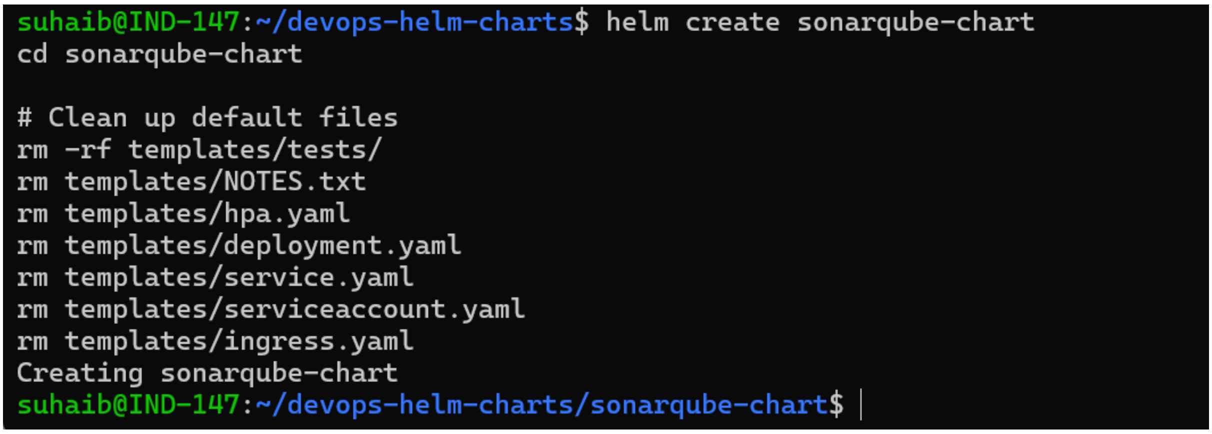
SonarQube Helm Chart Deployment

Step 1: Create SonarQube Helm Chart Structure

```
# Navigate to your project directory
cd ~/devops-helm-charts

# Create SonarQube Helm chart
helm create sonarqube-chart
cd sonarqube-chart

# Clean up default files
rm -rf templates/tests/
rm templates/NOTES.txt
rm templates/hpa.yaml
rm templates/deployment.yaml
rm templates/service.yaml
rm templates/serviceaccount.yaml
rm templates/ingress.yaml
```



```
suhaib@IND-147:~/devops-helm-charts$ helm create sonarqube-chart
cd sonarqube-chart

# Clean up default files
rm -rf templates/tests/
rm templates/NOTES.txt
rm templates/hpa.yaml
rm templates/deployment.yaml
rm templates/service.yaml
rm templates/serviceaccount.yaml
rm templates/ingress.yaml
Creating sonarqube-chart
suhaib@IND-147:~/devops-helm-charts/sonarqube-chart$
```

Step 2: Configure Chart.yaml

```
apiVersion: v2
name: sonarqube
```

```
description: A Helm chart for deploying SonarQube in Kubernetes
version: 0.1.0
appVersion: "10.3.0-community"
```

Step 3: Configure values.yaml

```
sonarqube:
  namespace: sonarqube
  image:
    repository: sonarqube
    tag: "10.3.0-community"
    pullPolicy: IfNotPresent

  resources:
    requests:
      memory: "2Gi"
      cpu: "500m"
    limits:
      memory: "4Gi"
      cpu: "2000m"

  server:
    replicas: 1
    port: 9000
    jvmOpts: "-Xmx2g -Xms512m"

  ingress:
    enabled: true
    hostname: sonarqube.yourdomain.com
    tls:
      enabled: true
      secretName: sonarqube-tls

  persistence:
    enabled: true
    storageClass: standard
    size: 20Gi
    dataSize: 10Gi
    logsSize: 5Gi
    extensionsSize: 5Gi

  admin:
    username: "admin"
    password: "admin123"  # Change in production

  database:
    # Set to true to use external database
    external: false
    # External DB settings (when external: true)
    host: ""
    port: 5432
    name: "sonarqube"
    username: "sonarqube"
    password: "sonarqube123"
    # Internal PostgreSQL settings (when external: false)
    postgresql:
      enabled: true
      image:
        repository: postgres
        tag: "13"
      storage: 10Gi
      username: "sonarqube"
      password: "sonarqube123"
      database: "sonarqube"

  config:
    sonarWebContext: "/"
    sonarJdbcMaxActive: 60
    sonarJdbcMaxIdle: 5
    sonarJdbcMinIdle: 2
    sonarJdbcMaxWait: 5000

  rbac:
    enabled: true
```

