

Lab3 Notes/Findings

Step 1: Install kubectl and minikube:

```
kacper@UNKNOWN:~$ curl -LO "https://dl.k8s.io/release/$(curl -Ls https://dl.k8s.io/release/stable.txt)/bin/linux/amd64/kubectl"
chmod +x kubectl && sudo mv kubectl /usr/local/bin/
kubectl version --client
% Total % Received % Xferd Average Speed Time Time Time Current
Dload Upload Total Spent Left Speed
100 138 100 138 0 0 863 0 --:--:-- --:--:-- --:--:-- 867
100 57.3M 100 57.3M 0 0 16.4M 0 0:00:03 0:00:03 --:--:-- 17.2M
[sudo] password for kacper:
Client Version: v1.33.0
Kustomize Version: v5.6.0
kacper@UNKNOWN:~$ curl -LO https://storage.googleapis.com/minikube/releases/latest/minikube-linux-amd64
sudo install -m 755 minikube-linux-amd64 /usr/local/bin/minikube
minikube start --driver=docker
% Total % Received % Xferd Average Speed Time Time Time Current
Dload Upload Total Spent Left Speed
100 119M 100 119M 0 0 16.8M 0 0:00:07 0:00:07 --:--:-- 18.3M
🐳 minikube v1.35.0 on Fedora 42
🌟 Using the docker driver based on user configuration
🔧 Using Docker driver with root privileges
👍 Starting "minikube" primary control-plane node in "minikube" cluster
📦 Pulling base image v0.0.46 ...
📦 Downloading Kubernetes v1.32.0 preload ...
> preloaded-images-k8s-v18-v1...: 333.57 MiB / 333.57 MiB 100.00% 7.98 Mi
> gcr.io/k8s-minikube/kicbase...: 500.31 MiB / 500.31 MiB 100.00% 10.80 Mi
🔥 Creating docker container (CPUs=2, Memory=15900MB) ...
🔧 Preparing Kubernetes v1.32.0 on Docker 27.4.1 ...
  ▪ Generating certificates and keys ...
  ▪ Booting up control plane ...
  ▪ Configuring RBAC rules ...
🔗 Configuring bridge CNI (Container Networking Interface) ...
🔧 Verifying Kubernetes components...
  ▪ Using image gcr.io/k8s-minikube/storage-provisioner:v5
🌟 Enabled addons: storage-provisioner, default-storageclass
🏡 Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
kacper@UNKNOWN:~$
```

Step 2: Create a Cluster:

```
kacper@UNKNOWN:~$ minikube status
minikube
type: Control Plane
host: Running
kubelet: Running
apiserver: Running
kubeconfig: Configured

kacper@UNKNOWN:~$ kubectl cluster-info
Kubernetes control plane is running at https://192.168.49.2:8443
CoreDNS is running at https://192.168.49.2:8443/api/v1/namespaces/kube-system/services/kube-dns:dns/proxy

To further debug and diagnose cluster problems, use 'kubectl cluster-info dump'.
kacper@UNKNOWN:~$ kubectl get nodes
NAME STATUS ROLES AGE VERSION
minikube Ready control-plane 2m52s v1.32.0
kacper@UNKNOWN:~$
```

Step 3: Deploy an App:

```
kacper@UNKNOWN:~$ kubectl create deployment hello \
--image=registry.k8s.io/echoserver:1.10
kubectl expose deployment hello --type=NodePort --port=8080
deployment.apps/hello created
service/hello exposed
kacper@UNKNOWN:~$ kubectl get pods -w
NAME                                READY   STATUS    RESTARTS   AGE
hello-64b88c4dc7-gp5bw             1/1     Running   0           11s
^Z
[1]+  Stopped                  kubectl get pods -w
kacper@UNKNOWN:~$ minikube service hello --url      # returns http://<nodeIP>:<nodePort>
curl $(minikube service hello --url)
http://192.168.49.2:32477

Hostname: hello-64b88c4dc7-gp5bw

Pod Information:
  -no pod information available-

Server values:
  server_version=nginx: 1.13.3 - lua: 10008

Request Information:
  client_address=10.244.0.1
  method=GET
  real_path=/
  query=
  request_version=1.1
  request_scheme=http
  request_uri=http://192.168.49.2:8080/

Request Headers:
  accept=/*/*
  host=192.168.49.2:32477
  user-agent=curl/8.11.1

Request Body:
  -no body in request-
```

