



Red Hat build of Apache Camel 4.4

红帽构建的 Apache Camel for Spring Boot 发行 注记

红帽构建的 Apache Camel 新功能

Red Hat build of Apache Camel 4.4 红帽构建的 Apache Camel for Spring Boot 发行注记

红帽构建的 Apache Camel 新功能

法律通告

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.

摘要

描述红帽构建的 Apache Camel 产品，并提供了有关本版本中新内容的最新详情。

目录

第 1 章 RED HAT BUILD OF APACHE CAMEL FOR SPRING BOOT 4.4 发行注记	3
1.1. 红帽构建的 APACHE CAMEL FOR SPRING BOOT 中的功能	3
1.2. 红帽构建的 APACHE CAMEL FOR SPRING BOOT 支持的平台、配置、数据库和扩展	3
1.3. JAVAX 到 JAKARTA 软件包命名空间更改	3
1.4. RED HAT BUILD OF APACHE CAMEL FOR SPRING BOOT 的重要备注	4
1.5. 修复了红帽构建的 APACHE CAMEL FOR SPRING BOOT 的问题	4
1.6. 红帽构建的 APACHE CAMEL FOR SPRING BOOT 的已知问题	7
第 2 章 其他资源	9

第 1 章 RED HAT BUILD OF APACHE CAMEL FOR SPRING BOOT 4.4 发行注记

1.1. 红帽构建的 APACHE CAMEL FOR SPRING BOOT 中的功能

Red Hat build of Apache Camel for Spring Boot 引入了对 Spring Boot 的 Camel 支持，它为许多 Camel 组件提供自动配置 Camel 和启动程序。在 Spring 上下文中提供的 Camel 上下文自动探测 Camel 路由的建议自动配置，并将关键 Camel 工具（如制作者模板、消费者模板和类型转换器）注册为 beans。

1.2. 红帽构建的 APACHE CAMEL FOR SPRING BOOT 支持的平台、配置、数据库和扩展

- 有关红帽构建的 Apache Camel for Spring Boot 中支持的平台、配置和数据库的详情，请查看客户门户网站中的支持的配置页面（需要登录）。 <https://access.redhat.com/articles/6970899>
- 有关红帽构建的 Apache Camel for Spring Boot 扩展列表，请参阅 [Red Hat build of Apache Camel for Spring Boot Reference](#)（需要登录）。

1.3. JAVAX 到 JAKARTA 软件包命名空间更改

Java EE 移至 Eclipse Foundation 和 Jakarta EE 的建立，因为用于所有 EE API 的 Jakarta EE 9 软件包已改为 **jakartaGalaxy**

文档中的代码片段已更新为使用 **jakarta prerequisites** 命名空间，但您需要注意和查看您自己的应用程序。



注意

这个更改不会影响作为 Java SE 一部分的 javax 软件包。

将应用程序迁移到 EE 10 时，您需要：

- 将 **javax** 软件包中的 EE API 类的任何导入声明或其他源代码使用到 **jakarta**。
- 更改任何 EE 指定的系统属性或其他配置属性，其名称以 **javax** 开始。从 **jakarta** 开始。
- 使用 **META-INF/services/jakarta.[rest_of_name]** 名称格式来识别使用实现 EE 接口或以 **java.util.ServiceLoader** 机制引导的应用程序中的实施类。

1.3.1. 迁移工具

- 源代码迁移： [如何使用 Red Hat Migration Toolkit for Application Auto-Migration for a Application to the Jakarta EE 10 命名空间](#)
- bytecode 转换：对于源代码迁移不是选项的情况，开源 [Eclipse Transformer](#)

其他资源

- 背景： [在 Jakarta EE 权利上更新到 Java Trademarks](#)
- 红帽客户门户网站： [从 Jakarta EE 8 迁移到 EE 10 的 Red Hat JBoss EAP Application Migration](#)

- Jakarta EE: [Javax 到 Jakarta 命名空间生态系统 Progress](#)

