**Azure File**

* These are simple, secure and fully managed cloud file shares.
* We can secure data at rest and in-transit using SMB 3.0 and HTTPS.
* We can create high-performance file shares using the premium files storage tier.
* We can create and manage Azure file shares using Azure Portal and Azure Storage Explorer.
* For application workloads we can use for use cases like static content storage, shared configuration access to multiple JVMs etc.
* Containerization:

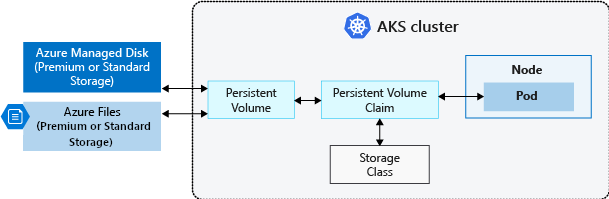
Azure file shares can be used as persistent volumes for stateful containers. Containers deliver "build once, run anywhere" capabilities that enable developers to accelerate innovation. For the containers that access raw data at every start, a shared file system is required to allow these containers to access the file system no matter which instance they run on.

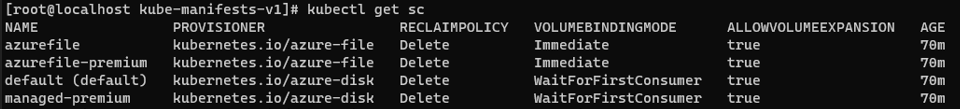
* Shared access: -

Azure file shares support the industry standard SMB and NFS protocols, meaning you can seamlessly replace your on-premises file shares with Azure file shares without worrying about application compatibility. Being able to share a file system across multiple machines, applications/instances is a significant advantage with Azure Files for applications that need shareability.

* Fully managed: -

Azure file shares can be created without the need to manage hardware or an OS. This means you don't have to deal with patching the server OS with critical security upgrades or replacing faulty hard disks.





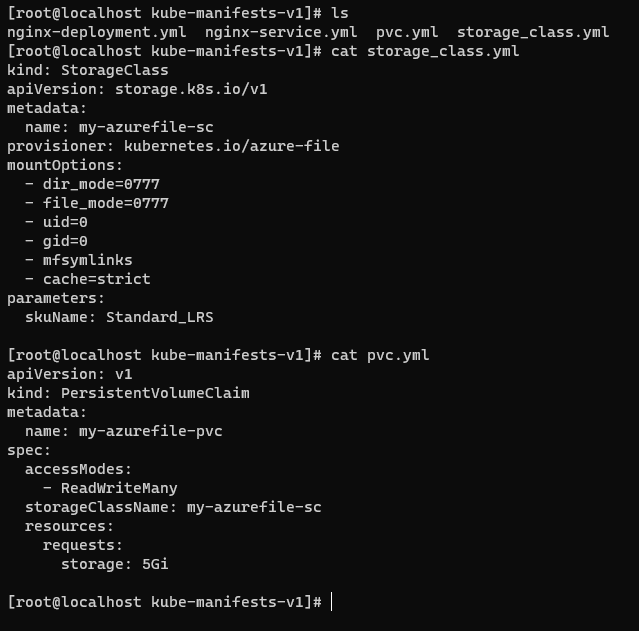
When you see the storage class in AKS you will see there are 2 different azure files storage. If you use this storage class then it only provides Standard\_LRS and Premium\_LRS.

