



Red Hat OpenShift Service on AWS 4

应用程序开发

为您的应用程序配置 Red Hat OpenShift Service on AWS

Red Hat OpenShift Service on AWS 4 应用程序开发

为您的应用程序配置 Red Hat OpenShift Service on AWS

法律通告

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[®] is the registered trademark of Linus Torvalds in the United States and other countries.

Java[®] is a registered trademark of Oracle and/or its affiliates.

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

MySQL[®] is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js[®] 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[®] 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.

摘要

本文档提供有关为应用程序开发配置 Red Hat OpenShift Service on AWS (ROSA) 的信息。这包括设置自定义通配符域。

目录

| | |
|---------------|---|
| 第1章 部署 | 3 |
| 1.1. 应用程序自定义域 | 3 |

第 1 章 部署

1.1. 应用程序自定义域



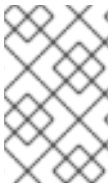
注意

从 Red Hat OpenShift Service on AWS 4.14 开始，自定义域 Operator 已被弃用。要在 AWS 4.14 上的 Red Hat OpenShift Service 中管理 Ingress，请使用 Ingress Operator。对于 Red Hat OpenShift Service on AWS 4.13 及更早的版本，这个功能不会改变。

您可以为应用程序配置自定义域。自定义域是可用于 AWS 应用程序上的 Red Hat OpenShift Service 的特定通配符域。

1.1.1. 为应用程序配置自定义域

顶级域 (TLD) 由操作 Red Hat OpenShift Service on AWS 的用户所有。Custom Domains Operator 使用自定义证书设置一个新的入口控制器作为第二天操作。然后，外部 DNS 可以使用这个 ingresscontroller 的公共 DNS 记录来创建用于自定义域的通配符 CNAME 记录。



注意

不支持自定义 API 域，因为 API 域由红帽控制。但是，客户可以更改自己的应用程序域。对于带有私有 **IngressController** 的专用自定义域，在 **CustomDomain** CR 中将 **.spec.scope** 设置为 **Internal**。

先决条件

- 具有 **dedicated-admin** 特权的用户帐户
- 唯一的域或通配符域，如 ***.apps.<company_name>.io**
- 一个自定义证书或通配符自定义证书，如 **CN=*.apps.<company_name>.io**
- 访问安装最新版本的 **oc** CLI 的集群



重要

不要在 **CustomDomain** CR 的 **metadata/name:** 部分中使用保留的名称 **default** 或 **apps*** (如 **apps** 或 **apps2**)。

流程

1. 从私钥和公共证书创建一个新的 TLS secret，其中 **fullchain.pem** 和 **privkey.pem** 是您的公共或私有通配符证书。

示例

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

2. 创建新的 **CustomDomain** 自定义资源(CR)：

示例 **<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 自定义域。
- ❷ 自定义域的负载均衡器的类型。如果您使用网络负载均衡器，此类型可以是默认的 **classic** 或 **NLB**。
- ❸ 上一步中创建的 secret。
- ❹ 可选：过滤由 CustomDomain ingress 提供服务的 Routes 组。如果没有提供值，则默认为没有过滤。
- ❺ 可选：过滤由 CustomDomain ingress 提供服务的 Namespaces 组。如果没有提供值，则默认为没有过滤。

3. 应用 CR：

示例

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

4. 获取新创建的 CR 的状态：

```
$ oc get customdomains
```

输出示例

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

5. 使用端点值，向受管 DNS 供应商（如 Route53）添加新的通配符 CNAME 记录。

示例


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

6. 创建新应用并公开它：

示例

```
$ 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!
```

故障排除

- [创建 TLS secret 时出错](#)
- [故障排除：CustomDomain 处于 NotReady 状态](#)

1.1.2. 为自定义域续订证书

您可以使用 **oc** CLI 工具使用 Custom Domains Operator (CDO) 续订证书。

先决条件

- 已安装最新版本的 **oc** CLI 工具。

流程

1. 创建新 secret

```
$ 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. 删除旧的 secret

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

故障排除

- [创建 TLS secret 时出错](#)

