Service EPG Configuration Examples

For more information on service EPGs, see:

- Cloud Service Endpoint Groups
- Creating a Service EPG Using the Cisco Cloud APIC GUI
- Creating a Service EPG Using the REST API

The following sections provide configuration examples for service EPGs.

Azure Kubernetes Services (AKS) Service EPG Configuration Example

This section provides procedures for configuring an example service EPG with the following settings:

- **Service Type**: Azure Kubernetes Services (AKS)
  - Azure Kubernetes Services (AKS) requires access to other services.
  - Cisco Cloud APIC automates the programming of the rules listed here:

    https://docs.microsoft.com/en-us/azure/aks/limit-egress-traffic#required-outbound-network-rules-and-fqdns-for-aks-clusters
- **Deployment type**: Cloud Native Managed. In this type of deployment, the service is instantiated in your VNet or subnet (created through the Cisco Cloud APIC). For example, an Azure Kubernetes Services (AKS) service could be deployed in a subnet that is managed by the Cisco Cloud APIC.
- **Access type**: Private

The procedures to configure this example service EPG for AKS are provided in the following sections.

Creating a Subnet in the Cloud Context Profile

These procedures describe how to create a subnet in a cloud context profile to be used by the Azure Kubernetes Services (AKS) service EPG. You will be making configurations through the Cisco Cloud APIC GUI in these procedures.

Before you begin

- In one browser window, log into your Cisco Cloud APIC GUI.
- In another browser window, log into your Azure account for the Cisco Cloud APIC infra tenant and go to the Azure management portal:

  https://portal.azure.com/#home

Step 1 In the Cisco Cloud APIC GUI, click the **Intent** icon.

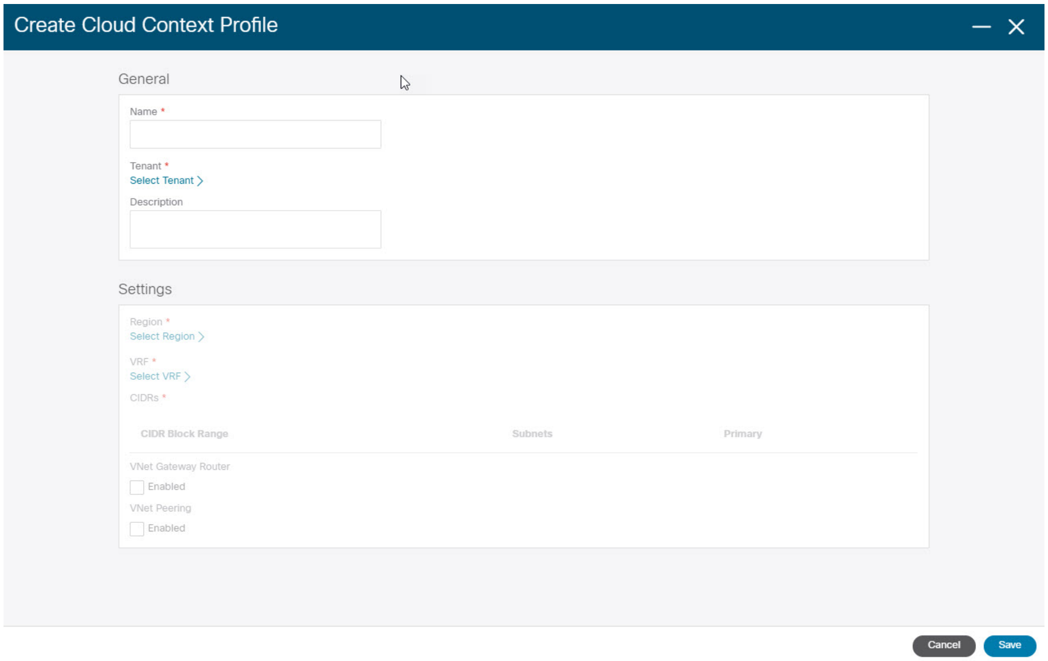
The **Intent** menu appears.

Step 2 Click the drop-down arrow below the **Intent** search box and choose **Application Management**.

A list of **Application Management** options appear in the **Intent** menu.

Step 3 From the **Application Management** list in the **Intent** menu, click **Create Cloud Context Profile**.

The **Create Cloud Context Profile** window appears.



Step 4 Enter the following information in the **Create Cloud Context Profile** window.

- **Name**: Enter the name of the cloud context profile. For example, **ct_ctxprofile_eastus**.
- **Tenant**: Click **Select Tenant**, choose a tenant for the cloud context profile for this use case, then click **Select**.
- **Region**: Click **Select Region**, choose the region (for example, **eastus**), then click **Select**.
- **VRF**: Click **Select VRF**, select the appropriate VRF, then click **Select**.
- **Add CIDR**: Enter the CIDR information.

  a. Click **Add CIDR**.

b. Enter the address in the **CIDR Block Range** field.

   For example, `30.1.0.0/16`.

c. Uncheck (disable) the **Primary** check box.

d. Click **Add Subnet** and enter the subnet address in the **Address** field.

   For example, `30.1.0.0/17`. Note that AKS cluster requires 338 addresses.

e. Click **Add**.

- **VNet Gateway Router**: Leave the box unchecked (unselected) for this field.
- **VNet Peering**: Check this box to enable VNet peering.

Step 5 Click **Save** when finished.

