



# Red Hat OpenShift GitOps 1.13

## 가시성

관찰 기능을 사용하여 Argo CD 로그를 보고 Argo CD 및 애플리케이션 리소스의 성능 및 상태 모니터링



## Red Hat OpenShift GitOps 1.13 가시성

---

관찰 기능을 사용하여 Argo CD 로그를 보고 Argo CD 및 애플리케이션 리소스의 성능 및 상태 모니터링

## 법적 공지

Copyright © 2024 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, the Red Hat logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux<sup>®</sup> is the registered trademark of Linus Torvalds in the United States and other countries.

Java<sup>®</sup> is a registered trademark of Oracle and/or its affiliates.

XFS<sup>®</sup> is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL<sup>®</sup> is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js<sup>®</sup> is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack<sup>®</sup> Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

## 초록

이 문서에서는 OpenShift GitOps에서 OpenShift Logging을 사용하고 Argo CD 인스턴스, 애플리케이션 상태 및 Argo CD 사용자 정의 리소스 워크로드의 성능을 모니터링하는 방법을 자세히 설명합니다.

---

## 차례

<b>1장. 로깅</b> .....	<b>3</b>
1.1. ARGO CD 로그 보기	3
<b>2장. 모니터링</b> .....	<b>4</b>
2.1. GITOPS 대시보드를 사용한 모니터링	4
2.2. ARGO CD 인스턴스 모니터링	4
2.3. GITOPS OPERATOR 성능 모니터링	8
2.4. 애플리케이션 리소스 및 배포에 대한 상태 정보 모니터링	9
2.5. ARGO CD 사용자 정의 리소스 워크로드 모니터링	11



# 1장. 로깅

## 1.1. ARGO CD 로그 보기

Red Hat OpenShift의 로깅 하위 시스템을 사용하여 Argo CD 로그를 볼 수 있습니다. 로깅 하위 시스템은 Kibana 대시보드에서 로그를 시각화합니다. OpenShift Logging Operator는 기본적으로 Argo CD로 로깅할 수 있습니다.


### 1.1.1. Argo CD 로그 저장 및 검색

Kibana 대시보드를 사용하여 Argo CD 로그를 저장하고 검색할 수 있습니다.

#### 사전 요구 사항

- Red Hat OpenShift GitOps Operator가 클러스터에 설치되어 있습니다.
- Red Hat OpenShift의 로깅 하위 시스템은 클러스터에 기본 구성으로 설치됩니다.

#### 프로세스

1. OpenShift Container Platform 웹 콘솔에서  메뉴 → **Observability** → **Logging** 으로 이동하여 Kibana 대시보드를 확인합니다.
2. 인덱스 패턴을 생성합니다.
  - a. 모든 인덱스를 표시하려면 인덱스 패턴을 \*로 정의하고 **다음 단계**를 클릭합니다.
  - b. **@timestamp for Time Filter** 필드 이름을 선택합니다.
  - c. **인덱스 패턴 생성**을 클릭합니다.
3. Kibana 대시보드의 탐색 패널에서 **Discover** 탭을 클릭합니다.
4. Argo CD의 로그를 검색하는 필터를 생성합니다. 다음 단계에서는 **openshift-gitops** 네임스페이스에 있는 모든 Pod의 로그를 검색하는 필터를 생성합니다.
  - a. **Add a filter +**를 클릭합니다.
  - b. **kubernetes.namespace\_name** 필드를 선택합니다.
  - c. **is operator**를 선택합니다.
  - d. **openshift-gitops** 값을 선택합니다.
  - e. **저장**을 클릭합니다.
5. 선택 사항: 검색 범위를 좁히려면 추가 필터를 추가합니다. 예를 들어 특정 Pod의 로그를 검색하려면 **kubernetes.pod\_name** 을 필드로 사용하여 다른 필터를 생성할 수 있습니다.
6. Kibana 대시보드에서 필터링된 Argo CD 로그를 확인합니다.