**Azure File with Custom Storage Class**

We will define our own custom storage class with desired permissions

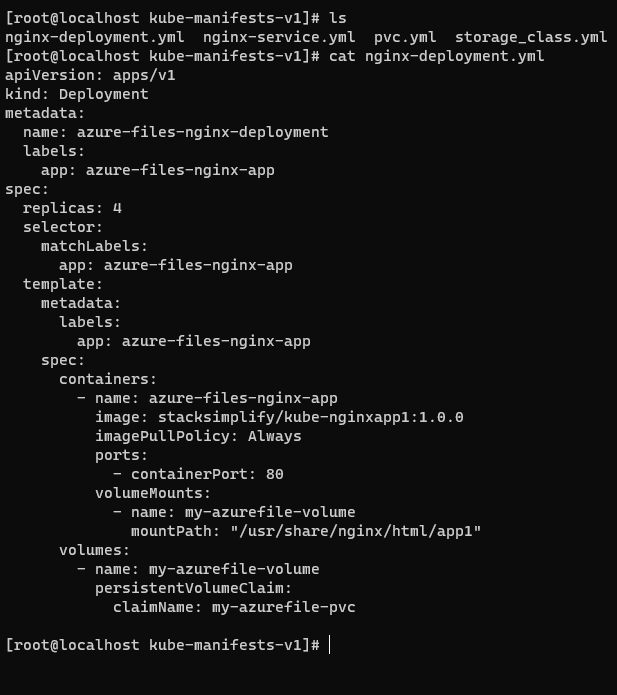
* Standard\_LRS - standard locally redundant storage (LRS)
* Standard\_GRS - standard geo-redundant storage (GRS)
* Standard\_ZRS - standard zone redundant storage (ZRS)
* Standard\_RAGRS - standard read-access geo-redundant storage (RA-GRS)
* Premium\_LRS - premium locally redundant storage (LRS)

1. Create custom storage class and pvc manifest.

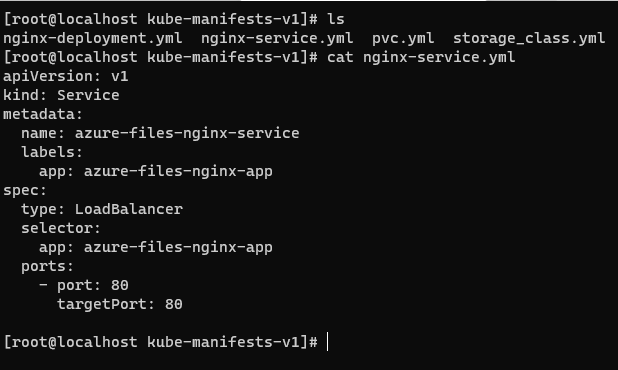


The custom storage class is mounted to pvc.

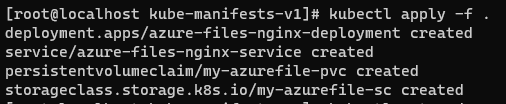
1. Now deploy nginx-deployment with attach pvc volume.