Step 4: Create Template Files

templates/configmap.yaml

```
# templates/configmap.yaml
apiVersion: v1
kind: ConfigMap
metadata:
  name: {{ .Release.Name }}-sonarqube-config
  namespace: {{ .Values.sonarqube.namespace }}
  labels:
    app.kubernetes.io/name: sonarqube
    app.kubernetes.io/instance: {{ .Release.Name }}
    app.kubernetes.io/component: configmap
data:
  sonar.properties: |
    sonar.web.context={{ .Values.sonarqube.config.sonarWebContext }}
    sonar.jdbc.maxActive={{ .Values.sonarqube.config.sonarJdbcMaxActive }}
    sonar.jdbc.maxIdle={{ .Values.sonarqube.config.sonarJdbcMaxIdle }}
    sonar.jdbc.minIdle={{ .Values.sonarqube.config.sonarJdbcMinIdle }}
    sonar.jdbc.maxWait={{ .Values.sonarqube.config.sonarJdbcMaxWait }}
    {{- if .Values.sonarqube.database.external }}
    sonar.jdbc.url=jdbc:postgresql://{{ .Values.sonarqube.database.host }}:{{ .Values.sonarqube.database.port }}/{{ .Values.sonarqube.database.name }}
    {{- else }}
    sonar.jdbc.url=jdbc:postgresql://{{ .Release.Name }}-postgresql:5432/{{ .Values.sonarqube.database.postgresql.database }}
    {{- end }}
    sonar.path.data=/opt/sonarqube/data
    sonar.path.logs=/opt/sonarqube/logs
    sonar.path.temp=/opt/sonarqube/temp
```

templates/secret.yaml

```
# templates/secret.yaml
apiVersion: v1
```

```
kind: Secret
metadata:
  name: {{ .Release.Name }}-sonarqube-secret
  namespace: {{ .Values.sonarqube.namespace }}
  labels:
    app.kubernetes.io/name: sonarqube
    app.kubernetes.io/instance: {{ .Release.Name }}
type: Opaque
data:
  admin-username: {{ .Values.sonarqube.admin.username | b64enc }}
  admin-password: {{ .Values.sonarqube.admin.password | b64enc }}
  {{- if .Values.sonarqube.database.external }}
  db-username: {{ .Values.sonarqube.database.username | b64enc }}
  db-password: {{ .Values.sonarqube.database.password | b64enc }}
  {{- else }}
  db-username: {{ .Values.sonarqube.database.postgresql.username | b64enc }}
  db-password: {{ .Values.sonarqube.database.postgresql.password | b64enc }}
  {{- end }}
```

templates/pvc.yaml

```
# templates/pvc.yaml
{{- if .Values.sonarqube.persistence.enabled }}
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: {{ .Release.Name }}-sonarqube-data-pvc
  namespace: {{ .Values.sonarqube.namespace }}
  labels:
    app.kubernetes.io/name: sonarqube
    app.kubernetes.io/instance: {{ .Release.Name }}
spec:
  accessModes:
    - ReadWriteOnce
  storageClassName: {{ .Values.sonarqube.persistence.storageClass }}
  resources:
    requests:
      storage: {{ .Values.sonarqube.persistence.dataSize }}
---
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: {{ .Release.Name }}-sonarqube-logs-pvc
  namespace: {{ .Values.sonarqube.namespace }}
  labels:
    app.kubernetes.io/name: sonarqube
    app.kubernetes.io/instance: {{ .Release.Name }}
spec:
  accessModes:
    - ReadWriteOnce
  storageClassName: {{ .Values.sonarqube.persistence.storageClass }}
  resources:
    requests:
      storage: {{ .Values.sonarqube.persistence.logsSize }}
---
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: {{ .Release.Name }}-sonarqube-extensions-pvc
  namespace: {{ .Values.sonarqube.namespace }}
  labels:
    app.kubernetes.io/name: sonarqube
    app.kubernetes.io/instance: {{ .Release.Name }}
spec:
  accessModes:
    - ReadWriteOnce
  storageClassName: {{ .Values.sonarqube.persistence.storageClass }}
  resources:
    requests:
      storage: {{ .Values.sonarqube.persistence.extensionsSize }}
{{- end }}
```

templates/postgresql.yaml

```
# templates/postgresql.yaml
{{- if and (not .Values.sonarqube.database.external) .Values.sonarqube.database.postgresql.enabled }}
apiVersion: apps/v1
kind: Deployment
metadata:
  name: {{ .Release.Name }}-postgresql
  namespace: {{ .Values.sonarqube.namespace }}
  labels:
    app.kubernetes.io/name: postgresql
    app.kubernetes.io/instance: {{ .Release.Name }}
spec:
  replicas: 1
  selector:
    matchLabels:
      app: postgresql
      release: {{ .Release.Name }}
  template:
    metadata:
      labels:
        app: postgresql
        release: {{ .Release.Name }}
    spec:
      containers:
      - name: postgresql
        image: "{{ .Values.sonarqube.database.postgresql.image.repository }}:{{ .Values.sonarqube.database.postgresql.image.tag }}"
        ports:
        - containerPort: 5432
        env:
        - name: POSTGRES_DB
          value: {{ .Values.sonarqube.database.postgresql.database }}
        - name: POSTGRES_USER
          value: {{ .Values.sonarqube.database.postgresql.username }}
        - name: POSTGRES_PASSWORD
          value: {{ .Values.sonarqube.database.postgresql.password }}
        volumeMounts:
        - name: postgresql-data
```

```
          mountPath: /var/lib/postgresql/data
        resources:
          requests:
            memory: "256Mi"
            cpu: "250m"
          limits:
            memory: "512Mi"
            cpu: "500m"
      volumes:
      - name: postgresql-data
        persistentVolumeClaim:
          claimName: {{ .Release.Name }}-postgresql-pvc
---
apiVersion: v1
kind: Service
metadata:
  name: {{ .Release.Name }}-postgresql
  namespace: {{ .Values.sonarqube.namespace }}
  labels:
    app.kubernetes.io/name: postgresql
    app.kubernetes.io/instance: {{ .Release.Name }}
spec:
  selector:
    app: postgresql
    release: {{ .Release.Name }}
  ports:
  - port: 5432
    targetPort: 5432
    protocol: TCP
---
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: {{ .Release.Name }}-postgresql-pvc
  namespace: {{ .Values.sonarqube.namespace }}
  labels:
    app.kubernetes.io/name: postgresql
    app.kubernetes.io/instance: {{ .Release.Name }}
```

```
spec:
  accessModes:
    - ReadWriteOnce
  storageClassName: {{ .Values.sonarqube.persistence.storageClass }}
  resources:
    requests:
      storage: {{ .Values.sonarqube.database.postgresql.storage }}
{{- end }}
```

