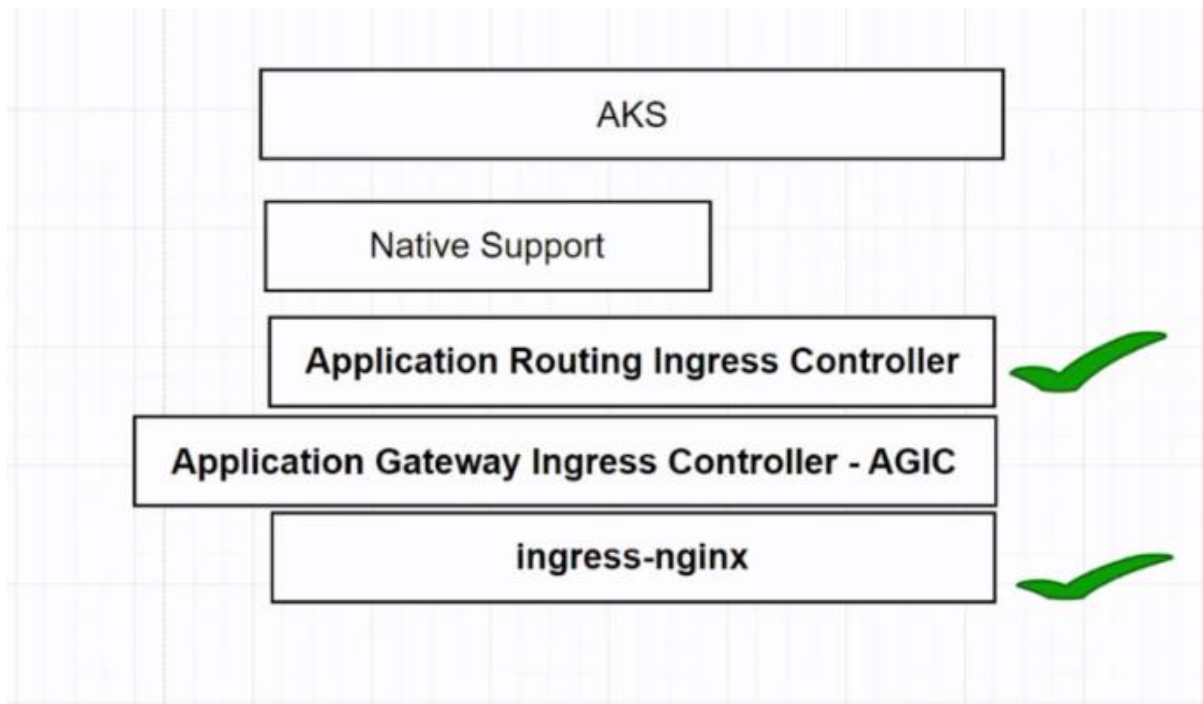


17 Nov 2024

AGENDA -



1) Create folder “9) 17 November Kubernetes” and open with vs code and connect to cluster.

az login

az account set --subscription 48f88df7-0d53-4866-a66f-82eb0ac469e3

az aks get-credentials --resource-group rgcloud --name k8scloud --overwrite-existing

2) ENABLE - enable AGIC in aks

<https://learn.microsoft.com/en-us/azure/application-gateway/tutorial-ingress-controller-add-on-existing>

Enable the AGIC add-on in existing AKS cluster through Azure CLI

If you'd like to continue using Azure CLI, you can continue to enable the AGIC add-on in the AKS cluster you created, **myCluster**, and specify the AGIC add-on to use the existing application gateway you created, **myApplicationGateway**.

```
Azure CLI Copy Open Cloud Shell  
  
appgwId=$(az network application-gateway show --name myApplicationGateway --res  
az aks enable-addons --name myCluster --resource-group myResourceGroup --addon
```

`az aks enable-addons --name k8scloud --resource-group rgcloud --addon ingress-appgw --appgw-id $appgwId`

`az aks enable-addons --name k8scloud --resource-group rgcloud --addon ingress-appgw --appgw-subnet-cidr 10.224.0.0/16`

`az aks enable-addons --name k8scloud --resource-group rgcloud --addon ingress-appgw --appgw-subnet-cidr 10.225.0.0/16`

```
Code: AddonInvalid  
Message: Application Gateway Ingress Controller addon is not supported with Azure CNI Overlay  
Target: networkProfile.networkPluginMode
```

So we have create new cluster as this cluster is not supported.

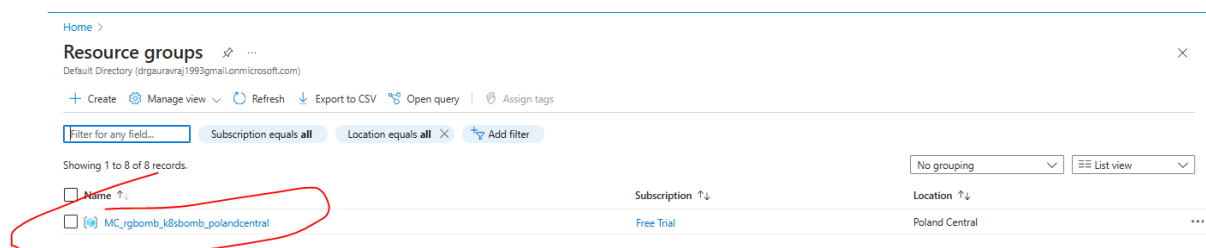
3) SEARCH – azure aks networking plugins

NOTE: 1) How do we expose anything in cluster = through loadbalancer service

2) Limitations of loadbalancer service = for an application only one ip can be used. So for resolving this issue we had brought ingress controller

3) Calico = supports network policy

4) With our cluster an extra rg gets created



Home > Resource groups > MC_rgbomb_k8sbomb_polandcentral

Resource groups

Default Directory (drgauravj1993@gmail.onmicros...

