**Prerequisites**

Before you begin, make sure you have the following:

* A Kubernetes cluster(MiniKube for local setup)
* The kubectl command-line tool installed and configured to connect to your Kubernetes cluster
* Access to a Git repository that you want to clone

**Init Container**

An init container is a specialized container that runs before the main container(s) within a pod. Its purpose is to perform initialization tasks and ensure that the environment is properly set up for the main container.

**Fetching Data from Git Repo and Storing in a Volume**

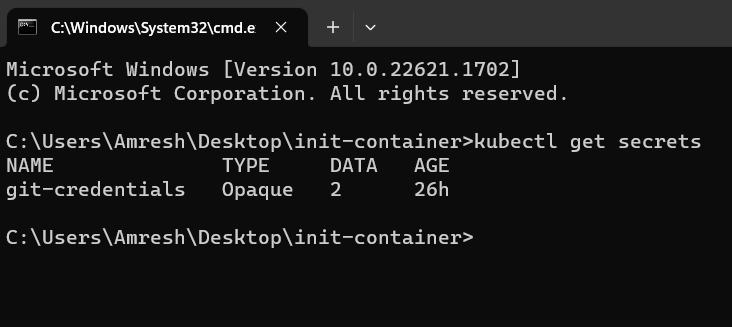
**Creating a Git Repository**

To fetch data from a Git repository, first, you need to create a repository that contains the desired data. You can use any Git hosting service like GitHub, GitLab, or Bitbucket to create the repository and upload the required files.

**Create a Secret for Git Credentials**

First, you need to create a secret containing the username and password for your Git repository. Use the following command to create the secret:

**kubectl create secret generic git-credentials --from-literal=username=amresh087 --from-literal=password=XXXXX**

****

**Defining a Pod with Init and Main Containers**

To use an init container to fetch data from the Git repository and store it in a volume, follow these steps:

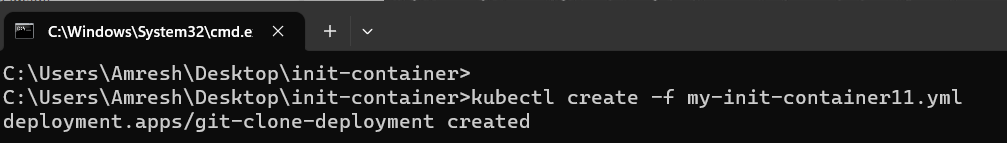
* Create a Kubernetes manifest file (e.g., **my-init-container11.yaml**) that defines a pod containing the init and main containers.

**apiVersion: apps/v1  
kind: Deployment  
metadata:  
  name: git-clone-deployment  
spec:  
  replicas: 1  
  selector:  
    matchLabels:  
      app: git-clone-app  
  template:  
    metadata:  
      labels:  
        app: git-clone-app  
    spec:  
      initContainers:  
        - name: git-clone-init  
          image: alpine/git  
          args:  
            - clone  
            -** [**https://github.com/amresh087/templet-repo.git**](https://github.com/amresh087/templet-repo.git) **- /data  
          env:  
            - name: GIT\_USERNAME  
              valueFrom:  
                secretKeyRef:  
                  name: git-credentials  
                  key: username  
            - name: GIT\_PASSWORD  
              valueFrom:  
                secretKeyRef:  
                  name: git-credentials  
                  key: password  
          volumeMounts:  
            - name: data-volume  
              mountPath: /data  
      containers:  
        - name: spring6-demo  
          image: amresh087/spring6-demo:latest  
          ports:  
          - containerPort: 8080           
          volumeMounts:  
            - name: data-volume  
              mountPath: /usr/share/config  
      volumes:  
        - name: data-volume  
          emptyDir: {}  
      imagePullSecrets:  
        - name: git-credentials**

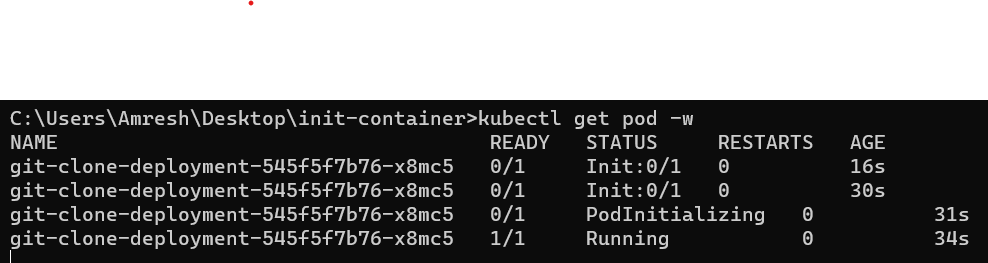
* Define a volume to store the fetched data. In this example, we will use an emptyDir volume, but you can choose other types based on your requirements.
* Specify the init container and the main container within the spec section of the manifest file.
* Mount the volume in both the init and main containers, ensuring they can access it.

