

DR Implementation In AKS Cluster

Step1: Create Two AKS Cluster in Different Region

- Primary-DR-aks with 1 zone and region is Canada Central
- Secondary-DR-aks with 1 zone and region is Japan East

Step2: Create on ACR in region Canada Central and add Geo-Geographical location for Japan East.

portal.azure.com/#browse/Microsoft.ContainerRegistry%2Fregistries

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

Home > Container registries

Default Directory (drgauravraj1993@gmail.onmicrosoft.com)

+ 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 0 to 0 of 0 records.

Name Type Resource group

devabottacr | Geo-replications

Container registry

Search + Add Refresh

Settings

- Access keys
- Encryption
- Identity
- Networking
- Microsoft Defender for Cloud
- Properties
- Locks

Services

- Repositories
- Webhooks
- Geo-replications**
- Tasks
- Connected registries (Preview)
- Cache

World map showing locations for Geo-replications.

Filter by name

Name	Location	Provisioning state	Status
japaneast	Japan East	Succeeded	Ready
canadacentral	Canada Central	Succeeded	Ready

Step3: Perform VNET Peering with both in different cluster.

The first screenshot shows the 'e2e-rg-vnet' address space configuration. The address space is 10.208.0.0/12 with an address count of 1,048,576. A peering named 'primary-peering-2' is shown connecting to 'aks-vnet-27019674' with address space 10.224.0.0/12.

The second screenshot shows the 'aks-vnet-27019674' address space configuration. The address space is 10.224.0.0/12 with an address count of 1,048,576. A warning message indicates that the address space overlaps with the 'e2e-rg-vnet' address space.

The third screenshot shows the 'e2e-rg-vnet' peering configuration. The peering 'primary-peering-2' is shown with a 'Fully Synchronized' status and a 'Connected' state.

The fourth screenshot shows the 'aks-vnet-27019674' peering configuration. The peering 'primary-peering-1' is shown with a 'Fully Synchronized' status and a 'Connected' state.

Step4: Enable ingress controller in both AKS

Step5: Run Below command In both AKS to make public IP Static

Note: Run Below Step in Both AKS Cluster.

- (Optional) First of all, check public ip of both aks (1 by 1) with below commands
`az network public-ip list --resource-group MC_e2e-rg_primary-aks-cluster_japaneast --query "[].{Name:name, IPAddress:ipAddress}" -o table`
- Then, make static public ip and dns name with below command:
`az network public-ip update --resource-group MC_e2e-rg_primary-aks-cluster_japaneast --name ingress-appgateway-appgwip --dns-name ingress-appgateway --allocation-method Static --sku Standard`

- c. Then check dns name by below command:

```
az network public-ip show --resource-group MC_e2e-rg_primary-aks-cluster_japaneast --name ingress-appgateway-appgwpip --query "dnsSettings.fqdn" -o tsv
```

- d. (optional)Then again verify public ips list

```
az network public-ip list --resource-group MC_e2e-rg_primary-aks-cluster_japaneast --query "[].{Name:name, IPAddress:ipAddress}" -o table
```

```
nancy@DESKTOP-7M9SJH4:~/DEVOPS/KUBERNETES/DR$ code .
nancy@DESKTOP-7M9SJH4:~/DEVOPS/KUBERNETES/DR$ az network public-ip list --resource-group MC_e2e-rg_primary-aks-cluster_japaneast --query "[].{Name:name, IPAddress:ipAddress}" -o table
Name                               IPAddress
-----
5dc4c2ff-7dca-4cc6-8594-e4f4d7afd32e 74.176.25.158
ingress-appgateway-appgwpip         74.176.152.13
nancy@DESKTOP-7M9SJH4:~/DEVOPS/KUBERNETES/DR$ az network public-ip show --resource-group MC_e2e-rg_primary-aks-cluster_japaneast --name ingress-appgateway-appgwpip --query "dnsSettings.fqdn" -o tsv
ingress-appgateway.japaneast.cloudapp.azure.com
nancy@DESKTOP-7M9SJH4:~/DEVOPS/KUBERNETES/DR$ az network public-ip show --resource-group MC_e2e-rg_primary-aks-cluster_japaneast --name ingress-appgateway-appgwpip --query "dnsSettings.fqdn" -o tsv
ingress-appgateway.japaneast.cloudapp.azure.com
nancy@DESKTOP-7M9SJH4:~/DEVOPS/KUBERNETES/DR$ az network public-ip list --resource-group MC_e2e-rg_primary-aks-cluster_japaneast --query "[].{Name:name, IPAddress:ipAddress}" -o table
Name                               IPAddress
-----
5dc4c2ff-7dca-4cc6-8594-e4f4d7afd32e 74.176.25.158
ingress-appgateway-appgwpip         74.176.152.13
nancy@DESKTOP-7M9SJH4:~/DEVOPS/KUBERNETES/DR$
```

