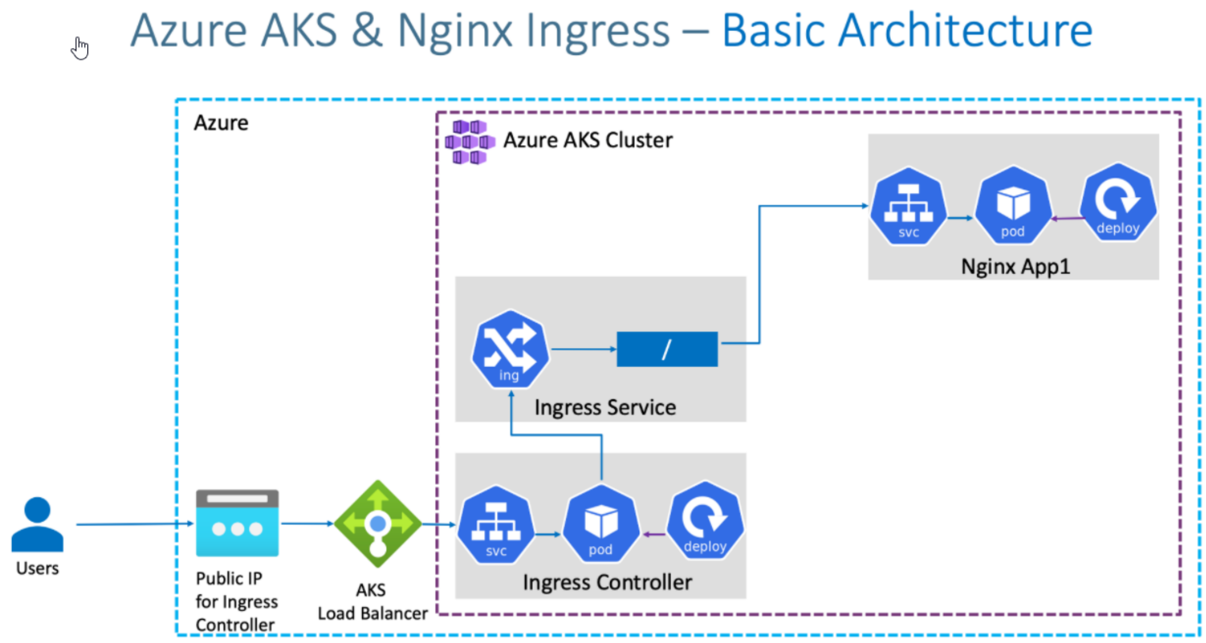
Azure Ingress

Ingress Basic Architecture



Step-01: Introduction

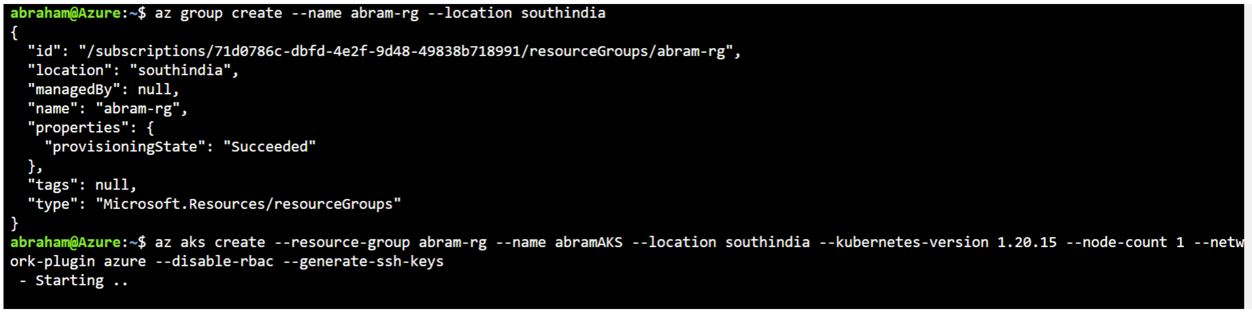
Objectives:

- We are going to create a Static Public IP for Ingress in Azure AKS
- Associate that Public IP to Ingress Controller during installation.
- We are going to create a namespace `ingress-basic` for Ingress Controller where all ingress controller related things will be placed.
- In future, we will install cert-manager for SSL certificates also in the same namespace.
- Caution Note: This namespace is for Ingress controller stuff, ingress resource we can create in any other namespaces and not an issue. Only condition is create ingress resource and ingress pointed application in same namespace (Example: App1 and Ingress resource of App1 should be in same namespace)
- Create / Review Ingress Manifest

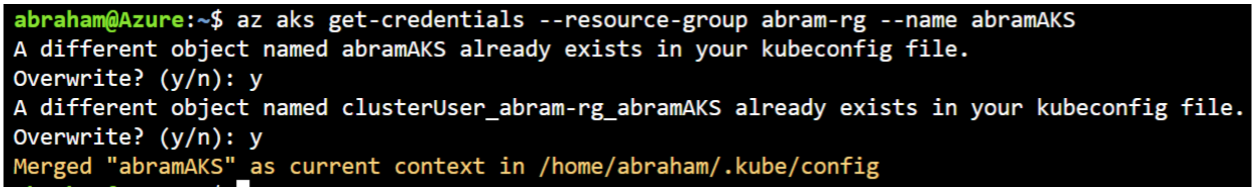
- Deploy a simple Nginx App1 with Ingress manifest and test it
- Clean-Up or delete application after testing

