



Red Hat build of Cryostat 3

Cryostat を使用した JFR レコーディングの管理

Red Hat build of Cryostat 3 Cryostat を使用した JFR レコーディングの管理

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

Red Hat build of Cryostat を使用して、コンテナ化されたアプリケーションにある Java 仮想マシン (JVM) のパフォーマンスをモニターする JDK Flight Recordings (JFR) を管理する方法を説明します。Cryostat 3.0 を使用すると、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。

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 |
| 第1章 セキュリティーオプション | 5 |
| 1.1. 認証情報の保存および管理 | 5 |
| 第2章 JFR レコーディングのアーカイブ | 8 |
| 2.1. JDK FLIGHT RECORDER (JFR) のレコーディングのアーカイブ | 8 |
| 2.2. 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ダウンロード | 10 |
| 2.3. JFR レコーディングを CRYOSTAT のアーカイブ場所にアップロードする | 11 |
| 第3章 イベントテンプレート | 14 |
| 3.1. カスタムイベントテンプレートの使用 | 14 |
| 第4章 統合アプリケーション | 18 |
| 4.1. GRAFANA での JFR レコーディングの表示 | 18 |

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 3.0 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、用語の置き換えは、今後の複数のリリースにわたって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 セキュリティーオプション

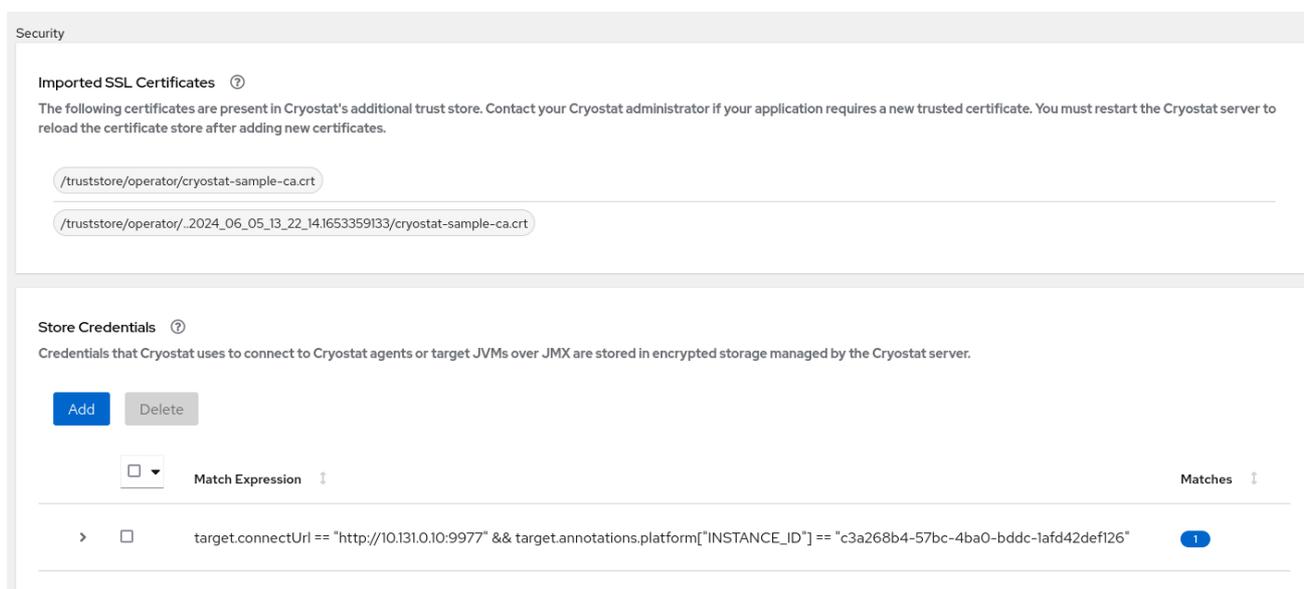
Cryostat のセキュリティー設定を行うと、Cryostat インスタンスをより適切に保護できます。

Cryostat は、ターゲット JVM アプリケーションの認証情報を暗号化し、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できます。Cryostat は、認証情報をデータベースに追加する HTTP リクエストと、その認証情報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 Red Hat build of Cryostat Operator によって生成されたパスワードを使用して、データベース内の認証情報を暗号化します。

Cryostat Operator を使用して、これらの証明書をシークレットに追加し、このシークレットを含めるように Cryostat カスタムリソース (CR) を設定することにより、特定のアプリケーションから SSL/TLS 証明書を信頼するように Cryostat を設定できます。詳細は、[Red Hat build of Cryostat Operator を使用した Cryostat の設定 : TLS 証明書の設定](#) を参照してください。

Cryostat Web コンソールの **Security** メニューをクリックすると、ターゲット JVM のインポートされた SSL/TLS 証明書のリストを表示できます。

