



Red Hat Service Interconnect 1.5

发行注记

有关本发行版本中的功能和问题的最新信息

Red Hat Service Interconnect 1.5 发行注记

有关本发行版本中的功能和问题的最新信息

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

本发行注记包含有关 Red Hat Service Interconnect 1.5 发行版本中包含的新功能、增强功能、修复和问题的最新信息。Red Hat Service Interconnect 是一个开源 Skupper 项目的红帽构建。

目录

第 1 章 新增和改变的功能	3
第 2 章 支持的配置	4
第 3 章 技术预览功能	6
第 4 章 已弃用的功能	7
第 5 章 升级站点	8
第 6 章 修复的问题	9
第 7 章 已知问题	10
附录 A. 关于 SERVICE INTERCONNECT 文档	13
使开源包含更多	13

第 1 章 新增和改变的功能

红帽构建的 [Skupper](#) 现在被称为 Red Hat Service Interconnect。

在 1.4 版本中指定了以下功能，但现已获得全面支持：

Podman 站点

在这个版本中，您可以在 RHEL 主机和 Kubernetes 命名空间上创建站点。如需更多信息，[请参阅使用 Skupper podman](#)。

控制台 (Console)

在这个版本中，您可以选择性地置备控制台来监控服务网络间的流量流。如需更多信息，[请参阅使用 Skupper 控制台](#)。此控制台的早期版本使用内存 prometheus 实例来填充流量视觉化。prometheus 实例和指标不适用于公共用途，红帽建议您只测试并提供反馈控制台功能。

第 2 章 支持的配置

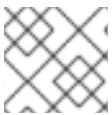
命令行界面

- Red Hat Enterprise Linux 8 x86-64 和 aarch64
- Red Hat Enterprise Linux 9 x86-64 和 aarch64
- Linux x86-64 和 aarch64
- macOS x86-64 ([技术预览功能](#))
- Windows x86-64 ([技术预览功能](#))

路由器

用于基于 Kubernetes 的站点，以及容器或机器的网关。

- Red Hat Enterprise Linux 8 x86-64 和 aarch64
- Red Hat Enterprise Linux 9 x86-64 和 aarch64



注意

Red Hat Service Interconnect 不支持作为消息传递路由器独立使用。

Red Hat Service Interconnect Operator

Operator 仅支持 OpenShift 4.x。

OpenShift 版本

- OpenShift 3.11
- OpenShift 4.12、4.13、4.14 和 4.15
- ROSA 和 ARO
- OpenShift Container Platform 和 OpenShift Dedicated

支持将所需组件镜像到断开连接的网络中安装 Red Hat Service Interconnect。

Ingress 类型

- LoadBalancer
- OpenShift 路由

CPU 架构

- x86-64 和 aarch64

多架构站点：

在 aarch64 架构集群或 Linux 主机上创建的[站点是一个技术预览功能](#)。

Windows 和 macOS CLI

Windows 和 macOS 的 Skupper CLI 是一个技术预览功能。

Kubernetes 发行版本

红帽在 [Kubernetes 的任何 CNCF 认证分发](#) 中提供了运行 Red Hat Service Interconnect 的帮助。

但请注意，Red Hat Service Interconnect 仅在 OpenShift 中测试。

Ingress 类型

- Contour
- nginx - 这需要配置 TLS 透传
- NodePort

升级

红帽支持从一个下游次版本升级到下一个次版本，而无需跳过。虽然红帽旨在在次版本间具有兼容性，但我们建议将所有站点升级到最新版本。



注意

如果您的应用程序需要长期实时连接，如 Kafka 客户端，请考虑将负载均衡器用作入口，而不是代理入口，如 OpenShift 路由。如果您使用 OpenShift 路由作为入口，则预期配置路由时中断。

有关最新版本的详情，请参考 [Red Hat Service Interconnect 支持的配置](#)。

第 3 章 技术预览功能

这个版本中的一些功能当前还处于技术预览状态。这部分论述了 Red Hat Service Interconnect 1.5 中的技术预览功能。



重要

技术预览功能不受红帽产品服务等级协议（SLA）支持，且功能可能并不完整。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[技术预览功能支持范围](#)。