**Below command we need to fire**

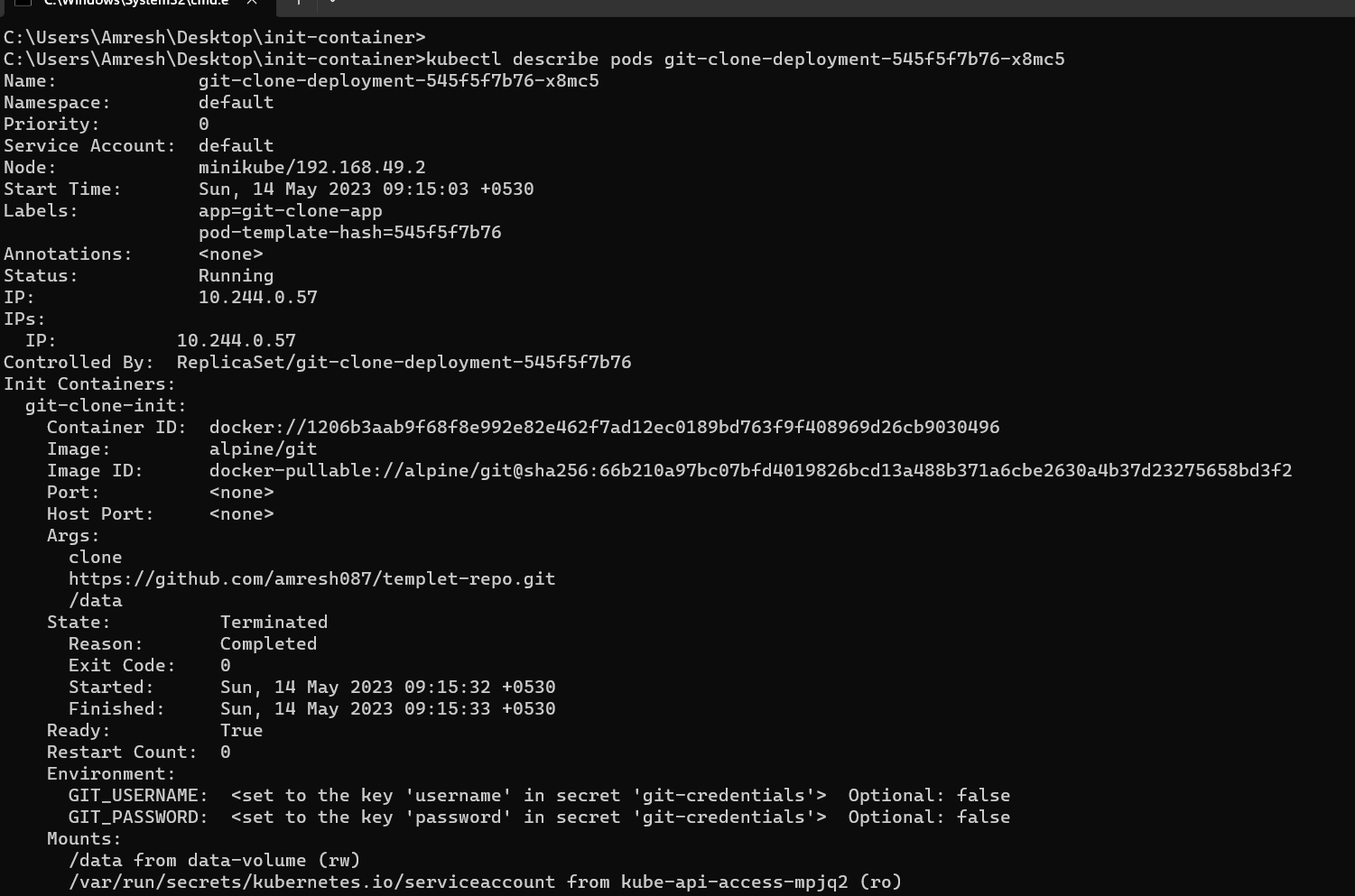
**kubectl create -f my-init-container11.yml**

