

# Spring Boot Actuator Web API Documentation

Andy Wilkinson, Stephane Nicoll

3.0.11-SNAPSHOT

# Table of Contents

|                                                                |    |
|----------------------------------------------------------------|----|
| 1. Overview .....                                              | 2  |
| 1.1. URLs .....                                                | 2  |
| 1.2. Timestamps .....                                          | 2  |
| 2. Audit Events ( <b>auditevents</b> ) .....                   | 3  |
| 2.1. Retrieving Audit Events .....                             | 3  |
| 2.1.1. Query Parameters .....                                  | 3  |
| 2.1.2. Response Structure .....                                | 3  |
| 3. Beans ( <b>beans</b> ) .....                                | 5  |
| 3.1. Retrieving the Beans .....                                | 5  |
| 3.1.1. Response Structure .....                                | 6  |
| 4. Caches ( <b>caches</b> ) .....                              | 8  |
| 4.1. Retrieving All Caches .....                               | 8  |
| 4.1.1. Response Structure .....                                | 8  |
| 4.2. Retrieving Caches by Name .....                           | 9  |
| 4.2.1. Query Parameters .....                                  | 9  |
| 4.2.2. Response Structure .....                                | 9  |
| 4.3. Evict All Caches .....                                    | 10 |
| 4.4. Evict a Cache by Name .....                               | 10 |
| 4.4.1. Request Structure .....                                 | 10 |
| 5. Conditions Evaluation Report ( <b>conditions</b> ) .....    | 11 |
| 5.1. Retrieving the Report .....                               | 11 |
| 5.1.1. Response Structure .....                                | 13 |
| 6. Configuration Properties ( <b>configprops</b> ) .....       | 14 |
| 6.1. Retrieving All @ConfigurationProperties Beans .....       | 14 |
| 6.1.1. Response Structure .....                                | 16 |
| 6.2. Retrieving @ConfigurationProperties Beans By Prefix ..... | 17 |
| 6.2.1. Response Structure .....                                | 18 |
| 7. Environment ( <b>env</b> ) .....                            | 20 |
| 7.1. Retrieving the Entire Environment .....                   | 20 |
| 7.1.1. Response Structure .....                                | 40 |
| 7.2. Retrieving a Single Property .....                        | 41 |
| 7.2.1. Response Structure .....                                | 41 |
| 8. Flyway ( <b>flyway</b> ) .....                              | 43 |
| 8.1. Retrieving the Migrations .....                           | 43 |
| 8.1.1. Response Structure .....                                | 43 |
| 9. Health ( <b>health</b> ) .....                              | 45 |
| 9.1. Retrieving the Health of the Application .....            | 45 |
| 9.1.1. Response Structure .....                                | 46 |

