



Red Hat build of Cryostat 2

Red Hat build of Cryostat 2.1 のリリースノート

Red Hat build of Cryostat 2 Red Hat build of Cryostat 2.1 のリリースノート

法律上の通知

Copyright © 2023 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

Red Hat build of Cryostat 2.1 のリリースノート には、Cryostat 2.1 の新機能の概要と、潜在的な既知の問題と考えられる回避策のリストが記載されています。

目次

はじめに	3
多様性を受け入れるオープンソースの強化	4
RED HAT ドキュメントへのフィードバック (英語のみ)	5
第1章 CRYOSTAT のサポートポリシー	6
第2章 新機能	7
自動化ルールのユーザーコンソール (UI)	7
JFR レコーディングへのメタデータとラベルの割り当て	7
クライアント側の通知の制御	7
カスタムリソース定義	7
Cryostat Operator の環境変数	8
JMC Agent プラグインのサポート	8
Cryostat 2.1 の Red Hat OpenShift 認証	8
Red Hat OpenShift 認証情報	9
Cryostat 2.1 での Java Management Extensions クレデンシャルの管理	9
新しい自動化ルール環境変数	9
リソース要件	9
サイドカーレポートコンテナ	10
第3章 機能拡張	11
Archives ビュー	11
cert-manager API	11
Create Target ダイアログボックス	11
Cryostat Operator サービスのカスタマイズ	12
Cryostat Operator の詳細	12
Cryostat のファイルダウンロードの動作	13
ファイルのアップロード機能	13
jfr-datasource コンテナ	14
Netty のパフォーマンスリグレッション	14
外部 JVM を使用した Red Hat OpenShift クラスター接続	15
RecordingPostHandler 動作の変更	15
Security メニュー項目	15
setCachedTargetSelect 実装	16
GUI マストヘッドのユーザー名	16
WebSocket API	16
第4章 サポートされない機能および非推奨の機能	17
ContainerJFR バリエーション	17
第5章 既知の問題	18
Cryostat Operator のアップグレード	18
Archived Recordings テーブルのエントリーの削除	18
Cryostat の API 処理の動作	18
重複したファイル名が Archived Recordings テーブルに表示される	20
第6章 このリリースに関連するアドバイザリー	21

はじめに

Red Hat build of Cryostat は、JDK Flight Recorder (JFR) のコンテナネイティブ実装です。これを使用すると、OpenShift Container Platform クラスターで実行されるワークロードで Java 仮想マシン (JVM) のパフォーマンスを安全にモニターできます。Cryostat 2.1 を使用すると、Web コンソールまたは HTTP API を使用して、コンテナ化されたアプリケーション内の JVM の JFR データを起動、停止、取得、アーカイブ、インポート、およびエクスポートできます。

ユースケースに応じて、Cryostat が提供するビルトインツールを使用して、Red Hat OpenShift クラスターに直接レコーディングを保存して分析したり、外部のモニタリングアプリケーションにレコーディングをエクスポートして、レコーディングしたデータをより詳細に分析したりできます。



重要

Red Hat build of Cryostat は、テクノロジープレビュー機能のみです。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 は、実稼働環境でこれらを使用すること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で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。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、Red Hat カスタマーポータルの [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、[Red Hat CTO である 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 (英語のみ)

Red Hat 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。以下の手順に従って、Red Hat ドキュメントのフィードバックをお寄せください。

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、**Multi-page HTML** 形式でドキュメントを表示している。

手順

1. **Feedback** ボタンをクリックして、既存の読者のコメントを表示します。



注記

フィードバック機能は、**Multi-page 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. 選択したテキストの近くに表示されるプロンプトメニューで、**Add Feedback** 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開きます。
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 のサポートポリシー

Red Hat は、Cryostat のメジャーバージョンを最低 6 カ月間サポートします。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される Red Hat OpenShift Container Platform 4.8 以降のバージョンにインストールしてデプロイできます。

関連情報

- Cryostat のライフサイクルポリシーの詳細は、Red Hat OpenShift Container Platform ライフサイクルポリシーの Web ページにある [Red Hat build of Cryostat](#) を参照してください。

第2章 新機能

Cryostat 2.1 では、Cryostat 製品の使用を強化する新機能が導入されています。

自動化ルールのユーザーコンソール (UI)

Cryostat 2.1 には、自動化ルール API 用のユーザーコンソール (UI) が追加されています。UI には次の主要な機能が含まれています。

- フォームビュー。タイプ入力をサポートしており、API とのユーザーの対話を簡素化します。
- 一致式ウィザード。特定のアプリケーションを対象とするカスタム一致式を作成できます。
- 選択したターゲットアプリケーションの JSON 形式のビュー。これにより、アプリケーションの概要情報を確認できます。この情報は、一致式の作成に必要な要素です。
- 色分けされた応答システム。式が選択したターゲットアプリケーションと一致するかどうかを示します。



注記

一致式ウィザードには、カスタム一致式の構文に精通する必要がある場合に参照できるカスタム一致式の例が含まれています。

JFR レコーディングへのメタデータとラベルの割り当て

Cryostat 2.1 で JFR レコーディングを作成する場合、キーと値のラベルのペアを持つメタデータをレコーディングに追加できます。さらに、ターゲット JVM 内にある JFR レコーディングにカスタムラベルを割り当てることができるため、JFR レコーディングを簡単に識別してより適切に管理できます。

メタデータとラベルの使用例としては、レコーディングに対するクエリーの実行やバッチ操作などがあります。Cryostat Web コンソールの **Recordings** メニューに移動して、JFR レコーディングのラベルとそのメタデータを編集できます。アーカイブにアップロードした JFR レコーディングのラベルとメタデータを編集することもできます。

クライアント側の通知の制御

Cryostat 2.1 は、発生する可能性のあるすべてのアクションと状態の変化に関する通知をブロードキャストするため、Cryostat Web クライアントに表示される通知の数が多くなります。その結果、**Settings** ページにクライアント側の通知を制御する機能が追加されました。