図1.1 ターゲット JVM でインポートされた SSL 証明書一覧の表示

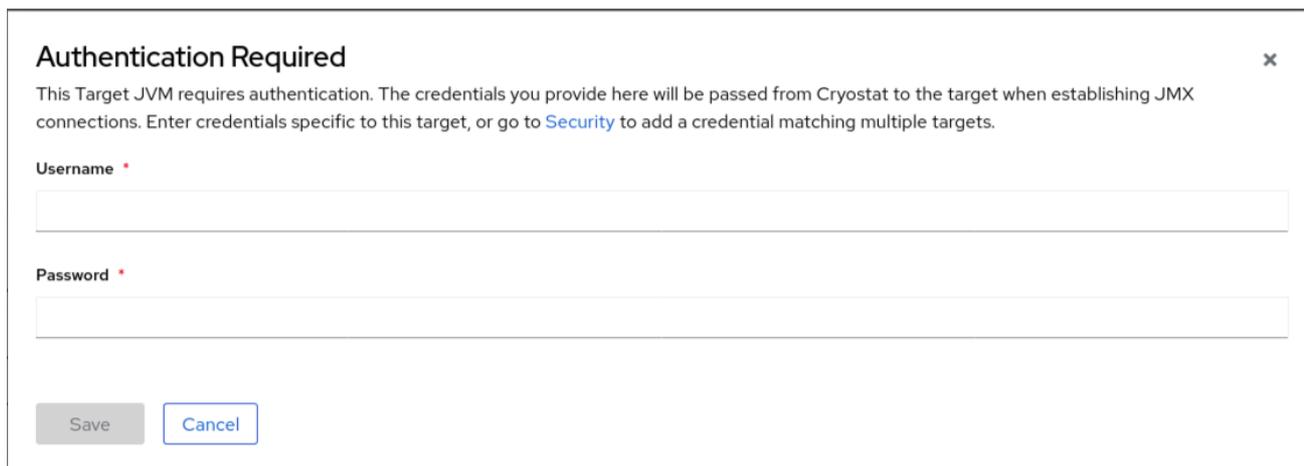


1.1. 認証情報の保存および管理

ターゲット JVM アプリケーションに対して Java Management Extensions (JMX) 認証または HTTP 認証を有効にすると、Cryostat がアプリケーションの JFR レコーディングにアクセスできるように、認証情報の入力を求められます。

Cryostat Web コンソールで **Recordings** または **Events** メニュー項目をクリックすると、コンソールで **Authentication Required** ウィンドウが開きます。ターゲット JVM アプリケーションのユーザー名およびパスワードを入力する必要があります。その後、レコーディングを表示したり、アプリケーションで追加のレコーディング操作を実行したりできます。

図1.2 Cryostat Authentication Required ウィンドウの例



Cryostat は、Cryostat エージェントまたはターゲット JVM への接続に使用する認証情報を保存しません。



重要

ターゲット JVM アプリケーションを再起動する必要がある場合は、アプリケーションの JFR レコーディングデータが失われないように、次のいずれかのタスクを完了してください。

- Cryostat Web コンソールの **Recordings** メニュー項目をクリックし、JFR レコーディングをアーカイブします。
- Cryostat アーカイブのストレージの場所にスナップショットのレコーディングをコピーするように Cryostat をスケジュールする自動化されたルールを作成します。

自動化されたルールを作成して複数のターゲット JVM をモニターする場合は、各ターゲット JVM 接続に対して認証情報を保存して再利用するように Cryostat を設定できます。この設定を使用すると、Cryostat Web コンソールでアプリケーションの JFR レコーディングに再度アクセスするときに、認証情報を再入力する必要がなくなります。

