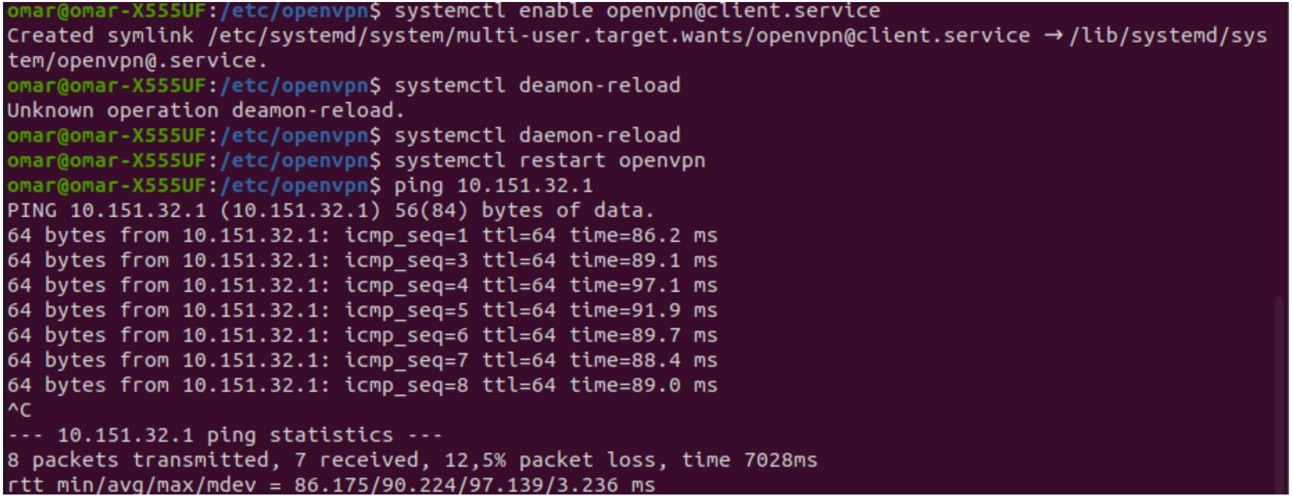
Rapport TP 4

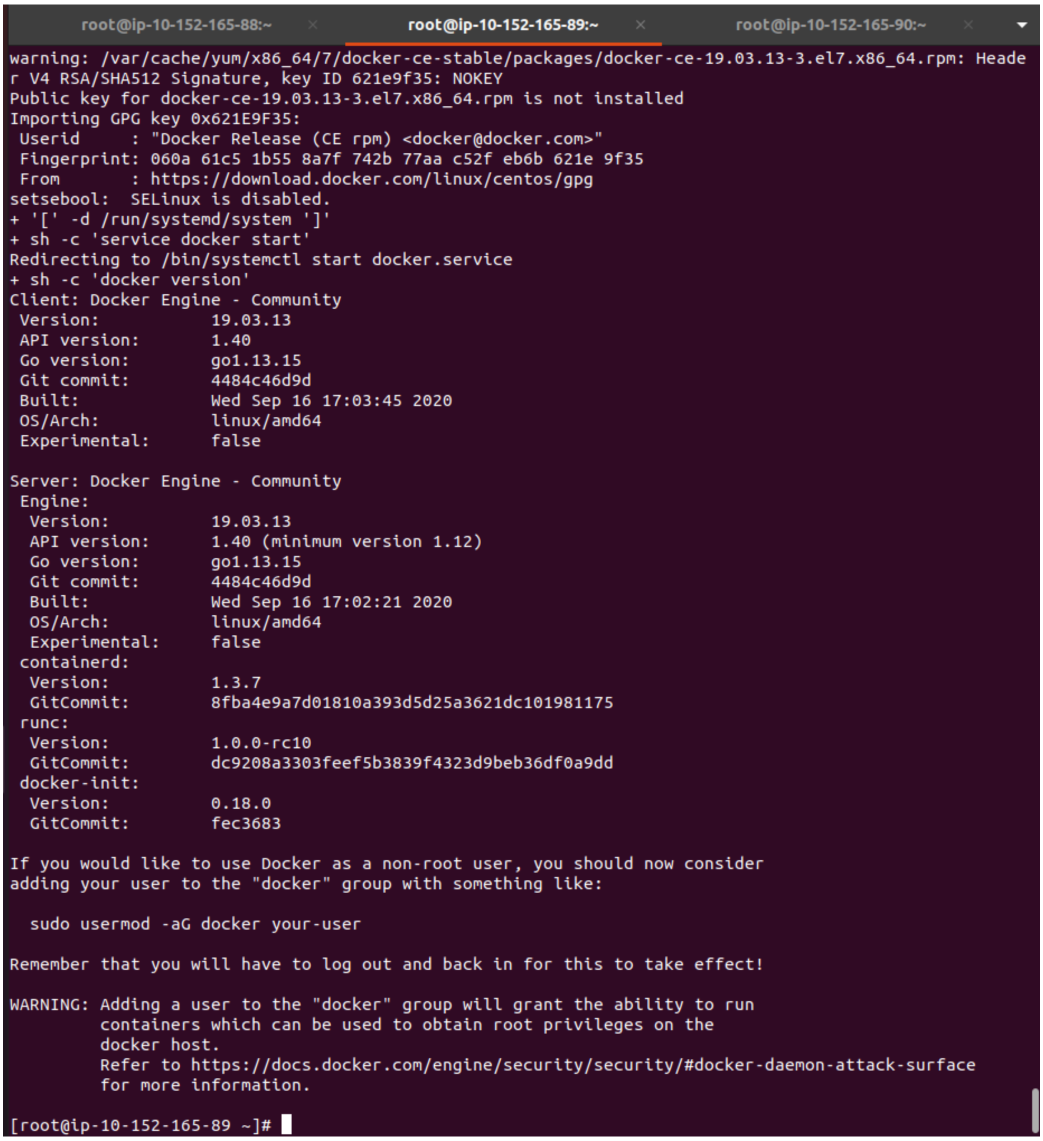
Omar Serrar – M1 ILSEN ALT

TP4

Connection au Datacenter Aws avec Openvpn.



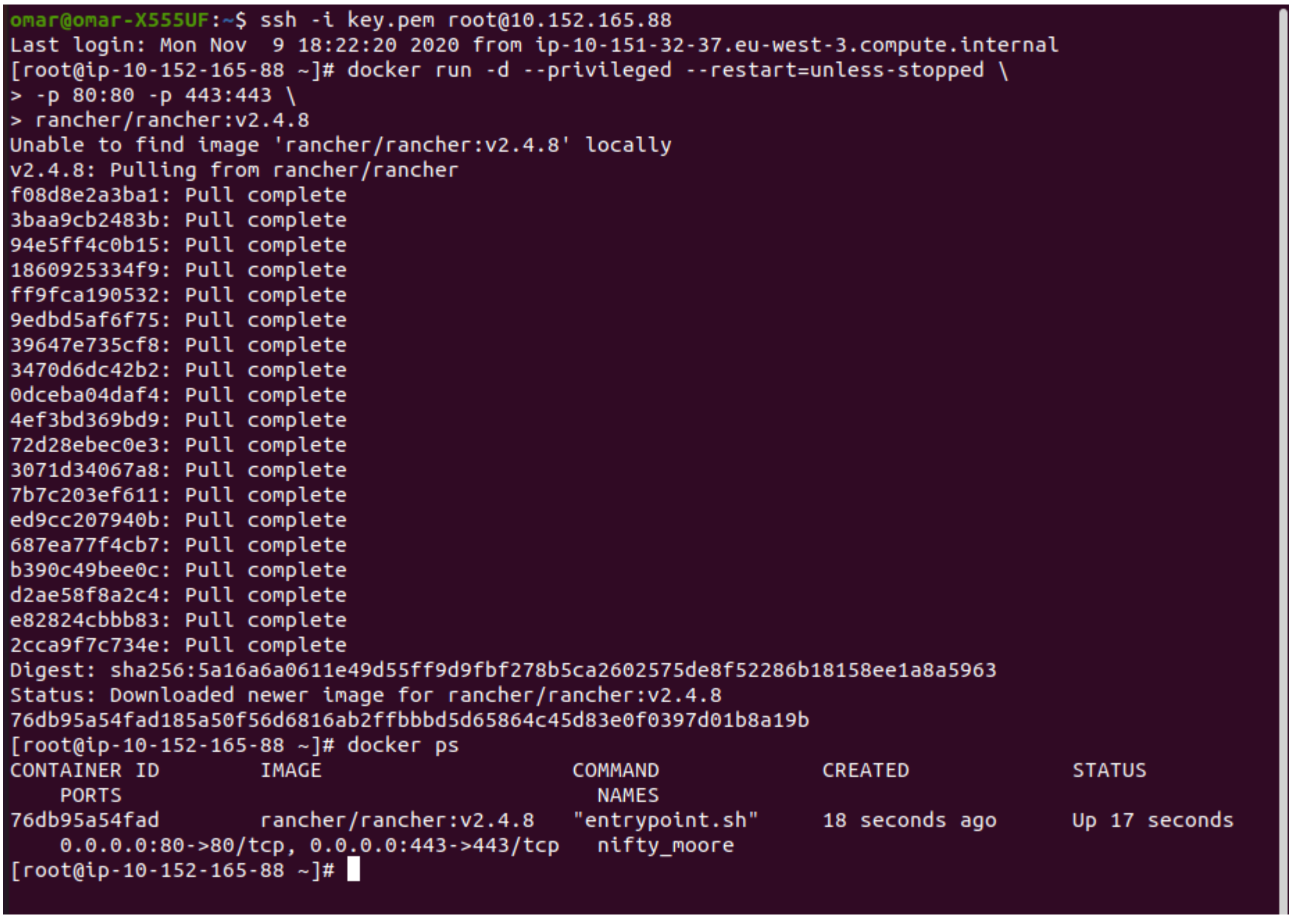
```
omar@omar-X555UF:/etc/openvpn$ systemctl enable openvpn@client.service
Created symlink /etc/systemd/system/multi-user.target.wants/openvpn@client.service → /lib/systemd/sys
tem/openvpn@.service.
omar@omar-X555UF:/etc/openvpn$ systemctl deamon-reload
Unknown operation deamon-reload.
omar@omar-X555UF:/etc/openvpn$ systemctl daemon-reload
omar@omar-X555UF:/etc/openvpn$ systemctl restart openvpn
omar@omar-X555UF:/etc/openvpn$ ping 10.151.32.1
PING 10.151.32.1 (10.151.32.1) 56(84) bytes of data.
64 bytes from 10.151.32.1: icmp_seq=1 ttl=64 time=86.2 ms
64 bytes from 10.151.32.1: icmp_seq=3 ttl=64 time=89.1 ms
64 bytes from 10.151.32.1: icmp_seq=4 ttl=64 time=97.1 ms
64 bytes from 10.151.32.1: icmp_seq=5 ttl=64 time=91.9 ms
64 bytes from 10.151.32.1: icmp_seq=6 ttl=64 time=89.7 ms
64 bytes from 10.151.32.1: icmp_seq=7 ttl=64 time=88.4 ms
64 bytes from 10.151.32.1: icmp_seq=8 ttl=64 time=89.0 ms
^C
--- 10.151.32.1 ping statistics ---
8 packets transmitted, 7 received, 12,5% packet loss, time 7028ms
rtt min/avg/max/mdev = 86.175/90.224/97.139/3.236 ms
```