Step-02: Create Static Public IP

Creating a cluster
RG = abram-rg
ClusterName = abramAKS

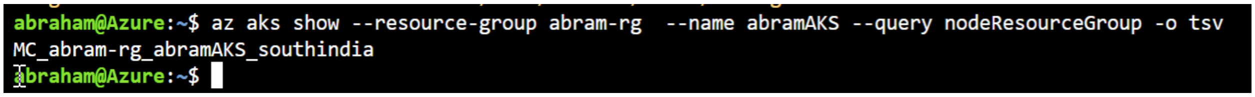


```
abraham@Azure:~$ az group create --name abram-rg --location southindia
{
  "id": "/subscriptions/71d0786c-dbfd-4e2f-9d48-49838b718991/resourceGroups/abram-rg",
  "location": "southindia",
  "managedBy": null,
  "name": "abram-rg",
  "properties": {
    "provisioningState": "Succeeded"
  },
  "tags": null,
  "type": "Microsoft.Resources/resourceGroups"
}
abraham@Azure:~$ az aks create --resource-group abram-rg --name abramAKS --location southindia --kubernetes-version 1.20.15 --node-count 1 --netw
ork-plugin azure --disable-rbac --generate-ssh-keys
 - Starting ..
```



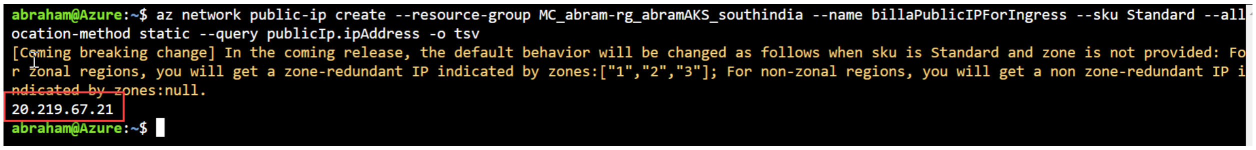
```
abraham@Azure:~$ az aks get-credentials --resource-group abram-rg --name abramAKS
A different object named abramAKS already exists in your kubeconfig file.
Overwrite? (y/n): y
A different object named clusterUser_abram-rg_abramAKS already exists in your kubeconfig file.
Overwrite? (y/n): y
Merged "abramAKS" as current context in /home/abraham/.kube/config
```

```
# Get the resource group name of the AKS cluster
az aks show --resource-group abram-rg  --name abramAKS --query
nodeResourceGroup -o tsv
```



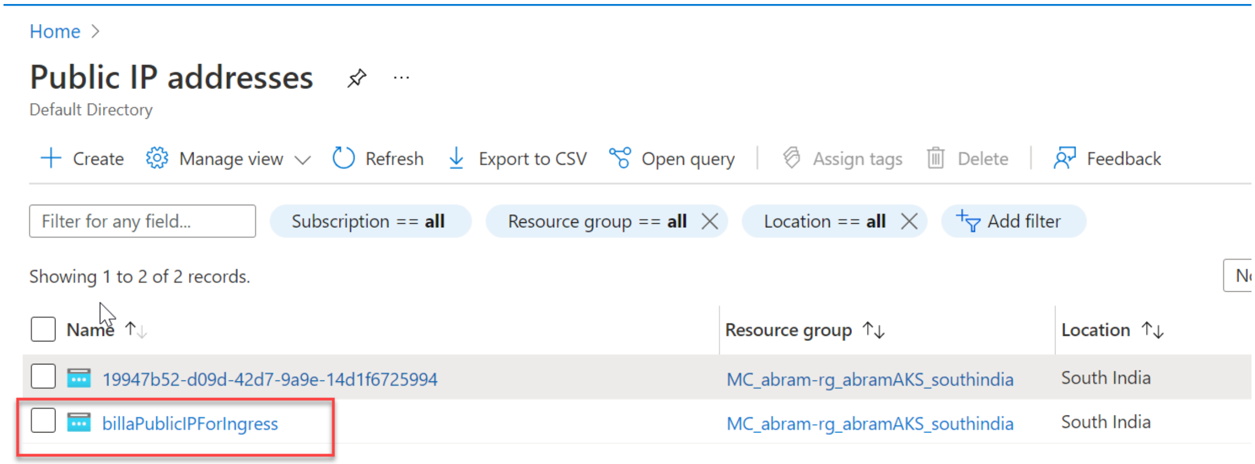
```
abraham@Azure:~$ az aks show --resource-group abram-rg  --name abramAKS --query nodeResourceGroup -o tsv
MC_abram-rg_abramAKS_southindia
abraham@Azure:~$
```

```
# TEMPLATE - Create a public IP address with the static allocation
az network public-ip create --resource-group
<REPLACE-OUTPUT-RG-FROM-PREVIOUS-COMMAND> --name myAKSPublicIPForIngress --sku
Standard --allocation-method static --query publicIp.ipAddress -o tsv

# REPLACE - Create Public IP: Replace Resource Group value
az network public-ip create --resource-group MC_aks-rg1_aksdemo1_centralus
--name myAKSPublicIPForIngress --sku Standard --allocation-method static
--query publicIp.ipAddress -o tsv
```



```
abraham@Azure:~$ az network public-ip create --resource-group MC_abram-rg_abramAKS_southindia --name billaPublicIPForIngress --sku Standard --all
ocation-method static --query publicIp.ipAddress -o tsv
[Coming breaking change] In the coming release, the default behavior will be changed as follows when sku is Standard and zone is not provided: Fo
r zonal regions, you will get a zone-redundant IP indicated by zones:["1","2","3"]; For non-zonal regions, you will get a non zone-redundant IP i
ndicated by zones:null.
20.219.67.21
abraham@Azure:~$
```