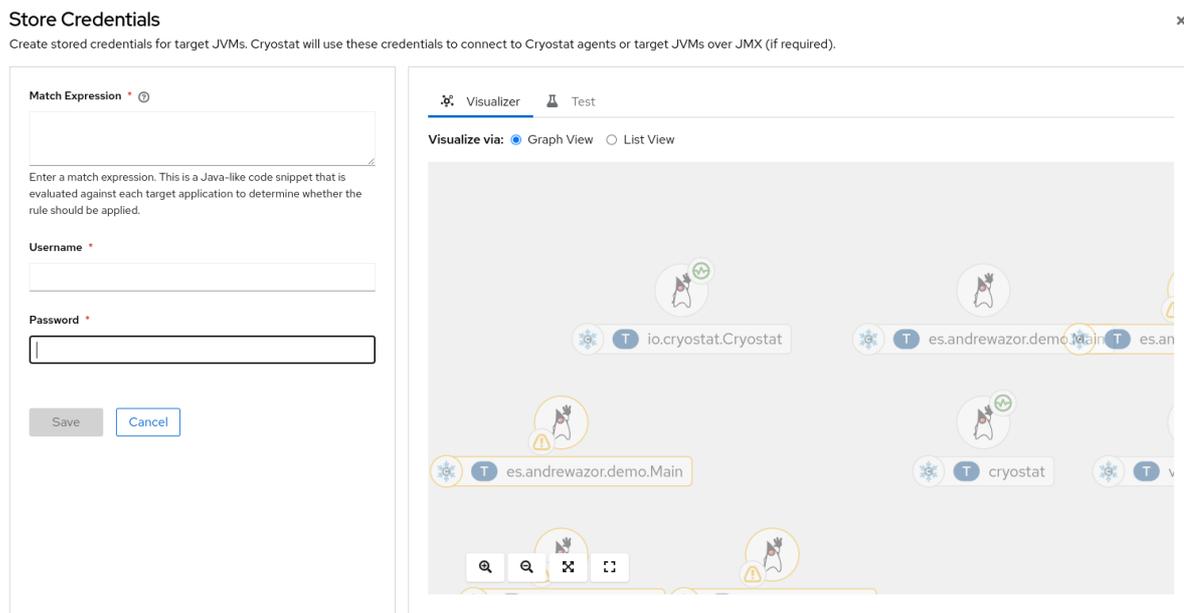
前提条件

- ターゲット JVM アプリケーションの JMX または HTTP 認証を有効にした。

手順

1. **Security** メニュー項目をクリックします。
2. **Store Credentials** ウィンドウで、**Add** ボタンをクリックします。**Store Credentials** ウィンドウが開きます。

図1.3 Store Credentials ウィンドウの例



3. Match Expression フィールドに、一致式の詳細を指定します。



注記

疑問符アイコンを選択すると、Match Expression Hint スニペットに推奨の構文が表示されます。

4. Save をクリックします。Store Credentials ウィンドウにテーブルエントリーが表示され、ターゲット JVM の Match Expression が示されます。

図1.4 Store Credentials ペインのテーブルエントリーの例

Store Credentials ⓘ

Credentials that Cryostat uses to connect to Cryostat agents or target JVMs over JMX are stored in encrypted storage managed by the Cryostat server.

| Match Expression | Matches |
|---|---------|
| > <input type="checkbox"/> target.connectUri == "http://sample-app-38912/" && target.annotations.platform["INSTANCE_ID"] == "fe8320d8-0f91-4b4d-875c-655cb608bec8" | 1 |
| > <input type="checkbox"/> target.connectUri == "http://sample-app-28911/" && target.annotations.platform["INSTANCE_ID"] == "e286c791-8e9c-4427-8a75-51479859c9f7" | 1 |
| > <input type="checkbox"/> target.connectUri == "http://sample-app-18910/" && target.annotations.platform["INSTANCE_ID"] == "1cle77f9-3ad2-4cc3-95f8-1468e14d5442" | 1 |
| > <input type="checkbox"/> target.connectUri == "http://quarkus-test-agent:9977/" && target.annotations.platform["INSTANCE_ID"] == "9e665fa0-4726-4150-afee-29515637194d" | 1 |



重要

セキュリティ上の理由から、テーブルエントリーにはユーザー名やパスワードは表示されません。

5. オプション: ターゲット JVM の保存済み認証情報を削除する場合は、このターゲット JVM のテーブルエントリーの横にあるチェックボックスを選択してから、Delete をクリックします。

第2章 JFR レコーディングのアーカイブ

アクティブな JFR レコーディングをアーカイブして、JFR レコーディングからのデータ損失を防ぐことができます。アーカイブした JFR レコーディングは、ダウンロードまたはアップロードできるため、ニーズに合わせてレコーディングを分析できます。

アーカイブした JFR レコーディングは、**Archives** メニューの **All Targets**、**All Archives**、および **Uploads** の3つの見出しのいずれかの下に時系列で表示されます。JFR レコーディングに対して実行したアクションによっては、レコーディングが各テーブルの下に表示される場合があります。

2.1. JDK FLIGHT RECORDER (JFR) のレコーディングのアーカイブ

アクティブな JFR レコーディングをアーカイブして、JFR レコーディングからのデータ損失を防ぐことができます。Cryostat が古い JFR レコーディングデータを新しいデータに置き換えてストレージ容量を節約したり、ターゲット JVM が突然停止または再起動したりすると、データの損失が発生する可能性があります。

レコーディングのアーカイブを作成すると、Cryostat はアクティブな JFR レコーディングのデータをコピーし、そのデータを Cryostat インスタンスの永続ストレージの場所に保存します。Red Hat build of Cryostat Operator は、この永続ストレージの場所を Red Hat OpenShift クラスタ内の関連付けられた永続ボリューム要求 (PVC) に構築します。

