GCP Certification Series: 4.2 Managing Kubernetes Engine resources

Prashanta Paudel

Nov 7, 2018 · 5 min read

We have already learn introductory and basic functions in earlier topic so we will directly go into doing tasks mentioned in syllabus.

Viewing current running cluster inventory (nodes, pods, services)

After the kubernetes clusters are up and running you can check the status of the nodes,pods and services.

First let us make two clusters in Europe north 1 and asisa east1, both regional cluster.

```
prashantagcppaudel@cloudshell:~ (fourpointtwo-221807)$
gcloud container clusters create asia-east-cluster --
machine-type=f1-micro --num-nodes=2 --region=asia-east1
WARNING: Starting in 1.12, new clusters will have basic
authentication disabled by default. Basic authentication can
be enabled (or disabled) manually using the `--[no-]enable-
basic-auth` flag.
WARNING: Starting in 1.12, new clusters will not have a
client certificate issued. You can manually enable (or
disable) the issuance of the client certificate using the `-
-[no-]issue-client-certificate` flag.
WARNING: Currently VPC-native is not the default mode during
cluster creation. In the future, this will become the
default mode and can be disabled using `--no-enable-ip-
alias` flag. Use `--[no-]enable-ip-alias` flag to suppress
this warning.
WARNING: Starting in 1.12, default node pools in new
clusters will have their legacy Compute Engine instance
metadata endpoints disabled by default. To create a cluster
with legacy instance metadata endpoints disabled in the
default node pool, run `clusters create` with the flag `--
metadata disable-legacy-endpoints=true`.
This will enable the autorepair feature for nodes. Please
see https://cloud.google.com/kubernetes-engine/docs/node-
auto-repair formore information on node autorepairs.
WARNING: Starting in Kubernetes v1.10, new clusters will no
longer get compute-rw and storage-ro scopes added to what is
specifiedin --scopes (though the latter will remain included
in the default --scopes). To use these scopes, add them
```

```
explicitly to --scopes. To use the new behavior, set
container/new_scopes_behavior property (gcloud config set
container/new_scopes_behavior true).
Creating cluster asia-east-cluster in asia-east1...done.
Created
[https://container.googleapis.com/v1/projects/fourpointtwo-
221807/zones/asia-east1/clusters/asia-east-cluster].
To inspect the contents of your cluster, go to:
https://console.cloud.google.com/kubernetes/workload_/gcloud
/asia-east1/asia-east-cluster?project=fourpointtwo-221807
kubeconfig entry generated for asia-east-cluster.
NAME               LOCATION    MASTER_VERSION  MASTER_IP
MACHINE_TYPE  NODE_VERSION  NUM_NODES  STATUS
asia-east-cluster  asia-east1  1.9.7-gke.6     35.221.167.78
f1-micro      1.9.7-gke.6   6          RUNNING


prashantagcppaudel@cloudshell:~ (fourpointtwo-221807)$
gcloud container clusters create europe-cluster --machine-
type=n1-standard-1 --num-nodes=2 --region=europe-north1
WARNING: Starting in 1.12, new clusters will have basic
authentication disabled by default. Basic authentication can
be enabled (or disabled) manually using the `--[no-]enable-
basic-auth` flag.
WARNING: Starting in 1.12, new clusters will not have a
client certificate issued. You can manually enable (or
disable) the issuance of the client certificate using the `-
-[no-]issue-client-certificate` flag.
WARNING: Currently VPC-native is not the default mode during
cluster creation. In the future, this will become the
default mode and can be disabled using `--no-enable-ip-
alias` flag. Use `--[no-]enable-ip-alias` flag to suppress
this warning.
WARNING: Starting in 1.12, default node pools in new
clusters will have their legacy Compute Engine instance
metadata endpoints disabled by default. To create a cluster
with legacy instance metadata endpoints disabled in the
default node pool, run `clusters create` with the flag `--
metadata disable-legacy-endpoints=true`.
This will enable the autorepair feature for nodes. Please
see https://cloud.google.com/kubernetes-engine/docs/node-
auto-repair formore information on node autorepairs.
WARNING: Starting in Kubernetes v1.10, new clusters will no
longer get compute-rw and storage-ro scopes added to what is
specifiedin --scopes (though the latter will remain included
in the default --scopes). To use these scopes, add them
explicitly to --scopes. To use the new behavior, set
container/new_scopes_behavior property (gcloud config set
container/new_scopes_behavior true).
Creating cluster europe-cluster in europe-north1...done.
Created
[https://container.googleapis.com/v1/projects/fourpointtwo-
221807/zones/europe-north1/clusters/europe-cluster].
To inspect the contents of your cluster, go to:
https://console.cloud.google.com/kubernetes/workload_/gcloud
/europe-north1/europe-cluster?project=fourpointtwo-221807
kubeconfig entry generated for europe-cluster.
NAME            LOCATION       MASTER_VERSION  MASTER_IP
MACHINE_TYPE   NODE_VERSION  NUM_NODES  STATUS
europe-cluster  europe-north1  1.9.7-gke.6     35.228.64.238
n1-standard-1  1.9.7-gke.6   6          RUNNING
```

To list the clusters running in Kubernetes

```
$ gcloud container clusters list
NAME              LOCATION       MASTER_VERSION  MASTER_IP
MACHINE_TYPE   NODE_VERSION  NUM_NODES  STATUS
asia-east-cluster  asia-east1     1.9.7-gke.6
35.221.167.78  f1-micro       1.9.7-gke.6   6
RUNNING
europe-cluster     europe-north1  1.9.7-gke.6
35.228.64.238  n1-standard-1  1.9.7-gke.6   6
RUNNING
```