The public IP : 20.219.67.21



Public IP addresses

Default Directory

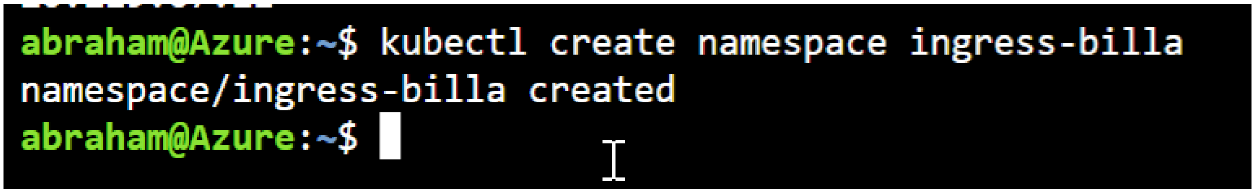
Showing 1 to 2 of 2 records.

□ Name ↑↓	Resource group ↑↓	Location ↑↓
□ 19947b52-d09d-42d7-9a9e-14d1f6725994	MC_abram-rg_abramAKS_southindia	South India
□ billaPublicIPForIngress	MC_abram-rg_abramAKS_southindia	South India

- Make a note of Static IP which we will use in next step when installing Ingress Controller

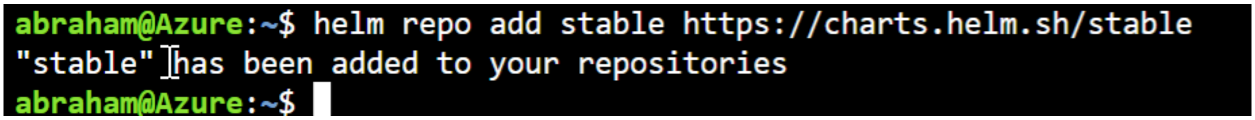
Step-03: Install Ingress Controller

```
# Install Helm3 (if not installed)
brew install helm

# Create a namespace for your ingress resources
kubectl create namespace ingress-billa
```

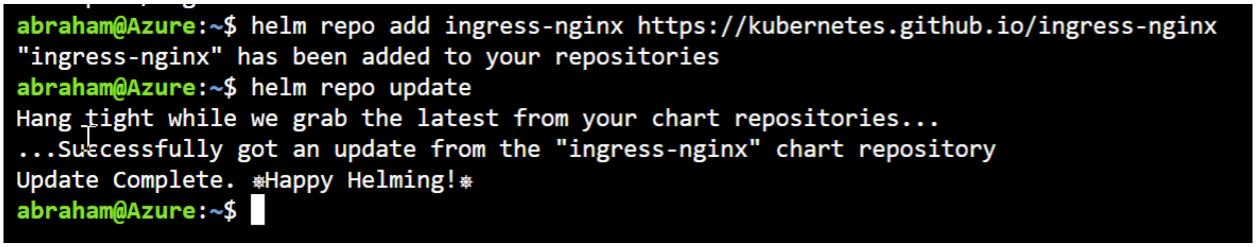


```
abraham@Azure:~$ kubectl create namespace ingress-billa
namespace/ingress-billa created
abraham@Azure:~$
```

```
# Add the official stable repository
helm repo add ingress-nginx https://kubernetes.github.io/ingress-nginx
helm repo add stable https://charts.helm.sh/stable
helm repo update
```

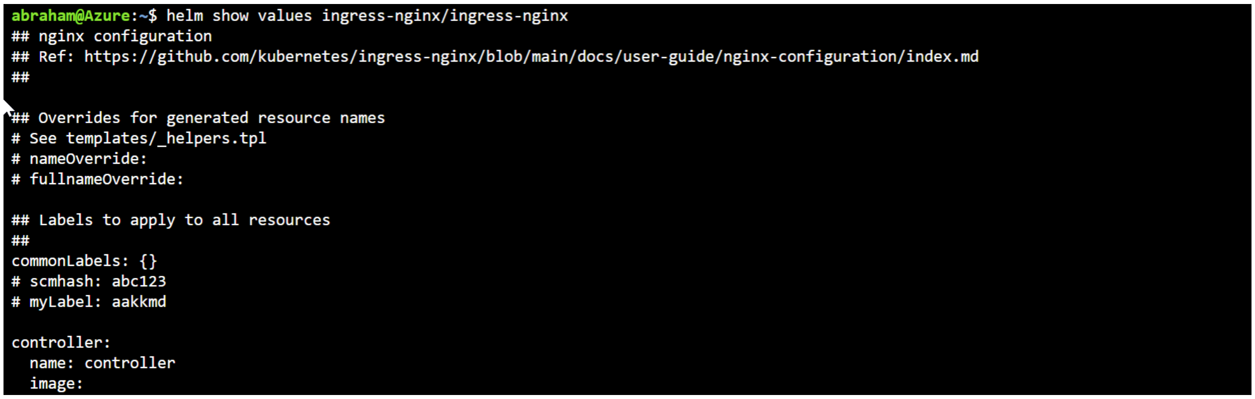


```
abraham@Azure:~$ helm repo add stable https://charts.helm.sh/stable
"stable" has been added to your repositories
abraham@Azure:~$
```