JFR レコーディングは、その設定に関係なくアーカイブできます。さらに、JFR レコーディングからスナップショットをアーカイブすることもできます。

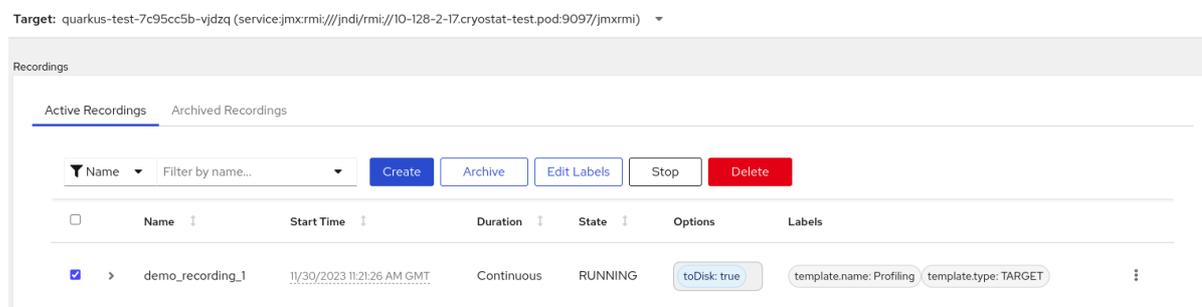
前提条件

- Cryostat インスタンスの認証情報を入力した。
- ターゲット JVM レコーディングを作成し、認証情報を入力しており、**Recordings** メニューにアクセスできる。[JDK Flight Recorder \(JFR\)レコーディングの作成\(Cryostat を使用した JFR レコーディング の作成\)](#)を参照してください。

手順

1. **Active Recordings** タブで、JFR レコーディングのチェックボックスを選択します。**Archive** ボタンは、**Active Recordings** ツールバーで有効になります。

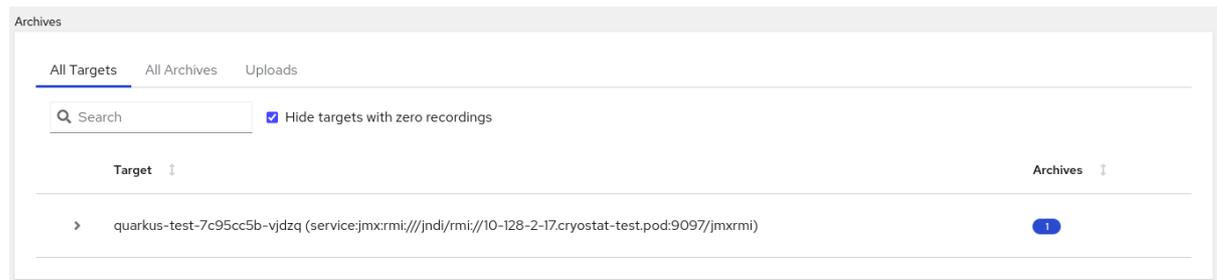
図2.1 JFR レコーディング用のアーカイブボタン



2. **Archive** ボタンをクリックします。Cryostat は、JFR レコーディングのアーカイブを作成します。選択したターゲット JVM に関連するその他のレコーディングとともに、**Archived Recordings** タブの下からアーカイブされたレコーディングを表示できます。

または、**All Targets** テーブルの下からアーカイブされたレコーディングを表示できます。

図2.2 All Targets テーブルの下にリストされたターゲット JVM アプリケーションの例



ヒント

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選択します。

JVM ターゲットエントリーの横にある下矢印 (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。

3. **All Targets** テーブルから、確認する各ターゲット JVM アプリケーションの横にあるチェックボックスを選択します。テーブルには、アーカイブされた各レコーディングとそのソースの場所がリスト表示されます。
4. **All Archives** テーブルに移動します。このテーブルは **All Targets** テーブルに似ていますが、**All Archives** テーブルには、Cryostat が Cryostat 内で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。



注記

アーカイブされたファイルに認識可能な JVM アプリケーションがない場合、そのファイルは引き続き **All Archives** テーブルにリストされますが、見出し **lost** の下のネストされたテーブル内で開きます。

5. **オプション:** アーカイブされたレコーディングを削除するには、特定のアーカイブされた JFR レコーディング項目の横にあるチェックボックスを選択し、プロンプトが表示されたら **Delete** をクリックします。

