

Lab 5: Pod Scaling, Readiness Probe und Self Healing

In diesem Lab wird aufgezeigt, wie man Applikationen in OpenShift skaliert. Des Weiteren wird angeschaut, wie OpenShift dafür sorgt, dass jeweils die Anzahl erwarteter Pods gestartet wird und wie schlussendlich eine Applikation der Plattform zurückmelden kann, dass sie bereit für Requests ist.

Vorbereiten der Beispiels Applikation für die Skalierung

Bevor wir hier mit den Lab Aufgaben beginnen können, braucht es zufror noch eine kleine Vorarbeit.

1. Als erstes wird ein neues Projekt erstellt:

```
# oc new-project [MEINNAME]-scale
```

2. Anschliessend fügen wir dem Projekt eine Applikation hinzu:

```
# oc new-app appuio/example-php-docker-helloworld --name=appuio-php-docker
```

3. Zum Schluss stellen wir den Service zur Verfügung: (expose)

```
# oc expose service appuio-php-docker
```

Wenn wir nun unsere Beispiels Applikation skalieren wollen, müssen wir unserem ReplicationController (rc) mitteilen, dass wir stets **drei Replicas** des Images am Laufen haben wollen.

Schauen wir uns mal den ReplicationController (rc) etwas genauer an:

```
# oc get rc
```

NAME	DESIRED	CURRENT	AGE
appuio-php-docker-1	1	1	33s

Für mehr Details:

```
# oc get rc appuio-php-docker-1 -o json
```

Der rc sagt uns, wie viele Pods wir erwarten (DESIRED) und wie viele aktuell deployt sind (CURRENT).

Aufgabe: LAB5.1 skalieren der Beispiel Applikation

Nun skalieren wir unsere Beispiels Applikation auf 3 Replicas:

```
# oc scale --replicas=3 dc appuio-php-docker
```

Anschliessend überprüfen wir die Anzahl Replicas auf dem ReplicationController:

```
# oc get rc
```

NAME	DESIRED	CURRENT	AGE
appuio-php-docker-1	3	3	1m

und zeigen entsprechend die Pods an:

```
# oc get pods
```

NAME	READY	STATUS	RESTARTS	AGE
appuio-php-docker-1-2uc89	1/1	Running	0	21s
appuio-php-docker-1-evcre	1/1	Running	0	21s
appuio-php-docker-1-tolpx	1/1	Running	0	2m

Zum Schluss schauen wir uns den Service an. Der sollte jetzt alle drei Endpoints referenzieren:

```
# oc describe svc appuio-php-docker
```

```
Name:          appuio-php-docker
Namespace:     techlab-scale
Labels:       app=appuio-php-docker
Selector:     app=appuio-php-docker,deploymentconfig=appuio-php-docker
Type:        ClusterIP
IP:          172.30.166.88
Port:        8080-tcp      8080/TCP
Endpoints:   10.1.3.23:8080,10.1.4.13:8080,10.1.5.15:8080
Session Affinity:  None
No events.
```

Skalieren von Pods innerhalb eines Services ist sehr schnell, da OpenShift einfach eine neue Instanz

des Docker Images als Container startet.

Tipp: OpenShift v3 unterstützt auch Autoscaling, die Dokumentation dazu ist unter dem folgenden Link zu finden: https://docs.openshift.com/container-platform/3.9/dev_guide/pod_autoscaling.html

Last update: **2018/07/11 16:20**