Step6: Perform Below Step for Creating traffic Manager:

- Create Traffic Manager with Name, RG, location, Routing(Priority) etc.
- Add endpoint: name, add 1st dns name(generate above), give priority 1, add
- Add endpoint: name, add 2nd dns name(generate above), give priority 2, add
- In configuration > add path (if applicable)

[Home](#) > [Load balancing | Traffic Manager](#) >

Create Traffic Manager profile ...

Name *

Routing method

Priority

Subscription *

Free Trial

Resource group *

[Create new](#)

Resource group location ⓘ

Japan East

Home > Load balancing | Traffic Manager > dr-traffic-manager

Load balancing | Traffic Manager

Search

Overview

Filter for any field...

Name ↑

dr-traffic-manager

Load Balancing Services

- Application Gateway
- Front Door and CDN profiles
- Load Balancer
- Traffic Manager

dr-traffic-manager | Configuration

Search

Save Discard

Priority

DNS time to live (TTL) * 60 seconds

Endpoint monitor settings

Protocol HTTP

Port * 80

Path * /

Custom Header settings

Configure in this format, host:contoso.com,customheader:contoso

Microsoft Azure Upgrade

Search resources, services, and docs (G+I)

Home > Load balancing | Traffic Manager > dr-traffic-manager

Load balancing | Traffic Manager

Search

Overview

Filter for any field...

Name ↑

dr-traffic-manager

Load Balancing Services

- Application Gateway
- Front Door and CDN profiles
- Load Balancer
- Traffic Manager

dr-traffic-manager | Endpoints

Search endpoints

Name ↑	Status ↑	Monitor status ↑	Type ↑	Priority ↑
dr-endpoint	Disabled	Degraded	Azure endpoint	1
dr-endpoint-2	Enabled	Degraded	Azure endpoint	2

Microsoft Azure Upgrade

Search resources, ser

Home > Load balancing | Traffic Manager > dr-traffic-manager >

dr-endpoint

dr-traffic-manager

Save Discard Delete

Status

Disabled

Monitor status

Degraded

Type

Azure endpoint

Target resource type ⓘ

Public IP address

Target resource *

ingress-appgateway-appgwpip (Canada Central)

Priority *

1

Custom Header settings ⓘ

Configure in this format, host:contoso.com,customheader:contoso

⚠ Do NOT input sensitive customer data in this field (i.e. APIKeys, Secrets, and Auth tokens etc.).

