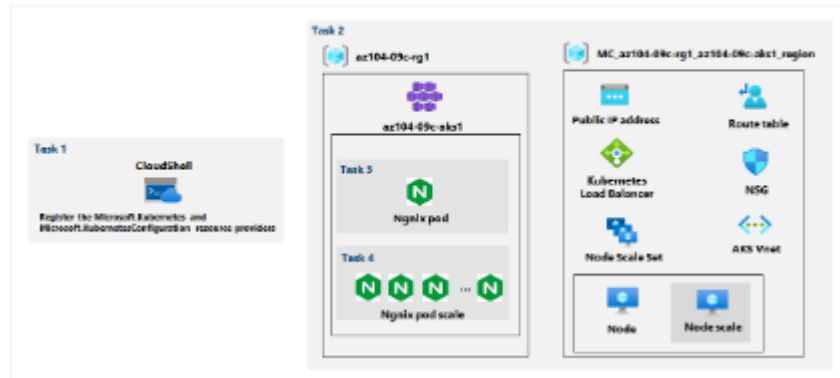


# Lab 09c - Implement Azure Kubernetes Service.

## Architecture diagram



- Task 1: Register the Microsoft.Kubernetes and Microsoft.KubernetesConfiguration resource providers.
  - In this task, you will register resource providers necessary to deploy an Azure Kubernetes Services cluster.

```
PS /home/stanislaw> Register-AzResourceProvider -ProviderNamespace Microsoft.Kubernetes

ProviderNamespace : Microsoft.Kubernetes
RegistrationState  : Registering
ResourceTypes      : {connectedClusters, locations, locations/operationStatuses, registeredSubscriptions...}
Locations          : {West Europe, East US, West Central US, South Central US...}

PS /home/stanislaw>
PS /home/stanislaw> Register-AzResourceProvider -ProviderNamespace Microsoft.KubernetesConfiguration

ProviderNamespace : Microsoft.KubernetesConfiguration
RegistrationState  : Registering
ResourceTypes      : {sourceControlConfigurations, extensions, fluxConfigurations, operations...}
Locations          : {East US, West Europe, West Central US, West US 2...}

PS /home/stanislaw> 
```

- Register the Microsoft.Kubernetes and Microsoft.KubernetesConfiguration resource providers.

- Task 2: Deploy an Azure Kubernetes Service cluster.
  - In this task, you will deploy an Azure Kubernetes Services cluster by using the Azure portal.

The screenshot shows the Azure portal interface for creating and managing a Kubernetes cluster. The top section, 'Create Kubernetes cluster', displays the configuration for a new cluster named 'az104-9c-aks1' in the 'az104-9c-rg1' resource group. The configuration includes a 'Dev/Test' preset, 'East US 2' region, 'None' availability zones, 'Free' AKS pricing tier, '1.24.10' Kubernetes version, and 'Standard B4ms' node size. The node count is set to 1. The bottom section, 'Deployment is in progress', shows the deployment details for 'microsoft.aks-20230407182142', including the start time and correlation ID. A table lists the resources, showing 'az104-9c-aks1' as a 'Microsoft.ContainerService' resource with a 'Created' status.

- Deploy an Azure Kubernetes Service cluster

The screenshot shows the 'Node pools' page for the 'az104-9c-aks1' cluster. The page displays a table with the following data:

Node pool	Provisioning state	Power state	Node count	Mode	Kubernetes version	Node size	Operating system
agentpool	Succeeded	Running	1/1 ready	System	1.24.10	Standard_B4ms	Linux

- verify that the cluster consists of a single pool with one node.

