ICE Server on AKS

Product guide for prerelease

Document version 1841, produced on Friday, September 06, 2024.

main 90adc8bf40040649230176bbdd465f6261a2d8e0

Contents

List of Tables

1 Introduction

This guide provides instructions for running an ICE Server on Microsoft Azure Kubernetes Service (AKS).

> [!NOTE] If you are using Rancher on AKS, then be aware that the instructions below may require modification. Consult your Rancher Administrator for guidance.

2 Configuring AKS

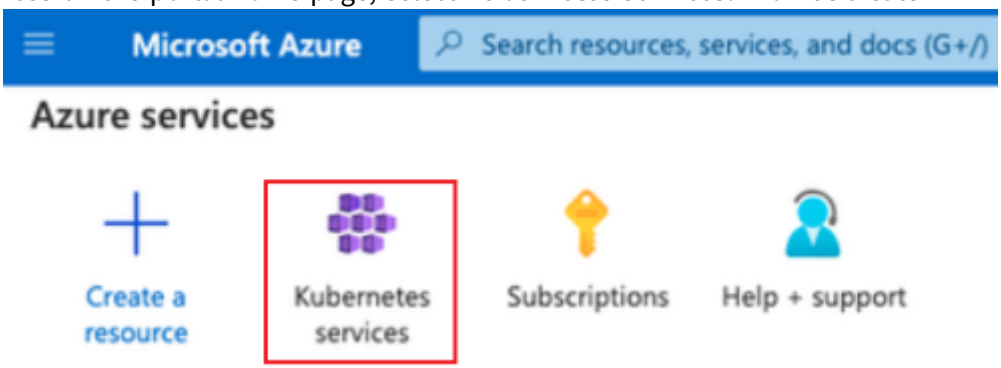
Use the following instructions to prepare AKS for operation with an ICE Server.

Once initiated, the Kubernetes services wizard moves through a series of tabbed pages. The instructions below only cover those changes required to run an ICE Server. Fields that are not covered in the instructions may be left at their default values, or modified as necessary.

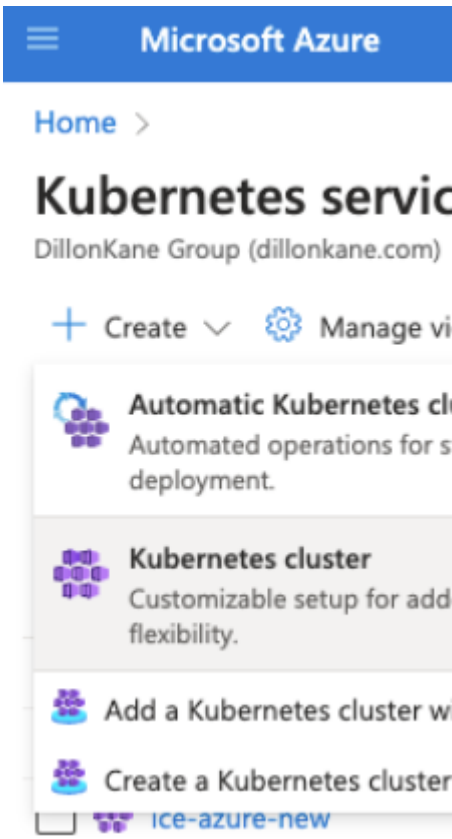
> [!NOTE]
>
> The following two points are covered in the procedures below, and called out here for reference.
>
> - **Kubernetes Version:** Only Kubernetes versions **1.28.x** or **1.29.x** are supported for ICE Server versions **3.5.x** and **3.6.x**
> - **Resource Group:** The AKS will provide best results in its own resource group.

To configure AKS

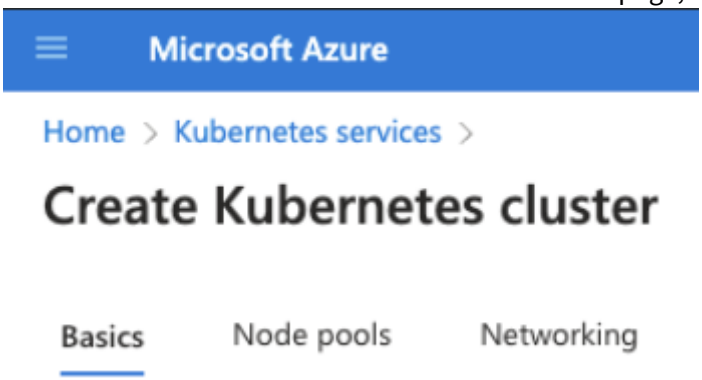
1. Sign in to the Azure portal.
2. Under **Azure services** on the portal home page, select **Kubernetes services.** Do **not** create Azure hosted VMs.



