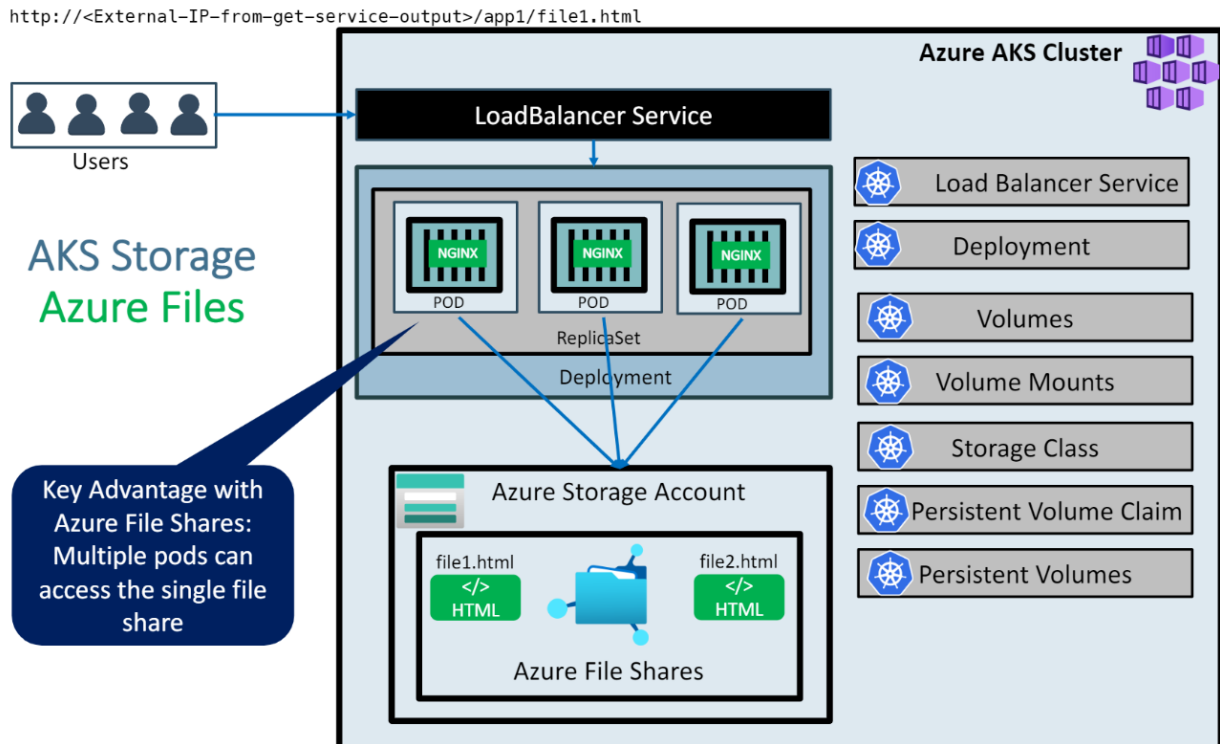


Azure Files for AKS Storage



Step-01: Introduction

- Understand Azure Files
- We are going to write a Deployment Manifest for NGINX Application which will have its static content served from Azure File Shares in app1 folder
- We are going to mount the file share to a specific path `mountPath:`
`"/usr/share/nginx/html/app1"` in the Nginx container

kube-manifests-v1: Custom Storage Class

- We will define our own custom storage class with desired permissions
 - Standard_LRS - standard locally redundant storage (LRS)
 - Standard_GRS - standard geo-redundant storage (GRS)
 - Standard_ZRS - standard zone redundant storage (ZRS)

- Standard_RAGRS - standard read-access geo-redundant storage (RA-GRS)
- Premium_LRS - premium locally redundant storage (LRS)

kube-manifests-v2: AKS defined default storage class

- With default AKS created storage classes only below two options are available for us.
 - Standard_LRS - standard locally redundant storage (LRS)
 - Premium_LRS - premium locally redundant storage (LRS)
- Important Note: Azure Files support premium storage in AKS clusters that run Kubernetes 1.13 or higher, minimum premium file share is 100GB

Step-02: Create or Review kube-manifests-v1 and Nginx Files

- Kube Manifests
 - 01-Storage-Class.yml

```
abraham@Azure:~$ cat kubectldir/stroage-class.yml
```

```
kind: StorageClass
```

```
apiVersion: storage.k8s.io/v1
```

```
metadata:
```

```
  name: my-azurefile-sc
```

```
provisioner: kubernetes.io/azure-file
```

```
mountOptions:
```

```
- dir_mode=0777
```

```
- file_mode=0777
```

```
- uid=0
```

```
- gid=0
```