Windows 和 macOS

在这个版本中，您可以在 Windows 和 macOS 上使用 **skupper** CLI。

Multiarch

在这个版本中，您可以在 64 位 ARM 集群和 Linux 主机(aarch64)上创建站点。

第 4 章 已弃用的功能

本节论述了支持但已从 Red Hat Service Interconnect 中弃用的功能。

协议

http 和 **http2** 协议已弃用，当提供类似可观察性的功能可用时，将在以后的发行版本中删除。红帽建议使用 **tcp** 协议，除非需要 **http** 或 **http2** observability。

第 5 章 升级站点

此 Red Hat Service Interconnect 发行版本与之前的版本兼容，但红帽建议将所有站点升级到 1.5 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。

要升级站点：

```
$ skupper update
```

有关升级基于 Operator 的站点的说明，请参阅安装。https://access.redhat.com/documentation/zh-cn/red_hat_service_interconnect/1.5/html-single/installation/



注意

如果您在 1.4 之前使用发行版本（这是默认设置）启用了控制台，则升级的站点也会启用控制台。Service Interconnect 1.4 及更新的版本的默认设置是没有启用控制台。

要升级网关，请删除网关并重新创建。

端口协商限制

如果您的协议协商通信端口，如活跃的 FTP，则无法使用该协议在服务网络中进行通信。

第 6 章 修复的问题

如需补丁版本中修复的问题列表，请参阅 [Red Hat Service Interconnect 1.5.x 解决问题](#)。

第 7 章 已知问题

- **SKUPPER-1802** - skupper 服务状态报告当前站点中不存在的服务

如果您在网络上禁用了 **service-sync**，并在一个站点上公开服务，则 **skupper service status** 命令将这些服务报告为在所有其他站点上现有的，即使该服务在这些站点上不存在。

临时解决方案

要解决这个问题，请使用 **skupper service status -v** 了解公开的服务和可用性。

例如，在 **west** 网站上：

```
$ skupper service status

Services exposed through Skupper:
└─ backend:8080 (tcp)
```

但是，该服务不会在 **west** 站点上存在。使用详细输出来显示服务仅存在于 **east** 站点上：

```
$ skupper service status -v

Services exposed through Skupper:
└─ backend:8080 (tcp)
  └─ Sites:
    └─ 316fbe31-299b-490b-9391-7b46507d76f1(east)
      └─ policy: disabled
        └─ Targets:
          └─ backend:8080 name=backend-9d84544df-rbj
```

- **SKUPPER-869** - 为 TCP 传输启用闲置连接超时

如果端点被终止，例如客户端被终止，其他端点会观察到半关闭的连接。如果其他端点没有关闭连接或试图向连接发送数据，Skupper 路由器不会释放分配给该连接的内存

临时解决方案

如果可能，请避免使用此行为的客户端服务器配置。例如，如果服务器自动关闭 dormant 连接，或者尝试与客户端通信，则 Skupper 路由器会在客户端终止时释放内存。

- **SKUPPER-805** - skupper init 在 OCP 3.11 上不适用于普通用户。

临时解决方案

有两个临时解决方案：

- 使用 YAML 配置站点。
- 使用以下权限创建服务帐户来运行 **skupper CLI**：

```
---
apiVersion: rbac.authorization.k8s.io/v1
kind: Role
metadata:
  name: skupper-non-admin
rules:
- apiGroups:
  - ""
```

- resources:
 - configmaps
 - pods
 - pods/exec
 - services
 - secrets
 - serviceaccounts
- verbs:
 - get
 - list
 - watch
 - create
 - update
 - delete
- apiGroups:
 - apps
- resources:
 - deployments
 - statefulsets
 - daemonsets
- verbs:
 - get
 - list
 - watch
 - create
 - update
 - delete
- apiGroups:
 - route.openshift.io
- resources:
 - routes
- verbs:
 - get
 - list
 - watch
 - create
 - delete
- apiGroups:
 - networking.k8s.io
- resources:
 - ingresses
 - networkpolicies
- verbs:
 - get
 - list
 - watch
 - create
 - delete
- apiGroups:
 - projectcontour.io
- resources:
 - httpproxies
- verbs:
 - get
 - list
 - watch

```
- create
- delete
- apiGroups:
  - rbac.authorization.k8s.io
  resources:
  - rolebindings
  - roles
  verbs:
  - get
  - list
  - watch
  - create
  - delete
```

您可以将上面的 YAML 保存到 **role.yaml** 中，应用它并将角色绑定到用户名：

```
$ oc apply -f role.yaml
$ oc policy add-role-to-user skupper-non-admin <username> -n <namespace-name> --role-namespace=<namespace-name>
```


附录 A. 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