What to do next

Go to Creating the Cloud Service EPG for AKS, on page 3.

Creating the Cloud Service EPG for AKS

These procedures describe how to create the cloud service EPG with the Azure Kubernetes Services (AKS) service type. You will be making configurations through the Cisco Cloud APIC GUI in these procedures.

Before you begin

Complete the procedures in Creating a Subnet in the Cloud Context Profile, on page 1 before proceeding with these procedures.

Step 1 In the Cisco Cloud APIC GUI, click the **Intent** icon.

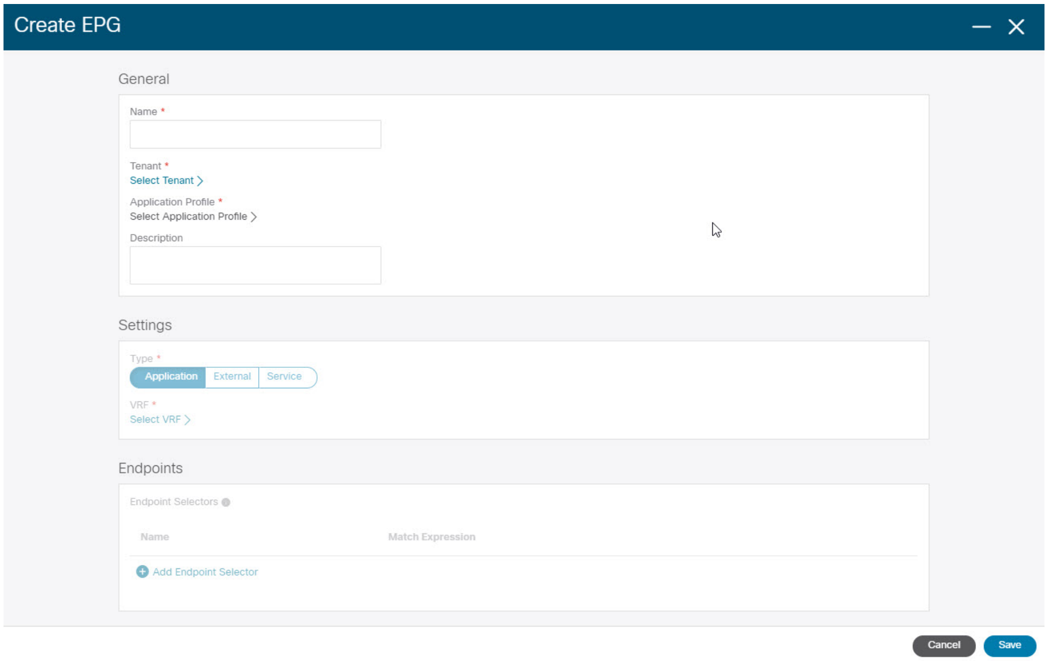
The **Intent** menu appears.

Step 2 Click the drop-down arrow below the **Intent** search box and choose **Application Management**.

A list of **Application Management** options appear in the **Intent** menu.

Step 3 From the **Application Management** list in the **Intent** menu, click **Create EPG**.

The **Create EPG** window appears.



Create EPG

General

Name *

Tenant *
Select Tenant >

Application Profile *
Select Application Profile >

Description

Settings

Type *
Application | External | Service

VRF *
Select VRF >

Endpoints

Endpoint Selectors

Name	Match Expression

Add Endpoint Selector

Step 4 Enter the following information in the **Create EPG** window.

- **Name**: Enter the name of the cloud service EPG. For example, **svc-Hub-AzureAKS**.
- **Tenant**: Click **Select Tenant**, choose a tenant for the cloud service EPG for this use case, then click **Select**.
- **Application Profile**: Click **Select Application Profile**, choose the application profile, then click **Select**.
- **Type**: Choose **Service** as the EPG type.
- **VRF**: Click **Select VRF**, select the appropriate VRF, then click **Select**.
- **Service Type** : Choose the **Azure Kubernetes Services (AKS)** service type.
- **Deployment Type**: Choose the **Cloud Native Managed** deployment type.
- **Access Type**: Choose the **Private** access type.

Step 5 Click **Add Endpoint Selector**.

The **Add Endpoint Selector** window appears.

For this use case, we will be creating an endpoint selector where the IP address matches the subnet information configured in the previous step, `30.1.0.0/17`. Having the IP address in the endpoint selector match the subnet in the previous step allows the Cisco Cloud APIC to program the NSG to allow all of the required rules for this service type.

Step 6 In the **Add Endpoint Selector** window, enter a name in the **Name** field.

Step 7 Click the **Key** drop-down list to choose a key.

At this time, **IP** is the only option available as a key for this access type.

Step 8 Click the **Operator** drop-down list and choose **equals**.

Step 9 In the **Value** field, enter `30.1.0.0/17`, then click the check mark to validate the entry.

Step 10 Click **Add**.

Step 11 Click **Save** when finished.

What to do next

Go to Verifying the Outbound Security Rules, on page 5.

Verifying the Outbound Security Rules

These procedures describe how to verify that the necessary outbound security rules are getting configured correctly. The Cisco Cloud APIC configures all of the outbound security rules in Azure that are needed for AKS to be deployed in the Azure portal.

Before you begin

Complete the procedures in Creating the Cloud Service EPG for AKS, on page 3 before proceeding with these procedures.

