



Red Hat AMQ Streams 2.1

AMQ Streams 2.1 on RHEL リリースノート

Red Hat Enterprise Linux 上の本 AMQ Streams リリースにおける主な新機能と変更
内容

Red Hat AMQ Streams 2.1 AMQ Streams 2.1 on RHEL リリースノート

Red Hat Enterprise Linux 上の本 AMQ Streams リリースにおける主な新機能と変更内容

Enter your first name here. Enter your surname here.

Enter your organisation's name here. Enter your organisational division here.

Enter your email address here.

法律上の通知

Copyright © 2022 | You need to change the HOLDER entity in the en-US/Release_Notes_for_AMQ_Streams_2.1_on_RHEL.ent file |.

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

本リリースノートでは、AMQ Streams 2.1 リリースで導入された新機能、強化された機能、および修正についてまとめています。

目次

多様性を受け入れるオープンソースの強化	3
第1章 機能	4
1.1. KAFKA 3.1.0 のサポート	4
1.2. IBM POWER アーキテクチャーのサポート	4
1.2.1. IBM Power の要件	4
1.2.2. IBM Power ではサポートされていません	5
第2章 機能拡張	6
2.1. KAFKA 3.1.0 で強化された機能	6
2.2. OAUTH 設定オプション	6
第3章 テクノロジーレビュー	7
3.1. KAFKA STATIC QUOTA プラグインの設定	7
3.2. CRUISE CONTROL によるクラスターのリバランス	7
第4章 非推奨の機能	8
4.1. JAVA 8	8
4.2. KAFKA MIRRORMAKER 1	8
4.3. ID レプリケーションポリシー	8
第5章 修正された問題	9
第6章 既知の問題	10
第7章 サポート対象のインテグレーション製品	11
第8章 重要なリンク	12

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。これは大規模な取り組みであるため、これらの変更は今後の複数のリリースで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。

第1章 機能

AMQ Streams 2.1 には、本セクションで説明する機能が導入されています。

AMQ Streams バージョン 2.1 は Strimzi 0.28.x をベースとしています。



注記

本リリースで解決された機能強化とバグをすべて確認するには、[AMQ Streams の Jira プロジェクト](#) を参照してください。

1.1. KAFKA 3.1.0 のサポート

AMQ Streams は Apache Kafka バージョン 3.1.0 に対応するようになりました。

AMQ Streams は Kafka 3.1.0 を使用します。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がサポートされます。

アップグレードの手順は、[AMQ Streams and Kafka upgrades](#) を参照してください。

詳細は、[Kafka 3.0.0](#) および [Kafka 3.1.0](#) のリリースノートを参照してください。



注記

Kafka 3.0.x は、AMQ Streams 2.1 にアップグレードする目的でのみサポートされます。

サポートされているバージョンの詳細については、[AMQ Streams Component Details](#) を参照してください。

Kafka 3.1.0 には、Kafka 3.0.0 と同じバージョンの ZooKeeper バージョン 3.6.3 が必要です。そのため、AMQ Streams 2.0 から AMQ Streams 2.1 にアップグレードする場合、ZooKeeper のアップグレードは必要ありません。



警告

Kafka 3.1.0 では、**自己管理モード** を利用できます。このモードでは、Kafka は Raft プロトコルを使用することで、ZooKeeper を使用せずに実行されます。**AMQ Streams では、自己管理モードはサポートされません。**

1.2. IBM POWER アーキテクチャーのサポート

AMQ Streams 2.1 は、IBM Power **ppc64le** アーキテクチャーで実行できます。

IBM Power のサポートは、Red Hat Enterprise Linux 8 の Kafka 3.0.0 以降および Open JDK 11 で実行している AMQ Streams に適用されます。AMQ Streams 1.8 以前のバージョンの Kafka バージョンに ppc64le バイナリーは **含まれていません**。

1.2.1. IBM Power の要件

- Red Hat Enterprise Linux 8
- Open JDK 11
- Kafka 3.0.0 以降

1.2.2. IBM Power ではサポートされていません

- Red Hat Enterprise Linux 7
- Kafka 2.8.0 以前
- Open JDK 8、Oracle JDK 8 および 11、IBM JDK 8

第2章 機能拡張

AMQ Streams 2.1 では、多くの機能拡張が追加されました。

2.1. KAFKA 3.1.0 で強化された機能

Kafka 3.1.0 に導入された機能拡張の概要は、[Kafka 3.1.0 Release Notes](#) を参照してください。

2.2. OAUTH 設定オプション

新しい OAuth 設定プロパティが OAuth 認証構成に導入されました。

タイムアウトとグループ情報の抽出に関連するプロパティ。

Timeout プロパティ

- **oauth.connect.timeout.seconds** は、タイムアウト前に認証サーバーに接続する最大時間を秒単位で指定します。
- **oauth.read.timeout.seconds** は、タイムアウト前に認証サーバーから読み取る最大時間を秒単位で指定します。

デフォルトは両方とも 60 秒です。

Groups プロパティ

- **oauth.groups.claim** は、JWT トークンまたはイントロスペクションエンドポイント応答からグループ情報を抽出するための Json Path クエリーを指定します。デフォルトでは設定されません。
- **oauth.groups.claim.delimiter** は、単一の区切り文字列として返されるときにグループ情報を解析するための区切り文字を指定します。デフォルト値は「,」(コンマ) です。

Kafka ブローカーリスナーの OAuth 設定例

```
listener.name.client.oauthbearer.sasl.jaas.config=org.apache.kafka.common.security.oauthbearer.OAuthBearerLoginModule required \
# ...
oauth.connect.timeout.seconds=60
oauth.read.timeout.seconds=60
oauth.groups.claim="$groups"
oauth.groups.claim.delimiter=","
```