|                                                                |    |
|----------------------------------------------------------------|----|
| 9.2. Retrieving the Health of a Component .....                | 47 |
| 9.2.1. Response Structure .....                                | 47 |
| 9.3. Retrieving the Health of a Nested Component .....         | 48 |
| 9.3.1. Response Structure .....                                | 48 |
| 10. Heap Dump ( <b>heapdump</b> ) .....                        | 49 |
| 10.1. Retrieving the Heap Dump .....                           | 49 |
| 11. HTTP Exchanges ( <b>httpexchanges</b> ) .....              | 50 |
| 11.1. Retrieving the HTTP Exchanges .....                      | 50 |
| 11.1.1. Response Structure .....                               | 50 |
| 12. Info ( <b>info</b> ) .....                                 | 52 |
| 12.1. Retrieving the Info .....                                | 52 |
| 12.1.1. Response Structure .....                               | 52 |
| Build Response Structure .....                                 | 52 |
| Git Response Structure .....                                   | 53 |
| 13. Spring Integration graph ( <b>integrationgraph</b> ) ..... | 54 |
| 13.1. Retrieving the Spring Integration Graph .....            | 54 |
| 13.1.1. Response Structure .....                               | 55 |
| 13.2. Rebuilding the Spring Integration Graph .....            | 56 |
| 14. Liquibase ( <b>liquibase</b> ) .....                       | 57 |
| 14.1. Retrieving the Changes .....                             | 57 |
| 14.1.1. Response Structure .....                               | 57 |
| 15. Log File ( <b>logfile</b> ) .....                          | 59 |
| 15.1. Retrieving the Log File .....                            | 59 |
| 15.2. Retrieving Part of the Log File .....                    | 61 |
| 16. Loggers ( <b>loggers</b> ) .....                           | 62 |
| 16.1. Retrieving All Loggers .....                             | 62 |
| 16.1.1. Response Structure .....                               | 63 |
| 16.2. Retrieving a Single Logger .....                         | 63 |
| 16.2.1. Response Structure .....                               | 63 |
| 16.3. Retrieving a Single Group .....                          | 64 |
| 16.3.1. Response Structure .....                               | 64 |
| 16.4. Setting a Log Level .....                                | 64 |
| 16.4.1. Request Structure .....                                | 65 |
| 16.5. Setting a Log Level for a Group .....                    | 65 |
| 16.5.1. Request Structure .....                                | 65 |
| 16.6. Clearing a Log Level .....                               | 65 |
| 17. Mappings ( <b>mappings</b> ) .....                         | 66 |
| 17.1. Retrieving the Mappings .....                            | 66 |
| 17.1.1. Response Structure .....                               | 69 |
| 17.1.2. Dispatcher Servlets Response Structure .....           | 69 |
| 17.1.3. Servlets Response Structure .....                      | 71 |

|                                                              |    |
|--------------------------------------------------------------|----|
| 17.1.4. Servlet Filters Response Structure .....             | 71 |
| 17.1.5. Dispatcher Handlers Response Structure .....         | 71 |
| 18. Metrics ( <b>metrics</b> ) .....                         | 74 |
| 18.1. Retrieving Metric Names .....                          | 74 |
| 18.1.1. Response Structure .....                             | 74 |
| 18.2. Retrieving a Metric .....                              | 74 |
| 18.2.1. Query Parameters .....                               | 75 |
| 18.2.2. Response structure .....                             | 75 |
| 18.3. Drilling Down .....                                    | 76 |
| 19. Prometheus ( <b>prometheus</b> ) .....                   | 77 |
| 19.1. Retrieving All Metrics .....                           | 77 |
| 19.1.1. Query Parameters .....                               | 79 |
| 19.2. Retrieving Filtered Metrics .....                      | 80 |
| 20. Quartz ( <b>quartz</b> ) .....                           | 81 |
| 20.1. Retrieving Registered Groups .....                     | 81 |
| 20.1.1. Response Structure .....                             | 81 |
| 20.2. Retrieving Registered Job Names .....                  | 81 |
| 20.2.1. Response Structure .....                             | 82 |
| 20.3. Retrieving Registered Trigger Names .....              | 82 |
| 20.3.1. Response Structure .....                             | 83 |
| 20.4. Retrieving Overview of a Job Group .....               | 83 |
| 20.4.1. Response Structure .....                             | 84 |
| 20.5. Retrieving Overview of a Trigger Group .....           | 84 |
| 20.5.1. Response Structure .....                             | 85 |
| 20.6. Retrieving Details of a Job .....                      | 87 |
| 20.6.1. Response Structure .....                             | 88 |
| 20.7. Retrieving Details of a Trigger .....                  | 89 |
| 20.7.1. Common Response Structure .....                      | 89 |
| 20.7.2. Cron Trigger Response Structure .....                | 90 |
| 20.7.3. Simple Trigger Response Structure .....              | 91 |
| 20.7.4. Daily Time Interval Trigger Response Structure ..... | 92 |
| 20.7.5. Calendar Interval Trigger Response Structure .....   | 94 |
| 20.7.6. Custom Trigger Response Structure .....              | 95 |
| 21. Scheduled Tasks ( <b>scheduledtasks</b> ) .....          | 96 |
| 21.1. Retrieving the Scheduled Tasks .....                   | 96 |
| 21.1.1. Response Structure .....                             | 97 |
| 22. Sessions ( <b>sessions</b> ) .....                       | 98 |
| 22.1. Retrieving Sessions .....                              | 98 |
| 22.1.1. Query Parameters .....                               | 99 |
| 22.1.2. Response Structure .....                             | 99 |
| 22.2. Retrieving a Single Session .....                      | 99 |