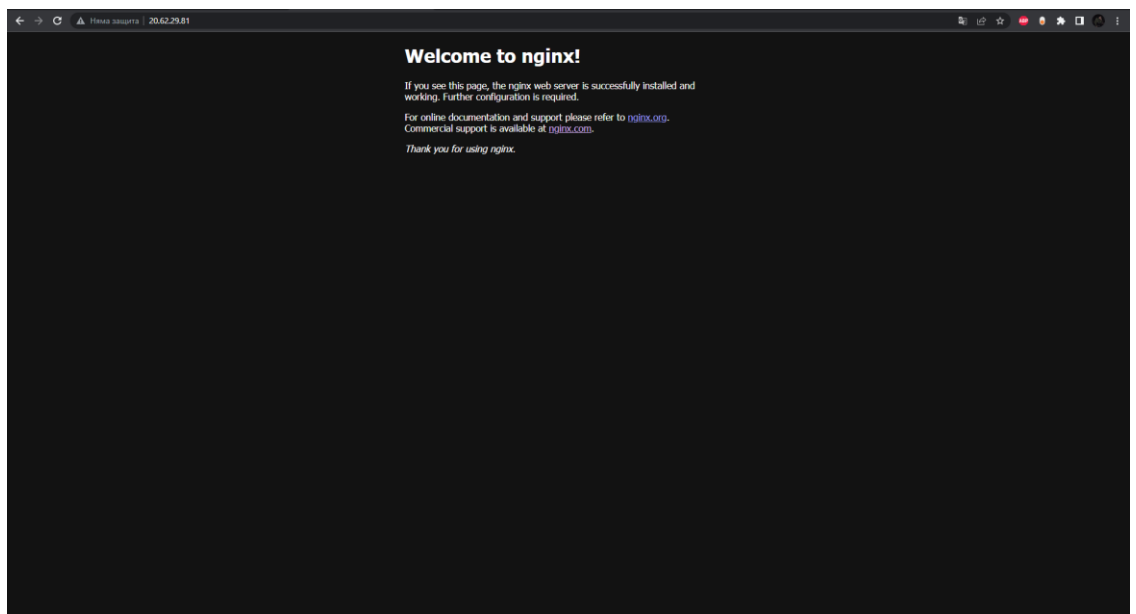
- Task 3: Deploy pods into the Azure Kubernetes Service cluster.
  - In this task, you will deploy a pod into the Azure Kubernetes Service cluster.

```
stanislav [ ~ ]$ RESOURCE_GROUP='az104-09c-rg1'

AKS_CLUSTER='az104-9c-aks1'

az aks get-credentials --resource-group $RESOURCE_GROUP --name $AKS_CLUSTER
Merged "az104-9c-aks1" as current context in /home/stanislav/.kube/config
stanislav [ ~ ]$ kubectl get nodes
NAME                                STATUS    ROLES    AGE      VERSION
aks-agentpool-36281101-vmss000000 Ready     agent    8m53s    v1.24.10
stanislav [ ~ ]$ kubectl create deployment nginx-deployment --image=nginx
deployment.apps/nginx-deployment created
stanislav [ ~ ]$ kubectl get pods
NAME                                READY    STATUS    RESTARTS   AGE
nginx-deployment-85c6d5f6dd-v4dm8  1/1      Running   0           20s
stanislav [ ~ ]$ kubectl get deployment
NAME                READY   UP-TO-DATE   AVAILABLE   AGE
nginx-deployment    1/1     1             1           28s
stanislav [ ~ ]$ kubectl expose deployment nginx-deployment --port=80 --type=LoadBalancer
service/nginx-deployment exposed
stanislav [ ~ ]$ kubectl get service
NAME                TYPE        CLUSTER-IP    EXTERNAL-IP    PORT(S)          AGE
kubernetes          ClusterIP   10.0.0.1      <none>         443/TCP          10m
nginx-deployment    LoadBalancer 10.0.227.96   20.62.29.81    80:30025/TCP     10s
stanislav [ ~ ]$
```

- pod available from Internet



- Verify that the browser page displays the Welcome to nginx! message.