Step 1 In the Azure portal, navigate to the network security group for the subnet that was automatically created:

a) Navigate to appropriate resource group.
b) Select the subnet that was used for the AKS service EPG.
c) Locate the necessary outbound security group.

Step 2 Locate the **Outbound security rules** area in the page and verify that the outbound security rules for the NSG are configured correctly.

For more information on the outbound security rules, see:

https://docs.microsoft.com/en-us/azure/aks/limit-egress-traffic

What to do next

Go to Creating a Kubernetes Service, on page 5.

Creating a Kubernetes Service

These procedures describes how to create a Kubernetes service. You will be making configurations through the Azure portal in these procedures.

Note The following procedure describes how to create a Kubernetes service through the Azure portal. An alternative method for creating a Kubernetes service is also provided in the Using Azure Kubernetes Service with Cisco Cloud APIC document.

Before you begin

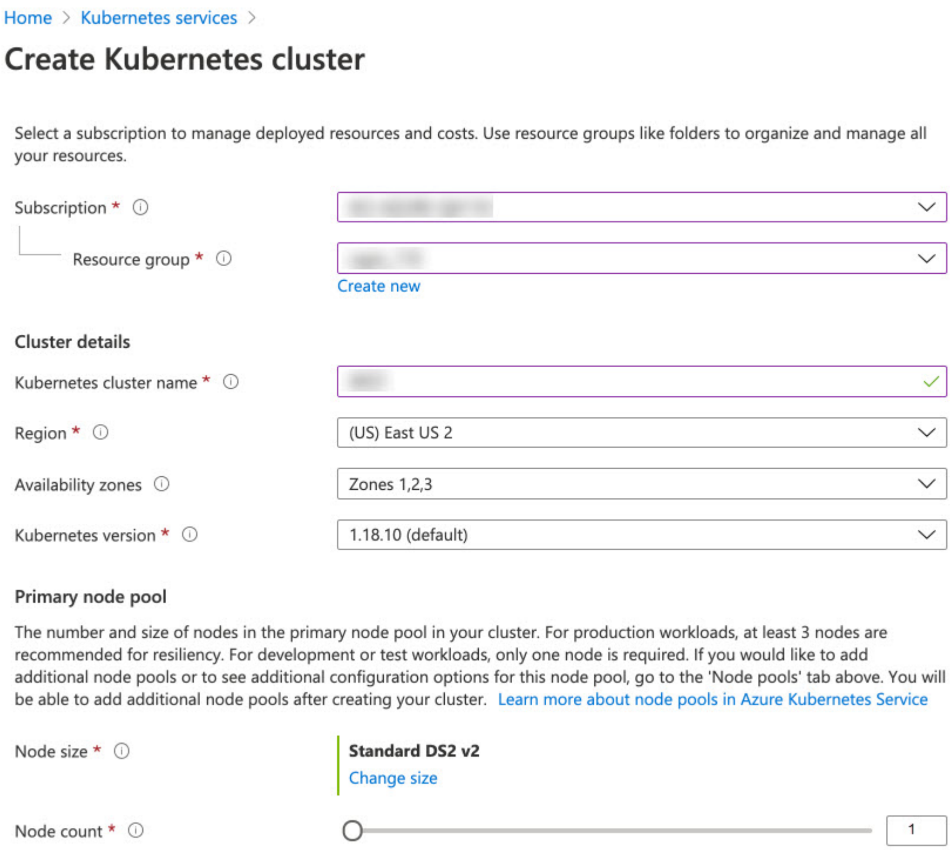
Complete the procedures in Verifying the Outbound Security Rules, on page 5 before proceeding with these procedures.

Step 1 In the Azure portal, search for the term `Kubernetes Service by Microsoft` and click on the search result.

The **Kubernetes Service** page appears.

Step 2 Click **Create** in the **Kubernetes Service** page.

The **Create Kubernetes cluster** page appears.



Home > Kubernetes services >

Create Kubernetes cluster

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

Subscription * ⓘ [redacted]

Resource group * ⓘ [redacted]

Create new

Cluster details

Kubernetes cluster name * ⓘ [redacted] ✓

Region * ⓘ (US) East US 2

Availability zones ⓘ Zones 1,2,3

Kubernetes version * ⓘ 1.18.10 (default)

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 * ⓘ **Standard DS2 v2**

Change size

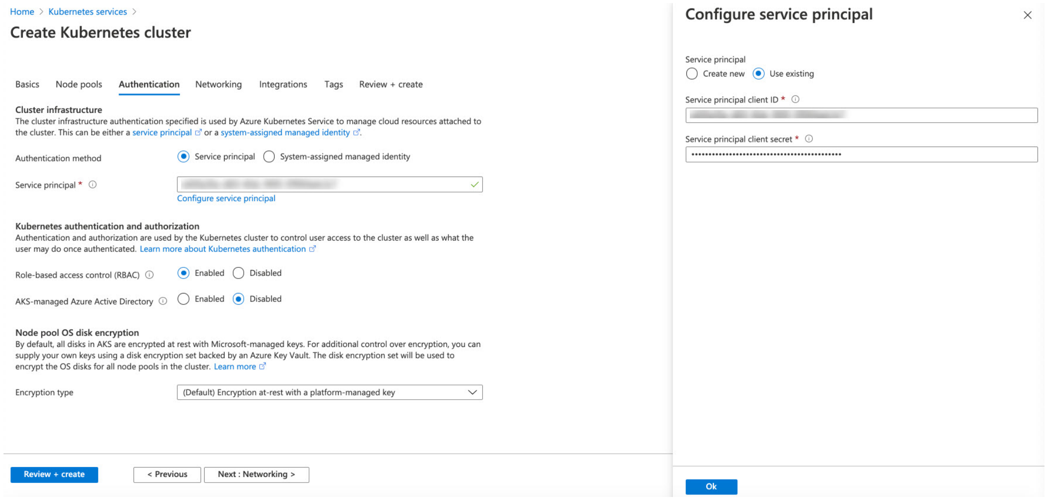
Node count * ⓘ 1

Step 3 In the **Basics** tab, configure the following areas:

- **Subscription**: Select the appropriate subscription.
- **Resource Group**: Select the appropriate resource group.
- **Kubernetes cluster name**: Enter a unique name for this Kubernetes cluster.
- **Region**: Select the appropriate region.
- **Kubernetes version**: Leave the default selection as-is.
- **Node size**: Leave the default selection as-is.

- **Node count**: Verify that the scroll bar is all the way to the left so that the entry for this field is `1`.

Step 4 Click **Next: Node pools**. Leave the default entries as-is and click **Next: Authentication** to advance to the **Authentication** tab.



Home > Kubernetes services >

Create Kubernetes cluster

Basics | Node pools | Authentication | Networking | Integrations | Tags | Review + create

Cluster infrastructure

The cluster infrastructure authentication specified is used by Azure Kubernetes Service to manage cloud resources attached to the cluster. This can be either a service principal or a system-assigned managed identity.

Authentication method ● Service principal ○ System-assigned managed identity

Service principal * [redacted] ✓

Configure service principal

Kubernetes authentication and authorization

Authentication and authorization are used by the Kubernetes cluster to control user access to the cluster as well as what the user may do once authenticated. Learn more about Kubernetes authentication

Role-based access control (RBAC) ● Enabled ○ Disabled

AKS-managed Azure Active Directory ○ Enabled ● Disabled

Node pool OS disk encryption

By default, all disks in AKS are encrypted at rest with Microsoft-managed keys. For additional control over encryption, you can supply your own keys using a disk encryption set backed by an Azure Key Vault. The disk encryption set will be used to encrypt the OS disks for all node pools in the cluster. Learn more

Encryption type (Default) Encryption at-rest with a platform-managed key

Configure service principal

Service principal

○ Create new ● Use existing

Service principal client ID * [redacted]

Service principal client secret * ••

Step 5 In the **Authentication** tab, configure the following areas:

- **Authentication method**: Choose **Service principal**.

  The **Service principal** field appears.

- **Service principal**: Click **Configure service principal**.

  In the **Configure service principal** window, configure the following areas:

  - **Service principal**: Choose either **Create new** or **Use existing**.

    If you choose **Use existing**, enter the following information for the existing service principal:

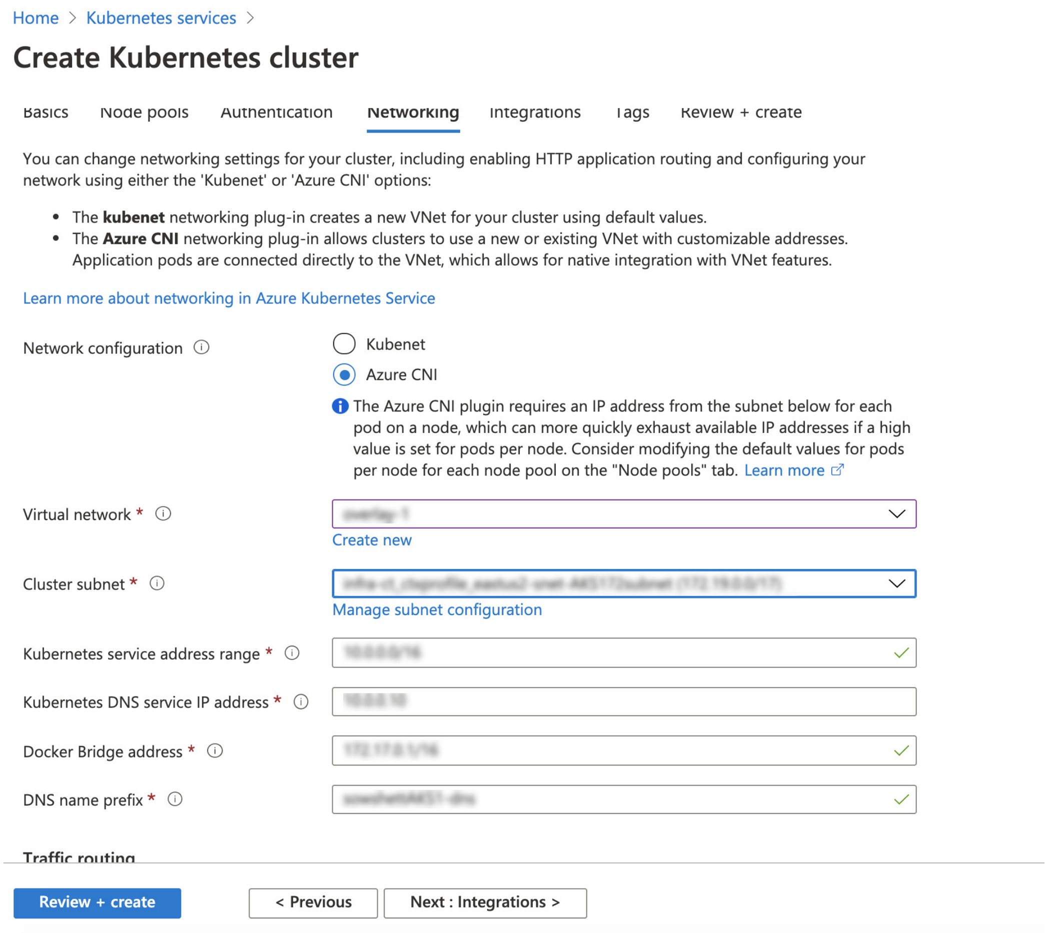
    - **Service principal client ID**
    - **Service principal client secret**

