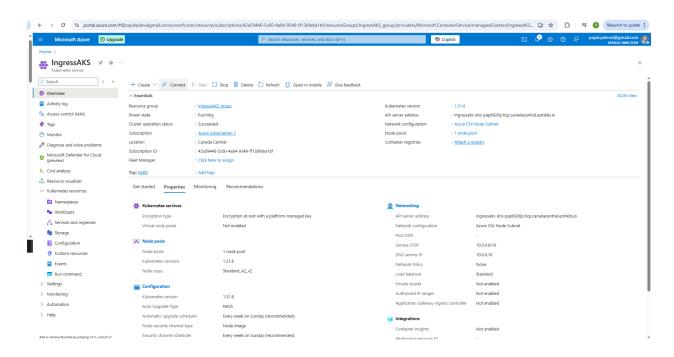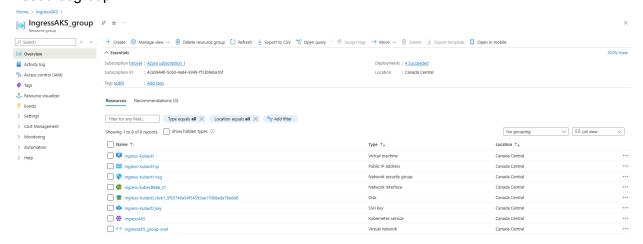# Ingress & Nginx Ingress Controller Architecture

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.
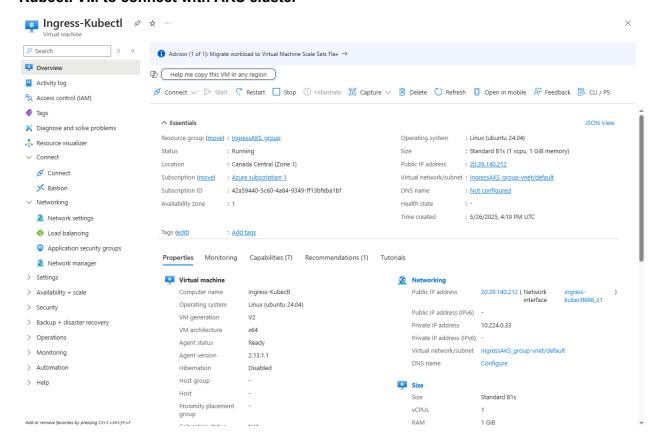
AKS Cluster
Name : IngressAKS

Resourcegroup

Kubectl VM to connect with AKS cluster

Connect to Kubernetes using Azure CLI

az aks get-credentials --resource-group IngressAKS_group --name IngressAKS
--overwrite-existing

Create a Namespace to deploy the Nginxcontroller

Create Admission Controller Roles & Service Account

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
```

```yaml
    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
```

```yaml
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:
```

```yaml
  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
```

```yaml
---
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

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

```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
```

```yaml
      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
```

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

```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:
```

```yaml
        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:

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:

LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tCk1JSUJkakNDQVJ1Z0F3SUJBZ0lRYy9MbTIY
c2hiTC80dXBUZzFpbU1QakFLQmdncWhrak9QUVFEQWpBUE1HcKQ3dZRFZRUUtFd1J1
YVd3eE1DQVhEVEVkxTURVeU5qRTNNekV5TTFvWUR6SXhhNalV3TlRRBeU1UY3pNVEl6V2pBU
ApNUTB3Q3dZRFZRUUtFd1J1YVd3eE1Ga3dFd1lIS29aSXpqMENBUVlJS29aSXpqMERBUW
NEUWdBRVZwNjdIClhmQlVVNEJUU1Bb0dDQ3dZRFZRUUtZ3dqVjB3UWN4RjFVVVQrc2wrWlpvT0dR
vQml3RUtBc0NtQkEzRWdUZZUJ3MGF4S3VXdVMKczY4MU5pSngwwdHRoTXFOWE1GVXdEZZ1
lEVllwUEFSC9CQVFEQWdJCTUdBMVVkSlFTU1Bb0dDQ3NHQVFVRgpCd01CCTUE4R
0ExVWRFd0VCL3dRRk1BTUJBZjh3SFFZRFZSME9CQllFRkduducklPUW5wWENqN1cyanhjRzN
6dXg3CjJ6ZjJNQW9HQ0NxR1NNNDlCQU1DQTBrQU1FWUNJUUNbUwwwMWxNRnnR4K0VB
RzRtWmE4TmN6b1lLbjdaaU29uTmEKOUtdVJPeXY5Z0loQU14RUxPdUN0YkdN2ZydFJLY3
dTQWtjZXo5WDMwY2U2a0ptMDIvdWgxTWwKLS0tLS1FTkQgQ0VSVElGSUNBVEUtLS0tLQo
=

    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]$ 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

```yaml
  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
```

```yaml
    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
```

```yaml
  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
```

```yaml
      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:
```

```yaml
      - 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
```

```yaml
    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

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:

```yaml
      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
```

```yaml
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

404 Not Found

nginx

Welcome to nginx!

If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

```
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>
```

Home > Kubernetes services > IngressAKS

IngressAKS | Services and ingresses
Kubernetes service

Services | Ingresses

Filter by namespace

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	10.40.184	20.248.118.19	80:32483/TCP,4...	52 minutes
ingress-nginx-controller-admission	ingress-nginx	Ok	ClusterIP	10.187.220		443/TCP	52 minutes
nginx-metrics	app-routing-system	Ok	ClusterIP	10.0.159.26		10254/TCP	39 minutes

Home > Kubernetes services > IngressAKS

Kubernetes services ⋯
Default Directory (payalpdevalgmail.onmicrosoft.c...

IngressAKS | Namespaces
Kubernetes service

⋯ Group by none ∨

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

☐ Name ↑
☐ IngressAKS

Search

+ Create ∨   🗑 Delete   ↻ Refresh   Show labels   Give feedback

- 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

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 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

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

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

agentpool | Overview ⋯
Node pool

Search

- Overview
- Nodes
- Configuration

↑ Upgrade Kubernetes   Update image   ↗ Scale node pool   🗑 Delete   ↻ Refresh   Give feedback

Essentials

Provisioning state ℹ	: Succeeded	Cluster	: IngressAKS
Power state ℹ	: Running	Operating system	: Ubuntu Linux
Availability zones	: None	Kubernetes version	: 1.31.8
Mode	: System	Target node count ℹ	: 1 node
		Ready nodes ℹ	: 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

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

aks-agentpool-23298611-vmss000000 | Overview
Node

🔍 Search   ⤢ «

- Overview
- Pods
- YAML
- Events

🔄 Refresh   🔗 Connect   📟 Give feedback

∧ Essentials

Status ⓘ	: Ready	Cluster	: IngressAKS
Node pool	: agentpool	Kubernetes version	: 1.31.8
Region	: australiacentral	Host name	: aks-agentpool-23298611-vmss000000
Internal IP	: 10.224.0.4	Node image version	: AKSUbuntu-2204gen2containerd-202505.14.0
External IP	: N/A		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	25b715118742460bba9b6367464e15c2
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... : d9ae0844-3a82-11f0-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-s... : 42a59440-5c60-4a84-9349-ff... ⧉

kubernetes.azure.com/node-i... : 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/storagetier : Premium_LRS ⧉