Now add some workloads to this cluster

$ kubectl run app—image gcr.io/google-samples/hello-app:1.0

To view pods

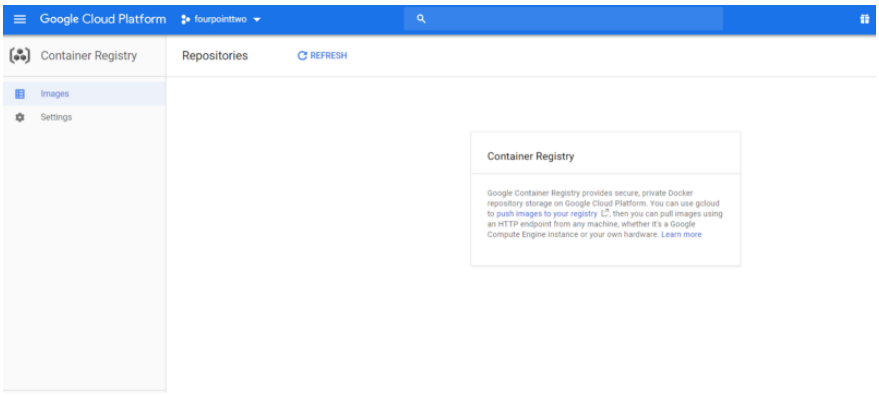
#kubectl get pods

```
gcloud compute instances list


prashantagcppaudel@cloudshell:~ (fourpointtwo-221807)$
kubectl get deployment
NAME           DESIRED   CURRENT   UP-TO-DATE   AVAILABLE
AGE
frontend       3         3         3            3
2h
redis-master   1         1         1            1
2h
redis-slave    2         2         2            2
2h
prashantagcppaudel@cloudshell:~ (fourpointtwo-221807)$
```

Browsing the container image registry

First, go to Console and then to Container Registry, where you can see the container images.



Container Registry

Working with the nodes

You add more replicas to your application's Deployment resource by using the `kubectl scale` command. To add two additional replicas to your Deployment (for a total of three), run the following command:

```
kubectl scale deployment hello-web --replicas=3
```

You can see the new replicas running on your cluster by running the following commands:

```
kubectl get deployment hello-web
```

Output:

```
NAME        DESIRED   CURRENT   UP-TO-DATE   AVAILABLE   AGE
hello-web   3         3         3            2           1m

kubectl get pods
```

Output:

```
NAME                          READY     STATUS    RESTARTS
AGE
hello-web-4017757401-ntgdb    1/1       Running   0
9s
hello-web-4017757401-pc4j9    1/1       Running   0
9s
hello-web-4017757401-px7tx    1/1       Running   0
1m
```

Now, you have multiple instances of your application running independently of each other and you can use the `kubectl scale` command to adjust the capacity of your application.

The load balancer you provisioned in the previous step will start routing traffic to these new replicas automatically.

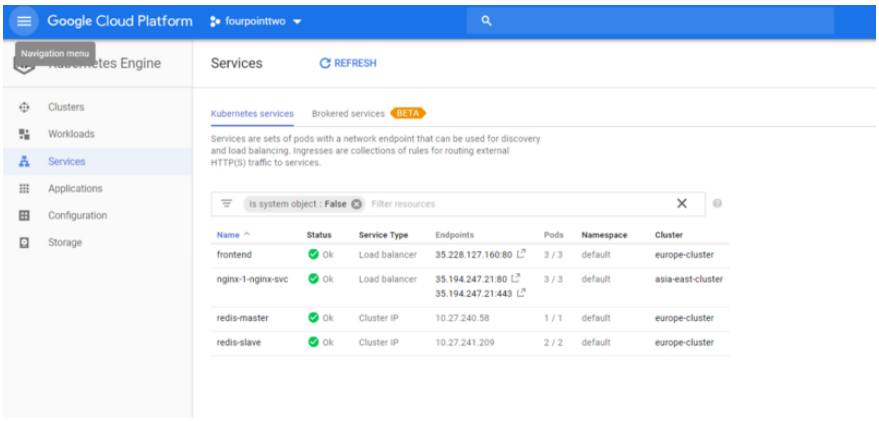
Working with Pods

Once the kubernetes is up and running you can view and edit pods.

```
prashantagcppaudel@cloudshell:~ (fourpointtwo-221807)$
kubectl get pods
NAME                              READY     STATUS    RESTARTS
AGE
frontend-67f65745c-c7j8q          1/1       Running   0
2h
frontend-67f65745c-q7hnl          1/1       Running   0
2h
frontend-67f65745c-rm2s8          1/1       Running   0
2h
redis-master-585798d8ff-9mbfz     1/1       Running   0
2h
redis-slave-865486c9df-m6gwd      1/1       Running   0
2h
redis-slave-865486c9df-wjft6      1/1       Running   0
2h
```

Working with Services

You can see all the services running in kubernetes by going to Kubernetes engine>services


Google Cloud Platform
fourpointtwo
Navigation menu
Clusters
Workloads
Services
Applications
Configuration
Storage
Services
REFRESH
Kubernetes services
Brokered services
BETA
Services are sets of pods with a network endpoint that can be used for discovery and load balancing. Ingresses are collections of rules for routing external HTTP(S) traffic to services.
Is system object : False
Filter resources
Name
Status
Service Type
Endpoints
Pods
Namespace
Cluster
frontend
Ok
Load balancer
35.228.127.160:80
3 / 3
default
europe-cluster
nginx-1-nginx-svc
Ok
Load balancer
35.194.247.21:80
35.194.247.21:443
3 / 3
default
asia-east-cluster
redis-master
Ok
Cluster IP
10.27.240.58
1 / 1
default
europe-cluster
redis-slave
Ok
Cluster IP
10.27.241.209
2 / 2
default
europe-cluster

services