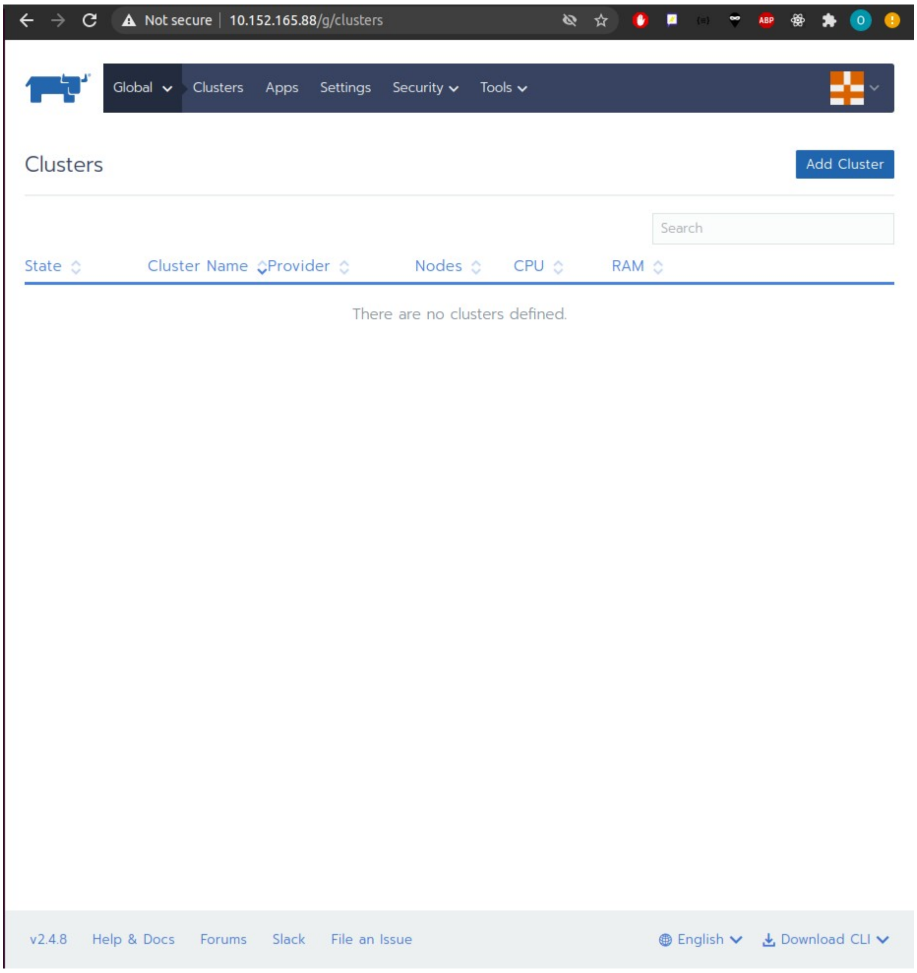
```
warning: /var/cache/yum/x86_64/7/docker-ce-stable/packages/docker-ce-19.03.13-3.el7.x86_64.rpm: Heade
r V4 RSA/SHA512 Signature, key ID 621e9f35: NOKEY
Public key for docker-ce-19.03.13-3.el7.x86_64.rpm is not installed
Importing GPG key 0x621E9F35:
 Userid     : "Docker Release (CE rpm) <docker@docker.com>"
 Fingerprint: 060a 61c5 1b55 8a7f 742b 77aa c52f eb6b 621e 9f35
 From       : https://download.docker.com/linux/centos/gpg
setsebool:  SELinux is disabled.
+ '[' -d /run/systemd/system ']'
+ sh -c 'service docker start'
Redirecting to /bin/systemctl start docker.service
+ sh -c 'docker version'
Client: Docker Engine - Community
 Version:           19.03.13
 API version:       1.40
 Go version:        go1.13.15
 Git commit:        4484c46d9d
 Built:             Wed Sep 16 17:03:45 2020
 OS/Arch:           linux/amd64
 Experimental:      false

Server: Docker Engine - Community
 Engine:
  Version:          19.03.13
  API version:      1.40 (minimum version 1.12)
  Go version:       go1.13.15
  Git commit:       4484c46d9d
  Built:            Wed Sep 16 17:02:21 2020
  OS/Arch:          linux/amd64
  Experimental:     false
 containerd:
  Version:          1.3.7
  GitCommit:        8fba4e9a7d01810a393d5d25a3621dc101981175
 runc:
  Version:          1.0.0-rc10
  GitCommit:        dc9208a3303feef5b3839f4323d9beb36df0a9dd
 docker-init:
  Version:          0.18.0
  GitCommit:        fec3683

If you would like to use Docker as a non-root user, you should now consider
adding your user to the "docker" group with something like:

  sudo usermod -aG docker your-user

Remember that you will have to log out and back in for this to take effect!

WARNING: Adding a user to the "docker" group will grant the ability to run
         containers which can be used to obtain root privileges on the
         docker host.
         Refer to https://docs.docker.com/engine/security/security/#docker-daemon-attack-surface
         for more information.

[root@ip-10-152-165-89 ~]# 
```

