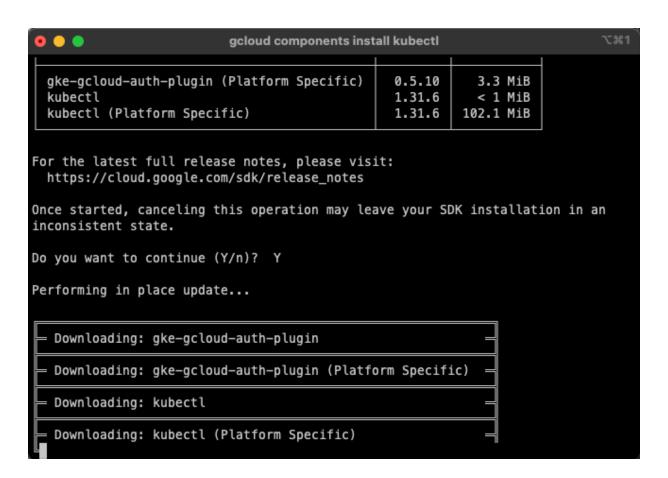# ✅ Labo 8 Kubernetes - 04/05/2025

⚠️ Voorbereiding cloud en Docker

Hiervoor heb ik weer Google Cloud SDK nodig. Maar deze heb ik ondertussen al van vorige labo.

Eerst ga ik **Kubernetes installeren**:

◯ gcloud components install kubectl

```
> gcloud init

Welcome! This command will take you through the configuration of gcloud.

Settings from your current configuration [default] are:
core:
  account: younes.elazzouzi@student.kdg.be
  disable_usage_reporting: 'False'
  project: cs2-elazzouzi-younes

Pick configuration to use:
 [1] Re-initialize this configuration [default] with new settings
 [2] Create a new configuration
Please enter your numeric choice:  []
```

Nu ga ik een **cluster aanmaken**:

○ gcloud init

○ gcloud container clusters create cs2-cluster \

○   --num-nodes=1 \

○   --release-channel=rapid \

○   --enable-ip-alias \

○   --zone=europe-west1-b


```
 > ~/Dow/google-cloud-sdk · · · · · CS2-ElAzzouzi-Younes G < 06:32:16 PM ⊘
> gcloud container clusters create cs2-cluster \
  --num-nodes=1 \
  --release-channel=rapid \
  --enable-ip-alias \
  --zone=europe-west1-b

Note: The Kubelet readonly port (10255) is now deprecated. Please update your wo
rkloads to use the recommended alternatives. See https://cloud.google.com/kubern
etes-engine/docs/how-to/disable-kubelet-readonly-port for ways to check usage an
d for migration instructions.
Creating cluster cs2-cluster in europe-west1-b... Cluster is being configured..
. .:
Creating cluster cs2-cluster in europe-west1-b...:
```

NAME	LOCATION	MASTER_VERSION	MASTER_IP	MACHINE_TYPE	NODE_VERSION	NUM_NODES	STATUS
cs2-cluster	europe-west1-b	1.32.2-gke.1297002	34.22.237.250	e2-medium	1.32.2-gke.1297002	1	RUNNING

🚨 (ik kreeg de melding disable the buelet read-only port error)
Deze los je gewoon op door de API te enablen.

Kubernetes Engine API

Google Enterprise API

Builds and manages container-based applications, powered by the open source Kubernetes technology.

Try this API ☑

○ gcloud container clusters get-credentials cs2-cluster \
○   --zone=europe-west1-b
○ kubectl get nodes


```
 > gcloud container clusters get-credentials cs2-cluster \
   --zone=europe-west1-b

Fetching cluster endpoint and auth data.
kubeconfig entry generated for cs2-cluster.

 > kubectl get nodes
NAME                                          STATUS   ROLES    AGE   VERSION
gke-cs2-cluster-default-pool-4d512634-sq8b    Ready    <none>   74s   v1.32.2-gke.1297002
```

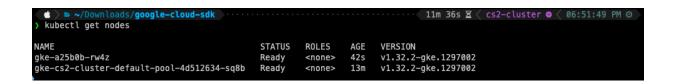## ⚠️ Windows node toevoegen

Dat doe ik door de volgende commando's uit te voeren:
○ gcloud container node-pools create windows-pool \
○   --cluster=cs2-cluster \
○   --image-type=WINDOWS_LTSC_CONTAINERD \
○   --machine-type=e2-standard-2 \
○   --num-nodes=1 \
○   --zone=europe-west1-b

```
gcloud container node-pools create windows-pool \
--cluster=cs2-cluster \
--image-type=WINDOWS_LTSC_CONTAINERD \
--machine-type=e2-standard-2 \
--num-nodes=1 \
--zone=europe-west1-b
```

