



# OpenShift Dedicated 4

## 애플리케이션

애플리케이션에 대한 OpenShift Dedicated 구성



# OpenShift Dedicated 4 애플리케이션

---

애플리케이션에 대한 OpenShift Dedicated 구성

## 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 애플리케이션 배포에 OpenShift Dedicated를 구성하는 방법에 대한 정보를 제공합니다. 여기에는 사용자 정의 와일드카드 도메인 설정이 포함됩니다.

## 차례

1장. 배포 .....	3
1.1. 애플리케이션용 사용자 정의 도메인	3



## 1장. 배포

### 1.1. 애플리케이션용 사용자 정의 도메인



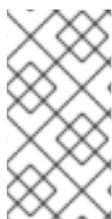
#### 참고

OpenShift Dedicated 4.14부터 Custom Domain Operator는 더 이상 사용되지 않습니다. OpenShift Dedicated 4.14에서 Ingress를 관리하려면 Ingress Operator를 사용합니다. OpenShift Dedicated 4.13 및 이전 버전에서는 기능이 변경되지 않습니다.

애플리케이션에 대한 사용자 정의 도메인을 구성할 수 있습니다. 사용자 정의 도메인은 OpenShift Dedicated 애플리케이션에서 사용할 수 있는 특정 와일드카드 도메인입니다.

#### 1.1.1. 애플리케이션의 사용자 정의 도메인 구성

최상위 도메인(ScanSetting)은 OpenShift Dedicated 클러스터를 운영하는 고객이 소유합니다. Custom Domains Operator는 사용자 정의 인증서를 사용하여 2일 차 작업으로 새 수신 컨트롤러를 설정합니다. 그런 다음 이 Ingress 컨트롤러의 퍼블릭 DNS 레코드를 사용하여 사용자 정의 도메인에 사용할 와일드카드 CNAME 레코드를 만들 수 있습니다.



#### 참고

Red Hat은 API 도메인을 제어하므로 사용자 정의 API 도메인이 지원되지 않습니다. 그러나 고객은 애플리케이션 도메인을 변경할 수 있습니다. 비공개 **IngressController**가 있는 프라이빗 사용자 정의 도메인의 경우 **CustomDomain** CR에서 **.spec.scope**를 **Internal**로 설정합니다.

#### 사전 요구 사항

- **dedicated-admin** 권한이 있는 사용자 계정
- 고유한 도메인 또는 와일드카드 도메인(예: **\*.apps.<company\_name>.io**)
- **CN=\*.apps.<company\_name>.io**와 같은 사용자 정의 인증서 또는 와일드카드 사용자 정의 인증서
- 최신 버전의 **oc** CLI가 설치된 클러스터에 액세스



#### 중요

**CustomDomain** CR의 **metadata/name:** 섹션에서 예약된 이름 기본값 또는 **apps \***(예: **apps** 또는 **apps2**)를 사용하지 마십시오.

#### 절차

1. 개인 키 및 공용 인증서에서 새 TLS 시크릿을 생성합니다. 여기서 **fullchain.pem** 및 **privkey.pem**은 공개 또는 개인 와일드카드 인증서입니다.

#### 예제

```
$ oc create secret tls <name>-tls --cert=fullchain.pem --key=privkey.pem -n <my_project>
```

2. 새 **CustomDomain** CR(사용자 정의 리소스)을 생성합니다.

예: **<company\_name>-custom-domain.yaml**

```
apiVersion: managed.openshift.io/v1alpha1
kind: CustomDomain
metadata:
  name: <company_name>
spec:
  domain: apps.<company_name>.io ❶
  scope: External
  loadBalancerType: Classic ❷
  certificate:
    name: <name>-tls ❸
    namespace: <my_project>
  routeSelector: ❹
    matchLabels:
      route: acme
  namespaceSelector: ❺
    matchLabels:
      type: sharded
```

- ❶ 사용자 정의 도메인입니다.
- ❷ 사용자 정의 도메인의 로드 밸런서 유형입니다. 이 유형은 네트워크 로드 밸런서를 사용하는 경우 기본 클래식 또는 **NLB** 일 수 있습니다.
- ❸ 이전 단계에서 생성한 시크릿입니다.
- ❹ 선택 사항: CustomDomain ingress에서 서비스를 제공하는 경로 집합을 필터링합니다. 값을 제공하지 않으면 기본값은 필터링되지 않습니다.
- ❺ 선택 사항: CustomDomain Ingress에서 서비스를 제공하는 네임스페이스 집합을 필터링합니다. 값을 제공하지 않으면 기본값은 필터링되지 않습니다.

3. CR을 적용합니다.

예제

```
$ oc apply -f <company_name>-custom-domain.yaml
```

4. 새로 생성된 CR의 상태를 가져옵니다.

```
$ oc get customdomains
```

출력 예

NAME	ENDPOINT	DOMAIN	STATUS
<company_name>	xxrywp.<company_name>.cluster-01.opln.s1.openshiftapps.com		
*.apps.<company_name>.io	Ready		

5. 엔드포인트 값을 사용하여 관리형 DNS 공급자(예: Route53, Azure DNS 또는 Google DNS)에 새 와일드카드 CNAME 레코드 세트를 추가합니다.

## 예제

```
*.apps.<company_name>.io -> xxrywp.<company_name>.cluster-01.opln.s1.openshiftapps.com
```

6. 새 애플리케이션을 생성하고 노출합니다.

## 예제

```
$ oc new-app --docker-image=docker.io/openshift/hello-openshift -n my-project
```

```
$ oc create route <route_name> --service=hello-openshift hello-openshift-tls --hostname hello-openshift-tls-my-project.apps.<company_name>.io -n my-project
```

```
$ oc get route -n my-project
```

```
$ curl https://hello-openshift-tls-my-project.apps.<company_name>.io
Hello OpenShift!
```

## 문제 해결

- [TLS 시크릿을 생성하는 동안 오류 발생](#)
- [문제 해결: NotReady 상태의 CustomDomain](#)

### 1.1.2. 사용자 정의 도메인의 인증서 갱신

**oc** CLI 툴을 사용하여 Custom Domains Operator(CDO)로 인증서를 갱신할 수 있습니다.

#### 사전 요구 사항

- 최신 버전의 **oc** CLI 툴이 설치되어 있습니다.

#### 절차

1. 새 보안 생성

```
$ oc create secret tls <secret-new> --cert=fullchain.pem --key=privkey.pem -n <my_project>
```

2. Patch CustomDomain CR

```
$ oc patch customdomain <company_name> --type='merge' -p '{"spec":{"certificate":{"name":"<secret-new>"}}}'
```

3. 이전 보안 삭제

```
$ oc delete secret <secret-old> -n <my_project>
```

## 문제 해결

- [TLS 시크릿을 생성하는 동안 오류 발생](#)