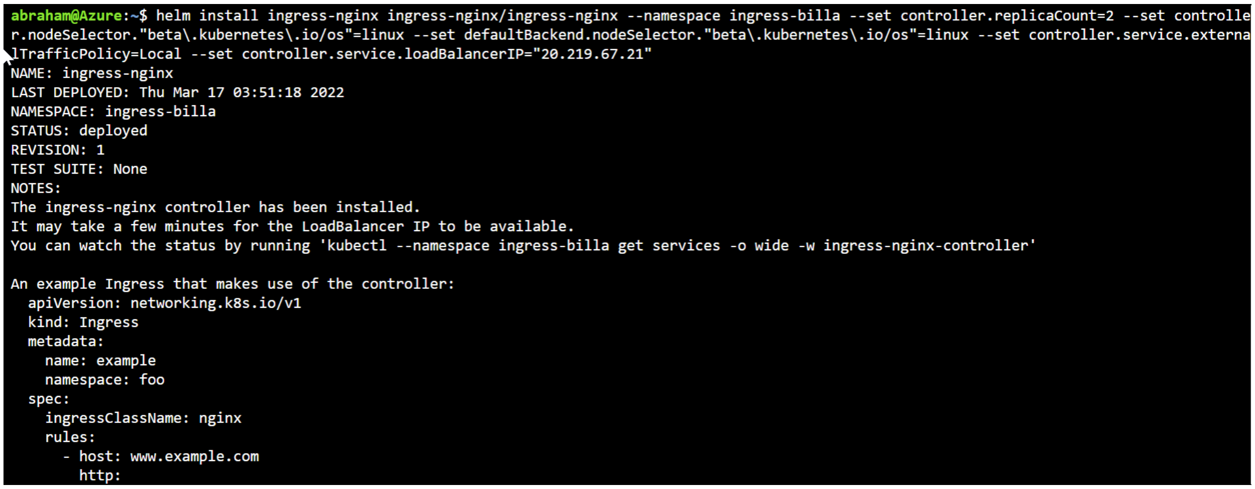
```
abraham@Azure:~$ helm repo add ingress-nginx https://kubernetes.github.io/ingress-nginx
"ingress-nginx" has been added to your repositories
abraham@Azure:~$ helm repo update
Hang tight while we grab the latest from your chart repositories...
...Successfully got an update from the "ingress-nginx" chart repository
Update Complete. ⎈Happy Helming!⎈
abraham@Azure:~$
```

```
#  Customizing the Chart Before Installing.
helm show values ingress-nginx/ingress-nginx
```



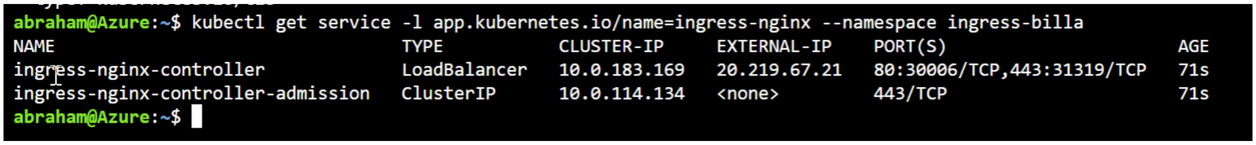
```
abraham@Azure:~$ helm show values ingress-nginx/ingress-nginx
## nginx configuration
## Ref: https://github.com/kubernetes/ingress-nginx/blob/main/docs/user-guide/nginx-configuration/index.md
##

## Overrides for generated resource names
# See templates/_helpers.tpl
# nameOverride:
# fullnameOverride:

## Labels to apply to all resources
##
commonLabels: {}
# scmhash: abc123
# myLabel: aakkmd

controller:
  name: controller
  image:
```

```
# Use Helm to deploy an NGINX ingress controller
helm install ingress-nginx ingress-nginx/ingress-nginx \
    --namespace ingress-basic \
    --set controller.replicaCount=2 \
    --set controller.nodeSelector."beta\.kubernetes\.io/os"=linux \
    --set defaultBackend.nodeSelector."beta\.kubernetes\.io/os"=linux \
    --set controller.service.externalTrafficPolicy=Local \
    --set controller.service.loadBalancerIP="REPLACE_STATIC_IP"

# Replace Static IP captured in Step-02
helm install ingress-nginx ingress-nginx/ingress-nginx --namespace
ingress-billa --set controller.replicaCount=2 --set
controller.nodeSelector."beta\.kubernetes\.io/os"=linux --set
defaultBackend.nodeSelector."beta\.kubernetes\.io/os"=linux --set
controller.service.externalTrafficPolicy=Local --set
controller.service.loadBalancerIP="20.219.67.21"
```



