



## Red Hat build of Cryostat 2

### Red Hat build of Cryostat 2.2 のリリースノート

リリースノート



# Red Hat build of Cryostat 2 Red Hat build of Cryostat 2.2 のリリースノート

---

## 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 © 2023 | You need to change the HOLDER entity in the en-US/Release\_notes\_for\_the\_Red\_Hat\_build\_of\_Cryostat\_2.2.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<sup>®</sup> is the registered trademark of Linus Torvalds in the United States and other countries.

Java<sup>®</sup> is a registered trademark of Oracle and/or its affiliates.

XFS<sup>®</sup> is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL<sup>®</sup> is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js<sup>®</sup> 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<sup>®</sup> 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 2.2 のリリースノート」には、Cryostat 2.2 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
JFR レコーディングの自動アーカイブ .....	7
GraphQL API .....	7
Pluggable Discovery API .....	8
ConfigMap を使用したパーミッションマッピング .....	8
Cryostat CRD のセキュリティーコンテキスト .....	8
JSON での自動化ルール定義のアップロードのサポート .....	8
第3章 機能拡張 .....	9
JMX クレデンシャル変更後の自動化ルールの動作 .....	9
Cryostat Web コンソールの削除プロンプト .....	9
Edit Recording Labels ウィンドウ .....	10
Archives メニューの強化 .....	10
Archive Recordings テーブルの問題を修正 .....	11
レコーディングのフィルタリング .....	11
OpenJDK 17 および Eclipse Vert.x のサポート .....	12
initialDelay 自動化ルールプロパティ .....	13
Archived Recordings のテーブルを更新 .....	13
第4章 サポートされない機能および非推奨の機能 .....	14
OpenShift Operator CRD のサポートを削除 .....	14
第5章 修正された問題 .....	15
Cryostat 2.2.1 で修正された問題 .....	15
JMX セッションの正常な認証 .....	15
設定の変更に伴いデプロイメントの再作成を適用 .....	15
Cryostat 2.2.0 で修正された問題 .....	15
自動分析レポートの結果 .....	15
レコーディングの Create ボタンのアクセスの問題を修正 .....	15



## 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 2.2 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。テクノロジープレビューの機能は、最新の製品機能をいち早く提供して、開発段階で機能のテストを行いフィードバックを提供していただくことを目的としています。

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、[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](#) をご覧ください。



## RED HAT ドキュメントへのフィードバック (英語のみ)

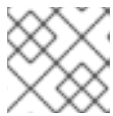
弊社のドキュメントに関するご意見やご感想をお寄せください。フィードバックをお寄せいただくには、ドキュメントのテキストを強調表示し、コメントを追加できます。以下の手順に従って、Red Hat ドキュメントのフィードバックをお寄せください。

### 前提条件

- Red Hat カスタマーポータルにログインする。
- 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、**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. 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を作成しました。
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 2.2 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。

### JFR レコーディングの自動アーカイブ

Cryostat 2.2 では、**Archive on Stop** チェックボックスが **Recordings** メニューに追加されたため、新しい JFR レコーディングを Red Hat OpenShift の永続ボリューム要求 (PVC) ストレージの場所に自動的にアーカイブするように Cryostat を設定できます。デフォルトでは、**Archive on Stop** チェックボックスはオンになっています。**Continuous** チェックボックスをオンにすると、**Archive on Stop** チェックボックスはオフになります。

さらに、CLI で **curl** コマンドを実行して、単一の JFR レコーディングを Red Hat OpenShift の PVC ストレージの場所に自動的にアーカイブするように Cryostat に要求することもできます。たとえば、次の **curl** コマンドは、指定された名前、期間、およびイベントテンプレートを持つ JFR レコーディングファイルを自動的にアーカイブするように Cryostat を指定します。

```
$ curl --data "recordingName=foo2&duration=5&events=template=ALL&archiveOnStop=true" localhost:8181/api/v1/targets/localhost/recordings
```