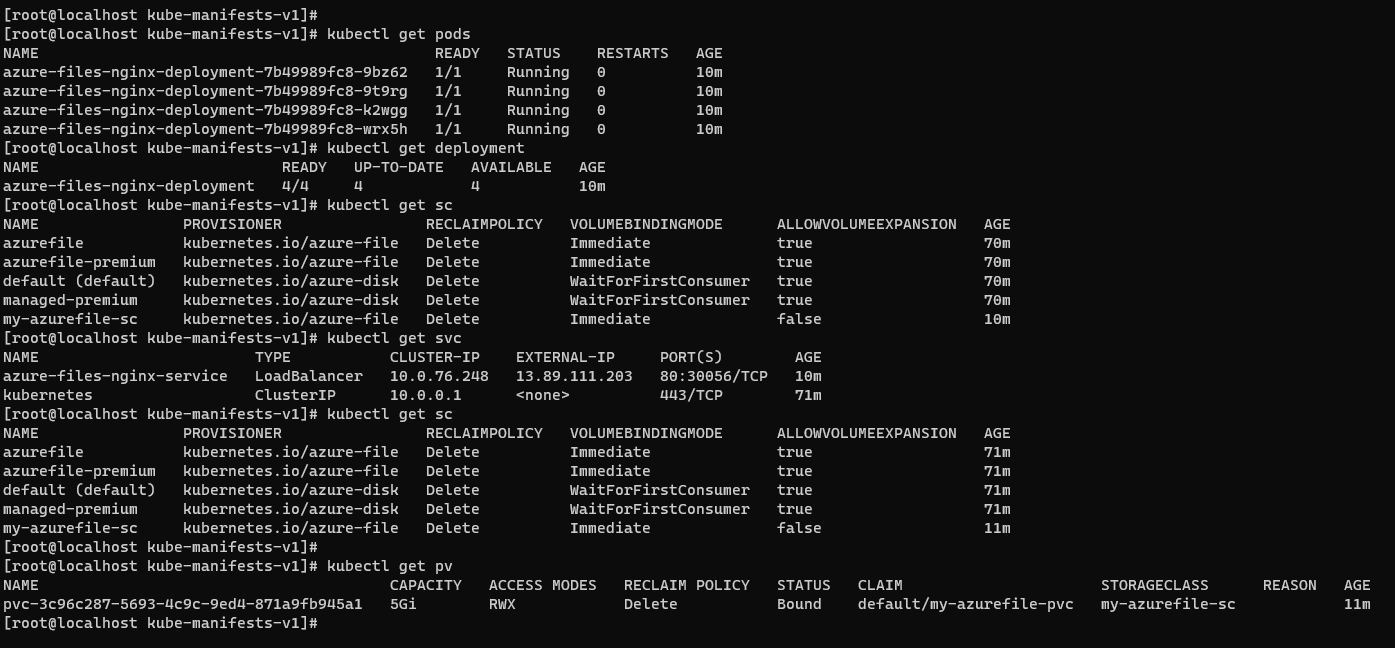


1. Create LoadBalancer service for nginx-deployment.

