



# Red Hat Service Interconnect 1.5

## 安装

安装 CLI 或 Operator



## Red Hat Service Interconnect 1.5 安装

---

安装 CLI 或 Operator

## 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<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.

## 摘要

本指南论述了如何安装 Red Hat Service Interconnect 1.5。Red Hat Service Interconnect 是一个开源 Skupper 项目的红帽构建。

---

# 目录

<b>第 1 章 安装 CLI</b> .....	<b>3</b>
1.1. 下载二进制文件	3
1.2. 使用红帽软件包	3
<b>第 2 章 安装 OPERATOR</b> .....	<b>5</b>
2.1. 使用 CLI 为所有命名空间安装 OPERATOR	5
2.2. 使用 CLI 为单一命名空间安装 OPERATOR	6
2.3. 使用 OPENSIFT 控制台安装 OPERATOR	7
<b>第 3 章 升级 CLI 和站点</b> .....	<b>9</b>
<b>第 4 章 升级 RED HAT SERVICE INTERCONNECT OPERATOR 和站点</b> .....	<b>10</b>
<b>第 5 章 安装策略系统</b> .....	<b>11</b>
<b>第 6 章 使用您的订阅</b> .....	<b>12</b>
6.1. 访问您的帐户	12
6.2. 激活订阅	12
6.3. 为系统注册软件包	12
<b>附录 A. SKUPPER 策略 CRD 的 YAML</b> .....	<b>13</b>
<b>第 7 章 KUBERNETES 上的部署选项</b> .....	<b>15</b>
7.1. 扩展以增加流量	15
7.2. 创建高可用性站点	15
<b>附录 B. 关于 SERVICE INTERCONNECT 文档</b> .....	<b>17</b>
使开源包含更多	17



# 第 1 章 安装 CLI

Skupper CLI 提供了创建 Kubernetes 和 Podman 站点的方法。

安装 CLI 的方法有两种：

- [第 1.1 节 “下载二进制文件”](#)
- [第 1.2 节 “使用红帽软件包”](#)

## 1.1. 下载二进制文件

下载 Skupper CLI 二进制文件是开始使用 Red Hat Service Interconnect 的快速方法。但是，请考虑在 Linux 上使用红帽软件包来获得最新更新。

### 流程

1. 为 Linux、macOS 或 Windows 下载二进制文件，在 [Software Downloads](#) 处选择 1.5 的最新版。  
对于带有 Apple silicon 的 Mac，在 x86-64 下载上使用 [Rosetta 2](#) 和 Skupper CLI for Mac。
2. 解压下载的文件，并将 Skupper 可执行文件放在 PATH 中。
3. 验证安装：

```
$ skupper version
client version      1.5.3-rh-5
```

## 1.2. 使用红帽软件包

在 Linux 上安装红帽软件包可确保您收到 Skupper CLI 的最新更新。

### 先决条件

- 确定您的订阅已激活，且您的系统已经注册。有关使用客户门户网站激活红帽订阅并为软件包注册您的系统的详情，请参考 [第 6 章 使用您的订阅](#)。

### 流程

1. 使用 `subscription-manager` 命令订阅所需的软件包存储库。对于主发行流，将 `<version>` 替换为 `1`，用于长期支持发行流。



### 注意

将 `&lt;version>` 替换为 `1` 安装 `1.5`，而 `1.5` 是主发行流并在进一步发布后更改。

Red Hat Enterprise Linux 8

```
$ sudo subscription-manager repos --enable=service-interconnect-_  
x86_64-rpms
```

Red Hat Enterprise Linux 9

```
$ sudo subscription-manager repos --enable=service-interconnect-_  
x86_64-rpms
```

- a. 使用 **yum** 或 **dnf** 命令安装 **skupper** 命令：

```
$ sudo dnf install skupper-cli
```

#### 附加信息

- 请参阅“Hello world”教程 [的示例](#)。
- 使用 `man containers.conf` 查看有关 `podman` 配置的更多信息。



## 第 2 章 安装 OPERATOR

Red Hat Service Interconnect Operator 在 OpenShift 中创建和管理站点。



### 注意

Red Hat Service Interconnect Operator 只在 OpenShift 4 上被支持。安装 Operator 需要集群的管理员特权。

### 2.1. 使用 CLI 为所有命名空间安装 OPERATOR

本节中的步骤演示了如何使用 `oc` 命令在给定的 OpenShift 集群中安装和部署 Red Hat Service Interconnect Operator 的最新版本。通过为所有命名空间安装 Operator，您可以在任意命名空间中创建站点。如需有关 `oc` 命令的更多信息，请参阅 [OpenShift CLI 入门](#)。