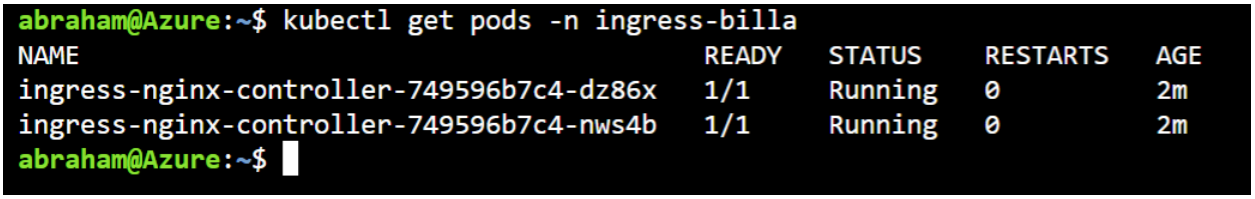
```
abraham@Azure:~$ helm install ingress-nginx ingress-nginx/ingress-nginx --namespace ingress-billa --set controller.replicaCount=2 --set controlle
r.nodeSelector."beta\.kubernetes\.io/os"=linux --set defaultBackend.nodeSelector."beta\.kubernetes\.io/os"=linux --set controller.service.externa
lTrafficPolicy=Local --set controller.service.loadBalancerIP="20.219.67.21"
NAME: ingress-nginx
LAST DEPLOYED: Thu Mar 17 03:51:18 2022
NAMESPACE: ingress-billa
STATUS: deployed
REVISION: 1
TEST SUITE: None
NOTES:
The ingress-nginx controller has been installed.
It may take a few minutes for the LoadBalancer IP to be available.
You can watch the status by running 'kubectl --namespace ingress-billa get services -o wide -w ingress-nginx-controller'

An example Ingress that makes use of the controller:
  apiVersion: networking.k8s.io/v1
  kind: Ingress
  metadata:
    name: example
    namespace: foo
  spec:
    ingressClassName: nginx
    rules:
      - host: www.example.com
        http:
```

```
# List Services with labels
kubectl get service -l app.kubernetes.io/name=ingress-nginx --namespace
ingress-billa
```



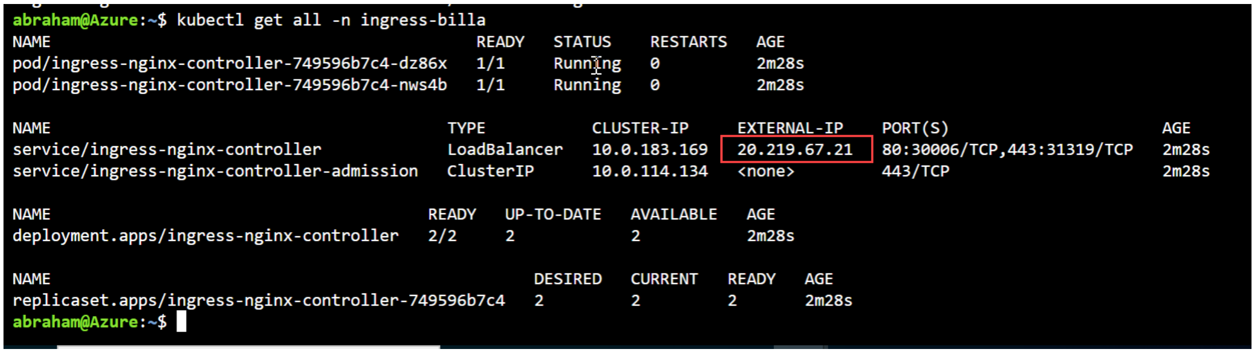
```
abraham@Azure:~$ kubectl get service -l app.kubernetes.io/name=ingress-nginx --namespace ingress-billa
NAME                                 TYPE           CLUSTER-IP     EXTERNAL-IP    PORT(S)                      AGE
ingress-nginx-controller             LoadBalancer   10.0.183.169   20.219.67.21   80:30006/TCP,443:31319/TCP   71s
ingress-nginx-controller-admission   ClusterIP      10.0.114.134   <none>         443/TCP                      71s
abraham@Azure:~$
```

```
# List Pods
kubectl get pods -n ingress-billa
```



```
abraham@Azure:~$ kubectl get pods -n ingress-billa
NAME                                        READY   STATUS    RESTARTS   AGE
ingress-nginx-controller-749596b7c4-dz86x   1/1     Running   0          2m
ingress-nginx-controller-749596b7c4-nws4b   1/1     Running   0          2m
abraham@Azure:~$
```