|                                                                      |     |
|----------------------------------------------------------------------|-----|
| 22.2.1. Response Structure .....                                     | 100 |
| 22.3. Deleting a Session .....                                       | 100 |
| 23. Shutdown ( <b>shutdown</b> ) .....                               | 101 |
| 23.1. Shutting Down the Application .....                            | 101 |
| 23.1.1. Response Structure .....                                     | 101 |
| 24. Application Startup ( <b>startup</b> ) .....                     | 102 |
| 24.1. Retrieving the Application Startup Steps .....                 | 102 |
| 24.1.1. Retrieving a snapshot of the Application Startup Steps ..... | 102 |
| 24.1.2. Draining the Application Startup Steps .....                 | 103 |
| 24.1.3. Response Structure .....                                     | 104 |
| 25. Thread Dump ( <b>threaddump</b> ) .....                          | 106 |
| 25.1. Retrieving the Thread Dump as JSON .....                       | 106 |
| 25.1.1. Response Structure .....                                     | 128 |
| 25.2. Retrieving the Thread Dump as Text .....                       | 130 |

This API documentation describes Spring Boot Actuators web endpoints.

# Chapter 1. Overview

Before you proceed, you should read the following topics:

- [URLs](#)
- [Timestamps](#)



In order to get the correct JSON responses documented below, Jackson must be available.

## 1.1. URLs

By default, all web endpoints are available beneath the path `/actuator` with URLs of the form `/actuator/{id}`. The `/actuator` base path can be configured by using the `management.endpoints.web.base-path` property, as shown in the following example:

```
management.endpoints.web.base-path=/manage
```

The preceding `application.properties` example changes the form of the endpoint URLs from `/actuator/{id}` to `/manage/{id}`. For example, the URL `info` endpoint would become `/manage/info`.

## 1.2. Timestamps

All timestamps that are consumed by the endpoints, either as query parameters or in the request body, must be formatted as an offset date and time as specified in [ISO 8601](#).

# Chapter 2. Audit Events (auditevents)

The `auditevents` endpoint provides information about the application's audit events.

## 2.1. Retrieving Audit Events

To retrieve the audit events, make a `GET` request to `/actuator/auditevents`, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/auditevents?principal=alice&after=2023-09-21T00%3A38%3A50.325000515Z&type=logout' -i -X GET
```

The preceding example retrieves `logout` events for the principal, `alice`, that occurred after 09:37 on 7 November 2017 in the UTC timezone. The resulting response is similar to the following:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.spring-boot.actuator.v3+json
Content-Length: 127

{
  "events" : [ {
    "timestamp" : "2023-09-21T00:38:50.325590378Z",
    "principal" : "alice",
    "type" : "logout"
  } ]
}
```

### 2.1.1. Query Parameters

The endpoint uses query parameters to limit the events that it returns. The following table shows the supported query parameters:

| Parameter              | Description                                                                 |
|------------------------|-----------------------------------------------------------------------------|
| <code>after</code>     | Restricts the events to those that occurred after the given time. Optional. |
| <code>principal</code> | Restricts the events to those with the given principal. Optional.           |
| <code>type</code>      | Restricts the events to those with the given type. Optional.                |

### 2.1.2. Response Structure

The response contains details of all of the audit events that matched the query. The following table describes the structure of the response:

| Path               | Type   | Description                               |
|--------------------|--------|-------------------------------------------|
| events             | Array  | An array of audit events.                 |
| events[].timestamp | String | The timestamp of when the event occurred. |
| events[].principal | String | The principal that triggered the event.   |
| events[].type      | String | The type of the event.                    |

# Chapter 3. Beans (beans)

The **beans** endpoint provides information about the application's beans.