#### 先决条件

- 使用 `cluster-admin` 帐户访问 OpenShift 集群。如需支持的 OpenShift 版本，请参阅 [发行注记](#)。

#### 流程

1. 以集群管理员身份登录 OpenShift。例如：

```
$ oc login -u system:admin
```

2. 完成 [Red Hat Container Registry Authentication](#) 中所述的步骤。
3. 使用以下内容创建一个名为 `subscription-all.yaml` 的文件：

```
apiVersion: operators.coreos.com/v1alpha1
kind: Subscription
metadata:
  name: skupper-operator
  namespace: openshift-operators
spec:
  channel: stable-1
  installPlanApproval: Automatic
  name: skupper-operator
  source: redhat-operators
  sourceNamespace: openshift-marketplace
  startingCSV: skupper-operator.v1.5.3-rh-1
```



### 注意

如果要保持限制为 1.5.x 版本的更新，请将 `channel` 的值设置为 `stable-1.5`。

如果没有指定 `startCSV`，订阅默认为最新的 operator 版本。

如果将 `installPlanApproval` 指定为 `Manual`，则站点不会自动升级到 Service Interconnect 的最新版本。有关手动升级站点的信息，请参阅 [第 4 章 升级 Red Hat Service Interconnect Operator 和站点](#)。

4. 应用订阅 YAML：

```
$ oc apply -f subscription-all.yaml
```

### 附加信息

- 有关使用 YAML [创建站点的说明](#)，请参阅[使用 Skupper](#)。

## 2.2. 使用 CLI 为单一命名空间安装 OPERATOR

本节中的步骤演示了如何使用 `oc` 命令在给定的 OpenShift 集群中安装和部署 Red Hat Service Interconnect Operator 的最新版本。为单个命名空间安装 Operator 允许您在指定命名空间中创建站点。如需有关 `oc` 命令的更多信息，请参阅 [OpenShift CLI 入门](#)。

### 先决条件

- 使用 `cluster-admin` 帐户访问 OpenShift 集群。如需支持的 OpenShift 版本，请参阅[发行注记](#)。

### 流程

1. 以集群管理员身份登录 OpenShift。例如：

```
$ oc login -u system:admin
```

2. 完成 [Red Hat Container Registry Authentication](#) 中所述的步骤。
3. 在您要创建站点的命名空间中创建 Operator 组：
  - a. 使用以下内容创建一个名为 `operator-group.yaml` 的文件：

```
kind: OperatorGroup
apiVersion: operators.coreos.com/v1
metadata:
  name: skupper-operator
  namespace: my-namespace
spec:
  targetNamespaces:
    - my-namespace
```

其中 `my-namespace` 是您要创建站点的命名空间的名称。

- b. 应用 Operator 组 YAML：

```
$ oc apply -f operator-group.yaml
```

4. 使用以下内容创建一个名为 `subscription-myns.yaml` 的文件：

```
apiVersion: operators.coreos.com/v1alpha1
kind: Subscription
metadata:
  name: skupper-operator
  namespace: my-namespace
spec:
  channel: stable-1
  installPlanApproval: Automatic
```

```
name: skupper-operator
source: redhat-operators
sourceNamespace: openshift-marketplace
startingCSV: skupper-operator.v1.5.3-rh-1
```

其中 `my-namespace` 是您要创建站点的命名空间的名称。



### 注意

如果要保持限制为 1.5.x 版本的更新，请将 `channel` 的值设置为 `stable-1.5`。

如果没有指定 `startCSV`，订阅默认为最新的 operator 版本。

如果将 `installPlanApproval` 指定为 `Manual`，则站点不会自动升级到 Service Interconnect 的最新版本。有关手动升级站点的信息，请参阅 [第 4 章 升级 Red Hat Service Interconnect Operator 和站点](#)。

### 5. 应用订阅 YAML :

```
$ oc apply -f subscription-myns.yaml
```

### 附加信息

- 有关使用 YAML [创建站点的说明](#)，请参阅[使用 Skupper](#)。

## 2.3. 使用 OPENSIFT 控制台安装 OPERATOR

本节中的步骤演示了如何使用 OpenShift 控制台中的 OperatorHub 在给定的 OpenShift 命名空间中安装和部署 Red Hat Service Interconnect Operator 的最新版本。

