



# Red Hat Service Interconnect 1.5

## リリースノート

このリリースの機能および問題に関する最新情報



# Red Hat Service Interconnect 1.5 リリースノート

---

このリリースの機能および問題に関する最新情報

## 法律上の通知

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.

## 概要

このリリースノートには、Red Hat Service Interconnect 1.5 リリースに含まれる新機能、機能拡張、修正、および問題に関する最新情報が含まれています。Red Hat Service Interconnect は、オープンソースの Skupper プロジェクトの Red Hat ビルドです。

---

## 目次

第1章 新機能および変更された機能 .....	3
第2章 サポートされる構成 .....	4
第3章 テクノロジープレビュー機能 .....	6
第4章 非推奨の機能 .....	7
第5章 サイトのアップグレード .....	8
第6章 修正された問題 .....	9
第7章 既知の問題 .....	10
付録A SERVICE INTERCONNECT のドキュメントについて .....	13
多様性を受け入れるオープンソースの強化 .....	13



## 第1章 新機能および変更された機能

Red Hat build of [Skupper](#) は、Red Hat Service Interconnect として知られています。

以下の機能は、バージョン 1.4 のテクノロジープレビューに指定されていましたが、完全にサポートされるようになりました。

### Podman サイト

Red Hat Service Interconnect の今回のリリースでは、RHEL ホストおよび Kubernetes namespace でサイトを作成できます。詳細は、[Using Skupper podman](#) を参照してください。

### Console

Red Hat Service Interconnect の今回のリリースでは、任意でコンソールをプロビジョニングし、サービスネットワーク全体のトラフィックフローを監視できます。詳細は、[Using the Skupper console](#) を参照してください。この初期バージョンのコンソールでは、メモリー内の prometheus インスタンスを使用して、トラフィックの可視化を設定します。prometheus インスタンスおよびメトリクスは、一般公開を目的としていません。Red Hat では、コンソール機能のみをテストし、フィードバックを提供することを推奨しています。

## 第2章 サポートされる構成

### コマンドラインインターフェイス

- Red Hat Enterprise Linux 8 x86-64 および aarch64
- Red Hat Enterprise Linux 9 x86-64 および aarch64
- Linux x86-64 および aarch64
- macOS x86-64 (テクニカルプレビュー機能)
- Windows x86-64 (テクニカルプレビュー機能)

### ルーター

Kubernetes ベースのサイトで使用するほか、コンテナまたはマシンのゲートウェイとして使用します。

- Red Hat Enterprise Linux 8 x86-64 および aarch64
- Red Hat Enterprise Linux 9 x86-64 および aarch64



#### 注記

Red Hat Service Interconnect は、メッセージングルーターとしてのスタンドアロン使用はサポートされていません。

### Red Hat Service Interconnect Operator

Operator は OpenShift 4.x でのみサポートされます。

### OpenShift のバージョン

- OpenShift 3.11
- OpenShift 4.12、4.13、4.14、4.15 および 4.16
- ROSA および ARO
- OpenShift Container Platform および OpenShift Dedicated

必要なコンポーネントをクラスターにミラーリングすることにより、非接続ネットワークに Red Hat Service Interconnect をインストールすることがサポートされています。

### Ingress タイプ

- LoadBalancer
- OpenShift ルート

### CPU アーキテクチャー

- x86\_64 および AArch64



## Multiarch サイト :

aarch64 アーキテクチャクラスターまたは Linux ホスト上に作成されたサイトは [テクニカルプレビュー機能](#) です。

## Windows および macOS CLI

Windows および macOS 用の Skupper CLI は、[テクニカルプレビュー機能](#) です。

## Kubernetes ディストリビューション

Red Hat は [Kubernetes の CNCF 認定ディストリビューション](#) 上で Red Hat Service Interconnect を実行するためのサポートを提供します。

ただし、Red Hat Service Interconnect は OpenShift でのみテストされている点に注意してください。

## Ingress タイプ

- Contour
- Nginx - これには、TLS パススルーの設定が必要です
- NodePort

## Upgrades

Red Hat は、1つのダウンストリームマイナーバージョンから次のマイナーバージョンへと、連続したアップグレードをサポートしています。Red Hat はマイナーバージョン間の互換性を確保することを目指していますが、すべてのサイトを最新バージョンにアップグレードすることを推奨します。



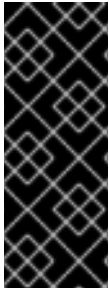
### 注記

Kafka クライアントなど、長時間継続する接続を必要とするアプリケーションがある場合は、OpenShift ルートなどのプロキシ入力ではなく、ロードバランサーを Ingress として使用することを検討してください。OpenShift ルートを Ingress として使用すると、ルートが設定されるたびに中断が発生することが予想されます。

最新リリースの詳細は、[Red Hat Service Interconnect Supported Configurations](#) を参照してください。

## 第3章 テクノロジープレビュー機能

現在、今回のリリースに含まれる機能にはテクノロジープレビューのものがあります。このセクションでは、Red Hat Service Interconnect 1.5 のテクノロジープレビュー機能を説明します。