  Note Make a note of the entries that you enter in these two fields. You will be using the entries in these fields later in these procedures.

  Click **OK** to return to the **Authentication** tab in the **Create Kubernetes cluster** window.

- **Role-based acces control (RBAC)**: Choose **Enabled**.
- **AKS-managed Azure Active Directory**: Choose **Disabled**.
- **Encryption type**: Leave the default selection as-is.

Step 6 Click **Next: Networking** to advance to the **Networking** tab.



Home > Kubernetes services >

Create Kubernetes cluster

Basics | Node pools | Authentication | Networking | Integrations | Tags | Review + create

You can change networking settings for your cluster, including enabling HTTP application routing and configuring your network using either the 'Kubenet' or 'Azure CNI' options:

- The **kubenet** networking plug-in creates a new VNet for your cluster using default values.
- The **Azure CNI** networking plug-in allows clusters to use a new or existing VNet with customizable addresses. Application pods are connected directly to the VNet, which allows for native integration with VNet features.

Learn more about networking in Azure Kubernetes Service

Network configuration ⓘ
○ Kubenet
● Azure CNI

ⓘ The Azure CNI plugin requires an IP address from the subnet below for each pod on a node, which can more quickly exhaust available IP addresses if a high value is set for pods per node. Consider modifying the default values for pods per node for each node pool on the "Node pools" tab. Learn more

Virtual network * ⓘ [redacted]
Create new

Cluster subnet * ⓘ [redacted]
Manage subnet configuration

Kubernetes service address range * ⓘ [redacted] ✓

Kubernetes DNS service IP address * ⓘ [redacted]

Docker Bridge address * ⓘ [redacted] ✓

DNS name prefix * ⓘ [redacted] ✓

Traffic routing

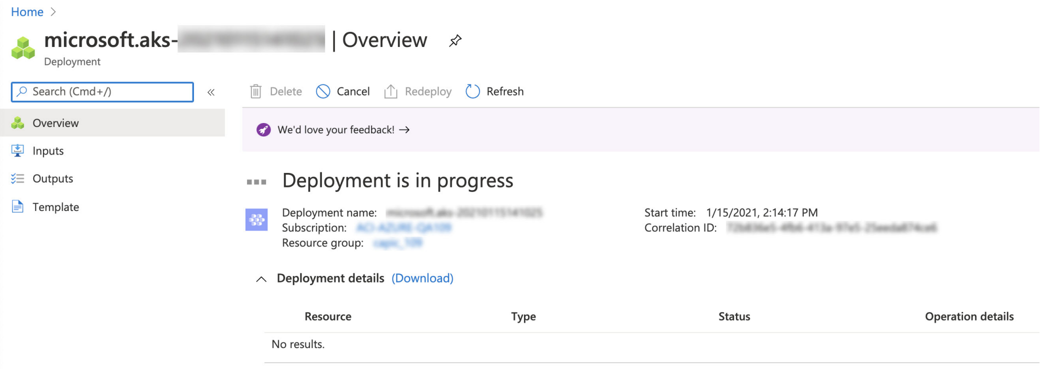
Step 7 In the **Networking** tab, configure the following areas:

- **Network configuration**: Choose **Azure CNI**.
- **Virtual network**: Choose the corresponding virtual network.
- **Cluster subnet**: Choose the Cisco Cloud APIC-managed subnet.
- **Kubernetes service address range**: Leave the default selection as-is, or change the entry, if necessary.
- **Kubernetes DNS service IP address**: Leave the default selection as-is, or change the entry, if necessary.
- **Docker Bridge address**: Leave the default selection as-is, or change the entry, if necessary.
- **DNS name prefix**: Leave the default selection as-is, or change the entry, if necessary.
- **Load balancer**: Standard
- **Enable HTTP application routing**: Leave the default selection as-is (not enabled), or change the entry, if necessary.
- **Enable Private cluster**: Leave the default selection as-is (not enabled), or change the entry, if necessary.

Step 8 Click **Next: Integration**, then **Next: Tags**, to advance through those screens without changing any of the default entries, then click **Next: Review+Create**.

Step 9 In the **Review+Create** window, click **Create**, then click **Create** again after the validations pass to create the Kubernetes cluster.

You will see the message `Deployment is in progress` and the Overview screen for the Kubernetes service will appear.



Home >

microsoft.aks-[redacted] | Overview

Deployment

Deployment is in progress

Deployment name: [redacted]
Subscription: [redacted]
Resource group: [redacted]

Start time: 1/15/2021, 2:14:17 PM
Correlation ID: [redacted]

Deployment details (Download)

Resource	Type	Status	Operation details
No results.			