NAME	MACHINE_TYPE	DISK_SIZE_GB	NODE_VERSION
windows-pool	e2-standard-2	100	1.32.2-gke.1297002
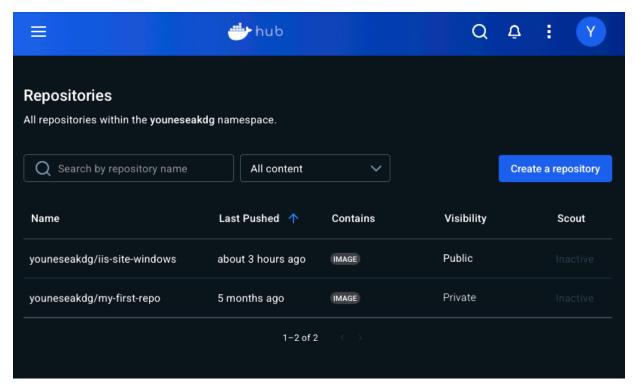
Wil ik checken of de node efectief is uitgevoerd?

◯ kubectl get nodes

```
kubectl get nodes
```

NAME	STATUS	ROLES	AGE	VERSION
gke-a25b0b-rw4z	Ready	<none>	42s	v1.32.2-gke.1297002
gke-cs2-cluster-default-pool-4d512634-sq8b	Ready	<none>	13m	v1.32.2-gke.1297002

⚠️ **Windows container deployen**

Hiervoor maak ik een **.yaml file** aan en ga ik die dan deployen.
Handig om te weten is hoe mijn docker repo heet.

Want ik gebruik die in het .yaml file.

```
> cat iis-site-windows.yaml

       File: iis-site-windows.yaml

    1    apiVersion: apps/v1
    2    kind: Deployment
    3    metadata:
    4      name: iis-deployment
    5    spec:
    6      replicas: 1
    7      selector:
    8        matchLabels:
    9          app: iis
   10      template:
   11        metadata:
   12          labels:
   13            app: iis
   14        spec:
   15          nodeSelector:
   16            "kubernetes.io/os": windows
   17          containers:
   18          - name: iis
   19            image: younseakdg/iis-site-windows
   20            ports:
   21            - containerPort: 80
   22
   23    apiVersion: v1
   24    kind: Service
   25    metadata:
   26      name: iis-service
   27    spec:
   28      selector:
   29        app: iis
   30      ports:
   31        - protocol: TCP
   32          port: 80
   33          targetPort: 80
   34      type: LoadBalancer
   35
```

Deployen van de file:

○ kubectl apply -f iis-site-windows.yaml

○ kubectl get pods

```
> kubectl apply -f iis-site-windows.yaml

deployment.apps/iis-deployment created
```

```
> kubectl get pods
NAME                             READY   STATUS            RESTARTS   AGE
iis-deployment-65687866fd-4qchk  0/1     InvalidImageName  0          14s
```

⚠️ **Service aanmaken voor externen**

Ik voer het volgende commando uit:

○ kubectl expose deployment iis-deployment \

○ --type=LoadBalancer \

○ --name=iis-service \

○ --port=80

```
 ⌘  ▸  ~/Downloads/google-cloud-sdk ................................. cs2-cluster ◉  06:54:54 PM ⊘
❯ kubectl expose deployment iis-deployment \
--type=LoadBalancer \
--name=iis-service \
--port=80

service/iis-service exposed
```

Nu wil ik het IP ophalen:

◯ kubectl get service iis-service

```
 ⌘  ▸  ~/Downloads/google-cloud-sdk ................................. cs2-cluster ◉  06:55:28 PM ⊘
❯ kubectl get service iis-service

NAME          TYPE           CLUSTER-IP       EXTERNAL-IP    PORT(S)        AGE
iis-service   LoadBalancer   34.118.230.162   <pending>      80:30652/TCP   38s
```

