



Red Hat Streams for Apache Kafka 2.7

RHEL 上の Streams for Apache Kafka 2.7 のリリースノート

Red Hat Enterprise Linux 上の Streams for Apache Kafka のこのリリースにおける新機能と変更点のハイライト

Red Hat Streams for Apache Kafka 2.7 RHEL 上の Streams for Apache Kafka 2.7 のリリースノート

Red Hat Enterprise Linux 上の Streams for Apache Kafka のこのリリースにおける新機能と変更点のハイライト

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[®] 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.

概要

リリースノートでは、Streams for Apache Kafka 2.7 リリースで導入された新機能、機能拡張、および修正についてまとめています。

目次

第1章 STREAMS FOR APACHE KAFKA の名称変更のお知らせ	3
第2章 FEATURES	4
2.1. KAFKA 3.7.0 のサポート	4
2.2. KRAFT: ZOOKEEPER ベースから KRAFT ベースの KAFKA クラスターへの移行のサポート	4
2.3. KRAFT: KRAFT ベースのクラスターで KAFKA をアップグレードする	5
2.4. RHEL 7 がサポート対象外に	5
第3章 機能拡張	6
3.1. KAFKA 3.7.0 の機能拡張	6
3.2. KAFKA BRIDGE のテキスト形式	6
第4章 テクノロジープレビュー	7
4.1. KRAFT モード	7
4.2. KAFKA STATIC QUOTA プラグインの設定	7
第5章 非推奨の機能	8
5.1. STREAMS FOR APACHE KAFKA 2.7.0 での JAVA 11 の非推奨化	8
5.2. KAFKA MIRRORMAKER 2 のアイデンティティレプリケーションポリシー	8
5.3. KAFKA MIRRORMAKER 1	8
5.4. KAFKA BRIDGE の SPAN 属性	9
第6章 修正された問題	10
第7章 既知の問題	11
7.1. RHEL 7 との非互換性	11
7.2. FIPS モードで実行する場合の JMX 認証	11
第8章 サポートされる構成	12
8.1. サポート対象のプラットフォーム	12
8.2. サポートされるクライアント	12
8.3. サポートされる APACHE KAFKA エコシステム	12
8.4. その他のサポートされる機能	13
8.5. ストレージ要件	13
第9章 コンポーネントの詳細	14
第10章 サポート対象となる RED HAT 製品との統合	16
10.1. RED HAT BUILD OF KEYCLOAK (旧称 RED HAT SINGLE SIGN-ON)	16

第1章 STREAMS FOR APACHE KAFKA の名称変更のお知らせ

ブランディングの取り組みの一環として、AMQ Streams の名称が **Streams for Apache Kafka** に変更されます。この変更は、Red Hat の Apache Kafka 向け製品に対するお客様の認知度を高めることを目的としています。現在は移行期間中であり、AMQ Streams という古い呼称が使用されている可能性があります。当社では、新しい名称を反映するために、ドキュメント、リソース、メディアの更新に積極的に取り組んでいます。

第2章 FEATURES

Streams for Apache Kafka 2.7 では、このセクションで説明する機能が導入されています。

RHEL 上の Streams for Apache Kafka 2.7 は、Apache Kafka 3.7.0 に基づいています。



注記

このリリースの機能拡張と解決されたバグをすべて確認するには、[Streams for Apache Kafka の Jira プロジェクト](#) を参照してください。

2.1. KAFKA 3.7.0 のサポート

Streams for Apache Kafka は、Apache Kafka バージョン 3.7.0 をサポートおよび使用するようになりました。サポート対象は、Red Hat によってビルドされた Kafka ディストリビューションのみです。

アップグレードの手順については、次のガイドの Streams for Apache Kafka および Kafka のアップグレード手順を参照してください。

- [KRaft モードでの RHEL 上の Streams for Apache Kafka の使用](#)
- [ZooKeeper を使用した RHEL 上の Streams for Apache Kafka の使用](#)

詳細は、[Kafka 3.7.0 リリースノート](#)を参照してください。

Kafka 3.6.x は、Streams for Apache Kafka 2.7 にアップグレードする目的でのみサポートされます。新しいバイナリーを使用するためにローリング更新を実行することが推奨されます。