Wait until the Kubernetes service is deployed successfully before proceeding (the amount of time it takes to deploy varies). Once this process is completed, the main AKS service will be in your original resource group. Azure will also create an additional resource group specifically for the Kubernetes service, with all of the agentpools VM scales set.

What to do next

Go to Verifying the New Kubernetes Service, on page 9.

Verifying the New Kubernetes Service

These procedures describe how to verify that the new Kubernetes service is in the resource group that was created specifically for the Kubernetes service.

Before you begin

Complete the procedures in Creating a Kubernetes Service, on page 5 before proceeding with these procedures.

Step 1 In the Azure portal, click on **Resource groups** in the left nav bar to navigate to the resource groups page.

Step 2 In the **Resource groups** page, locate the resource group that was created specifically for the Kubernetes service and click the link for that resource group.

The resource group created specifically for the Kubernetes service will have the following format:

```
MC_resourcegroupname_clustername_region
```

Where:

- *resourcegroupname* is the name of the resource group created specifically for the Kubernetes service (`MC_aks` is the resource group name used by default by Azure)
- *clustername* is the Kubernetes cluster name that you provided in Step 3, on page 6 in Creating a Kubernetes Service, on page 5.
- *region* is the region that you selected in Step 3, on page 6 in Creating a Kubernetes Service, on page 5.

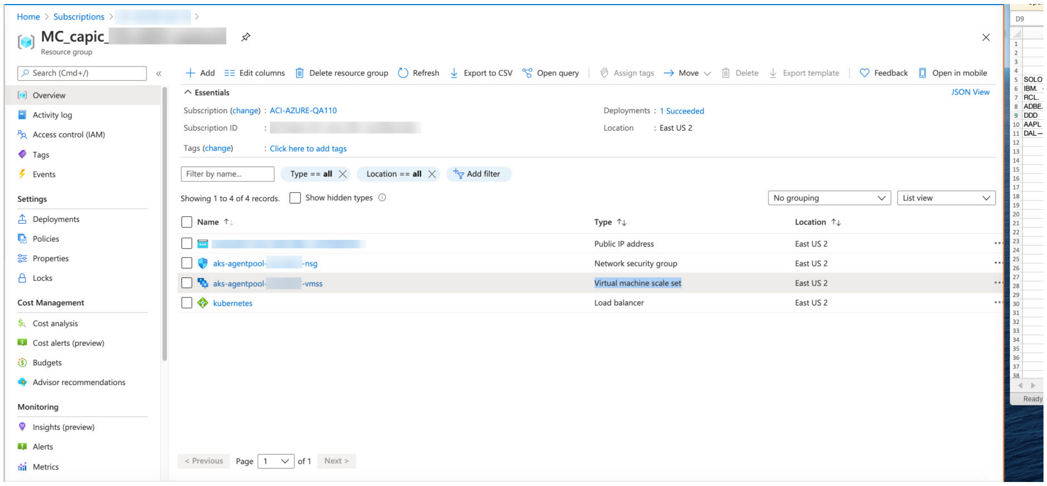
For example:

`MC_aks_acme-aks-cluster_centralus`

The Overview page for the Kubernetes service resource group appears.

Step 3 Locate the line for the **Virtual machine scale set** and click on that link.

This is where the AKS agent is running.



Essentials

Subscription (change) : ACI-AZURE-QA110

Subscription ID : [redacted]

Tags (change) : Click here to add tags

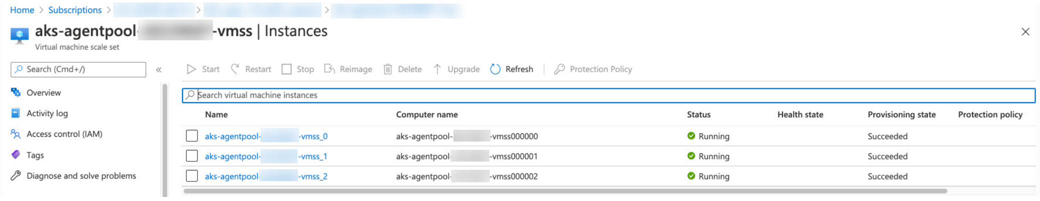
Deployments : 1 Succeeded

Location : East US 2

Showing 1 to 4 of 4 records. □ Show hidden types

Name	Type	Location
[redacted]	Public IP address	East US 2
aks-agentpool-[redacted]-nsg	Network security group	East US 2
aks-agentpool-[redacted]-vmss	Virtual machine scale set	East US 2
kubernetes	Load balancer	East US 2

Step 4 In the left nav bar, click on **Instances** to display the virtual machine instances for this Kubernetes service resource group.