Cryostat 2.1 ユーザーは、カテゴリごとに通知を有効または無効にしたり、すべてのグラフィック通知を一括で有効または無効にしたりできます。

Cryostat バックエンドは引き続き通知メッセージを送信し、Web クライアントはそれを受信します。通知を無効にすると、メッセージがコンソールに表示されなくなります。通知を再度有効にすると、以前の通知を閲覧できます。

カスタムリソース定義

Cryostat インスタンスのカスタムターゲットリソース定義を作成できるようになりました。これにより、Cryostat Operator は、Cryostat のデフォルトプロトコル以外の JMX プロトコルを使用してターゲットアプリケーションに接続できます。デフォルトのプロトコルは通常、JMX-RMI です。

カスタムターゲットリソース定義は YAML ファイルで設定されています。定義には次の属性を指定できます。

- **alias**: リソース定義のオプションの名前を設定します。

- **annotation.cryostat**: 定義のオプションのアノテーションを定義します。自動化ルールはこのアノテーションを使用してルールをターゲット JVM に適用できます。
- **connectUrl**: JMX サービス URL や **host:port** ペアなどのターゲット URL を指定します。この URL は、Cryostat がターゲット JVM アプリケーションへの JMX 接続を開くときに使用する必要があります。この属性は必須です。

カスタムターゲットオブジェクトの作成時に、Cryostat Operator はオブジェクトの RESTful API エンドポイント **POST /api/v2/targets** を使用します。オブジェクトが作成されると、REST HTTP API でオブジェクトの **connectUrl** を **targetId** URL パラメーターとして使用できます。

TargetDeleteHandler を使用して、Cryostat Operator からカスタムターゲットリソース定義を削除できます。このハンドラーは、**DELETE/api/v2/targets/:connectUrl** エンドポイント要求を読み取り、Cryostat Operator から定義を削除しようとします。

TargetsPostHandler と **TargetDeleteHandler** の両方に、ハンドラーが要求を処理できない場合に詳細なエラーメッセージを提供するコード化されたエラーメッセージが含まれています。

Cryostat Operator の環境変数

Cryostat 2.1 には、Cryostat Operator の動作を変更するために設定できる次の環境変数が組み込まれています。

- **CRYOSTAT_REPORT_GENERATION_MAX_HEAP**: デフォルトは **200** MiB です。自動化ルール分析レポートを生成するためにコンテナのサブプロセスによって使用される最大ヒープサイズを設定します。
- **CRYOSTAT_MAX_WS_CONNECTIONS**: デフォルトは **unlimited** です。Cryostat アプリケーションがサポートする WebSocket クライアント接続の最大数を設定します。
- **CRYOSTAT_TARGET_CACHE_SIZE**: デフォルトは **-1** です。これはキャッシュが無制限であることを示します。OpenShift Operator が Cryostat アプリケーションにキャッシュできる JMX 接続の最大数を設定します。
- **CRYOSTAT_TARGET_CACHE_TTL**: デフォルトは **10** です。これは、JMX 接続が Cryostat インスタンスのメモリーにキャッシュされる時間 (秒) を示します。

JMC Agent プラグインのサポート

Cryostat 2.1 は、プローブテンプレートを管理する API ハンドラーのセットを使用することにより、JMC Agent プラグインをサポートします。

JMC Agent アプリケーションをインストールし、それをビルドして JAR ファイルを生成すると、JMC Agent プラグインを使用して Cryostat アプリケーションのエージェント機能にアクセスできます。このプラグインは、実行中のアプリケーションに JDK Flight Recorder (JFR) 機能を追加するなど、JMC エージェント機能を Cryostat インスタンスに提供します。

Cryostat 2.1 の Red Hat OpenShift 認証

Cryostat 2.1 は、Red Hat OpenShift のビルトイン OAuth サーバーをそのフレームワークに統合します。有効にすると、Red Hat OpenShift のユーザー名とパスワードを使用して Cryostat にログインできます。この統合された機能は、Cryostat 2.0 で提供されていたメカニズムよりも優れたメカニズムを提供します。Cryostat 2.0 では、Red Hat OpenShift Web コンソールから Red Hat OpenShift 認証トークンを手動でコピーし、コンソールの Cryostat **Application URL** セクションにトークンの詳細を貼り付ける必要がありました。

さらに、Red Hat OpenShift に割り当てられたロールベースアクセス制御 (RBAC) ロールを使用して Cryostat 機能へのアクセスを制限することもできます。

Cryostat 2.1 リリースには、**GET /health** 応答オブジェクトに以下のキーが含まれています。

- **DATASOURCE_CONFIGURED**
- **DASHBOARD_CONFIGURED**
- **REPORTS_CONFIGURED**
- **REPORTS_AVAILABLE**

Red Hat OpenShift 認証情報

Red Hat OpenShift Web コンソールから Cryostat 2.1 Web コンソールにログインすると、Cryostat Operator はセッション期間中に Red Hat OpenShift アカウントにユーザー名とパスワードの認証情報を一時的に保存します。これにより、Cryostat Web コンソールからログアウトする前に Cryostat Web コンソールセッションが終了するのを防ぐことができます。

Cryostat 2.1 での Java Management Extensions クレデンシャルの管理

コンテナ化された Java 仮想マシン (JVM) への認証に使用される Java Management Extension (JMX) クレデンシャルを保存および管理できます。この機能は、Cryostat に複数の JVM のクレデンシャルを記憶させ、再利用する場合に便利です。

JMX クレデンシャルを Cryostat に追加すると、クレデンシャルを表示できなくなります。これにより、Cryostat Web コンソールに入力した後はクレデンシャルが表示されなくなるため、クレデンシャルのセキュリティが確保されます。クレデンシャルを置き換える場合は、クレデンシャルを削除して、再度追加する必要があります。

新しい自動化ルール環境変数

Cryostat 2.1 より前のバージョンでは、自動化ルールの JMX 接続が以前に Cryostat インスタンスのメモリーにキャッシュされていた場合、この JMX 接続は閉じられていました。この問題は、**CRYOSTAT_TARGET_CACHE_MAX_CONNECTIONS** 環境変数の動作が原因で発生しました。

