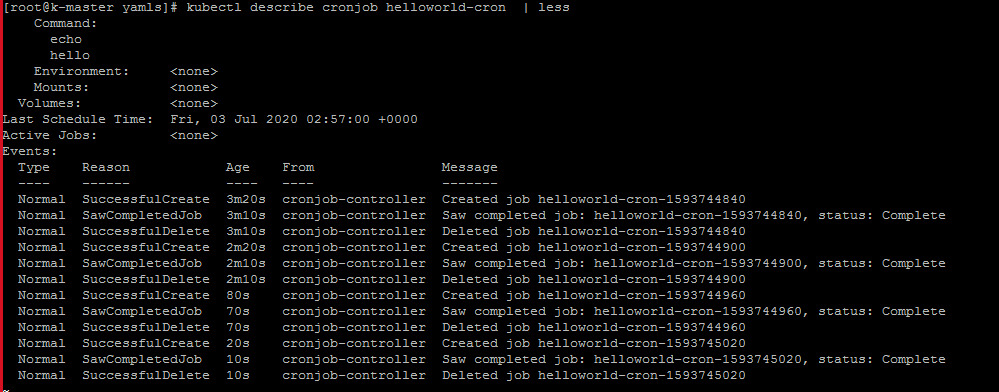
restartPolicy: Never

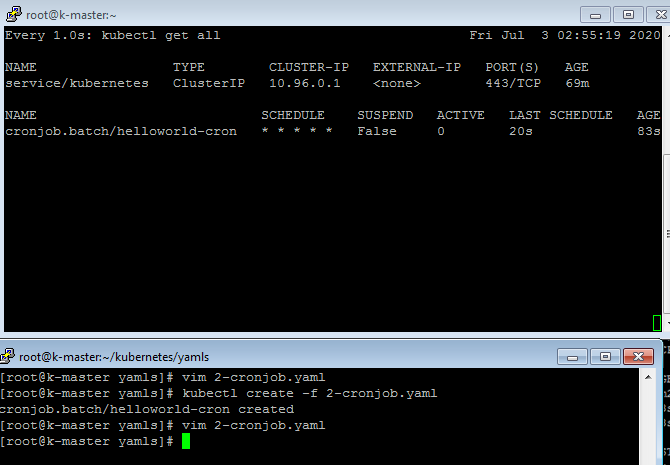


kubectl create -f 2-cronjob.yaml

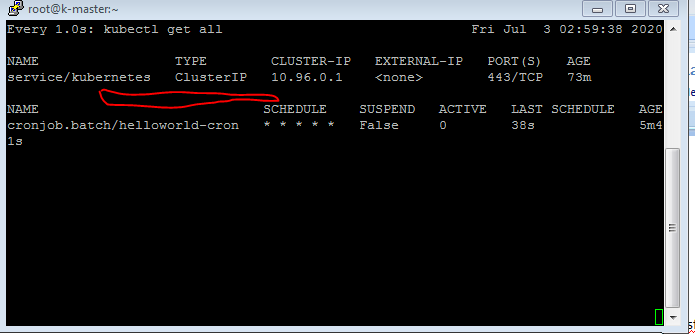
kubectl describe cronjob helloworld-cron | less

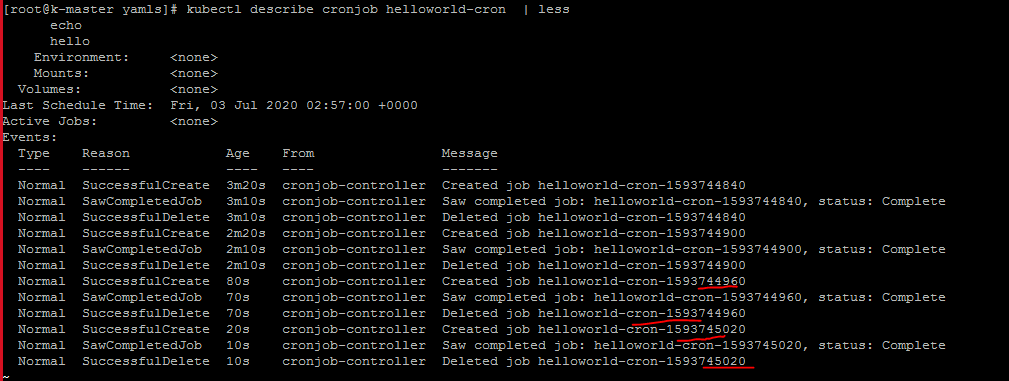
you can see here, job history is getting delete automatic, because we have set sussessfulJobHistoryLimit is set 0

