

## Ingress & Nginx Ingress Controller Architecture

Here is a high-level architecture of Kubernetes ingress using the Nginx ingress controller. In this guide, we will learn to build the setup in the architecture.

### AKS Cluster

Name : IngressAKS

The screenshot displays the Microsoft Azure portal interface for the 'IngressAKS' Kubernetes service. The left sidebar shows the navigation menu with options like Overview, Activity log, Access control (IAM), Tags, Monitor, and more. The main content area is divided into sections: Essentials, Properties, Monitoring, and Recommendations. The Properties section is currently selected, showing details about the Kubernetes service, including its version (1.31.8), API server address, network configuration, and node pools. The Networking section shows details about the API server address, network configuration, and DNS service IP. The Integrations section shows details about container insights and workspace resource ID.

Section	Property	Value
Essentials	Resource group	IngressAKS_group
	Power state	Running
	Cluster operation status	Succeeded
	Subscription	azure_subscription_1
	Location	Canada Central
Properties	Subscription ID	42a59440-5c60-4a84-9349-f13bfeba1bf
	Fleet Manager	Click here to assign
	Tags (edit)	Add tags
	Get started	Properties
Kubernetes services	Encryption type	Encryption at rest with a platform-managed key
	Virtual node pools	Not enabled
	Node pools	1 node pool
Configuration	Kubernetes versions	1.31.8
	Node sizes	Standard_A2_v2
	Kubernetes version	1.31.8
	Auto Upgrade Type	Patch
Networking	API server address	ingressaks-dns-pap6926p.hcp.canadacentral.azmk8s.io
	Network configuration	Azure CNI Node Subnet
	Pod CIDR	-
	Service CIDR	10.0.0.0/16
Integrations	DNS service IP	10.0.0.10
	Network Policy	None
	Load balancer	Standard
	Private cluster	Not enabled

### Resourcegroup

The screenshot displays the Microsoft Azure portal interface for the 'IngressAKS\_group' resource group. The left sidebar shows the navigation menu with options like Overview, Activity log, Access control (IAM), Tags, Resource visualizer, Events, Settings, Cost Management, Monitoring, Automation, and Help. The main content area is divided into sections: Overview, Essentials, and Resources. The Resources section is currently selected, showing a list of resources within the resource group. The resources include Ingress-Kubectl, Ingress-Kubectl-ip, Ingress-Kubectl-nsg, Ingress-Kubectl686\_z1, Ingress-Kubectl\_disk1, Ingress-Kubectl\_key, IngressAKS, and IngressAKS\_group-vnet.

Name	Type	Location
Ingress-Kubectl	Virtual machine	Canada Central
Ingress-Kubectl-ip	Public IP address	Canada Central
Ingress-Kubectl-nsg	Network security group	Canada Central
Ingress-Kubectl686_z1	Network interface	Canada Central
Ingress-Kubectl_disk1	Disk	Canada Central
Ingress-Kubectl_key	SSH key	Canada Central
IngressAKS	Kubernetes service	Canada Central
IngressAKS_group-vnet	Virtual network	Canada Central

## Kubectl VM to connect with AKS cluster

**Ingress-Kubectl** Virtual machine

Advisor (1 of 1): Migrate workload to Virtual Machine Scale Sets Flex →

Help me copy this VM in any region

Connect Start Restart Stop Hibernate Capture Delete Refresh Open in mobile Feedback CLI / PS

**Essentials**

Resource group (move) : [IngressAKS\\_group](#)

Status : Running

Location : Canada Central (Zone 1)

Subscription (move) : [Azure subscription 1](#)

Subscription ID : 42a59440-5c60-4a84-9349-ff13bfeba1bf

Availability zone : 1

Operating system : Linux (ubuntu 24.04)

Size : Standard B1s (1 vcpu, 1 GiB memory)

Public IP address : [20.39.140.212](#)

Virtual network/subnet : [IngressAKS\\_group-vnet/default](#)

DNS name : [Not configured](#)

Health state : -

Time created : 5/26/2025, 4:19 PM UTC

Tags (edit) : [Add tags](#)

**Properties** Monitoring Capabilities (7) Recommendations (1) Tutorials

**Virtual machine**

Computer name : Ingress-Kubectl

Operating system : Linux (ubuntu 24.04)

VM generation : V2

VM architecture : x64

Agent status : Ready

Agent version : 2.13.1.1

Hibernation : Disabled

Host group : -

Host : -

Proximity placement group : -

**Networking**

Public IP address : [20.39.140.212](#) (Network interface [ingress-kubectl686\\_z1](#))

Public IP address (IPv6) : -

Private IP address : 10.224.0.33

Private IP address (IPv6) : -

Virtual network/subnet : [IngressAKS\\_group-vnet/default](#)

DNS name : [Configure](#)

**Size**

Size : Standard B1s

vCPUs : 1

RAM : 1 GiB

## Connect to Kubernetes using Azure CLI

az aks get-credentials --resource-group IngressAKS\_group --name IngressAKS --overwrite-existing

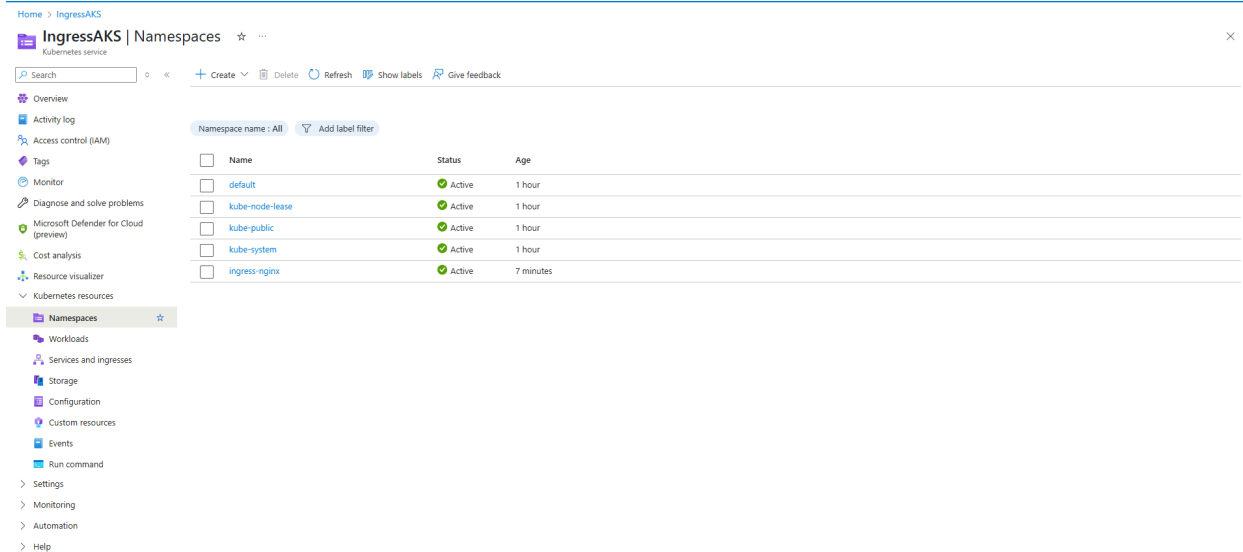
```
Switch to PowerShell Restart Manage files New session Editor Web preview Settings Help
Requesting a Cloud Shell.Succeeded.
Connecting terminal...

Welcome to Azure Cloud Shell

Type "az" to use Azure CLI
Type "help" to learn about Cloud Shell

Your Cloud Shell session will be ephemeral so no files or system changes will persist beyond your current session.
payal [ ~ ]$ az account set --subscription 42a59440-5c60-4a84-9349-ff13bfeba1bf
payal [ ~ ]$ az aks get-credentials --resource-group IngressAKS_group --name IngressAKS --overwrite-existing
Merged "IngressAKS" as current context in /home/payal/.kube/config
payal [ ~ ]$ kubectl get deployments --all-namespaces=true
NAMESPACE   NAME                                     READY   UP-TO-DATE   AVAILABLE   AGE
kube-system  azure-wi-webhook-controller-manager    2/2     2             2           21m
kube-system  coredns                                2/2     2             2           22m
kube-system  coredns-autoscaler                     1/1     1             1           22m
kube-system  eraser-controller-manager              1/1     1             1           20m
kube-system  connectivity-agent                     2/2     2             2           22m
kube-system  connectivity-agent-autoscaler           1/1     1             1           22m
kube-system  metrics-server                         2/2     2             2           22m
payal [ ~ ]$ kubectl get deployments --namespace kube-system
NAME                                     READY   UP-TO-DATE   AVAILABLE   AGE
azure-wi-webhook-controller-manager    2/2     2             2           21m
coredns                                2/2     2             2           23m
coredns-autoscaler                     1/1     1             1           23m
eraser-controller-manager              1/1     1             1           21m
connectivity-agent                     2/2     2             2           23m
connectivity-agent-autoscaler           1/1     1             1           23m
metrics-server                         2/2     2             2           23m
payal [ ~ ]$ kubectl get deployments --namespace coredns
```