### 1.1.2. 추가 리소스

- [웹 콘솔을 사용하여 Red Hat OpenShift의 로깅 하위 시스템 설치](#)

## 2장. 모니터링

### 2.1. GITOPS 대시보드를 사용한 모니터링

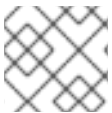
Red Hat OpenShift GitOps 모니터링 대시보드를 사용하여 GitOps 인스턴스의 그래프 뷰에 액세스하여 클러스터 전체의 각 인스턴스의 동작 및 사용을 관찰할 수 있습니다.

3개의 GitOps 대시보드를 사용할 수 있습니다.

- **GitOps 개요:** 애플리케이션, 상태 및 동기화 상태, 애플리케이션 및 동기화 활동 수를 포함하여 클러스터에 설치된 모든 GitOps 인스턴스의 개요를 확인합니다.
- **GitOps 구성 요소:** application-controller, repo-server, server 및 기타 GitOps 구성 요소에 대한 CPU 또는 메모리와 같은 자세한 정보를 봅니다.
- **GitOps gRPC 서비스:** Red Hat OpenShift GitOps의 다양한 구성 요소 간에 gRPC 서비스 활동과 관련된 메트릭을 확인합니다.

#### 2.1.1. GitOps 모니터링 대시보드 액세스

모니터링 대시보드는 Operator에 의해 자동으로 배포됩니다. OpenShift Container Platform 웹 콘솔의 관리자 화면에서 GitOps 모니터링 대시보드에 액세스할 수 있습니다.



#### 참고

대시보드 콘텐츠를 비활성화하거나 변경하는 것은 지원되지 않습니다.

#### 사전 요구 사항

- OpenShift Container Platform 웹 콘솔에 액세스할 수 있습니다.
- Red Hat OpenShift GitOps Operator는 기본 네임스페이스 **openshift-gitops-operator**에 설치됩니다.
- **openshift-gitops-operator** 네임스페이스에서 클러스터 모니터링이 활성화됩니다.
- 정의된 네임스페이스에 Argo CD 애플리케이션을 설치했습니다(예: **openshift-gitops**).

#### 프로세스

1. 웹 콘솔의 관리자 화면에서 **모니터링** → **대시보드**로 이동합니다.
2. 대시보드 드롭다운 목록에서 원하는 GitOps 대시보드: **GitOps(Overview)**, **GitOps / Components** 또는 **GitOps / gRPC Services**를 선택합니다.
3. 선택 사항: 네임스페이스, 클러스터 및 간격 드롭다운 목록에서 특정 **네임스페이스**, **클러스터** 및 **간격**을 선택합니다.
4. GitOps 대시보드에서 원하는 GitOps 메트릭을 확인합니다.

### 2.2. ARGO CD 인스턴스 모니터링



기본적으로 Red Hat OpenShift GitOps Operator는 정의된 네임스페이스에 설치된 Argo CD 인스턴스(예: **openshift-gitops**)를 자동으로 감지하고 이를 클러스터의 모니터링 스택에 연결하여 동기화 외 애플리케이션에 대한 경고를 제공합니다.

### 2.2.1. 사전 요구 사항

- **cluster-admin** 권한이 있는 클러스터에 액세스할 수 있습니다.
- OpenShift Container Platform 웹 콘솔에 액세스할 수 있습니다.
- 클러스터에 Red Hat OpenShift GitOps Operator가 설치되어 있습니다.
- 정의된 네임스페이스에 Argo CD 애플리케이션을 설치했습니다(예: **openshift-gitops**).

### 2.2.2. Prometheus 지표를 사용하여 Argo CD 상태 모니터링

Prometheus 지표 쿼리를 실행하여 Argo CD 애플리케이션의 상태를 모니터링할 수 있습니다.

#### 프로세스

1. 웹 콘솔의 **개발자** 화면에서 Argo CD 애플리케이션이 설치된 네임스페이스를 선택하고 **모니터링** → **메트릭**으로 이동합니다.
2. 쿼리 선택 드롭다운 목록에서 **사용자 지정 쿼리**를 선택합니다.
3. Argo CD 애플리케이션의 상태를 확인하려면 **Expression** 필드에 다음 예제와 유사한 Prometheus Query Language(PromQL) 쿼리를 입력합니다.