### GraphQL API

Cryostat 2.2 には GraphQL API エンドポイント **/api/v2.2/graphql** が組み込まれています。これにより、API を使用してターゲット JVM に対してより短く、簡単なクエリーを実行できます。このクエリーは、ターゲット JVM のアクティブなレコーディングとアーカイブされたレコーディングに対しても実行できます。

API は、一般的な Cryostat アーカイブに対してクエリーを実行できます。その後、クエリーをカスタマイズして、アクティブまたはアーカイブされたレコーディングの以下のタスクを自動化できます。

- アーカイブ
- 削除
- 起動
- 停止

Cryostat 2.2 では、次のクエリー機能が導入されています。

- すべてのクエリータイプに適用される新しいフィルターオプション。
- GraphQL API で JFR レコーディングのラベルを更新できます。Cryostat 2.2 より前のバージョンでは、ワークロード機能が制限されている HTTP REST API を使用して、JFR レコーディングのラベルを更新することしかできませんでした。

GraphQL API は、ワークロード機能が制限されている HTTP REST API よりも強力です。たとえば、HTTP REST API では、Red Hat OpenShift のコンテナ内にあるスケールされた各レプリカで起動するレコーディングのコピーごとに API 要求を作成する必要があります。GraphQL API は、1つの API 要求でこのタスクを実現できます。これにより、API のパフォーマンスが向上し、Cryostat インスタンスのネットワークトラフィックを削減できます。

Cryostat 2.2 は HTTP REST API をサポートし、JWT トークンを使用して JFR ファイルをダウンロードする新しいエンドポイントなど、この API の機能拡張をいくつか備えています。ただし、この API には、より多くのユーザー介入が必要です。たとえば、応答データに対して反復操作を実行する場合、API JSON 応答を解析するカスタムクライアントを作成する必要があります。

## Pluggable Discovery API

Cryostat は `/api/v2.2/discovery` エンドポイントをサポートしているため、Pluggable Discovery API を使用して外部クライアントを Cryostat に登録できます。

API は、Cryostat の環境に基づいてクライアントを登録または登録解除できます。



### 注記

Red Hat OpenShift の Cryostat Operator は、Cryostat のサービスアカウントを作成します。Red Hat OpenShift でアプリケーションを作成した後、アプリケーションはサービスアカウントを介して Cryostat の Discovery Plugin API に接続できます。

Cryostat のビルトインターゲット検出メカニズムを無効にすることで、Discovery Plugin API を Cryostat のプライマリーターゲット検出メカニズムとして使用できます。Cryostat のビルトイン検出メカニズムを無効にするには、次の環境変数を **true** に設定します。

```
CRYOSTAT_DISABLE_BUILTIN_DISCOVERY=true
```

Discovery Plugin API には、Cryostat のビルトインターゲット検出メカニズムのすべての機能が含まれています。

### ConfigMap を使用したパーミッションマッピング

プロパティファイルからの特定のパーミッションを含む **ConfigMap** リソースを作成できるようになりました。作成すると、Cryostat Operator が Cryostat インスタンスにパーミッションを適用できるようになります。Cryostat Operator が Cryostat にマウントするデフォルトのパーミッションファイルを使用する必要はなくなりました。

### Cryostat CRD のセキュリティーコンテキスト

Red Hat OpenShift で Cryostat カスタムリソース (CRD) のセキュリティーコンテキストを指定できるようになりました。これにより、Cryostat Operator を設定して、Cryostat ワークロードを Red Hat OpenShift クラスタ全体でより適切に分散できるようになりました。

Cryostat 2.2 より前のバージョンでは、Cryostat Operator はデフォルトのセキュリティーコンテキストを使用していました。これにより、Cryostat ワークロードを Red Hat OpenShift クラスタ全体に分散するときに Cryostat Operator のパーミッションが制限されていました。

### JSON での自動化ルール定義のアップロードのサポート

JSON で自動化ルール定義をアップロードすることにより、自動化ルールのルール定義を再利用できます。この機能は、レコーディングを **Archives** メニューにアップロードする方法と似ています。