+ Create Manage view ...

Filter for any field...

Name ↑

- MC_rgbomb_k8sbomb_polandcentral
- MC_rghappy_dev-todo-aks-cluster-zzz...
- NetworkWatcherRG
- rgbomb
- rgcloud
- rghappy
- satsishapp
- satrg

Overview

Activity log

Access control (IAM)

Tags

Resource visualizer

Events

Settings

Cost Management

Monitoring

Automation

Help

MC_rgbomb_k8sbomb_polandcentral

Search

+ Create Manage view Delete resource group Refresh Export to CSV Open query Assign tags ...

Essentials

Resources Recommendations

Filter for any field... Type equals all Location equals all Add filter

Showing 1 to 9 of 9 records. Show hidden types No grouping List view

Name	Type	Location
7174b21f-93ca-4a39-ae38-76e1cb4ce19f	Public IP address	Poland Central
aks-agentpool-26165618-nsg	Network security group	Poland Central
aks-agentpool-39931302-vmss	Virtual machine scale set	Poland Central
aks-vnet-26165618	Virtual network	Poland Central
ingress-appgateway	Application gateway	Poland Central
ingress-appgateway-appgwip	Public IP address	Poland Central
ingressapplicationgateway-k8sbomb	Managed Identity	Poland Central

< Previous Page 1 of 1 Next >

Give feedback

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

Home > Resource groups > MC_rgbomb_k8sbomb_polandcentral > aks-vnet-26165618

aks-vnet-26165618 | Subnets

Virtual network

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Settings

Address space

Connected devices

Subnets

Bastion

DDoS protection

Create subnets to segment the virtual network address space into smaller ranges for use by your applications. When you deploy resources into a subnet, Azure assigns the resource an IP address from the subnet.

Search subnets

Name	IPAM Pool	IPv4	IPv6	Available IPs	Delegated to	Security group	Route table
aks-subnet	-	10.224.0.0/16	-	more than ...	-	aks-agentp...	-
ingress-appgateway-s...	-	-	-	-	-	-	-

Edit subnet

Select an address space and configure your subnet. You can customize a default subnet or select from subnet templates if you plan to add select services later. Learn more

Subnet ID /subscriptions/48f88df7-0d53-4866-a66f-82eb0ac469e3/resourceGroups/MC_rgbomb_k8s...

Subnet purpose Default

Name aks-subnet

IPv4

Include an IPv4 address space Address prefix '10.224.0.0/16' cannot be updated because the subnet is in use.

IPv4 address space 10.224.0.0/16

Starting address 10.224.0.0

Size /16

Subnet address range 10.224.0.0 - 10.224.255.255

aks-vnet-13770513 | Subnets

Virtual network

+ Subnet Refresh Manage users Delete

Create subnets to segment the virtual network address space into smaller ranges for use by your applications. When you deploy resources into a subnet, Azure assigns the resource an IP address from the subnet.

Search subnets

Name	IPAM Pool	IPv4	IPv6	Available IPs	Delegated to	Security group	Route table
aks-subnet	-	10.224.0.0/16	-	more than ...	-	aks-agentp...	-

5) **az aks enable-addons --name k8spop --resource-group rgpop --addon ingress-appgw --appgw-subnet-cidr 10.224.0.0/16** = run to add application gateway

Ab isme dusra cidr banalo i.e. 224 ko 226 krdo

az aks enable-addons --name k8sjack --resource-group rgjack --addon ingress-appgw --appgw-subnet-cidr 10.226.0.0/16

```
target: AddonProfiles.ingressapplicationgateway
PS C:\V4\KUBERNETES\9) 17 November Kubernetes> az aks enable-addons --name k8sjack --resource-group rgjack --addon ingress-appgw --appgw-subnet-cidr 10.226.0.0/16
Running ..
```

NOTE: Application gateway gets created on its personal subnet only

6) So now application gateway got created

Home > Resource groups > MC_rgjack_k8sjack_polandcentral > aks-vnet-11857573

aks-vnet-11857573 | Subnets

Virtual network

+ Subnet Refresh Manage users Delete

Create subnets to segment the virtual network address space into smaller ranges for use by your applications. When you deploy resources into a subnet, Azure assigns the resource an IP address from the subnet.