1.4. RED HAT BUILD OF APACHE CAMEL FOR SPRING BOOT 的重要备注

1.4.1. 支持 IBM Power 和 IBM Z

IBM Power 和 IBM Z 现在支持 Red Hat build of Camel Spring Boot。

1.4.2. 支持 EIP 断路器

Camel Spring Boot 的 Circuit Breaker EIP 支持 Resilience4j 配置。此配置提供了与 Resilience4j 集成，以用作 Camel 路由中的 Circuit Breaker。

1.4.3. 支持有状态事务

Red Hat build of Camel Example Spring Boot 提供了一个 [Camel Spring Boot JTA quickstart](#)。此快速入门演示了如何在 Spring Boot 上运行 Camel Service，它支持两个外部事务资源上的 JTA 事务：数据库 (MySQL) 和消息代理 (Artemis)。这些外部资源由 OpenShift 提供，它必须在运行此快速入门前启动。

1.5. 修复了红帽构建的 APACHE CAMEL FOR SPRING BOOT 的问题

以下小节列出了红帽构建的 Apache Camel for Spring Boot 中修复的问题。

1.5.1. Red Hat build of Apache Camel for Spring Boot 版本 4.4 增强

表 1.1. Red Hat build of Apache Camel for Spring Boot 版本 4.4 增强

问题	描述
CSB-470	支持用于 Camel for Spring Boot 的 Hawtio 控制台
CSB-1246	camel-olingo4 支持
CSB-1693	添加 Kafka Batch Consumer
CSB-2460	[RFE] 支持组件 camel-smb
CSB-2479	增强 XML IO DSL 以支持 YAML DSL 中的 Bean
CSB-2649	Camel for Spring Boot 支持 IBM Z/P
CSB-2841	为 camel-ssh 组件提供配置算法的支持
CSB-2968	添加对 camel-flink 的支持
CSB-2973	为 azure blob 存储添加 Azure SAS 支持
CSB-3025	创建并支持新的 Camel CICS 组件

问题	描述
CSB-3061	支持组件 camel-mvapich
CSB-3236	离线 Maven 构建器脚本
CSB-3244	支持组件 camel-jasypt
CSB-3357	支持组件 camel-kudu
CSB-3331	支持 cxf-integration-tracing-opentelemetry
CSB-3371	支持组件 camel-groovy
CSB-3462	BeanIO 支持
CSB-4117	camel-cics - 支持通过频道的连接

1.5.2. Red Hat build of Apache Camel for Spring Boot 版本 4.4 修复的问题

表 1.2. Red Hat build of Apache Camel for Spring Boot 版本 4.4 解决了的问题

问题	描述
CSB-1913	CVE-2023-35116 jackson-databind: 通过 cyclic 依赖项拒绝服务
CSB-2007	CVE-2023-2976 guava: 不安全的临时目录创建
CSB-2041	AWS SQS 组件, OCP 探测会导致 POD 错误
CSB-2139	[Micrometer Observability] Unable to see trace id and span id in MDC
CSB-2644	请提供显示用于连接池的 Camel AMQP/JMS 的示例
CSB-2846	CVE-2023-5632 mosquito: Possible Denial for Service due due over CPE consumption
CSB-3042	[camel-mail] java.lang.ClassNotFoundException: org.eclipse.angus.mail.imap.IMAPStore
CSB-3294	使用 CXF 和 JSON 路径时 org.ow2.asm:asm 的依赖项聚合错误
CSB-3298	org.bouncycastle:bcprov-jdk18on:jar:1.72 的依赖项聚合错误