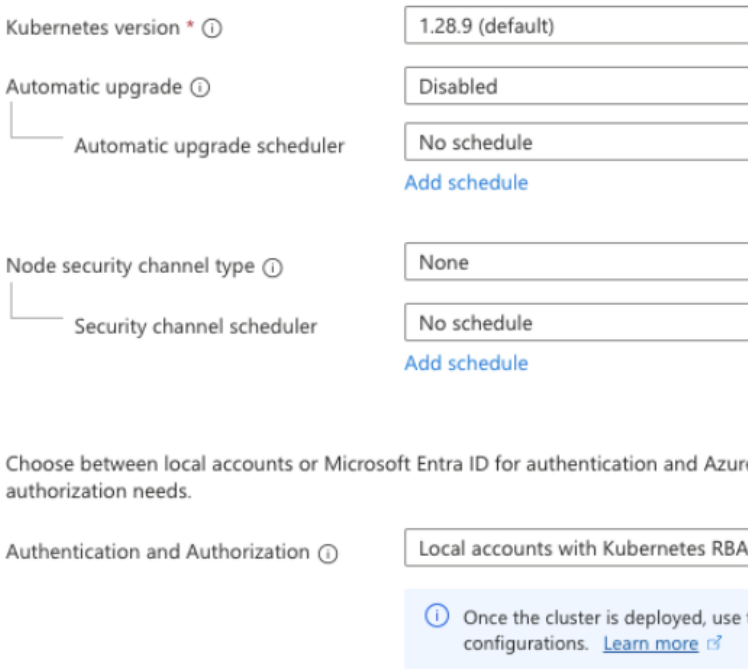
3. The Kubernetes services page opens.



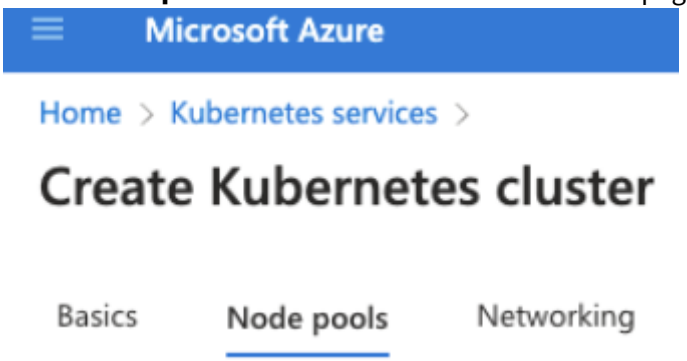
4. Click **CREATE,** select **Kubernetes cluster,** and proceed to the next screen.
5. In the **Basics** tab of the **Kubernetes services** page, configure the following properties:



- **RESOURCE GROUP** Use a dedicated resource group exclusively for your deployment
- **CLUSTER PRESET CONFIGURATION** For PRODUCTION deployments, choose `production standard` For POC / DEV / TEST, you may choose any setting
- **REGION** For PRODUCTION deployment, choose a region with multiple availability zones For POC / DEV / TEST, choose any
- **AVAILABILITY ZONES** For PRODUCTION deployment, choose all three regions For POC / DEV / TEST, choose any



- **KUBERNETES VERSION** Select **1.28.x** or **1.29.x**
- **AUTOMATIC UPGRADE** Select `Disabled` + `no schedule`
- **NODE SECURITY CHANNEL TYPE** Select `None` + `no schedule`
- In the **Node pools** tab of the **Kubernetes services** page, configure the following properties:



NODE SIZE For production deployment, equivalent of D8s_v5 or higher is recommended for each node.
Each VM requires the following:

- 8 cores minimum
- 32 Gb memory minimum
- 500Gb disk minimum / 1Tb recommended.
- I/O optimized.

General guideline for VM type vs number of supported users: **Standard_D8s_v5 x 3** ~5000 users **Standard_D8s_v4 x 3** ~3500 users **Standard_D8s_v3 x 3** ~2000 users **Standard_D8s_v3 x 1** ~500 users

- **NODE POOL**

  You will require a minimum of 3 nodes. Ensure your Azure subscription has enough capacity for 3 nodes by verifying current quota and usage using Azure's cloud powershell:

```
Get-AzVMUsage -Location "YOUR REGION"
```