Search subnets

	Name ↑	IPAM Pool ↑	IPv4 ↑	IPv6 ↑	Available IPs ↑	Delegated to ↑	Security gro... ↑	Route table ↑
<input type="checkbox"/>	aks-subnet	-	10.224.0.0/16	-	more than ...	-	aks-agentp...	-
<input type="checkbox"/>	applicationgateway-subnet	-	10.226.0.0/16	-	availability ...	-	-	-

Home > Load balancing

Load balancing | Application Gateway

Search

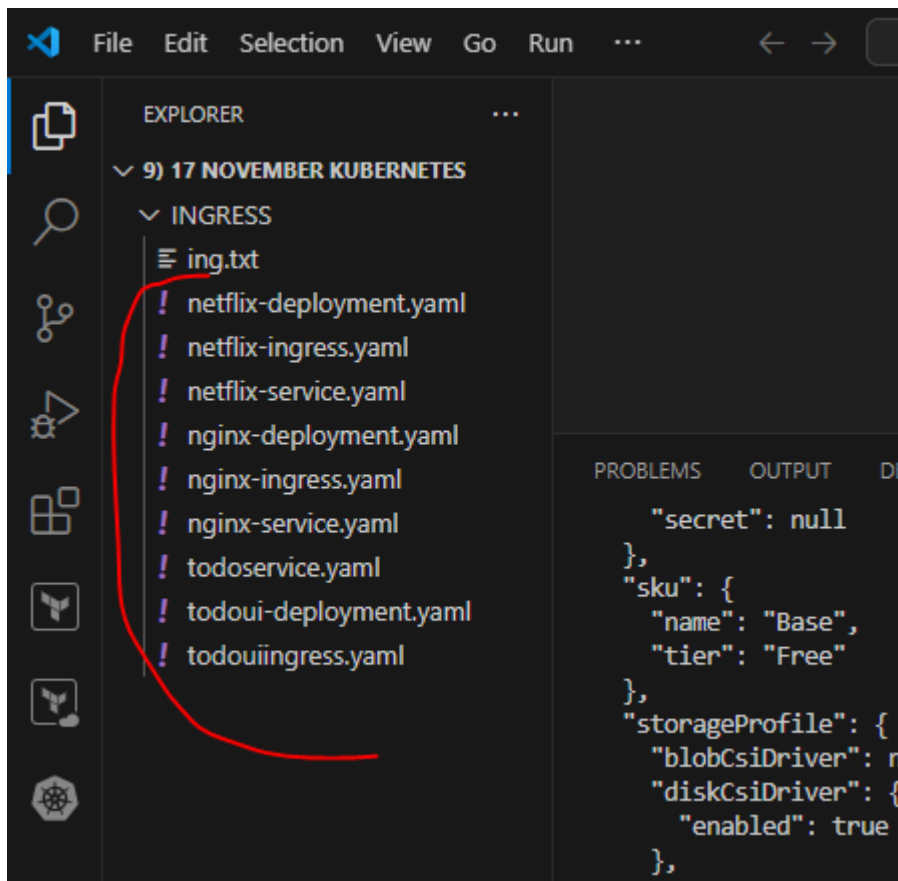
+ Create Manage view Refresh Export to CSV Open query Assign tags

Filter for any field... Subscription equals all Resource group equals all Location equals all Add filter

Showing 1 to 1 of 1 records.

	Name ↑	Public I... ↑	Public I... ↑	Private... ↑	Private... ↑	Resource group ↑	Location ↑	Subscription ↑
<input type="checkbox"/>	applicationgateway	74.248.80.157	-	-	-	MC_rgjack_k8sjack_polan...	Poland Central	Free Trial

7) Copy last class content in today's class



+++++

AGENDA – DO NETFLIX DEPLOYMENT

8) **kubectl apply -f netflix-deployment.yaml**

```
PS C:\4) KUBERNETES\9) 17 November Kubernetes\INGRESS> kubectl apply -f netflix-deployment.yaml
deployment.apps/netflix created
PS C:\4) KUBERNETES\9) 17 November Kubernetes\INGRESS> 
```

9) **kubectl apply -f netflix-ingress.yaml**

```
PS C:\4) KUBERNETES\9) 17 November Kubernetes\INGRESS> kubectl apply -f netflix-ingress.yaml
ingress.networking.k8s.io/netflix-rule created
PS C:\4) KUBERNETES\9) 17 November Kubernetes\INGRESS> 
```

10) **kubectl apply -f netflix-service.yaml**

```
PS C:\4) KUBERNETES\9) 17 November Kubernetes\INGRESS> kubectl apply -f netflix-service.yaml
service/netflix created
PS C:\4) KUBERNETES\9) 17 November Kubernetes\INGRESS> 
```

11) **kubectl get ingressclass** =

```
PS C:\4) KUBERNETES\9) 17 November Kubernetes\INGRESS> kubectl get ingressclass
NAME                                CONTROLLER                PARAMETERS  AGE
azure-application-gateway          azure/application-gateway  <none>      75m
PS C:\4) KUBERNETES\9) 17 November Kubernetes\INGRESS> 
```

12) **kubectl get ingress** = isme ip address nhi hai

```
PS C:\4) KUBERNETES\9) 17 November Kubernetes\INGRESS> kubectl get ingress
NAME      CLASS  HOSTS                ADDRESS  PORTS  AGE
netflix-rule  nginx  netflix.dhondhu.online  <empty>  80     3m33s
PS C:\4) KUBERNETES\9) 17 November Kubernetes\INGRESS> 
```

```
INGRESS > ! netflix-ingress.yaml > {} spec > [ ] rules > {} 0 > {} http > [ ] path
1  apiVersion: networking.k8s.io/v1 #ye ingress rule ki y
2  kind: Ingress
3  metadata:
4    name: netflix-rule
5    labels:
6      name: netflix-rule
7  spec:
8    ingressClassName: nginx
9    rules:
10 - host: netflix.dhondhu.online
11   http:
```

So upar dono commands se pata chala ki humne code me class name – nginx pass kiya hai but “**kubectl get ingressclass**” se pata chala ki real me class ka name “**azure-application-gateway**” hai, to hum is name ko code me pass krenge else purana class rakhne se wo kisi controller se connect nhi hoga to wo faltu pada rhega.