- mfsymlinks
- cache=strict

parameters:

skuName: Standard_LRS

- 02-Persistent-Volume-Claim.yml

abraham@Azure:~/kubectldir\$ cat persistent-volume-claim.yml

apiVersion: v1

kind: PersistentVolumeClaim

metadata:

name: my-azurefile-pvc

spec:

accessModes:

- ReadWriteMany

storageClassName: my-azurefile-sc

resources:

requests:

storage: 5Gi

- 03-Nginx-Deployment.yml

abraham@Azure:~/kubectldir\$ cat nginx-deployment.yml

apiVersion: apps/v1

kind: Deployment

metadata:

name: azure-files-nginx-deployment

labels:

app: azure-files-nginx-app

spec:

replicas: 4

selector:

matchLabels:

app: azure-files-nginx-app

template:

metadata:

labels:

app: azure-files-nginx-app

spec:

containers:

- name: azure-files-nginx-app

image: stacksimplify/kube-nginxapp1:1.0.0

imagePullPolicy: Always

ports:

- containerPort: 80

volumeMounts:

- name: my-azurefile-volume

mountPath: "/usr/share/nginx/html/app1"

volumes:

- name: my-azurefile-volume

persistentVolumeClaim:

claimName: my-azurefile-pvc

- 04-Nginx-Service.yml

abraham@Azure:~/kubectldir\$ cat nginx-service.yaml

apiVersion: v1

kind: Service

metadata:

name: azure-files-nginx-service

labels:

app: azure-files-nginx-app

spec:

type: LoadBalancer

selector:

app: azure-files-nginx-app

ports:

- port: 80

targetPort: 80

- nginx-files
 - file1.html
 - file2.html
- k8s Deployment manifest - core item for review

```
    volumeMounts:
      - name: my-azurefile-volume
        mountPath: "/usr/share/nginx/html/app1"
  volumes:
    - name: my-azurefile-volume
      persistentVolumeClaim:
        claimName: my-azurefile-pvc
```