Cryostat 2.1 の JMX キャッシュコンポーネント

は、**CRYOSTAT_TARGET_CACHE_MAX_CONNECTIONS** 環境変数の代わりに

CRYOSTAT_TARGET_CACHE_SIZE 環境変数を使用するようになりました。そのため、開いた JMX 接続が、自動化ルールによって Cryostat インスタンスのメモリーに自動的にキャッシュされなくなります。これにより、自動化ルールが原因でキャッシュストレージ領域がいっぱいになり、使用中の JMX 接続が閉じる問題を防止できます。この問題が発生すると、遅延と応答時間が長くなり、パフォーマンスが低下する可能性があります。

CRYOSTAT_TARGET_CACHE_SIZE 環境変数は、Cryostat インスタンスのメモリーにキャッシュする JMX 接続の最大数を指定します。この環境変数には、次の値を指定できます。

- < 0: デフォルト値は **-1** です。0 未満の値は、キャッシュサイズが無制限であることを示します。この場合、JMX 接続は非アクティブ制限に達した場合にのみメモリーから削除されます。
- 0: ゼロの値は、JMX 接続が閉じられるとすぐにメモリーから削除されることを示します。
- > 0: 0 より大きい値は、指定数の JMX 接続を Cryostat インスタンスのメモリーにキャッシュできることを示します。キャッシュ量がそのレベルに達したときに新しい接続が作成されると、新しい接続の保存を容易にするために、最も古い JMX 接続が閉じられ、メモリーから削除されます。

自動化ルールは、以前にキャッシュされた JMX 接続を再利用できます。JMX 接続が存在しない場合、Cryostat Operator は自動化ルール用の新しい JMX 接続を作成します。この接続はメモリーにキャッシュされません。

リソース要件

デフォルトでは、Cryostat Operator は、Red Hat OpenShift の Cryostat インスタンスのメイン Pod で動作する 3 つのコンテナのそれぞれに対して、リソース要求や制限を指定せずに Cryostat アプリ

ケーションをデプロイします。Cryostat 2.1には、Cryostat カスタムリソース (CR) を使用して、次の3つのコンテナのそれぞれにリソース要求または制限を指定できる機能があります。

- **core**: Cryostat バックエンドサービスと Web アプリケーションを実行します。
- **datasource**: JFR レコーディングを Grafana でサポートされるファイル形式に変換する JFR データソースを実行します。
- **grafana**: Cryostat アプリケーションに関連付けられている Grafana インスタンスを実行します。

サイドカーレポートコンテナ

Cryostat 2.1では、サイドカーレポートコンテナを使用して、JDK Flight Recordings (JFR) の自動分析レポートを生成できます。

Cryostat 2.1 より前のバージョンでは、メインの Cryostat コンテナを使用して分析レポートを生成する必要がありました。このアプローチはリソースを大量に消費し、メインの Cryostat コンテナに追加のリソースをプロビジョニングする必要がある場合があるため、Cryostat アプリケーションの実行パフォーマンスに影響を与える可能性があります。

サイドカーレポートコンテナで分析レポートを生成することにより、Cryostat Operator を効率的に使用して、Cryostat アプリケーションのリソースをプロビジョニングできます。これにより、コンテナと対話する Cryostat Operator が HTTP および JMC 接続を介したオーバーヘッドの少ない操作の実行に集中できるため、Cryostat コンテナのリソースフットプリントが削減されます。

さらに、サイドカーレポートコンテナを複製し、ニーズに合わせてこの複製を設定することもできます。

第3章 機能拡張

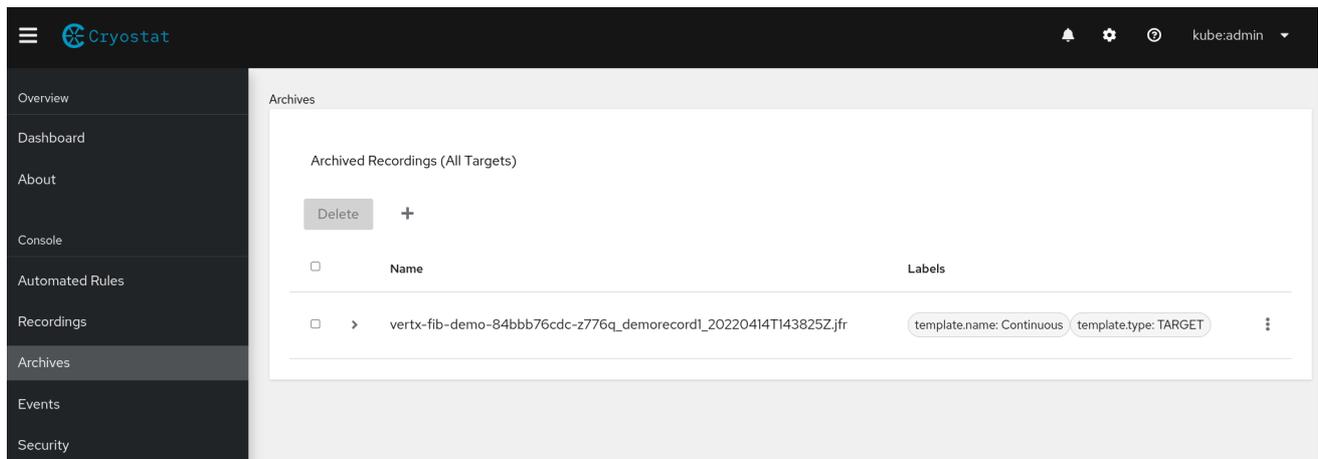
Cryostat 2.1 には、Cryostat 2.0 の機能に基づいて構築された機能拡張が含まれています。

Archives ビュー

Cryostat 2.1 Web コンソールには、**Archives** メニュー項目が含まれています。このメニュー項目を選択すると、**Archives Recording** テーブルがコンソールに表示されます。このテーブルは、Cryostat 2.0 のテーブルを改良したものです。分割ビューを維持しつつ、GraphQL クエリーを使用してテーブルデータを入力するように改良されました。