⚠️ Service toevoegen aan Yaml

Ik breidt de .yaml file wat verder uit:

```
 ⌘  ▸  ~/Downloads/google-cloud-sdk ...................................... 24s ⏳  07:15:05 PM ⊘
❯ cat iis-site-windows.yaml

    File: iis-site-windows.yaml

 1   apiVersion: apps/v1
 2   kind: Deployment
 3   metadata:
 4     name: iis-deployment
 5   spec:
 6     replicas: 1
 7     selector:
 8       matchLabels:
 9         app: iis
10     template:
11       metadata:
12         labels:
13           app: iis
14       spec:
15         nodeSelector:
16           "kubernetes.io/os": windows
17         containers:
18         - name: iis
19           image: younseakdg/iis-site-windows
20           ports:
21           - containerPort: 80
22
23   apiVersion: v1
24   kind: Service
25   metadata:
26     name: iis-service
27   spec:
28     selector:
29       app: iis
30     ports:
31       - protocol: TCP
32         port: 80
33         targetPort: 80
34     type: LoadBalancer
35
```

Nu ga ik de nieuwe yaml toepassen:

◯ kubectl delete -f iis-site-windows.yaml
◯ kubectl apply -f iis-site-windows.yaml

```
 ⌘  📁 ~/Downloads/google-cloud-sdk ······················· cs2-cluster ● 06:59:09 PM ⊙
› kubectl delete -f iis-site-windows.yaml

service "iis-service" deleted

 ⌘  📁 ~/Downloads/google-cloud-sdk ·············· 31s ⧗ cs2-cluster ● 07:00:00 PM ⊙
› kubectl apply -f iis-site-windows.yaml
service/iis-service created
```

⚠️ Ingress gebruiken

Is een manier om HTTPS verkeer van buitenaf naar mijn cluster door te sturen op basis van IP. Werkt als reverse proxy als NGINX bv.

Eerst ga ik NGINX pullen:

⚪ kubectl apply -f https://raw.githubusercontent.com/kubernetes/ingress-nginx/controller-v1.8.1/deploy/static/provider/cloud/deploy.yaml

Eens die is binnegehaald:

⚪ kubectl get pods -n ingress-nginx


```
 ⌘  📁 ~/Downloads/google-cloud-sdk ······················· cs2-cluster ● 07:00:09 PM ⊙
› kubectl apply -f https://raw.githubusercontent.com/kubernetes/ingress-nginx/controller-v1.8.1/deploy/static/provider/cl
ud/deploy.yaml

namespace/ingress-nginx created
serviceaccount/ingress-nginx created
serviceaccount/ingress-nginx-admission created
role.rbac.authorization.k8s.io/ingress-nginx created
role.rbac.authorization.k8s.io/ingress-nginx-admission created
clusterrole.rbac.authorization.k8s.io/ingress-nginx created
clusterrole.rbac.authorization.k8s.io/ingress-nginx-admission created
rolebinding.rbac.authorization.k8s.io/ingress-nginx created
rolebinding.rbac.authorization.k8s.io/ingress-nginx-admission created
clusterrolebinding.rbac.authorization.k8s.io/ingress-nginx created
clusterrolebinding.rbac.authorization.k8s.io/ingress-nginx-admission created
configmap/ingress-nginx-controller created
service/ingress-nginx-controller created
service/ingress-nginx-controller-admission created
deployment.apps/ingress-nginx-controller created
job.batch/ingress-nginx-admission-create created
job.batch/ingress-nginx-admission-patch created
ingressclass.networking.k8s.io/nginx created
validatingwebhookconfiguration.admissionregistration.k8s.io/ingress-nginx-admission created

 ⌘  📁 ~/Downloads/google-cloud-sdk ·············· 5s ⧗ cs2-cluster ● 07:03:28 PM ⊙
› kubectl get pods -n ingress-nginx

NAME                                        READY   STATUS      RESTARTS   AGE
ingress-nginx-admission-create-65qfr        0/1     Completed   0          20s
ingress-nginx-admission-patch-rqxsl         0/1     Completed   1          20s
ingress-nginx-controller-66cb9865b5-5xd5v   0/1     Running     0          21s
```