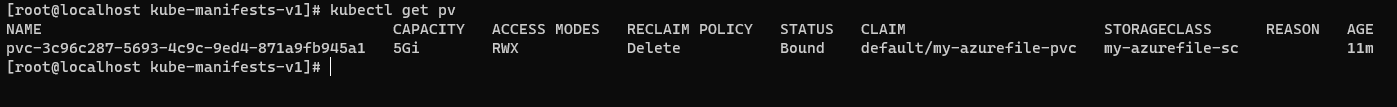


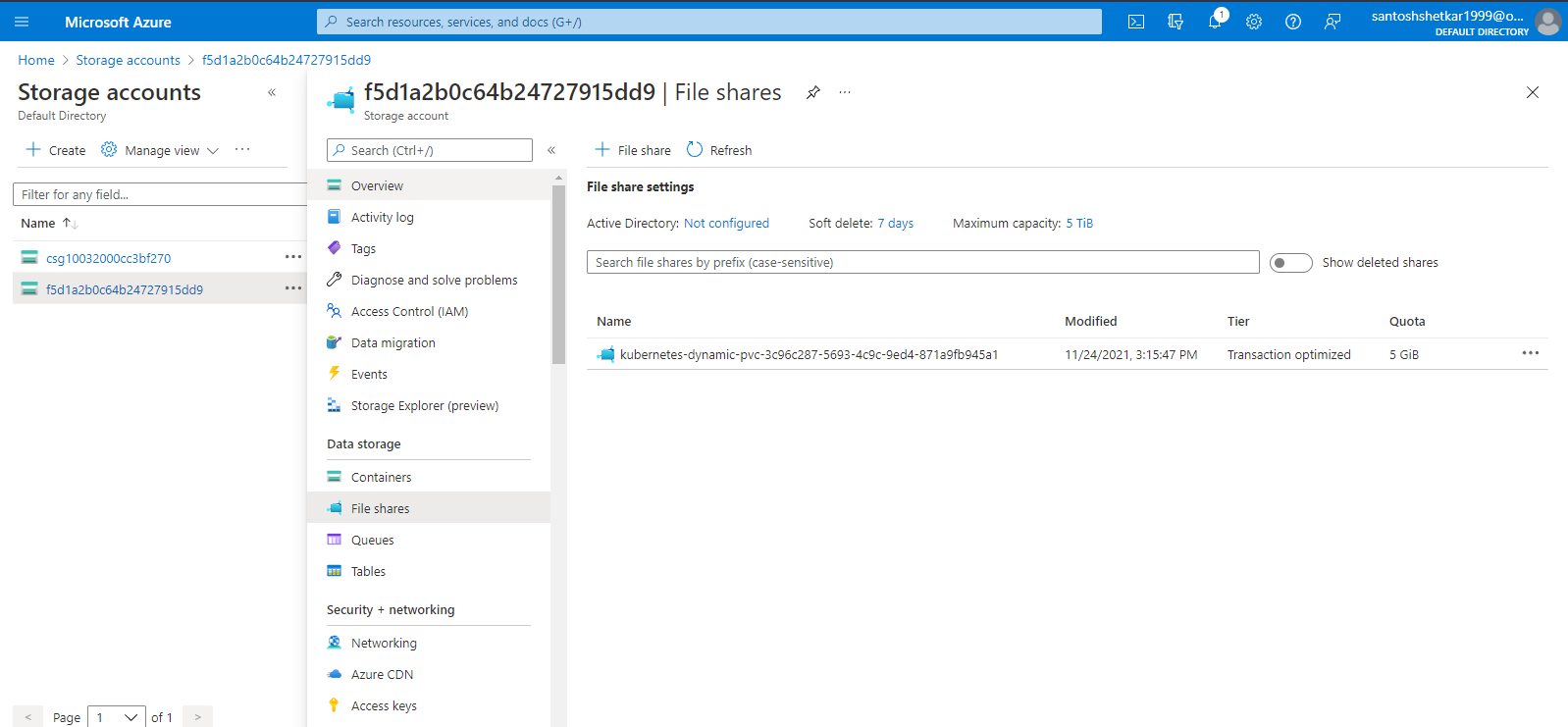
1. Now deploy all manifest and verify it.