図2.3 アーカイブされた JFR レコーディングの削除

The screenshot shows the Cryostat Archives page for a target named 'quarkus-test-7c95cc5b-vjdzq'. A recording named 'quarkus-test-7c95cc5b-vjdzq_demo_recording_1_20231130T123324Z.jfr' is selected. The recording details include labels for 'template.name: Profiling' and 'template.type: TARGET', and a size of 703.1 KB. Below the recording details is an 'Automated Analysis' section with various categories and their status:

| Category | Status |
|-----------------------|---|
| DMS | 1 DMS Incidents |
| biased_locking | 1 Biased Locking Revocat... |
| classloading | 1 Class Leak, 1 Class Loading Pressure |
| code_cache | 1 Code Cache |
| environment_variables | 1 Passwords in Environme... |
| exceptions | 1 Thrown Exceptions, 1 Thrown Errors |
| file_io | 1 File Read Peak Duration, 1 File Write Peak Duration |
| garbage_collection | 1 GC Pause Peak Duration, 1 GC Pauses |
| gc_configuration | 1 Compressed Ooops |
| heap | 1 Free Physical Memory, 1 GC Pressure |
| java_application | 1 Application Halts, 1 High JVM CPU Load |
| jvm_information | 1 GC Setup, 1 Stackdepth Setting |
| lock_instances | 1 Context Switches, 1 Java Blocking |
| memoryleak | 1 Heap Live Set Trend |
| method_profiling | 1 Method Profiling |
| processes | 1 Competing CPU Ratio ..., 1 Competing Processes |
| recording | 1 Discouraged Recording ..., 1 Lost Flight Recorder Bu... |
| socket_io | 1 Socket Read Peak Dura..., 1 Socket Write Peak Dura... |

Total size: 703.1 KB



注記

Cryostat は、ターゲット JVM アプリケーションのアドレス、アクティブなレコーディングの名前、作成されたアーカイブ済みレコーディングのタイムスタンプに基づいて、アーカイブされたレコーディングに名前を割り当てます。

関連情報

- [Persistent storage using local volumes \(Red Hat OpenShift\)](#) を参照してください。

2.2. 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ダウンロード

Cryostat を使用して、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ローカルシステムにダウンロードできます。

前提条件

- Cryostat インスタンスの認証情報を入力した。
- JFR レコーディングを作成した。 [JDK Flight Recorder \(JFR\)レコーディングの作成\(Cryostat を使用した JFR レコーディング の作成\)](#)を参照してください。
- オプション: SSL 証明書をアップロードしたか、認証情報をターゲット JVM に提供した。
- オプション: JFR レコーディングをアーカイブした。 [JDK Flight Recorder \(JFR\)レコーディングのアーカイブ\(Cryostat を使用した JFR レコーディング の管理\)](#)を参照してください。

手順

1. Cryostat インスタンスの **Recordings** メニューまたは **Archives** メニューに移動します。

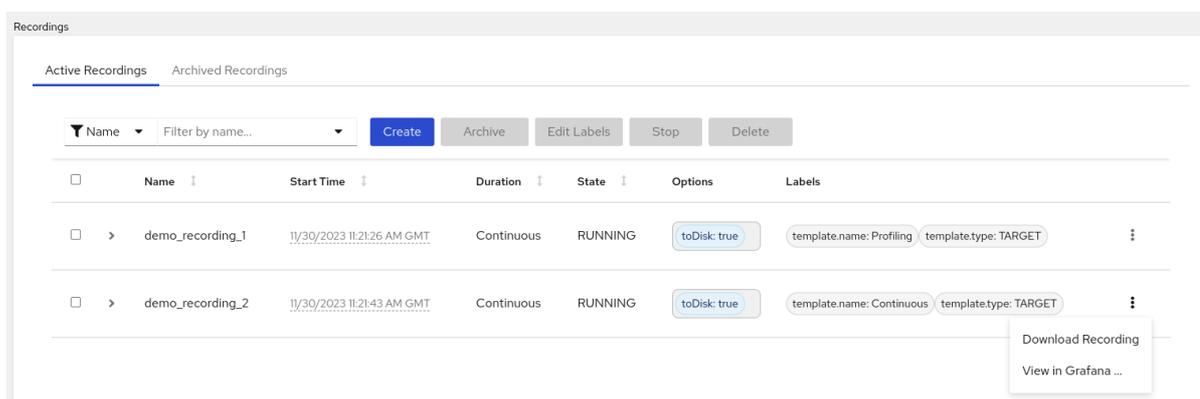


注記

以降の手順では、例として **Recordings** メニューを使用しますが、**Archives** メニューでも同様の手順を実行できます。

2. **Active Record** タブまたは **Archived Recordings** タブをクリックして、必要なレコーディングを決定します。
3. リストされた JFR レコーディングを見つけ、そのオーバーフローメニューを選択します。