## Create a Namespace to deploy the Nginxcontroller



```
payal [ ~/nginx-ingress-controller/manifests ]$ kubectl create ns ingress-nginx
namespace/ingress-nginx created
payal [ ~/nginx-ingress-controller/manifests ]$ kubectl get namespaces
NAME                STATUS    AGE
default             Active   67m
ingress-nginx       Active   13s
kube-node-lease     Active   67m
kube-public         Active   67m
kube-system         Active   67m
payal [ ~/nginx-ingress-controller/manifests ]$
```

## Create Admission Controller Roles & Service Account

```
payal [ ~/nginx-ingress-controller/manifests ]$ kubectl apply -f admission-service-account.yaml
serviceaccount/ingress-nginx-admission created
role.rbac.authorization.k8s.io/ingress-nginx-admission created
rolebinding.rbac.authorization.k8s.io/ingress-nginx-admission created
clusterrole.rbac.authorization.k8s.io/ingress-nginx-admission created
clusterrolebinding.rbac.authorization.k8s.io/ingress-nginx-admission created
payal [ ~/nginx-ingress-controller/manifests ]$
```

### Yaml admission-service-account.yaml

```
payal [ ~/nginx-ingress-controller/manifests ]$ cat admission-service-account.yaml
```

apiVersion: v1

kind: ServiceAccount

metadata:

labels:

app.kubernetes.io/component: admission-webhook

app.kubernetes.io/instance: ingress-nginx

app.kubernetes.io/name: ingress-nginx

name: ingress-nginx-admission

namespace: ingress-nginx

---

apiVersion: rbac.authorization.k8s.io/v1

kind: Role

metadata:

annotations:

app.kubernetes.io/component: admission-webhook

app.kubernetes.io/instance: ingress-nginx

app.kubernetes.io/name: ingress-nginx

name: ingress-nginx-admission

namespace: ingress-nginx

rules:

- apiGroups:

- ""

resources:

- secrets

verbs:

- get

- create

---

apiVersion: rbac.authorization.k8s.io/v1

kind: RoleBinding

metadata:

labels:

app.kubernetes.io/component: admission-webhook

app.kubernetes.io/instance: ingress-nginx

app.kubernetes.io/name: ingress-nginx

name: ingress-nginx-admission

namespace: ingress-nginx

roleRef:

apiGroup: rbac.authorization.k8s.io

kind: Role

name: ingress-nginx-admission

subjects:

- kind: ServiceAccount

name: ingress-nginx-admission

namespace: ingress-nginx

---

apiVersion: rbac.authorization.k8s.io/v1

kind: ClusterRole

```

metadata:
  labels:
    app.kubernetes.io/component: admission-webhook
    app.kubernetes.io/instance: ingress-nginx
    app.kubernetes.io/name: ingress-nginx
  name: ingress-nginx-admission
rules:
- apiGroups:
  - admissionregistration.k8s.io
  resources:
  - validatingwebhookconfigurations
  verbs:
  - get
  - update

```

---

```

apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRoleBinding
metadata:
  labels:
    app.kubernetes.io/component: admission-webhook
    app.kubernetes.io/instance: ingress-nginx
    app.kubernetes.io/name: ingress-nginx
  name: ingress-nginx-admission
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: ClusterRole
  name: ingress-nginx-admission
subjects:
- kind: ServiceAccount
  name: ingress-nginx-admission
  namespace: ingress-nginx

```

### • Create Validating Webhook Configuration

```

payal [ ~/nginx-ingress-controller/manifests ]$ kubectl apply -f validating-webhook.yaml
validatingwebhookconfiguration.admissionregistration.k8s.io/ingress-nginx-admission created
payal [ ~/nginx-ingress-controller/manifests ]$

```

```

payal [ ~/nginx-ingress-controller/manifests ]$ cat admission-service-account.yaml
apiVersion: v1
kind: ServiceAccount
metadata:

```

labels:  
 app.kubernetes.io/component: admission-webhook  
 app.kubernetes.io/instance: ingress-nginx  
 app.kubernetes.io/name: ingress-nginx  
name: ingress-nginx-admission  
namespace: ingress-nginx

---

apiVersion: rbac.authorization.k8s.io/v1  
kind: Role  
metadata:  
 annotations:  
 app.kubernetes.io/component: admission-webhook  
 app.kubernetes.io/instance: ingress-nginx  
 app.kubernetes.io/name: ingress-nginx  
 name: ingress-nginx-admission  
 namespace: ingress-nginx  
rules:  
- apiGroups:  
 - ""  
 resources:  
 - secrets  
 verbs:  
 - get  
 - create