13) So pass **ingressClassName** as **azure-application-gateway** in code

```
! netflix-ingress.yaml 4  name: netflix-rule
! netflix-service.yaml 5  labels:
! nginx-deployment.yaml 6    name: netflix-rule
! nginx-ingress.yaml 7  spec:
! nginx-service.yaml 8    ingressClassName: azure-application-gateway
9  rules:
```

14) **kubectl delete -f netflix-ingress.yaml** = delete old ingress rule of Netflix

```
PS C:\4) KUBERNETES\9) 17 November Kubernetes\INGRESS> kubectl delete -f netflix-ingress.yaml
ingress.networking.k8s.io "netflix-rule" deleted
PS C:\4) KUBERNETES\9) 17 November Kubernetes\INGRESS> |
```

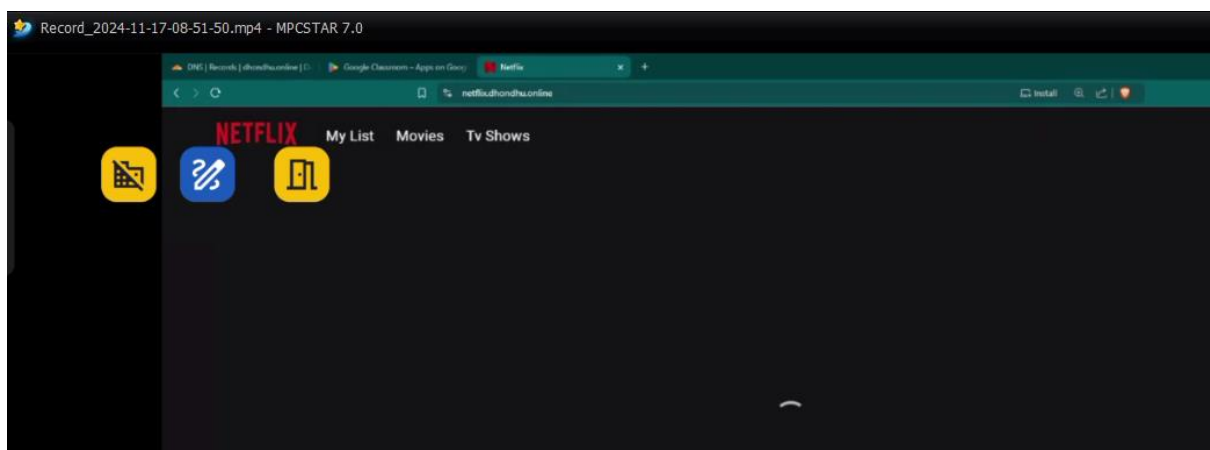
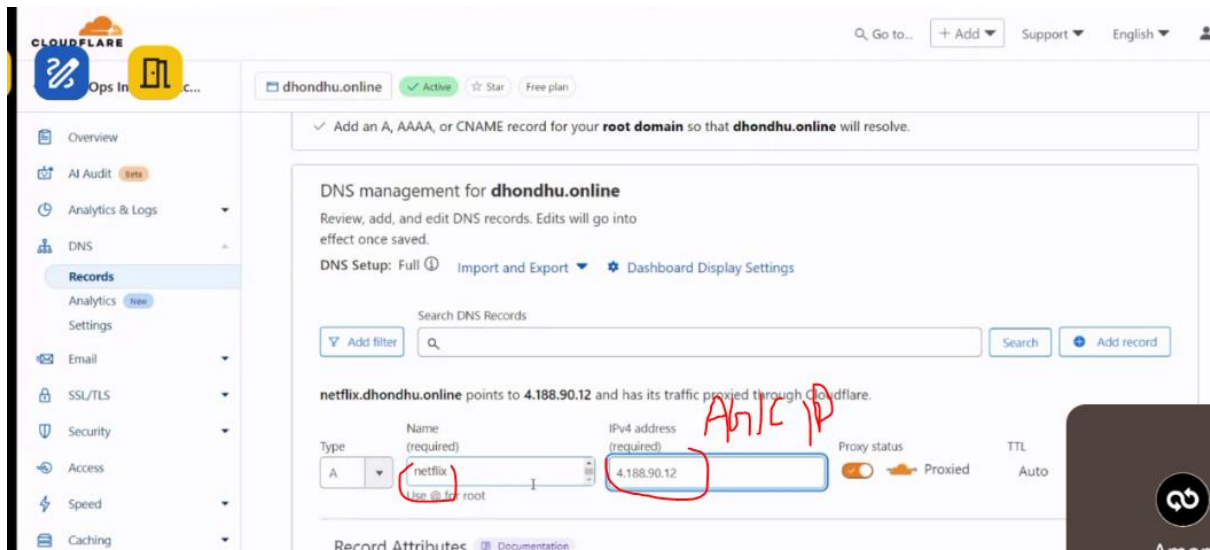
15) **kubectl apply -f netflix-ingress.yaml** = again create with **ingressClassName** as **azure-application-gateway** in code

```
PS C:\4) KUBERNETES\9) 17 November Kubernetes\INGRESS> kubectl apply -f netflix-ingress.yaml
ingress.networking.k8s.io/netflix-rule created
PS C:\4) KUBERNETES\9) 17 November Kubernetes\INGRESS> |
```