### 重要

テクノロジープレビュー機能は、Red Hat 製品のサービスレベルアグリーメント (SLA) の対象外であり、機能的に完全ではないことがあります。Red Hat では、実稼働環境での使用を推奨していません。テクノロジープレビューの機能は、最新の製品機能をいち早く提供して、開発段階で機能のテストを行いフィードバックを提供していただくことを目的としています。Red Hat のテクノロジープレビュー機能のサポート範囲に関する詳細は、[テクノロジープレビュー機能のサポート範囲](#) を参照してください。

### Windows および macOS

Red Hat Service Interconnect のこのリリースでは、Windows および macOS で **skupper** CLI を使用できるようになりました。

### Multiarch

Red Hat Service Interconnect のこのリリースでは、64 ビット ARM クラスターと Linux ホスト (aarch64) 上にサイトを作成できます。

## 第4章 非推奨の機能

このセクションでは、サポートはされていますが、Red Hat Service Interconnect では非推奨になっている機能を説明します。

### プロトコル

**http** プロトコルおよび **http2** プロトコルは非推奨になりました。同様の可観測性を提供する機能が利用可能になると、今後のリリースで削除される予定です。Red Hat は、**http** または **http2** の可観測性が不要でない限り、**tcp** プロトコルを使用することを推奨します。

## 第5章 サイトのアップグレード

Red Hat Service Interconnect のこのリリースは、以前のバージョンと互換性がありますが、Red Hat ではすべてのサイトをバージョン 1.5 にアップグレードすることを推奨しています。



### 注記

すべてのサイトを更新して、同じバージョンの Service Interconnect がサービスネットワーク全体で実行されるようにします。更新プロセス中に、わずかなダウンタイムが発生することが予想されます。

サイトをアップグレードするには、以下を実行します。

```
$ skupper update
```

Operator ベースのサイトのアップグレードに関する詳細は、[インストール](#) を参照してください。



### 注記

デフォルトであった 1.4 より前のリリースを使用してコンソールを有効にした場合、アップグレードされたサイトでもコンソールが有効になります。Service Interconnect 1.4 以降のデフォルトでは、コンソールは有効になっていません。

ゲートウェイをアップグレードするには、そのゲートウェイを削除し、再作成します。

### ポートネゴシエーションの制限

プロトコルが通信ポート (アクティブな FTP など) をネゴシエートする場合、そのプロトコルを使用してサービスネットワーク全体で通信することはできません。

## 第6章 修正された問題

パッチリリースで修正された問題のリストは、[Red Hat Service Interconnect 1.5.x の解決済みの問題](#) を参照してください。

## 第7章 既知の問題

- **SKUPPER-1802** - skupper service status reports a service that doesn't exist on the current site  
ネットワーク上で **service-sync** を無効にし、あるサイトでサービスを公開した場合、**skupper service status** コマンドは、そのサービスが他のすべてのサイトに存在しないにもかかわらず、それらのサービスが他のすべてのサイトに存在すると報告します。

### 回避策

この問題を回避するには、**skupper service status -v** を使用して、公開されているサービスと可用性を確認します。

たとえば、**west** サイトでは次のようになります。

```
$ skupper service status

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

ただし、**west** サイトにはそのサービスは存在しません。詳細出力には、サービスが **east** サイトにのみ存在することが示されています。

```
$ skupper service status -v

Services exposed through Skupper:
└─ backend:8080 (tcp)
  └─ Sites:
    └─ 316f3e31-299b-490b-9391-7b46507d76f1(east)
      └─ policy: disabled
        └─ Targets:
          └─ backend:8080 name=backend-9d84544df-rbzj
```

- **SKUPPER-869** - TCP トランスポートのアイドル接続タイムアウトを有効にする  
クライアントが強制終了するなど、エンドポイントが終了すると、もう一方のエンドポイントは半分切断された接続を監視します。他のエンドポイントが接続を切断しないか、接続へのデータ送信を試行しないと、Skupper ルーターによってその接続に割り当てられたメモリーが解放されません。

### 回避策

可能であれば、この動作を使用するクライアントサーバー設定を避けてください。たとえば、サーバーが休止状態の接続を自動的に閉じるか、クライアントとの通信を試みる場合は、クライアントが終了すると、Skupper ルーターはメモリーを解放します。

- **SKUPPER-805** - OCP 3.11 の通常ユーザーでは **skupper init** が使用できない  
**回避策**

以下の2つの回避策があります。

- YAML を使用してサイトを設定します。
- 以下のパーミッションでサービスアカウントを作成し、**skupper** CLI を実行します。

```
---
```

```
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
```

上記の YAML を **role.yaml** に保存し、これを適用して、以下を使用してロールをユーザー名にバインドできます。

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



## 付録A SERVICE INTERCONNECT のドキュメントについて

### 多様性を受け入れるオープンソースの強化

Red Hat では、コード、ドキュメント、Web プロパティにおける配慮に欠ける用語の置き換えに取り組んでいます。まずは、マスター (master)、スレーブ (slave)、ブラックリスト (blacklist)、ホワイトリスト (whitelist) の 4 つの用語の置き換えから始めます。この取り組みは膨大な作業を要するため、用語の置き換えは、今後の複数のリリースにわたって段階的に実施されます。詳細は、[Red Hat CTO である Chris Wright のメッセージ](#) をご覧ください。

改訂日時: 2024-08-15