Dan maak ik een andere yaml file waar ik die ingress define:

```
  ~/Downloads/google-cloud-sdk                                    07:05:19 PM
> touch iis-ingress.yaml

  ~/Downloads/google-cloud-sdk                                    07:05:40 PM
> vi iis-ingress.yaml

  ~/Downloads/google-cloud-sdk                              6s    07:05:51 PM
> cat iis-ingress.yaml

        File: iis-ingress.yaml

    1   apiVersion: networking.k8s.io/v1
    2   kind: Ingress
    3   metadata:
    4     name: iis-ingress
    5     annotations:
    6       nginx.ingress.kubernetes.io/rewrite-target: /
    7   spec:
    8     rules:
    9     - http:
   10         paths:
   11         - path: /
   12           pathType: Prefix
   13           backend:
   14             service:
   15               name: iis-service
   16               port:
   17                 number: 80
   18

  ~/Downloads/google-cloud-sdk                                    07:05:57 PM
>
```

```
  ~/Downloads/google-cloud-sdk                                    07:09:45 PM
> cat iis-ingress.yaml

        File: iis-ingress.yaml

    1   apiVersion: networking.k8s.io/v1
    2   kind: Ingress
    3   metadata:
    4     name: iis-ingress
    5     annotations:
    6       nginx.ingress.kubernetes.io/rewrite-target: /
    7       nginx.ingress.kubernetes.io/use-regex: "true"
    8   spec:
    9     rules:
   10     - http:
   11         paths:
   12         - path: /?(.*)
   13           pathType: Prefix
   14           backend:
   15             service:
   16               name: iis-service
   17               port:
   18                 number: 80
   19
```

Deployen van de ingrress en testen:
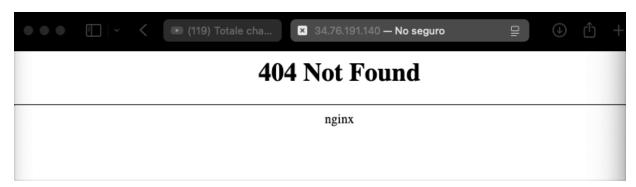
◯ kubectl apply –f iis-ingress.yaml

◯ kubectl get ingress

```
  ~/Downloads/google-cloud-sdk                                    cs2-cluster ● 07:05:57 PM ○
 kubectl apply -f iis-ingress.yaml

ingress.networking.k8s.io/iis-ingress created

  ~/Downloads/google-cloud-sdk                                    cs2-cluster ● 07:06:21 PM ○
 kubectl get ingress

NAME          CLASS    HOSTS   ADDRESS   PORTS   AGE
iis-ingress   <none>   *                 80      7s

  ~/Downloads/google-cloud-sdk                                                07:06:28 PM ○
```

Nu kan ik het IP-adres gaan ophalen:

○ kubectl get svc -n ingress-nginx


```
  ~/Downloads/google-cloud-sdk                                    cs2-cluster ● 07:07:09 PM ○
 kubectl get svc -n ingress-nginx

NAME                                TYPE           CLUSTER-IP       EXTERNAL-IP     PORT(S)                      AGE
ingress-nginx-controller            LoadBalancer   34.118.234.223   34.76.191.140   80:30752/TCP,443:30983/TCP   4m12s
ingress-nginx-controller-admission  ClusterIP      34.118.227.133   <none>          443/TCP                      4m11s
```

Bij het opzoeken naar de IP adres kreeg ik 404. Pods is niet running dus even aanpassen.
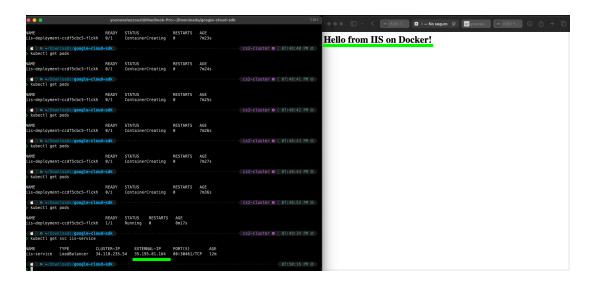
```
> kubectl describe pod iis-deployment-65687866fd-4qchk
Name:              iis-deployment-65687866fd-4qchk
Namespace:         default
Priority:          0
Service Account:   default
Node:              gke-a25b0b-rw4z/10.132.0.13
Start Time:        Sat, 03 May 2025 18:54:40 +0200
Labels:            app=iis
                   pod-template-hash=65687866fd
Annotations:       <none>
Status:            Pending
IP:                10.36.1.224
IPs:
  IP:              10.36.1.224
Controlled By:  ReplicaSet/iis-deployment-65687866fd
Containers:
  iis:
    Container ID:
    Image:         YOUR_DOCKER_HUB_USERNAME/YOUR_IMAGE_NAME:TAG
    Image ID:
    Port:          80/TCP
    Host Port:     0/TCP
    State:         Waiting
      Reason:      InvalidImageName
    Ready:         False
    Restart Count: 0
    Environment:   <none>
apiVersion: apps/v1
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-dhqwt (ro)
Conditions:
  Type                       Status
  PodReadyToStartContainers  True
  Initialized                True
  Ready                      False
  ContainersReady            False
  PodScheduled               True
Volumes:
  kube-api-access-dhqwt:
    Type:                    Projected (a volume that contains injected data from multiple sources)
    TokenExpirationSeconds:  3607
    ConfigMapName:           kube-root-ca.crt
    ConfigMapOptional:       <nil>
    DownwardAPI:             true
QoS Class:                   BestEffort
Node-Selectors:              kubernetes.io/os=windows
Tolerations:                 node.kubernetes.io/not-ready:NoExecute op=Exists for 300s
```



```
> kubectl apply -f iis-site-windows.yaml
deployment.apps/iis-deployment created
service/iis-service unchanged

> kubectl get pods
NAME                            READY   STATUS             RESTARTS   AGE
iis-deployment-ccdf5cbc5-flckh  0/1     ContainerCreating  0          7s
```