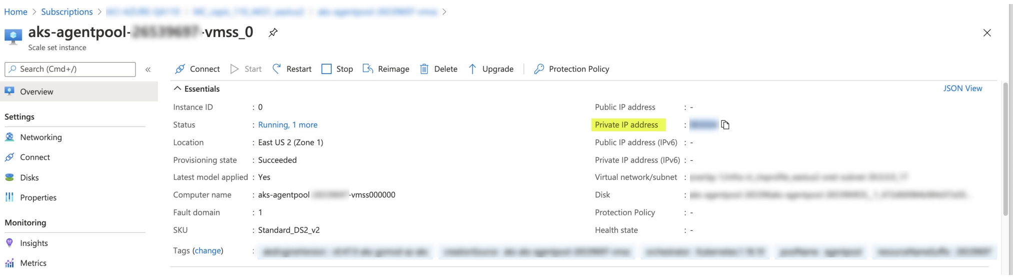


Home > Subscriptions > [redacted] > [redacted] > [redacted]
aks-agentpool[redacted]-vmss | Instances
Virtual machine scale set
Search (Cmd+/)
Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems
Start
Restart
Stop
Reimage
Delete
Upgrade
Refresh
Protection Policy
Search virtual machine instances

Name	Computer name	Status	Health state	Provisioning state	Protection policy
aks-agentpool-[redacted]-vmss_0	aks-agentpool-[redacted]-vmss000000	Running		Succeeded	
aks-agentpool-[redacted]-vmss_1	aks-agentpool[redacted]-vmss000001	Running		Succeeded	
aks-agentpool[redacted]-vmss_2	aks-agentpool-[redacted]-vmss000002	Running		Succeeded	

Step 5 Click on any of the three instances in this window, then verify that the IP address shown in the **Private IP address** field matches your hub subnet IP address.

All three instances shown in this window should have an IP address from the subnet that you selected in Step 7, on page 8 in Creating a Kubernetes Service, on page 5.



Essentials

Instance ID : 0

Status : Running, 1 more

Location : East US 2 (Zone 1)

Provisioning state : Succeeded

Latest model applied : Yes

Computer name : aks-agentpool[redacted]-vmss000000

Fault domain : 1

SKU : Standard_DS2_v2

Tags (change) : [redacted]

Public IP address : -

Private IP address : [redacted]

Public IP address (IPv6) : -

Private IP address (IPv6) : -

Virtual network/subnet : [redacted]

Disk : [redacted]

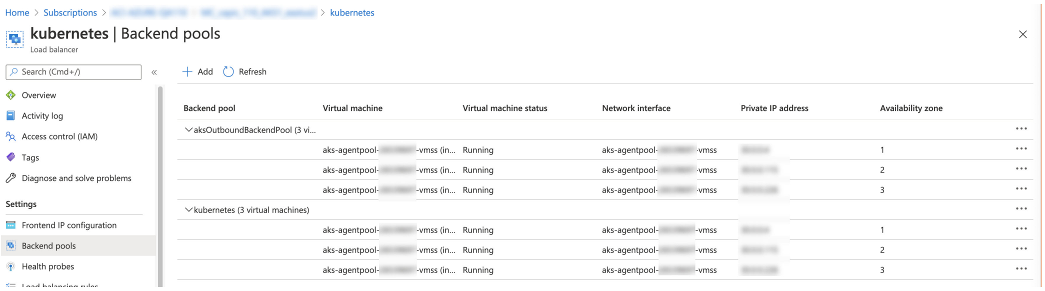
Protection Policy : -

Health state : -

Step 6 Navigate back to the Overview page for the Kubernetes service resource group, then locate the `kubernetes` entry, shown with `Load balancer` as the Type, and click that link.

The Overview page for the Kubernetes load balancer appears.

Step 7 In the left nav bar, click **Backend pools** to view the AKS agents.



Backend pool	Virtual machine	Virtual machine status	Network interface	Private IP address	Availability zone
aksOutboundBackendPool (3 vi...					
	aks-agentpool-[redacted]-vmss (in...	Running	aks-agentpool-[redacted]-vmss	[redacted]	1
	aks-agentpool-[redacted]-vmss (in...	Running	aks-agentpool-[redacted]-vmss	[redacted]	2
	aks-agentpool-[redacted]-vmss (in...	Running	aks-agentpool-[redacted]-vmss	[redacted]	3
kubernetes (3 virtual machines)					
	aks-agentpool-[redacted]-vmss (in...	Running	aks-agentpool-[redacted]-vmss	[redacted]	1
	aks-agentpool-[redacted]-vmss (in...	Running	aks-agentpool-[redacted]-vmss	[redacted]	2
	aks-agentpool-[redacted]-vmss (in...	Running	aks-agentpool-[redacted]-vmss	[redacted]	3

Step 8 If a virtual machine was created as part of the process of configuring the contract (for example, if a virtual machine was created for the consumer), and if you have AKS as the provider, verify that the rules were configured correctly.

a) In the Azure portal, navigate back to the infra resource group.

b) Choose **Group by type** for the records shown in the Overview page for the infra resource group.

c) Scroll down until you see the **Virtual machine** area and click on the virtual machine for the consumer in your contract.

The Overview window for that virtual machine appears.

d) In the left nav bar, under **Settings**, click **Networking**.

The Networking window for that virtual machine appears, with information on the inbound and outbound port rules.

e) Click on the **Outbound port rules** tab, then click on one of the outbound port rules listed in the table.

A window slides in from the right, displaying additional information on these outbound port rules. For example, the entry in the **Destination IP addresses/CIDR ranges** area provides information on the addresses that are associated with the AKS cluster.

What to do next

Go to Installing the Azure and AKS CLI, on page 11.

Installing the Azure and AKS CLI

These procedures describe how to install the Azure and AKS CLI.

Before you begin

Complete the procedures in Verifying the New Kubernetes Service, on page 9 before proceeding with these procedures.

Step 1 On the consumer VM that has internet access, install the Azure CLI.

