Set Up Kubernetes Environment

Last Modified on 01/03/2024 4:15 am EST

Overview

Kubernetes is a powerful container orchestration tool that automates container deployment, scaling, and management. It simplifies the process of managing containerized applications across different environments, and makes it easy to deploy and scale applications on-demand. It allows developers to deploy and manage containers across multiple hosts, and provides tools for managing container lifecycles, resource allocation, and scaling.

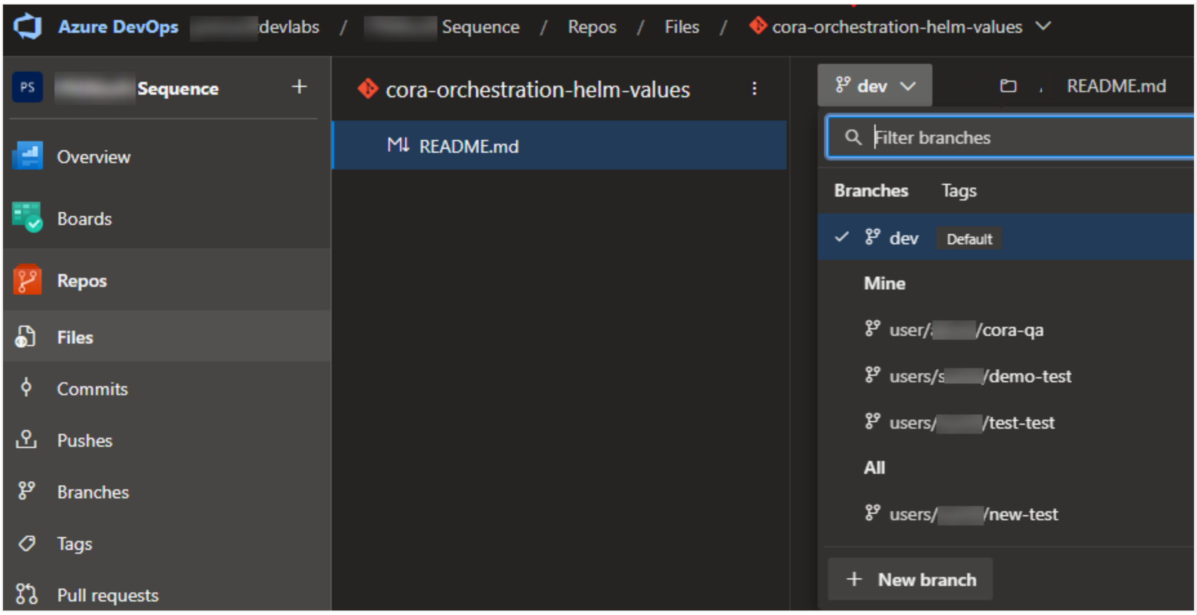
Procedure to set up a local environment for Kubernetes

1. Install WSL.
   For details, see this link.
2. Verify WSL installation.
   In command line (CMD) run, `wsl --list --verbose` .
3. Install Rancher Desktop.
   For details, see this link.
4. Run Rancher Desktop and verify installation:
   a. Verify Helm installation.
      In CMD run, `helm version` .
   b. Verify K8S installation.
      In CMD run, `kubectl version --short` .
5. Create a namespace for the cluster.
   In CMD run, `kubectl create namespace monitoring` .
6. Extract *o11y-charts.zip*, and open it with CMD.
7. Install cluster with Helm.
   In CMD run, `helm install o11y . -n monitoring -f values.yaml` .
8. Verify cluster installation.
   In CMD run, `kubectl get pod -A -n monitoring` .
9. Open Rancher Desktop Preferences:
   a. Right-click on Rancher icon, and then select Preferences.
   b. Go to Port Forwarding settings.
10. Forward required ports in Rancher Desktop Port Forwarding.
    a. Find o11y-grafana and forward port to access Grafana.
    b. Find o11y-loki and forward port for Promtail agent.
    c. Find o11y-prometheus-server and forward port to access Prometheus.
    d. Find o11y-tempo grpc-tempo-otlp and forward port for OTEL Collector agent.
11. Extract attached *agents.zip* to C:\agents.
12. Configure Promtail agent, open and edit C:\agents\promtail\config.yaml.
    a. Get port number from step 10.b.
    b. Get path to app logs from:
       Administration || Flowtime || JES || ADSS || BRS > Configuration\Logging\logging.sinks.json
13. Run Promtail agent. In CMD run, `run.bat` .
14. Configure OTEL Collector agent. Open and edit C:\agents\otel\config.yaml.
    a. For the exporters section, get port number from step 10.d.
    b. For the receivers section, get port number from:
       Administration || Flowtime || JES || ADSS || BRS> Configuration\ Observability\ o11y.netfx.json.
15. Run OTEL Collector agent.
    In CMD run, `run.bat`