Installation de Rancher

Installation de Rancher sur la première machine



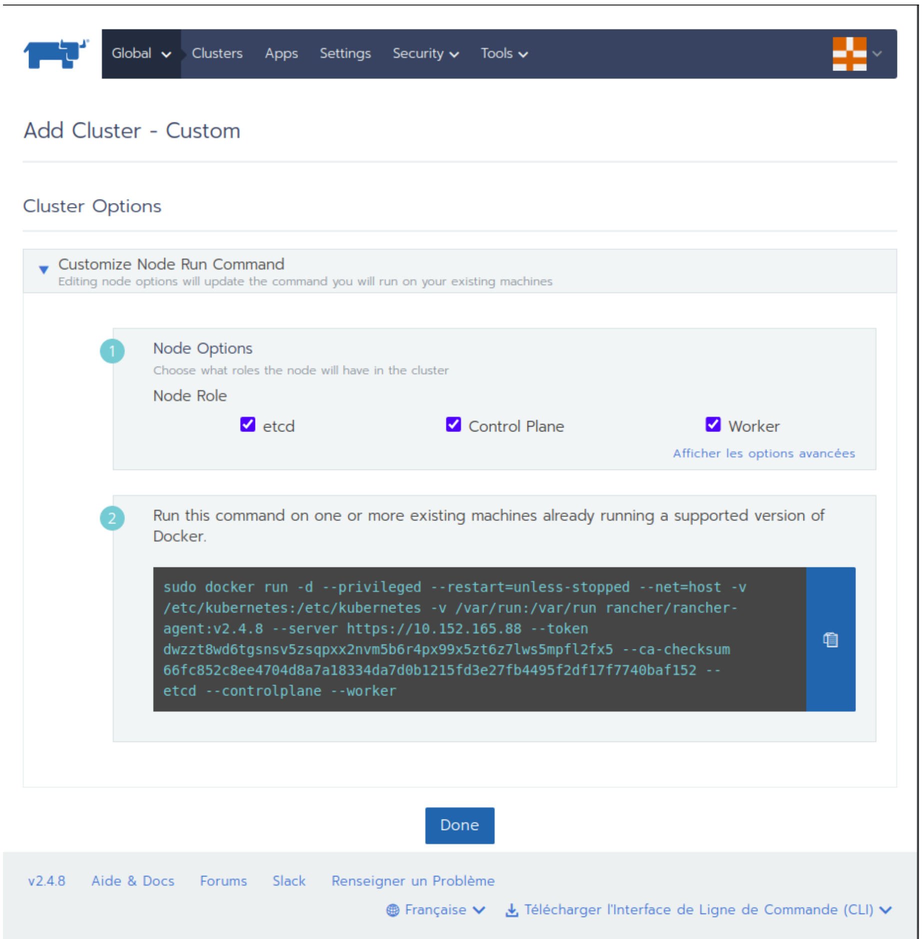
```
omar@omar-X555UF:~$ ssh -i key.pem root@10.152.165.88
Last login: Mon Nov  9 18:22:20 2020 from ip-10-151-32-37.eu-west-3.compute.internal
[root@ip-10-152-165-88 ~]# docker run -d --privileged --restart=unless-stopped \
> -p 80:80 -p 443:443 \
> rancher/rancher:v2.4.8
Unable to find image 'rancher/rancher:v2.4.8' locally
v2.4.8: Pulling from rancher/rancher
f08d8e2a3ba1: Pull complete
3baa9cb2483b: Pull complete
94e5ff4c0b15: Pull complete
1860925334f9: Pull complete
ff9fca190532: Pull complete
9edbd5af6f75: Pull complete
39647e735cf8: Pull complete
3470d6dc42b2: Pull complete
0dceba04daf4: Pull complete
4ef3bd369bd9: Pull complete
72d28ebec0e3: Pull complete
3071d34067a8: Pull complete
7b7c203ef611: Pull complete
ed9cc207940b: Pull complete
687ea77f4cb7: Pull complete
b390c49bee0c: Pull complete
d2ae58f8a2c4: Pull complete
e82824cbbb83: Pull complete
2cca9f7c734e: Pull complete
Digest: sha256:5a16a6a0611e49d55ff9d9fbf278b5ca2602575de8f52286b18158ee1a8a5963
Status: Downloaded newer image for rancher/rancher:v2.4.8
76db95a54fad185a50f56d6816ab2ffbbbd5d65864c45d83e0f0397d01b8a19b
[root@ip-10-152-165-88 ~]# docker ps
CONTAINER ID        IMAGE                    COMMAND             CREATED             STATUS
    PORTS                                      NAMES
76db95a54fad        rancher/rancher:v2.4.8   "entrypoint.sh"     18 seconds ago      Up 17 seconds
    0.0.0.0:80->80/tcp, 0.0.0.0:443->443/tcp   nifty_moore
[root@ip-10-152-165-88 ~]# 
```



Clusters

State	Cluster Name	Provider	Nodes	CPU	RAM

There are no clusters defined.



Add Cluster - Custom

Cluster Options

Customize Node Run Command
Editing node options will update the command you will run on your existing machines

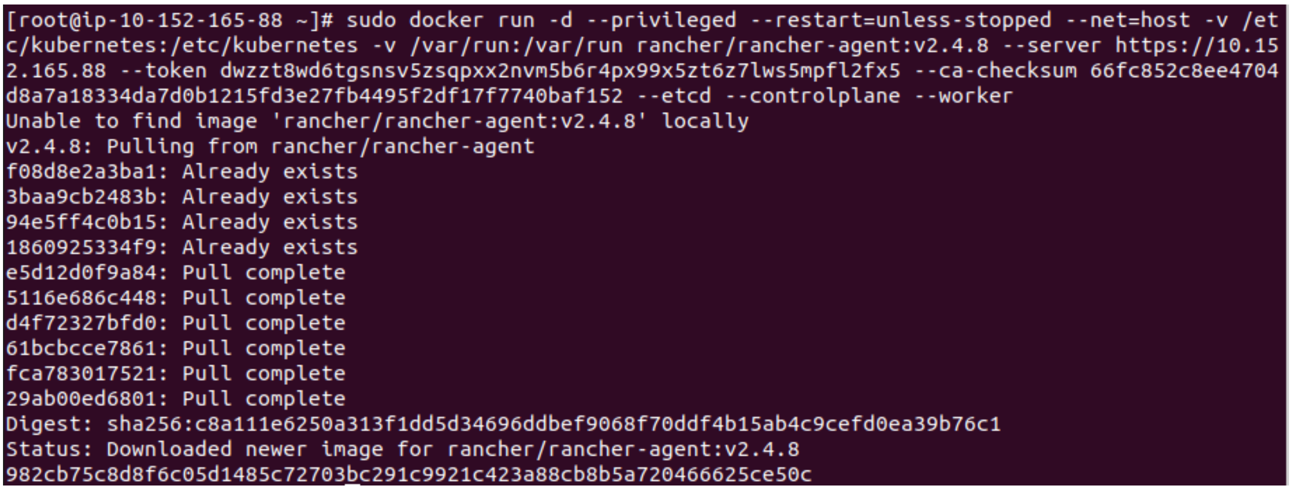
1 Node Options
Choose what roles the node will have in the cluster

Node Role

☑ etcd ☑ Control Plane ☑ Worker

2 Run this command on one or more existing machines already running a supported version of Docker.

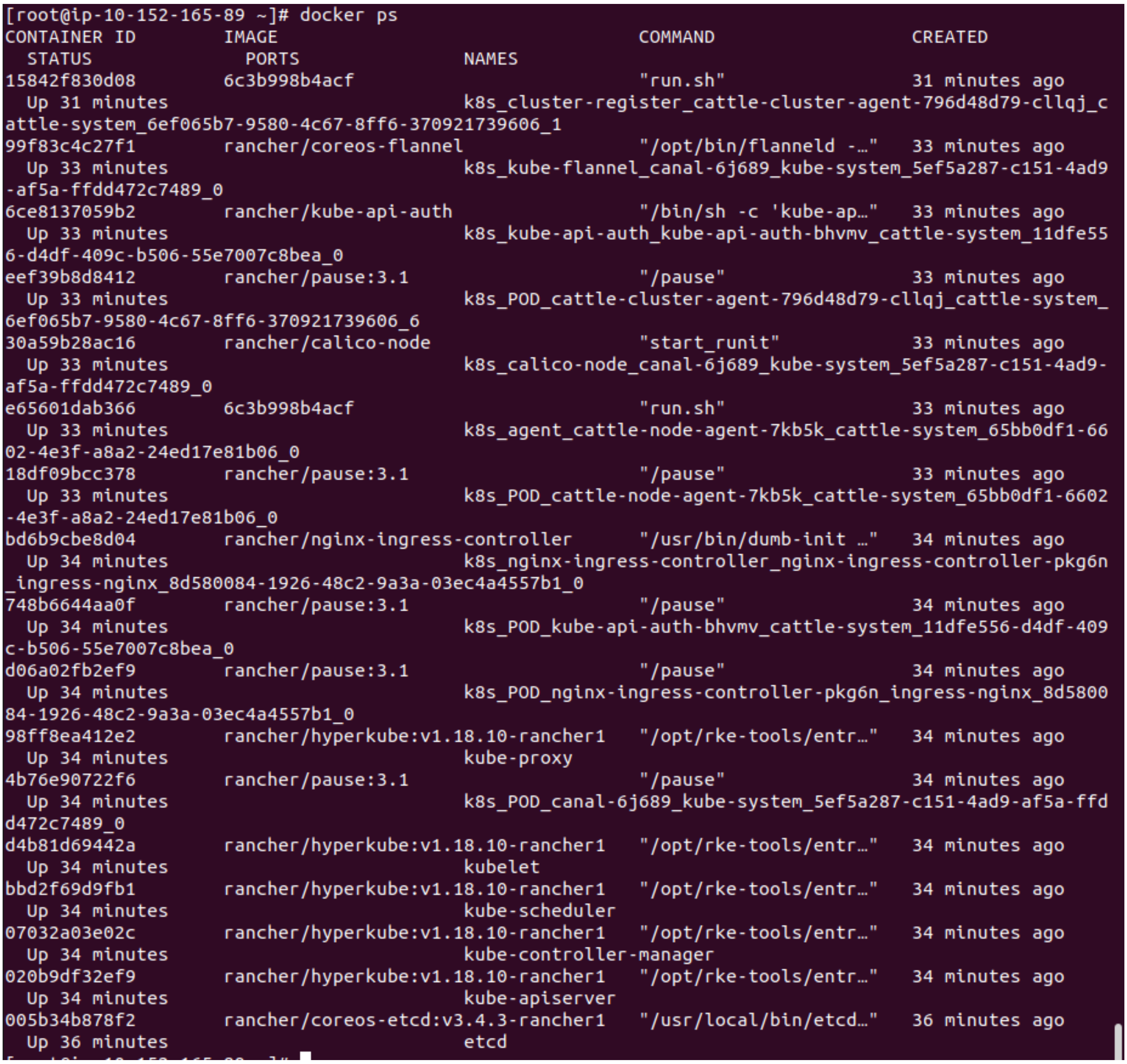
```
sudo docker run -d --privileged --restart=unless-stopped --net=host -v
/etc/kubernetes:/etc/kubernetes -v /var/run:/var/run rancher/rancher-
agent:v2.4.8 --server https://10.152.165.88 --token
dwzzt8wd6tgsnsv5zsqpxx2nvm5b6r4px99x5zt6z7lws5mpfl2fx5 --ca-checksum
66fc852c8ee4704d8a7a18334da7d0b1215fd3e27fb4495f2df17f7740baf152 --
etcd --controlplane --worker
```



```
[root@ip-10-152-165-88 ~]# sudo docker run -d --privileged --restart=unless-stopped --net=host -v /et
c/kubernetes:/etc/kubernetes -v /var/run:/var/run rancher/rancher-agent:v2.4.8 --server https://10.15
2.165.88 --token dwzzt8wd6tgsnsv5zsqpxx2nvm5b6r4px99x5zt6z7lws5mpfl2fx5 --ca-checksum 66fc852c8ee4704
d8a7a18334da7d0b1215fd3e27fb4495f2df17f7740baf152 --etcd --controlplane --worker
Unable to find image 'rancher/rancher-agent:v2.4.8' locally
v2.4.8: Pulling from rancher/rancher-agent
f08d8e2a3ba1: Already exists
3baa9cb2483b: Already exists
94e5ff4c0b15: Already exists
1860925334f9: Already exists
e5d12d0f9a84: Pull complete
5116e686c448: Pull complete
d4f72327bfd0: Pull complete
61bcbcce7861: Pull complete
fca783017521: Pull complete
29ab00ed6801: Pull complete
Digest: sha256:c8a111e6250a313f1dd5d34696ddbef9068f70ddf4b15ab4c9cefd0ea39b76c1
Status: Downloaded newer image for rancher/rancher-agent:v2.4.8
982cb75c8d8f6c05d1485c72703bc291c9921c423a88cb8b5a720466625ce50c
```