## 3.1. Retrieving the Beans

To retrieve the beans, make a **GET** request to **/actuator/beans**, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/beans' -i -X GET
```

The resulting response is similar to the following:

```

HTTP/1.1 200 OK
Content-Type: application/vnd.spring-boot.actuator.v3+json
Content-Length: 1089

{
  "contexts" : {
    "application" : {
      "beans" : {

"org.springframework.boot.autoconfigure.web.servlet.DispatcherServletAutoConfiguration
$DispatcherServletRegistrationConfiguration" : {
      "aliases" : [ ],
      "scope" : "singleton",
      "type" :
"org.springframework.boot.autoconfigure.web.servlet.DispatcherServletAutoConfiguration
$DispatcherServletRegistrationConfiguration",
      "dependencies" : [ ]
    },

"org.springframework.boot.autoconfigure.context.PropertyPlaceholderAutoConfiguration"
: {
      "aliases" : [ ],
      "scope" : "singleton",
      "type" :
"org.springframework.boot.autoconfigure.context.PropertyPlaceholderAutoConfiguration",
      "dependencies" : [ ]
    },

"org.springframework.boot.autoconfigure.web.servlet.DispatcherServletAutoConfiguration
" : {
      "aliases" : [ ],
      "scope" : "singleton",
      "type" :
"org.springframework.boot.autoconfigure.web.servlet.DispatcherServletAutoConfiguration
",
      "dependencies" : [ ]
    }
  }
}

```

### 3.1.1. Response Structure

The response contains details of the application's beans. The following table describes the structure of the response:

| Path                            | Type   | Description                                     |
|---------------------------------|--------|-------------------------------------------------|
| contexts                        | Object | Application contexts keyed by id.               |
| contexts.*.parentId             | String | Id of the parent application context, if any.   |
| contexts.*.beans                | Object | Beans in the application context keyed by name. |
| contexts.*.beans.*.aliases      | Array  | Names of any aliases.                           |
| contexts.*.beans.*.scope        | String | Scope of the bean.                              |
| contexts.*.beans.*.type         | String | Fully qualified type of the bean.               |
| contexts.*.beans.*.resource     | String | Resource in which the bean was defined, if any. |
| contexts.*.beans.*.dependencies | Array  | Names of any dependencies.                      |

# Chapter 4. Caches (caches)

The **caches** endpoint provides access to the application's caches.

## 4.1. Retrieving All Caches

To retrieve the application's caches, make a **GET** request to **/actuator/caches**, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/caches' -i -X GET
```

The resulting response is similar to the following:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.spring-boot.actuator.v3+json
Content-Length: 435

{
  "cacheManagers" : {
    "anotherCacheManager" : {
      "caches" : {
        "countries" : {
          "target" : "java.util.concurrent.ConcurrentHashMap"
        }
      }
    },
    "cacheManager" : {
      "caches" : {
        "cities" : {
          "target" : "java.util.concurrent.ConcurrentHashMap"
        },
        "countries" : {
          "target" : "java.util.concurrent.ConcurrentHashMap"
        }
      }
    }
  }
}
```

### 4.1.1. Response Structure

The response contains details of the application's caches. The following table describes the structure of the response:

| Path                 | Type          | Description                 |
|----------------------|---------------|-----------------------------|
| <b>cacheManagers</b> | <b>Object</b> | Cache managers keyed by id. |

| Path                                         | Type                | Description                                      |
|----------------------------------------------|---------------------|--------------------------------------------------|
| <code>cacheManagers.*.caches</code>          | <code>Object</code> | Caches in the application context keyed by name. |
| <code>cacheManagers.*.caches.*.target</code> | <code>String</code> | Fully qualified name of the native cache.        |

## 4.2. Retrieving Caches by Name

To retrieve a cache by name, make a `GET` request to `/actuator/caches/{name}`, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/caches/cities' -i -X GET
```

The preceding example retrieves information about the cache named `cities`. The resulting response is similar to the following:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.spring-boot.actuator.v3+json
Content-Length: 113

