

Madvisor - AutoML



Table of content

	Page
1. Prerequisite.....	2
1. Create ACR.....	2
2. ACI deployment of ACR image.....	3
3. AKS deployment of ACR image.....	3

Prerequisite

You need to have a Resource Group and Azure Container Registry created before subscribing to this offer.

Create ACR

ACR_NAME=<registry-name> #Container registry name

RES_GROUP=<resource group name> #Resource Group name

IMG_NAME=<name of your image>

Create a resource group:

```
az group create --resource-group $RES_GROUP --location eastus
```

Create Container registry:

```
az acr create --resource-group $RES_GROUP --name $ACR_NAME --sku Standard --location eastus --admin-enabled true
```

get the ACR password:

```
ACR_password = az acr credential show -n $ACR_NAME --query "passwords[0].value" -o tsv
```

Reference:

<https://docs.microsoft.com/en-us/azure/container-registry/container-registry-get-started-azure-cli>

<https://docs.microsoft.com/en-us/azure/container-registry/container-registry-tutorial-quick-task>

After subscribing to this offer via browser, use Azure CLI to access AutoML in the container:

ACI deployment of ACR image

IMG_NAME=automl

```
az container create --resource-group $RES_GROUP --name $IMG_NAME --image
$ACR_NAME.azurecr.io/$IMG_NAME:<tag>--registry-login-server $ACR_NAME.azurecr.io
--registry-username $ACR_NAME --registry-password $ACR_password --cpu 4 --memory 16
--dns-name-label <desired dns name> --ports 80
```

Reference:

<https://docs.microsoft.com/en-us/azure/container-instances/container-instances-using-azure-container-registry#deploy-container-with-azure-cli>

<https://markheath.net/post/build-container-images-with-acr>

It is also possible to have this container as part of AKS cluster:

AKS deployment of ACR image

Install the AKS CLI:

```
az aks install-cli
```

If you don't have a cluster already, create one:

CLUSTER_NAME=<cluster name you want to have>

```
az aks create -n $CLUSTER_NAME -g $RES_GROUP --node-count <number of cluster nodes>
--enable-addons monitoring --generate-ssh-keys --service-principal <your service principal> --
client-secret <service principal client secret>
```

See the details of the pods created:

```
kubectl get nodes
```

Download the configuration settings and set the current context to connect to your kube cluster:

```
az aks get-credentials -g $RES_GROUP -n $CLUSTER_NAME
```

Create a “docker-registry” secret in the cluster, which you can then use in your yaml file:

```
kubectl create secret docker-registry <secret name> --docker-server= $ACR_NAME.azurecr.io  
--docker-username= $ACR_NAME --docker-password= $ACR_password --docker-email=<Any  
valid email>
```

Create a file named kubernetes.yaml with the contents below. Make sure you use the correct image name and secret that we have previously defined.

```
apiVersion: apps/v1  
kind: Deployment  
metadata:  
  name: automl  
spec:  
  replicas: 1  
  selector:  
    matchLabels:  
      app: automl  
  template:  
    metadata:  
      labels:  
        app: automl  
    spec:  
      imagePullSecrets:  
        - name: <secret name>  
      containers:  
        - name: automl  
          image: <acr name>.azurecr.io/<image name>:latest  
          ports:  
            - containerPort: 80  
---  
apiVersion: v1  
kind: Service  
metadata:  
  name: automl  
spec:  
  type: LoadBalancer  
  ports:  
    - port: 80  
  selector:  
    app: automl
```

Deploy the application into your cluster:

```
kubectl apply -f kubernetes.yaml
```

See the details of the pods created:

```
kubectl get pods
```

Stop an AKS Cluster

```
az aks stop --name $CLUSTER_NAME --resource-group $RES_GROUP
```

Start an AKS Cluster

```
az aks start --name $CLUSTER_NAME --resource-group $RES_GROUP
```

Check AKS Cluster logs

```
kubectl logs ${POD_NAME} ${CONTAINER_NAME}
```

Ref:

<https://medium.com/@pjbfg/azure-kubernetes-service-aks-pulling-private-container-images-from-azure-container-registry-acr-9c3e0a0a13f2>

<https://docs.microsoft.com/en-us/azure/aks/tutorial-kubernetes-deploy-cluster>

<https://docs.microsoft.com/en-us/azure/aks/tutorial-kubernetes-deploy-application>