예

```
sum(argocd_app_info{dest_namespace=~"<your_defined_namespace>",health_status!=""})
by (health_status) 1
```

- 1 < **your\_defined\_namespace** > 변수를 정의된 네임스페이스의 실제 이름(예: **openshift-gitops**)으로 바꿉니다.

### 2.2.3. Argo CD 인스턴스의 메트릭 자동 스크랩 비활성화

기본적으로 Red Hat OpenShift GitOps Operator는 모든 Argo CD 인스턴스에 대한 메트릭을 자동으로 스크랩하여 성능을 측정합니다. 결과적으로 Operator는 Argo CD 인스턴스가 설치된 네임스페이스에 다음 리소스 및 레이블을 생성합니다.

- **GitOps-operator-argocd-alerts** prometheus 규칙
- **<argocd\_namespace>-read** role
- **< argocd\_name >, < argocd\_name>-repo-server, < argocd\_name>-server** 서비스 모니터
- **<argocd\_namespace>-prometheus-k8s-read-binding** role binding
- **openshift.io/cluster-monitoring=true** label

클러스터에서 여러 Argo CD 인스턴스에 대한 메트릭을 스크랩하면 스토리지 사용량이 과도해질 수 있습니다. 예방 조치로 웹 콘솔의 YAML 보기를 사용하고 Argo CD 인스턴스의 메트릭 자동 스크랩을 비활성화하도록 **ArgoCD CR**(사용자 정의 리소스)을 구성합니다.

클러스터 관리자는 개별 인스턴스에 대한 메트릭 스크랩을 비활성화하여 사용자에게 정의된 네임스페이스를 관리할 수 있는 더 나은 제어, 유연성 및 안정성을 제공할 수 있습니다.

### 프로세스

1. OpenShift Container Platform 웹 콘솔에 로그인합니다.
2. 웹 콘솔의 관리자 화면에서 **Operator** → 설치된 **Operator** 를 클릭합니다.
3. 프로젝트 목록에서 사용자 정의 Argo CD 인스턴스가 설치된 프로젝트를 선택합니다.
4. 설치된 Operator 목록에서 **Red Hat OpenShift GitOps** 를 선택하고 **Argo CD** 탭으로 이동합니다.
5. 사용자 정의 Argo CD 인스턴스를 클릭합니다.
6. 메트릭의 자동 스크랩을 비활성화하도록 사용자 정의 Argo CD 인스턴스의 **ArgoCD CR**을 구성합니다.
  - a. **YAML** 탭을 클릭하고 **ArgoCD CR**의 YAML 파일을 편집합니다.
  - b. **ArgoCD CR**에서 **spec.monitoring.disableMetrics** 필드 값을 **true** 로 설정합니다.

### ArgoCD CR의 예

```
apiVersion: argoproj.io/v1beta1
kind: ArgoCD
metadata:
  name: example 1
  namespace: spring-petclinic 2
spec:
  monitoring:
    disableMetrics: true
```

- 1 사용자 정의 Argo CD 인스턴스의 이름입니다.
- 2 사용자 정의 Argo CD 인스턴스를 실행하려는 네임스페이스입니다.

## 작은 정보

또는 다음 명령을 사용하여 Red Hat OpenShift GitOps **argocd** CLI에서 메트릭 자동 스크랩을 비활성화합니다.

## 명령 예

```
$ oc patch argocd example -n spring-petclinic --type='json' -p='[{"op": "replace", "path": "/spec/monitoring/disableMetrics", "value": true}]'
```