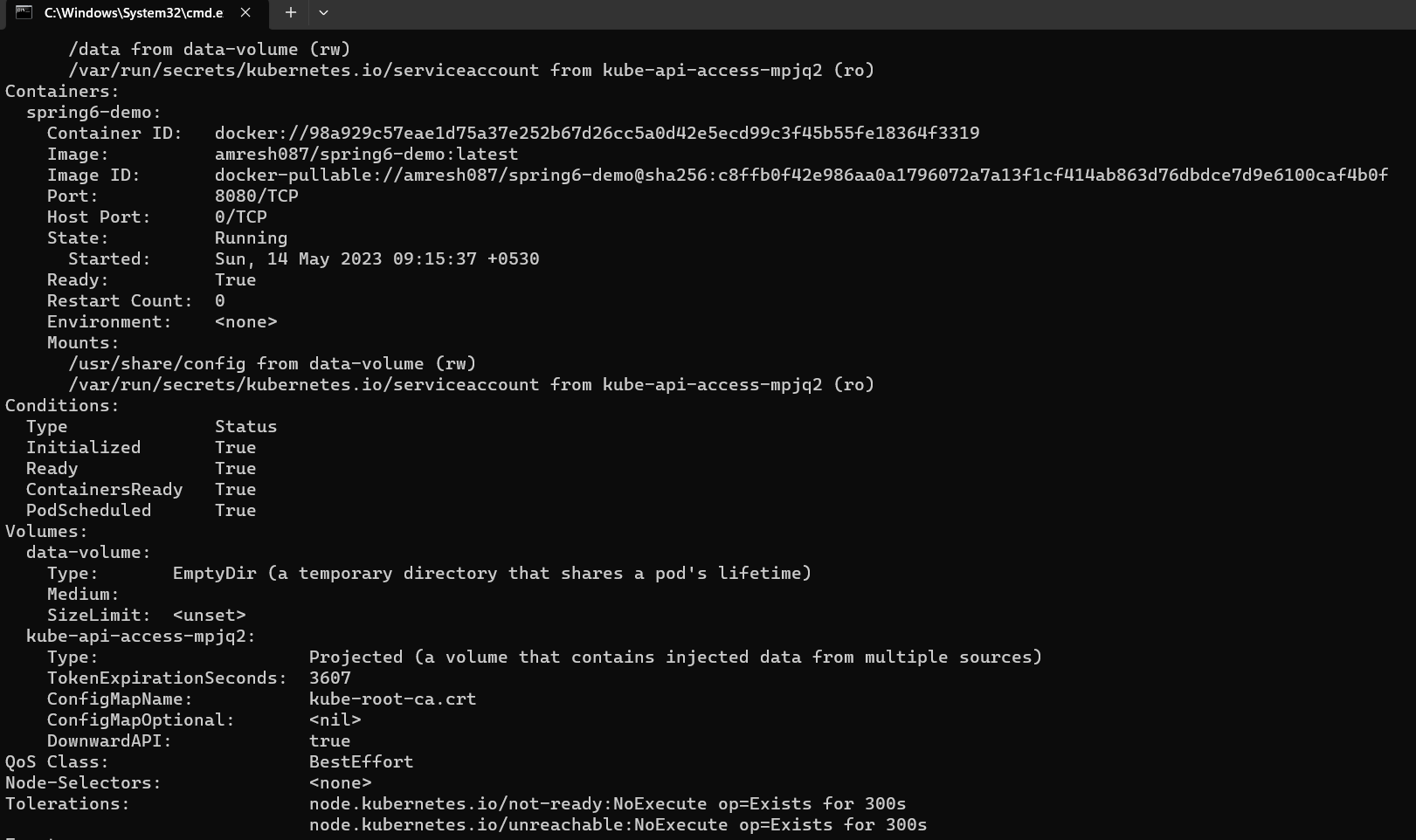
****

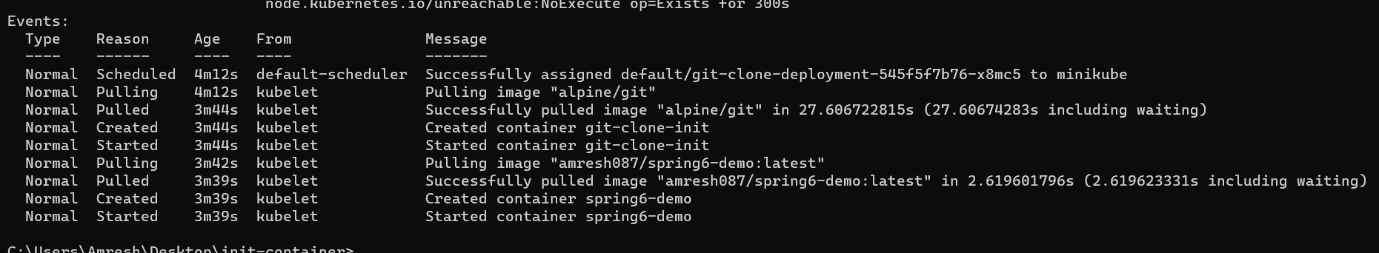
**kubectl get pods -w**

****

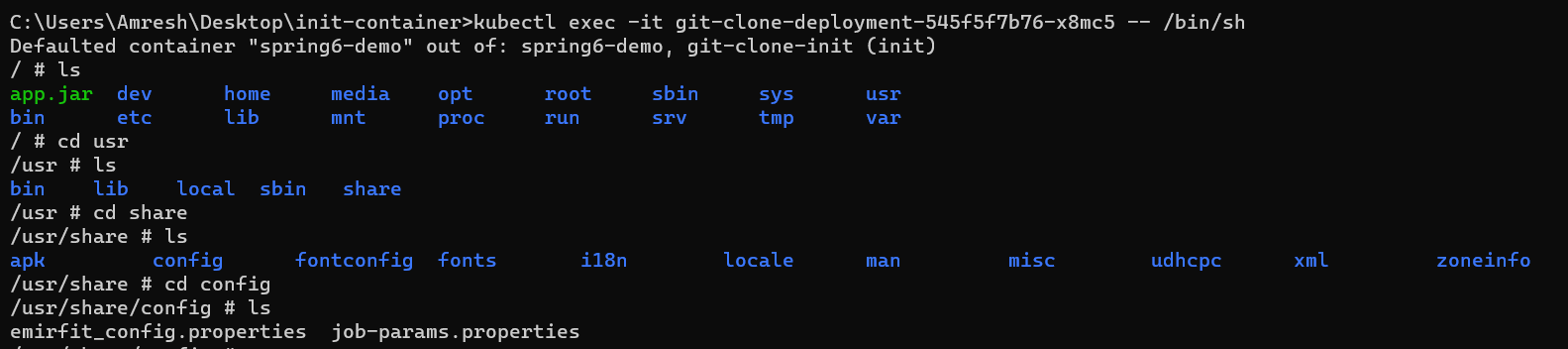
**kubectl describe pods <pod name>**

****

****

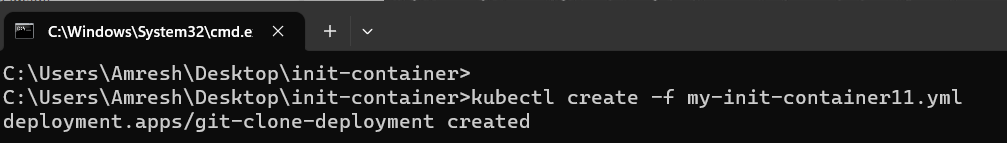