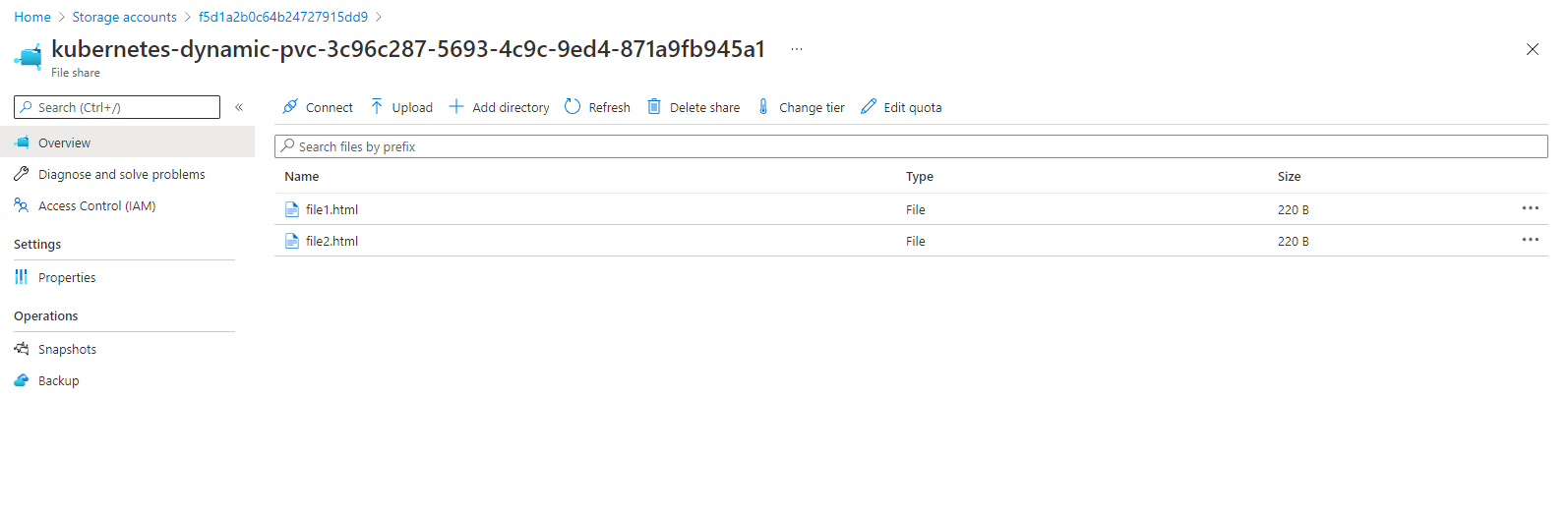


After deploy you will see the pv and azure file storage name is same.