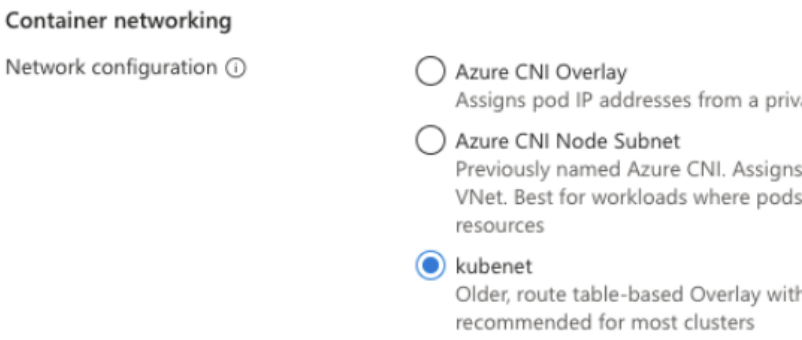
  If defined in the node pool, your nodes are distributed across your Availability Zones. Verify node distribution by running the following command:

```
kubectl get nodes \
  -o custom-columns=NAME:'{.metadata.name}',REGION:'{.metadata.labels.
     topology\.kubernetes\.io/region}',ZONE:'{metadata.labels.
     topology\.kubernetes\.io/zone}'
```

  Your nodes should appear spread across the availability zones:

```
NAME                                 REGION    ZONE
aks-agentpool-77403594-vmss000000    westus3   westus3-1
aks-agentpool-77403594-vmss000001    westus3   westus3-2
aks-agentpool-77403594-vmss000002    westus3   westus3-3
```

- Click **Next : Node Pools** and review the configuration.
- Click **Next: Access** and select **RBAC.**
- Click **Next: Networking.**



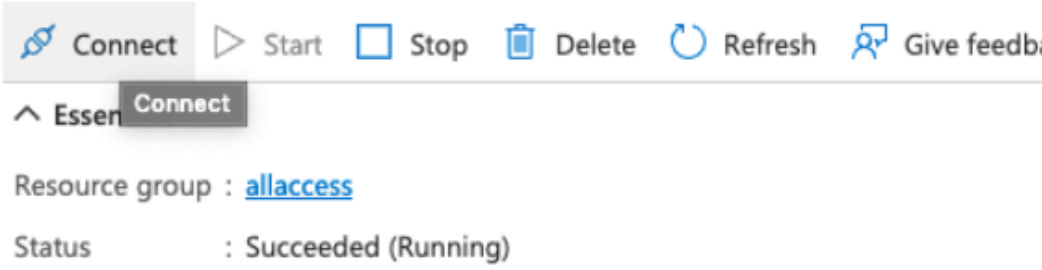
- In **Container networking,** select **kubenet.**
- When you are done with the wizard, or you have completed modifications to the AKS, click then **Review and create** button and complete the creation of the AKS.

3 Installing ICE Server

Use the following instructions to install an ICE Server on the AKS created in the previous section.

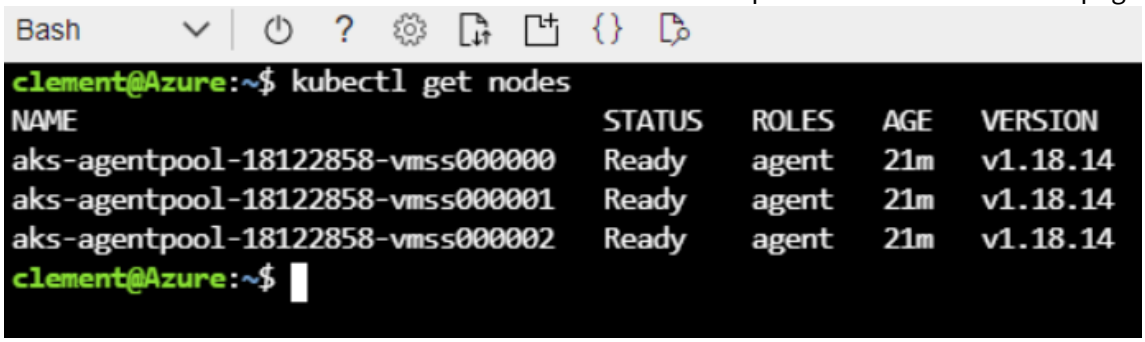
3.1 Connect to the Cluster using AZ Shell

1. From the **Home** page, navigate to the **Kubernetes Service** page, and select your cluster.



2. Click the **Connect** tab.
   A window appears displaying the two commands required to connect your cluster.
3. In the top right of the title bar, click the **powershell** icon.

   A CLI shell opens at the bottom of the page.



