Argo CD applications

Creating and deploying applications on the OpenShift cluster by using the Argo CD dashboard or oc tool
Creating and deploying applications on the OpenShift cluster by using the Argo CD dashboard or `oc` tool
Abstract

This document provides instructions for creating and deploying your applications to the OpenShift cluster either by using the Argo CD dashboard or by using the oc tool. It also discusses how to verify the self-healing behavior in Argo CD.
# Table of Contents

CHAPTER 1. DEPLOYING A SPRING BOOT APPLICATION WITH ARGO CD

1.1. PREREQUISITES ........................................ 3

1.2. CREATING AN APPLICATION BY USING THE ARGO CD DASHBOARD ... 3

1.3. CREATING AN APPLICATION BY USING THE OC TOOL .................... 4

1.4. VERIFYING ARGO CD SELF-HEALING BEHAVIOR ....................... 4
CHAPTER 1. DEPLOYING A SPRING BOOT APPLICATION WITH ARGO CD

With Argo CD, you can deploy your applications to the OpenShift cluster either by using the Argo CD dashboard or by using the `oc` tool.

1.1. PREREQUISITES

- Red Hat OpenShift GitOps is installed in your cluster.
- Logged into Argo CD instance.

1.2. CREATING AN APPLICATION BY USING THE ARGO CD DASHBOARD

Argo CD provides a dashboard which allows you to create applications.

Procedure

1. In the Argo CD dashboard, click NEW APP to add a new Argo CD application.

2. For this workflow, create a spring-petclinic application with the following configurations:

   - **Application Name**: spring-petclinic
   - **Project**: default
   - **Sync Policy**: Automatic
   - **Revision**: HEAD
   - **Path**: app
   - **Destination**: https://kubernetes.default.svc
   - **Namespace**: spring-petclinic

3. Click CREATE to create your application.

4. Open the Administrator perspective of the web console and navigate to Administration → Namespaces in the menu on the left.

5. Search for and select the namespace, then enter argocd.argoproj.io/managed-by=openshift-gitops in the Label field so that the Argo CD instance in the openshift-gitops namespace can manage your namespace.
1.3. CREATING AN APPLICATION BY USING THE `oc` TOOL

You can create Argo CD applications in your terminal by using the `oc` tool.

Procedure

1. Download the sample application:

   ```
   $ git clone git@github.com:redhat-developer/openshift-gitops-getting-started.git
   ```

2. Create the application:

   ```
   $ oc create -f openshift-gitops-getting-started/argo/app.yaml
   ```

3. Run the `oc get` command to review the created application:

   ```
   $ oc get application -n openshift-gitops
   ```

4. Add a label to the namespace your application is deployed in so that the Argo CD instance in the `openshift-gitops` namespace can manage it:

   ```
   $ oc label namespace spring-petclinic argocd.argoproj.io/managed-by=openshift-gitops
   ```

1.4. VERIFYING ARGO CD SELF-HEALING BEHAVIOR

Argo CD constantly monitors the state of deployed applications, detects differences between the specified manifests in Git and live changes in the cluster, and then automatically corrects them. This behavior is referred to as self-healing.

You can test and observe the self-healing behavior in Argo CD.

Prerequisites

- The sample `app-spring-petclinic` application is deployed and configured.

Procedure

1. In the Argo CD dashboard, verify that your application has the `Synced` status.

2. Click the `app-spring-petclinic` tile in the Argo CD dashboard to view the application resources that are deployed to the cluster.

3. In the OpenShift Container Platform web console, navigate to the `Developer` perspective.

4. Modify the Spring PetClinic deployment and commit the changes to the `app/` directory of the Git repository. Argo CD will automatically deploy the changes to the cluster.

   a. Fork the `OpenShift GitOps getting started repository`.

   b. In the `deployment.yaml` file, change the `failureThreshold` value to 5.

   c. In the deployment cluster, run the following command to verify the changed value of the `failureThreshold` field:

      ```
      $
      ```
5. Test the self-healing behavior by modifying the deployment on the cluster and scaling it up to two pods while watching the application in the OpenShift Container Platform web console.
   a. Run the following command to modify the deployment:

```
$ oc edit deployment spring-petclinic -n spring-petclinic
```

b. In the OpenShift Container Platform web console, notice that the deployment scales up to two pods and immediately scales down again to one pod. Argo CD detected a difference from the Git repository and auto-healed the application on the OpenShift Container Platform cluster.

6. In the Argo CD dashboard, click the **app-spring-petclinic** tile → **APP DETAILS** → **EVENTS**. The **EVENTS** tab displays the following events: Argo CD detecting out of sync deployment resources on the cluster and then resyncing the Git repository to correct it.

```
$ oc scale deployment spring-petclinic --replicas 2 -n spring-petclinic
```