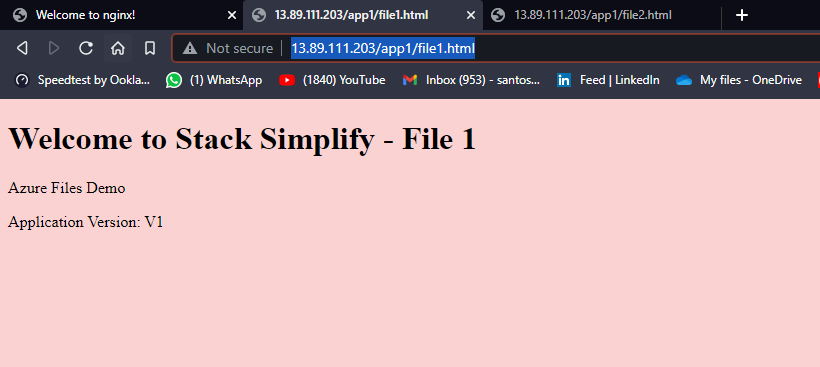


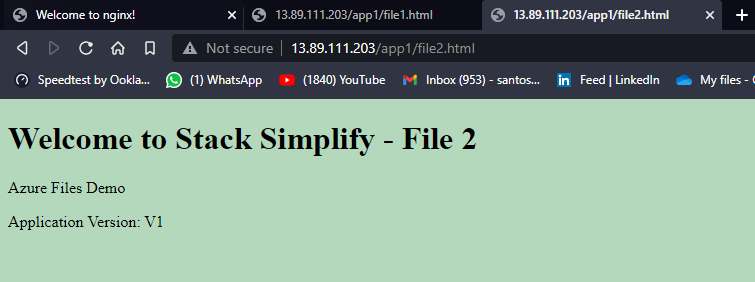


1. Now add some html file in created azure file storage.



1. Now access these files via browser using LoadBalancer public IP address.