注記

Kafka 3.7.0 は KRaft モードへのアクセスを提供します。KRaft モードでは、Raft プロトコルを利用することで ZooKeeper なしで Kafka が実行されます。

2.2. KRAFT: ZOOKEEPER ベースから KRAFT ベースの KAFKA クラスターへの移行のサポート

Streams for Apache Kafka の KRaft モードは、いくつかの制限がある [テクノロジープレビュー](#) ですが、このリリースでは、KRaft をサポートするいくつかの新機能が導入されています。KRaft の使用を支援するために、新しい使用ガイド [KRaft モードでの RHEL 上の Streams for Apache Kafka の使用](#) の提供を開始しました。

Kafka クラスターでメタデータ管理に ZooKeeper を使用している場合は、KRaft モードでの Kafka の使用に移行できるようになりました。

移行中に次の操作を行います。

1. クラスター管理用の ZooKeeper に代わるコントローラーノードのクォーラムをインストールします。
2. コントローラー設定で `zookeeper.metadata.migration.enable` フラグを `true` に設定して、KRaft の移行を有効にします。
3. ブローカーで `zookeeper.metadata.migration.enable` フラグを `true` に設定して、KRaft の移行を有効にします。

4. **broker** KRaft ロールとノード ID を追加して、ブローカーを KRaft の使用に切り替えます。
5. **zookeeper.metadata.migration.enable** プロパティを削除して、コントローラーを移行モードから切り替えます。

[KRaft モードへの移行](#) を参照してください。

2.3. KRAFT: KRAFT ベースのクラスターで KAFKA をアップグレードする

KRaft から KRaft へのアップグレードがサポートされるようになりました。インストールファイルを更新し、すべての Kafka ノードを設定して再起動します。その後、KRaft ベースの Kafka クラスターを、サポートされている新しい KRaft メタデータバージョンにアップグレードします。

KRaft メタデータバージョンの更新

```
./bin/kafka-features.sh --bootstrap-server <broker_host>:<port> upgrade --metadata 3.7
```

[KRaft ベースの Kafka クラスターのアップグレード](#) を参照してください。

2.4. RHEL 7 がサポート対象外に

RHEL 7 はサポート対象外になりました。この決定は、[既知の非互換性の問題](#) が原因で行われました。

第3章 機能拡張

Streams for Apache Kafka 2.7 では、いくつかの機能拡張が追加されています。

3.1. KAFKA 3.7.0 の機能拡張

Kafka 3.7.0 で導入された機能強化の概要については、[Kafka 3.7.0 リリースノート](#)を参照してください。

3.2. KAFKA BRIDGE のテキスト形式

プロデューサー操作を行う場合に、**POST** リクエストには、生成されるメッセージの **埋め込みデータ形式** を示す **Content-Type** ヘッダーを指定する必要があります。以前は、レコードとキーの値でサポートされている形式が JSON とバイナリーでした。**text** 形式も使用できるようになりました。

表3.1 サポートされているコンテンツタイプの形式

埋め込みデータ形式	Content-Type ヘッダー
JSON	Content-Type: application/vnd.kafka.json.v2+json
バイナリー	Content-Type: application/vnd.kafka.binary.v2+json
テキスト	Content-Type: application/vnd.kafka.text.v2+json

第4章 テクノロジープレビュー

Streams for Apache Kafka 2.7 に含まれるテクノロジープレビュー機能です。



重要

テクノロジープレビュー機能は、Red Hat の実稼働環境のサービスレベルアグリーメント (SLA) ではサポートされません。また、機能的に完全ではない可能性があるため、Red Hat はテクノロジープレビュー機能を実稼働環境に実装することは推奨しません。テクノロジープレビューの機能は、最新の技術をいち早く提供して、開発段階で機能のテストやフィードバックの収集を可能にするために提供されます。サポート範囲の詳細は、[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。

