



Red Hat AMQ 2021.Q2

使用 OpenShift 创建服务网络

用于 AMQ Interconnect 2.0 TECHNOLOGY PREVIEW

Red Hat AMQ 2021.Q2 使用 OpenShift 创建服务网络

用于 AMQ Interconnect 2.0 TECHNOLOGY PREVIEW

Enter your first name here. Enter your surname here.

Enter your organisation's name here. Enter your organisational division here.

Enter your email address here.

法律通告

Copyright © 2021 | You need to change the HOLDER entity in the en-US/Creating_a_service_network_with_OpenShift.ent file |.

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

本教程介绍了如何在 OpenShift 中创建 AMQ 互连站点以构建服务网络。

目录

前言	3
第1章 使用 OPENSIFT 创建服务网络	4
1.1. AMQ INTERCONNECT 2.0 介绍	4
1.2. 安装 SKUPPER CLI	4
1.3. 配置终端会话	5
1.4. 在两个集群中安装服务网络路由器	6
1.5. 连接命名空间以创建服务网络	6
1.6. 创建前端服务	7
1.7. 创建后端服务，并使其在服务网络上可用	8
1.8. 中断服务网络	9

前言

使开源包含更多

红帽承诺替换我们的代码、文档和网页属性中存在问题的语言。我们从这四个术语开始：master、slave、blacklist 和 whitelist。这些更改将在即将发行的几个发行本中逐渐实施。如需了解更多详细信息，请参阅 [CTO Chris Wright 信息](#)。



重要

红帽生产服务级别协议(SLA)不支持 AMQ Interconnect 2.0 技术预览功能，且可能无法完成。红帽不推荐在生产环境中使用它们。

这些技术预览功能可以使用户提早试用新的功能，并有机会在开发阶段提供反馈意见。有关红帽技术预览功能支持范围的更多信息，请参阅

<https://access.redhat.com/support/offerings/techpreview>。

第 1 章 使用 OPENSIFT 创建服务网络

本教程演示了如何使用 **skupper** 命令行界面(CLI)将 OpenShift 集群上的前端服务与 OpenShift 集群上的后端服务连接。

先决条件

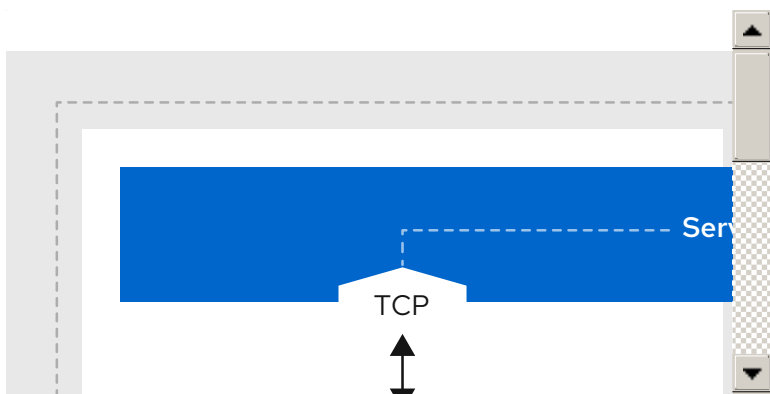
- 不需要访问两个 OpenShift 集群中的项目，即 **cluster-admin** 访问权限。
- 其中一个 OpenShift 集群必须可以从其他集群进行寻址。

本教程演示了如何连接以下命名空间：

- **west** - 运行前端服务，通常是一个公共集群。
- **east** - 运行后端服务。

1.1. AMQ INTERCONNECT 2.0 介绍

互连 2.0 引入了服务网络，跨混合云链接服务。服务网络实现在不同网络位置中运行的服务之间的通信。它允许根据地域分布的服务进行连接，如同这些服务在同一站点中运行一样。



例如，您可以在公共 OpenShift 集群中部署前端，并在私有 OpenShift 集群中部署后端，然后将它们连接到服务网络中。

服务网络提供以下功能：

- 默认安全性.使用私有的专用证书颁发机构(CA)，通过 mutual TLS 保护所有站点间的流量。
- 在 OpenShift 集群（甚至私有集群）之间轻松连接。
- 服务网络支持现有的基于 TCP 的应用程序，无需修改。
- 使用服务网络控制台监控分散在多个 OpenShift 集群的应用流量。

您可以使用 **skupper** CLI 部署和管理服务网络。

1.2. 安装 SKUPPER CLI

安装 **skupper** 命令行界面(CLI)提供了一种简单方法开始使用 AMQ Interconnect。

流程

1. 确保您的订阅已激活，您的系统已注册。
2. 订阅所需的软件仓库：

Red Hat Enterprise Linux 7

```
$ sudo subscription-manager repos --enable=interconnect-2-for-rhel-7-server-rpms
```

Red Hat Enterprise Linux 8

```
$ sudo subscription-manager repos --enable=interconnect-2-for-rhel-8-x86_64-rpms
```

3. 使用 **yum** 或 **dnf** 命令安装 **skupper** 软件包：

```
$ sudo yum install skupper
```

4. 验证安装。

```
$ skupper version
client version 0.6.0-redhat-interconnect-2.0.0
```

