



OpenJDK 11

Release notes for Cryostat 2.0

Legal Notice

Copyright © 2021 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.

Abstract

This document provides an overview of new features in Cryostat 2, as well as a list of potential known issues and possible workarounds.

Table of Contents

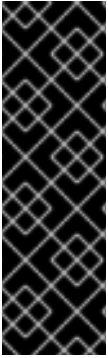
PREFACE	4
MAKING OPEN SOURCE MORE INCLUSIVE	5
PROVIDING FEEDBACK ON RED HAT DOCUMENTATION	6
CHAPTER 1. SUPPORT POLICY FOR CRYOSTAT 2.0	7
CHAPTER 2. NEW FEATURES	8
API permissions for the Cryostat application	8
Automated rules	8
cert-manager API	8
Deployment scenario tree view	9
Endpoint resource watch	9
Event search with TargetEventGetHandler	9
HTTP API behavior change	10
New environment variables	10
New handlers	10
New event template available on Cryostat	10
Pre-configured Grafana dashboard image	11
Response status code message	11
Support for client-defined connection protocols	11
CHAPTER 3. FEATURE ENHANCEMENTS	12
Cryostat web console GUI	12
Cryostat Operator topology view	12
Cryostat Operator controller manager	12
Cryostat cluster ConsoleLink namespace	12
Custom event templates	12
Encode SSL/TLS certificates with supported formats	12
Fabric8 Kubernetes and OpenShift clients	12
Grafana container version	13
Grafana error messages	13
Health check resource definition	13
initialization-resource annotation	13
OLM bundle descriptors	13
Security context constraint (SCC)	13
ServiceRef definition	13
Subprocess management	14
Subprocess report generation	14
truststore environment variable	14
TargetConnectionManager handler	14
Vertex server response management	14
WebSockets	15
CHAPTER 4. UNSUPPORTED AND DEPRECATED FEATURES	16
ContainerJFR variants	16
Event specifier string syntax	16
Interactive mode commands	16
TTY/TCP execution modes	16
WebSocket commands	16
WebSever IO cleanup	16

CHAPTER 5. KNOWN ISSUES	17
OpenShift SDN default CNI network provider	17
Archive upload API Out-Of-Memory (OOM)	17
File upload limit for the integrated Grafana component	18
CHAPTER 6. SECURITY FIXES	19
jsoup Java library	19

PREFACE

Cryostat is a container-native Java application that connects your Java workloads running inside an OpenShift cluster to your desktop JDK Mission Control application.

The primary feature for Cryostat 2.0 is that you can install and deploy Cryostat through the OpenShift Operator in the OperatorHub of the OpenShift Container Platform web console. After you install and deploy Cryostat, you can access a fully featured Cryostat instance and then explore any of the additional features listed in the *Release notes for Cryostat 2.0* guide.



IMPORTANT

The features listed in this guide are Technology Preview features. Technology Preview features are not supported with Red Hat production service level agreements (SLAs) and might not be functionally complete for production purposes. These features provide early access to upcoming product features, enabling customers to test functionality and provide feedback during the development process.

See [Technology Preview Features Support Scope](#) on the Red Hat Customer Portal for information about the support scope for Technology Preview features.

MAKING OPEN SOURCE MORE INCLUSIVE

Red Hat is committed to replacing problematic language in our code, documentation, and web properties. We are beginning with these four terms: master, slave, blacklist, and whitelist. Because of the enormity of this endeavor, these changes will be implemented gradually over several upcoming releases. For more details, see [our CTO Chris Wright's message](#).

PROVIDING FEEDBACK ON RED HAT DOCUMENTATION

We appreciate your feedback on our documentation. To provide feedback, you can highlight the text in a document and add comments.

This section explains how to submit feedback.

Prerequisites

- You are logged in to the Red Hat Customer Portal.
- In the Red Hat Customer Portal, view the document in **Multi-page HTML** format.

Procedure

To provide your feedback, perform the following steps:

1. Click the **Feedback** button in the top-right corner of the document to see existing feedback.



NOTE

The feedback feature is enabled only in the **Multi-page HTML** format.