4.1. KRAFT モード

KRaft モードはテクノロジープレビューとして利用できます。

現在、Streams for Apache Kafka の KRaft モードには次の制限があります。

- KRaft モードから ZooKeeper を使用するモードへのダウングレードはサポートされていません。
- 複数のディスクを使用する JBOD ストレージはサポートされていません。
- Kafka クラスタから削除された Kafka ノードの登録解除。

4.2. KAFKA STATIC QUOTA プラグインの設定

Kafka Static Quota プラグインのテクノロジープレビューを使用して、Kafka クラスタのブローカーにスループットおよびストレージの制限を設定します。バイトレートのしきい値およびストレージクォータを設定して、ブローカーと対話するクライアントに制限を設けることができます。

Kafka Static Quota プラグインの設定例

```
client.quota.callback.class= io.strimzi.kafka.quotas.StaticQuotaCallback
client.quota.callback.static.produce= 1000000
client.quota.callback.static.fetch= 1000000
client.quota.callback.static.storage.soft= 4000000000000
client.quota.callback.static.storage.hard= 5000000000000
client.quota.callback.static.storage.check-interval= 5
```

[Kafka Static Quota プラグインを使用したブローカーへの制限の設定](#) を参照してください。

第5章 非推奨の機能

以下の機能は、これまでの Streams for Apache Kafka リリースではサポート対象でしたが、このリリースで非推奨となりました。

5.1. STREAMS FOR APACHE KAFKA 2.7.0 での JAVA 11 の非推奨化

Java 11 のサポートは、Kafka 3.7.0 および Streams for Apache Kafka 2.7.0 で非推奨になりました。Java 11 は、今後、クライアントを含む Streams for Apache Kafka コンポーネントでサポートされなくなります。

Streams for Apache Kafka は Java 17 をサポートしています。新しいアプリケーションを開発する場合は、Java 17 を使用してください。また、現在 Java 11 を使用しているアプリケーションの Java 17 への移行も計画してください。



注記

Streams for Apache Kafka 2.4.0 で Java 8 のサポートが削除されました。現在 Java 8 を使用している場合は、同様に Java 17 への移行を計画してください。

5.2. KAFKA MIRRORMAKER 2 のアイデンティティレプリケーションポリシー

ID レプリケーションポリシーは、リモートトピックの自動名前変更機能の後継として MirrorMaker 2 で使用される機能です。名前の前にソースクラスターの名前を付ける代わりに、トピックは元の名前を保持します。この設定は、アクティブ/パッシブバックアップおよびデータ移行シナリオに特に役立ちます。

ID レプリケーションポリシーを実装するには、MirrorMaker 2 設定でレプリケーションポリシークラス (**replication.policy.class**) を指定する必要があります。以前は、Streams for Apache Kafka の **mirror-maker-2-extensions** コンポーネントに含まれる **io.strimzi.kafka.connect.mirror.IdentityReplicationPolicy** クラスを指定できました。ただし、このコンポーネントは非推奨となり、今後削除される予定です。したがって、Kafka 独自のレプリケーションポリシークラス (**org.apache.kafka.connect.mirror.IdentityReplicationPolicy**) を使用するように実装を更新することを推奨します。

[Streams for Apache Kafka と MirrorMaker 2 の使用](#) を参照してください。

5.3. KAFKA MIRRORMAKER 1

Kafka MirrorMaker は、データセンター内またはデータセンター全体の 2 台以上の Kafka クラスター間でデータをレプリケーションします。Kafka MirrorMaker 1 は Kafka 3.0.0 で非推奨となり、Kafka 4.0.0 で削除される予定です。MirrorMaker 2 が利用可能な唯一のバージョンになります。MirrorMaker 2 は、クラスター間のデータ転送を管理するコネクタである Kafka Connect フレームワークに基づいています。

その結果、MirrorMaker 1 は Streams for Apache Kafka でも非推奨になりました。MirrorMaker 1 (Streams for Apache Kafka ドキュメントでは **MirrorMaker** と表記) を使用している場合は、MirrorMaker 2 と **IdentityReplicationPolicy** クラスを使用してください。MirrorMaker 2 は、ターゲットクラスターに複製されたトピックの名前を変更します。**IdentityReplicationPolicy** 設定は、名前の自動変更をオーバーライドします。これを使用して、MirrorMaker 1 と同じ active/passive の一方向レプリケーションを作成します。