16) **kubectl get ingress** = so ab ip address mil gaya hai humare Netflix rule ko

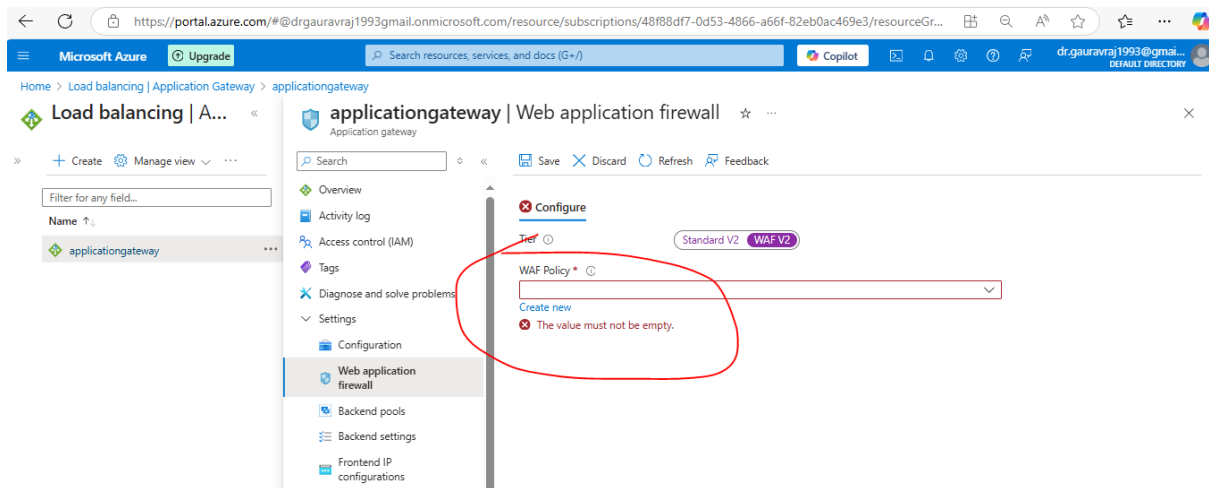
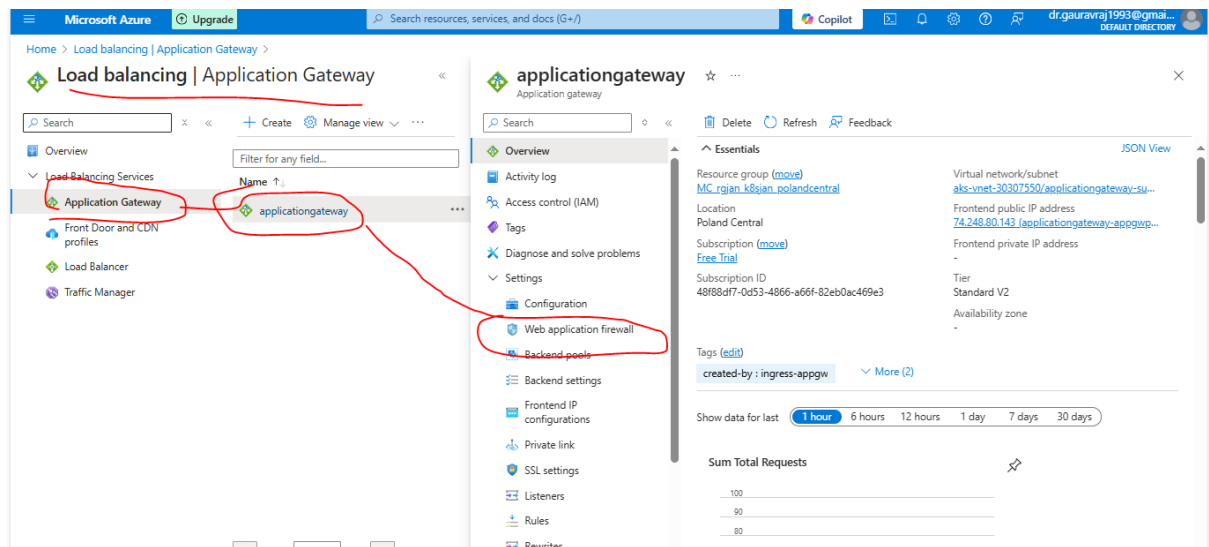
```
PS C:\4) KUBERNETES\9) 17 November Kubernetes\INGRESS> kubectl get ingress
NAME          CLASS                  HOSTS                ADDRESS          PORTS    AGE
netflix-rule  azure-application-gateway netflix.dhondhu.online 74.248.80.157   80       66s
```