Step-03: Deploy Kube Manifests V1

```
# Deploy
kubectl apply -f kube-manifests-v1/
```

```
abraham@Azure:~$ kubectl create -f kubectldir/
deployment.apps/azure-files-nginx-deployment created
service/azure-files-nginx-service created
persistentvolumeclaim/my-azurefile-pvc created
storageclass.storage.k8s.io/my-azurefile-sc created
abraham@Azure:~$
```

```
# Verify SC, PVC, PV
kubectl get sc, pvc, pv
```

```
# Verify Pod
kubectl get pods
kubectl describe pod <pod-name>
```

```
# Get Load Balancer Public IP
kubectl get svc
```

```

abraham@Azure:~/kubectldir$ kubectl get po,sc,pvc,rs,deploy,svc
NAME                                     READY   STATUS    RESTARTS   AGE
pod/azure-files-nginx-deployment-7b49989fc8-4p4sz  1/1     Running   0           4m53s
pod/azure-files-nginx-deployment-7b49989fc8-ltv94  1/1     Running   0           4m53s
pod/azure-files-nginx-deployment-7b49989fc8-qtjc2  1/1     Running   0           4m53s
pod/azure-files-nginx-deployment-7b49989fc8-zg5k4  1/1     Running   0           4m53s

NAME                                     PROVISIONER             RECLAIMPOLICY   VOLUMEBINDINGMODE   ALLOWVOLUMEEXPANSION   AGE
storageclass.storage.k8s.io/azurefile    kubernetes.io/azure-file Delete          Immediate            true                   30m
storageclass.storage.k8s.io/azurefile-premium kubernetes.io/azure-file Delete          Immediate            true                   30m
storageclass.storage.k8s.io/default (default) kubernetes.io/azure-disk Delete          WaitForFirstConsumer true                   30m
storageclass.storage.k8s.io/managed-premium kubernetes.io/azure-disk Delete          WaitForFirstConsumer true                   30m
storageclass.storage.k8s.io/my-azurefile-sc kubernetes.io/azure-file Delete          Immediate          false                  4m53s

NAME                                     STATUS    VOLUME                                     CAPACITY   ACCESS MODES   STORAGECLASS   AGE
persistentvolumeclaim/my-azurefile-pvc  Bound     pvc-078cd918-ca6f-4ab1-b8e5-5d21959bb024  5Gi        RWX             my-azurefile-sc 4m53s

NAME                                     DESIRED   CURRENT   READY   AGE
replicaset.apps/azure-files-nginx-deployment-7b49989fc8  4          4         4       4m53s

NAME                                     READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/azure-files-nginx-deployment  4/4      4             4           4m54s

NAME                                     TYPE          CLUSTER-IP   EXTERNAL-IP   PORT(S)          AGE
service/azure-files-nginx-service  LoadBalancer  10.0.156.73   52.140.37.45  80:31288/TCP     4m54s
service/kubernetes                  ClusterIP      10.0.0.1      <none>         443/TCP          30m
abraham@Azure:~/kubectldir$

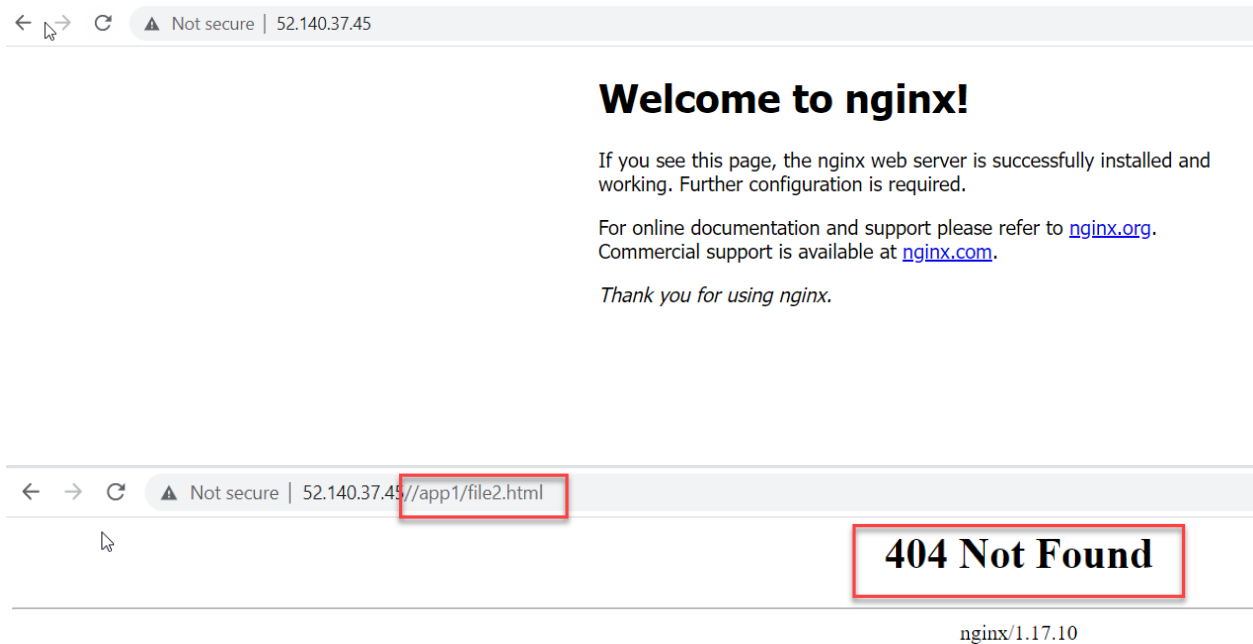
```

Access Application

The default page will appear but the app1<file and file2 > will not come up

<http://<External-IP-from-get-service-output>>

<http://<External-IP-from-get-service-output>/app1/index.html>



Step-04: Upload Nginx Files to Azure File Share

- Go to Storage Accounts

- Select and Open storage account under resource group mc_abram-rg_abramaks_southindia

Microsoft Azure

Search resources, services, and docs (G+)

Home >

Storage accounts

Default Directory

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

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

Showing 1 to 2 of 2 records.

Name	Type	Kind	Resource group	Location	Subscription
csg10032001ddb76984	Storage account	StorageV2	cloud-shell-storage-centralin...	Central India	Abraham Azure Account
f7b3a4979aca54869a3ffa7	Storage account	StorageV2	mc_abram-rg_abramaks_southindia	South India	Abraham Azure Account

- In Overview, go to File Shares

Microsoft Azure

Search resources, services, and docs (G+)

Home > Storage accounts >

f7b3a4979aca54869a3ffa7

Storage account

Search (Ctrl+)

Overview

Activity log

Tags

Diagnose and solve problems

Access Control (IAM)

Data migration

Events

Storage browser (preview)

Data storage

Containers

File shares

File service

Large file share

Active Directory

Soft delete

Share capacity

Queue service

CMK support

Disabled

Disabled

Disabled

Disabled

Disabled

Disabled

Enabled

Disabled

Not configured

Enabled (7 days)