2. Highlight the section of the document where you want to provide feedback.
3. Click the **Add Feedback** pop-up that appears near the highlighted text.
A text box appears in the feedback section on the right side of the page.
4. Enter your feedback in the text box and click **Submit**.
A documentation issue is created.
5. To view the issue, click the issue tracker link in the feedback view.

CHAPTER 1. SUPPORT POLICY FOR CRYOSTAT 2.0

Red Hat supports a major version of Cryostat for a minimum of 6 months. Red Hat bases this figure on the time that the product gets released on the Red Hat Customer Portal.

You can install and deploy Cryostat on Red Hat OpenShift Container Platform 4.6 or later that is running on an x86_64 architecture.

Additional resources

- For more information about the Cryostat life cycle policy, see [Cryostat](#) on the Red Hat OpenShift Container Platform Life Cycle Policy web page.

CHAPTER 2. NEW FEATURES

Cryostat 2.0 introduces new features that enhance your use of the Cryostat product.

API permissions for the Cryostat application

The Cryostat Operator creates a service account and then configures the Cryostat application to use the service account. The Cryostat Operator checks the service account to ensure its permissions remain in sync with those expected by the authorization check.

In Cryostat 1, the Cryostat Operator applied its own service account's permissions to a Cryostat application. This provided excessive permissions to the Cryostat application.

Automated rules

In Cryostat 2.0, you can use automated rules to enable JFR to continuously monitor a running target application. You do not need to restart or redeploy the application.

You could continue to enable always-on continuous monitoring for an application by setting JVM flags, but this method requires an application restart.

Cryostat 2.0 includes continuous monitoring event templates that you can use to create automated rules and templates. By using continuous monitoring event templates, you can reduce any downtime for specifying a JFR to continuously monitoring an application.

You must specify of the following automated rule definitions for an automated rule:

- One-shot rule definition: You can now create a simple configuration of one-shot automated rules for your JFR configuration. The **eventSpecifier** property contains an **archive** simple string value.
- **matchExpression** definition: Mandatory component of an automated rule that defines JVM targets that must receive an automated rule. Cryostat uses to apply rules to any JVM targets that match regular expressions defined in the **matchExpression** string expression. Valid string expressions include separator, operator, and simple expressions.
 - You can apply OpenShift labels or annotations as criteria for a **matchExpression** definition.

After you upload an automated rule to Cryostat in JSON format, Cryostat validates the definition file.

You can immediately activate a rule definition for any previous defined matching targets. You do not need to re-add or restart targets after you create a rule definition.

You can delete a rule definition by adding the HTTP query **clean=true** parameter to your **DELETE** requests for rule definitions. This parameter directs Cryostat to delete only active recordings. Archived recordings do not get targeted for deletion.

cert-manager API

You can now use the **Enable cert-manager integration** toggle button on the **Create Cryostat** page of your Cryostat Operator to enable or disable end-to-end encryption by using **cert-manager** for your Cryostat application.

Figure 2.1. The Enable cert-manager integration toggle button

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

Additionally, the Cryostat Operator now automatically checks that the **cert-manager** API exists before it deploys Cryostat. This ensures that the Cryostat Operator securely deploys Cryostat. Depending on your needs, you can explicitly set the Cryostat Operator **cert-manager** API check to disabled.

If Cryostat Operator detects a Cryostat instance that requires **cert-manager**, but you did not install **cert-manager**, then you'll receive a warning event for the Cryostat instance on the **Events** page of the OpenShift web console.

Deployment scenario tree view

After you deploy your application on OpenShift or any supported Kubernetes distribution, you can display your Cryostat's information in a JSON-formatted tree view. You can view the following components on the tree view:

- Pods
- Replicas
- Containers
- Endpoints

Endpoint resource watch

The **OpenShiftPlatformClient** automatically enables the endpoint resource watch to set an asynchronous target discovery for your target JVMs.

Event search with TargetEventHandler

You can use the **TargetEventGetHandler** to list or search for event types produced by your target JVM.

HTTP API behavior change

Prior to Cryostat 2.0, HTTP request handlers might override the default **isAsync** method setting of **true** when determining whether to install a handler for the Vertx server.

Each HTTP handler contains code that defines an HTTP API endpoint for Cryostat. The **isAsync** method defines whether an HTTP handler gets handled by either a separate worker thread or a main webserver thread.