17) Go to cloud flare and we can see that AGIC is running

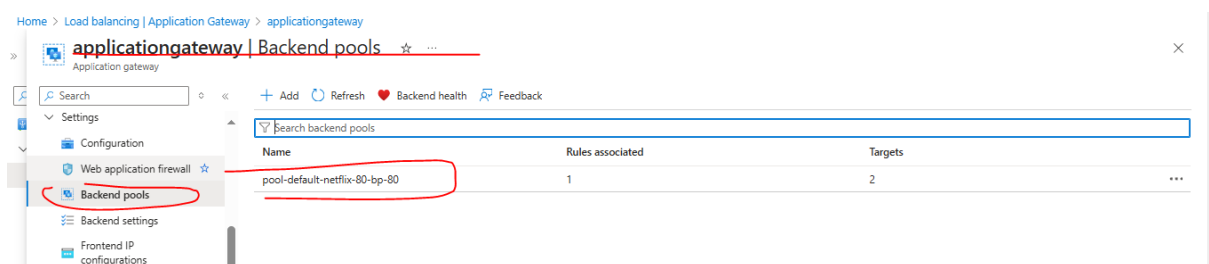


+++++

AGENDA = How to enable WAF on application gateway loadbalancer



1) In Backend pools section, we have pod ip addresses contained in it



```
PS C:\4) KUBERNETES\9) 17a November Kubernetes\INGRESS> kubectl get pods -o wide
NAME                                READY  STATUS   RESTARTS  AGE  IP            NODE                                NOMINATED NODE  READINESS GATES
netflix-6c87bb6d86-v4tr9            1/1    Running  0          67m  10.224.0.100  aks-agentpool-18774052-vmss000003  <none>          <none>
netflix-6c87bb6d86-z2h45            1/1    Running  0          67m  10.224.0.201  aks-agentpool-18774052-vmss000002  <none>          <none>
```

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

Home > Load balancing | Application Gateway > applicationgateway | Backend pools >

Edit backend pool ...

A backend pool is a collection of resources to which your application gateway can send traffic. A backend pool can contain virtual machines, virtual machines scale sets, IP addresses, domain names, or an App Service.

Name
pool-default-netflix-80-bp-80

Add backend pool without targets
Yes No

Backend targets
2 items

Target type	Target	
IP address or FQDN	10.224.0.100	...
IP address or FQDN	10.224.0.201	...
IP address or FQDN		

Associated rule
rr-7932ef741ec5397957dd2160ca25a415

2) In "Frontend IP configurations" section, ingress ip is configured

```
PS C:\4) KUBERNETES\9) 17a November Kubernetes\INGRESS> kubectl get ingress
NAME          CLASS          HOSTS          ADDRESS          PORTS    AGE
netflix-rule  azure-application-gateway  netflix.dhondhu.online  74.248.80.143    80       103m
PS C:\4) KUBERNETES\9) 17a November Kubernetes\INGRESS>
```

Home > Load balancing | Application Gateway > applicationgateway

applicationgateway | Frontend IP configurations ☆

Search frontend IP configurations

Type	Status	Name	IP address	Associated listeners
Public	Configured	appGatewayFrontendIP	74.248.80.143	applicationgateway-appg... fi-7932ef741ec5397957dd2160ca25a415
Private	Not configured	-	-	-

3) In "Listeners" section host url is configured


```

! netflix-deployment.yaml      3
! netflix-ingress.yaml        4
! netflix-service.yaml         5
! nginx-deployment.yaml       6
! nginx-ingress.yaml          7
! nginx-service.yaml          8
! todoservice.yaml            9
! todoui-deployment.yaml     10
                                11
                                metadata:
                                name: netflix-rule
                                labels:
                                name: netflix-rule
                                spec:
                                ingressClassName: azure-application-gateway
                                rules:
                                - host: netflix.dhondhu.online
                                http:

```

Microsoft Azure Upgrade Search resources

Home > Load balancing | Application Gateway > applicationgateway | Listeners >

fl-7932ef741ec5397957dd2160ca25a415 ...

applicationgateway

Listener name ⓘ

fl-7932ef741ec5397957dd2160ca25a415

Frontend IP * ⓘ

Public

Protocol ⓘ

☒ HTTP ☐ HTTPS

Port * ⓘ

80

Associated rule

rr-7932ef741ec5397957dd2160ca25a415

Listener type ⓘ

☐ Basic ☒ Multi site

Host type ⓘ

Single Multiple/Wildcard

Host names *

netflix.dhondhu.online

+++++

AGENDA = todoui application deployment

1) **kubectl apply -f todoui-deployment.yaml**

```

PS C:\4) KUBERNETES\9) 17a November Kubernetes\INGRESS> kubectl apply -f todoui-deployment.yaml
deployment.apps/todoui created
PS C:\4) KUBERNETES\9) 17a November Kubernetes\INGRESS>

```

2) `kubectl apply -f todoservice.yaml`

```
PS C:\4) KUBERNETES\9) 17a November Kubernetes\INGRESS> kubectl apply -f todoservice.yaml
service/todoui created
PS C:\4) KUBERNETES\9) 17a November Kubernetes\INGRESS> █
```

3) `kubectl apply -f todouiingress.yaml`

```
PS C:\4) KUBERNETES\9) 17a November Kubernetes\INGRESS> kubectl apply -f todouiingress.yaml
ingress.networking.k8s.io/todoui-ingress created
PS C:\4) KUBERNETES\9) 17a November Kubernetes\INGRESS> █
```