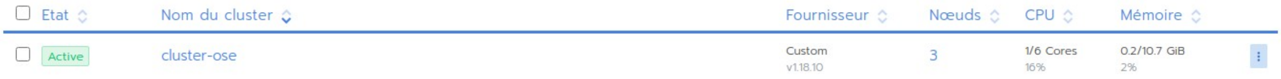
Pour verifier l’instalation de docker on peut taper la commande docker --version

1a)



```
[root@ip-10-152-165-89 ~]# docker ps
CONTAINER ID        IMAGE                                 COMMAND                  CREATED
  STATUS              PORTS               NAMES
15842f830d08        6c3b998b4acf                          "run.sh"                 31 minutes ago
  Up 31 minutes                           k8s_cluster-register_cattle-cluster-agent-796d48d79-cllqj_c
attle-system_6ef065b7-9580-4c67-8ff6-370921739606_1
99f83c4c27f1        rancher/coreos-flannel                "/opt/bin/flanneld -…"   33 minutes ago
  Up 33 minutes                           k8s_kube-flannel_canal-6j689_kube-system_5ef5a287-c151-4ad9
-af5a-ffdd472c7489_0
6ce8137059b2        rancher/kube-api-auth                 "/bin/sh -c 'kube-ap…"   33 minutes ago
  Up 33 minutes                           k8s_kube-api-auth_kube-api-auth-bhvmv_cattle-system_11dfe55
6-d4df-409c-b506-55e7007c8bea_0
eef39b8d8412        rancher/pause:3.1                     "/pause"                 33 minutes ago
  Up 33 minutes                           k8s_POD_cattle-cluster-agent-796d48d79-cllqj_cattle-system_
6ef065b7-9580-4c67-8ff6-370921739606_6
30a59b28ac16        rancher/calico-node                   "start_runit"            33 minutes ago
  Up 33 minutes                           k8s_calico-node_canal-6j689_kube-system_5ef5a287-c151-4ad9-
af5a-ffdd472c7489_0
e65601dab366        6c3b998b4acf                          "run.sh"                 33 minutes ago
  Up 33 minutes                           k8s_agent_cattle-node-agent-7kb5k_cattle-system_65bb0df1-66
02-4e3f-a8a2-24ed17e81b06_0
18df09bcc378        rancher/pause:3.1                     "/pause"                 33 minutes ago
  Up 33 minutes                           k8s_POD_cattle-node-agent-7kb5k_cattle-system_65bb0df1-6602
-4e3f-a8a2-24ed17e81b06_0
bd6b9cbe8d04        rancher/nginx-ingress-controller      "/usr/bin/dumb-init …"   34 minutes ago
  Up 34 minutes                           k8s_nginx-ingress-controller_nginx-ingress-controller-pkg6n
_ingress-nginx_8d580084-1926-48c2-9a3a-03ec4a4557b1_0
748b6644aa0f        rancher/pause:3.1                     "/pause"                 34 minutes ago
  Up 34 minutes                           k8s_POD_kube-api-auth-bhvmv_cattle-system_11dfe556-d4df-409
c-b506-55e7007c8bea_0
d06a02fb2ef9        rancher/pause:3.1                     "/pause"                 34 minutes ago
  Up 34 minutes                           k8s_POD_nginx-ingress-controller-pkg6n_ingress-nginx_8d5800
84-1926-48c2-9a3a-03ec4a4557b1_0
98ff8ea412e2        rancher/hyperkube:v1.18.10-rancher1   "/opt/rke-tools/entr…"   34 minutes ago
  Up 34 minutes                           kube-proxy
4b76e90722f6        rancher/pause:3.1                     "/pause"                 34 minutes ago
  Up 34 minutes                           k8s_POD_canal-6j689_kube-system_5ef5a287-c151-4ad9-af5a-ffd
d472c7489_0
d4b81d69442a        rancher/hyperkube:v1.18.10-rancher1   "/opt/rke-tools/entr…"   34 minutes ago
  Up 34 minutes                           kubelet
bbd2f69d9fb1        rancher/hyperkube:v1.18.10-rancher1   "/opt/rke-tools/entr…"   34 minutes ago
  Up 34 minutes                           kube-scheduler
07032a03e02c        rancher/hyperkube:v1.18.10-rancher1   "/opt/rke-tools/entr…"   34 minutes ago
  Up 34 minutes                           kube-controller-manager
020b9df32ef9        rancher/hyperkube:v1.18.10-rancher1   "/opt/rke-tools/entr…"   34 minutes ago
  Up 34 minutes                           kube-apiserver
005b34b878f2        rancher/coreos-etcd:v3.4.3-rancher1   "/usr/local/bin/etcd…"   36 minutes ago
  Up 36 minutes                           etcd
```

1b)



☐ Etat	Nom du cluster	Fournisseur	Nœuds	CPU	Mémoire
☐ Active	cluster-ose	Custom v1.18.10	3	1/6 Cores 16%	0.2/10.7 GiB 2%

1c)

Active

1d)

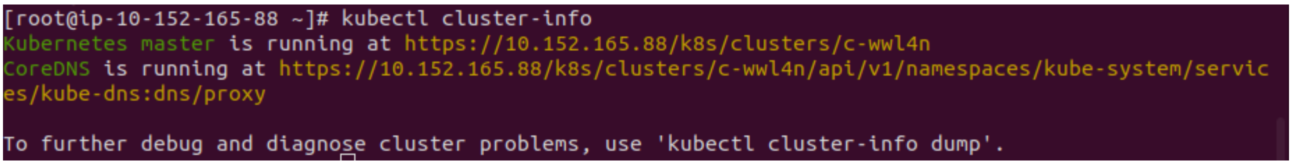
3

Installation de l'outil de ligne de commande Kubctl

2a)

chmod a+x kubectl

2b)

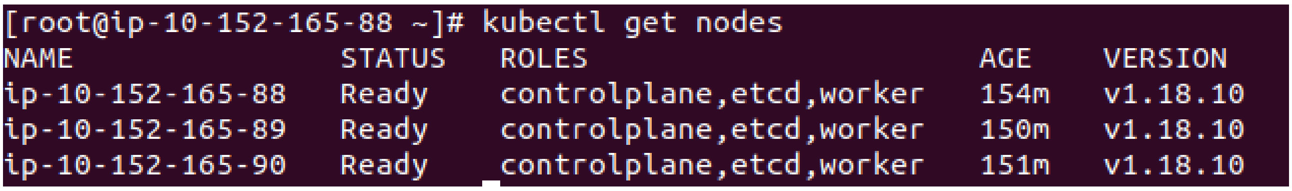


```
[root@ip-10-152-165-88 ~]# kubectl cluster-info
Kubernetes master is running at https://10.152.165.88/k8s/clusters/c-wwl4n
CoreDNS is running at https://10.152.165.88/k8s/clusters/c-wwl4n/api/v1/namespaces/kube-system/servic
es/kube-dns:dns/proxy

To further debug and diagnose cluster problems, use 'kubectl cluster-info dump'.
```