---

apiVersion: rbac.authorization.k8s.io/v1  
kind: RoleBinding  
metadata:  
 labels:  
 app.kubernetes.io/component: admission-webhook  
 app.kubernetes.io/instance: ingress-nginx  
 app.kubernetes.io/name: ingress-nginx  
 name: ingress-nginx-admission  
 namespace: ingress-nginx  
roleRef:  
 apiGroup: rbac.authorization.k8s.io  
 kind: Role  
 name: ingress-nginx-admission  
subjects:  
- kind: ServiceAccount  
 name: ingress-nginx-admission  
 namespace: ingress-nginx

```
---
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
  labels:
    app.kubernetes.io/component: admission-webhook
    app.kubernetes.io/instance: ingress-nginx
    app.kubernetes.io/name: ingress-nginx
  name: ingress-nginx-admission
rules:
- apiGroups:
  - admissionregistration.k8s.io
  resources:
  - validatingwebhookconfigurations
  verbs:
  - get
  - update
```

```
---
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRoleBinding
metadata:
  labels:
    app.kubernetes.io/component: admission-webhook
    app.kubernetes.io/instance: ingress-nginx
    app.kubernetes.io/name: ingress-nginx
  name: ingress-nginx-admission
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: ClusterRole
  name: ingress-nginx-admission
subjects:
- kind: ServiceAccount
  name: ingress-nginx-admission
  namespace: ingress-nginx
```

## **Deploy Jobs To Update Webhook Certificates**

```
admission-service-account.yaml configmap.yaml deployment.yaml ingress-controller.yaml
payal [ ~/nginx-ingress-controller/manifests ]$ kubectl apply -f jobs.yaml
job.batch/ingress-nginx-admission-create created
job.batch/ingress-nginx-admission-patch created
payal [ ~/nginx-ingress-controller/manifests ]$
```

```
payal [ ~/nginx-ingress-controller/manifests ]$ cat jobs.yaml
```

```
apiVersion: batch/v1
```

```
kind: Job
```

```
metadata:
```

```
  labels:
```

```
    app.kubernetes.io/component: controller
```

```
    app.kubernetes.io/instance: ingress-nginx
```

```
    app.kubernetes.io/name: ingress-nginx
```

```
  name: ingress-nginx-admission-create
```

```
  namespace: ingress-nginx
```

```
spec:
```

```
  template:
```

```
    metadata:
```

```
      labels:
```

```
        app.kubernetes.io/component: controller
```

```
        app.kubernetes.io/instance: ingress-nginx
```

```
        app.kubernetes.io/name: ingress-nginx
```

```
      name: ingress-nginx-admission-create
```

```
    spec:
```

```
      containers:
```

```
        - args:
```

```
          - create
```

```
        -
```

```
      --host=ingress-nginx-controller-admission,ingress-nginx-controller-admission.$(POD_NAMESPACE).svc
```

```
        - --namespace=$(POD_NAMESPACE)
```

```
        - --secret-name=ingress-nginx-admission
```

```
      env:
```

```
        - name: POD_NAMESPACE
```

```
          valueFrom:
```

```
            fieldRef:
```

```
              fieldPath: metadata.namespace
```

```
      image: registry.k8s.io/ingress-nginx/kube-webhook-certgen:v20231011-8b53cabe0
```

```
      imagePullPolicy: IfNotPresent
```

```
      name: create
```

```
      securityContext:
```

```
        allowPrivilegeEscalation: false
```



```
nodeSelector:
  kubernetes.io/os: linux
restartPolicy: OnFailure
securityContext:
  runAsNonRoot: true
  runAsUser: 2000
serviceAccountName: ingress-nginx-admission
```

---

```
apiVersion: batch/v1
kind: Job
metadata:
  labels:
    app.kubernetes.io/component: admission-webhook
    app.kubernetes.io/instance: ingress-nginx
    app.kubernetes.io/name: ingress-nginx
  name: ingress-nginx-admission-patch
  namespace: ingress-nginx
spec:
  template:
    metadata:
      labels:
        app.kubernetes.io/component: admission-webhook
        app.kubernetes.io/instance: ingress-nginx
        app.kubernetes.io/name: ingress-nginx
      name: ingress-nginx-admission-patch
    spec:
      containers:
      - args:
        - patch
        - --webhook-name=ingress-nginx-admission
        - --namespace=$(POD_NAMESPACE)
        - --patch-mutating=false
        - --secret-name=ingress-nginx-admission
        - --patch-failure-policy=Fail
        env:
        - name: POD_NAMESPACE
          valueFrom:
            fieldRef:
              fieldPath: metadata.namespace
        image: registry.k8s.io/ingress-nginx/kube-webhook-certgen:v20231011-8b53cabe0
        imagePullPolicy: IfNotPresent
        name: patch
        securityContext:
          allowPrivilegeEscalation: false
```

```
nodeSelector:
  kubernetes.io/os: linux
restartPolicy: OnFailure
securityContext:
  runAsNonRoot: true
  runAsUser: 2000
serviceAccountName: ingress-nginx-admission
payal [ ~/nginx-ingress-controller/manifests ]$ cat jobs.yaml
```

```
apiVersion: batch/v1
```

```
kind: Job
```

```
metadata:
```

```
  labels:
```

```
    app.kubernetes.io/component: controller
```

```
    app.kubernetes.io/instance: ingress-nginx
```

```
    app.kubernetes.io/name: ingress-nginx
```

```
  name: ingress-nginx-admission-create
```

```
  namespace: ingress-nginx
```

```
spec:
```

```
  template:
```

```
    metadata:
```

```
      labels:
```

```
        app.kubernetes.io/component: controller
```

```
        app.kubernetes.io/instance: ingress-nginx
```

```
        app.kubernetes.io/name: ingress-nginx
```

```
      name: ingress-nginx-admission-create
```

```
    spec:
```

```
      containers:
```

```
        - args:
```

```
          - create
```

```
        -
```

```
--host=ingress-nginx-controller-admission,ingress-nginx-controller-admission.$(POD_NAMESPACE).svc
```

```
  - --namespace=$(POD_NAMESPACE)
```

```
  - --secret-name=ingress-nginx-admission
```

```
  env:
```

```
    - name: POD_NAMESPACE
```

```
      valueFrom:
```

```
        fieldRef:
```

```
          fieldPath: metadata.namespace
```

```
  image: registry.k8s.io/ingress-nginx/kube-webhook-certgen:v20231011-8b53cabe0
```

```
  imagePullPolicy: IfNotPresent
```

```
  name: create
```

```
  securityContext:
```

```
    allowPrivilegeEscalation: false
nodeSelector:
  kubernetes.io/os: linux
restartPolicy: OnFailure
securityContext:
  runAsNonRoot: true
  runAsUser: 2000
serviceAccountName: ingress-nginx-admission
```

---

```
apiVersion: batch/v1
kind: Job
metadata:
  labels:
    app.kubernetes.io/component: admission-webhook
    app.kubernetes.io/instance: ingress-nginx
    app.kubernetes.io/name: ingress-nginx
  name: ingress-nginx-admission-patch
  namespace: ingress-nginx
spec:
  template:
    metadata:
      labels:
        app.kubernetes.io/component: admission-webhook
        app.kubernetes.io/instance: ingress-nginx
        app.kubernetes.io/name: ingress-nginx
      name: ingress-nginx-admission-patch
    spec:
      containers:
        - args:
            - patch
            - --webhook-name=ingress-nginx-admission
            - --namespace=$(POD_NAMESPACE)
            - --patch-mutating=false
            - --secret-name=ingress-nginx-admission
            - --patch-failure-policy=Fail
          env:
            - name: POD_NAMESPACE
              valueFrom:
                fieldRef:
                  fieldPath: metadata.namespace
          image: registry.k8s.io/ingress-nginx/kube-webhook-certgen:v20231011-8b53cabe0
          imagePullPolicy: IfNotPresent
          name: patch
          securityContext:
```

```
allowPrivilegeEscalation: false
nodeSelector:
  kubernetes.io/os: linux
restartPolicy: OnFailure
securityContext:
  runAsNonRoot: true
  runAsUser: 2000
```