Step 4: Explore the App:

```
kacper@UNKNOWN:~$ kubectl get pods -o wide
NAME                READY   STATUS    RESTARTS   AGE   IP            NODE       NOMINATED NODE   READINESS GATES
hello-64b88c4dc7-gp5bw 1/1     Running   0           5m54s 10.244.0.4    minikube   <none>           <none>
kacper@UNKNOWN:~$ kubectl describe pod hello-64b88c4dc7-gp5bw
Name:                hello-64b88c4dc7-gp5bw
Namespace:           default
Priority:              0
Service Account:      default
Node:                 minikube/192.168.49.2
Start Time:           Mon, 05 May 2025 14:23:27 +0100
Labels:               app=hello
                     pod-template-hash=64b88c4dc7
Annotations:          <none>
Status:               Running
IP:                   10.244.0.4
IPs:                  IP: 10.244.0.4
Controlled By:        ReplicaSet/hello-64b88c4dc7
Containers:
  echoserver:
    Container ID:      docker://53718330db5dc5c78e2ca9bc23712866d19ad38e316f2c620547ffd77da476fb
    Image:              registry.k8s.io/echoserver:1.10
    Image ID:           docker-pullable://registry.k8s.io/echoserver@sha256:cb5c1bddd1b5665e1867a7fa1b5fa843a47ee433bbb75d4293888b71def53229
    Port:               <none>
    Host Port:          <none>
    State:              Running
      Started:          Mon, 05 May 2025 14:23:32 +0100
    Ready:              True
    Restart Count:      0
    Environment:        <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-b8mjp (ro)
Conditions:
  Type              Status
  PodReadyToStartContainers  True
  Initialized        True
  Ready              True
  ContainersReady    True
  PodScheduled       True
Volumes:
  kube-api-access-b8mjp:
    Type:              Projected (a volume that contains injected data from multiple sources)
    TokenExpirationSeconds: 3607
    ConfigMapName:       kube-root-ca.crt
    Optional:            false
    DownwardAPI:         true
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   6m1s  default-scheduler  Successfully assigned default/hello-64b88c4dc7-gp5bw to minikube
  Normal  Pulling     6m    kubelet         Pulling image "registry.k8s.io/echoserver:1.10"
  Normal  Pulled      5m56s  kubelet         Successfully pulled image "registry.k8s.io/echoserver:1.10" in 3.987s (3.987s including waiting). Image size: 95361986 bytes.
  Normal  Created     5m56s  kubelet         Created container: echoserver
  Normal  Started     5m55s  kubelet         Started container echoserver
kacper@UNKNOWN:~$ kubectl logs hello-64b88c4dc7-gp5bw --tail=5
.....+++
writing new private key to '/certs/privateKey.key'
-----
Starting nginx
10.244.0.1 - - [05/May/2025:13:24:08 +0000] "GET / HTTP/1.1" 200 436 "-" "curl/8.11.1"
```

App own shell

```
kacper@UNKNOWN:~$ kubectl exec -it hello-64b88c4dc7-gp5bw -- /bin/sh
#
```

Step 5: Scale the App:

```
kacper@UNKNOWN:~$ kubectl scale deployment hello --replicas=4
deployment.apps/hello scaled
kacper@UNKNOWN:~$ kubectl get deployments hello
NAME      READY   UP-TO-DATE   AVAILABLE   AGE
hello     4/4     4            4           37m
kacper@UNKNOWN:~$
```

Scaling adjusts the ReplicaSet size; Kubernetes schedules new Pods across nodes and keeps the requested number running.

Check load-balancing:

```
kacper@UNKNOWN:~$ for i in {1..10}; do curl -s $(minikube service hello --url) | grep -o "Hostname.*"; done
Hostname: hello-64b88c4dc7-szzbd
Hostname: hello-64b88c4dc7-q5dxl
Hostname: hello-64b88c4dc7-gp5bw
Hostname: hello-64b88c4dc7-gp5bw
Hostname: hello-64b88c4dc7-b4x7l
Hostname: hello-64b88c4dc7-b4x7l
Hostname: hello-64b88c4dc7-szzbd
Hostname: hello-64b88c4dc7-gp5bw
Hostname: hello-64b88c4dc7-gp5bw
Hostname: hello-64b88c4dc7-gp5bw
```

Step 6: Update the App:

```
kacper@UNKNOWN:~$ kubectl set image deployment hello \
  hello=k8s.gcr.io/echoserver:1.10
error: unable to find container named "hello"
kacper@UNKNOWN:~$ kubectl set image deployment hello \
  hello=k8s.gcr.io/echoserver:1.10
kubectl rollout status deployment hello
kubectl rollout history deployment hello
error: unable to find container named "hello"
deployment "hello" successfully rolled out
deployment.apps/hello
REVISION  CHANGE-CAUSE
1         <none>
kacper@UNKNOWN:~$ kubectl rollout restart deployment hello
deployment.apps/hello restarted
kacper@UNKNOWN:~$
```

Step 7: (Extra) Cleanup:

```
kacper@UNKNOWN:~$ kubectl delete service,deployment hello
minikube stop
service "hello" deleted
deployment.apps "hello" deleted
👋 Stopping node "minikube" ...
🔴 Powering off "minikube" via SSH ...
🔴 1 node stopped.
kacper@UNKNOWN:~$
```