### 先决条件

- 使用 `cluster-admin` 帐户访问 OpenShift 集群。如需支持的 OpenShift 版本，请参阅[发行注记](#)。

### 流程

1. 在 OpenShift Web 控制台中，进入到 Operators → OperatorHub。
2. 从可用的 Operator 列表中选择 Red Hat Service Interconnect Operator，然后点 Install。
3. 在 Operator 安装 页面中，有两个安装模式 选项：
  - 集群中的所有命名空间
  - 集群中的特定命名空间  
在本例中，选择 A specific namespace on the cluster
4. 选择一个 Update approval 选项。  
默认情况下，选择 Automatic 批准，站点将升级到最新版本的 Service Interconnect。如果选择手动 批准，则站点不会自动升级到最新版本的 Service Interconnect。有关手动升级站点的信息，请参阅 [第 4 章 升级 Red Hat Service Interconnect Operator 和站点](#)
5. 选择您要在其中安装 Operator 的命名空间，然后点 Install。  
Installed Operators 页会出现，并显示 Operator 安装的状态。

6. 验证 Red Hat Service Interconnect Operator 是否已显示并等待 Status 变为 Succeeded。
7. 如果安装不成功，请排除错误：
  - a. 在 Installed Operators 页中点 Red Hat Service Interconnect Operator。
  - b. 选择 Subscription 标签页并查看所有失败或错误。

有关安装 Operator 的更多信息，请参阅 [OpenShift 文档](#)

#### 附加信息

- 有关使用 YAML [创建站点的说明](#)，请参阅[使用 Skupper](#)。

## 第 3 章 升级 CLI 和站点

升级站点需要 Skupper CLI 的最新版本。



### 注意

更新所有站点，以确保 Service Interconnect 的同一版本在您的服务网络中运行。在更新的过程中，可能会有短暂的停机时间。

### 流程

#### 1. 升级 skupper CLI :

```
$ dnf upgrade skupper-cli
```

#### 2. 升级每个站点 :

a. 将上下文设置为您要升级的站点。

b. 运行 upgrade 命令 :

```
$ skupper update
```

## 第 4 章 升级 RED HAT SERVICE INTERCONNECT OPERATOR 和 站点

此发行版本将 Operator 频道名称从 **alpha** 更改。现在，您可以从以下频道中选择：

- **stable-1.5** - 将更新限制为 1.5.x。
- **stable-1** - 将更新限制为 1.x。
- **稳定** - 所有版本。

您必须按照安装 Operator 来使用 1.5 所述再次 [安装 Operator](#)。

如果在安装 Red Hat Service Interconnect Operator 时选择了自动更新，则当您选择的频道有新版本 Service Interconnect 时，站点都会升级到将来的版本。如果选择手动更新，请完成此流程来升级站点。

### 流程

1. 登录到 OpenShift 控制台。
2. 进入到 Installed Operators 页面。
3. 如果 Upgrade available 显示为 Red Hat Service Interconnect Operator 的 Status，点该文本。
4. 在 InstallPlan 详情页面中，点 Preview InstallPlan。
5. 点 Approve 来升级站点。

## 第 5 章 安装策略系统

在集群上安装 Skupper 策略系统可让您控制在集群中使用 Skupper 的方式。



### 注意

在没有特定策略规则的情况下在集群中应用策略系统会禁止站点链接和服务暴露。如果要在现有站点的集群中安装策略系统，则必须在安装策略系统前创建策略以避免中断。

### 先决条件

- 使用 `cluster-admin` 权限访问 Kubernetes 集群。
- 安装了 Red Hat Service Interconnect Operator

### 流程

1. 登录到集群。
2. 部署策略 CRD :

```
$ kubectl apply -f skupper_cluster_policy_crd.yaml  
  
customresourcedefinition.apiextensions.k8s.io/skupperclusterpolicies.skupper.io created  
clusterrole.rbac.authorization.k8s.io/skupper-service-controller created
```

其中，在 [附录 A, Skupper 策略 CRD 的 YAML](#) 附录中指定 `skupper_cluster_policy_crd.yaml` 的内容。

### 附加信息

有关使用 [策略的更多信息](#)，请参阅 [使用策略保护服务网络](#)。

## 第 6 章 使用您的订阅

Red Hat Service Interconnect 通过软件订阅提供。要管理您的订阅，请访问红帽客户门户中的帐户。

### 6.1. 访问您的帐户

#### 流程

1. 转至 [access.redhat.com](https://access.redhat.com)。
2. 如果您还没有帐户，请创建一个帐户。
3. 登录到您的帐户。

### 6.2. 激活订阅

#### 流程

1. 转至 [access.redhat.com](https://access.redhat.com)。
2. 导航到 My Subscriptions。
3. 导航到 激活订阅 并输入您的 16 位激活号。

### 6.3. 为系统注册软件包

要在 Red Hat Enterprise Linux 上安装此产品的 RPM 软件包，必须注册您的系统。如果您使用下载的发行文件，则不需要这一步。

#### 流程

1. 转至 [access.redhat.com](https://access.redhat.com)。
2. 进入 Registration Assistant。
3. 选择您的操作系统版本，再继续到下一页。
4. 使用您的系统终端中列出的命令完成注册。