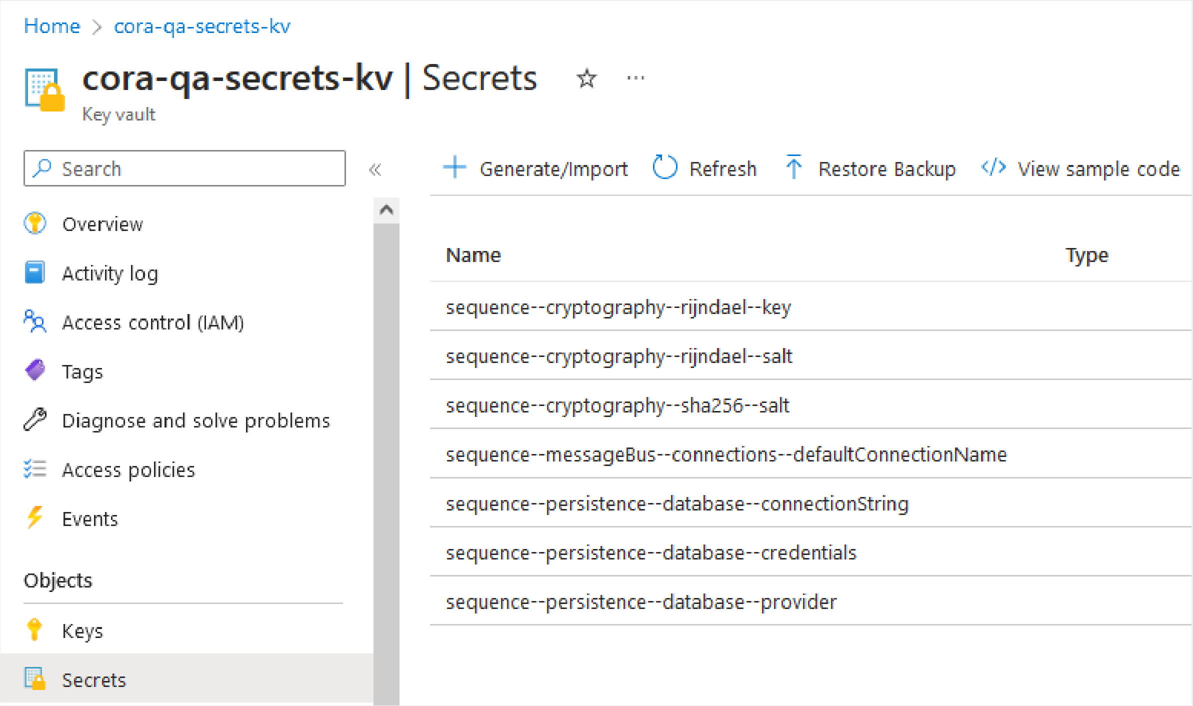
16. Access Grafana by navigating to http://localhost:XXXX in your web browser, where XXXX is the port number from step 10.a.
    - Default user: admin
    - Default password: admin
17. Access Prometheus by navigating to http://localhost:XXXX in your web browser, where XXXX is the port number from step 10.c.