Cryostat Web コンソールの **Automated Rules** メニューから、次のタスクを実行できるようになりました。

- 任意のテキストエディターでルール定義ファイルを設定する。またはファイルの追加コピーを作成する。
- ルール定義を無効にする。無効にすると、ルールが JFR レコーディングとやり取りできなくなります。ルール定義は後の段階で有効にすることができます。
- continuous などのルール定義ファイルを JSON 形式でダウンロードする。機能は、**Recordings** メニューからレコーディングをダウンロードする方法と似ています。
- ルール定義を JSON 形式でアップロードする。この機能を使用すると、自動化ルールのルール定義を再利用できます。機能は、レコーディングを **Archives** メニューにアップロードする方法と似ています。

## 第3章 機能拡張

Cryostat 2.2 には、Cryostat 2.1 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。

### JMX クレデンシャル変更後の自動化ルールの動作

Cryostat 2.2 より前のバージョンでは、**Security** メニューでアプリケーションの JMX クレデンシャルを入力する前にターゲット JVM アプリケーションを監視する自動化ルールを作成した場合、自動化ルールは警告なしで失敗していました。

Cryostat 2.2 ではこの問題が解決されています。今回の更新により、自動化ルールを作成した後に、問題なく JMX クレデンシャルを入力できるようになりました。アプリケーションの正しい JMX クレデンシャルを入力すると Cryostat が自動化ルールを再試行するため、自動化ルールを再作成する必要はありません。



#### 注記

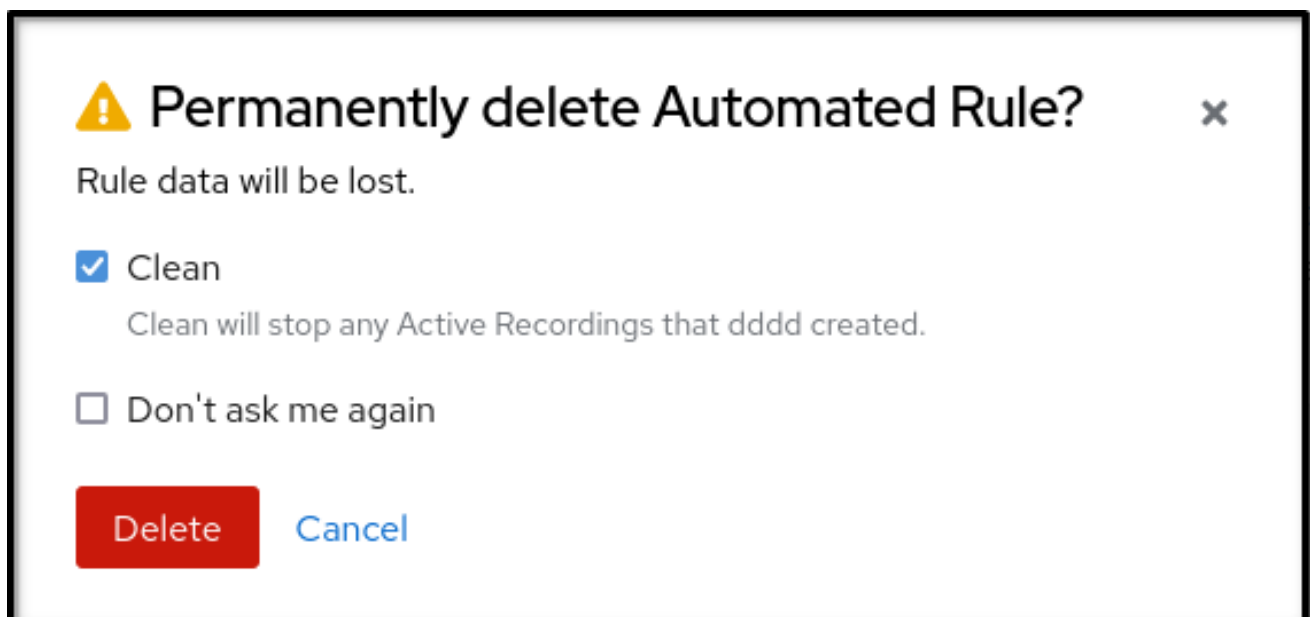
Cryostat 2.2 は、ターゲット JVM アプリケーションの JMX クレデンシャルを暗号化し、Red Hat OpenShift の永続ボリューム要求 (PVC) に格納されているデータベースに保存します。

