BDD Deployment and Service Setup Report

June 20, 2024

1 Introduction

This report details the steps and configurations used to create a BDD (Business Driven Development) deployment and service in a Kubernetes cluster. It includes YAML configurations for deployment, services, persistent volume claims, and secrets, followed by the commands used to apply these configurations.

2 Deployment Configuration

The deployment configuration is as follows:

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: bdd-odoo-deployment
  namespace: icgroup
  labels:
    app: bdd-odoo
spec:
  replicas: 1
  selector:
    matchLabels:
      app: bdd-odoo
  template:
    metadata:
      labels:
        app: bdd-odoo
    spec:
      containers:
      - name: bdd-odoo
        image: postgres:latest
        ports:
        - containerPort: 5432
        env:
        - name: POSTGRES_DB
          value: odoo
        - name: POSTGRES_USER
          value: odoo
        - name: POSTGRES_PASSWORD
          valueFrom:
            secretKeyRef:
              name: odoo-db-secret
              key: password
        volumeMounts:
```

```
        - mountPath: /var/lib/postgresql/data
          name: bdd-odoo-storage
        resources:
            requests:
              memory: "256Mi"
              cpu: "500m"
            limits:
              memory: "512Mi"
              cpu: "1"
      volumes:
      - name: bdd-odoo-storage
        persistentVolumeClaim:
          claimName: bdd-odoo-pvc
```

3 Service Configuration

The service configuration is as follows:

```
apiVersion: v1
kind: Service
metadata:
  name: bdd-odoo-service
  namespace: icgroup
  labels:
    app: bdd-odoo
spec:
  selector:
    app: bdd-odoo
  ports:
    - protocol: TCP
      port: 5432
      targetPort: 5432
  type: ClusterIP
```

4 Persistent Volume Claim (PVC) Configuration

The PVC configuration is as follows:

```
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: bdd-odoo-pvc
  namespace: icgroup
spec:
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 10Gi
```

5 Secret Configuration

The secret configuration is as follows:

```
apiVersion: v1
kind: Secret
metadata:
  name: odoo-db-secret
  namespace: icgroup
type: Opaque
data:
  password: cGFzc3dvcmQ=  # "password" en base64
```

6 Applying the Configurations

To apply the above configurations to your Kubernetes cluster, use the following commands:

```
kubectl apply -f secret.yaml
kubectl apply -f pvc.yaml
kubectl apply -f deployment.yaml
kubectl apply -f cluster-IP-service.yaml
```

7 Conclusion

This report covered the necessary steps and configurations required to set up a BDD deployment and service in a Kubernetes environment. By following the steps and applying the provided configurations, the BDD deployment should be successfully created and operational.