1.3. 配置终端会话

这个步骤描述了如何配置终端会话来避免在不同集群中配置 AMQ Interconnect 时出现问题。

下表显示了您可以如何设置终端会话。

表 1.1. 终端会话

西终端会话	东部终端会话
<pre>\$ oc project west</pre>	<pre>\$ oc project east</pre>

先决条件

- 已安装 OpenShift CLI。有关如何安装 **oc** 的更多说明，请参阅 [OpenShift CLI 文档](#)。



注意

在 OpenShift 4.6 及更高版本中，您可以使用 Web 终端执行以下步骤，如 [Web 终端 文档](#) 中所述。

流程

1. 启动一个终端会话来处理 **west** 命名空间并设置 **KUBECONFIG** 环境变量：

```
$ export KUBECONFIG=$HOME/.kube/config-west
```

此会话稍后称为 *west* 终端会话。

2. 启动一个终端会话来处理 **east** 命名空间并设置 **KUBECONFIG** 环境变量：

```
$ export KUBECONFIG=$HOME/.kube/config-east
```

此会话稍后称为 *east* 终端会话。

3. 在每个终端会话中，登录到 OpenShift 集群，例如：

```
$ oc login
```

1.4. 在两个集群中安装服务网络路由器

1. 在 *west* 终端会话中：

- a. 创建 **west** 项目（命名空间）：

```
$ oc new-project west
```

- b. 创建服务网络路由器：

```
$ skupper init
```

- c. 检查站点状态：

```
$ skupper status
```

输出结果应类似如下：

```
Skupper enabled for namespace 'west'. It is not connected to any other sites.
```

2. 在 *east* 终端会话中：

- a. 创建 **east** 项目（命名空间）：

```
$ oc new-project east
```

- b. 创建服务网络路由器：

```
$ skupper init
```

- c. 检查站点状态：

```
$ skupper status
```

输出结果应类似如下：

```
Skupper enabled for namespace 'east'. It is not connected to any other sites.
```

1.5. 连接命名空间以创建服务网络

安装服务网络路由器后，您可以将它们安全地连接在一起，并允许服务在服务网络间共享。

流程

1. 在 west 终端会话中，创建一个连接令牌以允许连接到 west 命名空间：

```
$ skupper token create $HOME/secret.yaml
```

这个命令在主目录中创建 **secret.yaml** 文件，您可以使用该文件来创建安全连接。

2. 在 east 终端会话中，使用令牌来创建到 west 命名空间的连接：

```
$ skupper link create $HOME/secret.yaml
```

3. 从 west 终端会话检查站点状态：

```
$ skupper status
```

输出结果应类似如下：

```
Skupper is enabled for namespace "west" in interior mode. It is connected to 1 other site. It
has no exposed services.
The site console url is: https://<skupper-url>
The credentials for internal console-auth mode are held in secret: 'skupper-console-users'
```

1.6. 创建前端服务

frontend 服务是一个简单的 Python 应用，显示来自 backend 应用的消息。

流程

执行 west 终端会话中的所有任务：

1. 部署 frontend 服务：

```
$ oc create deployment hello-world-frontend --image quay.io/skupper/hello-world-frontend
```

2. 将 frontend 部署公开为集群服务：

```
$ oc expose deployment hello-world-frontend --port 8080 --type LoadBalancer
```

3. 为 frontend 创建路由：

```
$ oc expose svc/hello-world-frontend
```

4. 检查 frontend 路由：

- a. 获取路由详情：

```
$ oc get routes
```

输出结果应类似如下：

```
NAME          HOST/PORT
hello-world-frontend <frontend-url>
```

- b. 进入浏览器中的 **<frontend-url>** 值，您会看到类似以下内容的消息，因为 frontend 还无法与后端通信：

```
Trouble! HTTPConnectionPool(host='hello-world-backend', port=8080): Max retries
exceeded with url: /api/hello (Caused by
NewConnectionError('<urllib3.connection.HTTPConnection object at 0x7fbcdf0d1d0>:
Failed to establish a new connection: [Errno -2] Name or service not known'))
```

要解决这种情况，您必须创建后端服务并在服务网络中可用。

1.7. 创建后端服务，并使其在服务网络上可用

后端服务在 **east** 命名空间中运行，默认在服务网络中不可用。您可以使用 **skupper** 命令将服务公开给服务网络中的所有命名空间。backend 应用是一个简单的 Python 应用，它传递消息到 frontend 应用。

流程

1. 在 east 终端会话中部署后端服务：

```
$ oc create deployment hello-world-backend --image quay.io/skupper/hello-world-backend
```

2. 从 east 终端会话公开服务中的后端服务：

```
$ skupper expose deployment hello-world-backend --port 8080 --protocol tcp
```

3. 从 west 终端会话检查站点状态：

```
$ skupper status
```

输出结果应类似如下：

```
Skupper is enabled for namespace "west" in interior mode. It is connected to 1 other site. It
has 1 exposed service.
```

该服务从 **east** 命名空间公开。

4. 检查 west 终端会话中的 frontend 路由：

- a. 获取路由详情：

```
$ oc get routes
```

输出结果应类似如下：

```
NAME          HOST/PORT
hello-world-frontend <frontend-url>
```

- b. 进入浏览器中的 **<frontend-url>** 值，您会看到类似如下的消息：

```
I am the frontend. The backend says 'Hello from hello-world-backend-78cd4d7d8c-plr9 (1)'.
```

这显示了 frontend 如何从不同的 OpenShift 集群通过服务网络调用后端服务。

其它资源

- [使用控制台监控 AMQ 互连站点](#)
- [使用 CLI 配置 AMQ 互连站点](#)

1.8. 中断服务网络

这个步骤描述了如何删除您创建的服务网络。

1. 从 west 终端会话中删除 **west** 命名空间：

```
$ oc delete project west
```

2. 从 east 终端会话中删除 **east** 命名空间：

```
$ oc delete project east
```

2021-08-27 00:34:32 +1000 修订