Cryostat とターゲット JVM アプリケーションの間の接続を確立している間、Cryostat Operator は JMX クレデンシャルをメモリー内に保存します。Cryostat は、JMX クレデンシャルをデータベースに追加する HTTP リクエストと、そのクレデンシャルを使用してターゲットアプリケーションに接続する JMX 接続で SSL/TLS をサポートします。また、Cryostat は、ユーザーによって提供されたパスワードまたは Cryostat Operator によって生成されたパスワードを使用して、データベース内のクレデンシャルを暗号化します。

### Cryostat Web コンソールの削除プロンプト

Cryostat 2.2 では、Cryostat Web コンソールの **Delete** 機能が更新されているため、**Delete** をクリックすると、Web コンソールに次のプロンプトが表示されます。

図3.1 Cryostat Web コンソールで開く Delete プロンプト



Web コンソールで次のいずれかの削除操作を完了すると、このプロンプトにアクセスできます。

- **Automated Rules** メニューから、自動化ルールを削除する。

- **Recordings** メニューまたは **Archives** メニューから、アクティブなレコーディングまたはアーカイブされたレコーディングを削除する。
- **Events** メニューなど、Cryostat Web コンソールのさまざまな場所から、イベントテンプレート、イベントタイプ、またはカスタムターゲットを削除する。
- **Security** メニューから、JMX クレデンシャルを削除する。



### 重要

アクティブなレコーディングまたはアーカイブされたレコーディングのレコーディングラベルを削除しようとしても、プロンプトは開きません。

このプロンプトでは、レコーディングを削除すると、レコーディングに関連付けられたすべてのデータが削除されることが通知されます。

プロンプトには、次のオプションが表示されます。

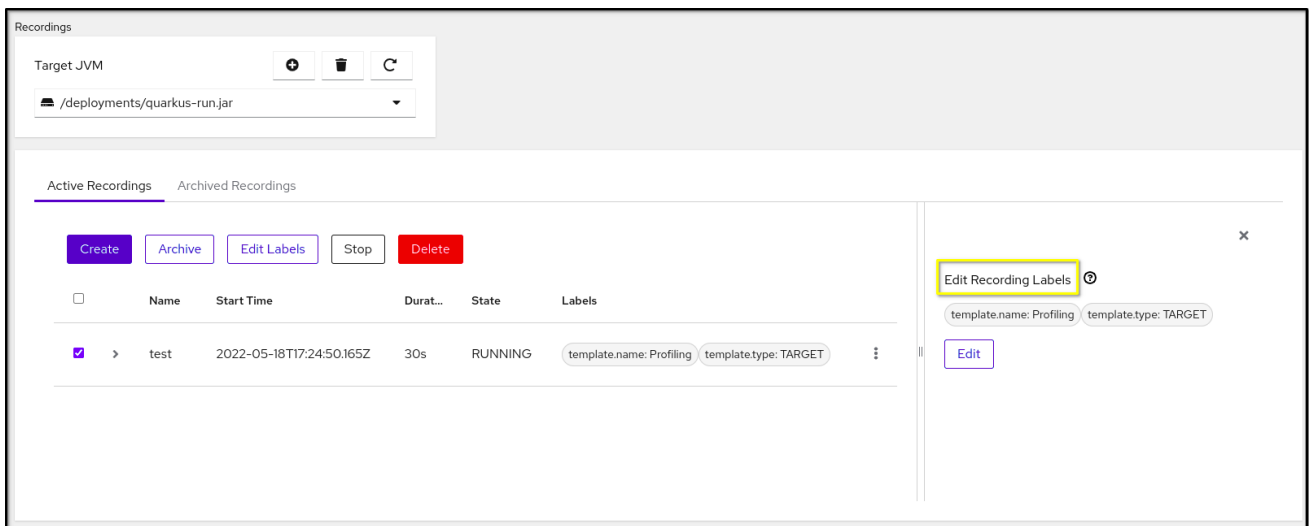
- レコーディングの削除を続行するには、**Delete** をクリックします。
- レコーディングを保持するには、**Cancel** をクリックします。