New environment variables

Cryostat 2.0 includes the following two new environment variables:

- **CRYOSTAT_TARGET_CACHE_MAX_CONNECTIONS**, which the OpenShift Operator sets to a value for specifying the maximum amount of open JMX connections.
- **CRYOSTAT_TARGET_CACHE_TTL**, which the OpenShift Operator sets to a value for specifying the amount of seconds for caching a JMX connection before they close from inactivity.



NOTE

The OpenShift Operator automatically determines values for these environment variables, so you cannot modify the values specified by the OpenShift Operator for these environment variables.

New handlers

Cryostat 2.0 introduces the following V2 handlers:

- **AbstractV2RequestHandler** that behaves similar to **AbstractAuthenticatedRequestHandler**, but maintaining the overall response format among V2 handlers.
- **TargetRecordingOptionsListGetHandler** that checks recording options supported by your target JVM. Use this handler before you create a recording, so you do not attempt to use recording options not supported by the target JVM.
- **TargetSnapshotPostHandler** that encodes a target response in JSON format for a customized snapshot recording. This V2 snapshot handler also includes the name of the recording, the URL for the downloadable recording file, the URL for the automated analysis report, and much more.



NOTE

A V1 handler can perform snapshot operations similar to a V2 handler, but a V1 handler's HTTP response renders only in plain string format and only contains the name of the snapshot recording.

New event template available on Cryostat

You can now select the **cryostat.jfc** event template from the list of available **Target JVM** templates on the Cryostat web console.

By selecting the **cryostat.jfc** event template, you automatically enable the following Cryostat settings:

- HTTP request information, such as status code, duration, and so on.

- WebSocket message information, such as duration, message type, and so on.
- JMX connection status, such as opened, closed, or failed.

Pre-configured Grafana dashboard image

Cryostat 2.0 includes a pre-configured Grafana dashboard image for viewing your application's metrics. You can also create your own Grafana dashboard image to meet your needs.

Cryostat 1 provided a Grafana image that offered basic customization by the OpenShift Operator during deployment of a Cryostat container.

Response status code message

Response status code messages for a **ClientUrlGetHandler** error message now display in the front of the error message. This is the default setting. The custom exception message remains in the response body of the error message.

Support for client-defined connection protocols

Cryostat 2 now adds a mountpoint for a **clientlib** directory, to which you can add additional JAR files to be loaded into the Cryostat classpath. This supports client-defined connection protocols as alternatives to JMX RMI.

CHAPTER 3. FEATURE ENHANCEMENTS

Cryostat 2.0 includes feature enhancements that build upon the Cryostat 1 offerings.

Cryostat web console GUI

You can access to the Cryostat information page through the **Help** icon in the upper-right corner of the Cryostat web console. On this console page, you can view your version of Cryostat.

Additionally, an archived record's generated name now includes a target alias. The target alias improves record retrieval by linking an alias to the record's generated name.

Cryostat Operator topology view

The Cryostat Operator now applies the correct **app.openshift.io/connects-to** annotation to a Cryostat deployment. This configuration links the Cryostat deployment with the Cryostat Operator deployment in the topology view on the OpenShift Container Platform web console.

Cryostat Operator controller manager

If the Cryostat Operator failed to deploy Cryostat because you did not install **cert-manager**, the Cryostat Operator no longer hangs when it deletes Cryostat custom resources.

Cryostat cluster ConsoleLink namespace

In Cryostat 1, a Cryostat Operator cluster might create a duplicate **ConsoleLink** namespace for a Cryostat cluster. To avoid this issue, Cryostat Operator now replaces the **GenerateName** object with a **SH256** hash value defined in **Name** object of the **ConsoleLink** resource definition.

Custom event templates

You can configure any Cryostat JFR recordings by using custom event templates. You can use custom event templates in the following ways:

- Instruct the Cryostat Operator to pre-configure Cryostat custom event templates by providing them with template files from stored **ConfigMap** objects.
- Upload custom event templates by using Cryostat web applications.

Encode SSL/TLS certificates with supported formats

You can add an SSL/TLS certificate on the Cryostat web interface for your target JVM application.

