Objective

The objective of to deploy an object store cluster

Setup

a. Examine MinIO image on Docker Hub (https://hub.docker.com/r/bitnami/minio)

MinIO Object Store

MinIO is an open source S3 compatible object storage (https://github.com/minio/minio). It is typically deployed as a 4 node cluster.

Each MinIO node requires an attached storage to hold its data.

A MinIO server is started with the following command line and options

```
minio server \
  --address=:9000 --console-address=:9090 \
  http://<domain_name>{0...3}/data
```

where `http://<domain_name>{0...3}/data` defines a 4 node server pool and the attached storage mount point on each node is `/data`. The `{0…3}` expands to 0, 1, 2 and 3.

For example, if a 4 node server pool were to be deployed to `node1.example.net`, `node1.example.net`, `node2.example.net` and `node3.example.net` and `node4.example.net`, then the following command should be used to start the MinIO server on each of the node

```
minio server \
  --address=:9000 --console-address=:9090 \
  http://node{1...4}.example.net/data
```

Port 9000 is the MinIO server port and 9090 is the web console.

The username and password are set with **`MINIO_ROOT_USER`** and **`MINIO_ROOT_PASSWORD`** environment variables respectively. The pair is used to login to the web console.

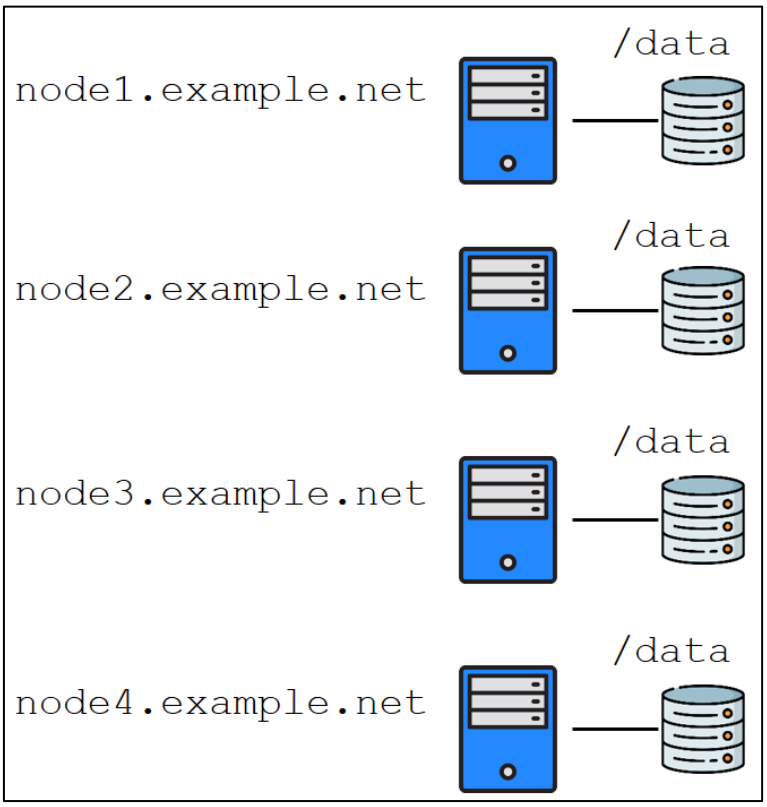
See https://min.io/docs/minio/linux/reference/minio-server/minio-server.html#command-minio.server for more command line options.

MinIO's readiness and health check are on port 900 on the following endpoints

```
/minio/health/ready
/minio/health/live
```

They are accessed with the HTTP `GET` method.

The following figure shows the above mentioned deployment.



Workshop

Create a Kubernetes deployment to deploy a 4 node MinIO cluster. Set the CPU and memory consumption of each server to 200m and 250MB respectively.

Scale the nodes only if the server requires significantly more resources.

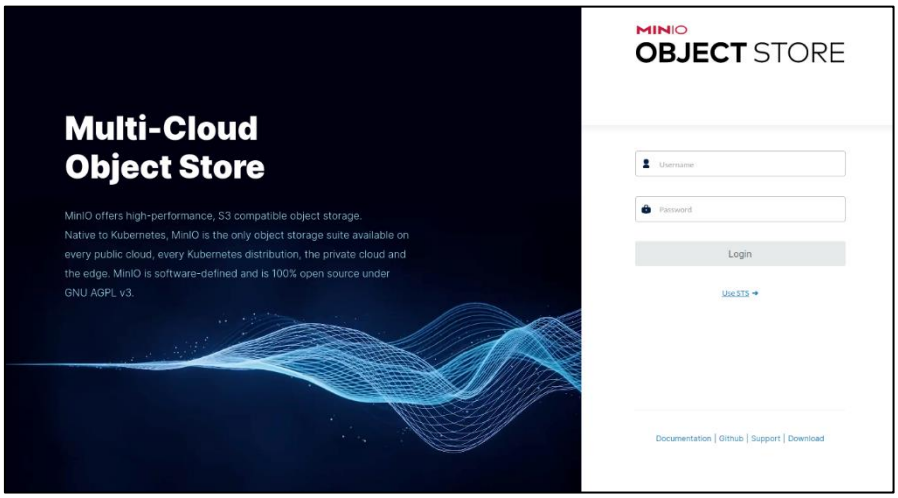
Expose the MiniIO port and web console as

```
data-<Ingress IP>.nip.io
console-<Ingress IP>.nip.io
```

Note: we are using `nip.io` to simulate a domain name for the 2 endpoints.

Protect both the endpoints from DDoS attack.

Navigate to the console endpoint and you will be greeted with the following page



Use the configure username and password to login.

Submission

Create a Git repo for this course if you have not done so. Clone the Git repo. This repo will be used for all the assignment for this course. This should the same repos as you used for previous workshops.

Email the repo's URL to your instructor. Email will be provided.

Create a directory called `workshop04` inside your repo. Placed all the file for this workshop inside `workshop04` directory.