## 출력 예

```
argocd.argoproj.io/example patched
```

7. Operator가 정의된 네임스페이스에 **openshift.io/cluster-monitoring=false** 라벨을 추가하는지 확인합니다.
  - a. 관리 → 네임스페이스로 이동합니다.  
네임스페이스 페이지에 생성된 네임스페이스가 표시됩니다.
  - b. 정의된 네임스페이스를 클릭하고 YAML 탭으로 이동하여 **metadata.labels** 섹션 아래에 Operator가 **openshift.io/cluster-monitoring=false** 레이블이 추가되었는지 확인합니다.
8. Operator가 정의된 네임스페이스에서 다음 리소스를 삭제하는지 확인합니다.
  - a. 홈 → 검색으로 이동합니다.
  - b. 리소스 목록에서 **PrometheusRule, Role, RoleBinding, ServiceMonitors** 를 선택합니다.  
검색 페이지에는 선택한 리소스가 표시됩니다.
  - c. 검색 페이지에서 **PrometheusRule** 섹션에서 **gitops-operator-argocd-alerts** prometheus 규칙이 제거되었는지 확인합니다.
  - d. **Roles** 섹션의 필터 목록에서 네임스페이스 역할을 선택합니다.
  - e. < **argocd\_namespace** >-**read** 역할이 제거되었는지 확인합니다.
  - f. **RoleBindings** 섹션의 필터 목록에서 **Namespace RoleBindings** 를 선택합니다.
  - g. < **argocd\_namespace** >-**prometheus-k8s-read-binding** 역할 바인딩 이 제거되었는지 확인합니다.
  - h. **ServiceMonitors** 섹션에서 < **argocd\_name** >, < **argocd\_name** >-**repo-server**, < **argocd\_name** >-**server** 서비스 모니터가 제거되었는지 확인합니다.



## 참고

**spec.monitoring.disableMetrics** 필드 값을 **false** 로 수정하여 인스턴스에 대한 메트릭을 활성화할 수 있습니다. 그러면 Operator에서 필요한 역할, 역할 바인딩 및 서비스를 모니터링하고 **openshift.io/cluster-monitoring=true** 레이블을 정의된 네임스페이스에 추가합니다.

### 2.2.4. 추가 리소스

- [Argo CD 사용자 정의 리소스 워크로드 모니터링](#)

## 2.3. GITOPS OPERATOR 성능 모니터링

Red Hat OpenShift GitOps Operator는 성능에 대한 지표를 내보냅니다. 이러한 지표를 선택하는 OpenShift 모니터링 스택을 사용하면 Operator의 성능을 모니터링하고 분석할 수 있습니다. Operator는 OpenShift Container Platform 웹 콘솔을 사용하여 볼 수 있는 다음 메트릭을 노출합니다.

표 2.1. GitOps Operator 성능 지표

메트릭 이름	유형	설명
<b>active_argocd_instances_total</b>	게이지	지정된 시간에 클러스터 전체에서 현재 Operator에서 관리하는 총 활성 Argo CD 인스턴스 수입니다.
<b>active_argocd_instances_by_phase</b>	게이지	지정된 단계에서 활성 Argo CD 인스턴스 수(예: pending 또는 available)입니다.
<b>active_argocd_instance_reconciliation_count</b>	카운터	지정된 시간에 지정된 네임스페이스의 인스턴스에 대해 발생한 총 조정 수입니다.
<b>controller_runtime_reconcile_time_seconds_per_instance_bucket</b>	카운터	인스턴스의 지정된 기간 내에 완료된 조정 주기 수입니다. 예를 들어 <b>controller_runtime_reconcile_time_seconds_per_instance_bucket{le="0.5"}</b> 은 지정된 인스턴스에 대해 완료하는 데 0.5 초 미만의 조정 수를 보여줍니다.
<b>controller_runtime_reconcile_time_seconds_per_instance_count</b>	카운터	지정된 인스턴스에 대해 관찰된 총 조정 주기 수입니다.
<b>controller_runtime_reconcile_time_seconds_per_instance_sum</b>	카운터	지정된 인스턴스에 대해 관찰된 조정에 걸린 총 시간입니다.



### 참고

게이지는 위 또는 아래로 갈 수 있는 값입니다. 카운터는 상승할 수 있는 값입니다.

### 2.3.1. GitOps Operator 지표 액세스

OpenShift Container Platform 웹 콘솔의 관리자 화면에서 Operator 지표에 액세스하여 Operator의 성능을 추적할 수 있습니다.

#### 사전 요구 사항

- OpenShift Container Platform 웹 콘솔에 액세스할 수 있습니다.

- Red Hat OpenShift GitOps Operator는 기본 **openshift-gitops-operator** 네임스페이스에 설치됩니다.
- **openshift-gitops-operator** 네임스페이스에서 클러스터 모니터링이 활성화됩니다.