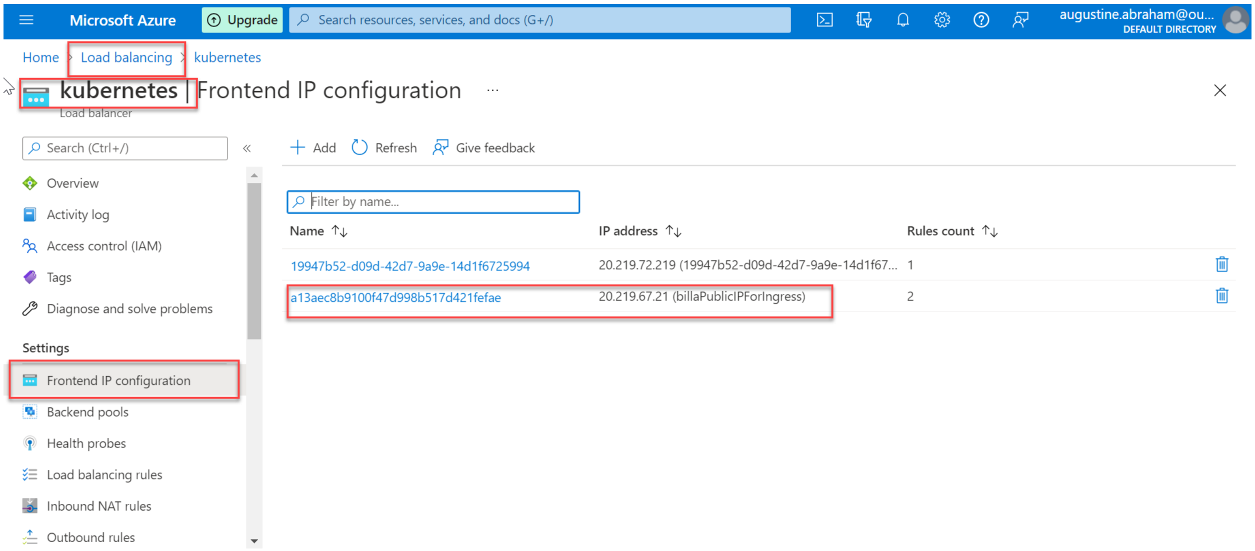
```
kubectl get all -n ingress-billa
```



```
abraham@Azure:~$ kubectl get all -n ingress-billa
NAME                                            READY   STATUS    RESTARTS   AGE
pod/ingress-nginx-controller-749596b7c4-dz86x   1/1     Running   0          2m28s
pod/ingress-nginx-controller-749596b7c4-nws4b   1/1     Running   0          2m28s

NAME                                         TYPE           CLUSTER-IP     EXTERNAL-IP    PORT(S)                      AGE
service/ingress-nginx-controller             LoadBalancer   10.0.183.169   20.219.67.21   80:30006/TCP,443:31319/TCP   2m28s
service/ingress-nginx-controller-admission   ClusterIP      10.0.114.134   <none>         443/TCP                      2m28s

NAME                                       READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/ingress-nginx-controller   2/2     2            2           2m28s

NAME                                                  DESIRED   CURRENT   READY   AGE
replicaset.apps/ingress-nginx-controller-749596b7c4   2         2         2       2m28s
abraham@Azure:~$
```

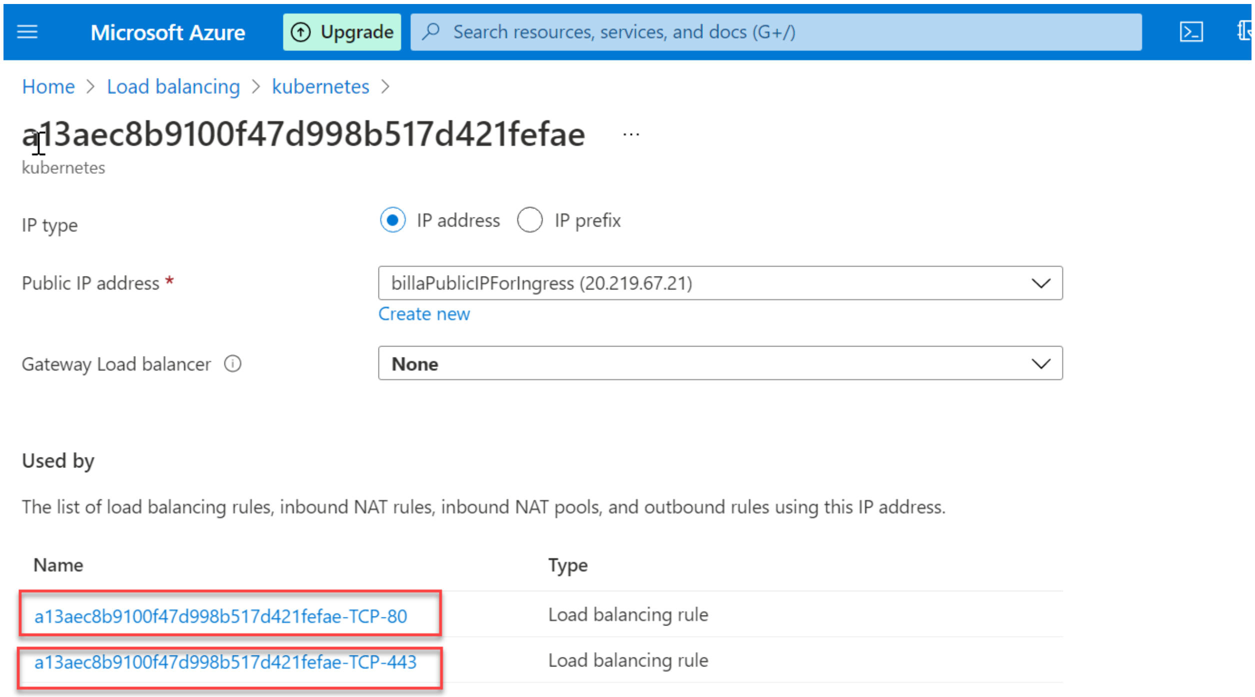
```
# Access Public IP
http://<Public-IP-created-for-Ingress>

# Output should be
```

```
404 Not Found from Nginx

# Verify Load Balancer on Azure Mgmt Console
Primarily refer Settings -> Frontend IP Configuration
```



kubernetes | Frontend IP configuration

Name	IP address	Rules count
19947b52-d09d-42d7-9a9e-14d1f6725994	20.219.72.219 (19947b52-d09d-42d7-9a9e-14d1f67...	1
a13aec8b9100f47d998b517d421fefae	20.219.67.21 (billaPublicIPForIngress)	2


Microsoft Azure
Upgrade
Search resources, services, and docs (G+/)
Home > Load balancing > kubernetes >
Create new

a13aec8b9100f47d998b517d421fefae

kubernetes

IP type: ● IP address ○ IP prefix

Public IP address *: billaPublicIPForIngress (20.219.67.21)

Gateway Load balancer: None

Used by

The list of load balancing rules, inbound NAT rules, inbound NAT pools, and outbound rules using this IP address.