[Streams for Apache Kafka と MirrorMaker 2 の使用](#) を参照してください。

5.4. KAFKA BRIDGE の SPAN 属性

Kafka Bridge の次の span 属性が非推奨になりました。該当する場合は代替属性を記載しています。

- **http.method** が **http.request.method** に置き換えられました。
- **http.url** が **url.scheme**、**url.path**、および **url.query** に置き換えられました。
- **messaging.destination** が **messaging.destination.name** に置き換えられました。
- **http.status_code** が **http.response.status_code** に置き換えられました。
- **messaging.destination.kind=topic** が置き換えられました。代替属性はありません。

Kafka Bridge は分散トレーシングに OpenTelemetry を使用します。これらの変更は、OpenTelemetry のセマンティック規則の変更に沿ったものです。これらの属性は、Kafka Bridge の今後のリリースで削除される予定です。

第6章 修正された問題

RHEL 上の Streams for Apache Kafka 2.7 で修正された問題です。

Kafka 3.7.0 で修正された問題の詳細は、[Kafka 3.7.0 リリースノート](#)を参照してください。

表6.1 修正された問題

課題番号	説明
ENTMQST-5839	OAuth の問題 (oauth.fallback.username.prefix が反映されない) の修正
ENTMQST-5753	複数の HTTP リクエストで別々の埋め込み形式を使用して生成することが認められない
ENTMQST-5504	KRaft が有効な場合の Kafka および Strimzi のアップグレードのサポートを追加する
ENTMQST-3994	ZooKeeper から KRaft への移行

表6.2 CVE (Common Vulnerabilities and Exposures) の修正

課題番号	説明
ENTMQST-5886	CVE-2023-43642 の不具合が snappy-java の SnappyInputStream で発見される
ENTMQST-5885	CVE-2023-52428 Nimbus JOSE+JWT (9.37.2 より前)
ENTMQST-5884	CVE-2022-4899 の脆弱性が zstd v1.4.10 で発見される
ENTMQST-5883	CVE-2021-24032 の不具合が zstd で発見される
ENTMQST-5882	CVE-2024-23944 Apache ZooKeeper: 永続ウォッチャー処理における情報漏洩
ENTMQST-5881	lz4 の CVE-2021-3520 の不具合
ENTMQST-5835	CVE-2024-29025 netty-codec-http: 制限やスロットリングのないリソースの割り当て
ENTMQST-5646	CVE-2024-1023 vert.x: io.vertx/vertx-core: Vertx での Netty FastThreadLocal データ構造の使用によるメモリーリーク
ENTMQST-5667	CVE-2024-1300 vertx-core: io.vertx:vertx-core: TCP サーバーが TLS および SNI サポートで設定されている場合のメモリーリーク

第7章 既知の問題

このセクションでは、RHEL 上の Streams for Apache Kafka 2.7 に関する既知の問題を説明します。

7.1. RHEL 7 との非互換性

RHEL 7 を Kafka 3.7 と併用する場合、互換性の問題が発生することが確認されています。その結果、RHEL 7 はサポート対象外になりました。この問題は、RHEL 7 上の GCC (GNU Compiler Collection) のバージョンが古く、Kafka 3.7 に必要な RocksDB JNI ライブラリーのバージョン ([org.rocksdb:rocksdbjni:7.9.2](https://github.com/rocksdb/rocksdbjni)) と互換性がないために発生します。

RocksDB JNI バージョン 7.9.2 には、RHEL 7 で使用可能なものよりも新しいバージョンの GCC および関連する **libstdc++** ライブラリーが必要です。RocksDB に依存する Snappy 圧縮と Kafka Streams は、これらの古いライブラリーが原因で、RHEL 7 では正しく機能しません。

推奨設定

- Kafka 3.7 および最新の Streams for Apache Kafka 機能との互換性を確保するために、RHEL 7 で実行されているクライアントとブローカーを RHEL 8 にアップグレードしてください。
- RHEL 7 を引き続き使用する場合は、Streams for Apache Kafka 2.5 LTS または 2.6 の使用を検討してください。