Valid SSL/TLS certificates are in DER-encoded **base-64** or **binary** formats. Either format supports the following extensions:

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

You can now specify a generated certificate at runtime, so that your target JVM application can use the SSL/TLS for JMX connections. Cryostat attempts to open a JMX connection to a target JVM that uses an SSL/TLS certificate. For a successful JMX connection, Cryostat must pass all its authentication checks on the target JVM certificate that you provided at runtime .

You can use the **POST** handler to accept, validate and write the certificate.

Fabric8 Kubernetes and OpenShift clients

Cryostat 2.0 supports version 4.12.0 of Fabric8 Kubernetes and of OpenShift clients. This version enhances application compatibility with Cryostat and reduces downstream build errors.

Grafana container version

Cryostat 2.0 replaces version 6.4.4 of the Grafana container with version 7.3.6.

Grafana error messages

The **502** error message now relates to a failed JMX connection, while the **512** error messages relates to invalid responses sent to the JFR container.

Health check resource definition

Cryostat Operator replaces the **api/v1/clienturl** resource definition with the **health** resource definition. The Cryostat Operator now uses the **health** endpoint when performing containerized JVM health checks.

initialization-resource annotation

Cryostat Operator now includes an **initialization-resource** annotation in its CSV file's configuration. This annotation enhances the Cryostat Operator instance running on the OpenShift Container Platform web console by providing you with graphical hints to create a Cryostat custom resource for your Cryostat cluster.

OLM bundle descriptors

For Cryostat 2.0, an OLM bundle no longer requires setting an integer value in its **eventOptions** descriptor for checking a JFR recording's duration. Instead, the OLM bundle now uses requires duration formatted values that are defined in its **EventOptions** string.

Supported duration units include **s**, **m**, and **h**. For example, **2h15m** denotes a JFR recording length of 2 hours and 15 minutes.

Security context constraint (SCC)

The Cryostat Operator now defaults to using the **restricted** SCC setting. A pod contained in a Cryostat cluster can now use any permitted **fsGroup** value listed under the **restricted** SCC setting. Meaning that Cryostat pods mounted to a persistent storage (PV) can now have read/write access to their directories.

A pod's read/write access level depends on the pod's **fsGroup** GID value, which the Cryostat Operator configures to adhere to the built-in restricted **SecurityContextConstraint**. A **SecurityContext** element contains pod-level security attributes.

Before Cryostat 2.0, the Cryostat Operator was set to the **default** setting, which caused **fsGroup** access issues for a pod running in a Cryostat cluster

For more information about the permitted range of **fsGroup** values for your Cryostat cluster, see [About pre-allocated security context constraints values](#) in the *OpenShift* documentation website.

ServiceRef definition

A **ServiceRef** definition includes the following new properties that Cryostat includes in all **ServiceRef** objects returned from **GET** communications with HTTP API handlers:

- Annotations, such as the Java application name, labels, or port number.
- User-specific **ServiceRef** values, such as an alias or **connectURL**.

You can use the following two handlers create or delete a **ServiceRef** definition:

- **POST**, which creates a new **ServiceRef** if no existing targets with an identical service URL exist.
- **DELETE**, which removes any **ServiceRef** definitions from the **CustomTargetPlatformClient** if it matches the value specified in the **targetID** path parameter.

Subprocess management

Before Cryostat generates an automated analysis report, Cryostat creates a child subprocess. A child subprocess protects the parent process by accumulating any large memory loads consumed by the report generation.

For example, on a Linux operating system the out-of-memory (OOM) Killer detects a process that tries to request additional memory that is not available from the system and stops the process. If a child subprocess exists, the OOM Killer stops this subprocess but does not interfere with the running parent process.

You can set a minimum or maximum JVM heap size for a child subprocess by using the Cryostat environment variable: **CRYOSTAT_REPORT_GENERATION_MAX_HEAP**. Be aware that a low minimum value might stop a child subprocess before it generates a report, while a high maximum value might cause additional memory constraints on the parent process.

Subprocess report generation

Cryostat 2.0 provides an enhancement to using either **ActiveRecordingReportCache** or **ArchivedRecordingReportCache** to generate a subprocessed report for your Java application. These classes are simplified as follows:

- **ActiveRecordingReportCache** now automatically includes the path to disk recordings.
- **ArchivedRecordingReportCache** calls a utility method that uses the connection manager of a parent process to copy a record to a local disk file. The class passes the local path location to the subprocess.