{
  "target" : "java.util.concurrent.ConcurrentHashMap",
  "name" : "cities",
  "cacheManager" : "cacheManager"
}
```

### 4.2.1. Query Parameters

If the requested name is specific enough to identify a single cache, no extra parameter is required. Otherwise, the `cacheManager` must be specified. The following table shows the supported query parameters:

| Parameter                 | Description                                                                                |
|---------------------------|--------------------------------------------------------------------------------------------|
| <code>cacheManager</code> | Name of the cacheManager to qualify the cache. May be omitted if the cache name is unique. |

### 4.2.2. Response Structure

The response contains details of the requested cache. The following table describes the structure of the response:

| Path                      | Type                | Description         |
|---------------------------|---------------------|---------------------|
| <code>name</code>         | <code>String</code> | Cache name.         |
| <code>cacheManager</code> | <code>String</code> | Cache manager name. |

| Path                | Type                | Description                               |
|---------------------|---------------------|-------------------------------------------|
| <code>target</code> | <code>String</code> | Fully qualified name of the native cache. |

## 4.3. Evict All Caches

To clear all available caches, make a `DELETE` request to `/actuator/caches` as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/caches' -i -X DELETE
```

## 4.4. Evict a Cache by Name

To evict a particular cache, make a `DELETE` request to `/actuator/caches/{name}` as shown in the following curl-based example:

```
$ curl
'http://localhost:8080/actuator/caches/countries?cacheManager=anotherCacheManager' -i
-X DELETE \
-H 'Content-Type: application/x-www-form-urlencoded'
```



As there are two caches named `countries`, the `cacheManager` has to be provided to specify which `Cache` should be cleared.

### 4.4.1. Request Structure

If the requested name is specific enough to identify a single cache, no extra parameter is required. Otherwise, the `cacheManager` must be specified. The following table shows the supported query parameters:

| Parameter                 | Description                                                                                |
|---------------------------|--------------------------------------------------------------------------------------------|
| <code>cacheManager</code> | Name of the cacheManager to qualify the cache. May be omitted if the cache name is unique. |

# Chapter 5. Conditions Evaluation Report

## (conditions)

The `conditions` endpoint provides information about the evaluation of conditions on configuration and auto-configuration classes.