**serviceAccountName: ingress-nginx-admission**

```
payal [ ~/nginx-ingress-controller/manifests ]$ kubectl get jobs -n ingress-nginx
NAME                                STATUS    COMPLETIONS   DURATION   AGE
ingress-nginx-admission-create     Complete  1/1            9s         109s
ingress-nginx-admission-patch      Complete  1/1            10s        109s
payal [ ~/nginx-ingress-controller/manifests ]$
```

## Verify the PatchBundle

```
payal [ ~/nginx-ingress-controller/manifests ]$ kubectl describe ValidatingWebhookConfiguration ingress-nginx-admission
Name:          ingress-nginx-admission
Namespace:     app.kubernetes.io/component=admission-webhook
Labels:        app.kubernetes.io/instance=ingress-nginx
               app.kubernetes.io/name=ingress-nginx
Annotations:   <none>
API Version:   admissionregistration.k8s.io/v1
Kind:          ValidatingWebhookConfiguration
Metadata:
  Creation Timestamp:  2025-05-26T17:32:51Z
  Generation:         3
  Resource Version:    24152
  UID:                 d7fa0b9b-e76b-48c4-9621-bb3df351aa14
Webhooks:
  Admission Review Versions:
    v1
  Client Config:
    Ca Bundle:  E30tE1cRdG7iBDRWJUSU2J00F0R80tL80tck1J3U3kaKNDQVJ120F38U3B018Ty9M67lyc3hnc90dX8U2sPbbU1QaKFLQmdacWbaaB9OUVFFOWpBUE1RMHt0Q3d2PFB0UvPd1J1Vd3e1D0VhEVE2kTUVVwU5qg2NNaK05TFVWU6G0XhMa1V
    y3pVpVE1Cv2pB0UgW7B3Qld2R2RUtPd1J1VY43e1Ga3dPd111829a8Xp-qMENSUV1J829a8Xp-qMER8UWNWU48RV2WjdiYWHclhm01VNEJUJ2VYtUtR23dqJ83UNN4R3fVWV0rc2wmlpe0G0Qcm13RUt8c0NT-qK2z8Wd0ZU13M2P4S3VXqVMKczY4M05PSnqwd8
    1GvXvE211V1w0FF8SC9CQFFPQWdJRU1CTUd8MvVks1FR7U1Bb0dQ3NHQVfV8gpCd01CTUE4R0ExVWRfD0VCL3dRAk1BT0JB2jh38FF2R23ME9CQ11F8kduck1F0W5wWENqN1cyanhjRzN6dXg3CjJ62jJN0W9HQ0NkR1NNND1CQ1DQ7BzQ01FWUNJUUNhUwWmXN8
    sRtWmE4TmN6b11lbJdaG029uTmEK0tcdVJPeXY5201oQ014R0xPdUN0YkdwN22ydfUJY3dTCWtj2Xo5WDMwY2U2a0ptMDIdvMgX7WwK50L01FTk0gQ0VSV2LGSUNBEU1S0L1G0=
  Service:
    Name:          ingress-nginx-controller-admission
    Namespace:     ingress-nginx
    Path:          /networking/v1/ingresses
    Port:          443
    Failure Policy: Fail
    Match Policy:   Equivalent
    Name:          validate.nginx.ingress.kubernetes.io
    Namespace Selector:
      Match Expressions:
        Key:      kubernetes.azure.com/managedby
        Operator: NotIn
        Values:
          aks
        Key:      control-plane
        Operator: NotIn
        Values:
          true
```

```
payal [ ~/nginx-ingress-controller/manifests ]$ kubectl describe ValidatingWebhookConfiguration
ingress-nginx-admission
Name:          ingress-nginx-admission
Namespace:     app.kubernetes.io/component=admission-webhook
Labels:        app.kubernetes.io/component=admission-webhook
               app.kubernetes.io/instance=ingress-nginx
               app.kubernetes.io/name=ingress-nginx
Annotations:   <none>
API Version:   admissionregistration.k8s.io/v1
Kind:          ValidatingWebhookConfiguration
Metadata:
  Creation Timestamp:  2025-05-26T17:32:51Z
  Generation:         3
```

Resource Version: 24152

UID: d7fa0b9b-e76b-48c4-9621-bb3df351aa14

Webhooks:

Admission Review Versions:

v1

Client Config:

Ca Bundle:

LS0tLS1CRUdJTiBDRVJUSUZJQ0FURStLS0tCk1JSUJkakNDQVJ1Z0F3SUJBZ0lRYY9MbTIYc2hiTC80dXBUZzFpbU1QakFLQmdncWhrak9QUVFEQWpBUE1RMHcKQ3dZRFZRUUtFd1J1YVd3eE1DQVhEVEkxTURVeU5qRTNNEkV5TTFvWUR6SXhNaIV3TIRBeU1UY3pNVEl6V2pBUApNUTB3Q3dZRFZRUUtFd1J1YVd3eE1Ga3dFd1IIS29aSXpqMENBUVJIS29aSXpqMERBUWNEUWdBRVZwNjdIYWJHCihmQIVVNEJUZ2VYTUtKZ3dqVjB3UWN4RjFVVVQrc2wrWlpuOGQvQm13RUtBc0NtQkEzRWdUZUJ3MGF4S3VXdVMKczY4MU5pSngwdHRoTXFOWE1GVXdEZ1IEVIlwUEFRSC9CQVFEQWdJRUICTUdBMVVsSIFRTU1Bb0dDQ3NHQVFVRgpCd01CTUE4R0ExVWRFd0VCL3dRRk1BTUJBZjh3SFFZRFZSMER9CQlIFRkduduckIPUW5wWENqN1cyanhjRzN6dXg3CjJ6ZjJNQW9HQ0NxR1NNNDICQU1DQTBQU1FWUNJUUNHbUwwMWxNRnR4K0VBZzRtRmE4TmN6b1lBjdaU29uTmEKOUtxdVJPeXY5Z0loQU14RUxPdUN0YkdWN2ZydFJLY3dTWjZlZXo5WDMwY2U2a0ptMDIvdWgxTWwKLS0tLS1FTkQgQ0VSVEIGSUNBVEU1LS0tLQo=

Service:

Name: ingress-nginx-controller-admission

Namespace: ingress-nginx

Path: /networking/v1/ingresses

Port: 443

Failure Policy: Fail

Match Policy: Equivalent

Name: validate.nginx.ingress.kubernetes.io

Namespace Selector:

Match Expressions:

Key: kubernetes.azure.com/managedby

Operator: NotIn

Values:

aks

Key: control-plane

Operator: NotIn

Values:

true

Object Selector:

Rules:

API Groups:

networking.k8s.io

API Versions:

v1

Operations:

CREATE  
UPDATE  
Resources:  
  ingresses  
Scope:        \*  
Side Effects:  None  
Timeout Seconds: 10  
Events:        <none>

## Create Ingress Controller Roles & Service Account

```
payal [ ~/nginx-ingress-controller/manifests ]$ kubectl apply -f ingress-service-account.yaml
serviceaccount/ingress-nginx created
serviceaccount/ingress-nginx-admission configured
role.rbac.authorization.k8s.io/ingress-nginx created
role.rbac.authorization.k8s.io/ingress-nginx-admission configured
clusterrole.rbac.authorization.k8s.io/ingress-nginx created
clusterrole.rbac.authorization.k8s.io/ingress-nginx-admission configured
rolebinding.rbac.authorization.k8s.io/ingress-nginx created
rolebinding.rbac.authorization.k8s.io/ingress-nginx-admission configured
clusterrolebinding.rbac.authorization.k8s.io/ingress-nginx created
clusterrolebinding.rbac.authorization.k8s.io/ingress-nginx-admission configured
payal [ ~/nginx-ingress-controller/manifests ]$
```

```
payal [ ~/nginx-ingress-controller/manifests ]$ cat ingress-service-account.yaml
```

```
apiVersion: v1
automountServiceAccountToken: true
kind: ServiceAccount
metadata:
  labels:
    app.kubernetes.io/component: controller
    app.kubernetes.io/instance: ingress-nginx
    app.kubernetes.io/name: ingress-nginx
    app.kubernetes.io/part-of: ingress-nginx
    app.kubernetes.io/version: 1.9.4
  name: ingress-nginx
  namespace: ingress-nginx
---
apiVersion: v1
kind: ServiceAccount
metadata:
  labels:
    app.kubernetes.io/component: admission-webhook
    app.kubernetes.io/instance: ingress-nginx
    app.kubernetes.io/name: ingress-nginx
    app.kubernetes.io/part-of: ingress-nginx
    app.kubernetes.io/version: 1.9.4
```

name: ingress-nginx-admission  
namespace: ingress-nginx

---

apiVersion: rbac.authorization.k8s.io/v1

kind: Role

metadata:

labels:

app.kubernetes.io/component: controller  
app.kubernetes.io/instance: ingress-nginx  
app.kubernetes.io/name: ingress-nginx  
app.kubernetes.io/part-of: ingress-nginx  
app.kubernetes.io/version: 1.9.4

name: ingress-nginx

namespace: ingress-nginx

rules:

- apiGroups:

- ""

resources:

- namespaces

verbs:

- get

- apiGroups:

- ""

resources:

- configmaps

- pods

- secrets

- endpoints

verbs:

- get

- list

- watch

- apiGroups:

- ""

resources:

- services

verbs:

- get

- list

- watch

- apiGroups:

- networking.k8s.io

resources:

- ingresses

verbs:  
- get  
- list  
- watch  
- apiGroups:  
- networking.k8s.io  
resources:  
- ingresses/status  
verbs:  
- update  
- apiGroups:  
- networking.k8s.io  
resources:  
- ingressclasses  
verbs:  
- get  
- list  
- watch  
- apiGroups:  
- coordination.k8s.io  
resourceNames:  
- ingress-nginx-leader  
resources:  
- leases  
verbs:  
- get  
- update  
- apiGroups:  
- coordination.k8s.io  
resources:  
- leases  
verbs:  
- create  
- apiGroups:  
- ""  
resources:  
- events  
verbs:  
- create  
- patch  
- apiGroups:  
- discovery.k8s.io  
resources:  
- endpointslices



verbs:

- list
- watch
- get

---

apiVersion: rbac.authorization.k8s.io/v1

kind: Role

metadata:

labels:

- app.kubernetes.io/component: admission-webhook
- app.kubernetes.io/instance: ingress-nginx
- app.kubernetes.io/name: ingress-nginx
- app.kubernetes.io/part-of: ingress-nginx
- app.kubernetes.io/version: 1.9.4
- name: ingress-nginx-admission
- namespace: ingress-nginx

rules:

- apiGroups:

- ""

resources:

- secrets

verbs:

- get
- create

---

apiVersion: rbac.authorization.k8s.io/v1

kind: ClusterRole

metadata:

labels:

- app.kubernetes.io/instance: ingress-nginx
- app.kubernetes.io/name: ingress-nginx
- app.kubernetes.io/part-of: ingress-nginx
- app.kubernetes.io/version: 1.9.4
- name: ingress-nginx

rules:

- apiGroups:

- ""

resources:

- configmaps
- endpoints
- nodes
- pods
- secrets
- namespaces

```
verbs:
- list
- watch
- apiGroups:
  - coordination.k8s.io
resources:
- leases
verbs:
- list
- watch
- apiGroups:
  - ""
resources:
- nodes
verbs:
- get
- apiGroups:
  - ""
resources:
- services
verbs:
- get
- list
- watch
- apiGroups:
  - networking.k8s.io
resources:
- ingresses
verbs:
- get
- list
- watch
- apiGroups:
  - ""
resources:
- events
verbs:
- create
- patch
- apiGroups:
  - networking.k8s.io
resources:
- ingresses/status
verbs:
```

```
- update
- apiGroups:
  - networking.k8s.io
  resources:
  - ingressclasses
  verbs:
  - get
  - list
  - watch
- apiGroups:
  - discovery.k8s.io
  resources:
  - endpointslices
  verbs:
  - list
  - watch
  - get
---
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
  labels:
    app.kubernetes.io/component: admission-webhook
    app.kubernetes.io/instance: ingress-nginx
    app.kubernetes.io/name: ingress-nginx
    app.kubernetes.io/part-of: ingress-nginx
    app.kubernetes.io/version: 1.9.4
  name: ingress-nginx-admission
rules:
- apiGroups:
  - admissionregistration.k8s.io
  resources:
  - validatingwebhookconfigurations
  verbs:
  - get
  - update
---
apiVersion: rbac.authorization.k8s.io/v1
kind: RoleBinding
metadata:
  labels:
    app.kubernetes.io/component: controller
    app.kubernetes.io/instance: ingress-nginx
    app.kubernetes.io/name: ingress-nginx
```

```
  app.kubernetes.io/part-of: ingress-nginx
  app.kubernetes.io/version: 1.9.4
  name: ingress-nginx
  namespace: ingress-nginx
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: Role
  name: ingress-nginx
subjects:
- kind: ServiceAccount
  name: ingress-nginx
  namespace: ingress-nginx
---
apiVersion: rbac.authorization.k8s.io/v1
kind: RoleBinding
metadata:
  labels:
    app.kubernetes.io/component: admission-webhook
    app.kubernetes.io/instance: ingress-nginx
    app.kubernetes.io/name: ingress-nginx
    app.kubernetes.io/part-of: ingress-nginx
    app.kubernetes.io/version: 1.9.4
  name: ingress-nginx-admission
  namespace: ingress-nginx
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: Role
  name: ingress-nginx-admission
subjects:
- kind: ServiceAccount
  name: ingress-nginx-admission
  namespace: ingress-nginx
---
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRoleBinding
metadata:
  labels:
    app.kubernetes.io/instance: ingress-nginx
    app.kubernetes.io/name: ingress-nginx
    app.kubernetes.io/part-of: ingress-nginx
    app.kubernetes.io/version: 1.9.4
  name: ingress-nginx
roleRef:
  apiGroup: rbac.authorization.k8s.io
```

```

kind: ClusterRole
name: ingress-nginx
subjects:
- kind: ServiceAccount
  name: ingress-nginx
  namespace: ingress-nginx
---
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRoleBinding
metadata:
  labels:
    app.kubernetes.io/component: admission-webhook
    app.kubernetes.io/instance: ingress-nginx
    app.kubernetes.io/name: ingress-nginx
    app.kubernetes.io/part-of: ingress-nginx
    app.kubernetes.io/version: 1.9.4
  name: ingress-nginx-admission
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: ClusterRole
  name: ingress-nginx-admission
subjects:
- kind: ServiceAccount
  name: ingress-nginx-admission
  namespace: ingress-nginx