図2.4 JFR レコーディングのオーバーフローメニューの表示



4. 以下のいずれかのオプションを選択します。
 - a. オーバーフローメニューから、**Download Recording** をクリックします。オペレーティングシステムの設定に応じて、ファイル保存ダイアログが開きます。JFR バイナリーファイルと JSON ファイルを任意の場所に保存します。
 - b. **All Targets** テーブルから、リストされている JFR レコーディングのオーバーフローメニューを選択します。**Download** をクリックして、アーカイブされたファイルを、メタデータとラベル情報を含む JSON ファイルとともにローカルシステムに保存します。
5. **オプション:** ダウンロードしたファイルを Java Mission Control (JMC) デスクトップアプリケーションで表示します。



注記

.jfr ファイルをダウンロードせずに、Cryostat アプリケーションでレコーディングからのデータを表示する場合は、**View in Grafana** オプションをクリックします。

2.3. JFR レコーディングを CRYOSTAT のアーカイブ場所にアップロードする

ローカルシステムから Cryostat のアーカイブ場所に、JFR レコーディングをアップロードできます。

Cryostat ストレージ領域を節約するために、JFR レコーディングをスケールダウンまたは削除している可能性があります。JFR レコーディングをダウンロードした場合は、インスタンスのスケールアップまたは再デプロイ時に Cryostat インスタンスにアップロードできます。

さらに、以前の Cryostat インスタンスから新しい Cryostat インスタンスにファイルをアップロードすることもできます。Cryostat 分析ツールは、新しい Cryostat インスタンスにアップロードされたレコーディングを使用します。

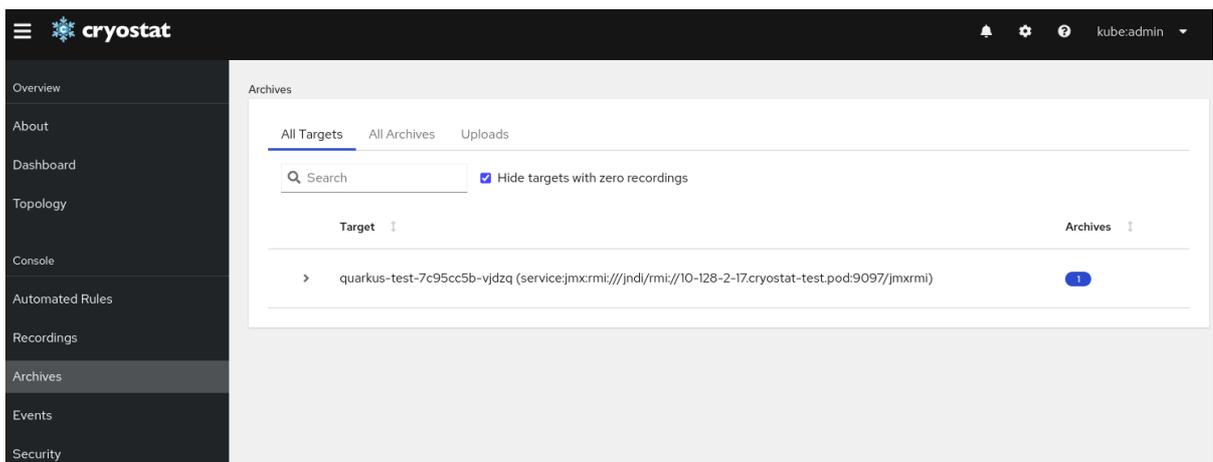
前提条件

- Cryostat インスタンスの認証情報を入力した。
- JFR レコーディングを作成した。[JDK Flight Recorder \(JFR\)レコーディングの作成\(Cryostat を使用した JFR レコーディング の作成\)](#)を参照してください。
- [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 ダウンロード\(Cryostat を使用した JFR レコーディングの管理\)](#)を参照してください。

手順

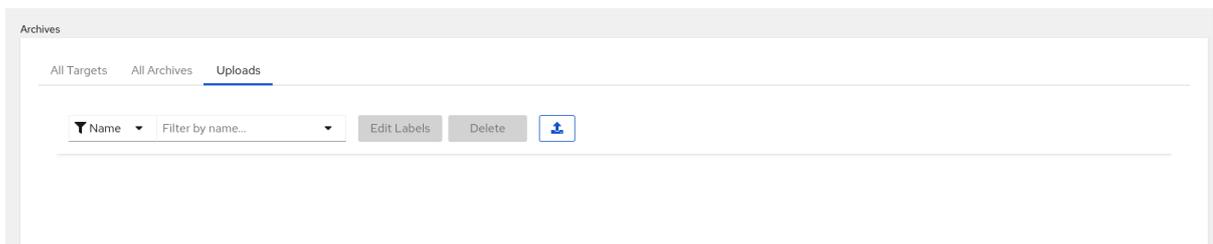
1. Cryostat インスタンスの Archives メニューに移動します。

