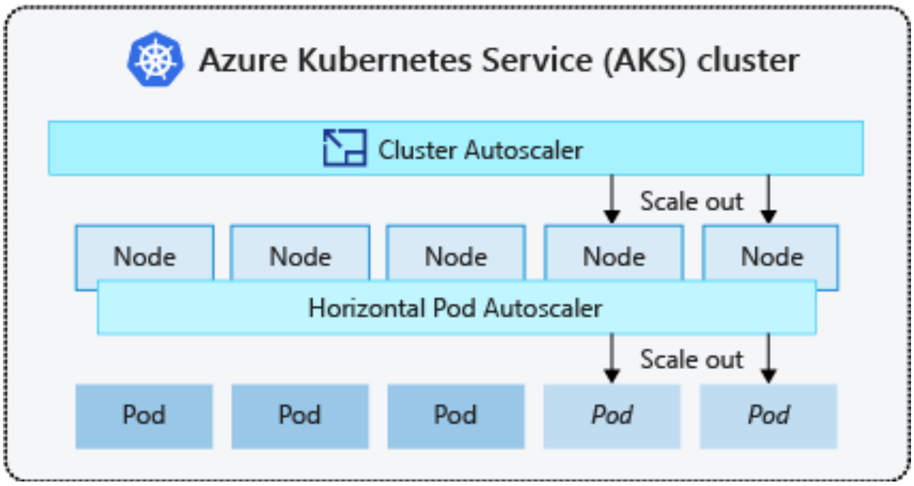
Kubernetes Toy Setup to Reproduce Issues



Creation of Resource Group

```
    az group create --name myResourceGroup --location eastus
```

Creation of AKS Cluster and Enabling Cluster Autoscaler

```
    az aks create --resource-group myResourceGroup --name myAKSCluster --node-vm-size Standard_DS1_v2 --node-count 1 --vm-set-type VirtualMachineScaleSets --load-balancer-sku standard --enable-cluster-autoscaler --min-count 1 --max-count 3
```

More Information

1. Automatically scale a cluster to meet application demands on Azure Kubernetes Service (AKS)
2. az aks - commands
3. Dv2 and DSv2-series

Run and Expose php-apache Server

Define a Docker Image

```
$ vi Dockerfile

FROM php-5-apache
COPY index.php /var/www/index.php
RUN chmod a+rx index.php
```

Define CPU Intensive Application

```
$ vi index.php

<?php
    $x = 0.0001;
    for ($i = 0; $i <= 1000000; $i++) {
    $x += sqrt($x);
    }
    echo "OK!";
?>
```

Define Kubernetes Configuration for the Application

```
$ vi php-apache.yml

apiVersion: apps/v1
kind: Deployment
metadata:
    name: php-apache
spec:
    selector:
        matchLabels:
            run: php-apache
    replicas: 1
    template:
        metadata:
            labels:
                run: php-apache
        spec:
            containers:
                - name: php-apache
                  image: k8s.gcr.io/hpa-example
                  ports:
                      - containerPort: 80
                  resources:
                      limits:
                          cpu: 500m
                          memory: 128Mi
                      requests:
                          cpu: 200m
                          memory: 64Mi
---
apiVersion: v1
kind: Service
metadata:
    name: php-apache
    labels:
        run: php-apache
spec:
    ports:
        - port: 80
selector:
    run: php-apache
```

Deploy and Test the Setup

```
kubectl apply -f php-apache.yaml
kubectl get all
kubectl get nodes
```

Enable Autoscaling the Pods

```
    kubectl autoscale deployment php-apache --cpu-percent=50 --min=1 --max=10
```

Test Autoscaling Setup

```
    kubectl get hpa
```

Increase Load

```
    kubectl run -i --tty load-generator --rm --image=busybox --restart=Never -- /bin/sh -c "while sleep 0.01; do wget -q -O- http://php-apache; done"
```

Test Autoscaling

```
    kubectl get hpa
    kubectl get deployment php-apache
```

Stop Load

1. Hit `Ctrl` + `C`

   ```
   kubectl get deployment php-apache
   ```

Delete the Entire Setup

```
    az group delete --name myResourceGroup --yes --no-wait
```