Dit kwam omdat ik de default image gebruikte.
Nu even wachten terwijl ie build. Eens die **Running** is kan ik weer ernaar surfen.


```
> kubectl get pods
NAME                            READY   STATUS    RESTARTS   AGE
iis-deployment-ccdf5cbc5-flckh  1/1     Running   0          8m17s
```

En als ik nu surf naar het ip-adres:

Hello from IIS on Docker!

⚠️ Start/Stop K8s clusters met PWSH

```
> cat startk8.ps1

File: startk8.ps1

1   param (
2       [int]$LinuxNodes = 1,
3       [int]$WindowsNodes = 1,
4       [string]$ClusterName = "cs2-cluster",
5       [string]$Yaml = ""
6   )
7
8   if ($LinuxNodes -lt 1) {
9       Write-Host "Fout: Je moet minstens 1 Linux node hebben!" -ForegroundColor Red
10      exit 1
11  }
12
13  gcloud container clusters create $ClusterName `
14      --num-nodes=$LinuxNodes `
15      --release-channel=rapid `
16      --enable-ip-alias `
17      --zone=europe-west1-b
18
19  if ($WindowsNodes -ge 1) {
20      gcloud container node-pools create windows-pool `
21          --cluster=$ClusterName `
22          --image-type=WINDOWS_LTSC_CONTAINERD `
23          --machine-type=e2-standard-2 `
24          --num-nodes=$WindowsNodes `
25          --zone=europe-west1-b
26  }
27
28  gcloud container clusters get-credentials $ClusterName `
29      --zone=europe-west1-b
30
31  if ($Yaml -ne "") {
32      kubectl apply -f $Yaml
33  }
34
```

```
File: stopk8.ps1

1  param (
2      [string]$ClusterName = "cs2-cluster"
3  )
4
5  gcloud container clusters delete $ClusterName `
6      --zone=europe-west1-b `
7      --quiet
8
```

Hierboven 2 scripts 1tje voor het aanmaken van een cluster en 1tje voor het verwijderen ervan.


```
younes_elazzouzi@cloudshell:~ (cs2-elazzouzi-younes)$ pwsh
PowerShell 7.4.6

A new PowerShell stable release is available: v7.5.1
Upgrade now, or check out the release page at:
  https://aka.ms/PowerShell-Release?tag=v7.5.1

PS>./startk8.ps1
Note: The Kubelet readonly port (10255) is now deprecated. Please update your workloads to use th
e recommended alternatives. See https://cloud.google.com/kubernetes-engine/docs/how-to/disable-ku
belet-readonly-port for ways to check usage and for migration instructions.
ERROR: (gcloud.container.clusters.create) ResponseError: code=409, message=Already exists: projec
ts/cs2-elazzouzi-younes/zones/europe-west1-b/clusters/cs2-cluster.
ERROR: (gcloud.container.node-pools.create) ResponseError: code=409, message=Already exists: proj
ects/cs2-elazzouzi-younes/zones/europe-west1-b/clusters/cs2-cluster/nodePools/windows-pool.
Fetching cluster endpoint and auth data.
kubeconfig entry generated for cs2-cluster.
PS>./stopk8.ps1
Deleting cluster cs2-cluster...working.
```

Testing en het werkt!