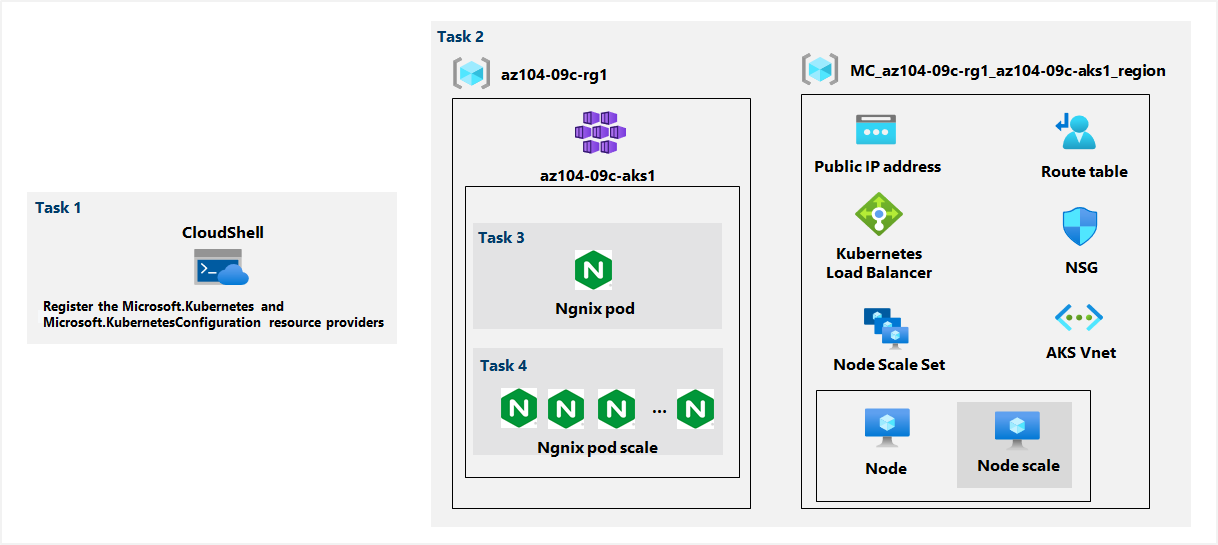
Objectives:

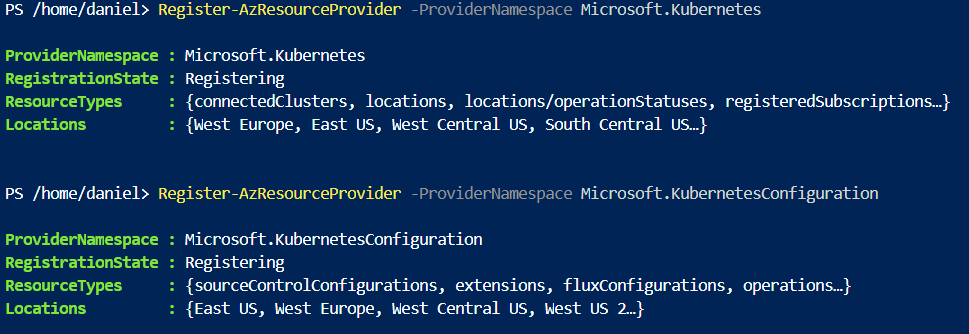
* Task 1: Register the Microsoft.Kubernetes and Microsoft.KubernetesConfiguration resource providers.
* Task 2: Deploy an Azure Kubernetes Service cluster.
* Task 3: Deploy pods into the Azure Kubernetes Service cluster.
* Task 4: Scale containerized workloads in the Azure Kubernetes service cluster.

Architecture diagram:



Task 1: Register the Microsoft.Kubernetes and Microsoft.KubernetesConfiguration resource providers.

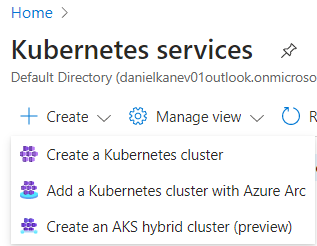
In this task, we have to open up the Azure PowerShell and register the Microsoft.Kubernetes and Microsoft.KubernetesConfiguration resource providers with the following commands:

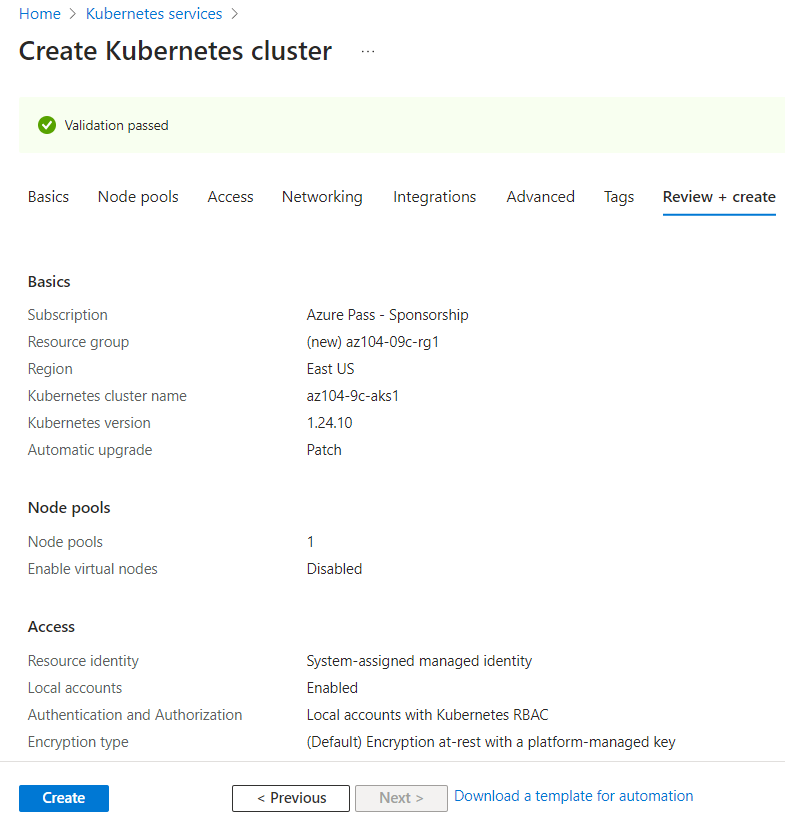


Task 2: Deploy an Azure Kubernetes Service cluster.

In this task, we will deploy an Azure Kubernetes Services cluster by using the Azure portal.

Here, we just create a new cluster with the required settings:

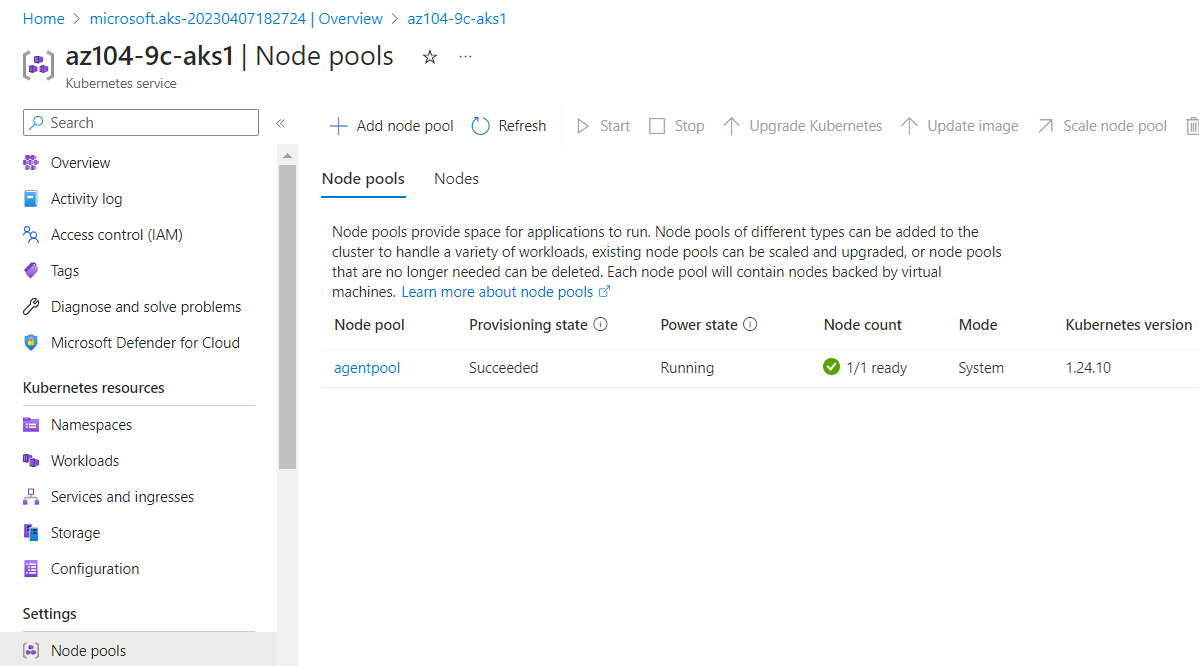




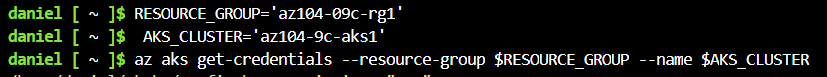
Task 3: Deploy pods into the Azure Kubernetes Service cluster.

