

## **Cómo usar Azure Kubernetes Service para la orquestación de contenedores**

La contenedorización ha transformado la forma en que se crean, implementan y escalan las aplicaciones. Los contenedores proporcionan un entorno ligero, portátil y uniforme para que los desarrolladores agrupen sus programas y sus dependencias en una sola unidad.

Pero cuando aumenta el número de contenedores de un sistema, controlarlos y coordinarlos se vuelve más difícil. Afortunadamente, Kubernetes es una potente solución de orquestación de contenedores.

Este tutorial le enseña cómo aprovechar Azure Kubernetes Service (AKS) para la orquestación de contenedores en el entorno de nube de Azure.

### **Terminologías clave**

Antes de sumergirnos en la guía, definamos algunas terminologías esenciales:

1. Contenerización: el proceso de empaquetar el código de su aplicación con todos los archivos y bibliotecas juntos en un solo contenedor para garantizar la coherencia y la portabilidad en diferentes entornos.
2. Kubernetes: una plataforma de orquestación de contenedores de código abierto que automatiza la implementación, el escalamiento y la gestión de aplicaciones en contenedores.
3. Azure Kubernetes Service (AKS): un servicio de Kubernetes administrado proporcionado por Microsoft Azure que simplifica la implementación y la administración de clústeres de Kubernetes en el entorno de nube de Azure.
4. Pod: la unidad implementable más pequeña en Kubernetes, que representa uno o más contenedores que se ejecutan juntos en un nodo.
5. Implementación: un recurso de Kubernetes que define cuántas réplicas de un pod deben ejecutarse y la plantilla para crearlas.
6. Nodo: una máquina virtual o física en un clúster de Kubernetes en el que se implementan contenedores.
7. kubectl: La herramienta de línea de comandos utilizada para interactuar con clústeres de Kubernetes.


## **Cómo usar Azure Kubernetes Service (AKS)**

### **Crear un grupo de recursos de Azure**

Para comenzar, inicie sesión en [Azure Portal](#), haga clic en "Crear un recurso" y busque "Grupo de recursos". Haga clic en "Grupo de recursos" y luego en "Crear".

# Resource group

Microsoft



## Resource group

Microsoft | Azure Service

★ 4.5 (16 ratings)

Plan

Resource group

▼

Create

[Add to Favorites](#)


Después de eso, completemos la información requerida, como el nombre del grupo de recursos y la región, luego haga clic en "Revisar + crear" y luego en "Crear" para crear el grupo de recursos.

✓ Resource group created

X

Creating resource group 'Tutorial' in subscription 'Azure for Students' succeeded.

Go to resource group

 Pin to dashboard


### Crear un clúster de AKS

En el portal de Azure, haga clic en "Crear" y busque "Azure Kubernetes Service". Ahora hagamos clic en "Azure Kubernetes Service" y luego en "Crear".

[Home](#) > [Tutorial](#) > [Marketplace](#) >

## Azure Kubernetes Service (AKS) ...

Microsoft




### Azure Kubernetes Service (AKS) [Add to Favorites](#)

Microsoft | Azure Service

★ 4.0 (424 ratings)

Plan

Azure Kubernetes Service (AKS) 

Create

Complete la información requerida, como el nombre del clúster AKS, el grupo de recursos (use el creado en el Paso 1) y la región.


### Detalles del clúster

Primero, elijamos nuestro plan y el recurso que creamos anteriormente:

[Home](#) > [Azure Kubernetes Service \(AKS\)](#) >


## Create Kubernetes cluster ...


[Basics](#) [Node pools](#) [Access](#) [Networking](#) [Integrations](#) [Advanced](#) [Tags](#) [Review + create](#)


Azure Kubernetes Service (AKS) manages your hosted Kubernetes environment, making it quick and easy to deploy and manage containerized applications without container orchestration expertise. It also eliminates the burden of ongoing operations and maintenance by provisioning, upgrading, and scaling resources on demand, without taking your applications offline. [Learn more](#) 


### Project details

Select a subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription \* 

Azure for Students 

Resource group \* 

Tutorial 

[Create new](#)















Complete the required information, such as the AKS cluster name, the resource group (use the one created in Step 1) and the region.

Asegúrese de que la configuración preestablecida sea Estándar (\$\$). Para obtener más información sobre las configuraciones preestablecidas, consulte la configuración del clúster en Azure Portal. Escriba un nombre de clúster de Kubernetes, como AKScluster.

Seleccione una región para el clúster AKS y deje el valor predeterminado seleccionado para la versión de Kubernetes.