Name	Type
a13aec8b9100f47d998b517d421fefae-TCP-80	Load balancing rule
a13aec8b9100f47d998b517d421fefae-TCP-443	Load balancing rule

Step-04: Review Application k8s manifests

- Ingress-Basic.yml

```
abraham@Azure:~/ingress-deployment$ cat Ingress-Basic.yml
apiVersion: networking.k8s.io/v1beta1
kind: Ingress
metadata:
  name: nginxapp1-ingress-service
  annotations:
    kubernetes.io/ingress.class: nginx
spec:
  rules:
  - http:
      paths:
      - path: /
        backend:
          serviceName: app1-nginx-clusterip-service
          servicePort: 80
```

- NginxApp1-Deployment.yml

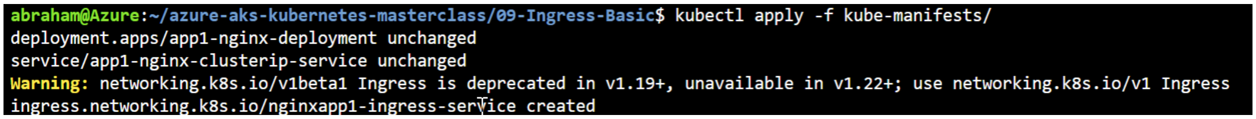
```
$ cat 01-NginxApp1-Deployment.yml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: app1-nginx-deployment
  labels:
    app: app1-nginx
spec:
  replicas: 1
  selector:
    matchLabels:
      app: app1-nginx
  template:
    metadata:
      labels:
        app: app1-nginx
    spec:
      containers:
        - name: app1-nginx
          image: stacksimplify/kube-nginxapp1:1.0.0
          ports:
            - containerPort: 80
```

- NginxApp1-ClusterIP-Service.yml

```
$ cat 02-NginxApp1-ClusterIP-Service.yml
apiVersion: v1
kind: Service
metadata:
  name: app1-nginx-clusterip-service
  labels:
    app: app1-nginx
spec:
  type: ClusterIP
  selector:
    app: app1-nginx
  ports:
    - port: 80
      targetPort: 80
```

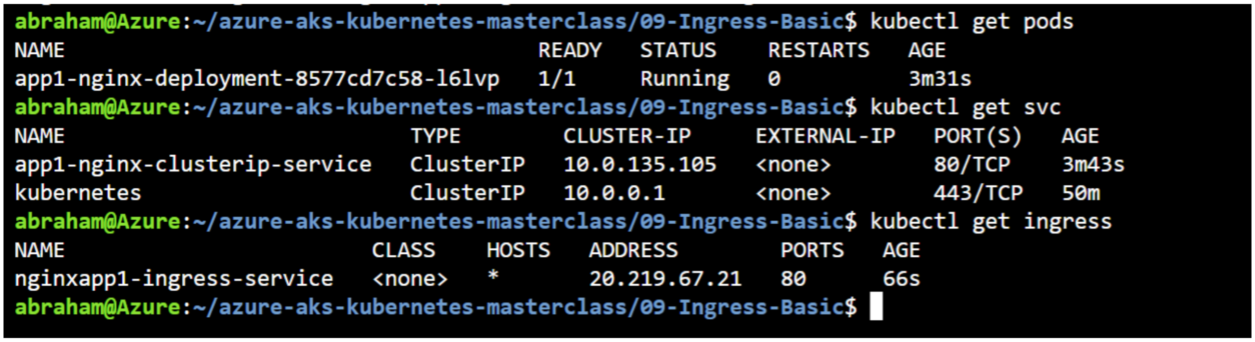
Step-05: Deploy Application k8s manifests and verify

```
# Deploy
kubectl apply -f ingress-deployment/
```