For more information, see:

https://docs.microsoft.com/en-us/cli/azure/install-azure-cli-linux

For example, to install the Azure CLI in an Ubuntu Linux VM in Azure:

```
# curl -sL https://aka.ms/InstallAzureCliDeb | sudo bash
```

Step 2 Download and install **kubectl**, the Kubernetes command-line tool, and **kubelogin**, a client-go credential (exec) plugin implementing azure authentication:

```
# az aks install-cli
```

Step 3 Log in with the service principle information that you entered in Step 5, on page 7 in Creating a Kubernetes Service, on page 5 in these procedures:

```
# az login --service-principal --username <service_principal_client_id>
--password '<service_principal_client_secret>' --tenant <tenant_ID>
```

Where:

- *<service_principal_client_id>* is the entry from the **Service principal client ID** field in Step 5, on page 7 in Creating a Kubernetes Service, on page 5.
- *<service_principal_client_secret>* is the entry from the **Service principal client secret** field in Step 5, on page 7 in Creating a Kubernetes Service, on page 5.
- *<tenant_ID>* is the tenant associated with the service principal (the Azure Active Directory tenant ID). To locate the tenant ID information for this command:

  a. Sign in to the Azure portal.

  b. Select **Azure Active Directory**.

  c. Select **Properties**.

  d. Scroll down to the **Tenant ID** field. Your tenant ID is displayed in the box.

  For more information, see:

  https://docs.microsoft.com/en-us/azure/active-directory/fundamentals/active-directory-how-to-find-tenant

For example:

```
# az login --service-principal --username 12a3b456-7c89-1234-5de6-7f89012gh3i4
--password 'secretkey12341234!' --tenant 98765zy4-xwv-3ut2-1uts-rq0pon98m765
```

Step 4 Set a subscription to be the current active subscription.

```
# az account set --subscription <AKS_rg_subscription_ID>
```

Where *<AKS_rg_subscription_ID>* is the subscription ID of the resource group that Azure created for the Kubernetes service in Verifying the New Kubernetes Service, on page 9.

For example:

```
# az account set --subscription 56klm789n-o0p1-234q-5r6s-7t890123u4v5
```

Step 5 From the consumer VM, enter the following to log in and connect to AKS.

```
root@hub-vm:/home/capic# az aks get-credentials --resource-group <resource_group> --name
<AKS_cluster_name> --admin
```

Where:

- *<resource_group>* is the name of the infra resource group
- *<AKS_cluster_name>* is the name for the Kubernetes cluster that was entered in Step 3, on page 6 in Creating a Kubernetes Service, on page 5

For example:

```
root@hub-vm:/home/capic# az aks get-credentials --resource-group capic_infra_westus --name azureaksclus
 --admin
```

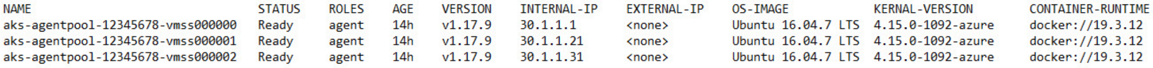
A message similar to the following appears:

```
Merged "azureaksclus-admin" as current context in /root/.kube/config
```

Step 6 Check the internal IP addresses of each of the nodes.

```
root@hub-vm:/home/capic# kubectl get nodes -o wide
```

Output similar to the following appears:



```
NAME                                STATUS   ROLES   AGE   VERSION   INTERNAL-IP   EXTERNAL-IP   OS-IMAGE           KERNAL-VERSION        CONTAINER-RUNTIME
aks-agentpool-12345678-vmss000000   Ready    agent   14h   v1.17.9   30.1.1.1      <none>        Ubuntu 16.04.7 LTS 4.15.0-1092-azure     docker://19.3.12
aks-agentpool-12345678-vmss000001   Ready    agent   14h   v1.17.9   30.1.1.21     <none>        Ubuntu 16.04.7 LTS 4.15.0-1092-azure     docker://19.3.12
aks-agentpool-12345678-vmss000002   Ready    agent   14h   v1.17.9   30.1.1.31     <none>        Ubuntu 16.04.7 LTS 4.15.0-1092-azure     docker://19.3.12
```

The IP addresses listed in the `INTERNAL-IP` column are in your hub subnet.

Note In the example output above, the entries in the `EXTERNAL-IP` column are shown as `<none>` because the **Access Type** was set to `Private` in Step 4, on page 4 in Creating the Cloud Service EPG for AKS, on page 3. IP addresses would be displayed in the `EXTERNAL-IP` column if the **Access Type** is set to `Public and Private`.

Step 7 (Optional) Assign an admin role to a new user, if necessary.

a) In the Azure portal, navigate back to the infra resource group.

b) In the records area in the page, scroll down until you find the **Kubernetes service** entries.

c) Click on the Kubernetes service that you configured.

   The Overview page for that Kubernetes service is displayed.

d) In the left nav bar, click on **Access Control (IAM)**.

   The Access Control (IAM) for that Kubernetes service is displayed.

e) Click **+ Add**, then select **Add role assignment** from the drop-down menu.

f) In the **Add role assignment** page, make the following selections:

   - In the **Role** field, select **Azure Kubernetes Service Cluster Admin Role** from the drop-down menu.
   - In the **Assign access to** field, select **User, group, or service principal**.
   - Select the appropriate key.

g) Click **Save** at the bottom of the screen.