Setup Storage Provisioner and Storage Class

****

**** Deploying Cluster Storage Provisioner ****

****

after setting up a cluster, we have to configure storage class for creating PVs for PVC of the clients and application dynamically. after setting up storage class, the PVC will access to storage class by its name or if the storage class has been set to default there is no need to set the name of storage class in PVC. therefor PVC will request to SC for generating a new PV and assign it to requested pod and finally pod can mount its desired volume to that storage. for setting up this environment first of all we need nfs-server and nfs-provisioner to make this approach happen. so follow the steps below to setting up your environment.

Steps {

Step1: adding disk storage and format it as new volume.

For this purpose we have to add a physical storage to the server and configure it as a lvm or normal disk storage and make partitions,
after that we have to format the disk and then we mount that storage and configure on fstab file to mount this path automatically after rebooting the server.

Step 2: Install nfs-server

For installing nfs server we can use nfs-server docker-compose based. After adding new volume to nfs server we create a folder under /opt
called “/opt/nfs-server” and for configuring nfs-server we use the docker-compose file like below:

```
version: !!str 3
services:
  nfs-server:
    image: itsthenetwork/nfs-server-alpine
    restart: unless-stopped
    privileged: true
    environment:
      - "SHARED_DIRECTORY=/nfsshare"
    ports:
      - 111:111
      - 111:111/udp
      - 2049:2049
      - 2049:2049/udp
    volumes:
      - /mnt/nfs:/nfsshare
networks:
  nfs:
```

in this docker-compose file we serve the directory /mnt/nfs to the container as shared directory for publish nfs server.
After that we up this compose file and our nfs-server is ready.

Step 3: Configure nfs provisioner on kubernetes cluster

For this approach we use a deployed helm chart. So first we have to clone that repository

```
git clone https://github.com/kubernetes-sigs/nfs-subdir-external-provisioner.git && cd nfs-subdir-external-provisioner/charts/nfs-subdir-external-provisioner/
```

step 4: change values of nfs section to configure nfs-provisioner in values file.

```
nfs:
  server: 172.16.20.73
  path: /
  mountOptions:
  volumeName: nfs-subdir-external-provisioner-root
  # Reclaim policy for the main nfs volume
  reclaimPolicy: Retain
```

step 5: set defaultClass to true for make this storage class as default

```
storageClass:
  create: true
  # Set a provisioner name. If unset, a name will be generated.
  # provisionerName:
  # Set StorageClass as the default StorageClass
  # Ignored if storageClass.create is false
  defaultClass: true
```

step 6: create namespace "nfs" and install this provisioner on nfs namespace.

```
helm install nfs-provisioner . -n nfs
```

After these steps we have a storage provisioner up and ready to deploy pvc of our apps.

}