### 5.1. Retrieving the Report

To retrieve the report, make a `GET` request to `/actuator/conditions`, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/conditions' -i -X GET
```

The resulting response is similar to the following:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.spring-boot.actuator.v3+json
Content-Length: 3705

{
  "contexts" : {
    "application" : {
      "positiveMatches" : {
        "EndpointAutoConfiguration#endpointOperationParameterMapper" : [ {
          "condition" : "OnBeanCondition",
          "message" : "@ConditionalOnMissingBean (types:
org.springframework.boot.actuate.endpoint.invoke.ParameterValueMapper; SearchStrategy:
all) did not find any beans"
        } ],
        "JacksonEndpointAutoConfiguration#endpointObjectMapper" : [ {
          "condition" : "OnClassCondition",
          "message" : "@ConditionalOnClass found required classes
'com.fasterxml.jackson.databind.ObjectMapper',
'org.springframework.http.converter.json.Jackson2ObjectMapperBuilder'"
        }, {
          "condition" : "OnPropertyCondition",
          "message" : "@ConditionalOnProperty (management.endpoints.jackson.isolated-
object-mapper) matched"
        } ],
        "EndpointAutoConfiguration#endpointCachingOperationInvokerAdvisor" : [ {
          "condition" : "OnBeanCondition",
          "message" : "@ConditionalOnMissingBean (types:
org.springframework.boot.actuate.endpoint.invoker.cache.CachingOperationInvokerAdvisor
; SearchStrategy: all) did not find any beans"
        } ]
      },
      "negativeMatches" : {
```

```

"WebFluxEndpointManagementContextConfiguration" : {
  "notMatched" : [ {
    "condition" : "OnWebApplicationCondition",
    "message" : "not a reactive web application"
  } ],
  "matched" : [ {
    "condition" : "OnClassCondition",
    "message" : "@ConditionalOnClass found required classes
'org.springframework.web.reactive.DispatcherHandler',
'org.springframework.http.server.reactive.HttpHandler'"
  } ]
},
"GsonHttpMessageConvertersConfiguration.GsonHttpMessageConverterConfiguration"
: {
  "notMatched" : [ {
    "condition" :
"GsonHttpMessageConvertersConfiguration.PreferGsonOrJacksonAndJsonbUnavailableConditio
n",
    "message" : "AnyNestedCondition 0 matched 2 did not; NestedCondition on
GsonHttpMessageConvertersConfiguration.PreferGsonOrJacksonAndJsonbUnavailableCondition
.JacksonJsonbUnavailable NoneNestedConditions 1 matched 1 did not; NestedCondition on
GsonHttpMessageConvertersConfiguration.JacksonAndJsonbUnavailableCondition.JsonbPrefer
red @ConditionalOnProperty (spring.mvc.converters.preferred-json-mapper=jsonb) did not
find property 'spring.mvc.converters.preferred-json-mapper'; NestedCondition on
GsonHttpMessageConvertersConfiguration.JacksonAndJsonbUnavailableCondition.JacksonAvai
lable @ConditionalOnBean (types:
org.springframework.http.converter.json.MappingJackson2HttpMessageConverter;
SearchStrategy: all) found bean 'mappingJackson2HttpMessageConverter'; NestedCondition
on
GsonHttpMessageConvertersConfiguration.PreferGsonOrJacksonAndJsonbUnavailableCondition
.GsonPreferred @ConditionalOnProperty (spring.mvc.converters.preferred-json-
mapper=gson) did not find property 'spring.mvc.converters.preferred-json-mapper'"
  } ],
  "matched" : [ ]
},

"WebMvcEndpointManagementContextConfiguration#managementHealthEndpointWebMvcHandlerMap
ping" : {
  "notMatched" : [ {
    "condition" : "OnManagementPortCondition",
    "message" : "Management Port actual port type (SAME) did not match
required type (DIFFERENT)"
  } ],
  "matched" : [ ]
},
"unconditionalClasses" : [
"org.springframework.boot.autoconfigure.context.PropertyPlaceholderAutoConfiguration",
"org.springframework.boot.actuate.autoconfigure.endpoint.jackson.JacksonEndpointAutoCo
nfiguration",
"org.springframework.boot.actuate.autoconfigure.endpoint.EndpointAutoConfiguration" ]

```

```

    }
  }
}

```

### 5.1.1. Response Structure

The response contains details of the application's condition evaluation. The following table describes the structure of the response:

| Path                                                             | Type   | Description                                                |
|------------------------------------------------------------------|--------|------------------------------------------------------------|
| <code>contexts</code>                                            | Object | Application contexts keyed by id.                          |
| <code>contexts.*.positiveMatches</code>                          | Object | Classes and methods with conditions that were matched.     |
| <code>contexts.*.positiveMatches.*.[]condition</code>            | String | Name of the condition.                                     |
| <code>contexts.*.positiveMatches.*.[]message</code>              | String | Details of why the condition was matched.                  |
| <code>contexts.*.negativeMatches</code>                          | Object | Classes and methods with conditions that were not matched. |
| <code>contexts.*.negativeMatches.*.notMatched</code>             | Array  | Conditions that were matched.                              |
| <code>contexts.*.negativeMatches.*.notMatched.[]condition</code> | String | Name of the condition.                                     |
| <code>contexts.*.negativeMatches.*.notMatched.[]message</code>   | String | Details of why the condition was not matched.              |
| <code>contexts.*.negativeMatches.*.matched</code>                | Array  | Conditions that were matched.                              |
| <code>contexts.*.negativeMatches.*.matched.[]condition</code>    | String | Name of the condition.                                     |
| <code>contexts.*.negativeMatches.*.matched.[]message</code>      | String | Details of why the condition was matched.                  |
| <code>contexts.*.unconditionalClasses</code>                     | Array  | Names of unconditional auto-configuration classes if any.  |
| <code>contexts.*.parentId</code>                                 | String | Id of the parent application context, if any.              |

# Chapter 6. Configuration Properties

## (configprops)

The `configprops` endpoint provides information about the application's `@ConfigurationProperties` beans.