いずれかのオプションをクリックすると、Cryostat は前のメニューに戻ります。

## Edit Recording Labels ウィンドウ

Cryostat 2.2 では、**Recordings** メニューから **Edit Metadata** オプションが削除され、**Active Recordings** タブと **Archived Recordings** タブの下に **Edit Labels** ボタンが追加されます。

図3.2 Cryostat Web コンソールの Edit Recordings Labels ペイン



**Edit Labels** ボタンをクリックすると、Cryostat Web コンソールで **Editing Recordings Labels** ペインが開きます。

## Archives メニューの強化

**Archives** メニューでは、レコーディングが、ネストされた 3 つのテーブル (**All Targets**、**All Archives**、および **Uploads**) に分けられています。各テーブルには、時系列で結果がリストされます。

レコーディングをアーカイブした後、Cryostat は関連するターゲット JVM を **All Targets** テーブルに表示します。アーカイブされたレコーディングがないターゲット JVM エントリーを削除するには、**Hide targets with zero recordings** チェックボックスをクリックします。

JVM ターゲットエントリーの横にある下矢印アイコン (v) をクリックすると、フィルター機能にアクセスできます。この機能では、ラベルを編集してフィルターを強化したり、**Delete** ボタンをクリックしてフィルターを削除したりできます。

**All Archives** テーブルは **All Targets** テーブルに似ていますが、**All Archives** テーブルには、Cryostat がアーカイブしたファイルからのターゲット JVM アプリケーションがリスト表示されます。

**Uploads** テーブルでは、アップロードされたすべての JFR レコーディングを表示できます。**Uploads** テーブルには、**All Targets** テーブルと同様のフィルタリングメカニズムが組み込まれています。

**Archives** メニューのフィルタリングメカニズムを使用して、認識可能なターゲット JVM アプリケーションを持たない可能性があるアーカイブファイルを見つけることもできます。

### Archive Recordings テーブルの問題を修正

Cryostat 2.2 では、**Archive Recordings** テーブルから重複ファイルの1つが削除されるという Cryostat 2.1 の問題が修正されています。この問題は、次の手順を完了すると発生します。

1. 特定のターゲットに属する JFR レコーディングをアーカイブします。Cryostat は、アーカイブファイルに **my\_recording\_9093\_20220322T172832Z.jfr** などの名前を付けます。
2. 同じ JFR レコーディングを再度アーカイブするか、同じファイルを Cryostat のアーカイブの場所にアップロードします。
3. Cryostat が誤ってファイルの1つを削除することがあります。

次のいずれかの操作を完了すると、ファイルが間違って削除されたことがわかる場合があります。

- 生成された自動分析レポートを表示する。
- Grafana でアプリケーションメトリックを表示する。
- 既存の JFR レコーディングのレコーディングラベルを編集する。

Cryostat 2.2 は、同じ名前の JFR レコーディングファイルを検出しても、各ファイルのターゲットが異なる場合、ファイルの1つを削除しません。この動作は、ターゲット JVM アプリケーションに属するアーカイブファイルと同じ名前のファイルを再アップロードする場合にも適用されます。

問題の原因の詳細は、[重複したファイル名が Archived Recordings テーブルに表示される](#) (Red Hat build of Cryostat 2.1 のリリースノート) を参照してください。