7.2. FIPS モードで実行する場合の JMX 認証

JMX 認証を有効にして FIPS モードで Streams for Apache Kafka を実行すると、クライアントが認証に失敗する可能性があります。この問題を回避するには、FIPS モードでの実行中に JMX 認証を有効にしないでください。私たちはこの問題を調査しており、将来のリリースで解決できるよう取り組んでいます。

第8章 サポートされる構成

Streams for Apache Kafka 2.7 リリースでサポートされる構成です。

8.1. サポート対象のプラットフォーム

以下のプラットフォームは、記載されているバージョンの Red Hat Enterprise Linux (RHEL) 上で Kafka とともに実行される Streams for Apache Kafka 2.7 についてテストされています。

オペレーティングシステム	アーキテクチャー	JVM
RHEL 8 および 9	x86、amd64、ppc64le (IBM Power)、s390x (IBM Z および IBM® LinuxONE)、aarch64 (64 ビット ARM)	Java 11 (非推奨) および Java 17

プラットフォームは Open JDK 11 および 17 でテストされていますが、Java 11 は Streams for Apache Kafka 2.7.0 で非推奨になりました。IBM JDK はサポートされていますが、リリースごとに定期的にテストされていません。Oracle JDK 11 はサポート対象外です。

FIPS コンプライアンス

Streams for Apache Kafka 2.7.0 は FIPS 用に設計されています。

National Institute of Standards and Technology (NIST) によって承認されている RHEL のバージョンを確認するには、NIST Web サイトの [Cryptographic Module Validation Program](#) を参照してください。

8.2. サポートされるクライアント

Streams for Apache Kafka では、Red Hat によって構築されたクライアントライブラリーのみがサポートされます。現在、Streams for Apache Kafka は Java クライアントライブラリーのみを提供していません。

クライアントは Open JDK 11 および 17 でテストされています。

8.3. サポートされる APACHE KAFKA エコシステム

Streams for Apache Kafka では、Apache Software Foundation から直接リリースされた次のコンポーネントのみがサポートされます。

- Apache Kafka Broker
- Apache Kafka Connect
- Apache MirrorMaker
- Apache MirrorMaker 2
- Apache Kafka Java Producer、Consumer、および Management クライアント、Kafka Streams
- Apache ZooKeeper



注記

Apache ZooKeeper は、Apache Kafka の実装の詳細としてのみサポートされており、他の目的のために変更しないでください。さらに、ZooKeeper ノードに割り当てられたコアまたは vCPU は、サブスクリプションコンプライアンスの計算には含まれません。つまり、ZooKeeper ノードは顧客のサブスクリプションにはカウントされません。

8.4. その他のサポートされる機能

- Kafka Bridge
- Cruise Control
- 分散トレーシング

10章 [サポート対象となる Red Hat 製品との統合](#) も併せて参照してください。

8.5. ストレージ要件

Streams for Apache Kafka はブロックストレージでテストされており、Streams for Apache Kafka は、Kafka で一般的に使用される XFS および ext4 ファイルシステムと互換性があります。NFS などのファイルストレージオプションとは互換性がありません。

関連情報

Streams for Apache Kafka 2.5 LTS リリースでサポートされる構成の詳細は、[Streams for Apache Kafka 2.5 リリースノート](#) を参照してください。

第9章 コンポーネントの詳細

次の表に、Streams for Apache Kafka の各リリースのコンポーネントバージョンを示します。

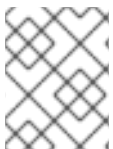


注記

Operator、コンソール、プロキシなどのコンポーネントは、OpenShift 上の Streams for Apache Kafka を使用する場合にのみ該当します。