4) `kubectl get ingressclass`

```
netflix-rule azure-application-gateway netflix.dhondhu.online 74.248.80.143
PS C:\4) KUBERNETES\9) 17a November Kubernetes\INGRESS> kubectl get ingressclass
NAME                                CONTROLLER                PARAMETERS  AGE
azure-application-gateway          azure/application-gateway  <none>      136m
PS C:\4) KUBERNETES\9) 17a November Kubernetes\INGRESS> █
```

5) `Kubectl get ingress` = So no ip is assigned as “`kubectl get ingressclass`” has different class name so pass that in yaml code i.e. “`azure-application-gateway`”

NAME	CLASS	HOSTS	ADDRESS	PORTS	AGE
netflix-rule	azure-application-gateway	netflix.dhondhu.online	74.248.80.143	80	123m
todoui-ingress	webapprouting.kubernetes.azure.com	todoui.dhondhu.online		80	4m46s

```
! nginx-deployment.yaml      6 | | name: todoui-ingress
! nginx-ingress.yaml         7 | | spec:
! nginx-service.yaml         8 | |   ingressClassName: webapprouting.kubernetes.azure.com
! todoservice.yaml           9 | |   rules:
! todoui-deployment.yaml    10 | |     - host: todoui.dhondhu.online
! todouiingress.yaml        11 | |     http:
                                12 | |       paths:
                                13 | |         - pathType: Prefix
```

Change it as below

```
! nginx-deployment.yaml      6 | | name: todoui-ingress
! nginx-ingress.yaml         7 | | spec:
! nginx-service.yaml         8 | |   ingressClassName: azure-application-gateway
! todoservice.yaml           9 | |   rules:
! todoui-deployment.yaml    10 | |     - host: todoui.dhondhu.online
! todouiingress.yaml        11 | |     http:
                                12 | |       paths:
                                13 | |         - pathType: Prefix
                                14 | |         path: "/"
```

6) `kubectl apply -f todouiingress.yaml`

```
netflix-rule azure-application-gateway netflix.dhondhu.online 74.248.80.143
PS C:\4) KUBERNETES\9) 17a November Kubernetes\INGRESS> kubectl apply -f todouiingress.yaml
ingress.networking.k8s.io/todoui-ingress created
PS C:\4) KUBERNETES\9) 17a November Kubernetes\INGRESS> █
```

7) `kubectl get ingress`

```
PS C:\4) KUBERNETES\9) 17a November Kubernetes\INGRESS> kubectl get ingress
NAME                                CLASS                HOSTS                                ADDRESS          PORTS  AGE
netflix-rule                       azure-application-gateway netflix.dhondhu.online 74.248.80.143  80    133m
todoui-ingress                     azure-application-gateway todoui.dhondhu.online 74.248.80.143  80    9s
PS C:\4) KUBERNETES\9) 17a November Kubernetes\INGRESS> █
```

8) So now we can see in portal we have got backend pool, listener, Frontend IP configurations for todoui application also

The screenshot shows the Azure Application Gateway portal. On the left, the 'Backend pools' and 'Listeners' sections are highlighted with red circles. The main area displays two tables:

Name	Rules associated	Targets
pool-default-netflix-80-bp-80	1	2
pool-default-todoui-80-bp-80	1	2

Name	Port	Protocol	Frontend IP	Associated rule	Host name
ft-7932ef741ec5397957dd2160ca2...	80	HTTP	Public IPv4	rr-7932ef741ec5397957dd2160ca...	> netflix.dhondhu.online
ft-05d2980d0cca42439094acbd3...	80	HTTP	Public IPv4	rr-05d2980d0cca42439094acbd3...	> todoui.dhondhu.online

Below the tables, the 'SSL Policy' section is visible, showing the 'Selected SSL Policy' as 'Default' and the 'Min protocol version' as 'TLSv1.0'.

+++++

AGENDA – ADDING AGIC INGRESS CONTROLLER THROUGH PORTAL MANUALLY

1) Go to aks cluster -> Networking -> virtual networking configuration -> manage -> tick ingress controller option -> save

The screenshot shows the Azure Kubernetes Service (AKS) portal for a cluster named 'k8sjan'. The 'Networking' section is highlighted with a red circle in the left sidebar. The main area displays the 'Networking' configuration page, which includes the following details:

- Resource group:** rgjan
- Kubernetes version:** 1.30.6
- Power state:** Running
- Cluster operation status:** Succeeded
- Subscription:** Free Trial
- Location:** Poland Central
- Subscription ID:** 48f8d7f-0d53-4866-a66f-82eb0ac469e3
- Tags:** Add tags
- Network configuration:** Azure CNI Node Subnet
- Node pools:** 1 node pool
- Container registries:** Attach a registry
- Encryption type:** Encryption at-rest with a platform-managed key
- Virtual node pools:** Not enabled
- Node pools:** 1 node pool
- Kubernetes versions:** 1.30.6
- Node sizes:** Standard_DS5_v2
- Network Policy:** Calico
- Load balancer:** Standard
- Private cluster:** Not enabled
- Authorized IP ranges:** Not enabled
- Application Gateway Ingress controller:** Enabled

Search

Refresh

Troubleshoot

Give feedback

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Microsoft Defender for Cloud