La commande affiche les services en cours et leurs adresse

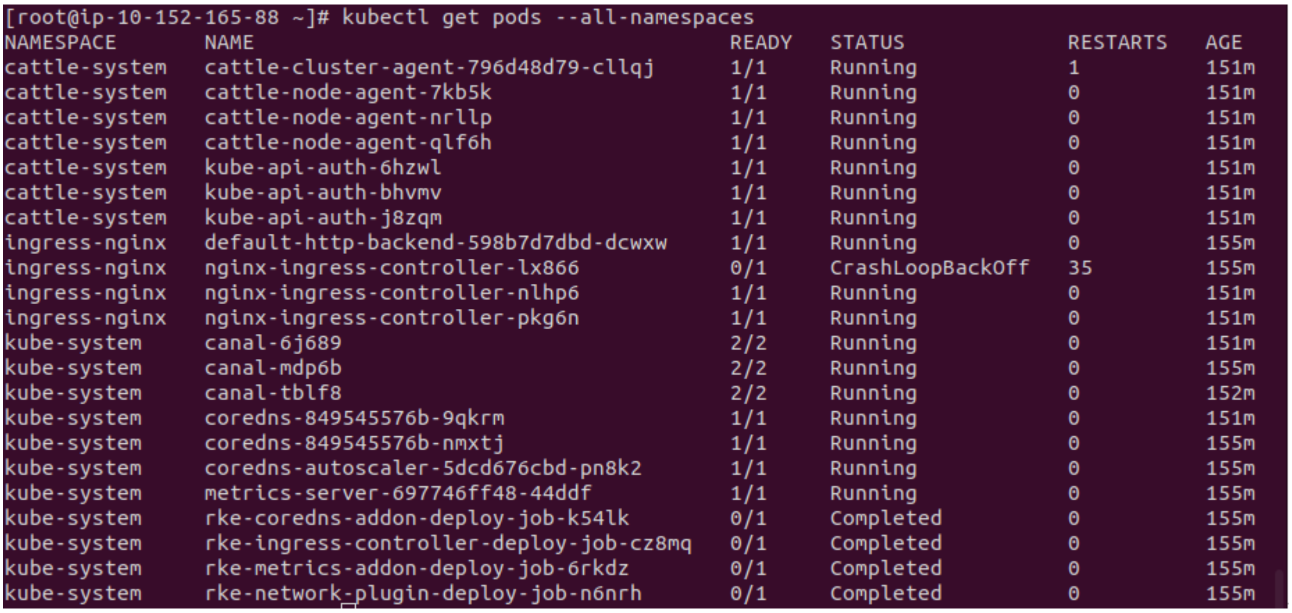
2c)

1- La commande affiche les noeuds



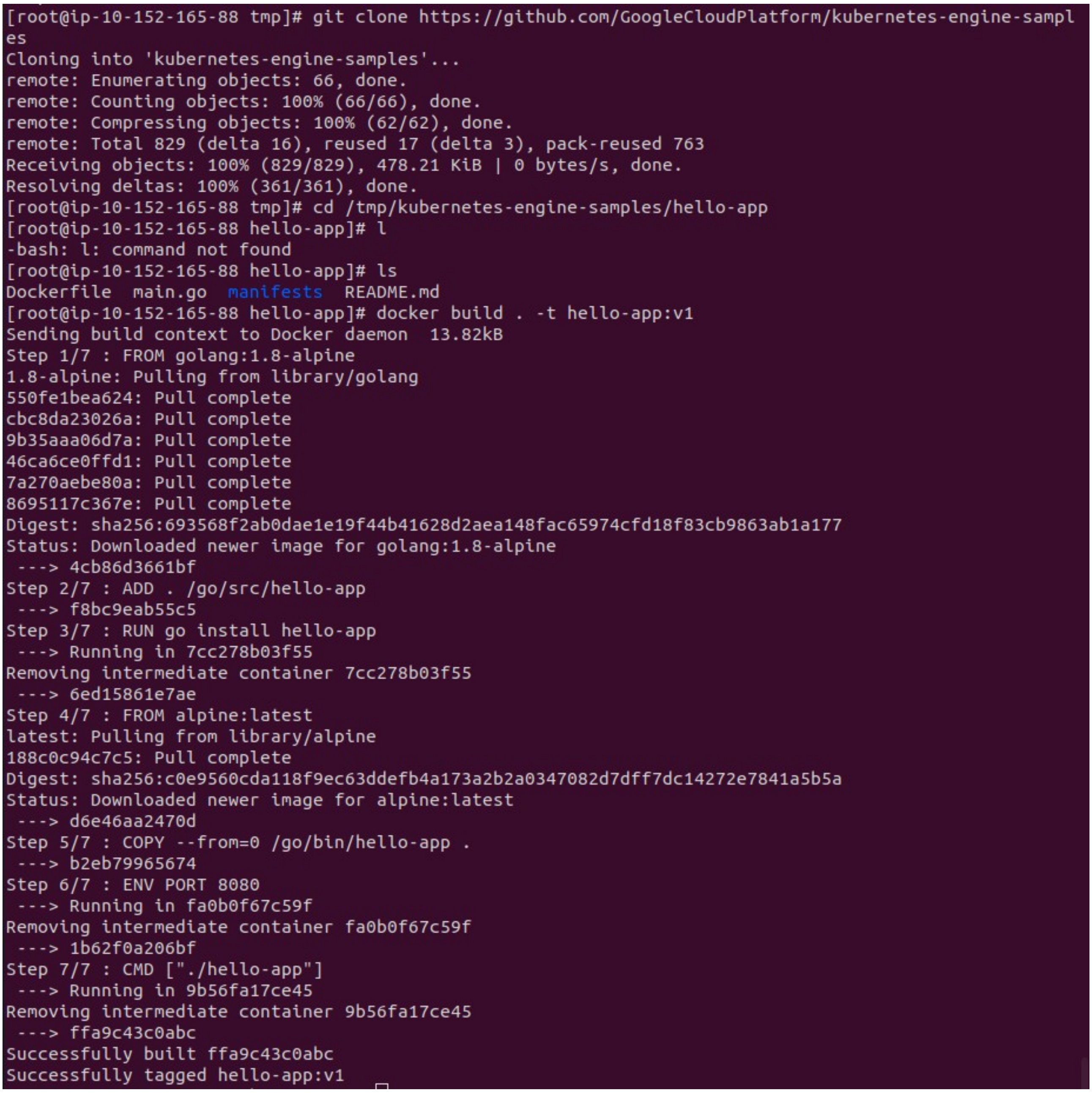
```
[root@ip-10-152-165-88 ~]# kubectl get nodes
NAME                STATUS   ROLES                      AGE    VERSION
ip-10-152-165-88    Ready    controlplane,etcd,worker   154m   v1.18.10
ip-10-152-165-89    Ready    controlplane,etcd,worker   150m   v1.18.10
ip-10-152-165-90    Ready    controlplane,etcd,worker   151m   v1.18.10
```

2- Affiche toutes les images des namespace



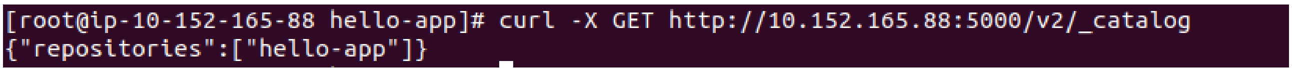
```
[root@ip-10-152-165-88 ~]# kubectl get pods --all-namespaces
NAMESPACE       NAME                                      READY   STATUS             RESTARTS   AGE
cattle-system   cattle-cluster-agent-796d48d79-cllqj      1/1     Running            1          151m
cattle-system   cattle-node-agent-7kb5k                   1/1     Running            0          151m
cattle-system   cattle-node-agent-nrllp                   1/1     Running            0          151m
cattle-system   cattle-node-agent-qlf6h                   1/1     Running            0          151m
cattle-system   kube-api-auth-6hzwl                       1/1     Running            0          151m
cattle-system   kube-api-auth-bhvmv                       1/1     Running            0          151m
cattle-system   kube-api-auth-j8zqm                       1/1     Running            0          151m
ingress-nginx   default-http-backend-598b7d7dbd-dcwxw     1/1     Running            0          155m
ingress-nginx   nginx-ingress-controller-lx866            0/1     CrashLoopBackOff   35         155m
ingress-nginx   nginx-ingress-controller-nlhp6            1/1     Running            0          151m
ingress-nginx   nginx-ingress-controller-pkg6n            1/1     Running            0          151m
kube-system     canal-6j689                               2/2     Running            0          151m
kube-system     canal-mdp6b                               2/2     Running            0          155m
kube-system     canal-tblf8                               2/2     Running            0          152m
kube-system     coredns-849545576b-9qkrm                  1/1     Running            0          151m
kube-system     coredns-849545576b-nmxtj                  1/1     Running            0          155m
kube-system     coredns-autoscaler-5dcd676cbd-pn8k2       1/1     Running            0          155m
kube-system     metrics-server-697746ff48-44ddf           1/1     Running            0          155m
kube-system     rke-coredns-addon-deploy-job-k54lk        0/1     Completed          0          155m
kube-system     rke-ingress-controller-deploy-job-cz8mq   0/1     Completed          0          155m
kube-system     rke-metrics-addon-deploy-job-6rkdz        0/1     Completed          0          155m
kube-system     rke-network-plugin-deploy-job-n6nrh       0/1     Completed          0          155m
```

