Lab 5 - Zadanie

Łukasz Chudy 92844

lab5-quota.yaml

```
apiVersion: v1
kind: ResourceQuota
metadata:
  name: resource-quota
  namespace: zad5
spec:
  hard:
    pods: "10"
    cpu: 2000m
    memory: 1.5Gi
```

lab5-worker-pod.yaml

```
apiVersion: v1
kind: Pod
metadata:
  name: worker
  namespace: zad5
spec:
  containers:
  - name: nginx-container
    image: nginx
    resources:
      limits:
        memory: "200Mi"
        cpu: "200m"
      requests:
        memory: "100Mi"
        cpu: "100m"
```

lab5-php-apache.yaml

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: php-apache
  namespace: zad5
spec:
  selector:
    matchLabels:
      run: php-apache
  template:
    metadata:
      labels:
        run: php-apache
    spec:
      containers:
        - name: php-apache
          image: registry.k8s.io/hpa-example
          ports:
            - containerPort: 80
          resources:
            limits:
              memory: 250Mi
              cpu: 250m
            requests:
              memory: 150Mi
              cpu: 150m
---
apiVersion: v1
kind: Service
metadata:
  name: php-apache
  namespace: zad5
  labels:
    run: php-apache
spec:
  ports:
    - port: 80
  selector:
    run: php-apache
```

zad5-HorizontalPodAutoscaler.yaml

Aby autoskaler działał poprawnie, musi mieć dostęp do metryk. Serwer metryk włączamy komendą:

```
minikube addons enable metrics-server
```

```
apiVersion: autoscaling/v1
kind: HorizontalPodAutoscaler
metadata:
  name: php-apache-hpa
  namespace: zad5
spec:
  scaleTargetRef:
    apiVersion: apps/v1
    kind: Deployment
    name: php-apache
  minReplicas: 1
  maxReplicas: 5
  targetCPUUtilizationPercentage: 50
```

Maksymalna liczba replik wynika z ograniczeń nadanych przez ResourceQuota. Umożliwia ona utworzenie 10 pod'ów, użycie CPU w ilości 2000m oraz pamięci w ilości 1.5Gi.

Działa również pod worker, który jest ograniczony do maksymalnie 200m CPU i 200Mi pamięci.

Po uwzględnieniu poda worker, dostępne zostaje 1800m CPU, 1.3Gi pamięci oraz 9 podów.

1800m / 250m = 7,2 1.3Gi / 250 = 5,2

Po przeanalizowaniu dostępnych zasobów i porównaniu ich z maksymalnymi używanymi przez pody, racjonalnym wyborem jest ustawienie autoscaler'a na maksymalnie 5 pod'ów.

Utworzenie obiektów

```
kubectl apply -f lab5-quota.yaml
```

```
kubectl apply -f zad5-worker-pod.yaml
```

```
kubectl apply -f lab5-php-apache.yaml
```

```
kubectl apply -f zad5-HorizontalPodAutoscaler.yaml
```

Polecenia do weryfikacji

```
kubectl run -i --tty load-generator --rm --image=busybox:1.28 --restart=Never -- /bin/sh -c "while sleep 0.01
```

```
C:\Users\lukas>kubectl run -i --tty load-generator --rm --image=busybox:1.28 --restart=Never -- /bin/sh -c "while sleep 0.01; do wget -q -O- http://php-apache.zad5.svc.cluster.local; done"
If you don't see a command prompt, try pressing enter.
OK!OK!OK!OK!OK!OK!OK!OK!OK!OK!OK!OK!OK!OK!OK!OK!OK!OK!OK!OK!OK!OK!OK!OK!OK!OK!OK!OK!OK!OK!OK!OK!OK!OK!OK!OK!OK!OK!OK!OK!
```

```
kubectl get hpa php-apache-hpa -n zad5
```

```
D:\Studia IX\3. Programowanie full-stack w chmurze\Sprawozdanie_5>kubectl get hpa php-apache-hpa -n zad5
NAME             REFERENCE               TARGETS   MINPODS   MAXPODS   REPLICAS   AGE
php-apache-hpa   Deployment/php-apache   0%/50%    1         5         1          39h

D:\Studia IX\3. Programowanie full-stack w chmurze\Sprawozdanie_5>kubectl get hpa php-apache-hpa -n zad5
NAME             REFERENCE               TARGETS    MINPODS   MAXPODS   REPLICAS   AGE
php-apache-hpa   Deployment/php-apache   140%/50%   1         5         3          39h
D:\Studia IX\3. Programowanie full-stack w chmurze\Sprawozdanie_5>kubectl get hpa php-apache-hpa -n zad5
NAME             REFERENCE               TARGETS   MINPODS   MAXPODS   REPLICAS   AGE
php-apache-hpa   Deployment/php-apache   86%/50%   1         5         3          39h

