

Kubernetes API

Adfinis**sy**Group

Be smart. Think open source.

Kubernetes API

Create a namespace for yourself

```
apiVersion: v1
kind: Namespace
metadata:
  name: example
```

Choose a name for your namespace and save it as `namespace.yaml`

Create a namespace for yourself

```
$ kubectl apply -f 00_namespace.yaml  
namespace/example created
```

Make this namespace your default

```
$ kubectl config set-context --current --namespace=example
```

Create a pod to start a single container

```
apiVersion: v1
kind: Pod
metadata:
  name: nginx-demo
spec:
  containers:
  - name: nginx-echo
    image: adfinissygroup/nginx-echo
```

Create a pod to start a single container

```
$ kubectl apply -f 01_pod.yaml  
pod/nginx-demo created
```


Use describe to check the Pod IP

```
$ kubectl describe pod nginx-demo | grep '^IP'  
IP: 172.17.0.14
```

Try to access the pod using curl

```
$ curl 172.17.0.14
```

Add labels to the pod

```
apiVersion: v1
kind: Pod
metadata:
  name: nginx-example
  labels:
    app: web
spec:
  containers:
  - name: nginx-echo
    image: adfinissygroup/nginx-echo
```

Add labels to the pod

```
$ kubectl apply -f 01_pod.yaml  
pod/nginx-example configured
```

Create a service for you pod

```
apiVersion: v1
kind: Service
metadata:
  labels:
    app: web
  name: nginx-example
spec:
  ports:
    - name: http
      port: 80
      protocol: TCP
      targetPort: 80
  selector:
    app: web
```

Create a service for your pod

```
$ kubectl apply -f 02_service.yaml  
service/nginx-example created
```

Create a service for you pod

```
apiVersion: v1
kind: Service
metadata:
  labels:
    app: web
  name: nginx-example
spec:
  ports:
    - name: http
      port: 80
      protocol: TCP
      targetPort: 80
  selector:
    app: web
```

Expose your service via ingress

```
apiVersion: extensions/v1beta1
kind: Ingress
metadata:
  name: nginx-example
  labels:
    app: web
spec:
  rules:
  - host: hello-world.example.com
    http:
      paths:
      - backend:
          serviceName: nginx-example
          servicePort: 80
```


Expose your service via ingress

```
$ kubectl apply -f 02_ingress.yaml  
ingress.extensions/nginx-example created
```

Check that you see the right content on your Ingress.

Create a deployment in your namespace

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-example
  labels:
    app: web
spec:
  selector:
    matchLabels:
      app: web
  template:
    metadata:
      labels:
        app: web
    spec:
      containers:
        - name: nginx-echo
          image: adfinissygroup/nginx-echo
```

Create a deployment in your namespace

```
$ kubectl apply -f 02_deployment.yaml  
deployment.apps/nginx-example created
```

Scale your deployment

```
$ kubectl scale deployment --replicas=2 nginx-example  
deployment.extensions/nginx-example scaled
```

Delete the single pod instance

```
$ kubectl delete pod nginx-example  
pod "nginx-example" deleted
```

Request a PVC for your deployment

```
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: nginx-example
  labels:
    app: web
spec:
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 16Mi
```

Request a PVC for your deployment

```
$ kubectl apply -f 01_pvc.yaml  
persistentvolumeclaim/nginx-example created
```

Patch the deployment to include the PVC

```
[{"op": "add",
 "path": "/spec/template/spec/containers/0/volumeMounts",
 "value": [{"mountPath": "/usr/local/apache2/htdocs",
 "name": "htdocs"}]}],
{
  "op": "add",
  "path": "/spec/template/spec/volumes",
  "value": [{"name": "htdocs",
 "persistentVolumeClaim": {
 "claimName": "nginx-example"}]}]}]
```


Patch the deployment to include the PVC

```
$ kubectl patch --type=json -p "$(cat patch.json)" deployments.apps nginx-example  
deployment.apps/nginx-example patched
```

Now you should see different content when you access your container via Ingress.

Repeat while showing events

```
$ kubectl get events -w
```

Attribution / License

- Slide Skeleton [https://de.wikipedia.org/wiki/Skeleton_\(Programmierung\)](https://de.wikipedia.org/wiki/Skeleton_(Programmierung))
- Slides Adfinis SyGroup AG, 2017, Attribution-NonCommercial 2.0 (CC BY-NC 2.0)

Feel Free to Contact Us

www.adfinis-sygroup.ch

[Tech Blog](#)

[GitHub](#)

info@adfinis-sygroup.ch

[Twitter](#)