Récupérez une image docker



```
[root@ip-10-152-165-88 tmp]# git clone https://github.com/GoogleCloudPlatform/kubernetes-engine-sampl
es
Cloning into 'kubernetes-engine-samples'...
remote: Enumerating objects: 66, done.
remote: Counting objects: 100% (66/66), done.
remote: Compressing objects: 100% (62/62), done.
remote: Total 829 (delta 16), reused 17 (delta 3), pack-reused 763
Receiving objects: 100% (829/829), 478.21 KiB | 0 bytes/s, done.
Resolving deltas: 100% (361/361), done.
[root@ip-10-152-165-88 tmp]# cd /tmp/kubernetes-engine-samples/hello-app
[root@ip-10-152-165-88 hello-app]# l
-bash: l: command not found
[root@ip-10-152-165-88 hello-app]# ls
Dockerfile  main.go  manifests  README.md
[root@ip-10-152-165-88 hello-app]# docker build . -t hello-app:v1
Sending build context to Docker daemon  13.82kB
Step 1/7 : FROM golang:1.8-alpine
1.8-alpine: Pulling from library/golang
550fe1bea624: Pull complete
cbc8da23026a: Pull complete
9b35aaa06d7a: Pull complete
46ca6ce0ffd1: Pull complete
7a270aebe80a: Pull complete
8695117c367e: Pull complete
Digest: sha256:693568f2ab0dae1e19f44b41628d2aea148fac65974cfd18f83cb9863ab1a177
Status: Downloaded newer image for golang:1.8-alpine
 ---> 4cb86d3661bf
Step 2/7 : ADD . /go/src/hello-app
 ---> f8bc9eab55c5
Step 3/7 : RUN go install hello-app
 ---> Running in 7cc278b03f55
Removing intermediate container 7cc278b03f55
 ---> 6ed15861e7ae
Step 4/7 : FROM alpine:latest
latest: Pulling from library/alpine
188c0c94c7c5: Pull complete
Digest: sha256:c0e9560cda118f9ec63ddefb4a173a2b2a0347082d7dff7dc14272e7841a5b5a
Status: Downloaded newer image for alpine:latest
 ---> d6e46aa2470d
Step 5/7 : COPY --from=0 /go/bin/hello-app .
 ---> b2eb79965674
Step 6/7 : ENV PORT 8080
 ---> Running in fa0b0f67c59f
Removing intermediate container fa0b0f67c59f
 ---> 1b62f0a206bf
Step 7/7 : CMD ["./hello-app"]
 ---> Running in 9b56fa17ce45
Removing intermediate container 9b56fa17ce45
 ---> ffa9c43c0abc
Successfully built ffa9c43c0abc
Successfully tagged hello-app:v1
```

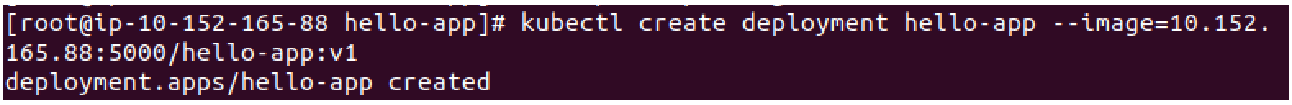
3a)



```
[root@ip-10-152-165-88 hello-app]# curl -X GET http://10.152.165.88:5000/v2/_catalog
{"repositories":["hello-app"]}
```

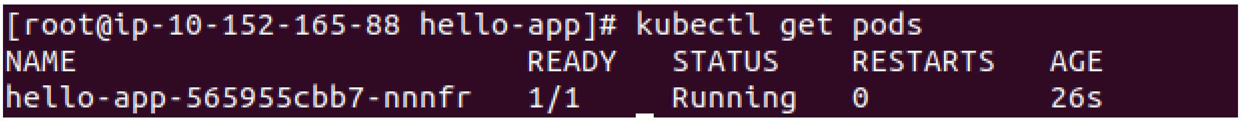
3b)

La commande permet de créer un déploiement pour l’image



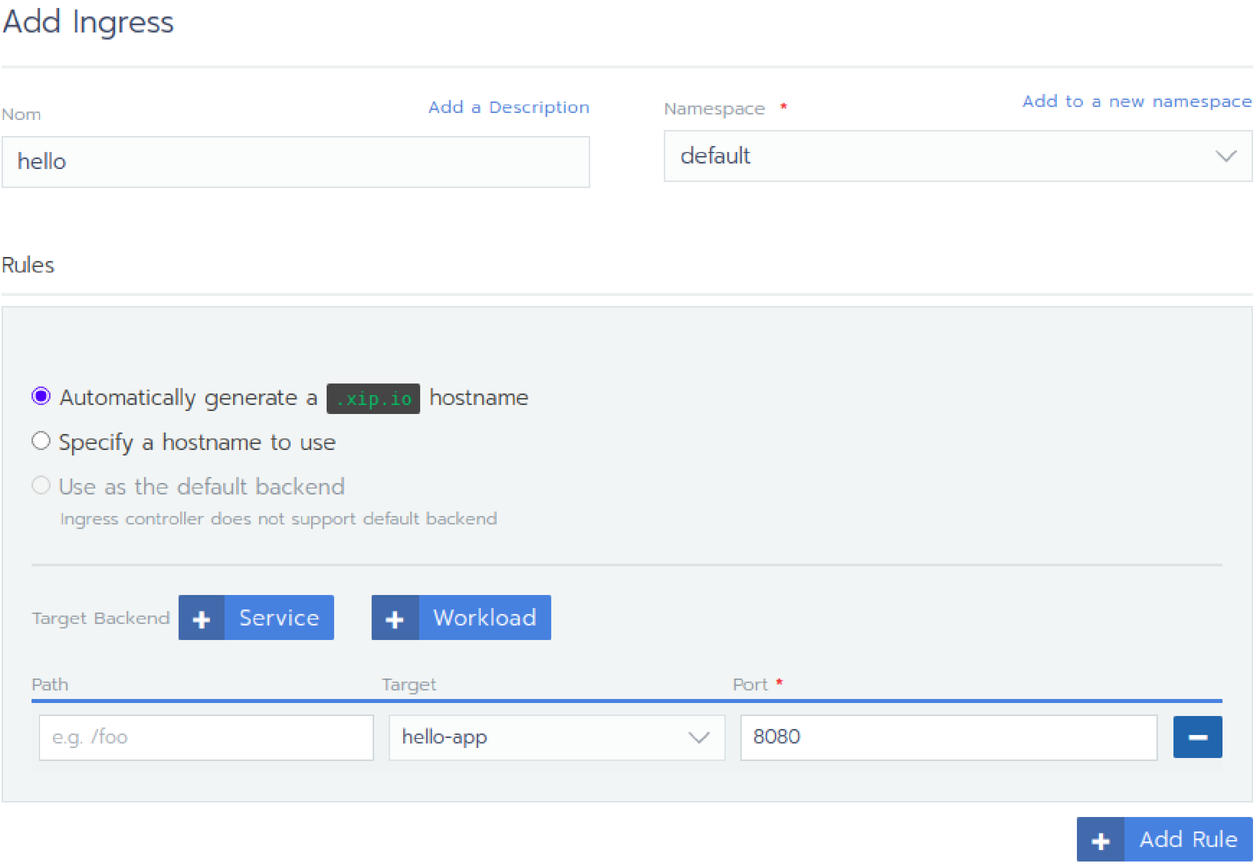
```
[root@ip-10-152-165-88 hello-app]# kubectl create deployment hello-app --image=10.152.
165.88:5000/hello-app:v1
deployment.apps/hello-app created
```

3c)



```
[root@ip-10-152-165-88 hello-app]# kubectl get pods
NAME                         READY   STATUS    RESTARTS   AGE
hello-app-565955cbb7-nnnfr   1/1     Running   0          26s
```

3d)



Add Ingress

Nom: hello

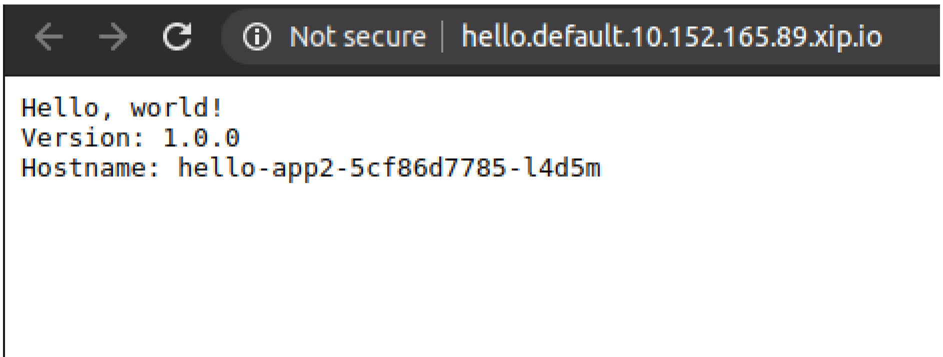
Namespace *: default

Rules

● Automatically generate a .xip.io hostname
○ Specify a hostname to use
○ Use as the default backend
Ingress controller does not support default backend