```

## Create ConfigMap

```

payal [ ~/nginx-ingress-controller/manifests ]$ kubectl apply -f configmap.yaml
configmap/ingress-nginx-controller created
payal [ ~/nginx-ingress-controller/manifests ]$

```

```

payal [ ~/nginx-ingress-controller/manifests ]$ cat configmap.yaml
apiVersion: v1

```

```

data:
  allow-snippet-annotations: "false"
  hsts: "false"
kind: ConfigMap
metadata:
  labels:
    app.kubernetes.io/component: controller
    app.kubernetes.io/instance: ingress-nginx
    app.kubernetes.io/name: ingress-nginx
    app.kubernetes.io/part-of: ingress-nginx
    app.kubernetes.io/version: 1.9.4

```

name: ingress-nginx-controller  
namespace: ingress-nginx

## Create Ingress Controller & Admission Controller Services

```
payal [ ~/nginx-ingress-controller/manifests ]$ kubectl apply -f services.yaml  
service/ingress-nginx-controller created  
service/ingress-nginx-controller-admission created  
payal [ ~/nginx-ingress-controller/manifests ]$
```

```
payal [ ~/nginx-ingress-controller/manifests ]$ cat services.yaml
```

apiVersion: v1

kind: Service

metadata:

labels:

app.kubernetes.io/component: controller  
app.kubernetes.io/instance: ingress-nginx  
app.kubernetes.io/name: ingress-nginx  
app.kubernetes.io/part-of: ingress-nginx  
app.kubernetes.io/version: 1.9.4

name: ingress-nginx-controller

namespace: ingress-nginx

spec:

ipFamilies:

- IPv4

ipFamilyPolicy: SingleStack

ports:

- appProtocol: http

name: http

port: 80

protocol: TCP

targetPort: http

- appProtocol: https

name: https

port: 443

protocol: TCP

targetPort: https

selector:

app.kubernetes.io/component: controller

app.kubernetes.io/instance: ingress-nginx

app.kubernetes.io/name: ingress-nginx

type: LoadBalancer

---

apiVersion: v1

kind: Service

metadata:

```
labels:
  app.kubernetes.io/component: controller
  app.kubernetes.io/instance: ingress-nginx
  app.kubernetes.io/name: ingress-nginx
  app.kubernetes.io/part-of: ingress-nginx
  app.kubernetes.io/version: 1.9.4
name: ingress-nginx-controller-admission
namespace: ingress-nginx
spec:
  ports:
  - appProtocol: https
    name: https-webhook
    port: 443
    targetPort: webhook
  selector:
    app.kubernetes.io/component: controller
    app.kubernetes.io/instance: ingress-nginx
    app.kubernetes.io/name: ingress-nginx
  type: ClusterIP
payal [ ~/nginx-ingress-controller/manifests ]$
```

### Create IngressClass

```
type: ClusterIP
payal [ ~/nginx-ingress-controller/manifests ]$ kubectl apply -f ingressclass.yaml
ingressclass.networking.k8s.io/nginx created
payal [ ~/nginx-ingress-controller/manifests ]$
```

```
payal [ ~/nginx-ingress-controller/manifests ]$ cat ingressclass.yaml
```

```
apiVersion: networking.k8s.io/v1
kind: IngressClass
metadata:
  labels:
    app.kubernetes.io/component: admission-webhook
    app.kubernetes.io/instance: ingress-nginx
    app.kubernetes.io/name: ingress-nginx
  name: nginx
spec:
  controller: k8s.io/ingress-nginx
```

## Create Ingress Controller Deployment

```
payal [ ~/nginx-ingress-controller/manifests ]$ kubectl apply -f deployment.yaml
deployment.apps/ingress-nginx-controller created
payal [ ~/nginx-ingress-controller/manifests ]$ cat deployment.yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    app.kubernetes.io/component: controller
    app.kubernetes.io/instance: ingress-nginx
    app.kubernetes.io/name: ingress-nginx
    app.kubernetes.io/part-of: ingress-nginx
    app.kubernetes.io/version: 1.9.4
  name: ingress-nginx-controller
  namespace: ingress-nginx
spec:
  minReadySeconds: 0
  revisionHistoryLimit: 10
  selector:
    matchLabels:
      app.kubernetes.io/component: controller
      app.kubernetes.io/instance: ingress-nginx
      app.kubernetes.io/name: ingress-nginx
  strategy:
    rollingUpdate:
      maxUnavailable: 1
    type: RollingUpdate
  template:
    metadata:
      labels:
        app.kubernetes.io/component: controller
        app.kubernetes.io/instance: ingress-nginx
        app.kubernetes.io/name: ingress-nginx
        app.kubernetes.io/part-of: ingress-nginx
        app.kubernetes.io/version: 1.9.4
    spec:
      containers:
      - args:
        - /nginx-ingress-controller
        - --election-id=ingress-nginx-leader
        - --controller-class=k8s.io/ingress-nginx
```

```
payal [ ~/nginx-ingress-controller/manifests ]$ kubectl apply -f deployment.yaml
```

```
deployment.apps/ingress-nginx-controller created
```

```
payal [ ~/nginx-ingress-controller/manifests ]$ cat deployment.yaml
```

```
apiVersion: apps/v1
```

```
kind: Deployment
```

```
metadata:
```

```
  labels:
```

```
    app.kubernetes.io/component: controller
```

```
    app.kubernetes.io/instance: ingress-nginx
```

```
    app.kubernetes.io/name: ingress-nginx
```

```
    app.kubernetes.io/part-of: ingress-nginx
```

```
    app.kubernetes.io/version: 1.9.4
```

```
  name: ingress-nginx-controller
```

```
  namespace: ingress-nginx
```

```
spec:
```

```
  minReadySeconds: 0
```

```
  revisionHistoryLimit: 10
```

```
  selector:
```

```
    matchLabels:
```



```
  app.kubernetes.io/component: controller
  app.kubernetes.io/instance: ingress-nginx
  app.kubernetes.io/name: ingress-nginx
strategy:
  rollingUpdate:
    maxUnavailable: 1
  type: RollingUpdate
template:
  metadata:
    labels:
      app.kubernetes.io/component: controller
      app.kubernetes.io/instance: ingress-nginx
      app.kubernetes.io/name: ingress-nginx
      app.kubernetes.io/part-of: ingress-nginx
      app.kubernetes.io/version: 1.9.4
  spec:
    containers:
      - args:
          - /nginx-ingress-controller
          - --election-id=ingress-nginx-leader
          - --controller-class=k8s.io/ingress-nginx
          - --ingress-class=nginx
          - --configmap=$(POD_NAMESPACE)/ingress-nginx-controller
          - --validating-webhook=:8443
          - --validating-webhook-certificate=/usr/local/certificates/cert
          - --validating-webhook-key=/usr/local/certificates/key
        env:
          - name: POD_NAME
            valueFrom:
              fieldRef:
                fieldPath: metadata.name
          - name: POD_NAMESPACE
            valueFrom:
              fieldRef:
                fieldPath: metadata.namespace
          - name: LD_PRELOAD
            value: /usr/local/lib/libmimalloc.so
        image: registry.k8s.io/ingress-nginx/controller:v1.9.5
        imagePullPolicy: IfNotPresent
      lifecycle:
        preStop:
          exec:
            command:
              - /wait-shutdown
```

