Red Hat Service Interconnect 1.4

Release notes

Latest information about features and issues in this release

Last Updated: 2024-02-26

Red Hat Service Interconnect 1.4 Release notes

Latest information about features and issues in this release

Legal Notice

Abstract

These release notes contain the latest information about new features, enhancements, fixes, and issues contained in the Red Hat Service Interconnect 1.4 release. Red Hat Service Interconnect is a Red Hat build of the open source Skupper project.

Table of Contents

CHAPTER 1. NEW AND CHANGED FEATURES

The Red Hat build of Skupper is now known as Red Hat Service Interconnect. The documentation for previous Limited Availability releases are available at Product Documentation for Red Hat Application Interconnect.

This release includes the following changes:

- Previous releases allowed you to expose services from other namespaces, however if the target of the service became unavailable **skupper service status** was unable to report the failure. With this release, **skupper init --enable-cluster-permissions** and **skupper expose --target-namespace <ns>** allows you to expose and monitor services with targets in other namespaces. See Exposing services from other namespaces for more information.

- You can now encrypt traffic from the pod to the Skupper router.

- You can now expose **deploymentconfig** objects from OpenShift 3.11 clusters.

- By default, a console is not provisioned when you create a site. To enable a console, see Using the Skupper console.

CHAPTER 2. TECHNOLOGY PREVIEW FEATURES

Some features in this release are currently in Technology Preview. This section describes the Technology Preview features in Red Hat Service Interconnect 1.4.

> **IMPORTANT**
>
> Technology Preview features are not supported with Red Hat production service level agreements (SLAs) and might not be functionally complete. Red Hat does not recommend using them in production. These features provide early access to upcoming product features, enabling customers to test functionality and provide feedback during the development process. For more information about the support scope of Red Hat Technology Preview features, see Technology Preview Features Support Scope .

Podman sites

With this release of Red Hat Service Interconnect, you can create sites on RHEL hosts as well as Kubernetes namespaces. See Using Skupper podman for more information.

Console

With this release of Red Hat Service Interconnect, you can optionally provision a console to monitor traffic flows across the service network. See Using the Skupper console for more information. This early version of the console uses an in-memory prometheus instance to populate traffic visualizations. The prometheus instance and metrics are not intended for public use, Red Hat encourage you to test and provide feedback on the console features only.

CHAPTER 3. SUPPORTED CONFIGURATIONS

You can create sites on OpenShift Container Platform versions 3.11, 4.10, 4.11 and 4.12. Commercially reasonable support is provided for any CNCF Certified Kubernetes cluster

The **skupper** CLI is supported on:

- RHEL 8 and 9

Gateways are supported on:

- RHEL 8 and 9

Ingress types:

- LoadBalancer
- OpenShift Routes

NOTE

If you have applications that require long lived connections, for example Kafka clients, consider using a load balancer as ingress instead of a proxy ingress such as OpenShift route. If you use an OpenShift route as ingress, expect interruptions whenever routes are configured.

For more information, see Red Hat Service Interconnect Supported Configurations .

CHAPTER 4. DEPRECATED FEATURES

This section describes features that are supported, but have been deprecated from Red Hat Service Interconnect.

Protocols

The **http** and **http2** protocols are deprecated and will be removed in a future release when a feature that provides similar observability becomes available. Red Hat recommends using the **tcp** protocol unless **http** or **http2** observability is required.

CHAPTER 5. UPGRADING SITES FROM RED HAT APPLICATION INTERCONNECT VERSION 1

This release of Red Hat Service Interconnect is compatible with Red Hat Application Interconnect version 1, however Red Hat recommends upgrading all sites to version 1.4.

NOTE

Update all sites to ensure the same version of Service Interconnect is running across your service network. You can expect some minimal downtime during the update process.

To upgrade a site:

```
$ skupper update
```

NOTE

If you enabled the console previously, which was the default, the upgraded site will also have the console enabled. The default for Service Interconnect 1.4 is that the console is not enabled. The console for Service Interconnect is a Technology Preview feature as described in Chapter 2, *Technology Preview features*.

To upgrade a gateway, delete the gateway and recreate it.

Port negotiation limitation

If your protocol negotiates the communication port, for example active FTP, you cannot use that protocol to communicate across a service network.

CHAPTER 6. FIXED ISSUES

See Red Hat Service Interconnect 1.4.x Resolved Issues for a list of issues that have been fixed in patch releases.

CHAPTER 7. KNOWN ISSUES

- **SKUPPER-1069 - skupper init fails on non-OpenShift clusters**
  If you create a site on a non-OpenShift cluster you might encounter a problem creating sites, with **skupper status** reporting **pending**.

  If you check the pods, you might see status similar to the following:

  ```
  $ kubectl get pods
  NAME                               READY   STATUS                      RESTARTS   AGE
  skupper-router-698478664c-6xq72            0/2     CreateContainerConfigError   0          17s
  skupper-service-controller-698c785d7-dqc8m   0/1     CreateContainerConfigError   0
  10s
  ```

  To confirm you are encountering this problem, search for a Kubernetes event similar to the following:

  ```
  $ kubectl get events| grep Warning

  Warning  Failed         17m (x4 over 17m)     kubelet           Error: container has
  runAsNonRoot and image will run as root
  ```

  Workaround

  Specify a user id as follows:

  ```
  $ skupper init --run-as-user 2000
  ```

  where 2000 is the id of the user you want to run the containers. You can assign any non-zero number as id.

  If you are creating sites using YAML, you specify the user in the data section:

  ```
   data:
    name: my-site
    run-as-user: "2000"
  ```

  NOTE

  This workaround does not enable you to deploy the Red Hat Service Interconnect console component on this site. You must deploy the console component on an OpenShift site until this issue is resolved.

- **SKUPPER-869 - Enable idle connection timeouts for the TCP transport**
  If an endpoint is terminated, for example a client is killed, the other endpoint observes a half-closed connection. If the other endpoint does not close the connection or attempt to send data to the connection, the Skupper router does not release the memory allocated to that connection

  Workaround

Avoid client server configurations that use this behavior if possible. For example, if a server automatically closes dormant connections, or attempts to communicate with client, the Skupper router frees the memory when a client is terminated.

- **SKUPPER-805 - skupper init doesn't work for ordinary user on OCP 3.11**
  Workaround

  Two workarounds are available:

  - Use YAML to configure a site.

  - Create a service account with the following permissions to run the **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
```

You can save the YAML above to **role.yaml**, apply it and bind the role to a username using:

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

APPENDIX A. ABOUT SERVICE INTERCONNECT DOCUMENTATION

MAKING OPEN SOURCE MORE INCLUSIVE

Red Hat is committed to replacing problematic language in our code, documentation, and web properties. We are beginning with these four terms: master, slave, blacklist, and whitelist. Because of the enormity of this endeavor, these changes will be implemented gradually over several upcoming releases. For more details, see our CTO Chris Wright's message .

Revised on 2024-02-26 17:45:22 UTC