### 프로세스

1. 웹 콘솔의 관리자 화면에서 **모니터링** → **메트릭** 으로 이동합니다.
2. **표현식 필드**에 지표를 입력합니다. 다음 메트릭에서 선택할 수 있습니다.
  - **active\_argocd\_instances\_total**
  - **active\_argocd\_instances\_by\_phase**
  - **active\_argocd\_instance\_reconciliation\_count**
  - **controller\_runtime\_reconcile\_time\_seconds\_per\_instance\_bucket**
  - **controller\_runtime\_reconcile\_time\_seconds\_per\_instance\_count**
  - **controller\_runtime\_reconcile\_time\_seconds\_per\_instance\_sum**
3. (선택 사항): 해당 속성으로 지표를 필터링합니다. 예를 들어 **Available** 단계에 따라 **active\_argocd\_instances\_by\_phase** 메트릭을 필터링합니다.

### 예

```
active_argocd_instances_by_phase{phase="Available"}
```

4. (선택 사항): **쿼리 추가** 를 클릭하여 여러 쿼리를 입력합니다.
5. **쿼리 실행**을 클릭하여 GitOps Operator 지표를 활성화하고 관찰합니다.

## 2.3.2. 추가 리소스

- [웹 콘솔에서 Red Hat OpenShift GitOps Operator 설치](#)

## 2.4. 애플리케이션 리소스 및 배포에 대한 상태 정보 모니터링

OpenShift Container Platform 웹 콘솔의 **개발자** 화면에 있는 Red Hat OpenShift GitOps **환경** 페이지에는 각 배포의 버전 링크와 함께 애플리케이션 환경의 성공적인 배포 목록이 표시됩니다.

OpenShift Container Platform 웹 콘솔의 **개발자** 화면에 있는 **애플리케이션 환경** 페이지에는 경로, 동기화 상태, 배포 구성 및 배포 기록과 같은 애플리케이션 리소스의 상태가 표시됩니다.

OpenShift Container Platform 웹 콘솔의 **개발자** 화면에 있는 환경 페이지는 Red Hat OpenShift GitOps Application Manager CLI(명령줄 인터페이스), **kam** 과 분리됩니다. **kam** 을 사용하여 OpenShift Container Platform 웹 콘솔의 **개발자** 화면에 표시하기 위해 환경에 대한 애플리케이션 환경 매니페스트를 생성할 필요가 없습니다. 고유한 매니페스트를 사용할 수 있지만 환경을 계속 네임스페이스로 표시해야 합니다. 또한 특정 레이블과 주석이 계속 필요합니다.

### 2.4.1. 환경 레이블 및 주석 설정

이 섹션에서는 Red Hat OpenShift GitOps 웹 콘솔의 **개발자** 화면에서 환경 애플리케이션을 **환경** 페이지에 표시하는 데 필요한 환경 레이블 및 주석에 대한 참조 설정을 제공합니다.

### 환경 레이블

환경 애플리케이션 매니페스트에는 **labels.openshift.gitops/environment** 및 **destination.namespace** 필드가 포함되어야 합니다. < **environment\_name** > 변수와 환경 애플리케이션 매니페스트 이름에 대해 동일한 값을 설정해야 합니다.

### 환경 애플리케이션 매니페스트의 사양

```
spec:
  labels:
    openshift.gitops/environment: <environment_name>
  destination:
    namespace: <environment_name>
# ...
```

### 환경 애플리케이션 매니페스트의 예

```
apiVersion: argoproj.io/v1beta1
kind: Application
metadata:
  name: dev-env 1
  namespace: openshift-gitops
spec:
  labels:
    openshift.gitops/environment: dev-env
  destination:
    namespace: dev-env
# ...
```

**1** 환경 애플리케이션 매니페스트의 이름입니다. 값 세트는 < **environment\_name** > 변수의 값과 동일합니다.

### 환경 주석

애플리케이션의 버전 컨트롤러 코드 소스를 지정하려면 환경 네임스페이스 매니페스트에 **annotations.app.openshift.io/vcs-uri** 및 **annotations.app.openshift.io/vcs-ref** 필드가 포함되어야 합니다. < **environment\_name** > 변수와 환경 네임스페이스 매니페스트 이름에 대해 동일한 값을 설정해야 합니다.