Archives メニュー項目の **Archived Recordings** のテーブルは、すべてのターゲット JVM のアーカイブを表示するという点で **Recordings** メニューに表示される **Archived Recordings** のテーブルとは異なります。

図3.1 Cryostat Web コンソールの Archives ビュー



cert-manager API

Cryostat 2.1 はバージョン **1.5.3** をサポートしているため、Cryostat Operator は cert-manager API を使用してターゲット JVM の TLS 証明書を設定するようになりました。

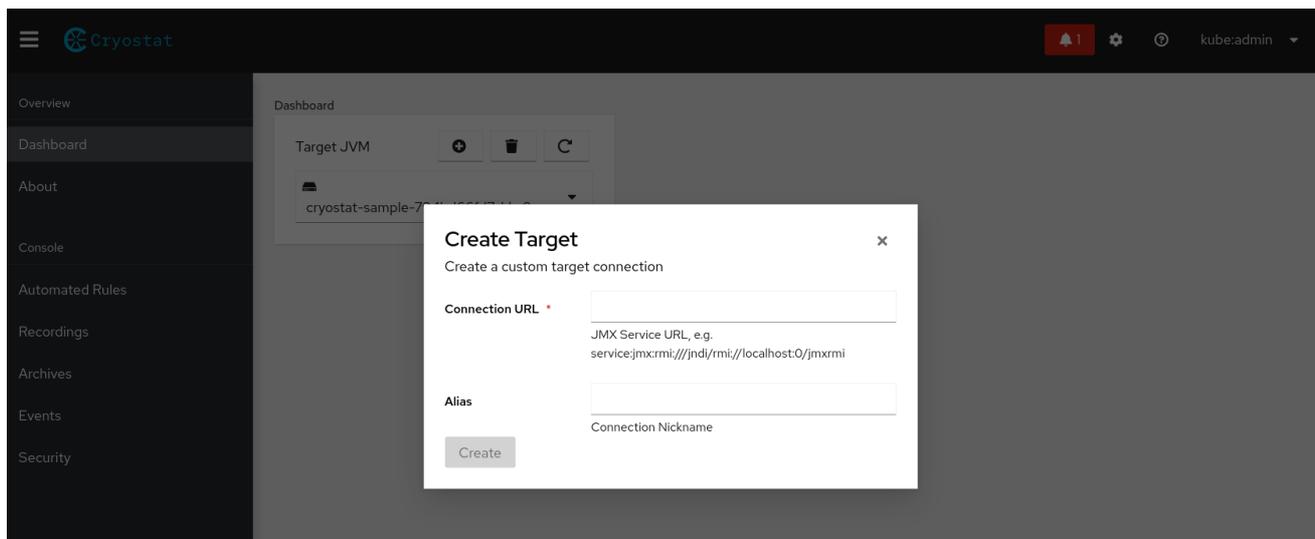
[v1.5.3 \(cert-manager\)](#) を参照してください。

Create Target ダイアログボックス

Cryostat 2.1 では、**Connection URL** フィールドに値を入力するまで、**Create Target** ダイアログボックスの **Create** ボタンが無効になります。

また、**Connection URL** フィールドには、JMX サービス URL の例が含まれています。フィールドに有効な URL を入力する必要がある場合、この URL を参考にすることができます。

図3.2 Dashboard メニュー項目の Create Target ダイアログボックス



Cryostat Operator サービスのカスタマイズ

Cryostat Operator の YAML 設定ファイルに **spec.serviceOptions** プロパティが追加されました。これにより Operator の以下のサービスオプションを変更できるようになりました。

- アノテーション
- ラベル
- ポート番号
- サービスのタイプ

デフォルトのサービスオプション値を変更すると、Cryostat Operator は次のコンポーネントのサービスを作成します。

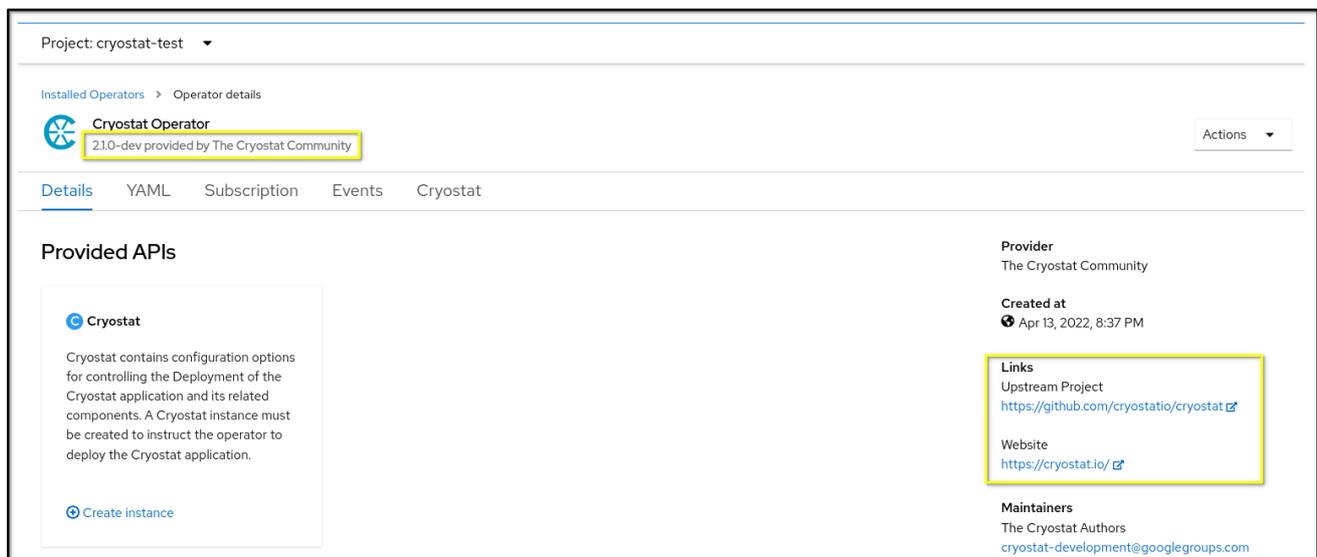
- Cryostat アプリケーション
- Grafana アプリケーション
- ジェネレーターマイクロサービスの報告