### レコーディングのフィルタリング

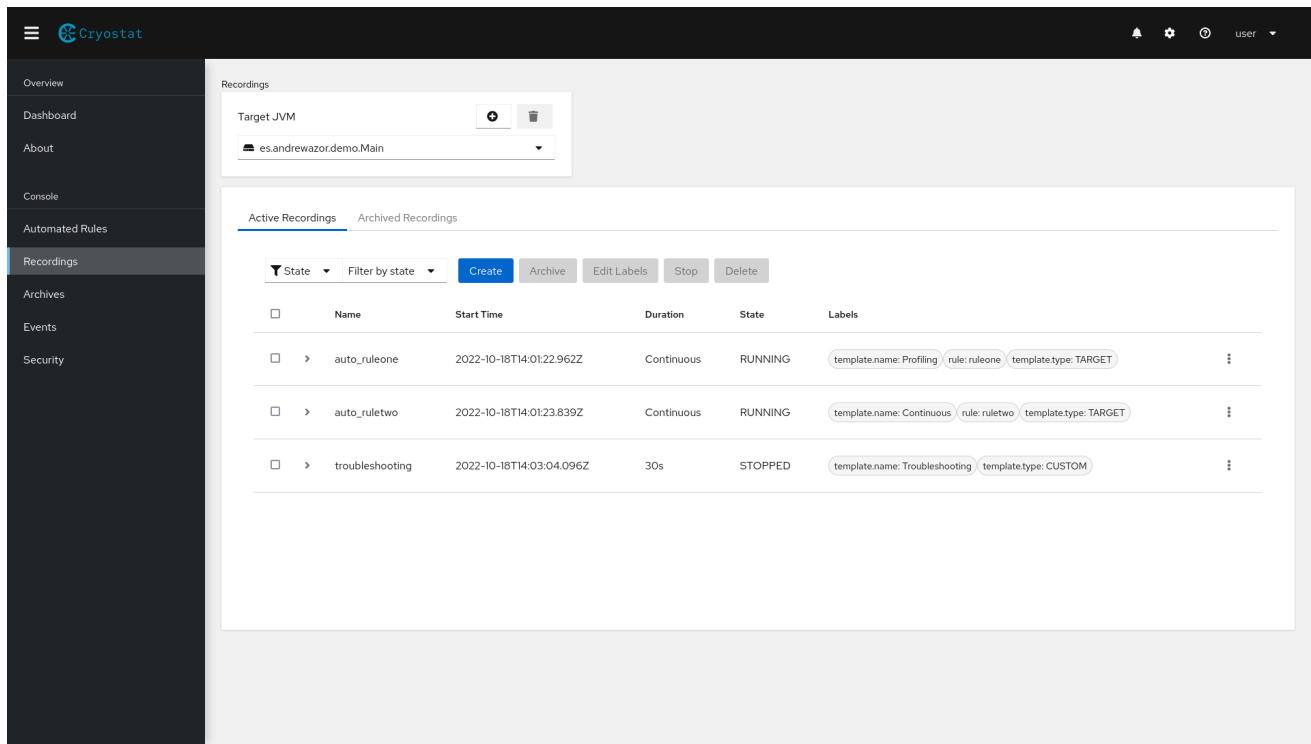
**Recordings** メニューの **Active Recordings** タブまたは **Archived Recordings** タブから、各 JFR レコーディングエントリーの横に表示されるチェックボックスを選択して、リストされている JFR レコーディングをフィルタリングできます。

チェックボックスをクリックすると、Cryostat で **Create**、**Archive**、**Edit Labels** などのボタンが有効になります。