## Create Kubernetes cluster ...







### Cluster details

Cluster preset configuration	Standard (\$\$) 
To quickly customize your Kubernetes cluster, choose one of the preset configurations above. You can modify these configurations at any time. <a href="#">Learn more and compare presets</a>	
Kubernetes cluster name * 	AKScluster 
Region * 	(Africa) South Africa North 
Availability zones 	Zones 1,2,3 
 High availability is recommended for standard configuration.	
AKS pricing tier 	Standard 
Kubernetes version * 	1.25.6 (default) 
Automatic upgrade 	Enabled with patch (recommended) 

Deje los valores predeterminados tal como están en el "grupo de nodos principal":

### Primary node pool

The number and size of nodes in the primary node pool in your cluster. For production workloads, at least 3 nodes are recommended for resiliency. For development or test workloads, only one node is required. If you would like to add additional node pools or to see additional configuration options for this node pool, go to the 'Node pools' tab above. You will be able to add additional node pools after creating your cluster. [Learn more about node pools in Azure Kubernetes Service](#)


Node size * 	<b>Standard DS2 v2</b>  Standard DS2_v2 is recommended for standard configuration. <a href="#">Change size</a>
Scale method * 	<input type="radio"/> Manual <input checked="" type="radio"/> Autoscale  Autoscaling is recommended for standard configuration.
Node count range * 	<input type="text" value="1"/>  <input type="text" value="5"/>

En las siguientes partes, deje las opciones predeterminadas y luego seleccione **Siguiente: Revisar + crear**:

Al acceder a la **Review + Create** pestaña, Azure valida los parámetros seleccionados. Si la validación es correcta, podemos establecer el clúster de AKS haciendo clic en **create**. Si la validación falla, indica qué parámetros deben modificarse.

[Home](#) > [Azure Kubernetes Service \(AKS\)](#) >

## Create Kubernetes cluster ...

 Validation passed

Basics

Node pools

Access

Networking

Integrations

Advanced

Tags

**Review + create**

**Basics**

Subscription	Azure for Students
Resource group	Tutorial
Region	South Africa North
Kubernetes cluster name	AKScluster
Kubernetes version	1.25.6
Automatic upgrade	Patch

**Node pools**

Node pools	1
Enable virtual nodes	Disabled

< Previous

Next >

Create

[Download a template for automation](#)

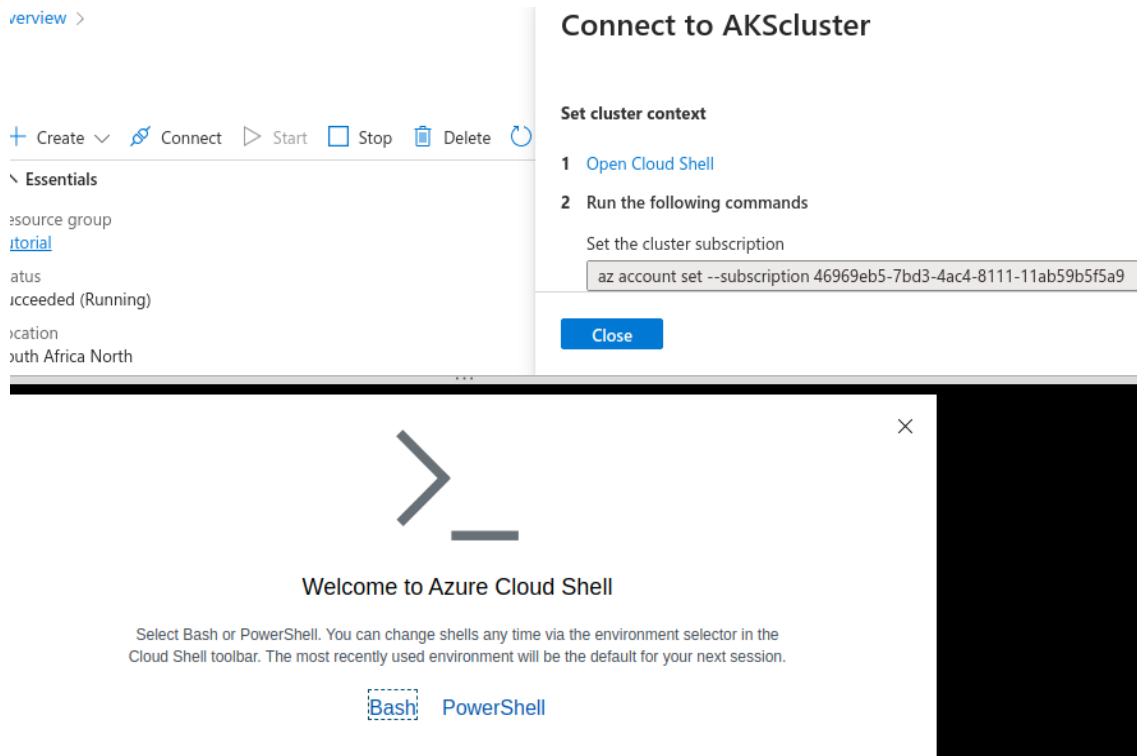
El clúster de AKS se crea en cuestión de minutos. Una vez finalizada la implementación, podemos acceder a nuestro recurso seleccionando "Ir al recurso" o seleccionando el recurso de AKS en el grupo de recursos del clúster de AKS.