templates/rbac.yaml

```
# templates/rbac.yaml
{{- if .Values.sonarqube.rbac.enabled }}
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
  name: {{ .Release.Name }}-sonarqube-role
  labels:
    app.kubernetes.io/name: sonarqube
    app.kubernetes.io/instance: {{ .Release.Name }}
rules:
- apiGroups: [""]
  resources: ["pods", "services", "configmaps", "secrets", "persistentvolumeclaims"]
  verbs: ["get", "list", "watch", "create", "update", "delete"]
- apiGroups: ["apps"]
  resources: ["deployments", "statefulsets"]
  verbs: ["get", "list", "watch", "create", "update", "delete"]
- apiGroups: ["extensions", "networking.k8s.io"]
  resources: ["ingresses"]
  verbs: ["get", "list", "watch", "create", "update", "delete"]
---
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRoleBinding
metadata:
  name: {{ .Release.Name }}-sonarqube-binding
  labels:
    app.kubernetes.io/name: sonarqube
    app.kubernetes.io/instance: {{ .Release.Name }}
subjects:
- kind: ServiceAccount
  name: {{ .Release.Name }}-sonarqube-sa
  namespace: {{ .Values.sonarqube.namespace }}
roleRef:
  kind: ClusterRole
  name: {{ .Release.Name }}-sonarqube-role
  apiGroup: rbac.authorization.k8s.io
---
apiVersion: v1
kind: ServiceAccount
metadata:
  name: {{ .Release.Name }}-sonarqube-sa
  namespace: {{ .Values.sonarqube.namespace }}
  labels:
    app.kubernetes.io/name: sonarqube
    app.kubernetes.io/instance: {{ .Release.Name }}
{{- end }}
```

templates/deployment.yaml

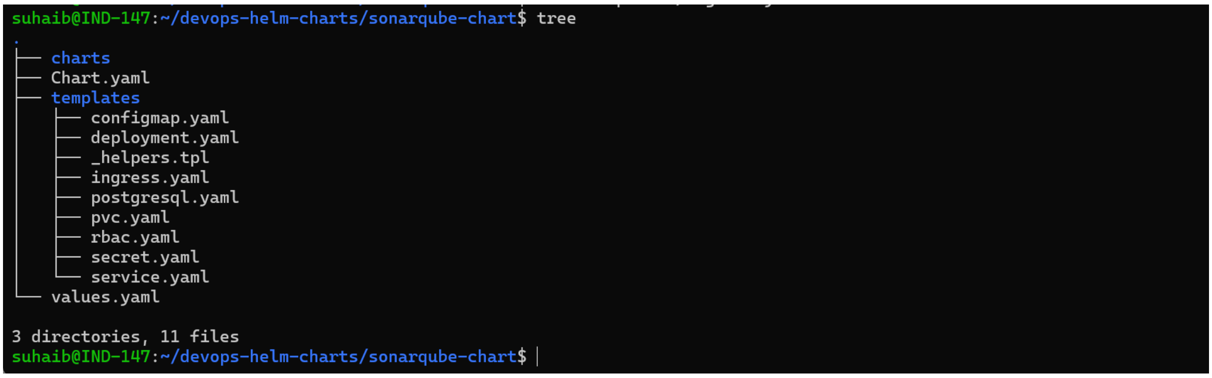
```
# templates/deployment.yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: {{ .Release.Name }}-sonarqube
  namespace: {{ .Values.sonarqube.namespace }}
  labels:
    app.kubernetes.io/name: sonarqube
    app.kubernetes.io/instance: {{ .Release.Name }}
spec:
  replicas: {{ .Values.sonarqube.server.replicas }}
  selector:
    matchLabels:
      app: sonarqube
      release: {{ .Release.Name }}
  template:
    metadata:
      labels:
        app: sonarqube
        release: {{ .Release.Name }}
    spec:
      {{- if .Values.sonarqube.rbac.enabled }}
      serviceAccountName: {{ .Release.Name }}-sonarqube-sa
      {{- end }}
      initContainers:
      - name: init-sysctl
        image: busybox:1.35
        command:
        - sh
        - -c
        - |
          sysctl -w vm.max_map_count=524288
          sysctl -w fs.file-max=131072
        securityContext:
          privileged: true
      containers:
      - name: sonarqube
        image: "{{ .Values.sonarqube.image.repository }}:{{ .Values.sonarqub
e.image.tag }}"
        imagePullPolicy: {{ .Values.sonarqube.image.pullPolicy }}
        ports:
        - containerPort: {{ .Values.sonarqube.server.port }}
          name: http
        env:
        - name: SONAR_JDBC_USERNAME
          valueFrom:
            secretKeyRef:
              name: {{ .Release.Name }}-sonarqube-secret
              key: db-username
        - name: SONAR_JDBC_PASSWORD
          valueFrom:
            secretKeyRef:
              name: {{ .Release.Name }}-sonarqube-secret
              key: db-password
        - name: SONAR_WEB_JAVAADDITIONALOPTS
          value: "{{ .Values.sonarqube.server.jvmOpts }}"
        volumeMounts:
        - name: sonarqube-config
          mountPath: /opt/sonarqube/conf/sonar.properties
          subPath: sonar.properties
        {{- if .Values.sonarqube.persistence.enabled }}
        - name: sonarqube-data
          mountPath: /opt/sonarqube/data
        - name: sonarqube-logs
          mountPath: /opt/sonarqube/logs
        - name: sonarqube-extensions
          mountPath: /opt/sonarqube/extensions
        {{- end }}
        resources:
          {{- toYaml .Values.sonarqube.resources | nindent 10 }}
        livenessProbe:
          httpGet:
            path: /api/system/status
            port: {{ .Values.sonarqube.server.port }}
          initialDelaySeconds: 120
          periodSeconds: 30
          timeoutSeconds: 10
        readinessProbe:
          httpGet:
            path: /api/system/status
            port: {{ .Values.sonarqube.server.port }}
          initialDelaySeconds: 30
          periodSeconds: 10
          timeoutSeconds: 5
      volumes:
      - name: sonarqube-config
        configMap:
          name: {{ .Release.Name }}-sonarqube-config
      {{- if .Values.sonarqube.persistence.enabled }}
      - name: sonarqube-data
        persistentVolumeClaim:
          claimName: {{ .Release.Name }}-sonarqube-data-pvc
      - name: sonarqube-logs
        persistentVolumeClaim:
          claimName: {{ .Release.Name }}-sonarqube-logs-pvc
      - name: sonarqube-extensions
        persistentVolumeClaim:
          claimName: {{ .Release.Name }}-sonarqube-extensions-pvc
      {{- end }}
```