Step7: traffic-manager dns name in godaddy with cname like “dr-traffic-manager.trafficmanager.net”

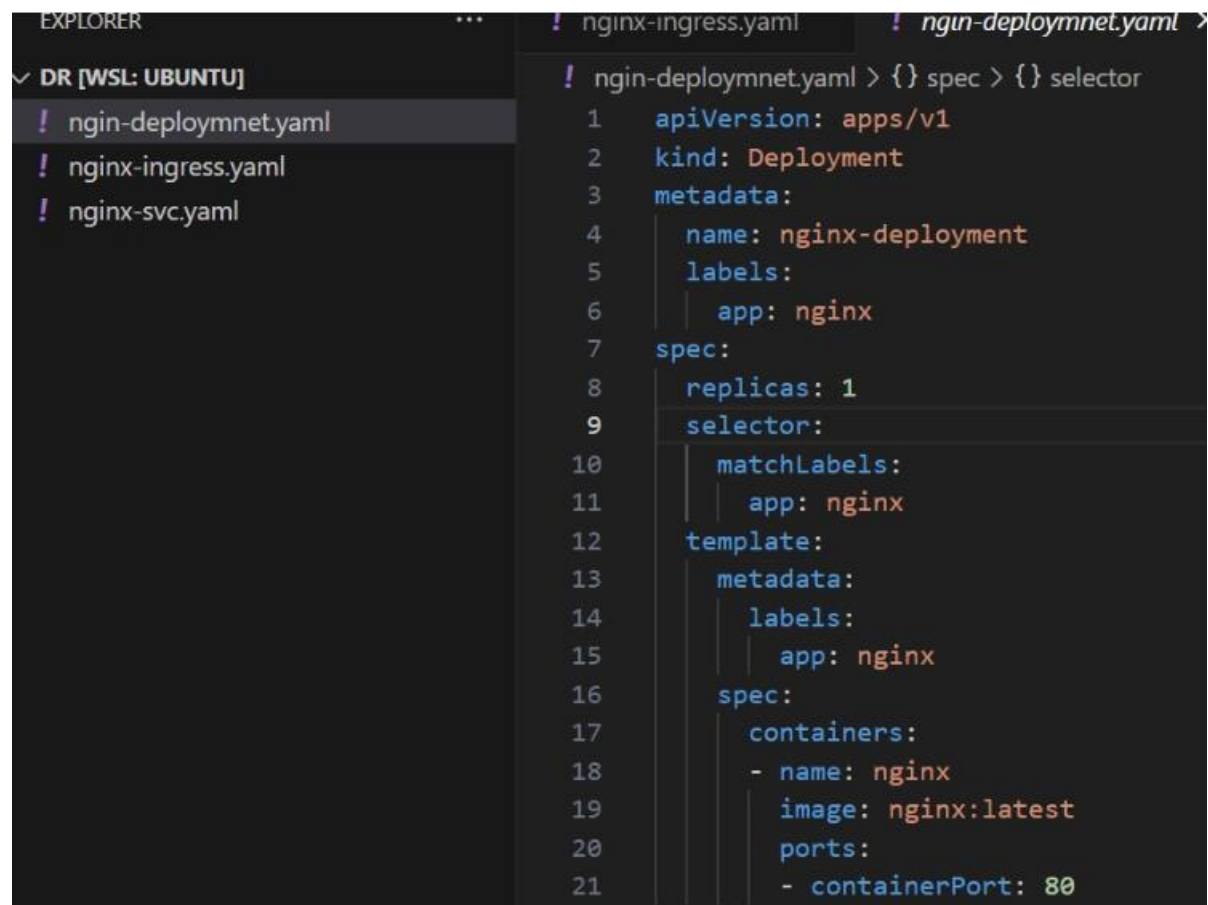
Step8: check by nslookup nslookup dr-traffic-manager.trafficmanager.net, if it is 1st aks ip or not

```
nancy@DESKTOP-7M9SJH4:~/DEVOPS/KUBERNETES/DR$ nslookup dr-traffic-manager.trafficmanager.net
Server:      10.255.255.254
Address:     10.255.255.254#53

Non-authoritative answer:
dr-traffic-manager.trafficmanager.net  canonical name = ingress-appgateway.japaneast.cloudapp.azure.com.
Name:   ingress-appgateway.japaneast.cloudapp.azure.com
Address: 74.176.152.13

nancy@DESKTOP-7M9SJH4:~/DEVOPS/KUBERNETES/DR$
```