Streams for Apache Kafka	Apache Kafka	Strimzi Operator	Kafka Bridge	Oauth	Cruise Control	Console	Proxy
2.7.0	3.7.0	0.40.0	0.28	0.15.0	2.5.128	0.1	0.5.1
2.6.0	3.6.0	0.38.0	0.27	0.14.0	2.5.128	-	-
2.5.1	3.5.0	0.36.0	0.26	0.13.0	2.5.123	-	-
2.5.0	3.5.0	0.36.0	0.26	0.13.0	2.5.123	-	-
2.4.0	3.4.0	0.34.0	0.25.0	0.12.0	2.5.112	-	-
2.3.0	3.3.1	0.32.0	0.22.3	0.11.0	2.5.103	-	-
2.2.2	3.2.3	0.29.0	0.21.5	0.10.0	2.5.103	-	-
2.2.1	3.2.3	0.29.0	0.21.5	0.10.0	2.5.103	-	-
2.2.0	3.2.3	0.29.0	0.21.5	0.10.0	2.5.89	-	-
2.1.0	3.1.0	0.28.0	0.21.4	0.10.0	2.5.82	-	-
2.0.1	3.0.0	0.26.0	0.20.3	0.9.0	2.5.73	-	-
2.0.0	3.0.0	0.26.0	0.20.3	0.9.0	2.5.73	-	-
1.8.4	2.8.0	0.24.0	0.20.1	0.8.1	2.5.59	-	-
1.8.0	2.8.0	0.24.0	0.20.1	0.8.1	2.5.59	-	-
1.7.0	2.7.0	0.22.1	0.19.0	0.7.1	2.5.37	-	-
1.6.7	2.6.3	0.20.1	0.19.0	0.6.1	2.5.11	-	-
1.6.6	2.6.3	0.20.1	0.19.0	0.6.1	2.5.11	-	-

Streams for Apache Kafka	Apache Kafka	Strimzi Operator	Kafka Bridge	Oauth	Cruise Control	Console	Proxy
1.6.5	2.6.2	0.20.1	0.19.0	0.6.1	2.5.11	-	-
1.6.4	2.6.2	0.20.1	0.19.0	0.6.1	2.5.11	-	-
1.6.0	2.6.0	0.20.0	0.19.0	0.6.1	2.5.11	-	-
1.5.0	2.5.0	0.18.0	0.16.0	0.5.0	-	-	-
1.4.1	2.4.0	0.17.0	0.15.2	0.3.0	-	-	-
1.4.0	2.4.0	0.17.0	0.15.2	0.3.0	-	-	-
1.3.0	2.3.0	0.14.0	0.14.0	0.1.0	-	-	-
1.2.0	2.2.1	0.12.1	0.12.2	-	-	-	-
1.1.1	2.1.1	0.11.4	-	-	-	-	-
1.1.0	2.1.1	0.11.1	-	-	-	-	-
1.0	2.0.0	0.8.1	-	-	-	-	-



注記

Strimzi 0.26.0 には Log4j の脆弱性が含まれています。製品に含まれるバージョンは、脆弱性が含まれていないバージョンに依存するように更新されました。

第10章 サポート対象となる RED HAT 製品との統合

Streams for Apache Kafka 2.7 は、次の Red Hat 製品との統合をサポートしています。

Red Hat build of Keycloak

OAuth 2.0 認証と OAuth 2.0 認可を提供します。

これらの製品によって Streams for Apache Kafka デプロイメントに導入できる機能の詳細は、製品ドキュメントを参照してください。

10.1. RED HAT BUILD OF KEYCLOAK (旧称 RED HAT SINGLE SIGN-ON)

Streams for Apache Kafka は、Red Hat build of Keycloak の [Authorization Services](#) による OAuth 2.0 トークンベースの認可をサポートしています。これにより、セキュリティーポリシーと権限を一元的に管理できます。



注記

Red Hat build of Keycloak は、現在メンテナンスサポート中の Red Hat Single Sign-On に代わるものです。Red Hat は、この移行を反映するために、ドキュメント、リソース、メディアの更新に取り組んでいます。当面の間、Streams for Apache Kafka ドキュメントの Single Sign-On の使用を説明している内容は、Red Hat build of Keycloak の使用にも適用されます。

関連情報

- [Red Hat build of Keycloak でサポートされる構成](#)
- [Red Hat Single Sign-On でサポートされる構成 \(メンテナンス\)](#)

改訂日時: 2024-07-09