有关注册您的系统的更多信息，请参阅以下资源之一：

- [Red Hat Enterprise Linux 8 - 注册系统并管理订阅](#)
- [Red Hat Enterprise Linux 9 - 注册系统并管理订阅](#)



## 附录 A. SKUPPER 策略 CRD 的 YAML

策略系统允许集群管理员限制集群中的 Skupper 使用。典型的 Skupper 使用不需要它。

以下 YAML 将 Skupper 策略 CRD 应用到集群。

```

---
apiVersion: apiextensions.k8s.io/v1
kind: CustomResourceDefinition
metadata:
  name: skupperclusterpolicies.skupper.io
spec:
  group: skupper.io
  versions:
  - name: v1alpha1
    served: true
    storage: true
    schema:
      openAPIV3Schema:
        type: object
        properties:
          spec:
            type: object
            properties:
              namespaces:
                type: array
                items:
                  type: string
              allowIncomingLinks:
                type: boolean
              allowedOutgoingLinksHostnames:
                type: array
                items:
                  type: string
              allowedExposedResources:
                type: array
                items:
                  type: string
              allowedServices:
                type: array
                items:
                  type: string
        scope: Cluster
      names:
        plural: skupperclusterpolicies
        singular: skupperclusterpolicy
        kind: SkupperClusterPolicy
---
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
  labels:
    application: skupper-service-controller
  name: skupper-service-controller
rules:
  - apiGroups:

```

- skupper.io  
resources:  
- skupperclusterpolicies  
verbs:  
- get  
- list  
- watch  
- apiGroups:  
- ""  
resources:  
- namespaces  
verbs:  
- get

## 第 7 章 KUBERNETES 上的部署选项

在 Kubernetes 上创建站点时，您可以使用许多选项。例如，您可以设置 pod 数量以及分配给每个 pod 的资源。本指南专注于以下目标：

- [第 7.1 节 “扩展以增加流量”](#)
- [第 7.2 节 “创建高可用性站点”](#)

### 7.1. 扩展以增加流量

对于最佳网络延迟和吞吐量，您可以使用 `router-cpu` 选项调整路由器的 CPU 分配。路由器 CPU 是管理 Skupper 网络性能的主要因素。



#### 注意

增加路由器数量不会提高网络性能。传入的 `router-to-router` 链接仅与一个活跃的路由器关联。在路由器响应时，额外的路由器不会接收流量

1. 确定您需要的路由器 CPU 分配。  
默认情况下，路由器 CPU 分配是 `BestEffort`，如 [Pod 服务质量类](#) 中所述。

考虑以下 CPU 分配选项：

路由器 CPU	描述
1	有助于避免在低资源集群中出现 <code>BestEffort</code> 的问题
2	适用于生产环境
5	最大性能

2. 如果使用 Skupper CLI，请使用 `--router-cpu` 选项设置路由器的 CPU 分配。例如：

```
$ skupper init --router-cpu 2
```

3. 如果使用 YAML，请通过为 `router-cpu` 属性设置值来设置路由器的 CPU 分配。例如：

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: "skupper-site"
data:
  name: "my-site"
  router-cpu: 2
```

### 7.2. 创建高可用性站点

默认情况下，Kubernetes 会重启任何变得无响应的路由器。（如果您遇到路由器重启，请考虑 [第 7.1 节 “扩展以增加流量”](#) 以提高响应性。）

如果您正在运行 Skupper 的集群非常忙碌，则 Kubernetes 可能需要调度新的路由器 pod。您可以通过在站点中部署两个路由器来“预调度”备份路由器。

1. 如果使用 Skupper CLI，请使用 `--routers` 选项将路由器数量设置为 2：

```
$ skupper init --routers 2
```

2. 如果使用 YAML，请通过设置 `router` 属性将路由器数量设置为 2：

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: "skupper-site"
data:
  name: "my-site"
  routers: 2
```

将路由器数量设置为超过 2 的路由器不会提供更高的可用性，并可能会对性能造成负面影响。

**注意：**当路由器重启或流量重定向到备份路由器时，客户端必须重新连接。

## 附录 B. 关于 SERVICE INTERCONNECT 文档

### 使开源包含更多

红帽致力于替换我们的代码、文档和 Web 属性中存在问题的语言。我们从这四个术语开始：**master**、**slave**、**黑名单**和**白名单**。由于此项工作十分艰巨，这些更改将在即将推出的几个发行版本中逐步实施。详情请查看 [CTO Chris Wright 的信息](#)。

*更新于 2024-06-25*