4. In the CLI shell, enter the following commands as shown:

```
az account set --subscription SUBSCRIPTION_ID
az aks get-credentials --resource-group RESOURCE_GROUP --name AKS_NAME
```

> [!NOTE]
>
> Helm and kubectl are already included in the AZ shell. There is no need to install any additional

> binary.

> [!TIP]
>
> You can reduce costs by disabling the Container Insight using the following command:

```
az aks disable-addons -a monitoring -n MyExistingManagedCluster -g
   MyExistingManagedClusterRG
```

3.2 Get KUBECONFIG

You may get the KUBECONFIG using the following command in the AZ shell.

```
scp ~/.kube/config YOUR_REMOTE_ACCOUNT@YOUR_REMOTE_SERVER:.kube/
   azure_config
```

Then export `KUBECONFIG=~/.kube/azure_config` to manage your cluster remotely with `helm`, `kubectl`.

3.3 Cassandra Nodes (Multi-Node only)

The default number of nodes in the Cassandra cluster is one (1). You must define the number of Cassandra nodes and the replication factor during the initial installation of an ICE Server. Ensure you define the correct number during install.

`nCassandranodes` defines how many Cassandra nodes are in your ICE Server installation:

- The default is 1 (single node)
- `nCassandranodes` = 1 does not offer any Cassandra data redundancy
- `nCassandranodes` must be less than or equal to # of Azure nodes
- replicationFactor defines how many full copies of the data you have in the cluster
- Each Cassandra node would contain `replicationFactor/nCassandranodes` of the data. **Example:** If `nCassandranodes` = 3 and `replicationFactor`= 2, then each Cassandra node has roughly 66% of your data. You can afford to lose one (1) node without compromising your database data availability.

The following table illustrates some basic combinations to help you understand the significance of the configurations:

# Azure nodes	**# Cassandra nodes** (nCassandranodes)	**Replication Factor** (replicationFactor)	**Cassandra redundancy**
1	1	1	none
2	1	1	none
2	2	2	Could lose 1 node
3	1	1	none
3	2	1	none
3	2	2	Could lose 1 node
3	3	1	none
3	3	2	Could lose 1 node
3	3	3	Could lose 2 nodes

To specify `nCassandranodes` and `replicationFactor` during initial ICE Server installation on a three nodes AKS cluster with 2 copies of data, add the following flags:

```
--set charts.instantConnectEnterprise.values.config.cassandra.
   nCassandranodes=3
--set charts.instantConnectEnterprise.values.config.cassandra.
   replicationFactor=2
```

3.4 ICE Server Installation

Refer to the **ICE Server Installation Guide** and the **ICE Server Administration Guide** for your current version of ICE Server.

4 Post Installation Modifications

The instructions in this section describe optional modifications performed after the installation of an ICE Server on AKS.

4.1 Rancher on AKS

The user ID must have admin `clusterrolebinding` to the default name space. Ideally, the same user should also have member role in Rancher.

Verify your ID has proper permissions by running the following command:

```
kubectl get all
```

An Admin ID will have complete access. None of the results should display as “forbidden.”

4.2 Exposing ICE Server Service

On AKS, you may create an external load balancer service to reach ICE Server’s Ingress. The following example describes how to perform this process.

Example: Create a file called `ice-azure-lb.yaml` with the following content, where `CLUSTERIP___ADDRESS____` is the static IP address associated with the cluster’s FQDN name that you have reserved in Azure.

If the cluster has no FQDN name (no https), or if you do not have any static IP reserved in Azure, you must skip the entire`____CLUSTER____IP___ADDRESS____` line.

```
apiVersion: v1
kind: Service
metadata:
  name: ice-azure-lb
  namespace: ice-release
spec:
  type: LoadBalancer
  loadBalancerIP: ____CLUSTER____IP___ADDRESS____
  selector:
    app.kubernetes.io/component: controller
    app.kubernetes.io/instance: ice-release-ice-ingress
    app.kubernetes.io/name: ingress-nginx
  ports:
  - name: tcp80
    port: 80
    targetPort: 80
  - name: tcp443
    port: 443
    targetPort: 443
  - name: tcp7443
    port: 7443
    targetPort: 7443
  - name: tcp8443
    port: 8443
    targetPort: 8443
```

Then create the service:

```
kubectl create -f ice-azure-lb.yaml
```

The load balancer may stay in pending state for a few minutes.

Record the EXTERNAL-IP address assigned automatically by AKS:

```
$ kubectl -n ice-release get service ice-azure-lb  -o wide
NAME           TYPE           CLUSTER-IP     EXTERNAL-IP    PORT(S)
    ....
ice-azure-lb   LoadBalancer   10.0.233.132   52.162.2.255   80:31782/TCP
    ,....
```