问题	描述
CSB-3302	添加对 findAndModify Operation 的支持
CSB-3316	CVE-2023-51074 json-path: 基于堆栈的缓冲区溢出 (在 Criteria.parse 方法中)
CSB-3331	支持 cxf-integration-tracing-opentelemetry
CSB-3438	CVE-2024-21733 tomcat : 在默认错误页面中调整不相关的请求正文
CSB-3454	camel-bean - 允许在组件上配置 bean 内省缓存
CSB-3601	使用 cxf-rt-rs-service-description-openapi-v3:4.0.2.fuse-redhat-00046 和 camel-openapi-java-starter:4.0.0.redhat-00039 时, 依赖项聚合错误
CSB-3713	CVE-2023-45860 Hazelcast: 在 CSV 文件源连接器中检查权限
CSB-3716	AMQP publisher 应用在启用了本地 JMS 事务的情况下丢失消息
CSB-3722	CVE-2024-26308 commons-compress: OutOfMemoryError unpacking broken Pack200 file
CSB-3725	commons-compress : 由损坏 DUMP 文件的死循环导致的服务 [rhint-camel-spring-boot-4]
CSB-3731	使用 XML DSL IO 时会忽略 restConfiguration 部分
CSB-3765	在向 JSON 进行 marshalling/unmarshalling XML 时发出。
CSB-3837	CVE-2023-5685 xnio: 当通知程序状态链出现问题时, StackOverflowException
CSB-3851	与 servlet/platform-http 一起使用时, onException 处理程序不会在正文响应中设置内容
CSB-3884	[Camel-sap] Unable to通过 CSB 配置属性连接到 SAP 服务器
CSB-3892	camel-file - 使用带有孤立标记文件检查的 min/max 深度时, 可以优化 ant 过滤器
CSB-3916	如果用户使用 OpenTelemetryTracingStrategy 和 opentelemetry.exclude-patterns 来排除 "direct*", 则会出现 NPE
CSB-3922	OpenTelemetryTracingStrategy 使用 opentelemetry.exclude-patterns "process*" 或 "bean*" 将 trace 分成 2 个分支

问题	描述
CSB-3925	在 camel-cics 中提供连接池的请求
CSB-4022	在 Jose P2C 参数上放置最大默认可配置限制，且仅在 WadlGenerator 中明确返回风格表，而不是其他 URL
CSB-4092	对于 JMS* 标头，键入 Conversion Error from byte[] to Long in Camel 4 from Kafka Topic
CSB-4095	camel-salesforce - 启动错误
CSB-4102	CVE-2024-22262 springframework: URL Parsing with Host Validation

1.6. 红帽构建的 APACHE CAMEL FOR SPRING BOOT 的已知问题

以下小节列出了 Red Hat build of Apache Camel for Spring Boot 的已知问题。

1.6.1. Red Hat build of Apache Camel for Spring Boot 版本 4.4 已知问题

如果 `spring.boot.actuator.autoconfigure` 不在依赖项中，则使用 Openshift Maven 插件在 OCP 上部署 [CSB-4318](#) 失败

JKube maven 插件使用以下条件来检查应用是否公开健康端点（使用 `SpringBootHealthCheckEnricher`）。两个类都位于 classpath 中：

- `org.springframework.boot.actuate.health.HealthIndicator`
- `org.springframework.web.context.support.GenericWebApplicationContext`

但是，在不配置 actuator 的情况下，`/actuator/health` 不会公开。这会在 JKube（它们都使用上述端点）配置的就绪度/实时探测之间创建差异，以及应用正在公开的内容。

此错误配置会导致 OpenShift Container Platform 上的部署配置失败，因为生成的 pod 不会处于 Ready 状态，因为没有配置探测对端点的调用。因此，为了使应用程序在 OpenShift Container Platform 上工作，它使用 JKube (openshift-maven-plugin) 部署，需要在依赖项中同时具有 Web 和 actuator 自动配置。

以下示例演示了如何配置 web 和 actuator 自动配置。

Example

```
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-web</artifactId>
</dependency>
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-actuator</artifactId>
</dependency>
```

更新 archetype，如下所示。从以下 archetype 构建的应用程序将使用 JKube 正确部署。

```
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-web</artifactId>
  <exclusions>
    <exclusion>
      <groupId>org.springframework.boot</groupId>
      <artifactId>spring-boot-starter-tomcat</artifactId>
    </exclusion>
  </exclusions>
</dependency>
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-undertow</artifactId>
</dependency>
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-actuator</artifactId>
</dependency>
```

此问题会影响缺少上述依赖项的自定义应用程序。

第 2 章 其他资源

- [支持的配置](#)
- [Red Hat build of Apache Camel for Spring Boot 入门](#)
- [迁移至 Red Hat build of Apache Camel for Spring Boot](#)
- [Red Hat build of Apache Camel for Spring Boot 参考](#)