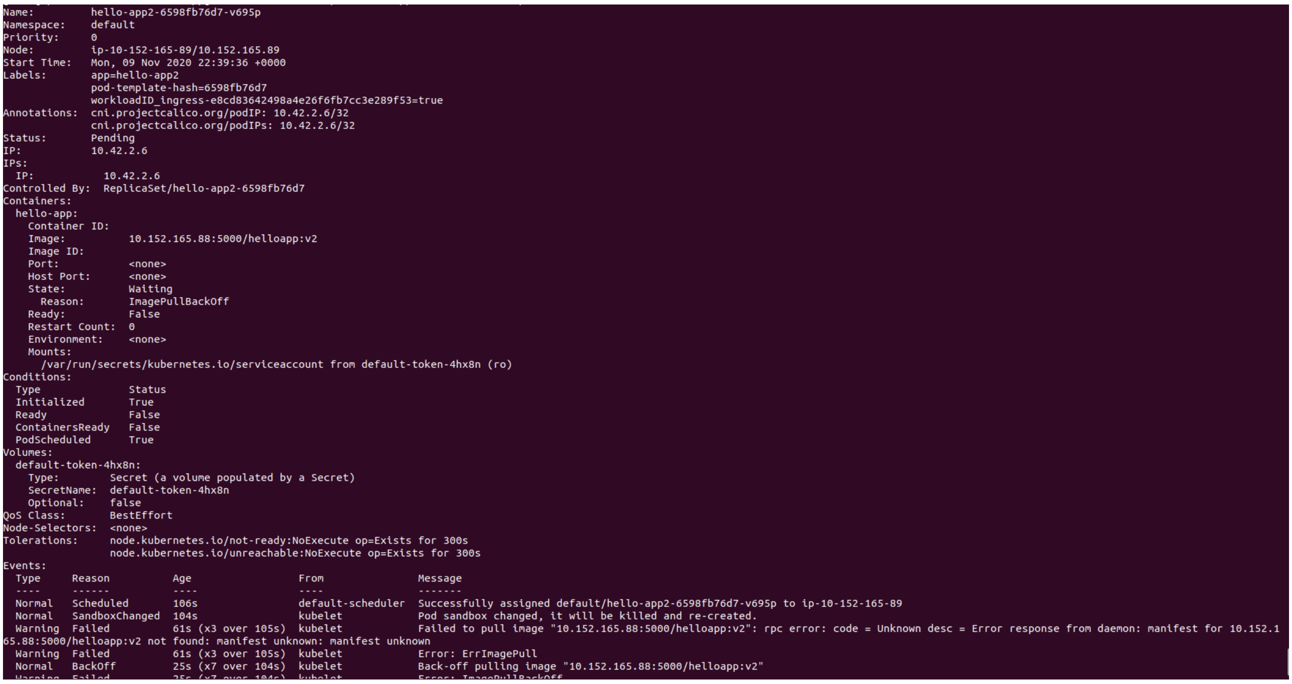
Path	Target	Port *
e.g. /foo	hello-app	8080

capture d’écran de http://hello.default.10.152.165.89.xip.io/ :



```
Hello, world!
Version: 1.0.0
Hostname: hello-app2-5cf86d7785-l4d5m
```

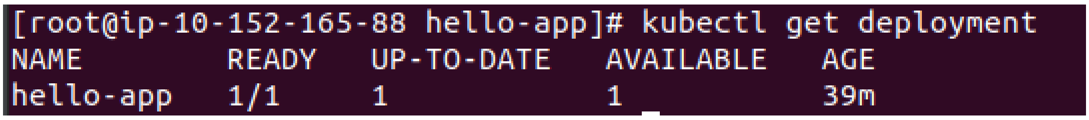
Mise à jour de l'application



```
Name:         hello-app2-6598fb76d7-v695p
Namespace:    default
Priority:     0
Node:         ip-10-152-165-89/10.152.165.89
Start Time:   Mon, 09 Nov 2020 22:39:36 +0000
Labels:       app=hello-app2
              pod-template-hash=6598fb76d7
              workloadID_ingress-e8cd83642498a4e26f6fb7cc3e289f53=true
Annotations:  cni.projectcalico.org/podIP: 10.42.2.6/32
              cni.projectcalico.org/podIPs: 10.42.2.6/32
Status:       Pending
IP:           10.42.2.6
IPs:
  IP:           10.42.2.6
Controlled By:  ReplicaSet/hello-app2-6598fb76d7
Containers:
  hello-app:
    Container ID:
    Image:          10.152.165.88:5000/helloapp:v2
    Image ID:
    Port:           <none>
    Host Port:      <none>
    State:          Waiting
      Reason:       ImagePullBackOff
    Ready:          False
    Restart Count:  0
    Environment:    <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from default-token-4hx8n (ro)
Conditions:
  Type              Status
  Initialized       True
  Ready             False
  ContainersReady   False
  PodScheduled      True
Volumes:
  default-token-4hx8n:
    Type:        Secret (a volume populated by a Secret)
    SecretName:  default-token-4hx8n
    Optional:    false
QoS Class:       BestEffort
Node-Selectors:  <none>
Tolerations:     node.kubernetes.io/not-ready:NoExecute op=Exists for 300s
                 node.kubernetes.io/unreachable:NoExecute op=Exists for 300s
Events:
  Type     Reason          Age                 From               Message
  ----     ------          ----                ----               -------
  Normal   Scheduled       106s                default-scheduler  Successfully assigned default/hello-app2-6598fb76d7-v695p to ip-10-152-165-89
  Normal   SandboxChanged  104s                kubelet            Pod sandbox changed, it will be killed and re-created.
  Warning  Failed          61s (x3 over 105s)  kubelet            Failed to pull image "10.152.165.88:5000/helloapp:v2": rpc error: code = Unknown desc = Error response from daemon: manifest for 10.152.1
65.88:5000/helloapp:v2 not found: manifest unknown: manifest unknown
  Warning  Failed          61s (x3 over 105s)  kubelet            Error: ErrImagePull
  Normal   BackOff         25s (x7 over 104s)  kubelet            Back-off pulling image "10.152.165.88:5000/helloapp:v2"
  Warning  Failed          25s (x7 over 104s)  kubelet            Error: ImagePullBackOff
```

Scalling des containers Haute dispo

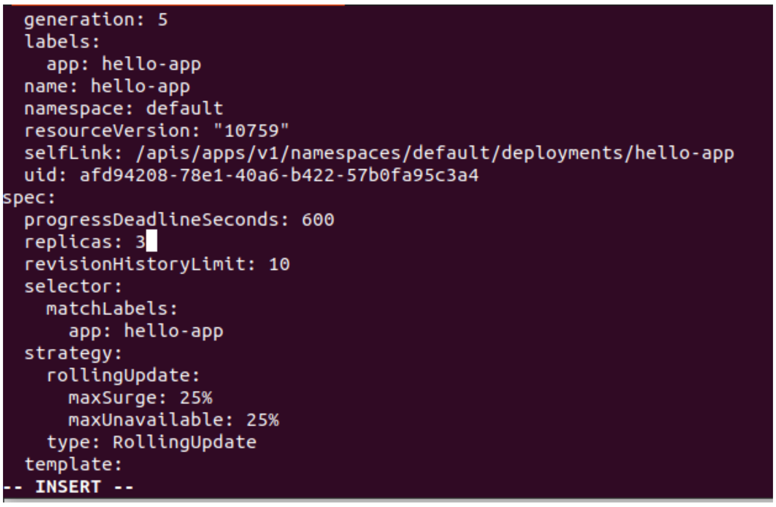
5a)



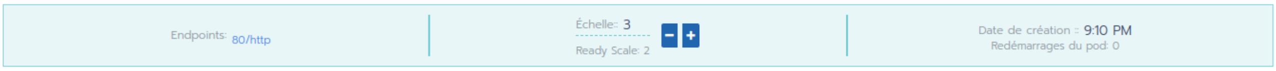
```
[root@ip-10-152-165-88 hello-app]# kubectl get deployment
NAME        READY   UP-TO-DATE   AVAILABLE   AGE
hello-app   1/1     1            1           39m
```

On voit la liste des deployments

5b)



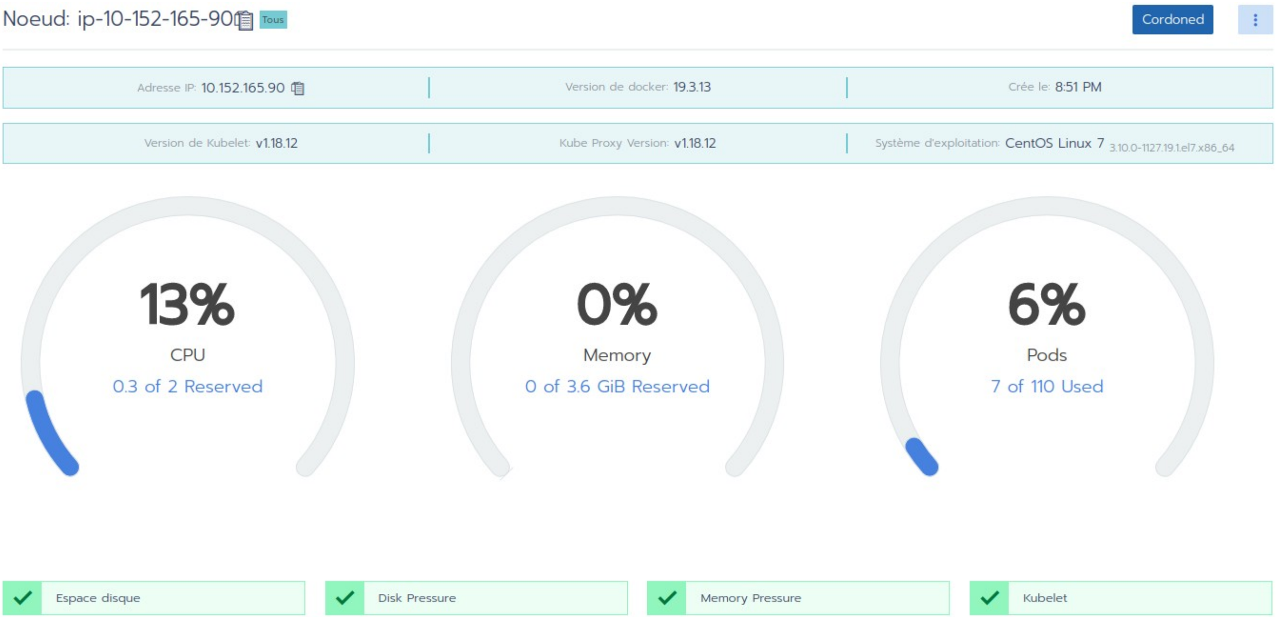
```
  generation: 5
  labels:
    app: hello-app
  name: hello-app
  namespace: default
  resourceVersion: "10759"
  selfLink: /apis/apps/v1/namespaces/default/deployments/hello-app
  uid: afd94208-78e1-40a6-b422-57b0fa95c3a4
spec:
  progressDeadlineSeconds: 600
  replicas: 3
  revisionHistoryLimit: 10
  selector:
    matchLabels:
      app: hello-app
  strategy:
    rollingUpdate:
      maxSurge: 25%
      maxUnavailable: 25%
    type: RollingUpdate
  template:
-- INSERT --
```