Procedure to install a Kubernetes environment on a cluster

1. In the GIT repository, create a new branch from dev for your deployment.
   AzureDevops>Repos><cora-orchestration-helm-values>
   You must create a branch with the following name convention:
   users/<branch-name>/<namespace>>
   For each environment, the namespace is unique within the cluster.
   For example,users/alice/cora-qa

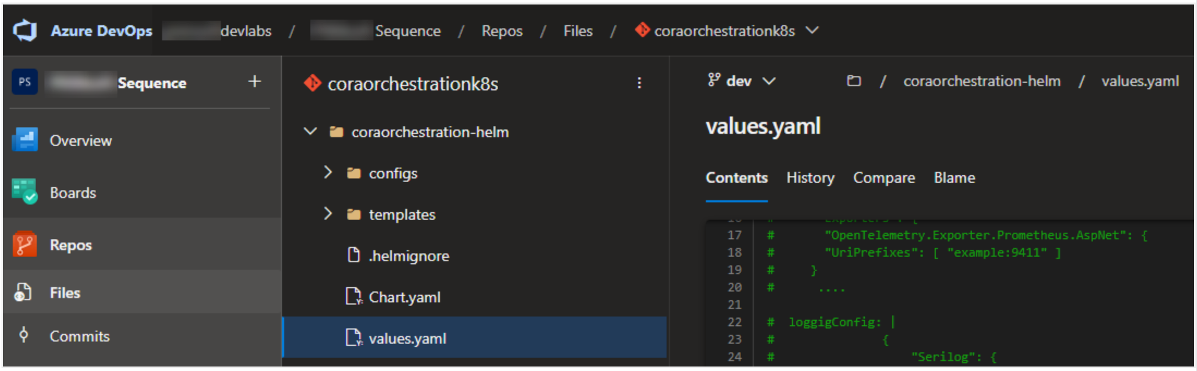

Azure DevOps [redacted]devlabs / [redacted] Sequence / Repos / Files / cora-orchestration-helm-values
PS [redacted] Sequence +
Overview
Boards
Repos
Files
Commits
Pushes
Branches
Tags
Pull requests
cora-orchestration-helm-values
README.md
dev
README.md
Filter branches
Branches
Tags
✓ dev Default
Mine
user/[redacted]/cora-qa
users/s[redacted]/demo-test
users/[redacted]/test-test
All
users/[redacted]/new-test
+ New branch

2. Add, in the *values.yaml* file, all configuration details like database credential, database connection string, database provider, connection name, that are present in Azure Key Vault.



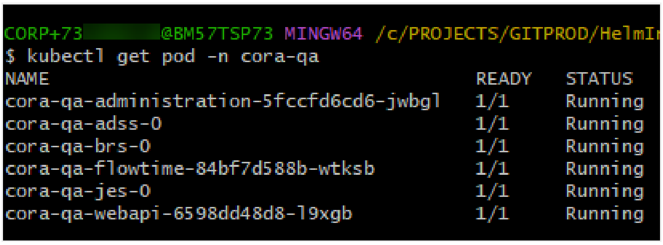
Name	Type
sequence--cryptography--rijndael--key	
sequence--cryptography--rijndael--salt	
sequence--cryptography--sha256--salt	
sequence--messageBus--connections--defaultConnectionName	
sequence--persistence--database--connectionString	
sequence--persistence--database--credentials	
sequence--persistence--database--provider	

3. Check the *values.yaml* file, if the parameters and variables are required to be changed.
   You can check the existing parameters and variables in the *values.yaml* file here.