図2.5 Cryostat Web コンソールの Archives メニュー



2. **オプション:** Uploads テーブルから、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 テーブルやその他の出力など、他のテーブルと同様のフィルタリングメカニズムも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。

図2.6 Archives メニューの Uploads テーブル



3. アップロードアイコンをクリックします。Re-Upload Archived Recording ウィンドウが、Cryostat Web インターフェイスに表示されます。

図2.7 Re-Upload Archived Recording ウィンドウ

4. **JFR File** フィールドで、**Upload** をクリックします。
5. 拡張子が **.jfr** のファイルである JFR レコーディングファイルを探し、**Submit** をクリックします。



注記

または、**.jfr** ファイルを **JFR File** フィールドにドラッグアンドドロップすることもできます。

JFR レコーディングファイルが **Uploads** テーブルで開きます。

図2.8 Uploads テーブルにある JFR レコーディングの例

| <input type="checkbox"/> | Name | Labels | Size |
|--------------------------|---|--------|-----------|
| <input type="checkbox"/> | > quarkus-test-agent_default_20230606T085613Z.jfr | - | 368.77 KB |

第3章 イベントテンプレート

Cryostat には、デフォルトのイベントテンプレートが含まれています。これを使用すると、ターゲット JVM のパフォーマンスをモニターするための JFR レコーディングをすばやく作成できます。

3.1. カスタムイベントテンプレートの使用

JDK Flight Recorder (JFR) レコーディングの作成時に、以下のいずれかのデフォルトイベントテンプレートを選択できます。

- 一定期間、または明示的に停止されるまで、ターゲット Java 仮想マシン (JVM) の基本的なデータを収集する継続テンプレート。
- 一定期間、または明示的に停止されるまで、ターゲット Java 仮想マシン (JVM) の詳細なデータを収集するプロファイリングテンプレート。

どちらかのデフォルトイベントテンプレートを使用して、ターゲット JVM のパフォーマンスをモニターするための JFR レコーディングを迅速に作成できます。どちらのイベントテンプレートも、必要に応じて後で編集できます。たとえば、デフォルトイベントテンプレートにはアプリケーション固有のカスタムイベントが含まれないため、これらのカスタムイベントをカスタムテンプレートに追加する必要があります。

Cryostat は **ALL** meta-template もサポートします。これを使用すると、JFR はターゲット JVM のすべてのイベントタイプをモニターできます。各イベントタイプにはデフォルト値があります。**ALL** meta-template には XML 定義が含まれないため、**ALL** meta-template の XML ファイルをダウンロードすることはできません。

前提条件

- **Installed Operators** オプションを使用して、Cryostat 3.0 を Red Hat OpenShift にインストールした。
- Red Hat OpenShift プロジェクトに Cryostat インスタンスを作成した。

手順

1. Cryostat インスタンスの **Dashboard** パネルで、ドロップダウンリストから **Target JVM** を選択します。
2. **オプション: Topology** パネルで、**Add to view** アイコンを選択してターゲット JVM を定義できます。アイコンを選択すると、カスタムターゲット接続 URL を定義するためのウィンドウが開きます。
 - a. **Connection URL** フィールドに、JVM の Java Management Extension (JMX) エンドポイントの URL を入力します。
 - b. **オプション: Alias** フィールドに、JMX Service URL のエイリアスを入力します。
 - c. **Create** をクリックします。

図3.1 Create Target ダイアログボックス

3. Cryostat Web コンソールのナビゲーションメニューから、**Events** をクリックします。Web コンソールで **Authentication Required** ダイアログが開く場合があります。プロンプトが表示されたら、**Authentication Required** ダイアログボックスに **Username** と **Password** を入力し、**Save** をクリックしてターゲット JVM に認証情報を提供します。



注記

選択したターゲット JMX で、JMX 接続に対して SSL/TLS 証明書が有効な場合は、プロンプトが表示されたら証明書を追加する必要があります。

Cryostat は、ターゲット JVM アプリケーションの認証情報を暗号化し、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できます。

4. **Event Templates** タブで、リストされたイベントテンプレートを見つけ、その他のオプションメニューを選択します。
5. その他のオプションメニューから、**Download** をクリックします。オペレーティングシステムの設定に応じて、ファイル保存ダイアログが開きます。ファイルを任意の場所に保存します。

図3.2 イベントテンプレートのその他のオプションメニューの例

6. デフォルトのファイルエディターでファイルを開き、必要に応じてファイルを編集します。設定の変更を保持するには、ファイルを保存する必要があります。



注記

後の段階でファイルの特定に役立つ **description** および **provider** 属性に値を追加できます。

7. **Events** メニューから **Event Templates** タブに移動し、**Upload** アイコンをクリックします。Cryostat Web コンソールで **Create Custom Event Template** 開きます。