Cost analysis

Kubernetes resources

Settings

Node pools

Cluster configuration

Security configuration

Application scaling

Networking

Overview

Public access

Virtual network integration

Virtual network integration allows you to deploy dedicated instances of a service into a virtual network. Services can then be privately accessed within the virtual network and from on-premises networks. [Learn more](#)

Virtual network ⓘ

aks-vnet-30307550

Subnet ⓘ

aks-subnet

Application Gateway ingress controller

Ingress controller ⓘ

Enabled

Application gateway

applicationgateway

Manage

Application Gateway ingress controller

The Application Gateway Ingress Controller is a Kubernetes application, which makes it possible for Azure Kubernetes Service (AKS) customers to leverage Azure's native Application Gateway L7 load-balancer to expose cloud software to the Internet. [Learn more](#)

Ingress controller ⓘ



Application gateway

applicationgateway

Save

Cancel

Give feedback

+++++

AGENDA – RAN NGINX CONTROLLER ALSO

1) SEARCH – ingress- nginx

<https://github.com/kubernetes/ingress-nginx>

scroll down

Get started

See the [Getting Started](#) document.

Do not use in multi-tenant Kubernetes production installations. This project assumes that users that can create Ingress objects are administrators of the cluster. See the [FAQ](#) for more.

Troubleshooting

Deployment

Installation Guide

Bare-metal considerations

Role Based Access Control (RBAC)

Upgrade

Hardening guide

- ... [Rancher Desktop](#)
- ... [minikube](#)
- ... [MicroK8s](#)
- ... [AWS](#)
- ... [GCE - GKE](#)
- ... [Azure](#)
- ... [Digital Ocean](#)
- ... [Scaleway](#)
- ... [Exoscale](#)
- ... [Oracle Cloud Infrastructure](#)

Deployment

Installation Guide

Bare-metal considerations

Role Based Access Control (RBAC)

Azure

```
kubectl apply -f https://raw.githubusercontent.com/kubernetes/ingress-nginx/controller-v1.12.0/deploy/static/provider/cloud/deploy.yaml
```

More information with regard to Azure annotations for ingress controller can be found in the

Table

Conte

Quick

Fire

Pre-

`kubectl apply -f https://raw.githubusercontent.com/kubernetes/ingress-nginx/controller-v1.12.0/deploy/static/provider/cloud/deploy.yaml` = this command runs our nginx controller

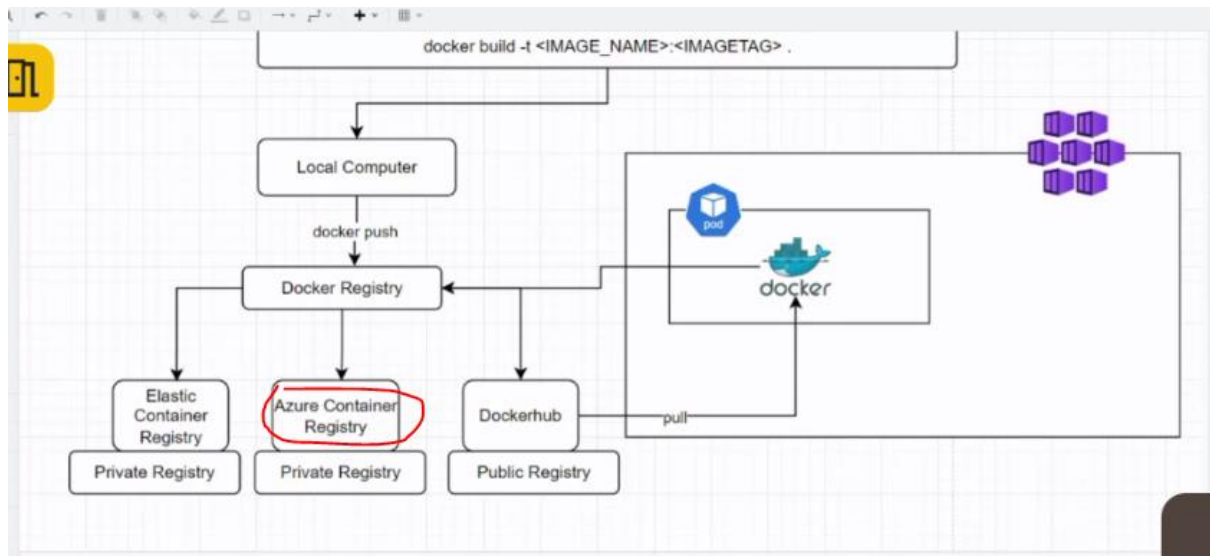
2) `kubectl get ingressclasses`

```
PS C:\4) KUBERNETES\9) 17a November Kubernetes\INGRESS> kubectl get ingressclasses
NAME                                CONTROLLER                PARAMETERS  AGE
azure-application-gateway          azure/application-gateway  <none>      177m
nginx                              k8s.io/ingress-nginx      <none>      81s
PS C:\4) KUBERNETES\9) 17a November Kubernetes\INGRESS>
```

+++++

AGENDA – docker

1) If our docker image is kept in local computer then we have to push that into different docker registries



INTERVIEW – We never have to say that we push our image to docker hub as it is public registry. Instead we have to say that we push our image to “Azure container registry” which is a private registry.