Cryostat のフィルタリング機能を使用すると、ターゲットの JFR レコーディングを正確に検索するフィルターを作成できます。

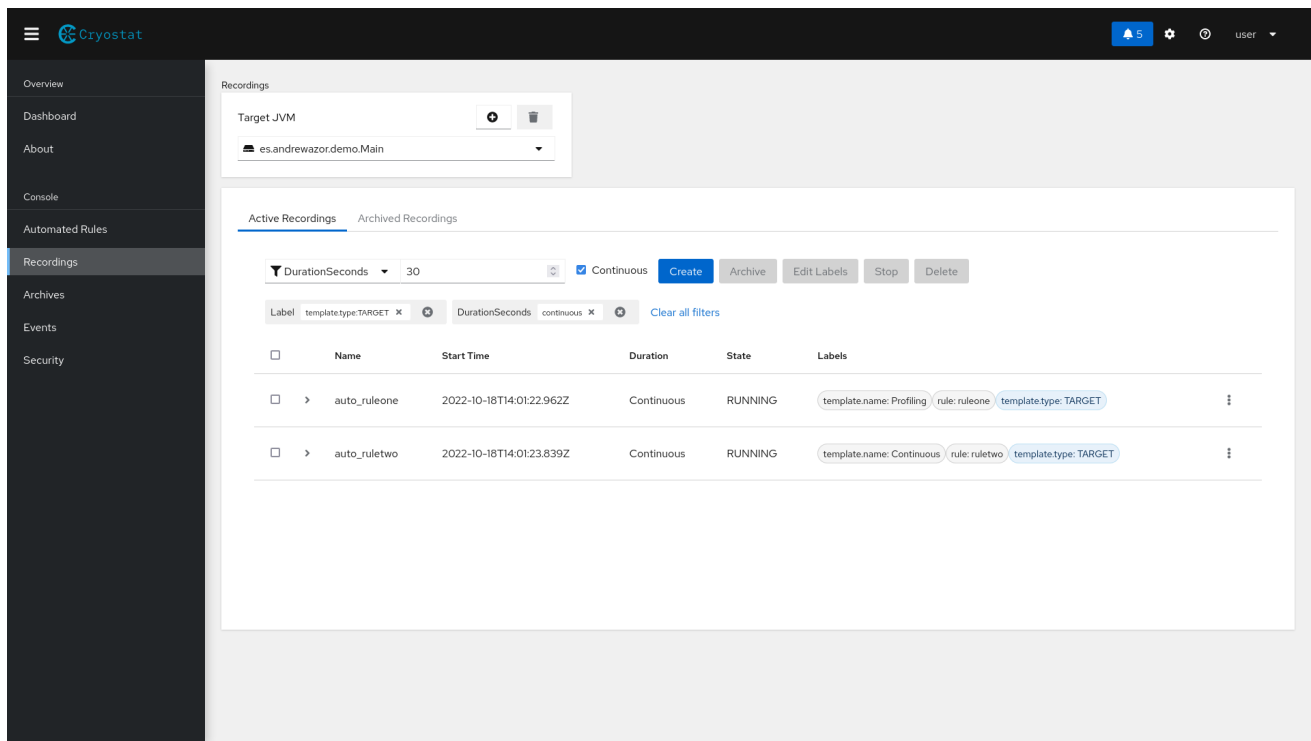
次の図は、3つのアクティブなレコーディングがリストされた **Active Recordings** テーブルを示しています。

図3.3 3つのアクティブなレコーディングがリストされた Active Recordings テーブルの例



次の例は、**template.type:TARGET** ラベルと **DurationSeconds: continuous** ラベルの条件が定義されたフィルターを示しています。フィルタークエリーが完了すると、フィルターのラベル条件に一致する2つの結果が表示されます。

図3.4 Active Recordings タブにある完成したフィルターの例



## OpenJDK 17 および Eclipse Vert.x のサポート

Cryostat 2.2 は、Vert.x 4 フレームワークで構築されています。このフレームワークは、パフォーマンスを向上させ、従来のバグを修正し、Cryostat の新機能を構築します。



さらに、Cryostat 2.2 は OpenJDK 17 で構築されています。この OpenJDK の実装により、パフォーマンスが向上し、Cryostat のメモリー要件が軽減されます。



## 注記

OpenJDK 17 で Cryostat アプリケーションを実行する場合、Cryostat は、OpenJDK 11.0.17 など、JFR テクノロジーをサポートする Cryostat の別のリリースで構築されたターゲット JVM アプリケーションと対話できます。

## initialDelay 自動化ルールプロパティ

Cryostat 2.2 リリースより前のバージョンでは、レコーディングを Cryostat のアーカイブにコピーする自動化ルールを作成した場合、**archivalPeriodSeconds** プロパティの値を指定してスケジュールを作成する必要がありました。この設定では、自動化ルールを使用しても、特定の時間間隔に基づいてレコーディングコピーをアーカイブに移動することしかできません。この **archivalPeriodSeconds** プロパティを使用してアーカイブ期間をずらすことはできません。