templates/service.yaml

```
# templates/service.yaml
apiVersion: v1
kind: Service
metadata:
  name: {{ .Release.Name }}-sonarqube-service
  namespace: {{ .Values.sonarqube.namespace }}
  labels:
    app.kubernetes.io/name: sonarqube
    app.kubernetes.io/instance: {{ .Release.Name }}
spec:
  selector:
    app: sonarqube
    release: {{ .Release.Name }}
  ports:
  - port: 80
    targetPort: {{ .Values.sonarqube.server.port }}
    protocol: TCP
    name: http
  type: ClusterIP
```

templates/ingress.yaml

```
# templates/ingress.yaml
{{- if .Values.sonarqube.ingress.enabled }}
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: {{ .Release.Name }}-sonarqube-ingress
  namespace: {{ .Values.sonarqube.namespace }}
  labels:
    app.kubernetes.io/name: sonarqube
    app.kubernetes.io/instance: {{ .Release.Name }}
  annotations:
    nginx.ingress.kubernetes.io/ssl-redirect: "true"
    nginx.ingress.kubernetes.io/proxy-body-size: "50m"
    nginx.ingress.kubernetes.io/proxy-read-timeout: "300"
    nginx.ingress.kubernetes.io/proxy-send-timeout: "300"
spec:
  ingressClassName: nginx
  rules:
  - host: {{ .Values.sonarqube.ingress.hostname }}
    http:
      paths:
      - path: /
        pathType: Prefix
        backend:
          service:
            name: {{ .Release.Name }}-sonarqube-service
            port:
              number: 80
  {{- if .Values.sonarqube.ingress.tls.enabled }}
  tls:
  - hosts:
    - {{ .Values.sonarqube.ingress.hostname }}
    secretName: {{ .Values.sonarqube.ingress.tls.secretName }}
  {{- end }}
{{- end }}
```



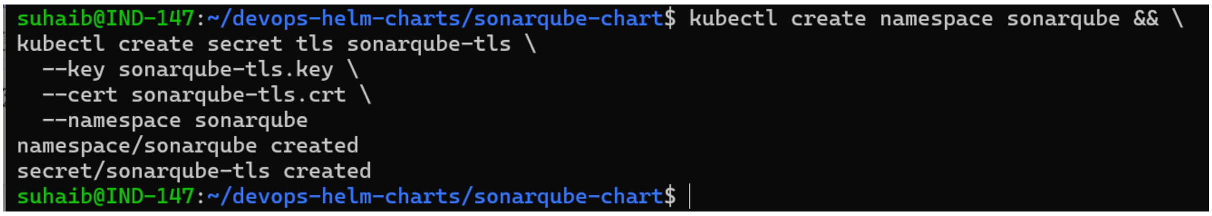
```
suhaib@IND-147:~/devops-helm-charts/sonarqube-chart$ tree
.
├── charts
├── Chart.yaml
├── templates
│   ├── configmap.yaml
│   ├── deployment.yaml
│   ├── _helpers.tpl
│   ├── ingress.yaml
│   ├── postgresql.yaml
│   ├── pvc.yaml
│   ├── rbac.yaml
│   ├── secret.yaml
│   └── service.yaml
└── values.yaml

3 directories, 11 files
suhaib@IND-147:~/devops-helm-charts/sonarqube-chart$ |
```