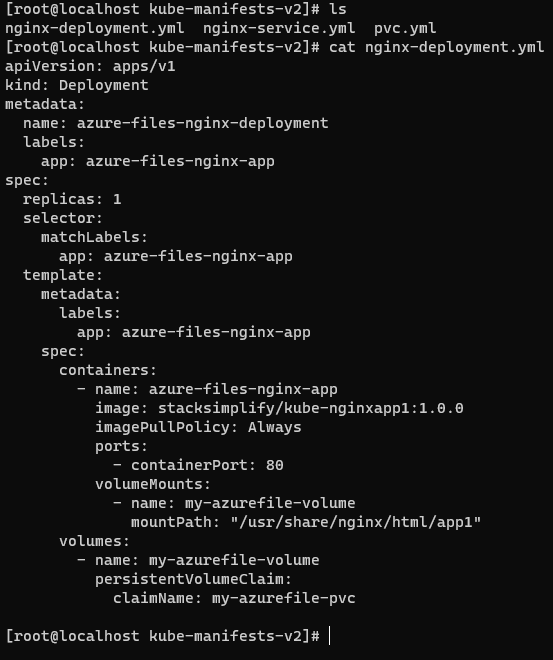


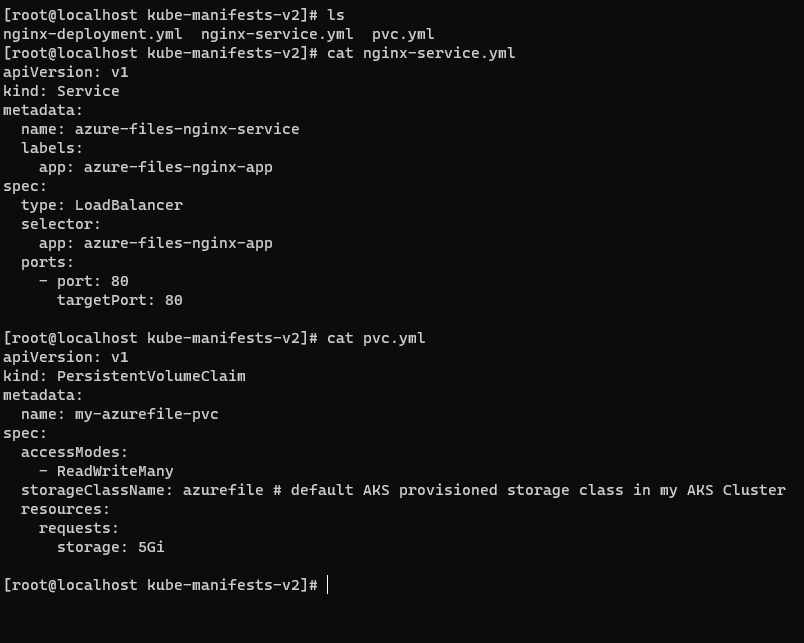


Successfully deployed Custom Azure file storage!!

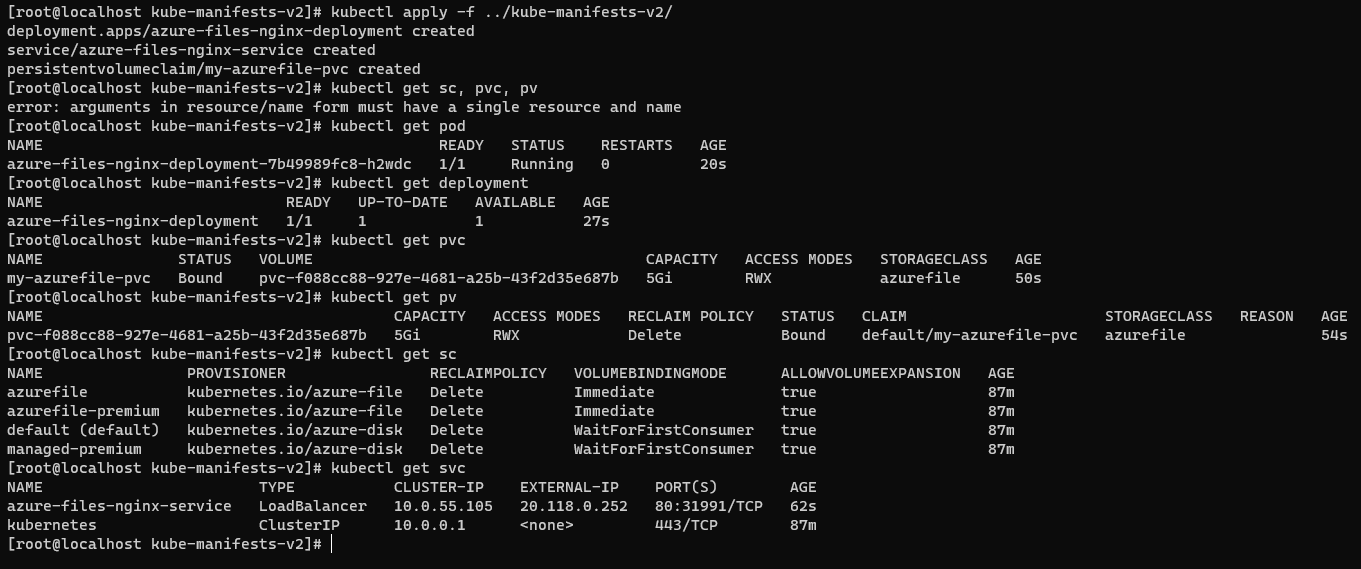
**Azure File with AKS defined default storage class**