5 TiB

Disabled

- Open File share with name which starts as kubernetes-dynamic-pv-xxxxxx

Home > Storage accounts > f7b3a4979aca54869a3ffa7

f7b3a4979aca54869a3ffa7 | File shares

Search (Ctrl+/) << + File share Refresh

File share settings

Active Directory: Not configured Soft delete: 7 days Maximum capacity: 5 TiB Security: Custom

Search file shares by prefix (case-sensitive) Show deleted shares

Name	Modified	Tier	Quota
kubernetes-dynamic-pvc-078cd918-ca6f-4ab1-b8e5-5d21959...	3/14/2022, 8:16:15 PM	Transaction optimized	5 GiB

- Click on Upload and upload
 - file1.html
 - File2.html

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

Home > Storage accounts > f7b3a4979aca54869a3ffa7 >

kubernetes-dynamic-pvc-078cd918-ca6f-4ab1-b8e5-5d21959bb024

File share

Search (Ctrl+/) << Connect Upload + Add directory Refresh Delete share Change tier Edit quota

Search files by prefix

Name	Type
No files found.	

Upload files

kubernetes-dynamic-pvc-078cd918-ca6f-4ab1-b8e5-5d21959bb024

Files

file1.html file2.html

☒ Overwrite if files already exist

Upload

Current uploads

Step-05: Access Application & Test

URLs

http://<External-IP-from-get-service-output>/app1/file1.html

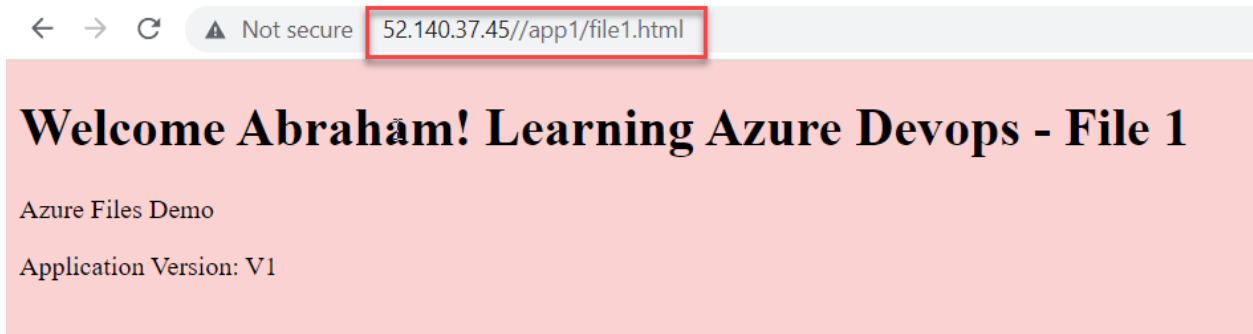
http://<External-IP-from-get-service-output>/app1/file2.html

← → ↻ ⚠ Not secure 52.140.37.45/app1/file2.html

Welcome Abraham! Learning Azure Devops - File 2

Azure Files Demo

Application Version: V1



Step-06: Clean-Up

```
# Delete
kubectl delete -f kubectldir/
```

```
abraham@Azure:~$ kubectl delete -f kubectldir/
deployment.apps "azure-files-nginx-deployment" deleted
service "azure-files-nginx-service" deleted
persistentvolumeclaim "my-azurefile-pvc" deleted
storageclass.storage.k8s.io "my-azurefile-sc" deleted
abraham@Azure:~$
```

Step-07: Create or Review kube-manifests-v2 and Nginx Files

- Kube Manifests
 - 01-Persistent-Volume-Claim.yml
 - 02-Nginx-Deployment.yml
 - 03-Nginx-Service.yml
- nginx-files
 - file1.html
 - file2.html

Default Provisions

Step-08: Deploy Kube Manifests V2 (Default Provisions)

```
# Deploy
kubectl apply -f kube-manifests-v2/

# Verify SC, PVC, PV
kubectl get sc, pvc, pv

# Verify Pod
kubectl get pods
kubectl describe pod <pod-name>

# Get Load Balancer Public IP
kubectl get svc

# Access Application
http://<External-IP-from-get-service-output>
```

Step-09: Upload Nginx Files to Azure File Share

- Go to Storage Accounts
- Select and Open storage account under resource group
mc_aks-rg1_aksdemo1_eastus
- In Overview, go to File Shares
- Open File share with name which starts as kubernetes-dynamic-pv-xxxxxx
- Click on Upload and upload
 - file1.html
 - file2.html

Step-10: Access Application & Test

```
# URLs
http://<External-IP-from-get-service-output>/app1/file1.html
http://<External-IP-from-get-service-output>/app1/file2.html
```

Step-11: Clean-Up

```
# Delete
```

```
kubectl delete -f kube-manifests-v2/
```