Step 5: Deployment and Configuration Commands

Create TLS Certificate for Ingress

```
# Create self-signed certificate for local development
openssl req -x509 -nodes -days 365 -newkey rsa:2048 \
  -keyout sonarqube-tls.key \
  -out sonarqube-tls.crt \
  -subj "/CN=sonarqube.local/O=sonarqube.local"

# Create TLS secret in Kubernetes
kubectl create namespace sonarqube && \
kubectl create secret tls sonarqube-tls \
  --key sonarqube-tls.key \
  --cert sonarqube-tls.crt \
  --namespace sonarqube
```

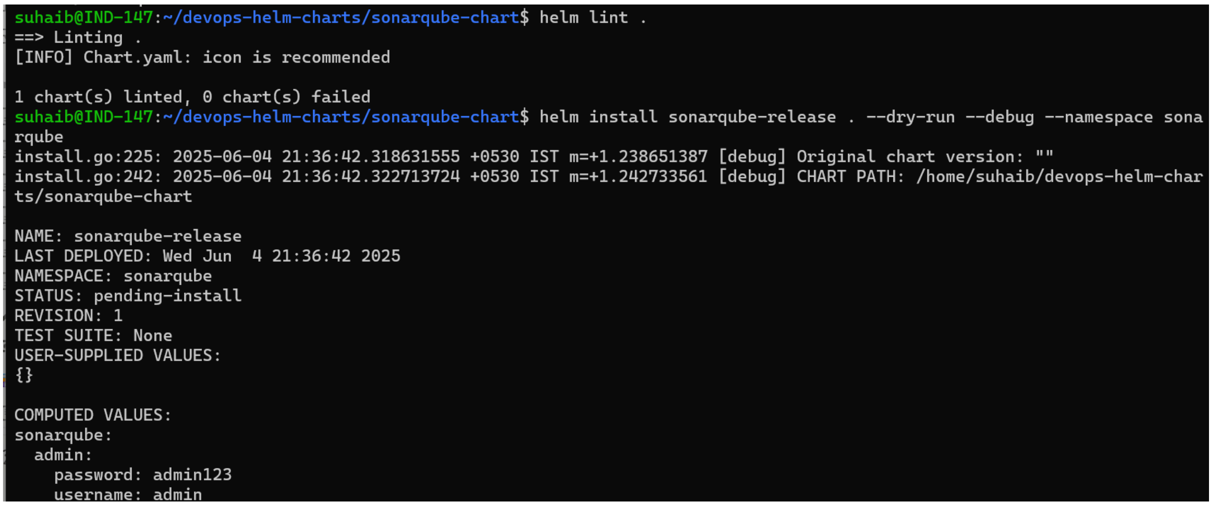
```
suhaib@IND-147:~/devops-helm-charts/sonarqube-chart$ openssl req -x509 -nodes -days 365 -newkey rsa:2048 \
  -keyout sonarqube-tls.key \
  -out sonarqube-tls.crt \
  -subj "/CN=sonarqube.local/O=sonarqube.local"
..+.....+...........+...+...+....+..+....++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++*...+++++++++
++++++++++++++++++++++++++++++++++++++++++++++++++++++++*......+.................+....+.........+..+...+.........+......
+....+......+.......+.........+.+......+........+.......+.......+......+........+...+.............+....+.........+...+......
.........+..+.+.....+.......+...+...........+......+.......+.........+......+..+.........+..+.+.........+..+...+...+....+........
.+...........+...+.......+.....+....+.....+......+...+.....+....+..+......+...........+............+...+......+.+...+.
..+.....+....+...+...+..+.+............+........+.+...........+.........+....+..+...+.+...........+....+..+...
+....+.....+.+..+........+..+................+.....+...+..................+.....+.+..............+.............+..+........
..+......+..+.+.....+.+........+.+...........+....+..............+....+...+...+.........+....+...+.............++++++++
+++++++++++++++++++++++++++++++++++++++++++++++++++++++++
..+....+...+.....+......+......+.+...+...........+.+.....+....+.+....+......+..............+.+++++++++++++++++++++++++++++++++
++++++++++++++++++++++++++++++++*....+...+...+.........+......+......+.......+..++++++++++++++++++++++++++++++++++++++++++++++++
+++++++++++++++++*.+....+.........+.......+...................+.......+.................+...+..+.......+.......+..
+......+...+.........+.............+...+...+....+...+......+...+...+...+....+...+...+..+.+.....+......................+.+..........
...................+..+.+...+..+.........+...+.+..........+...+..+...+..+.+..+.......+............+............+.........
+.+...+...+.........+.......+...+...+....+..........+......+.........+.........+.+..+...+....+..+..........+......+.......+...+.........
...+...........+......+...+.......+...+............+.....+.+..........+.............+...+.....+.......+.......+.............+...
....+...+.+......+.+...............+...+.+............+..........+.......+..........+.......+......+.+...+...............+......+...+.
.+...+...+....+.............+...+........+..+.+..+....+.........+...+...+..........+...........+.....+...+.....+.+.........................+..
...+......+...+.+..........+..+...........+.........+...+..+.......................+...+..+....+.............+......+.......+..+.+...
....+.+..+...+...+..............+...+..........+.+.........................+.....+.......................+...
..+...+.....+...+........+..+.+.........+......+.+..........+........+..........................+............+......+.....+...+.+..
```