In this task, we will deploy a pod into the Azure Kubernetes Service cluster.

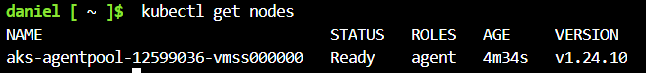
First, we go to our resource of the newly created cluster -> Settings section -> select Node pools and verify that the cluster consists of a single pool with one node.



Then, we open up the Cloud Shell and run a few commands to retrieve the credentials to access the AKS cluster:



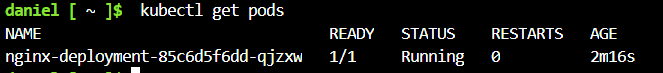
And we verify the connectivity to the cluster:



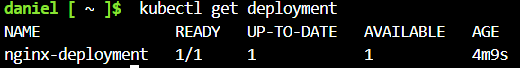
We then deploy the nginx image from the Docker Hub:



And verify that it has been created:



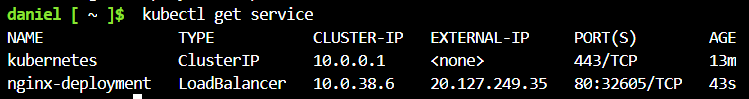
With the below command, we check the state of the deployment:



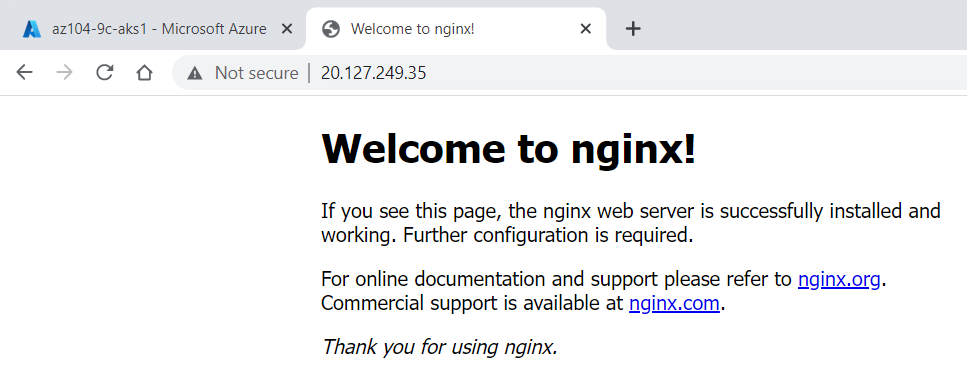
Making it available from the Internet:



And check whether a public IP address has been provisioned:



When we go to the public IP address of the nginx-deployment, the browser displays the correct message:



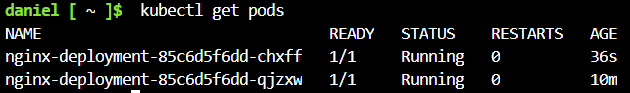
Task 4: Scale containerized workloads in the Azure Kubernetes service cluster.

In this task, we will scale horizontally the number of pods and then number of cluster nodes.

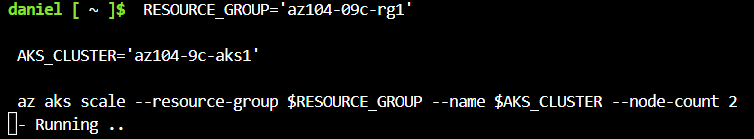
We start off by increasing the number of pods to 2:



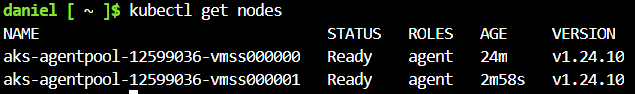
And verify that pods are now 2:



Next, we scale out the cluster by increasing the number of nodes to 2:



And verify that the outcome is correct:



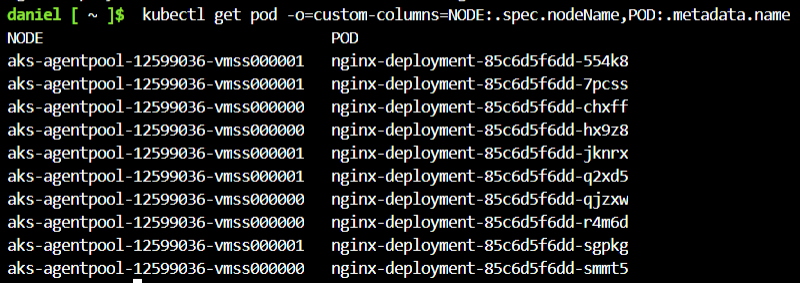
We then scale the deployment:



And verify the outcome:



We then review the pods distribution across cluster nodes:



Finally, we delete the deployment:

