



OpenJDK 11

Cryostat 2.0 リリースノート

リリースノート

OpenJDK 11 Cryostat 2.0 リリースノート

リリースノート

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 © 2021 | You need to change the HOLDER entity in the en-US/Release_notes_for_Cryostat_2.0.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

本書では、Cryostat 2 の新機能および概要と、潜在的な既知の問題および回避策を説明します。

目次

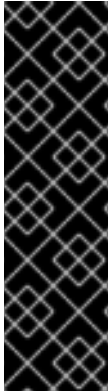
前書き	4
多様性を受け入れるオープンソースの強化	5
RED HAT ドキュメントへのフィードバック	6
第1章 CRYOSTAT 2.0 のサポートポリシー	7
第2章 新機能	8
Cryostat アプリケーションの API パーミッション	8
自動ルール	8
cert-manager API	8
デプロイメントシナリオツリービュー	9
エンドポイントリソースの監視	9
TargetEventGetHandlerでのイベント検索	9
HTTP API の動作変更	10
新しい環境変数	10
新しいハンドラー	10
Cryostat で利用可能な新規イベントテンプレート	10
事前設定された Grafana ダッシュボードイメージ	11
応答ステータスコードメッセージ	11
クライアント側の接続プロトコルのサポート	11
第3章 機能拡張	12
Cryostat Web コンソールの GUI	12
Cryostat Operator トポロジービュー	12
Cryostat Operator コントローラーマネージャー	12
Cryostat クラスタ ConsoleLink namespace	12
カスタムイベントテンプレート	12
サポートされる形式による SSL/TLS 証明書のエンコード	12
Fabric8 Kubernetes および OpenShift クライアント	13
Grafana コンテナのバージョン	13
Grafana エラーメッセージ	13
ヘルスチェックリソースの定義	13
initialization-resource アノテーション	13
OLM バンドル記述子	13
SCC (Security Context Constraints)	13
ServiceRef 定義	13
サブプロセス管理	14
サブプロセスレポートの生成	14
truststore 環境変数	14
TargetConnectionManager ハンドラー	14
Vertx サーバーの応答管理	15
WebSocket	15
第4章 サポートされない機能および非推奨の機能	16
ContainerJFR バリエーション	16
イベント指定子文字列構文	16
対話モードのコマンド	16
TTY/TCP 実行モード	16
WebSocket コマンド	16
WebSever IO クリーンアップ	16

第5章 既知の問題	17
OpenShift SDN デフォルト CNI ネットワークプロバイダー	17
第6章 セキュリティー修正	18
jsoup Java ライブラリー	18

前書き

Cryostat は、OpenShift クラスター内で実行している Java ワークロードをデスクトップの JDK Mission Control アプリケーションに接続するコンテナネイティブな Java アプリケーションです。

Cryostat 2.0 の主な機能は、OpenShift Operator を使用して OpenShift Container Platform Web コンソールの OperatorHub で Cryostat をインストールし、デプロイできることです。Cryostat をインストールおよびデプロイした後に、完全に機能する Cryostat インスタンスにアクセスしてから、**Cryostat 2.0 リリースノート**に記載されている追加機能を確認できます。



重要

本ガイドに記載の機能は、テクノロジープレビューの機能です。テクノロジープレビュー機能は Red Hat の実稼働環境でのサービスレベルアグリーメント (SLA) ではサポートされていないため、Red Hat では実稼働環境での使用を推奨していません。これらの機能は、近々発表予定の製品機能をリリースに先駆けてご提供することにより、お客様は機能性をテストし、開発プロセス中にフィードバックをお寄せいただくことができます。

テクノロジープレビュー機能のサポート範囲については、Red Hat カスタマーポータル内の「[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。

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、今後の複数のリリースで段階的に用語の置き換えを実施して参ります。詳細は、[弊社](#) の CTO、Chris Wright の [メッセージ](#) を参照してください。

RED HAT ドキュメントへのフィードバック

弊社のドキュメントに関するご意見やご感想をお寄せください。フィードバックをお寄せいただくには、ドキュメントのテキストを強調表示し、コメントを追加できます。

本セクションでは、フィードバックの送信方法を説明します。

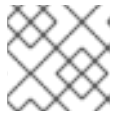
前提条件

- Red Hat カスタマーポータルにログインしている。
- Red Hat カスタマーポータルで、**マルチページ HTML** 形式でドキュメントを表示します。

手順

フィードバックを提供するには、以下の手順を実施します。

1. ドキュメントの右上隅にある **フィードバック** ボタンをクリックして、既存のフィードバックを確認します。



注記

フィードバック機能は、**マルチページ HTML** 形式でのみ有効です。

2. フィードバックを提供するドキュメントのセクションを強調表示します。
3. ハイライトされたテキスト近くに表示される **Add Feedback** ポップアップをクリックします。ページの右側のフィードバックセクションにテキストボックスが表示されます。
4. テキストボックスにフィードバックを入力し、**Submit** をクリックします。ドキュメントに関する問題が作成されます。
5. 問題を表示するには、フィードバックビューで問題トラッカーリンクをクリックします。

第1章 CRYOSTAT 2.0 のサポートポリシー

Red Hat は、最小 6 カ月間、Cryostat のメジャーバージョンをサポートします。Red Hat は、製品が Red Hat カスタマーポータルでリリースされる時期に基づいてこの数値を定めています。

Cryostat は、x86_64 アーキテクチャーで実行している Red Hat OpenShift Container Platform 4.6 以降にインストールおよびデプロイできます。

関連情報

- Cryostat サポートポリシーの詳細は、Red Hat カスタマーポータルの [OpenJDK のライフサイクルおよびサポートポリシー](#) を参照してください。

第2章 新機能

Cryostat 2.0 では、Cryostat 製品の使用を強化する新機能が導入されました。

Cryostat アプリケーションの API パーミッション

Cryostat Operator はサービスアカウントを作成し、続いて Cryostat アプリケーションがサービスアカウントを使用するように設定します。Cryostat Operator はサービスアカウントをチェックして、そのパーミッションが承認チェックによって予想されるものと同期したままになるようにします。

Cryostat 1 では、Cryostat Operator は独自のサービスアカウントのパーミッションを Cryostat アプリケーションに適用します。これにより、Cryostat アプリケーションに過剰なパーミッションが提供されます。

自動ルール

Cryostat 2.0 では、自動化されたルールを使用して、JFR が実行中のターゲットアプリケーションを継続的に監視できるようにすることができます。アプリケーションの再起動または再デプロイは必要ありません。

JVM フラグを設定して常に継続的な監視を有効にすることは可能ですが、この方法にはアプリケーションの再起動が必要になります。

Cryostat 2.0 には、自動化されたルールとテンプレートを作成するのに使用できる継続的な監視イベントテンプレートが含まれます。継続的なモニタリングイベントテンプレートを使用すると、JFR を指定してアプリケーションを継続的に監視するためのダウンタイムを短縮できます。

自動ルールには、以下の自動ルール定義を指定する必要があります。

- 1 回限りのルール定義: JFR 設定用の 1 回限りの自動ルールの簡単な設定を作成できるようになりました。 `eventSpecifier` プロパティには、 `archive` の単純な文字列値が含まれます。
- **matchExpression** 定義: 自動ルールを受け取る必要がある JVM ターゲットを定義する自動ルールの必須コンポーネント。Cryostat は、 **matchExpression** 文字列式で定義された正規表現に一致する JVM ターゲットにルールを適用するのに使用されます。有効な文字列式には、セパレーター、演算子、および簡単な式が含まれます。
 - OpenShift ラベルまたはアノテーションを **matchExpression** 定義の条件として適用できません。

自動ルールを JSON 形式で Cryostat にアップロードすると、Cryostat は定義ファイルを検証します。

以前に定義された一致するターゲットのルール定義をすぐにアクティブにすることができます。ルール定義の作成後にターゲットを再度追加または再起動する必要はありません。

HTTP クエリー `clean=true` パラメーターをルール定義の **DELETE** 要求に追加して、ルール定義を削除できます。このパラメーターは、アクティブな録画のみを削除するように Cryostat に指示します。アーカイブされた記録は削除対象ではありません。

cert-manager API

Cryostat Operator の **Create Cryostat** ページで **Enable cert-manager integration** トグルボタンで、Cryostat アプリケーションの **cert-manager** を使用してエンドツーエンドの暗号化を有効または無効にすることができますようになりました。

図2.1 Enable cert-manager integration トグルボタン

Name *

Labels

Minimal Deployment *

false

Deploy a pared-down Cryostat instance with no Grafana dashboard or jfr-datasource and no web-client UI.

Enable cert-manager Integration

true

Use cert-manager to secure in-cluster communication between Cryostat components. Requires cert-manager to be installed.

Event Templates >

List of Flight Recorder Event Templates to preconfigure in Cryostat

Storage Options >

Options to customize the storage for Flight Recordings and Templates

Trusted TLS Certificates >

List of TLS certificates to trust when connecting to targets

Network Options >

Options to control how the operator exposes the application over a network

さらに、Cryostat Operator は、Cryostat をデプロイする前に **cert-manager** API があることを自動的にチェックするようになりました。これにより、Cryostat Operator が Cryostat を安全にデプロイできるようになります。必要性に応じて、Cryostat Operator **cert-manager** API チェックを無効にするように明示的に設定できます。

Cryostat Operator が **cert-manager** を必要とする Cryostat インスタンスを検出している場合、**cert-manager** をインストールしていない場合は、OpenShift Web コンソールの **Events** ページの Cryostat インスタンスについての警告イベントを受け取ります。

デプロイメントシナリオツリービュー

アプリケーションを OpenShift またはサポートされる Kubernetes ディストリビューションにデプロイした後に、Cryostat の情報を JSON 形式のツリービューに表示できます。ツリービューで以下のコンポーネントを表示できます。

- Pod
- レプリカ
- コンテナ
- エンドポイント

エンドポイントリソースの監視

OpenShiftPlatformClient により、エンドポイントリソース監視がターゲット JVM の非同期ターゲット検出を自動的に設定できるようになります。

TargetEventGetHandler でのイベント検索

TargetEventGetHandler を使用して、ターゲット JVM によって生成されたイベントタイプを一覧表示または検索できます。

HTTP API の動作変更

Cryostat 2.0 よりも前のバージョンでは、Vertx サーバーのハンドラーをインストールするかどうかを決定する際に、デフォルトの **isAsync** メソッド設定を **true** に上書きする可能性があります。

各 HTTP ハンドラーには、Cryostat の HTTP API エンドポイントを定義するコードが含まれます。**isAsync** メソッドは、HTTP ハンドラーが別のワーカースレッドまたはメインの webserver スレッドによって処理されるかどうかを定義します。

新しい環境変数

Cryostat 2.0 には、以下の 2 つの新しい環境変数が含まれています。

- **CRYOSTAT_TARGET_CACHE_MAX_CONNECTIONS**。OpenShift Operator は、オープンな JMX 接続の最大数を指定する値に設定されます。
- **CRYOSTAT_TARGET_CACHE_TTL**。OpenShift オペレーターが、非アクティブから閉じる前に JMX 接続をキャッシュする秒数を指定する値に設定します。



注記

OpenShift Operator はこれらの環境変数の値を自動的に判別するため、これらの環境変数について OpenShift Operator によって指定された値を変更することはできません。

新しいハンドラー

Cryostat 2.0 では、以下の V2 ハンドラーが導入されました。

- **AbstractAuthenticatedRequestHandler** と似ていますが、V2 ハンドラー間で応答全体の応答形式を維持する **AbstractV2RequestHandler**。
- ターゲット JVM でサポートされる録画オプションを確認する **TargetRecordingOptionsListGetHandler**。録画を作成する前にこのハンドラーを使用します。そのため、ターゲット JVM でサポートされていない録画オプションを使用しません。
- カスタマイズされたスナップショットの録画のためにターゲット応答を JSON 形式でエンコードする **TargetSnapshotPostHandler**。この V2 スナップショットハンドラーには、録画の名前、ダウンロード可能な録画ファイルの URL、自動化分析レポートの URL などが含まれます。



注記

V1 ハンドラーは V2 ハンドラーと同様のスナップショット操作を実行できますが、V1 ハンドラーの HTTP 応答はプレーンの文字列形式でのみレンダリングされ、スナップショットの録画の名前のみが含まれます。

Cryostat で利用可能な新規イベントテンプレート

Cryostat Web コンソールの利用可能な **Target JVM** テンプレートの一覧から **cryostat.jfc** イベントテンプレートを選択できるようになりました。

cryostat.jfc イベントテンプレートを選択すると、以下の Cryostat 設定が自動的に有効になります。

- ステータスコード、期間などの HTTP 要求情報。
- 期間、メッセージタイプなどの WebSocket メッセージ情報。

- オープン、クローズ、失敗などの JMX 接続の状態。

事前設定された Grafana ダッシュボードイメージ

Cryostat 2.0 には、アプリケーションのメトリクスを表示するための事前設定された Grafana ダッシュボードイメージが含まれています。必要に応じて、独自の Grafana ダッシュボードイメージを作成することもできます。

Cryostat 1 は、Cryostat コンテナのデプロイメント時に OpenShift Operator によって基本的なカスタマイズを提供する Grafana イメージを提供していました。

応答ステータスコードメッセージ

ClientUrlGetHandler エラーメッセージの応答ステータスコードメッセージが、エラーメッセージの前に表示されるようになりました。これはデフォルト設定です。カスタム例外メッセージは、エラーメッセージの応答ボディ部に残ります。

クライアント側の接続プロトコルのサポート

Cryostat 2 は、**clientlib** ディレクトリーのマウントポイントを追加するようになりました。これにより、追加の JAR ファイルを Cryostat クラスパスに追加できます。これは、JMX RMI の代替として、クライアント定義の接続プロトコルをサポートします。

第3章 機能拡張

Cryostat 2.0 には、Cryostat 1 のオフラインでビルドされる機能拡張が含まれています。

Cryostat Web コンソールの GUI

Cryostat Web コンソールの右上隅にある **Help** アイコンを使用して Cryostat 情報ページにアクセスできます。このコンソールページで、Cryostat のバージョンを表示できます。

また、アーカイブされたレコードの生成された名にターゲットエイリアスが含まれるようになりました。ターゲットエイリアスは、レコードの生成された名前にエイリアスをリンクすることにより、レコードの取得が改善されます。

Cryostat Operator トポロジビュー

Cryostat Operator は適切な **app.openshift.io/connects-to** アノテーションを Cryostat デプロイメントに適用できるようになりました。この設定は、OpenShift Container Platform Web コンソールの topology ビューで Cryostat Operator デプロイメントと一緒に Cryostat デプロイメントをリンクします。

Cryostat Operator コントローラマネージャー

cert-manager をインストールしていないために Cryostat Operator が Cryostat のデプロイに失敗した場合は、Cryostat カスタムリソースの削除時に Cryostat Operator はハングしなくなりました。

Cryostat クラスター ConsoleLink namespace

Cryostat 1 では、Cryostat Operator クラスターは Cryostat クラスターの重複する **ConsoleLink namespace** が作成される可能性があります。この問題を回避するために、Cryostat Operator は **GenerateName** オブジェクトを、**ConsoleLink** リソース定義の **Name** オブジェクトで定義される **SH256** ハッシュ値に置き換えます。

カスタムイベントテンプレート

カスタムイベントテンプレートを使用して、すべての Cryostat JFR 録画を設定できます。カスタムイベントテンプレートは、以下の方法で使用できます。

- Cryostat Operator に、保存された **ConfigMap** オブジェクトのテンプレートファイルを指定して Cryostat カスタムイベントテンプレートを事前設定するよう指示します。
- Cryostat Web アプリケーションを使用してカスタムイベントテンプレートをアップロードします。

サポートされる形式による SSL/TLS 証明書のエンコード

ターゲット JVM アプリケーションの Cryostat Web インターフェースに SSL/TLS 証明書を追加できます。

有効な SSL/TLS 証明書は、DER でエンコードされた **base-64** または **binary** 形式です。どちらの形式も、以下の拡張機能をサポートします。

- **.der**
- **.cer**
- **.pem**

実行時に生成された証明書を指定して、ターゲット JVM アプリケーションが JMX 接続に SSL/TLS を使用できるようになりました。Cryostat は、SSL/TLS 証明書を使用するターゲット JVM への JMX 接続を開くことを試みます。成功した JMX 接続では、Cryostat は実行時に指定したターゲット JVM 証明書に認証チェックをすべて渡す必要があります。

POST ハンドラーを使用して、証明書を許可、検証、および記述できます。

Fabric8 Kubernetes および OpenShift クライアント

Cryostat 2.0 は、Fabric8 Kubernetes および OpenShift クライアントのバージョン 4.12.0 をサポートします。このバージョンでは、Cryostat とのアプリケーションの互換性が強化され、ダウンストリームのビルドエラーを削減します。

Grafana コンテナのバージョン

Cryostat 2.0 は、Grafana コンテナのバージョン 6.4.4 をバージョン 7.3.6 に置き換えます。

Grafana エラーメッセージ

502 エラーメッセージは失敗した JMX 接続に関連し、**512** エラーメッセージは JFR コンテナに送信された無効な応答に関連します。

ヘルスチェックリソースの定義

Cryostat Operator は、**api/v1/clienturl** リソース定義を **health** リソース定義に置き換えます。Cryostat Operator は、コンテナ化された JVM ヘルスチェックの実行時に **health** エンドポイントを使用するようになりました。

initialization-resource アノテーション

Cryostat Operator に、CSV ファイルの設定に **initialization-resource** アノテーションが含まれるようになりました。このアノテーションは、Cryostat クラスターの Cryostat カスタムリソースを作成するためにグラフィカルヒントを提供して OpenShift Container Platform Web コンソールで実行している Cryostat Operator インスタンスを強化します。

OLM バンドル記述子

Cryostat 2.0 の場合、OLM バンドルでは、JFR 録画期間をチェックするために、**eventOptions** 記述子に整数値を設定する必要がなくなりました。代わりに、OLM バンドルは **EventOptions** 文字列で定義される期間形式の値を使用するようになりました。

サポートされる期間単位には、**s**、**m**、および **h** が含まれます。たとえば、**2h15m** は、2 時間 15 分の JFR 録画の長さを示します。

SCC (Security Context Constraints)

Cryostat Operator はデフォルトで **restricted** SCC 設定を使用するようになりました。Cryostat クラスターに含まれる Pod は、**restricted** SCC 設定に一覧表示されている許可された **fsGroup** の値を使用できるようになりました。つまり、Cryostat Pod が永続ストレージ(PV)にマウントされると、そのディレクトリーへの読み取り/書き込みアクセスができるようになりました。

Pod の読み取り/書き込みアクセスレベルは、Pod の **fsGroup** GID の値により異なります。この値は、Cryostat Operator が組み込みの制限付き **SecurityContextConstraint** に準拠するように設定します。**SecurityContext** 要素には、Pod レベルのセキュリティ属性が含まれます。

Cryostat 2.0 より前は、Cryostat Operator が **default** 設定に設定されていたため、Cryostat クラスターで実行する Pod の **fsGroup** アクセスの問題が生まれました。

Cryostat クラスターで許可される **fsGroup** 値の範囲についての詳細は、**OpenShift** ドキュメントの Web サイトの [About pre-allocated security context constraints values](#) を参照してください。

ServiceRef 定義

ServiceRef 定義には、HTTP API ハンドラーを使用した **GET** 通信から返されたすべての **ServiceRef** オブジェクトに含まれる以下の新しいプロパティーが含まれています。

- Java アプリケーション名、ラベル、ポート番号などのアノテーション。
- エイリアスや **connectURL** などのユーザー固有の **ServiceRef** 値。

以下の 2 つのハンドラーを使用して **ServiceRef** 定義を作成または削除できます。

- **POST**: 同じサービス URL を持つ既存のターゲットが存在しない場合に、新規の **ServiceRef** を作成します。
- **DELETE**: **targetID** パスパラメーターで指定された値と一致する場合に、**CustomTargetPlatformClient** から **ServiceRef** 定義を削除します。

サブプロセス管理

Cryostat が自動分析レポートを生成する前に、Cryostat は子サブプロセスを作成します。子サブプロセスは、レポートの生成によって消費される大きなメモリー負荷を累積して親プロセスを保護します。

たとえば、Linux オペレーティングシステムでは、OOM (Out-of-memory) Killer が、システムから利用できない追加のメモリーを要求し、プロセスを停止するプロセスを検出します。子サブプロセスが存在する場合、OOM Killer はこのサブプロセスを停止しますが、実行中の親プロセスには干渉しません。

Cryostat 環境変数 (**CRYOSTAT_REPORT_GENERATION_MAX_HEAP**) を使用して、子サブプロセスの最小または最大 JVM ヒープサイズを設定できます。最小の値が低い場合は、レポートを生成する前に子サブプロセスを停止する可能性があります。maxumim の値が高くなると、親プロセスで追加のメモリー制約が発生する可能性があります。

サブプロセスレポートの生成

Cryostat 2.0 は、**ActiveRecordingReportCache** または **ArchivedRecordingReportCache** のいずれかを使用して Java アプリケーションのサブプロセスレポートを生成する拡張機能を提供します。これらのクラスは以下のように簡素化されます。

- **ActiveRecordingReportCache** には、ディスクの録画へのパスが自動的に含まれるようになりました。
- **ArchivedRecordingReportCache** は、親プロセスの接続マネージャーを使用してレコードをローカルディスクファイルにコピーするユーティリティーメソッドを呼び出します。クラスはローカルパスの場所をサブプロセスに渡します。

サブプロセスでは、以下のタスクを完了する必要がなくなりました。

- JMX 接続を確立します。
- SSL/TLS 証明書を処理します。
- JMX 認証の認証情報を処理します。

これらのタスクを削除すると、以下の方法でサブプロセスのワークロードが改善されます。

- より迅速なエンドツーエンドレポートの生成。
- コードベースの保守が容易になるように、コードの複雑さの軽減。
- SSL/TLS 認証および JMX 認証メカニズムの適切な処理。

truststore 環境変数

Cryostat 2.0 は **TRUSTSTORE** 環境変数名を **SSL_TRUSTSTORE_DIR** に置き換えます。

TargetConnectionManager ハンドラー

TargetConnectionManager は、ターゲット要求と Cryostat 間の接続に JMX 接続にタイムアウトしたキャッシュ値を使用することで、同時接続をサポートするようになりました。デフォルトでは、時間のキャッシュは 90 秒です。

Vertx サーバーの応答管理

Vertx サーバーは **TimeoutHandler** 実装を使用して、Cryostat リクエストハンドラーによって送信される遅延応答メッセージを自動的に終了できるようになりました。これにより、クライアントが Cryostat 要求ハンドラーからの応答を無期限に待機する必要がなくなります。

TimeoutHandler 実装は、Cryostat 要求ハンドラーから遅延応答を検出すると、**500** エラーメッセージを出力します。

WebSocket

WebSocket には以下の更新が含まれます。

- WebSocket 接続が双方向のインタラクティブチャネルから、一方向のプッシュ通知チャネル (NC) にアップグレードしました。
- WebSocket は一方向プッシュ通知にこのチャネルを使用するため、WebSocket 通知チャネルが **api/v1/command** から **api/v1/notifications** に変更になりました。
- WebSocket は、JFR データの分析に録画およびイベントテンプレートを使用すると、通知チャネルでイベントを送信できます。

WebSocket クライアントが一方向プッシュ NC に接続すると、クライアントは同じチャネルで他の接続クライアントが実行したアクションに関する情報を自動的に受信します。

第4章 サポートされない機能および非推奨の機能

Cryostat 2.0 は、メンテナンスコストの高さ、コミュニティの関心の低さ、およびその他の代替ソリューションにより、一部の機能を削除します。

ContainerJFR バリエーション

Cryostat namespace は **ContainerJFR** namespace を置き換えるため、Cryostat 2.0 は **ContainerJFR** namespace のバリエーションを指定しなくなりました。

イベント指定子文字列構文

Cryostat 2.0 には、イベント固有の文字列構文が含まれません。イベントテンプレートは、イベント固有の文字列構文に代わる機能を提供します。

対話モードのコマンド

HTTP API および通知チャンネルは、コマンドチャンネルを置き換えます。Cryostat 2.0 では Command Channel 実装または **CommandExecutor** インターフェースを使用することはできません。

TTY/TCP 実行モード

Cryostat 2.0 には、TTY および TCP ソケット実行モードやモードの設定は含まれません。これらの実行モードは、プラットフォーム認証またはターゲット認証に対応していませんでした。

引き続きサポートされる WebSocket 実行モードを使用できます。

WebSocket コマンド

Cryostat 2.0 は、すべての WebSocket コマンドが削除されました。これらのコマンドは Cryostat 1 リリースで非推奨になりました。

WebSever IO クリーンアップ

Cryostat 2.0 では、組み込み WebSever IO の低メモリ不足モードが削除されました。このモードを使用して、定数メモリーサイズのマッピングを組み込み WebServer IO に適用できなくなりました。

第5章 既知の問題

Cryostat リリースには、Red Hat が承認した問題が含まれる可能性があり、製品の開発中に後の段階で修正される可能性があります。既知の各問題について、説明とその解決を確認してください。

OpenShift SDN デフォルト CNI ネットワークプロバイダー

Cryostat 2.0 は、Cryostat ノードで実行しているノードとは異なるノードにある JVM の以下の設定を持つ OpenShift クラスターを接続できません。

- ソフトウェア定義ネットワーク (SDN) メソッドを、クラスター上の Pod 間の通信を行うための統一クラスターネットワークとして使用します。
- プラグインを作成してコンテナのネットワークインターフェースを設定するために、デフォルトの Container Network Interface (CNI) ネットワークプロバイダーを使用します。

この既知の問題を解決するには、SDN メソッドではなく Open Virtual Network (OVN) メソッドを使用するようにクラスターを設定します。OVN メソッドには、SDN メソッドと同様の設定が含まれます。

- Open vSwitch (OVS) を使用してネットワークトラフィックを管理します。
- CNI ネットワークプロバイダーをデフォルトとして設定するプラグインを使用します。

関連情報

- SND メソッドの詳細は、Red Hat OpenShift ドキュメントの [About the OpenShift SDN default CNI network provider](#) を参照してください。
- OVN 方式の詳細は、Red Hat OpenShift ドキュメントの [About the OVN-Kubernetes default Container Network Interface \(CNI\) network provider](#) を参照してください。

第6章 セキュリティー修正

Cryostat 2.0 には、以前のリリースの Cryostat から提供されるセキュリティーを強化するセキュリティー修正が含まれています。各セキュリティー修正で説明を確認してください。

jsoup Java ライブラリー

Cryostat 2.0 で事前にインストールされている **jsoup** Java ライブラリーは、バージョン 1.13.1 からバージョン 1.14.2 にアップグレードされました。この新しいバージョンは、HTML および XML 構文の **jsoup** Java ライブラリーの処理を強化します。

関連資料

jsoup 1.14.2 リリースに関連するバグ修正の詳細は、**jsoup** ドキュメントの Web サイト [jsoup Java HTML Parser release 1.14.2](#) を参照してください。

改訂日時： 2021-12-05 16:49:26 +1000