図3.3 Create Custom Event Template ウィンドウ

8. **Upload** をクリックし、デフォルトのファイルエディターを使用して、1つ以上の設定済みイベントテンプレートファイルを Cryostat Web コンソールにアップロードします。ファイルを **Template XML** ウィンドウにドラッグアンドドロップすることもできます。
9. **Submit** ボタンをクリックします。Cryostat Web コンソールで **Event Templates** タブが開き、カスタムイベントテンプレートが表示されます。
10. **オプション:** イベントテンプレートを作成すると、次のいずれかのオプションを選択し、テンプレートを使用して JFR レコーディングを作成できます。
 - **Automated Rules** メニューから **Create** をクリックし、**Template** リストからイベントテンプレートを選択します。
 - **Events** メニューから、リストされたイベントテンプレートを見つけ、その他のオプションメニューから **Create Recording** を選択します。
 - **Recordings** メニューの **Active Recordings** タブで、**Create** をクリックします。

関連情報

- [JDK Flight Recorder \(JFR\)レコーディングの作成\(Cryostat を使用した JFR レコーディング の作成\)](#)を参照してください。
- [SSL 証明書のアップロード](#) (Cryostat を使用した JFR レコーディングの管理)を参照してください。

- [JDK Flight Recorder \(JFR\)レコーディングのアーカイブ\(Cryostat を使用した JFR レコーディングの管理\)](#)を参照してください。
- [自動化ルールの有効化または無効化\(Cryostat での自動化ルールの使用\)](#)を参照してください。

第4章 統合アプリケーション

Cryostat は、JFR レコーディングに基づくデータの分析方法を強化できる特定のアプリケーションと統合されています。

4.1. GRAFANA での JFR レコーディングの表示

Cryostat 3.0 は Grafana アプリケーションと統合するため、Grafana で JFR レコーディングデータをプロットできます。プロットデータを期間ごとに区切って表示し、ターゲット JVM アプリケーションのパフォーマンスを正確に分析できます。

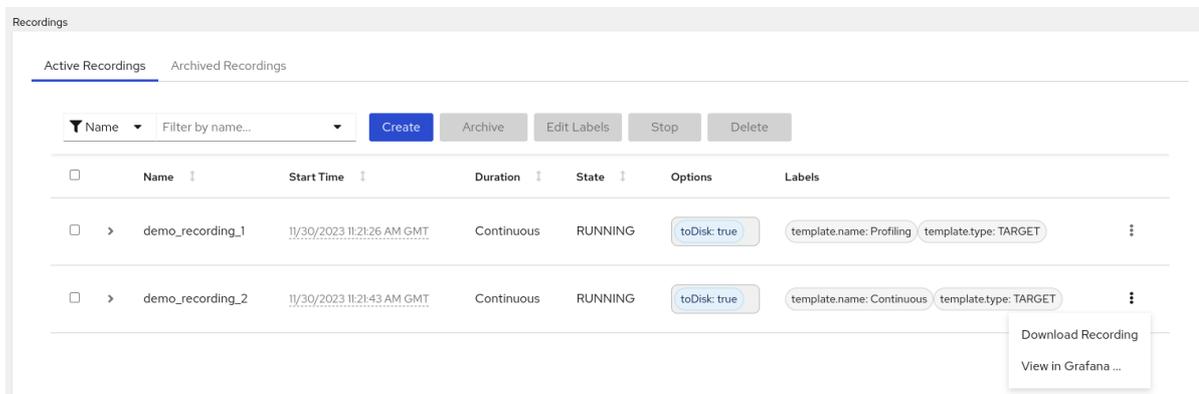
前提条件

- Cryostat インスタンスの認証情報を入力した。
- JFR レコーディングを作成した。[Cryostat Web コンソールでの JFR レコーディングの作成](#) を参照してください。

手順

1. Cryostat インスタンスの **Recordings** メニューまたは **Archives** メニューに移動します。
2. 必要に応じて、**Active Recordings** タブまたは **Archived Recordings** タブをクリックします。
3. JFR レコーディングを見つけ、オーバーフローメニューを選択します。

図4.1 JFR レコーディングの例で利用可能なオーバーフローメニュー項目



4. オーバーフローメニューから **View in Grafana** オプションをクリックします。新しい Web ブラウザーウィンドウで Grafana アプリケーションが開きます。
5. プロンプトが表示されたら、Grafana Web コンソールのログインページに Red Hat OpenShift 認証情報を入力します。ダッシュボードウィンドウが開き、JFR レコーディングのデータがさまざまな時系列プロットで表示されます。
6. **オプション:** プロットの時系列セグメントを選択して、プロットを操作します。画面上のデータが拡大され、その期間のデータのみが表示されます。

図4.2 プロットグラフありの Grafana ダッシュボード例



改訂日時: 2024-07-02