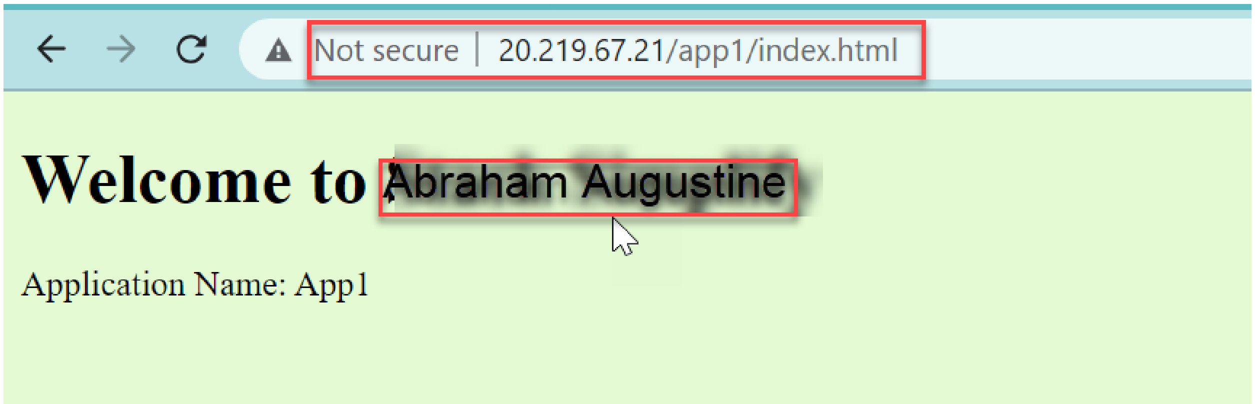
```
abraham@Azure:~/azure-aks-kubernetes-masterclass/09-Ingress-Basic$ kubectl apply -f kube-manifests/
deployment.apps/app1-nginx-deployment unchanged
service/app1-nginx-clusterip-service unchanged
Warning: networking.k8s.io/v1beta1 Ingress is deprecated in v1.19+, unavailable in v1.22+; use networking.k8s.io/v1 Ingress
ingress.networking.k8s.io/nginxapp1-ingress-service created
```

```
# List Pods
kubectl get pods

# List Services
kubectl get svc

# List Ingress
kubectl get ingress
```



```
abraham@Azure:~/azure-aks-kubernetes-masterclass/09-Ingress-Basic$ kubectl get pods
NAME                                     READY   STATUS    RESTARTS   AGE
app1-nginx-deployment-8577cd7c58-l6lvp   1/1     Running   0          3m31s
abraham@Azure:~/azure-aks-kubernetes-masterclass/09-Ingress-Basic$ kubectl get svc
NAME                            TYPE        CLUSTER-IP     EXTERNAL-IP   PORT(S)   AGE
app1-nginx-clusterip-service    ClusterIP   10.0.135.105   <none>        80/TCP    3m43s
kubernetes                      ClusterIP   10.0.0.1       <none>        443/TCP   50m
abraham@Azure:~/azure-aks-kubernetes-masterclass/09-Ingress-Basic$ kubectl get ingress
NAME                         CLASS    HOSTS   ADDRESS        PORTS   AGE
nginxapp1-ingress-service    <none>   *       20.219.67.21   80      66s
abraham@Azure:~/azure-aks-kubernetes-masterclass/09-Ingress-Basic$
```

```
# Access Application
http://<Public-IP-created-for-Ingress>/app1/index.html
```



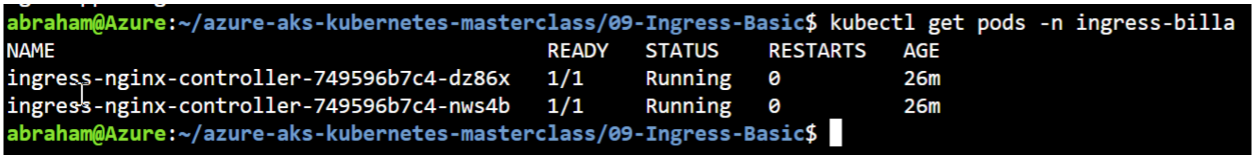
Not secure | 20.219.67.21/app1/index.html

Welcome to Abraham Augustine

Application Name: App1

```
http://<Public-IP-created-for-Ingress>
```

```
# Verify Ingress Controller Logs
kubectl get pods -n ingress-billa
```

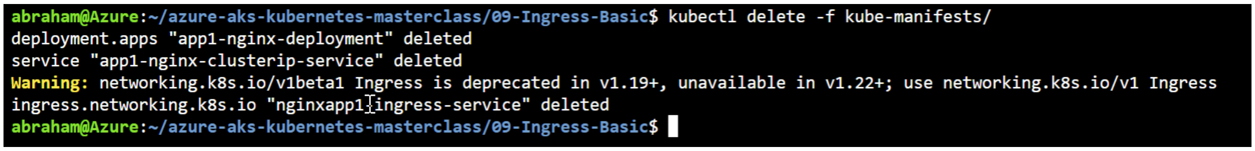


```
abraham@Azure:~/azure-aks-kubernetes-masterclass/09-Ingress-Basic$ kubectl get pods -n ingress-billa
NAME                                       READY   STATUS    RESTARTS   AGE
ingress-nginx-controller-749596b7c4-dz86x  1/1     Running   0          26m
ingress-nginx-controller-749596b7c4-nws4b  1/1     Running   0          26m
abraham@Azure:~/azure-aks-kubernetes-masterclass/09-Ingress-Basic$
```

```
kubectl logs -f <pod-name> -n ingress-billa
```

Step-06: Clean-Up Apps

```
# Delete Apps
kubectl delete -f kube-manifests/
```



```
abraham@Azure:~/azure-aks-kubernetes-masterclass/09-Ingress-Basic$ kubectl delete -f kube-manifests/
deployment.apps "app1-nginx-deployment" deleted
service "app1-nginx-clusterip-service" deleted
Warning: networking.k8s.io/v1beta1 Ingress is deprecated in v1.19+, unavailable in v1.22+; use networking.k8s.io/v1 Ingress
ingress.networking.k8s.io "nginxapp1-ingress-service" deleted
abraham@Azure:~/azure-aks-kubernetes-masterclass/09-Ingress-Basic$
```