****

**kubectl exec -it <pod name> -- /bin/sh**

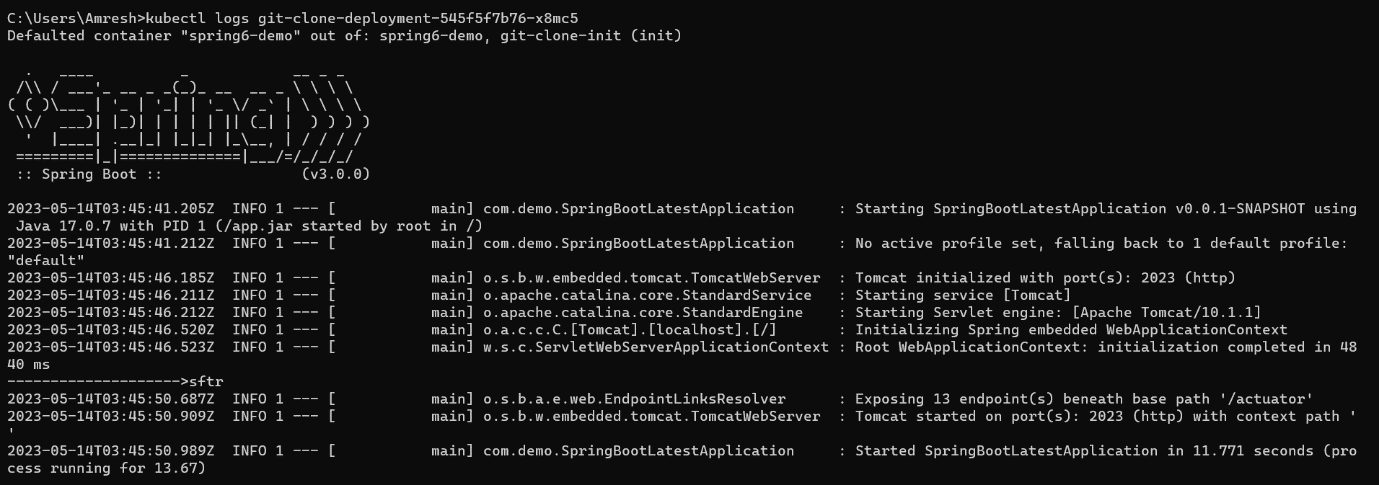
****

**/usr/share/config**

**kubectl delete -f my-init-container11.yml**

****

**Kubectl logs <pods name>**

****