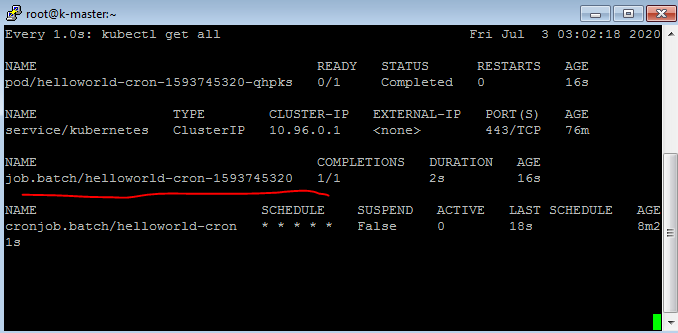




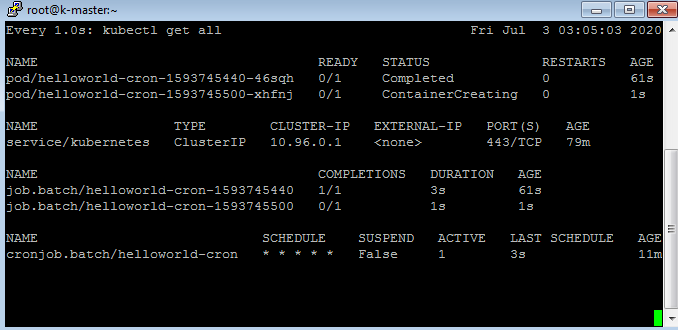
(you can see here, jobhistory not coming here)



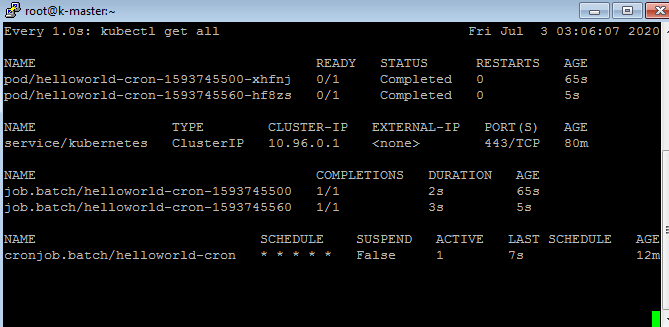




New JOb history creates job create after 60 sec



Oldjob histry dlete after 70 sec



We have added to maintain failJobHistoryLimit,so it keep history limit up to 7, it create on every 1 minute

apiVersion: batch/v1beta1

kind: CronJob

metadata:

name: helloworld-cron

spec:

schedule: "\* \* \* \* \*"

successfulJobsHistoryLimit: 1

failedJobsHistoryLimit: 7

jobTemplate:

spec:

template:

spec:

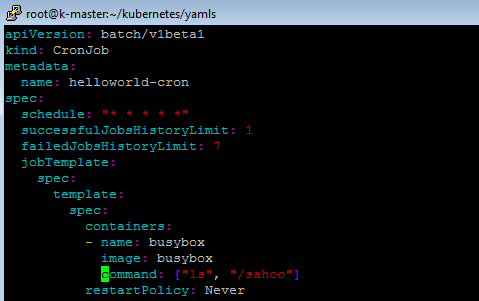
containers:

- name: busybox

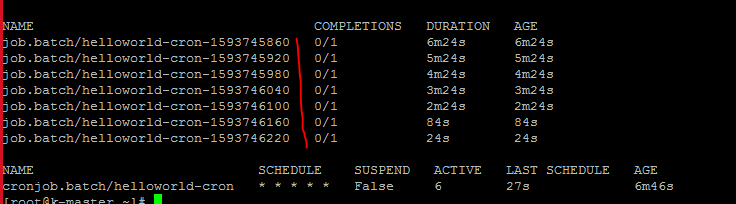
image: busybox

command: ["ls", "/sahoo"]

restartPolicy: Never



We have set failhistorylimit 7



If any wrong job is getting failed again and again,you want to suspend job,then we have to add below

apiVersion: batch/v1beta1

kind: CronJob

metadata:

name: helloworld-cron

spec:

schedule: "\* \* \* \* \*"

suspend: true

jobTemplate:

spec:

template:

spec:

containers:

- name: busybox

image: busybox

command: ["ls", "/sahoo"]

restartPolicy: Never

pod/helloworld-cron-1593746580-b5hgx 0/1 Error 0 2m24s

pod/helloworld-cron-1593746580-c2h6f 0/1 Error 0 2m14s

pod/helloworld-cron-1593746580-dxt6z 0/1 Error 0 2m27s

pod/helloworld-cron-1593746580-zrz6z 0/1 Error 0 94s

pod/helloworld-cron-1593746640-djrjl 0/1 Error 0 35s

pod/helloworld-cron-1593746640-dltdh 0/1 Error 0 75s

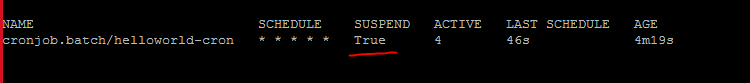
pod/helloworld-cron-1593746640-mpbff 0/1 Error 0 87s

pod/helloworld-cron-1593746640-vwbr8 0/1 Error 0 85s

pod/helloworld-cron-1593746700-b2tdh 0/1 Error 0 14s

pod/helloworld-cron-1593746700-m4l9g 0/1 Error 0 24s

pod/helloworld-cron-1593746700-sftqb 0/1 Error 0 27s



**To Suspend the job through command line**

[root@k-master yamls]# vim 2-cronjob.yaml

metadata:

name: helloworld-cron

spec:

schedule: "\* \* \* \* \*"

jobTemplate:

spec:

template:

spec:

containers:

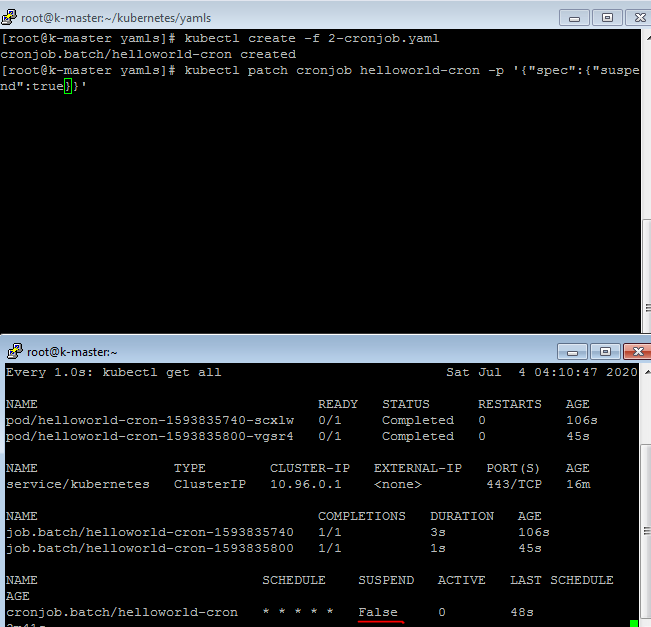
- name: busybox

image: busybox

command: ["echo", "Hello Kubernetes!!!"]

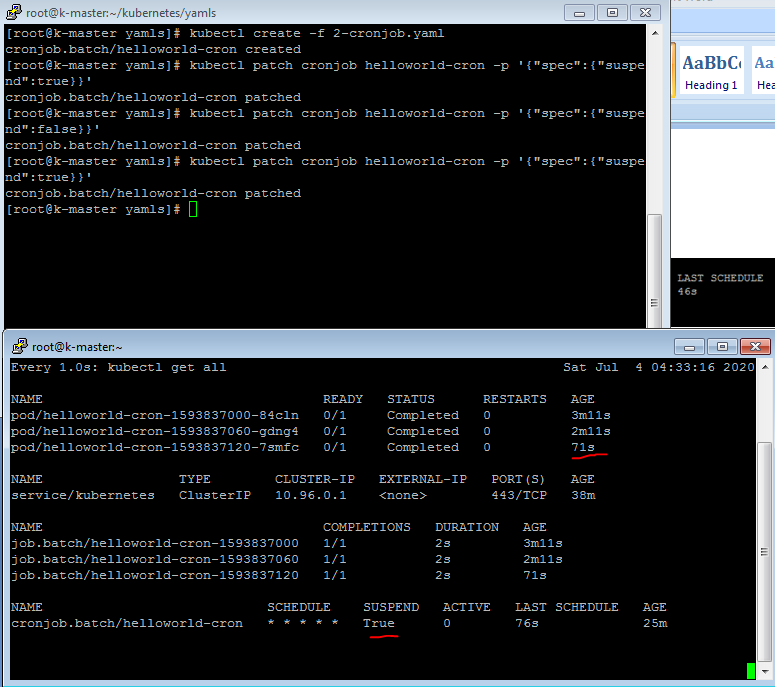
restartPolicy: Never

Before adding patch, suspend status false here



After applying patch

kubectl patch cronjob helloworld-cron -p '{"spec":{"suspend":true}}'



**Concurrency Policy 3 types**

concurrencyPolicy Allow :- more than 1 job can run at same time

concurrencyPolicy Forbid : if job running, next job will wait till the job complete

concurrencyPolicy Replace : if job is running, another job in queue, it will replace old job

Pending