- Task 4: Scale containerized workloads in the Azure Kubernetes service cluster.
  - In this task, you will scale horizontally the number of pods and then number of cluster nodes.

```
stanislav [ ~ ]$ RESOURCE_GROUP='az104-09c-rg1'

AKS_CLUSTER='az104-9c-aks1'

az aks scale --resource-group $RESOURCE_GROUP --name $AKS_CLUSTER --node-count 2
{
  "aadProfile": null,
  "addonProfiles": {
    "azureKeyvaultSecretsProvider": {
      "config": null,
      "enabled": false,
      "identity": null
    },
    "azurepolicy": {
      "config": null,
      "enabled": false,
      "identity": null
    }
  },
  "agentPoolProfiles": [
    {
      "availabilityZones": null,
      "count": 2,
      "creationData": null,
      "currentOrchestratorVersion": "1.24.10",
      "enableAutoScaling": false,
      "enableEncryptionAtHost": null,
      "enableFips": false,
      "enableNodePublicIp": false,
      "enableUltraSsd": null,
      "gpuInstanceProfile": null,
      "hostGroupId": null,
      "kubeletConfig": null,
      "kubeletDiskType": "OS",
      "linuxOsConfig": null,
      "maxCount": null,
      "maxPods": 110,
      "minCount": null,
      "mode": "System",
      "name": "agentpool",

```

- the following will scale out the cluster by increasing the number of nodes to 2

```
stanislav [ ~ ]$ kubectl get nodes
```

NAME	STATUS	ROLES	AGE	VERSION
aks-agentpool-36281101-vmss000000	Ready	agent	23m	v1.24.10
aks-agentpool-36281101-vmss000001	Ready	agent	3m35s	v1.24.10

- verify that the number of nodes increased to 2.

```
stanislav [ ~ ]$ kubectl scale --replicas=10 deployment/nginx-deployment
deployment.apps/nginx-deployment scaled
stanislav [ ~ ]$ kubectl get pods
```

NAME	READY	STATUS	RESTARTS	AGE
nginx-deployment-85c6d5f6dd-4swpb	1/1	Running	0	43s
nginx-deployment-85c6d5f6dd-525wc	1/1	Running	0	13m
nginx-deployment-85c6d5f6dd-55zjr	1/1	Running	0	43s
nginx-deployment-85c6d5f6dd-747g5	1/1	Running	0	43s
nginx-deployment-85c6d5f6dd-9sj96	1/1	Running	0	43s
nginx-deployment-85c6d5f6dd-gl9vw	1/1	Running	0	43s
nginx-deployment-85c6d5f6dd-j88f5	1/1	Running	0	43s
nginx-deployment-85c6d5f6dd-plmh8	1/1	Running	0	43s
nginx-deployment-85c6d5f6dd-v4dm8	1/1	Running	0	20m
nginx-deployment-85c6d5f6dd-xgfnd	1/1	Running	0	43s

```
stanislav [ ~ ]$
```

- verify the outcome of scaling the deployment

```
stanislav [ ~ ]$ kubectl get pod -o=custom-columns=NODE:.spec.nodeName,POD:.metadata.name
```

NODE	POD
aks-agentpool-36281101-vmss000001	nginx-deployment-85c6d5f6dd-4swpb
aks-agentpool-36281101-vmss000000	nginx-deployment-85c6d5f6dd-525wc
aks-agentpool-36281101-vmss000001	nginx-deployment-85c6d5f6dd-55zjr
aks-agentpool-36281101-vmss000001	nginx-deployment-85c6d5f6dd-747g5
aks-agentpool-36281101-vmss000000	nginx-deployment-85c6d5f6dd-9sj96
aks-agentpool-36281101-vmss000000	nginx-deployment-85c6d5f6dd-gl9vw
aks-agentpool-36281101-vmss000000	nginx-deployment-85c6d5f6dd-j88f5
aks-agentpool-36281101-vmss000001	nginx-deployment-85c6d5f6dd-plmh8
aks-agentpool-36281101-vmss000000	nginx-deployment-85c6d5f6dd-v4dm8
aks-agentpool-36281101-vmss000001	nginx-deployment-85c6d5f6dd-xgfnd

```
stanislav [ ~ ]$ kubectl delete deployment nginx-deployment
deployment.apps "nginx-deployment" deleted
stanislav [ ~ ]$
```

- review the pods distribution across cluster nodes