Endpoints: 80/http | Échelle: 3 Ready Scale: 2 | Date de création : 9:10 PM Redémarrages du pod: 0

Je constate que le hostname change

Cordon



Noeud: ip-10-152-165-90 Tous

Adresse IP: 10.152.165.90 | Version de docker: 19.3.13 | Crée le: 8:51 PM

Version de Kubelet: v1.18.12 | Kube Proxy Version: v1.18.12 | Système d'exploitation: CentOS Linux 7 3.10.0-1127.19.1.el7.x86_64

13% CPU 0.3 of 2 Reserved

0% Memory 0 of 3.6 GiB Reserved

6% Pods 7 of 110 Used

✓ Espace disque ✓ Disk Pressure ✓ Memory Pressure ✓ Kubelet

6a) On voit les ressources utilisé par le nœud

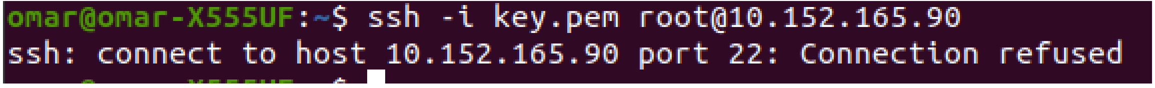
6b) 7 pods tournent sur le nœud

Après l’augmentation des réplicas à 6 le nombre de pods sur le nœud n’ont pas changé, c’est normal parce que le nœud ne peut pas ajouter des pods si il est cordonné.

Drain

7a) Drain va enlever tous les pods du nœud et va les deplacer dans les autres nœuds.

7b) L'acces au nœud par ssh est refusé :

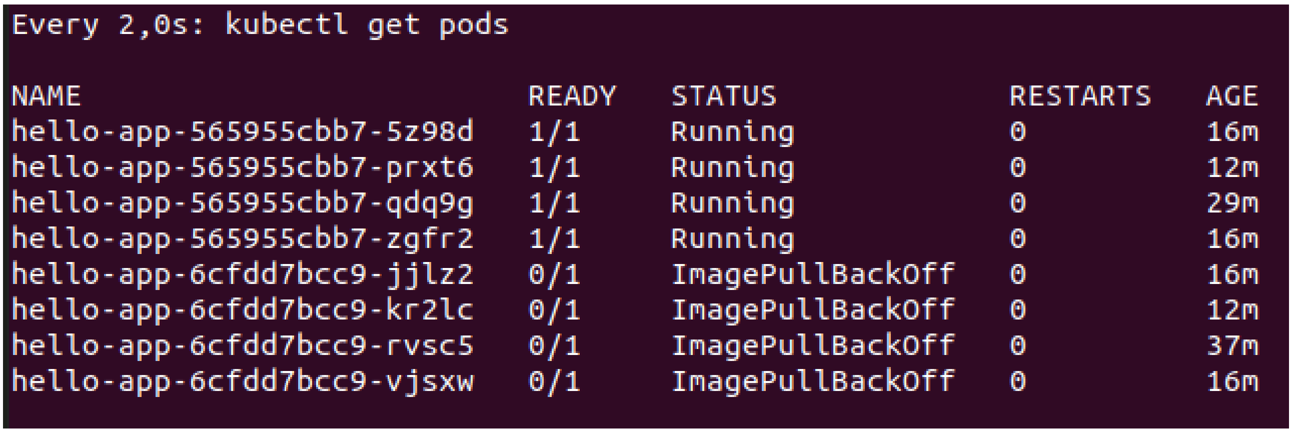


```
omar@omar-X555UF:~$ ssh -i key.pem root@10.152.165.90
ssh: connect to host 10.152.165.90 port 22: Connection refused
```

7c) L'état du cluster est **active**

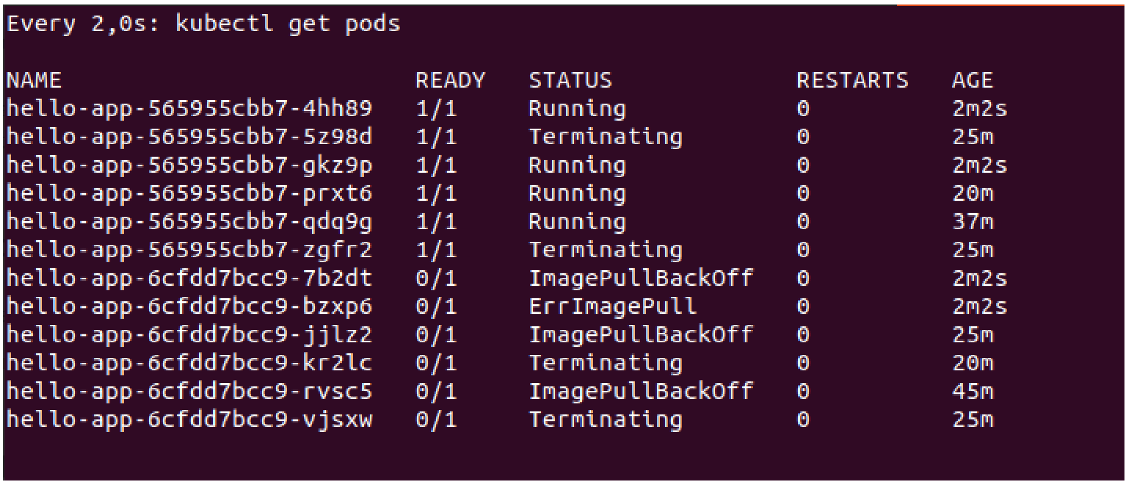
Rupture de service involontaire

Le nombre de pods pour l'application hello-world est **8**



```
Every 2,0s: kubectl get pods

NAME                         READY   STATUS             RESTARTS   AGE
hello-app-565955cbb7-5z98d   1/1     Running            0          16m
hello-app-565955cbb7-prxt6   1/1     Running            0          12m
hello-app-565955cbb7-qdq9g   1/1     Running            0          29m
hello-app-565955cbb7-zgfr2   1/1     Running            0          16m
hello-app-6cfdd7bcc9-jjlz2   0/1     ImagePullBackOff   0          16m
hello-app-6cfdd7bcc9-kr2lc   0/1     ImagePullBackOff   0          12m
hello-app-6cfdd7bcc9-rvsc5   0/1     ImagePullBackOff   0          37m
hello-app-6cfdd7bcc9-vjsxw   0/1     ImagePullBackOff   0          16m
```

8a) Quelques minutes après le reboot le nombre de pods est devenue **12**



```
Every 2,0s: kubectl get pods

NAME                         READY   STATUS             RESTARTS   AGE
hello-app-565955cbb7-4hh89   1/1     Running            0          2m2s
hello-app-565955cbb7-5z98d   1/1     Terminating        0          25m
hello-app-565955cbb7-gkz9p   1/1     Running            0          2m2s
hello-app-565955cbb7-prxt6   1/1     Running            0          20m
hello-app-565955cbb7-qdq9g   1/1     Running            0          37m
hello-app-565955cbb7-zgfr2   1/1     Terminating        0          25m
hello-app-6cfdd7bcc9-7b2dt   0/1     ImagePullBackOff   0          2m2s
hello-app-6cfdd7bcc9-bzxp6   0/1     ErrImagePull       0          2m2s
hello-app-6cfdd7bcc9-jjlz2   0/1     ImagePullBackOff   0          25m
hello-app-6cfdd7bcc9-kr2lc   0/1     Terminating        0          20m
hello-app-6cfdd7bcc9-rvsc5   0/1     ImagePullBackOff   0          45m
hello-app-6cfdd7bcc9-vjsxw   0/1     Terminating        0          25m
```

L'état du cluster est **active** l'état de l'application est **updating**

8b) Le nombre de pods maintenant est **8**

8c) L'interet de cette solution est d'eviter la rupture de l'application si un nœud est en panne.

Note : Je n’ai pas pus déployer la version 2 de hello-world sur le cluster, j’ai suivi les étapes mais ça m’affiche une erreur. C’est pour ça que l’état des pods de cette version est **ImagePullBackOff**