A subprocess no longer needs to complete the following tasks:

- Establish a JMX connection.
- Handle an SSL/TLS certificate.
- Handle JMX authorization credentials.

The removal of these tasks improves a subprocess's workload in the following ways:

- Quicker end-to-end report generation.
- Reduction in code complexity so that the codebase is easier to maintain.
- Proper and secure handling of SSL/TLS authentication and JMX authentication mechanisms.

truststore environment variable

Cryostat 2.0 replaces **TRUSTSTORE** environment variable name with **SSL_TRUSTSTORE_DIR**.

TargetConnectionManager handler

The **TargetConnectionManager** now support concurrent connections by using a JMX connection timed cache value for any connections between a targeted request and Cryostat. By default, the timed cache is 90 seconds.

Vertx server response management

The Vertx server can now use a **TimeoutHandler** implementation to automatically end a delayed response message sent by a Cryostat request handler. This prevents a client from having to wait indefinitely for a response from a Cryostat request handler.

The **TimeoutHandler** implementation throws a **500** error message when it detects a delayed response from a Cryostat request handler.

WebSockets

The WebSocket includes the following updates:

- WebSocket connection upgraded from a two-way interactive channel to a one-way push Notification Channel (NC).
- WebSocket Notification Channel changed from **api/v1/command** to **api/v1/notifications**, because the WebSocket now uses this channel for one-way push notifications.
- WebSocket can send events on the Notification Channel when you use recordings and event templates for analyzing your JFR data.

When your WebSocket client connects to a one-way push NC, the client automatically receives information about actions performed by other connected clients on the same channel.

CHAPTER 4. UNSUPPORTED AND DEPRECATED FEATURES

Cryostat 2.0 removes some features because of their high maintenance costs, low community interest, and better alternative solutions.

ContainerJFR variants

The Cryostat namespace replaces the **ContainerJFR** namespace, so Cryostat 2.0 no longer specifies any variants of the **ContainerJFR** namespace.

Event specifier string syntax

Cryostat 2.0 does not include any event specific string syntax. Event templates offer a better alternative to event specific string syntax.

Interactive mode commands

The HTTP API and Notification Channels replace the Command Channel. You cannot use any Command Channel implementations or their **CommandExecutor** interfaces in Cryostat 2.0.

TTY/TCP execution modes

Cryostat 2.0 does not contain any TTY and TCP socket execution modes or any mode's configurations. These execution modes did not support platform authentication or target authentication.

You can continue to use the supported WebSocket execution mode.

WebSocket commands

Cryostat 2.0 removed all WebSocket commands. These commands were deprecated in the Cryostat 1 release.

WebSever IO cleanup

Cryostat 2.0 removed the low-memory pressure mode for embedded WebSever IOs. You can no longer use this mode to apply a constant memory size mapping to an embedded WebServer IO.

CHAPTER 5. KNOWN ISSUES

Sometimes a Cryostat release might contain an issue or issues that Red Hat acknowledges and might fix at a later stage during the product’s development. Review each known issue for its description and its resolution.

OpenShift SDN default CNI network provider

Cryostat 2.0 cannot connect an OpenShift cluster that has the following configurations for JVMs located on nodes different to those running on the Cryostat node:

- Use the software-defined networking (SDN) method as the unified cluster network for communication between pods on the cluster.
- Use the default Container Network Interface (CNI) network provider for writing plug-ins to configure network interfaces for containers.

You can resolve this known issue by configuring your cluster to use the Open Virtual Network (OVN) method instead of the SDN method. The OVN method contains the following similar configurations to that of the SDN method:

- Uses the Open vSwitch (OVS) to manage network traffic.
- Uses a plug-in that sets the CNI network provider as default.

Additional resources

- For more information about the SDN method, see [About the OpenShift SDN default CNI network provider](#) in the Red Hat OpenShift documentation.
- For more information about the OVN method, see [About the OVN-Kubernetes default Container Network Interface \(CNI\) network provider](#) in the Red Hat OpenShift documentation.

Archive upload API Out-Of-Memory (OOM)

Cryostat 2.0 consumes more memory than expected when a client sends a request to the Cryostat HTTP **POST /api/v1/recordings** handler. This handler points to the **/opt/cryostat.d/recordings.d** directory on your Cryostat 2.0 instance and you can use the handler to upload **.jfr** binary files to this directory.