livenessProbe:  
failureThreshold: 5  
httpGet:  
  path: /healthz  
  port: 10254  
  scheme: HTTP  
initialDelaySeconds: 10  
periodSeconds: 10  
successThreshold: 1  
timeoutSeconds: 1  
name: controller  
ports:  
- containerPort: 80  
  name: http  
  protocol: TCP  
- containerPort: 443  
  name: https  
  protocol: TCP  
- containerPort: 8443  
  name: webhook  
  protocol: TCP  
readinessProbe:  
failureThreshold: 3  
httpGet:  
  path: /healthz  
  port: 10254  
  scheme: HTTP  
initialDelaySeconds: 10  
periodSeconds: 10  
successThreshold: 1  
timeoutSeconds: 1  
resources:  
  requests:  
    cpu: 100m  
    memory: 90Mi  
securityContext:  
  allowPrivilegeEscalation: true  
  capabilities:  
    add:  
      - NET\_BIND\_SERVICE  
    drop:  
      - ALL  
  runAsUser: 101  
volumeMounts:

- mountPath: /usr/local/certificates/  
name: webhook-cert  
readOnly: true
- dnsPolicy: ClusterFirst
- nodeSelector:  
kubernetes.io/os: linux
- serviceAccountName: ingress-nginx
- terminationGracePeriodSeconds: 300
- volumes:  
- name: webhook-cert  
secret:  
secretName: ingress-nginx-admission

## Troubleshooting:

### ⚠ What's Not Working (And Why)

! ingress-nginx-controller Pod is failing to schedule

FailedScheduling: 0/1 nodes are available: 1 Insufficient cpu.

! Readiness probe failed: HTTP probe failed with statuscode: 500

This likely happened due to the pod being forcefully replaced due to low resources.

## Resolution Steps

Check Node Capacity and Usage

Check how much CPU is available on your node:

Check Node resource utilization

kubectl describe node aks-ingressnode-31117155-vmss000000

```
Allocated resources:
 (Total limits may be over 100 percent, i.e., overcommitted.)
Resource           Requests          Limits
-----
cpu                1822m (95%)      12352m (650%)
memory            1125Mi (35%)     14342432Ki (442%)
ephemeral-storage 0 (0%)           0 (0%)
hugepages-1Gi     0 (0%)           0 (0%)
hugepages-2Mi     0 (0%)           0 (0%)
Events:
Type      Reason      Age   From      Message
----
Normal    RegisteredNode  11m   node-controller   Node aks-ingressnode-31117155-vmss000000 event: Registered Node aks-ingressnode-31117155-vmss000000 in Controller
payal [ ~/nginx-ingress-controller/manifests ]$
```



```
payal [ ~/nginx-ingress-controller/manifests ]$ kubectl create -f ingressresource.yaml
ingress.networking.k8s.io/my-app-ingress created
payal [ ~/nginx-ingress-controller/manifests ]$ curl http://20.248.118.19/
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
html { color-scheme: light dark; }
body { width: 35em; margin: 0 auto;
font-family: Tahoma, Verdana, Arial, sans-serif; }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>

<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
payal [ ~/nginx-ingress-controller/manifests ]$
```

```

payal [ ~/nginx-ingress-controller/manifests ]$ curl -v http://20.248.118.19
* Trying 20.248.118.19:80...
* Connected to 20.248.118.19 (20.248.118.19) port 80
* using HTTP/1.x
> GET / HTTP/1.1
> Host: 20.248.118.19
> User-Agent: curl/8.11.1
> Accept: */*
>
* Request completely sent off
< HTTP/1.1 200 OK
< Date: Tue, 27 May 2025 01:35:53 GMT
< Content-Type: text/html
< Content-Length: 615
< Connection: keep-alive
< Last-Modified: Wed, 16 Apr 2025 12:01:11 GMT
< ETag: "67ff9c07-267"
< Accept-Ranges: bytes
<
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx</title>

```

portal.azure.com/#@payalpdev@gmail.com/resource/subscriptions/42a59440-5c60-4a84-9349-f13bfeba1bf/resourceGroups/IngressAKS\_group/providers/Microsoft.ContainerService/managedClusters/IngressAKS...

Microsoft Azure Upgrade Search resources, services, and docs (G+)

Kubernetes services << IngressAKS | Services and ingresses

Default Directory (payalpdev@gmail.com/microsoft.com/...)

Group by none

Search

Create Delete Refresh Show labels Give feedback

Services Ingresses

Filter by namespace All namespaces Service name: All Add label filter

Name	Namespace	Status	Type	Cluster IP	External IP	Ports	Age
kubernetes	default	Ok	ClusterIP	10.0.0.1		443/TCP	1 hour
kube-dns	kube-system	Ok	ClusterIP	10.0.0.10		53/UDP,53/TCP	1 hour
metrics-server	kube-system	Ok	ClusterIP	10.0.9.37		443/TCP	1 hour
azure-wi-webhook-webhook-service	kube-system	Ok	ClusterIP	10.0.113.86		443/TCP	1 hour
ingress-nginx-controller	ingress-nginx	Ok	LoadBalancer	10.0.40.184	20.248.118.19	80:32483/TCP,4...	52 minutes
ingress-nginx-controller-admission	ingress-nginx	Ok	ClusterIP	10.0.187.220		443/TCP	52 minutes
nginx-metrics	app-routing-system	Ok	ClusterIP	10.0.159.26		10254/TCP	39 minutes

Overview Activity log Access control (IAM) Tags Monitor Diagnose and solve problems Microsoft Defender for Cloud (preview) Cost analysis Resource visualizer Favorites Kubernetes resources Namespaces Workloads Services and Ingresses Storage Configuration Custom resources Events Run command

Microsoft Azure

Upgrade

Search resources, services, and docs (G+)

Copilot

payal.pdevai@gmail.com

DEFAULT DIRECTORY (PAYALPDEVAI...)