In this we don’t want create custom storage so then only need to mention the default storage class name in pvc manifest.



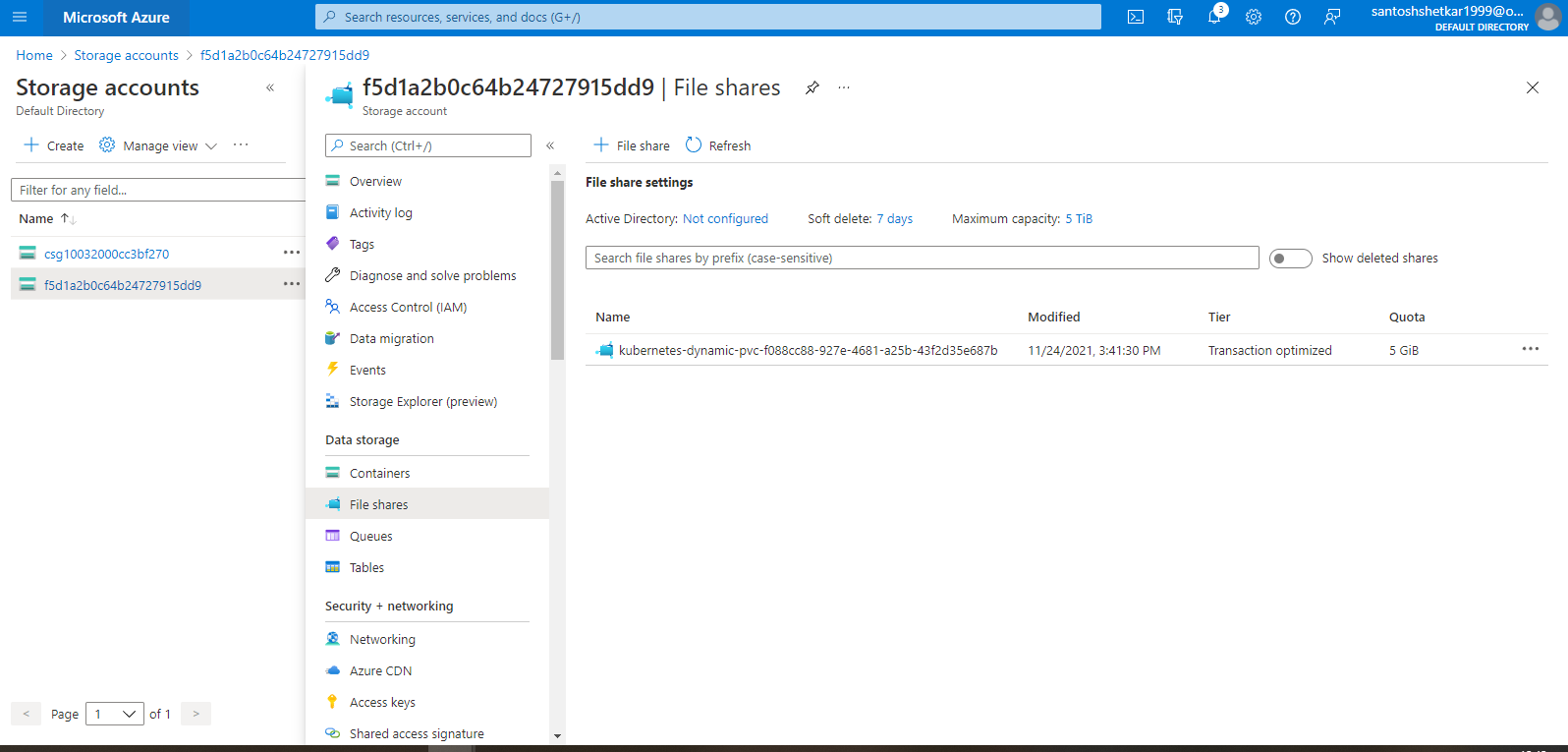


Deploy all manifest

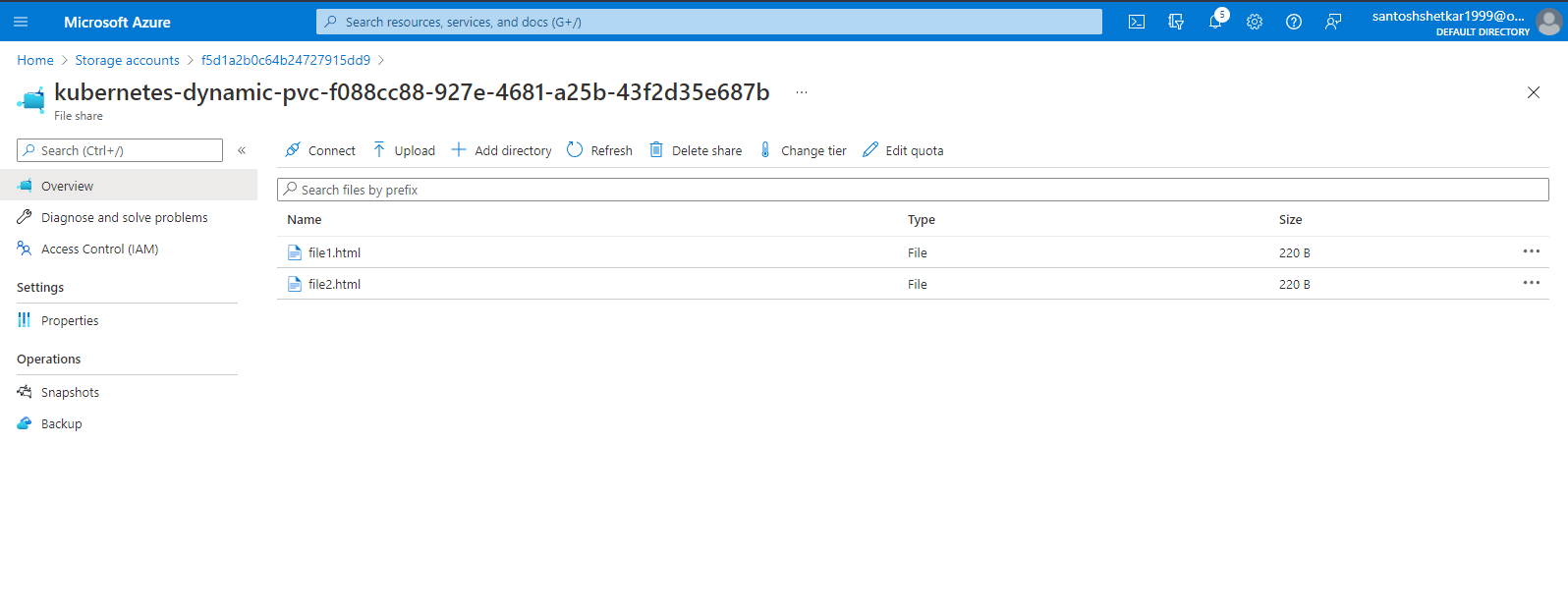


You will see the PV and azure file storage name is same.





Add some html file in azure file storage.



Now access these files via browser using LoadBalancer public IP address.