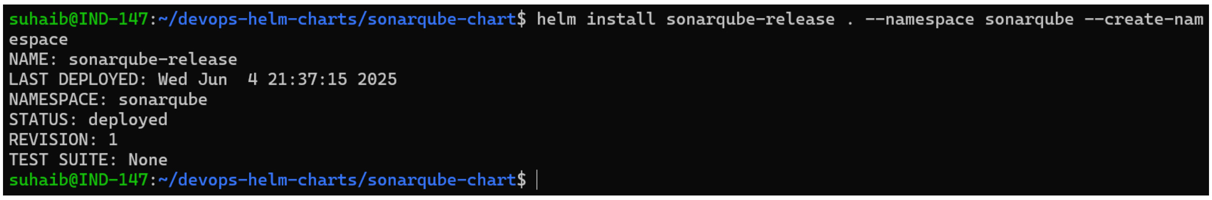
```
suhaib@IND-147:~/devops-helm-charts/sonarqube-chart$ kubectl create namespace sonarqube && \
kubectl create secret tls sonarqube-tls \
  --key sonarqube-tls.key \
  --cert sonarqube-tls.crt \
  --namespace sonarqube
namespace/sonarqube created
secret/sonarqube-tls created
suhaib@IND-147:~/devops-helm-charts/sonarqube-chart$
```

Deploy SonarQube Using Helm

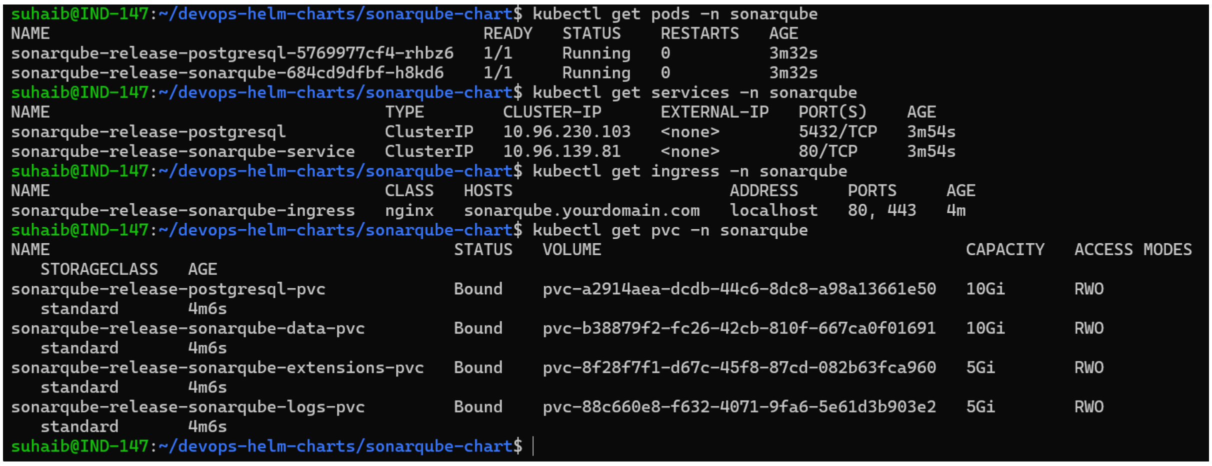
```
# Validate the Helm chart
helm lint .

# Dry run to check templates
helm install sonarqube-release . --dry-run --debug --namespace sonarqube

# Install SonarQube
helm install sonarqube-release . --namespace sonarqube --create-namespace

# Check deployment status
kubectl get pods -n sonarqube
kubectl get services -n sonarqube
kubectl get ingress -n sonarqube
kubectl get pvc -n sonarqube
```



```
suhaib@IND-147:~/devops-helm-charts/sonarqube-chart$ helm lint .
==> Linting .
[INFO] Chart.yaml: icon is recommended

1 chart(s) linted, 0 chart(s) failed
suhaib@IND-147:~/devops-helm-charts/sonarqube-chart$ helm install sonarqube-release . --dry-run --debug --namespace sona
rqube
install.go:225: 2025-06-04 21:36:42.318631555 +0530 IST m=+1.238651387 [debug] Original chart version: ""
install.go:242: 2025-06-04 21:36:42.322713724 +0530 IST m=+1.242733561 [debug] CHART PATH: /home/suhaib/devops-helm-char
ts/sonarqube-chart

NAME: sonarqube-release
LAST DEPLOYED: Wed Jun  4 21:36:42 2025
NAMESPACE: sonarqube
STATUS: pending-install
REVISION: 1
TEST SUITE: None
USER-SUPPLIED VALUES:
{}

COMPUTED VALUES:
sonarqube:
  admin:
    password: admin123
    username: admin
```