Step9: Now Deploy Application in both AKS cluster, run deployment.yaml, services.yaml and ingress.yaml



The screenshot shows the VS Code interface. On the left, the Explorer pane shows a file tree for 'DR [WSL: UBUNTU]' with files 'ngin-deploymnet.yaml', 'nginx-ingress.yaml', and 'nginx-svc.yaml'. The main editor pane shows the content of 'ngin-deploymnet.yaml' with line numbers 1 through 21. The file content is a Kubernetes Deployment manifest for nginx.

```
1  apiVersion: apps/v1
2  kind: Deployment
3  metadata:
4    name: nginx-deployment
5    labels:
6      app: nginx
7  spec:
8    replicas: 1
9    selector:
10     matchLabels:
11       app: nginx
12   template:
13     metadata:
14       labels:
15         app: nginx
16     spec:
17       containers:
18       - name: nginx
19         image: nginx:latest
20       ports:
21       - containerPort: 80
```

```

# Please edit the object below. Lines beginning with a '#' will be ignored,
# and an empty file will abort the edit. If an error occurs while saving this file will be
# reopened with the relevant failures.
#
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  annotations:
    kubectl.kubernetes.io/last-applied-configuration: |
      {"apiVersion":"networking.k8s.io/v1","kind":"Ingress","metadata":{"annotations":{},"name":"nginxdr.satisfy","namespace":"default","resourceVersion":"161485"},"spec":{"ingressClassName":"azure-application-gateway","rules":[{"host":"nginxdr.satisfy","service":{"name":"nginx-service-dr","port":{"number":80}},"path":"/","pathType":"Prefix"}]},"creationTimestamp":"2024-12-31T09:18:42Z","generation":4,"name":"nginx-ingress-dr","namespace":"default","resourceVersion":"161485","uid":"56e79e61-956a-4ea1-bc8c-618bdd059165"}
status:
  ingressClassName: azure-application-gateway
  rules:
  - host: dr-traffic-manager.trafficmanager.net
    http:
      paths:
      - backend:
          service:
            name: nginx-service-dr
            port:
              number: 80
          path: /
          pathType: Prefix

```

"/tmp/kubectl-edit-1586752864.yaml" 33L, 1227B

```

! nginx-svc.yaml
1  apiVersion: v1
2  kind: Service
3  metadata:
4    name: nginx-service
5    labels:
6      app: nginx
7  spec:
8    selector:
9      app: nginx # This should match the label in the pod spec
10   ports:
11     - protocol: TCP
12       port: 80 # The port exposed by the service
13       targetPort: 80 # The port on the pod the service should route traffic to
14   type: ClusterIP # For internal access within the cluster

```

Step10: Now In 2 AKS Cluster, go to 2nd nginx pod > /usr/share/nginx/html, edit index.html

Step11: Now, test application, in browser we can paster host name “dr-traffic-manager.trafficmanager.net”, Primary DR nginx will open.




Step12: Now test failover by Disaling 1 endpoint in Traffic manager

Microsoft Azure Upgrade Search resources, services, and documentation

Home > Load balancing | Traffic Manager > dr-traffic-manager >

dr-endpoint

dr-traffic-manager

 Save  Discard  Delete

Status

Disabled

Monitor status

Degraded

Type

Azure endpoint

Target resource type ⓘ

Public IP address

Target resource *


ingress-appgateway-appgwpip (Canada Central)

Priority *

1

Custom Header settings ⓘ

Configure in this format, host:contoso.com,customheader:contoso

 Do NOT input sensitive customer data in this field (i.e. APIKeys, Secrets, and Auth tokens etc.).

Browse dr-traffic-manager.trafficmanager.net, in Browser it will give u conut of 2 cluster deployment