### 환경 네임스페이스 매니페스트 사양

```
apiVersion: v1
kind: Namespace
metadata:
  annotations:
    app.openshift.io/vcs-uri: <application_source_url>
    app.openshift.io/vcs-ref: <branch_reference>
  name: <environment_name> 1
# ...
```

**1** 환경 네임스페이스 매니페스트의 이름입니다. 값 세트는 < **environment\_name** > 변수의 값과 동일합니다.

## 환경 네임스페이스 매니페스트의 예

```

apiVersion: v1
kind: Namespace
metadata:
  annotations:
    app.openshift.io/vcs-uri: https://example.com/<your_domain>/<your_gitops.git>
    app.openshift.io/vcs-ref: main
  labels:
    argocd.argoproj.io/managed-by: openshift-gitops
  name: dev-env
# ...

```

### 2.4.2. 상태 정보 확인

Red Hat OpenShift GitOps Operator는 **openshift-gitops** 네임스페이스에 GitOps 백엔드 서비스를 설치합니다.

#### 사전 요구 사항

- Red Hat OpenShift GitOps Operator는 **OperatorHub** 에서 설치됩니다.
- 애플리케이션이 Argo CD에서 동기화되었는지 확인합니다.

#### 프로세스

1. 개발자 화면에서 **환경**을 클릭합니다. **환경** 페이지에는 애플리케이션 목록과 해당 **환경 상태**가 표시됩니다.
2. **환경 상태** 열 아래에 있는 아이콘 위로 마우스 커서를 이동하여 모든 환경의 동기화 상태를 확인합니다.
3. 목록에서 애플리케이션 이름을 클릭하여 특정 애플리케이션의 세부 정보를 확인합니다.
4. **애플리케이션 환경** 페이지의 **개요** 탭의 **리소스** 섹션에 아이콘이 표시되면 아이콘을 마우스로 이동하여 상태 세부 정보를 가져옵니다.
  - 손상된 하트로 인해 리소스 문제가 애플리케이션 성능이 저하되었음을 나타냅니다.
  - 노란색 예보 기호는 리소스 문제가 애플리케이션 상태에 대한 데이터를 지연했음을 나타냅니다.
5. 애플리케이션의 배포 내역을 보려면 **Deployment History** 탭을 클릭합니다. 페이지에는 **마지막 배포, 설명** (커밋 메시지), **환경, 작성자**, 버전 등의 세부 정보가 포함되어 있습니다.

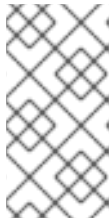
## 2.5. ARGO CD 사용자 정의 리소스 워크로드 모니터링

Red Hat OpenShift GitOps를 사용하면 특정 Argo CD 인스턴스에 대한 Argo CD 사용자 정의 리소스 워크로드의 가용성을 모니터링할 수 있습니다. Argo CD 사용자 정의 리소스 워크로드를 모니터링하면 경고를 활성화하여 Argo CD 인스턴스의 상태에 대한 최신 정보가 있습니다. 해당 Argo CD 인스턴스의 application-controller, repo-server 또는 서버와 같은 구성 요소 워크로드 Pod는 특정 이유로 발생할 수 없으며 준비된 복제본 수와 일정 기간 동안 원하는 복제본 수 사이에 드리프트가 있는 경우 Operator는 경고를 트리거합니다.

Argo CD 사용자 정의 리소스 워크로드 모니터링에 대한 설정을 활성화하고 비활성화할 수 있습니다.

### 2.5.1. 사전 요구 사항

- **cluster-admin** 역할의 사용자로 클러스터에 액세스할 수 있어야 합니다.
- Red Hat OpenShift GitOps가 클러스터에 설치되어 있습니다.
- 모니터링 스택은 **openshift-monitoring** 프로젝트의 클러스터에 구성되어 있습니다. 또한 Argo CD 인스턴스는 Prometheus를 통해 모니터링할 수 있는 네임스페이스에 있습니다.
- **kube-state-metrics** 서비스가 클러스터에서 실행 중입니다.
- 선택 사항: 사용자 정의 프로젝트에 이미 있는 Argo CD 인스턴스에 대한 모니터링을 활성화하는 경우 클러스터의 **사용자 정의 프로젝트에 대한 모니터링이 활성화되어** 있는지 확인합니다.