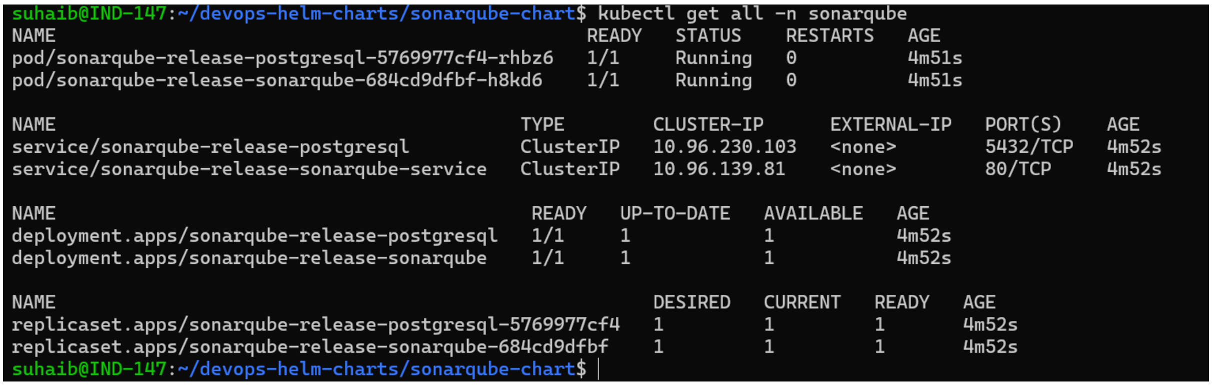
```
suhaib@IND-147:~/devops-helm-charts/sonarqube-chart$ helm install sonarqube-release . --namespace sonarqube --create-nam
espace
NAME: sonarqube-release
LAST DEPLOYED: Wed Jun  4 21:37:15 2025
NAMESPACE: sonarqube
STATUS: deployed
REVISION: 1
TEST SUITE: None
suhaib@IND-147:~/devops-helm-charts/sonarqube-chart$
```



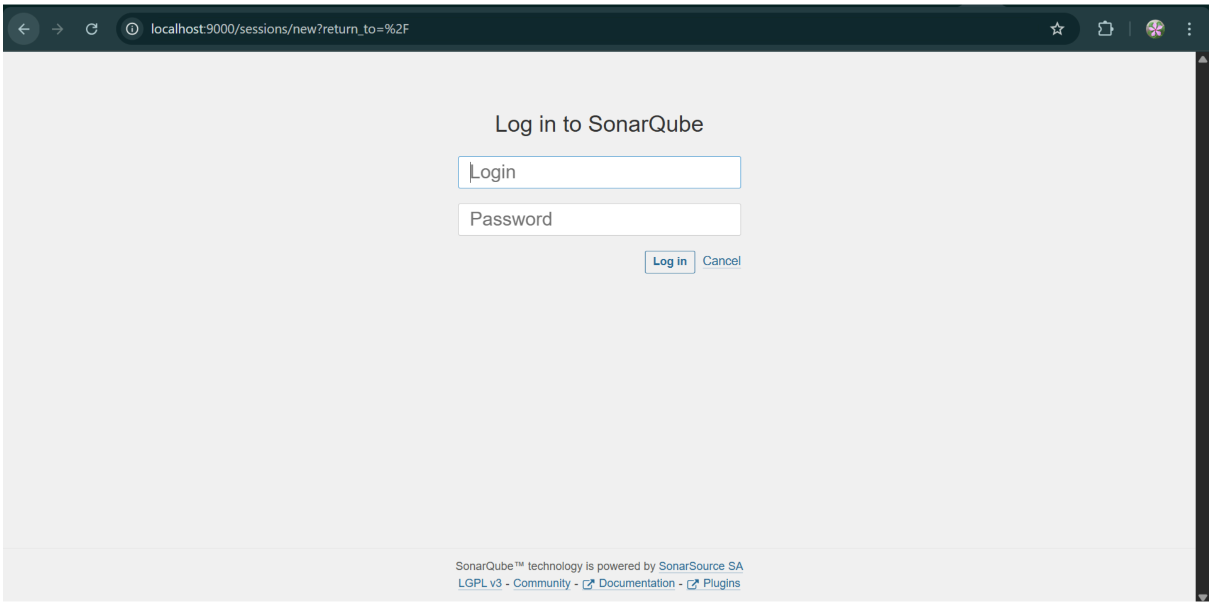
```
suhaib@IND-147:~/devops-helm-charts/sonarqube-chart$ kubectl get pods -n sonarqube
NAME                                              READY   STATUS    RESTARTS   AGE
sonarqube-release-postgresql-5769977cf4-rhbz6     1/1     Running   0          3m32s
sonarqube-release-sonarqube-684cd9dfbf-h8kd6      1/1     Running   0          3m32s
suhaib@IND-147:~/devops-helm-charts/sonarqube-chart$ kubectl get services -n sonarqube
NAME                                    TYPE        CLUSTER-IP      EXTERNAL-IP   PORT(S)    AGE
sonarqube-release-postgresql            ClusterIP   10.96.230.103   <none>        5432/TCP   3m54s
sonarqube-release-sonarqube-service     ClusterIP   10.96.139.81    <none>        80/TCP     3m54s
suhaib@IND-147:~/devops-helm-charts/sonarqube-chart$ kubectl get ingress -n sonarqube
NAME                                    CLASS   HOSTS                      ADDRESS     PORTS     AGE
sonarqube-release-sonarqube-ingress     nginx   sonarqube.yourdomain.com   localhost   80, 443   4m
suhaib@IND-147:~/devops-helm-charts/sonarqube-chart$ kubectl get pvc -n sonarqube
NAME                                          STATUS   VOLUME                                     CAPACITY   ACCESS MODES
   STORAGECLASS   AGE
sonarqube-release-postgresql-pvc              Bound    pvc-a2914aea-dcdb-44c6-8dc8-a98a13661e50   10Gi       RWO
   standard       4m6s
sonarqube-release-sonarqube-data-pvc          Bound    pvc-b38879f2-fc26-42cb-810f-667ca0f01691   10Gi       RWO
   standard       4m6s
sonarqube-release-sonarqube-extensions-pvc    Bound    pvc-8f28f7f1-d67c-45f8-87cd-082b63fca960   5Gi        RWO
   standard       4m6s
sonarqube-release-sonarqube-logs-pvc          Bound    pvc-88c660e8-f632-4071-9fa6-5e61d3b903e2   5Gi        RWO
   standard       4m6s
suhaib@IND-147:~/devops-helm-charts/sonarqube-chart$
```