## Configurar kubectl

Una vez creado correctamente el clúster AKS, debemos hacer clic en "Ir al recurso".

En la página de descripción general del clúster AKS, haga clic en "Conectar" y luego en "Abrir en Cloud Shell".

Azure Cloud Shell se abrirá en la parte inferior del portal. Si se le solicita, seleccione "Bash" como tipo de shell. Luego, seleccione "Crear almacenamiento".



Ejecute el siguiente comando en Cloud Shell para configurar kubectl para conectarse a nuestro clúster AKS:

```
az aks get-credentials --resource-group <your_resource_group_name> --name  
<your_aks_cluster_name>
```

En nuestro caso, el nombre de nuestro grupo de recursos es Tutorialy nuestro clúster AKS es AKScluster:

```
az aks get-credentials --resource-group Tutorial --name AKScluster
```

Esto producirá el siguiente resultado:

```
Merged "AKScluster" as current context in /home/valentine/.kube/config
```

Para verificar la conexión a nuestro clúster, utilice kubectl getpara devolver una lista de los nodos del clúster como se muestra a continuación:

NAME	STATUS	ROLES	AGE	VERSION
aks-agentpool-22140002-vmss000000	Ready	agent	16m	v1.25.6

## Implementar y administrar la aplicación

Ahora que kubectl está configurado para conectarse a nuestro clúster AKS, podemos implementar y administrar aplicaciones en él.

Cree un archivo de manifiesto de implementación de Kubernetes en Cloud Shell. Para ello, abra el editor de texto nano y cree el archivo "nginx-deployment.yaml". Use el siguiente comando:

```
nano nginx-deployment.yaml
```

Se abrirá el editor nano y podremos empezar a escribir el contenido del manifiesto de implementación. Para implementar un servidor web NGINX simple, use el siguiente contenido YAML:

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
spec:
  replicas: 3
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - name: nginx
          image: nginx:latest
          ports:
            - containerPort: 80
```

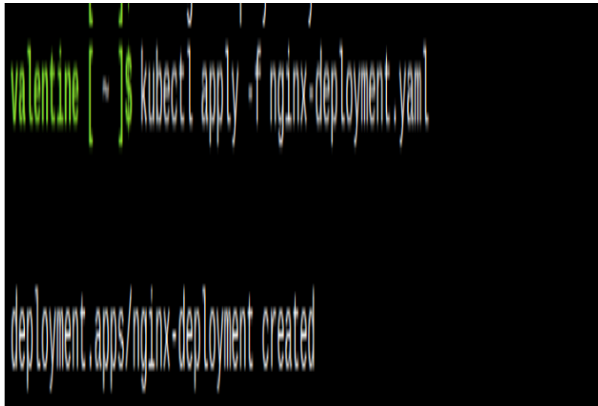
Después de copiar y pegar el código anterior, pulsaremos Ctrl + X para salir nano. Se nos pedirá que guardemos los cambios. Pulsaremos Y para confirmar y, cuando se nos pida que guardemos el archivo, pulsaremos Enter para guardarlo con el nombre predeterminado (nginx-deployment.yaml).

Ahora hemos creado el nginx-deployment.yaml archivo, que contiene la especificación YAML para implementar un servidor web NGINX simple con tres réplicas.

Implementemos la aplicación NGINX en nuestro clúster AKS usando el siguiente comando:

```
kubectl apply -f nginx-deployment.yaml
```

Este será el resultado:



### Escalar la aplicación

Podemos escalar fácilmente nuestra aplicación implementada según la demanda.

Para escalar la implementación de NGINX a cinco réplicas, usemos el siguiente comando:

```
kubectl scale deployment nginx-deployment --replicas=5
```

### Probar la aplicación

Ahora verifiquemos el estado de nuestra implementación usando el siguiente comando:

```
kubectl get deployment nginx-deployment --watch
```

El comando anterior mostrará el estado de la implementación, incluida la cantidad de réplicas deseadas, la cantidad de réplicas disponibles y el estado actual de la implementación como se muestra a continuación:

NAME	READY	UP-TO-DATE	AVAILABLE	AGE
nginx-deployment	3/3	3	3	13m

### Implementación como servicio

Para exponer la implementación como un servicio y acceder a ella externamente, necesitaremos crear un manifiesto de servicio de Kubernetes y aplicarlo usando `kubectl apply -f service-manifest.yaml`.

Por ejemplo, podemos crear un archivo de manifiesto de servicio llamado `nginx-service.yaml` agregar el siguiente contenido:

```
apiVersion: v1
kind: Service
metadata:
  name: nginx-service
spec:
  selector:
    app: nginx
  ports:
```



- protocol: TCP

port: 80

targetPort: 80

type: LoadBalancer

Guardemos el archivo y luego apliquémoslo usando el siguiente comando:

```
kubectl apply -f nginx-service.yaml
```

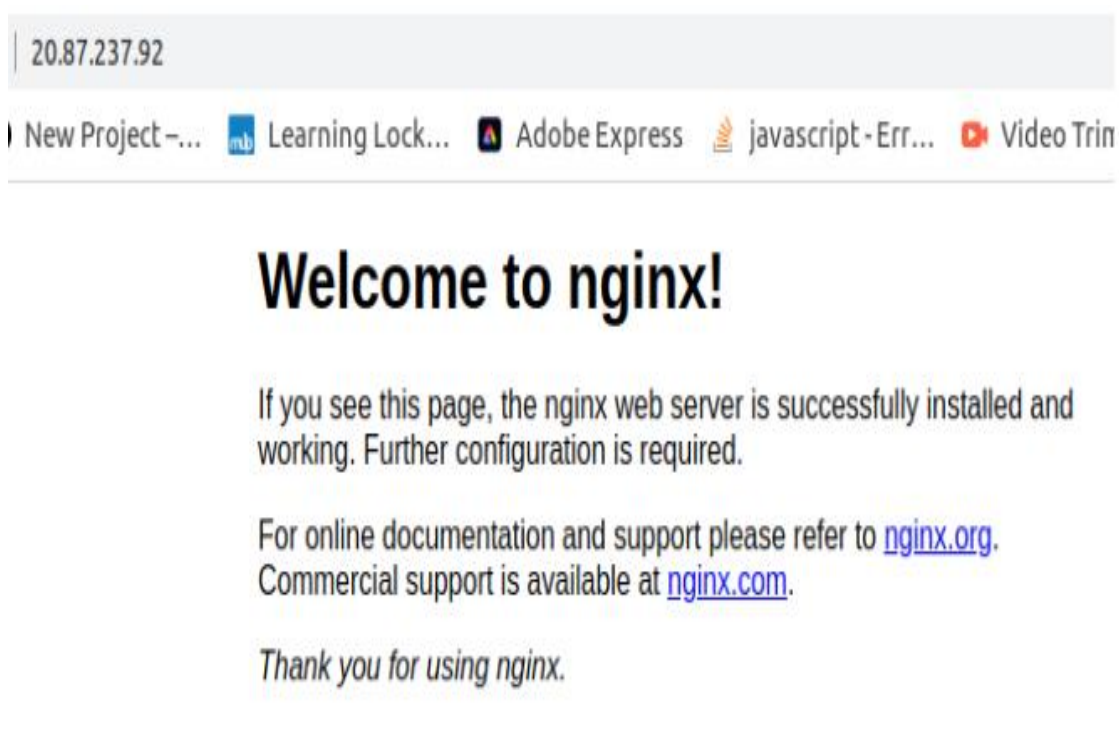
Esto creará un servicio de Kubernetes llamado nginx-service que se asigna a los pods implementados por nginx-deployment. El servicio será accesible externamente mediante un [nombre del servicio LoadBalancer] y podremos comprobar su estado usando:

```
kubectl get service nginx-service --watch
```

Deberíamos poder acceder a nuestro servidor web NGINX utilizando la dirección IP proporcionada.

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE
nginx-service	LoadBalancer	10.0.61.228	20.87.237.92	80:32415/TCP	93s

Visitemos la IP externa:



Implementamos con éxito una aplicación web en nuestro clúster de AKS, la expusimos a internet y gestionamos su escalado sin problemas con Azure Kubernetes Service.

**Monitorear el clúster**

AKS se integra perfectamente con Azure Monitor, lo que nos permite supervisar el rendimiento y el estado de nuestro clúster y aplicaciones de Kubernetes.

En Azure Portal, diríjase a su clúster de AKS y haga clic en "Supervisión". Allí podrá explorar las distintas opciones de supervisión, como el estado del clúster, el rendimiento y el diagnóstico.