Cryostat 2.2 には、ルール定義で定義できる **initialDelay** プロパティが組み込まれています。これを定義すると、自動化ルールにより、必要に応じてアーカイブスケジュールをずらすことができます。たとえば、アプリケーション起動時にレコーディングをすぐにアーカイブする場合などが考えられます。その後、アーカイブ間隔を **30 秒**ごとにスケジュールできます。

## Archived Recordings のテーブルを更新

Cryostat 2.2 は、**Recordings** メニューで開く **Archived Recording** テーブルを更新します。この表には、アーカイブされたファイルのサイズをキロバイト (KB) 単位で表示できる **Size** 列が含まれるようになりました。

さらに、このリリースでは、**Archived Recording** テーブルのスクロールバーを使用して、アーカイブされたファイルをすばやく見つけることができます。スクロールバーの表示は、テーブルの種類ごとに異なります。たとえば、**Active Recordings** テーブルでは、5 つ以上のアクティブな JFR レコーディングファイルがテーブルにリストされているときにスクロールバーにアクセスできます。

## 第4章 サポートされない機能および非推奨の機能

Cryostat 2.2 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。

### OpenShift Operator CRD のサポートを削除

Cryostat 2.2 では、**FlightRecorder** または **Recording** カスタムリソース定義 (CRD) を使用できません。Cryostat には、OpenShift Operator の CRD と同じ機能を提供する HTTP REST API および GraphQL API が組み込まれています。さらに、Cryostat API には、これらの CRD よりも多くの機能が含まれています。

## 第5章 修正された問題

Cryostat のリリースには、Cryostat の以前のリリースで特定された問題の修正が含まれている場合があります。問題の説明とその後の修正については、修正された各問題の注記を確認してください。

### Cryostat 2.2.1 で修正された問題

Cryostat 2.2.1 リリースで、次の問題が修正されました。

#### JMX セッションの正常な認証

Cryostat 2.2.1 より前のバージョンでは、Cryostat を JMX 経由でターゲット JVM アプリケーションに接続し、X-JMX-Authorization ヘッダーを介して JMX クレデンシャルを指定した場合に JMX クレデンシャルに失敗していました。今回の更新で、JMX クレデンシャルの失敗問題が解決され、JMX 経由で Cryostat をターゲット JVM アプリケーションに接続できるようになりました。

#### 設定の変更に伴いデプロイメントの再作成を適用

Cryostat 2.2.1 より前のバージョンでは、Cryostat は設定に加えられた変更に応じてローリング更新を実行できませんでした。この問題は、新しい JMX クレデンシャルデータベースのロックファイルに競合があるために発生しました。今回の更新により、Cryostat はローリング更新を試行せず、代わりに設定に加えられた変更に応じてデプロイメントを再作成するようになりました。

### Cryostat 2.2.0 で修正された問題

Cryostat 2.2.0 リリースで、次の問題が修正されました。

#### 自動分析レポートの結果

Cryostat 2.2.0 より前のバージョンでは、Cryostat は停止した JFR レコーディングを反映するように分析レポートを更新しませんでした。次の一連の手順は、Cryostat の以前のバージョンでこの動作を再現する方法を示しています。

1. JFR レコーディングを開始します。
2. 分析レポートを生成します。
3. JFR のレコーディングを停止します。
4. 生成された最新の分析レポートを表示します。

生成された最新のレポートには、停止したレコーディングではなく、最後のアクティブなレコーディングに基づくスナップショットの結果が表示されます。Cryostat は、停止したレコーディングのデータに基づいてレポートのコピーをキャッシュしません。そのため、最後のアクティブなレコーディングスナップショットに基づくレポートしか表示できません。

Cryostat 2.2.0 では、停止した JFR レコーディングからキャッシュされたデータに関するレポートを生成することで、この問題を修正しています。

#### レコーディングの Create ボタンのアクセスの問題を修正

Cryostat がターゲット JVM との接続を失い、この接続を再確立した後に、Cryostat が Recordings メニューの Create ボタンへのアクセスをブロックしたり、**500 NoRouteToHost** エラーメッセージを出力したりすることはなくなりました。

改訂日時: 2023-01-04 10:34:37 +1000