Verify Installation

```
# Check all resources
kubectl get all -n sonarqube

# Check logs
kubectl logs -n sonarqube deployment/sonarqube-release-sonarqube -f

# Check SonarQube status
kubectl port-forward -n sonarqube service/sonarqube-release-sonarqube-service 9000:80
# Then access http://localhost:9000
```



```
suhaib@IND-147:~/devops-helm-charts/sonarqube-chart$ kubectl get all -n sonarqube
NAME                                                  READY   STATUS    RESTARTS   AGE
pod/sonarqube-release-postgresql-5769977cf4-rhbz6     1/1     Running   0          4m51s
pod/sonarqube-release-sonarqube-684cd9dfbf-h8kd6      1/1     Running   0          4m51s

NAME                                            TYPE        CLUSTER-IP      EXTERNAL-IP   PORT(S)    AGE
service/sonarqube-release-postgresql            ClusterIP   10.96.230.103   <none>        5432/TCP   4m52s
service/sonarqube-release-sonarqube-service     ClusterIP   10.96.139.81    <none>        80/TCP     4m52s

NAME                                            READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/sonarqube-release-postgresql    1/1     1            1           4m52s
deployment.apps/sonarqube-release-sonarqube     1/1     1            1           4m52s

NAME                                                      DESIRED   CURRENT   READY   AGE
replicaset.apps/sonarqube-release-postgresql-5769977cf4   1         1         1       4m52s
replicaset.apps/sonarqube-release-sonarqube-684cd9dfbf    1         1         1       4m52s
suhaib@IND-147:~/devops-helm-charts/sonarqube-chart$
```



Log in to SonarQube

SonarQube™ technology is powered by SonarSource SA
LGPL v3 - Community - Documentation - Plugins

Step 6: Management and Maintenance

Upgrade SonarQube

```
# Update values.yaml with new configuration
# Then upgrade
helm upgrade sonarqube-release . --namespace sonarqube

# Check upgrade status
helm status sonarqube-release -n sonarqube
helm history sonarqube-release -n sonarqube
```

Rollback if needed

```
# Rollback to previous version
helm rollback sonarqube-release 1 -n sonarqube
```

Uninstall

```
# Uninstall SonarQube
helm uninstall sonarqube-release -n sonarqube
```

```
# Delete namespace (optional)
kubectl delete namespace sonarqube
```

Configuration Options

External Database Configuration

To use an external PostgreSQL database, update `values.yaml`:

```
sonarqube:
  database:
    external: true
    host: "your-postgres-host"
    port: 5432
    name: "sonarqube"
    username: "sonarqube"
    password: "your-secure-password"
```

Resource Customization

Adjust resources based on your requirements:

```
sonarqube:
  resources:
    requests:
      memory: "4Gi"
      cpu: "1000m"
    limits:
      memory: "8Gi"
      cpu: "4000m"
```

JVM Options

Customize JVM settings:

```
sonarqube:
  server:
    jvmOpts: "-Xmx4g -Xms1g -XX:+HeapDumpOnOutOfMemoryError"
```

Troubleshooting

Common Issues and Solutions

1. **Init container fails**: Ensure your cluster allows privileged containers
2. **Database connection issues**: Verify database credentials and connectivity
3. **Memory issues**: Increase resource limits and JVM heap size
4. **Storage issues**: Ensure sufficient disk space and proper StorageClass

Check System Requirements

```
# Verify system requirements are met
kubectl exec -n sonarqube deployment/sonarqube-release-sonarqube -- \
  sh -c "cat /proc/sys/vm/max_map_count && cat /proc/sys/fs/file-max"
```