Home > Kubernetes services > IngressAKS

Kubernetes services

Default Directory (payalpdevai@gmail.com@microsoft.co...)

Group by none

You are viewing a new version of Browse experience. Some features may be missing. Click here to access the old experience.

Name ?

IngressAKS

Overview

Activity log

Access control (IAM)

Tags

Monitor

Diagnose and solve problems

Microsoft Defender for Cloud (preview)

Cost analysis

Resource visualizer

Favorites

Kubernetes resources

Namespaces

Workloads

Services and ingresses

Storage

Configuration

Custom resources

IngressAKS | Namespaces

Kubernetes service

Search

Create

Delete

Refresh

Show labels

Give feedback

Namespace name: All

Add label filter

Name	Status	Age
default	Active	5 hours
kube-node-lease	Active	5 hours
kube-public	Active	5 hours
kube-system	Active	5 hours
ingress-nginx	Active	4 hours
app-routing-system	Active	4 hours

Microsoft Azure

Upgrade

Search resources, services, and docs (G+)

Copilot

payal.pdevai@gmail.com

DEFAULT DIRECTORY (PAYALPDEVAI...)

Home > Kubernetes services > IngressAKS

Kubernetes services

Default Directory (payalpdevai@gmail.com@microsoft.co...)

Group by none

You are viewing a new version of Browse experience. Some features may be missing. Click here to access the old experience.

Name ?

IngressAKS

Microsoft Defender for Cloud (preview)

Cost analysis

Resource visualizer

Favorites

Kubernetes resources

Namespaces

Workloads

Services and ingresses

Storage

Configuration

Custom resources

Events

Run command

Settings

Node pools

Upgrades

Security configuration

IngressAKS | Namespaces

Kubernetes service

Search

Create

Delete

Refresh

Show labels

Give feedback

Namespace name: All

Add label filter

Name	Status	Age
default	Active	5 hours
kube-node-lease	Active	5 hours
kube-public	Active	5 hours
kube-system	Active	5 hours
ingress-nginx	Active	4 hours
app-routing-system	Active	4 hours

Microsoft Azure

Upgrade

Search resources, services, and docs (G+)

Copilot

payal.pdevai@gmail.com

DEFAULT DIRECTORY (PAYALPDEVAI...)

Home > IngressAKS | Node pools > agentpool | Overview > IngressAKS | Node pools >

agentpool | Overview

Node pool

Search

Upgrade Kubernetes

Update image

Scale node pool

Delete

Refresh

Give feedback

Overview

Nodes

Configuration

Essentials

Provisioning state : Succeeded

Power state : Running

Availability zones : None

Mode : System

Cluster : IngressAKS

Operating system : Ubuntu Linux

Kubernetes version : 1.31.8

Target node count : 1 node

Ready nodes : 1/1 nodes ready

Node size : Standard\_D4as\_v5

Properties

Monitoring

Node pool

Max pods per node : 110

Public IPs per node : Disabled

Autoscaling : Disabled

Azure Spot Instance : Disabled

Maximum price : N/A

Scale eviction policy : N/A

Node image version : AKSUbuntu-2204gen2containerd-202505.14.0

Proximity placement group : N/A

Taints and labels

Taints (edit) : None

Labels (edit) : None

Configuration

Mode : System

Maximum surge : 10%

Node drain timeout : 30

Skip undrainable nodes : Disabled

OS disk size : 128 GB

OS disk type : Managed

Virtual network : Managed

Microsoft Azure

Upgrade

Search resources, services, and docs (G+)

Copilot

paypal.pdevval@gmail.com

DEFAULT DIRECTORY

Home > IngressAKS | Node pools > agentpool | Overview > IngressAKS | Node pools > agentpool | Nodes >

aks-agentpool-23298611-vmss000000 | Overview

Node

Search

Refresh

Connect

Give feedback

Overview

Pods

YAML

Events

Essentials

Status: Ready

Node pool: agentpool

Region: australiacentral

Internal IP: 10.224.0.4

External IP: N/A

Cluster: IngressAKS

Kubernetes version: 1.31.8

Host name: aks-agentpool-23298611-vmss000000

Node image version: AKSUbuntu-2204gen2containerd-202505.14.0

See more

Properties

Monitoring

Conditions

Node

Host name: aks-agentpool-23298611-vmss000000

Internal IP: 10.224.0.4

External IP: N/A

Kernel version: 5.15.0-1088-azure

Node image version: AKSUbuntu-2204gen2containerd-202505.14.0

Operating system: linux

Pod CIDR: N/A

Kubelet version: v1.31.8

Kube proxy version: v1.31.8

Daemon endpoints: 10250

Machine ID: 25b715118742460bba96b367464e15c2

System UUID: 717dc0bd-7e38-4027-8ab6-6a6997539a2c

Boot ID: ed82f18d-15c5-461d-9f3c-f00edcab1559

Container runtime version: containerd://1.7.27-1

Taints and Labels

Taints: None

Labels: agentpool: agentpool, beta.kubernetes.io/arch: amd64, beta.kubernetes.io/instance-type: Standard\_D4as\_v5, beta.kubernetes.io/os: linux, failure-domain.beta.kubernetes.io/region: australiacentral, failure-domain.beta.kubernetes.io/zone: 0, kubernetes.azure.com/agentpool: agentpool, kubernetes.azure.com/azure-cni-overlay: true, kubernetes.azure.com/MC\_IngressAKS\_group\_IngressAKS\_austr..., kubernetes.azure.com/consolidated-a-: d9aa0844-3a82-11f0-a1d1..., kubernetes.azure.com/kubelet-id-: 83f7d7c1-a02c-4944-8774-0..., kubernetes.azure.com/mode: system, kubernetes.azure.com/network-name: aks-vnet-84752779, kubernetes.azure.com/network-resourcegroup: IngressAKS\_group, kubernetes.azure.com/network-subnet: aks-subnet, kubernetes.azure.com/network-: 42a59440-5c60-4a84-9349-f..., kubernetes.azure.com/node-: AKSUbuntu-2204gen2container..., kubernetes.azure.com/nodenetw-: 042a1693-d6bf-47d4-a896-86..., kubernetes.azure.com/nodepool-type: VirtualMachineScaleSets, kubernetes.azure.com/os-sku: Ubuntu, kubernetes.azure.com/podnetwork-type: overlay, kubernetes.azure.com/role: agent, kubernetes.azure.com/storageprofile: managed, kubernetes.azure.com/storagegetier: Premium\_LRS

Add to resource dashboard by clicking 'Pin' in top left

Add to resource dashboard by clicking 'Pin' in top left