Cryostat Operator の詳細

OpenShift Container Platform (OCP) Web コンソールの Cryostat Operator の詳細ページに、次の機能拡張が追加されました。

- Cryostat Operator の名前参照が更新されました。Cryostat 2.1 がリリースされる前は、Cryostat アプリケーションと Cryostat Operator に OCP で類似した名前が付けられていました。
- Cryostat の Web サイトへのリンク

図3.3 Cryostat Operator の詳細ページの新たな機能拡張



Cryostat のファイルダウンロードの動作

Cryostat 2.0 の Web コンソールからファイルをダウンロードする場合 (たとえば、**Active Recordings** オーバーフローメニューから **Download Recording** 項目を選択した場合)、次の手順を完了する必要があります。

1. リモートファイルをデフォルトの Web ブラウザーのメモリーにダウンロードする。
2. Blob ファイル項目のローカルオブジェクト URL を作成する。

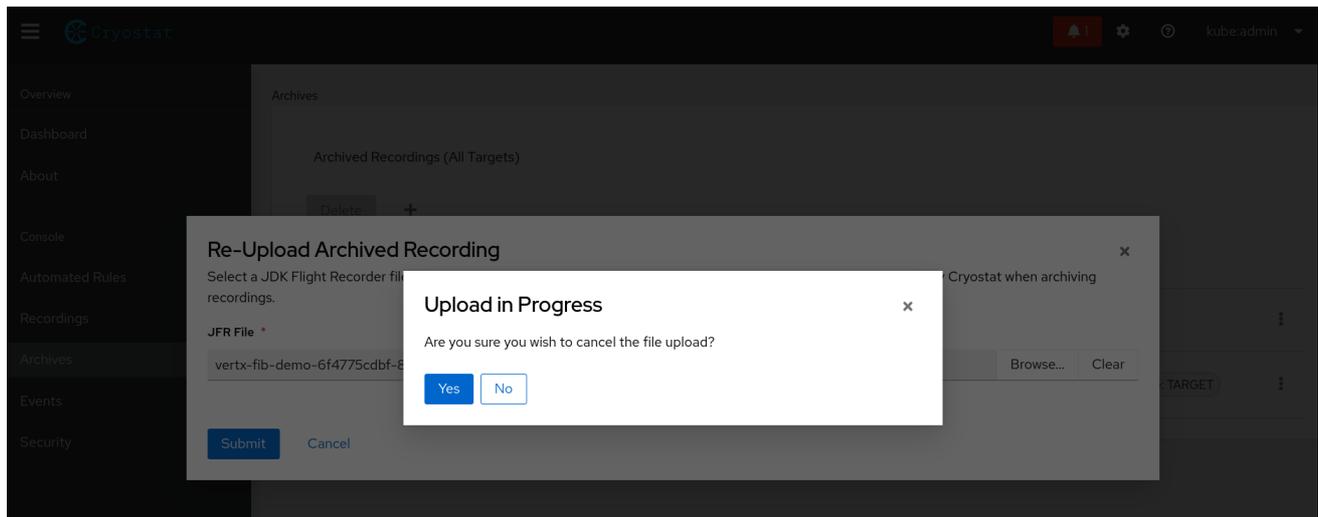
このような動作が原因で、Web ブラウザーのダウンロードメニューの **Cancel** オプションが役に立たなくなっていました。これは、JFR バイナリーファイルのダウンロード操作をキャンセルする場合に問題となることがありました。

Cryostat 2.1 は、Web ブラウザーで使用できる HTML 5 の **download** 属性を利用してファイルのダウンロードを管理します。この属性は、**href** 属性からアンカー要素を読み取り、Web ブラウザーにファイルをダウンロードするように指示します。このようなダウンロード操作により、Web ブラウザーで **Save File** メニューが表示される時間が短縮されるため、ファイルをローカルシステムに保存する前にダウンロード操作をキャンセルできるようになります。

ファイルのアップロード機能

Cryostat 2.1 では、大規模なファイルアップロード操作時に (たとえば、**Re-Upload Archived Recordings** ダイアログボックスからアーカイブされたレコーディングを再アップロードするとき)、**Cancel** ボタンをクリックしてファイルのアップロード操作を停止できます。**Upload in Progress** ダイアログボックスが表示されたら、キャンセル操作を続行するかどうかを選択する必要があります。

図3.4 Re-Upload Archived Recordings ダイアログボックスの Cancel ボタン

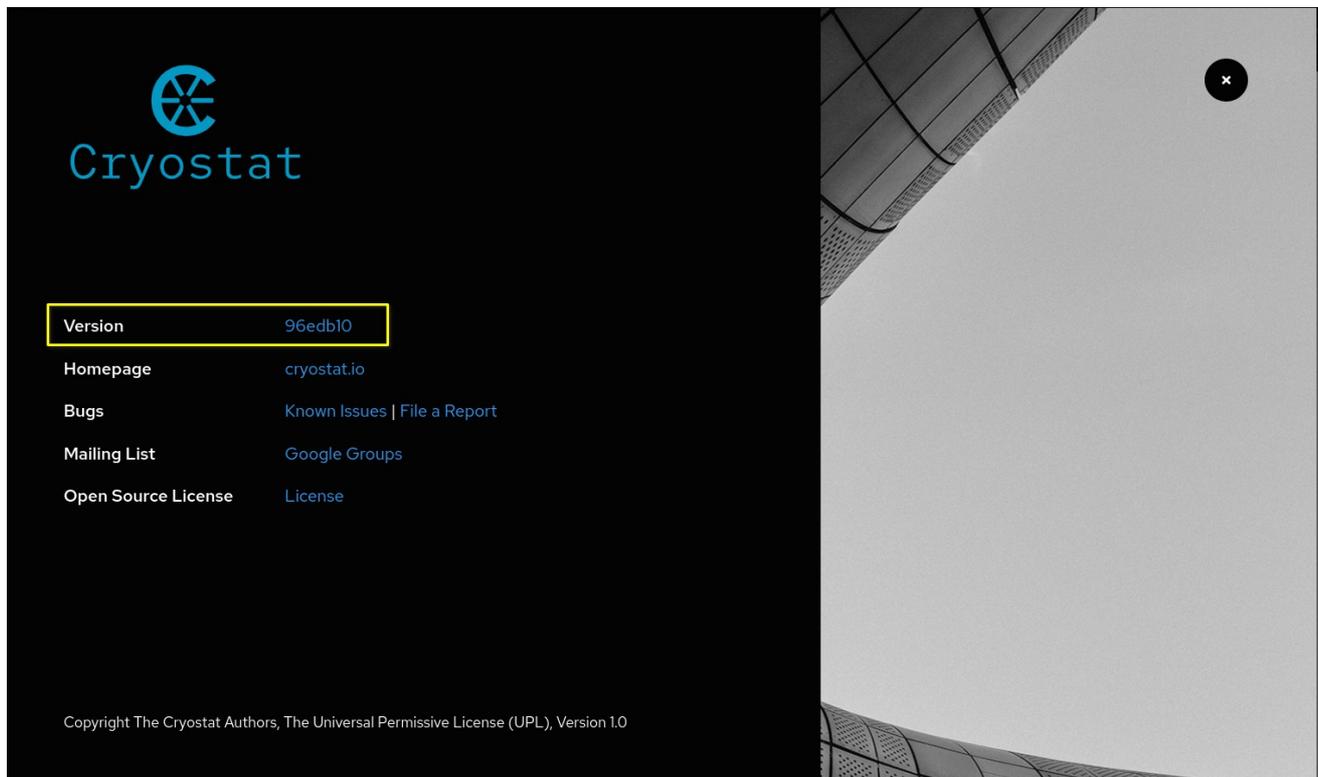


キャンセル操作が完了すると、Cryostat アプリケーションに転送されなかった JFR ファイルのサイズが Web ブラウザーに表示されます。

jfr-datasource コンテナー

Cryostat 2.0 には、Cryostat Web コンソールの **About** ページにバージョン番号が表示されないという問題がありました。Cryostat 2.1 では、**jfr-datasource** または **grafana** ダッシュボードの設定に関係なく、このページにバージョン番号が表示されるようにコードベースを変更することで、この問題を修正しています。

図3.5 Cryostat Web コンソールの About ページ



Netty のパフォーマンスリグレッション

Vert.x **BodyHandler** クラスを使用するハンドラーの実装で、Cryostat 2.0 で明らかだったパフォーマンスの問題が発生しなくなりました。たとえば、ハンドラーが標準の HTTP フォームのアップロードを求めているのに、ファイルのアップロードが許可されるという問題がありました。このファイルのアップ

ロードにより、Vert.xのリソース制約が発生する可能性があります。ハンドラーがそのようなファイルを、Vert.xの一時的なファイルアップロードの場所に永続的に保存する可能性があるためです。さらに、Nettyによる**POST** フォーム本文の解析により、API リクエストの処理中に、メモリー使用量が想定を上回る可能性があります。

Cryostat 2.1はVert.xバージョン**3.9.9**を使用します。このバージョンには、Nettyバージョン**4.1.67**へのアップグレードが含まれます。このアップグレードにより、ハンドラーがVert.xにファイルをアップロードする方法とロジックの両方が改善されました。

外部 JVM を使用した Red Hat OpenShift クラスター接続

Cryostat 2.0には、Cryostat ノードで実行されているノードとは異なるノードにある JVM と Red Hat OpenShift クラスターとの接続に関する既知の問題がありました。

Cryostat 2.1では、この問題は、デフォルトで **false** に設定されている新しい **CRYOSTAT_ENABLE_JDP_BROADCAST** 環境変数によって解決されています。この環境変数のデフォルト設定では、Red Hat OpenShift で Java Discovery Protocol (JDP) が無効になっているため、Cryostat 2.1は任意のノードにある JVM に接続できるようになります。

[既知の問題 \(Cryostat 2.0\)](#) を参照してください。

RecordingPostHandler 動作の変更

Cryostat 2.1では**RecordingPostHandler**実装が強化され、JFR バイナリーが順次解析されるようになりました。Cryostat 2.0の実装では、データを解析してからイベントのリストを構築していました。

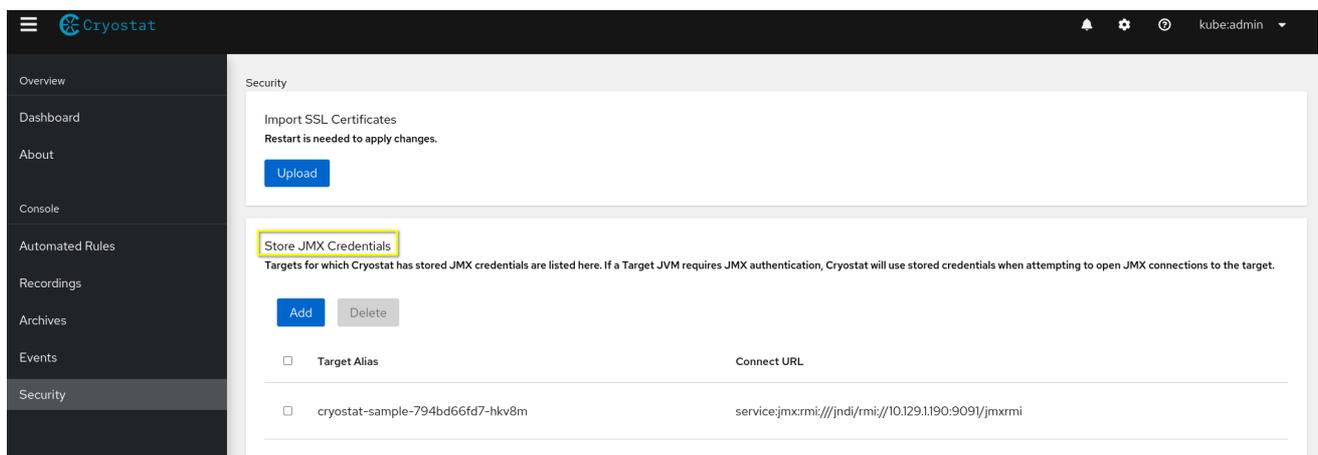
新しい実装には、以下の利点があります。

- シンプルな方法を提供します。
- 実行に必要なリソースは少なくなります。
- アップロードしたデータを以前の動作よりもはるかに高速に検証します。

Security メニュー項目

Cryostat 2.1 インスタンスで **Security** メニュー項目を選択すると、**Store JMX Credentials** タイルにアクセスできます。

図3.6 Security メニュー項目の Store JMX Credentials



Store JMX Credentials タイルは、JMX クレデンシャルが保存されているターゲット JVM を簡単に表示できる便利な方法です。

さらに、このタイル項目では、保存されているクレデンシャルを特定のターゲット JVM に追加できます。JMX 認証を必要とするターゲット JVM には、プロンプトが表示されたらユーザー名とパスワード

を指定する必要があります。Cryostat は、ターゲット JVM への JMX 接続を開くときに保存されたクレデンシャルを使用できます。

setCachedTargetSelect 実装

Cryostat 2.1 リリースより前は、Cryostat Web コンソールにログインして **Dashboard** に移動すると、前のセッションで選択した JVM が **Target JVM** ドロップダウンリストにデフォルト値として表示されていました。これは、Cryostat がこの JVM に接続できなくなった場合でも発生します。

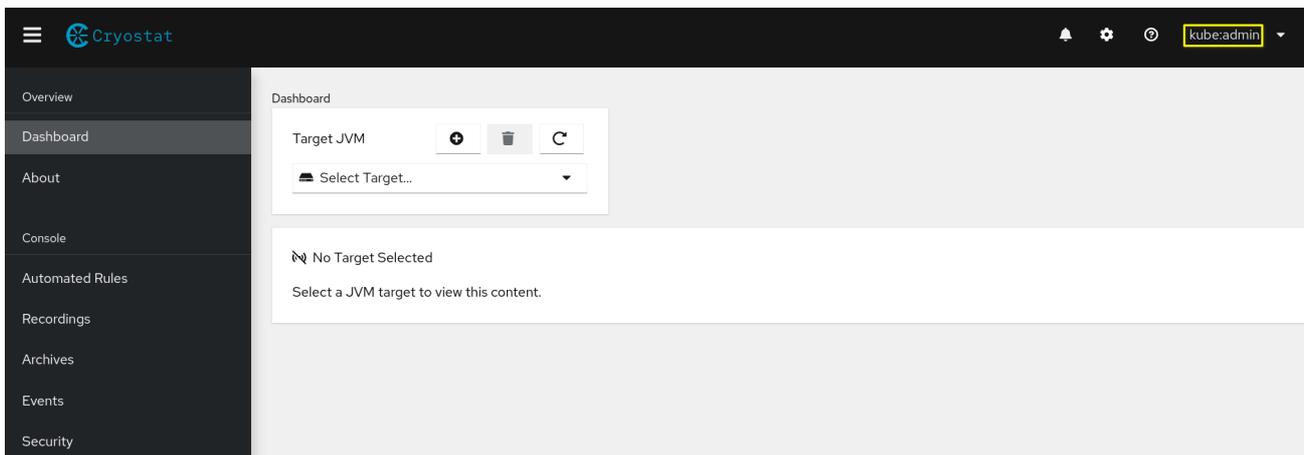
Cryostat 2.1 では、この問題は、新しい各セッションの開始時にターゲット JVM のリストを更新し、接続を確立できる JVM のみをリストすることによって解決されています。

Settings > Auto-Refresh に移動して、Cryostat Web コンソールの更新期間を設定できます。表示されたフィールドに、秒、分、または時間単位で値を指定できます。設定を完了するには、**Enable** チェックボックスを選択する必要があります。

GUI マストヘッドのユーザー名

Cryostat 2.1 は、`/v2.1/auth` エンドポイントからユーザー名を取得するため、Cryostat Web コンソールのマストヘッドにユーザー名を表示できます。Cryostat 2.0 では、Cryostat インスタンスを Basic 認証モードで起動する場合にしかユーザー名を表示できませんでした。

図3.7 Cryostat Web コンソールのマストヘッドに表示されるユーザー名



WebSocket API

Cryostat 2.1 では、WebSocket API が更新され、無制限の WebSocket クライアント接続がサポートされるようになりました。これまでは、WebSocket API は最大 64 クライアント接続しかサポートしませんでした。

Cryostat 2.1 の WebSocket API は、同じ方向のプッシュ通知チャンネル (NC) を使用している接続済みクライアントによって実行されるアクションに関する情報を自動的に受信できます。クライアントの数に制限はありません。

第4章 サポートされない機能および非推奨の機能

Cryostat 2.1では、メンテナンスコストの高さ、コミュニティの関心の低さ、より優れた代替ソリューションにより、一部の機能が削除されています。

ContainerJFR バリエーション

Cryostat 2.1では、**TimeoutHandler** 実装が削除されています。Vertx サーバーは、Cryostat リクエストハンドラーによって送信された遅延応答メッセージを自動的に終了するために、この実装を使用しなくなりました。この応答遅延の問題は、**WebClientAssetsHandler** と **StaticAssetsHandler** によって防止されるようになりました。

第5章 既知の問題

Cryostat のリリースには、Red Hat が認識し、後の製品開発段階で修正される可能性がある問題が含まれている場合があります。既知の各問題の説明と解決策を確認してください。

Cryostat Operator のアップグレード

説明

Cryostat Operator サブスクリプションを Cryostat 2.0 から Cryostat 2.1 に更新する場合、更新チャンネルを **stable-2.0** から **stable** に変更する必要があります。

このアップグレードプロセスを実行すると、Cryostat 2.0 から OpenShift Operator によって作成された一部の Red Hat OpenShift オブジェクトに、古い定義が含まれることとなります。この定義により、Cryostat 2.1 インスタンスと対話するときに Cryostat コンポーネント間の接続の問題が発生する可能性があります。

回避策

Cryostat 2.1 インスタンスに存在する Cryostat 2.0 サービスとデプロイメントを削除します。

Cryostat 2.0 サービスおよびデプロイメントを Cryostat 2.1 インスタンスから削除する例

```
oc project <cryostat_project>
cryostats=$(oc get cryostat --template \
  '{{range .items}}{{.metadata.name}}{\n"}{{end}}')
for cryostat in ${cryostats}; do
  oc delete svc,deploy -lapp="${cryostat}"
done
```

これらの定義を削除すると、Cryostat Operator が Cryostat 2.1 のオブジェクト定義を使用してサービスとデプロイメントを作成します。これにより、Cryostat コンポーネント間の接続が復元され、Cryostat 2.0 によって作成された JFR レコーディングが Cryostat 2.1 に保存されます。

関連情報

- [インストールされているオペレーターのアップグレード \(Red Hat OpenShift 4.8 ドキュメント\)](#) を参照してください。

Archived Recordings テーブルのエントリーの削除

説明

Recordings メニュー項目の **Archived Recordings** テーブルで、削除したアーカイブレコーディングがすぐに削除されません。

回避策

Recordings を除くメニュー項目を選択し、**Recordings** メニュー項目を選択します。**Archived Recordings** テーブルに、利用可能なすべてのアーカイブされたレコーディングが表示されます。

Cryostat の API 処理の動作

説明

Cryostat 2.1 は API リクエストを同時に処理しないため、Cryostat インスタンスのパフォーマンス問題が発生する可能性があります。

以下に示す図の次の記号は、アクションを示しています。

- 平行線 (|): Cryostat が API リクエストを処理している。
- 点線 (.): Cryostat がまだリクエストを処理していない。
- "x": Cryostat がリクエストを処理した。

図5.1 Cryostat による API リクエストの処理方法の例 (現在の動作)



API リクエスト A (低速リクエスト) が実行されている間、クライアントが API リクエスト B を発行し、続いて API リクエスト C (高速リクエスト) を発行します。Cryostat は、API リクエスト A の処理が終了するまで API リクエスト B を処理できません。この動作は、Cryostat が API リクエスト B および C を処理する方法についても同様です。

図5.2 Cryostat による API リクエストの処理方法の例 (必要な動作)



クライアントは異なる間隔で各リクエストを送信します。Cryostat は、低速なリクエストよりも先に高速なリクエストを処理できるため、API リクエスト B および C を API リクエスト A の前に処理します。この動作により、Cryostat は API リクエストを同時に処理し、クライアントに応答を送り返すことができるため、Cryostat のパフォーマンスの問題が軽減されます。

回避策

現在、この問題に対する回避策はありません。

重複したファイル名が Archived Recordings テーブルに表示される

説明

Recordings メニュー項目の **Archived Recordings** テーブルで、テーブル上にすでに存在するファイルと同じファイル名のファイルを再アップロードすると、問題が発生します。

この既知の問題が発生している状態で再アップロードされたファイルを削除または編集しようとすると、Cryostat は両方のファイルに変更を適用します。この変更はファイルの操作方法に影響を与えます。ユーザーが Cryostat を使用して誤ったファイルを分析する可能性があるためです。



注記

この問題は、以下のシナリオでは発生しません。

- 同じターゲット JVM に関連する 2 つのアクティブなレコーディングをすぐにアーカイブした場合。
- 同一のファイルを Cryostat に複数回再アップロードした場合。Cryostat は、再アップロードされた各ファイルを、特定のターゲット JVM に関連しないストレージ場所に保存します。

どちらのシナリオでも、Cryostat は最初のファイルと区別するために番号を 2 番目のファイルに追加します。

以下のディレクトリー構造の例は、選択したターゲット JVM **archive** に対応するストレージ場所にある **20220401T212052Z.jfr** アーカイブレコーディングファイルを示しています。ローカルシステムから、同じ名前が付けられたファイルを Cryostat のアーカイブ場所に再アップロードすると、Cryostat は、どのターゲット JVM にも対応しないストレージ場所 **unlabelled** にファイルを保存します。

```
archive/
├── file-uploads
├──
├── ONSXE5TJMNSTU2TNPA5HE3LJHIXS6L3KNZSGSL3SNVUTULZPMNZHS33TORQXIORZGA4
├── TGL3KNV4HE3LJ
├── ┌── es-andrewazor-demo-Main_foo_20220401T212052Z.jfr
├── └── unlabelled
├──     ├── es-andrewazor-demo-Main_50mb_20211224T172734Z.2.jfr
├──     ├── es-andrewazor-demo-Main_50mb_20211224T172734Z.jfr
├──     └── es-andrewazor-demo-Main_foo_20220401T212052Z.jfr
```

この例の場合、**Archived Recordings** テーブルへの再アップロード操作を行うと、現在の動作では、同じ名前のファイルが 2 つ表示されます。Cryostat は、各ファイルのストレージ場所が異なっていると認識するため、両方のファイルに同じ名前を適用します。

再アップロード操作を行う際に必要な Cryostat の動作は、再アップロードされたファイルに後続の番号を追加することです。そうすることで、このファイルの名前が、テーブルに前から存在するファイルの名前と異なるものになります。

回避策

現在、この問題に対する回避策はありません。

第6章 このリリースに関連するアドバイザリー

このリリースに含まれるバグ修正と CVE 修正に対して、次のアドバイザリーが発行されています。

- [RHSA-2022:1679](#)

改訂日時: 2023-09-19