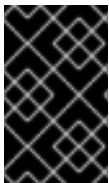
The Cryostat Operator sets a default memory limit of 512 MB for a Cryostat instance deployed in an OpenShift project. If you upload a 150 MB or larger **.jfr** file to your Cryostat 2.0 instance, the OpenShift cluster’s Out of Memory (OOM) killer stops the pod that contains your deployed Cryostat instance.

You can resolve this known issue by copying your **.jfr** binary file to the persistent storage location on your Cryostat 2.0 instance. By using this method, you do not need to send a client request to the Cryostat HTTP **POST /api/v1/recordings** handler to store the **.jfr** binary in the **/opt/cryostat.d/recordings.d** directory.

You can issue the following commands to copy your **.jfr** binary file to the persistent storage location on your Cryostat 2.0 instance:

```
oc exec -i -n <your_namespace> -c <cryostat_container_name> <cryostat_pod_name> -- mkdir /opt/cryostat.d/recordings.d/unlabelled
```

```
oc cp vertx-fib-demo-6f4775cdbf-82dvl_150mb_20211006t152006z.jfr <cryostat_pod_name>:/opt/cryostat.d/recordings.d/unlabelled/vertx-fib-demo-6f4775cdbf-82dvl_150mb_20211006t152006z.jfr -c <cryostat_container_name>
```



IMPORTANT

The previously stated **oc exec** command fails if an **unlabelled** directory already exists in your **/opt/cryostat.d/recordings.d/** path. You can choose to ignore the failed command message and continue with the **oc cp** command.

After you copy the **.jfr** binary file directly into the PVC archives location, you can use a **curl**, an **httplib**, or a **wget** command to verify that the **.jfr** file exists on your Cryostat 2.0 instance.

The following example demonstrates using a **curl** command to verify that Cryostat recognizes the uploaded file that was copied to the persistent storage location with the **oc cp** command. The **<cryostat_url>** value in the example resolves to <https://cryostat-sample-myproject.apps-crc.testing:443>, but you can replace the **<cryostat_url>** value with your application's URL. You can determine your application's URL by issuing the **oc status** command.

```
$ curl -kv -H Authorization:"Bearer $(oc whoami -t)" <cryostat_url>/api/v1/recordings
```

Additional resources

- For more information about the commands that you can use to resolve the Cryostat container memory limit issue, see [OPENJDK-495](#) in the *Red Hat OpenJDK Jira* project.

File upload limit for the integrated Grafana component

For Cryostat 2.0, the integrated **View in Grafana** component cannot accept JFR files larger than 10 MB, because of a configuration issue with the **jfr-datasource** container.

A deployed Cryostat 2.0 pod's **jfr-datasource** container uses default Quarkus settings, which includes the default **quarkus.http.limits.max-body-size** parameter. This parameter sets the maximum size limit for a file on Quarkus, and the parameter has a default value of 10 MB.

If a client attempts to upload a JFR file larger than 10 MB, the **jfr-datasource** web server rejects the file and throws an **HTTP 413** error message.

You can resolve this known issue by completing the following steps:

1. Navigate to your listed active or archive recording in the **Recordings** menu on your Cryostat 2.0 instance.
2. From the overflow menu for your target recording, click the **Download Recording** option.
3. Save the file to your preferred location on your local system.
4. Open the downloaded JFR file in your Java Mission Control (JMC) desktop application.

Additional resources

- For more information about the download functionality on the **Recordings** menu, see [Downloading an active recording or an archived recording](#) in the *Getting started with Cryostat 2.0* guide.

CHAPTER 6. SECURITY FIXES

Cryostat 2.0 includes security fixes that improve upon security offered from the previous release of Cryostat. Review each security fix for its description.

jsoup Java library

The **jsoup** Java library that is pre-installed with Cryostat 2.0 now upgraded from version 1.13.1 to version 1.14.2. This new version improves upon the **jsoup** Java library's handling of HTML and XML syntax.

Additional resources

For more information about the bug fixes associated with the **jsoup** 1.14.2 release, see [jsoup Java HTML Parser release 1.14.2](#) on the **jsoup** documentation website.

Revised on 2021-11-29 17:01:10 UTC