D:\Studia IX\3. Programowanie full-stack w chmurze\Sprawozdanie_5>kubectl get hpa php-apache-hpa -n zad5
NAME             REFERENCE               TARGETS   MINPODS   MAXPODS   REPLICAS   AGE
php-apache-hpa   Deployment/php-apache   72%/50%   1         5         5          39h
```

```
kubectl describe hpa php-apache-hpa -n zad5
```

```
D:\Studia IX\3. Programowanie full-stack w chmurze\Sprawozdanie_5>kubectl describe hpa php-apache-hpa -n zad5
Name:                     php-apache-hpa
Namespace:                zad5
Labels:                   <none>
Annotations:              autoscaling.alpha.kubernetes.io/conditions:
                            [{"type":"AbleToScale","status":"True","lastTransitionTime":"2023-11-26T17:08:12Z","reason":"ReadyForNewScale","message":"recommended size...
                          autoscaling.alpha.kubernetes.io/current-metrics:
                            [{"type":"Resource","resource":{"name":"cpu","currentAverageUtilization":46,"currentAverageValue":"70m"}}]
CreationTimestamp:        Sun, 26 Nov 2023 18:07:57 +0100
Reference:                Deployment/php-apache
Target CPU utilization:   50%
Current CPU utilization:  46%
Min replicas:             1
Max replicas:             5
Deployment pods:          5 current / 5 desired
Events:
  Type     Reason                        Age                From                       Message
  ----     ------                        ----               ----                       -------
  Warning  FailedComputeMetricsReplicas  39h (x12 over 39h)  horizontal-pod-autoscaler  invalid metrics (1 invalid out of 1), first error is: failed to get cpu resource metric value: failed to get cpu utilizatio
n: unable to get metrics for resource cpu: unable to fetch metrics from resource metrics API: the server could
he server could not find the requested resource (get pods.metrics.k8s.io)
  Warning  FailedGetResourceMetric       39h (x13 over 39h)  horizontal-pod-autoscaler  failed to get cpu utilization: unable to get metrics for resource cpu: unable to fetch metrics from resource metrics API: t
he server could not find the requested resource (get pods.metrics.k8s.io)
  Warning  FailedComputeMetricsReplicas  49m (x12 over 52m)  horizontal-pod-autoscaler  invalid metrics (1 invalid out of 1), first error is: failed to get cpu resource metric value: failed to get cpu utilizatio
n: unable to get metrics for resource cpu: unable to fetch metrics from resource metrics API: the server could
he server could not find the requested resource (get pods.metrics.k8s.io)
  Warning  FailedGetResourceMetric       37m (x61 over 52m)  horizontal-pod-autoscaler  failed to get cpu utilization: unable to get metrics for resource cpu: unable to fetch metrics from resource metrics API: t
he server could not find the requested resource (get pods.metrics.k8s.io)
  Normal   SuccessfulRescale             6m49s              horizontal-pod-autoscaler  New size: 3; reason: cpu resource utilization (percentage of request) above target
  Normal   SuccessfulRescale             4m49s              horizontal-pod-autoscaler  New size: 5; reason: cpu resource utilization (percentage of request) above target
```

```
kubectl get resourcequota -n zad5
```

```
D:\Studia IX\3. Programowanie full-stack w chmurze\Sprawozdanie_5>kubectl get resourcequota -n zad5
NAME             AGE   REQUEST                                               LIMIT
resource-quota   39h   cpu: 850m/2, memory: 850Mi/1536Mi, pods: 6/10

D:\Studia IX\3. Programowanie full-stack w chmurze\Sprawozdanie_5>
```

```
D:\Studia IX\3. Programowanie full-stack w chmurze\Sprawozdanie_5>kubectl get hpa php-apache-hpa -n zad5
NAME             REFERENCE               TARGETS   MINPODS   MAXPODS   REPLICAS   AGE
php-apache-hpa   Deployment/php-apache   45%/50%   1         5         5          39h
```

Ostateczny target 45%.

Używane CPU: 850m/2000m Używana pamięć: 850Mi/1536Mi Używane pody 6/10 Autoskaler utworzył 5 replik

resource-quota 39h cpu: 850m/2, memory: 850Mi/1536Mi, pods: 6/10