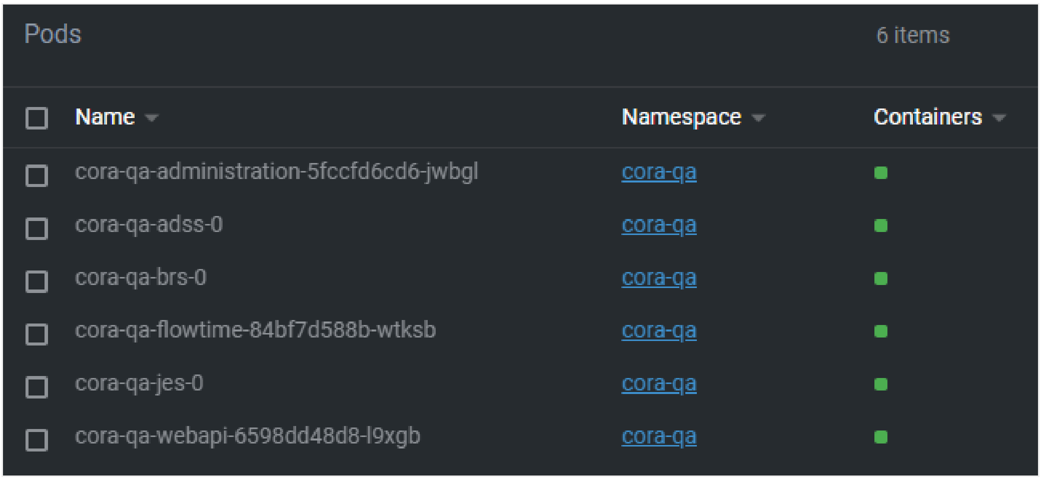
```
#        "OpenTelemetry.Exporter.Prometheus.AspNet": {
#        "UriPrefixes": [ "example:9411" ]
#      }
#       ....

#  loggigConfig: |
#                 {
#                     "Serilog": {
```

4. In repository, open the code in any IDE (VS is recommended).
   For example, ":users/alice/cora-qa" branch.
5. Push your changes to the GIT repository. (Or from CMD or IDE).
   The applications are deployed to the cluster automatically.
6. Wait for a few minutes for pipeline to finish deployment of applications in the cluster.
7. Check the cluster with the namespace. For example, namespace is *cora-qa* in our case here.
8. Create an all-web application, run https://<namespace>-<name application>.pnmsoftlabs.com.
   For example, to create namespace *cora-qa*, run the following:
   - Administration: https://cora-qa-admin.pnmsoftlabs.com
   - Flowtime:https://cora-qa-flow.pnmsoftlabs.com
   - WebApi: https://cora-qa-webapi.pnmsoftlabs.com
9. Check if the status of all the applications is running.
   In CMD, run `kubectl get pod -n cora-qa`.



```
CORP+73[redacted]@BM57TSP73 MINGW64 /c/PROJECTS/GITPROD/HelmI
$ kubectl get pod -n cora-qa
NAME                                   READY   STATUS
cora-qa-administration-5fccfd6cd6-jwbgl   1/1     Running
cora-qa-adss-0                         1/1     Running
cora-qa-brs-0                          1/1     Running
cora-qa-flowtime-84bf7d588b-wtksb      1/1     Running
cora-qa-jes-0                          1/1     Running
cora-qa-webapi-6598dd48d8-19xgb        1/1     Running
```

Or check in one of UI Kubernetes IDE.



Pods

6 items

□	Name	Namespace	Containers
□	cora-qa-administration-5fccfd6cd6-jwbgl	cora-qa	🟩
□	cora-qa-adss-0	cora-qa	🟩
□	cora-qa-brs-0	cora-qa	🟩
□	cora-qa-flowtime-84bf7d588b-wtksb	cora-qa	🟩
□	cora-qa-jes-0	cora-qa	🟩
□	cora-qa-webapi-6598dd48d8-l9xgb	cora-qa	🟩

For update configuration you need to change variable or parameters in the values.yaml file, and push the same to GIT repository and running pipeline.

10. For new version create new branch.