[リバランスパフォーマンスチューニング](#) を参照してください。

第3章 テクノロジープレビュー



重要

テクノロジープレビューの機能は、Red Hat の実稼働環境のサービスレベルアグリーメント (SLA) ではサポートされず、機能的に完全ではないことがあるため、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。

3.1. KAFKA STATIC QUOTA プラグインの設定

Kafka Static Quota プラグインを使用して、Kafka クラスター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。

3.2. CRUISE CONTROL によるクラスターのリバランス

[Cruise Control](#) をインストールし、これを使用して [最適化ゴール](#) (CPU、ディスク、ネットワーク負荷などに定義された制約) を使用し、Kafka をリバランスできます。バランス調整された Kafka クラスターでは、ワークロードがブローカー Pod 全体に均等に分散されます。

Cruise Control を使用すると、分散された Kafka クラスターを効率的に実行するための時間および労力を削減できます。

Cruise Control の zip 形式のディストリビューションは、[AMQ Streams ソフトウェアダウンロードページ](#) からダウンロードできます。Cruise Control をインストールするには、提供される Metrics Reporter を使用するように各 Kafka ブローカーを設定します。その後、最適化ゴールを含む Cruise Control プロパティーを設定し、提供されたスクリプトを使用して Cruise Control を開始します。

Cruise Control サーバーは、Kafka クラスター全体に対して単一のマシンでホストされます。

Cruise Control が実行されている場合、REST API を使用して以下を行うことができます。

- 複数の最適化ゴールから **ドライラン** 最適化プロポーザルを生成する。
- 最適化プロポーザルを開始し、Kafka クラスターのリバランスを行う。

通知、独自ゴールの作成、トピックレプリケーション係数の変更など、その他の Cruise Control の機能は現在サポートされていません。

[Cruise Control によるクラスターのリバランス](#) を参照してください。

第4章 非推奨の機能

このリリースで非推奨となり、これまでの AMQ Streams リリースではサポートされていた機能は次のとおりです。

4.1. JAVA 8

Java 8 のサポートは、Kafka 3.0.0 および AMQ Streams 2.0 で非推奨になりました。Java 8 は、将来、クライアントを含むすべての AMQ Streams コンポーネントでサポートされなくなります。

AMQ Streams は Java 11 をサポートします。新しいアプリケーションを開発するときは、Java 11 を使用してください。現在 Java 8 を使用しているアプリケーションを Java 11 に移行することを計画します。

4.2. 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 3.1.0 で非推奨となり、Kafka 4.0.0 で削除されます。MirrorMaker 2.0 のみが利用可能なバージョンになります。MirrorMaker 2.0 は Kafka Connect フレームワークをベースとし、コネクタによってクラスター間のデータ転送が管理されます。

そのため、Kafka MirrorMaker 1 のデプロイに使用される AMQ Streams **KafkaMirrorMaker** カスタムリソースが非推奨になりました。Kafka 4.0.0 が導入されると、**KafkaMirrorMaker** リソースは AMQ Streams から削除されます。

MirrorMaker 1 (AMQ Streams ドキュメントでは **MirrorMaker** と呼ばれる) を使用している場合は、**IdentityReplicationPolicy** と **KafkaMirrorMaker2** のカスタムリソースを使用します。MirrorMaker 2.0 では、ターゲットクラスターにレプリケートされたトピックの名前が変更されます。**IdentityReplicationPolicy** 設定は、名前の自動変更を上書きします。これを使用して、MirrorMaker 1 と同じアクティブ/パッシブの一方向レプリケーションを作成します。

[Kafka MirrorMaker 2.0 クラスターの設定](#) を参照してください。

4.3. ID 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 MirrorMaker2 で使用され、リモートトピックの自動名前変更をオーバーライドします。その名前の前にソースクラスターの名前を追加する代わりに、トピックが元の名前を保持します。このオプションの設定は、active/passive バックアップおよびデータ移行に役立ちます。

AMQ Streams Identity Replication Policy class

(**io.strimzi.kafka.connect.mirror.IdentityReplicationPolicy**) は現在非推奨であり、将来削除される予定です。Kafka 独自の ID レプリケーションポリシー (**class org.apache.kafka.connect.mirror.IdentityReplicationPolicy**) に更新できます。

[Kafka MirrorMaker 2.0 クラスターの設定](#) を参照してください。

第5章 修正された問題

AMQ Streams 2.1 on RHEL で修正された問題を以下の表に示します。Kafka 3.1.0 で修正された問題の詳細は、[Kafka 3.1.0 Release Notes](#) を参照してください。

表5.1 CVE (Common Vulnerabilities and Exposures) の修正

課題番号	説明
ENTMQST-2851	CVE-2021-3520 lz4: memmove 引数が原因で、整数オーバーフローのバグによるメモリー破損が発生する
ENTMQST-3631	CVE-2021-43797 netty: ヘッダー名の制御文字が原因で HTTP 要求スマグリングが発生する可能性がある

第6章 既知の問題

AMQ Streams 2.1 on RHEL に既知の問題はありません。

第7章 サポート対象のインテグレーション製品

AMQ Streams 2.1 は、以下の Red Hat 製品とのインテグレーションをサポートします。

Red Hat Single Sign-On

OAuth 2.0 認証と OAuth 2.0 認証を提供します。

これらの製品を使用することで AMQ Streams 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。

関連情報

- [Red Hat Single Sign-On Supported Configurations](#)

第8章 重要なリンク

- <https://access.redhat.com/articles/6644711>
- [AMQ Streams Component Details](#)

改訂日時: 2022-07-12 13:28:27 +1000