#### 참고

기본 **openshift-monitoring** 스택에서 조사하지 않는 네임스페이스에서 Argo CD 인스턴스에 대한 모니터링을 활성화하려면(예: **openshift-\*** 로 시작하지 않는 네임스페이스) 클러스터에서 사용자 워크로드 모니터링을 활성화해야 합니다. 이 작업을 사용하면 모니터링 스택에서 생성된 PrometheusRule을 선택할 수 있습니다.

### 2.5.2. Argo CD 사용자 정의 리소스 워크로드에 대한 모니터링 활성화

기본적으로 Argo CD 사용자 정의 리소스 워크로드에 대한 모니터링 구성은 **false** 로 설정됩니다.

Red Hat OpenShift GitOps를 사용하면 특정 Argo CD 인스턴스에 대한 워크로드 모니터링을 활성화할 수 있습니다. 결과적으로 Operator는 특정 Argo CD 인스턴스에서 관리하는 모든 워크로드에 대한 경고 규칙이 포함된 **PrometheusRule** 오브젝트를 생성합니다. 이러한 경고 규칙은 해당 구성 요소의 복제본 수가 일정 시간 동안 원하는 상태에서 드리프트될 때 경고 실행을 트리거합니다. Operator는 사용자가 **PrometheusRule** 오브젝트에 대한 변경 사항을 덮어쓰지 않습니다.

#### 프로세스

1. 지정된 Argo CD 인스턴스에서 **.spec.monitoring.enabled** 필드 값을 **true** 로 설정합니다.

#### Argo CD 사용자 정의 리소스의 예

```
apiVersion: argoproj.io/v1beta1
kind: ArgoCD
metadata:
  name: example-argocd
  labels:
    example: repo
spec:
  # ...
  monitoring:
    enabled: true
  # ...
```

2. Operator에서 생성한 PrometheusRule에 경고 규칙이 포함되어 있는지 확인합니다.

#### 경고 규칙 예

```
apiVersion: monitoring.coreos.com/v1
kind: PrometheusRule
```



```

metadata:
  name: argocd-component-status-alert
  namespace: openshift-gitops
spec:
  groups:
  - name: ArgoCDComponentStatus
    rules:
      # ...
      - alert: ApplicationSetControllerNotReady 1
        annotations:
          message: >-
            applicationSet controller deployment for Argo CD instance in
            namespace "default" is not running
        expr: >-
          kube_statefulset_status_replicas{statefulset="openshift-gitops-application-controller
statefulset",
          namespace="openshift-gitops"} !=
          kube_statefulset_status_replicas_ready{statefulset="openshift-gitops-application-
controller statefulset",
          namespace="openshift-gitops"}
        for: 1m
        labels:
          severity: critical

```

- 1 Argo CD 인스턴스에서 생성한 워크로드가 예상대로 실행되는지 확인하는 PrometheusRule의 경고 규칙입니다.

### 2.5.3. Argo CD 사용자 정의 리소스 워크로드에 대한 모니터링 비활성화

특정 Argo CD 인스턴스에 대한 워크로드 모니터링을 비활성화할 수 있습니다. 워크로드 모니터링을 비활성화하면 생성된 PrometheusRule가 삭제됩니다.

#### 프로세스

- 지정된 Argo CD 인스턴스에서 **.spec.monitoring.enabled** 필드 값을 **false** 로 설정합니다.

#### Argo CD 사용자 정의 리소스의 예

```

apiVersion: argoproj.io/v1beta1
kind: ArgoCD
metadata:
  name: example-argocd
  labels:
    example: repo
spec:
  # ...
  monitoring:
    enabled: false
  # ...

```

### 2.5.4. 추가 리소스

- [사용자 정의 프로젝트 모니터링 활성화](#)