### 6.1. Retrieving All `@ConfigurationProperties` Beans

To retrieve all of the `@ConfigurationProperties` beans, make a `GET` request to `/actuator/configprops`, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/configprops' -i -X GET
```

The resulting response is similar to the following:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.spring-boot.actuator.v3+json
Content-Length: 3411

{
  "contexts" : {
    "application" : {
      "beans" : {
        "management.endpoints.web.cors-
org.springframework.boot.actuate.autoconfigure.endpoint.web.CorsEndpointProperties" :
{
      "prefix" : "management.endpoints.web.cors",
      "properties" : {
        "allowedOrigins" : [ ],
        "maxAge" : "PT30M",
        "exposedHeaders" : [ ],
        "allowedOriginPatterns" : [ ],
        "allowedHeaders" : [ ],
        "allowedMethods" : [ ]
      },
      "inputs" : {
        "allowedOrigins" : [ ],
        "maxAge" : { },
        "exposedHeaders" : [ ],
        "allowedOriginPatterns" : [ ],
        "allowedHeaders" : [ ],
        "allowedMethods" : [ ]
      }
    },
    "management.endpoints.web-
org.springframework.boot.actuate.autoconfigure.endpoint.web.WebEndpointProperties" : {
      "prefix" : "management.endpoints.web",
```

```

    "properties" : {
      "pathMapping" : { },
      "exposure" : {
        "include" : [ "*" ],
        "exclude" : [ ]
      },
      "basePath" : "/actuator",
      "discovery" : {
        "enabled" : true
      }
    },
    "inputs" : {
      "pathMapping" : { },
      "exposure" : {
        "include" : [ {
          "value" : "*",
          "origin" : "\\management.endpoints.web.exposure.include\" from
property source \"Inlined Test Properties\""
        } ],
        "exclude" : [ ]
      },
      "basePath" : { },
      "discovery" : {
        "enabled" : { }
      }
    }
  },
  "spring.web-org.springframework.boot.autoconfigure.web.WebProperties" : {
    "prefix" : "spring.web",
    "properties" : {
      "localeResolver" : "ACCEPT_HEADER",
      "resources" : {
        "staticLocations" : [ "classpath:/META-INF/resources/",
"classpath:/resources/", "classpath:/static/", "classpath:/public/" ],
        "addMappings" : true,
        "chain" : {
          "cache" : true,
          "compressed" : false,
          "strategy" : {
            "fixed" : {
              "enabled" : false,
              "paths" : [ "/*" ]
            },
            "content" : {
              "enabled" : false,
              "paths" : [ "/*" ]
            }
          }
        }
      },
      "cache" : {
        "cachecontrol" : { },

```

```
"useLastModified" : true
    }
}
},
 "inputs" : {
    "localeResolver" : { },
    "resources" : {
        "staticLocations" : [ { }, { }, { }, { } ],
        "addMappings" : { },
        "chain" : {
            "cache" : { },
            "compressed" : { },
            "strategy" : {
                "fixed" : {
                    "enabled" : { },
                    "paths" : [ { } ]
                },
                "content" : {
                    "enabled" : { },
                    "paths" : [ { } ]
                }
            },
            "cache" : {
                "cachecontrol" : { },
                "useLastModified" : { }
            }
        }
    },
    "cache" : {
        "cachecontrol" : { },
        "useLastModified" : { }
    }
}
}
```

### 6.1.1. Response Structure

The response contains details of the application's `@ConfigurationProperties` beans. The following table describes the structure of the response:

| Path                          | Type   | Description                                           |
|-------------------------------|--------|-------------------------------------------------------|
| contexts                      | Object | Application contexts keyed by id.                     |
| contexts.*.beans.*            | Object | @ConfigurationProperties beans keyed by bean name.    |
| contexts.*.beans.*.prefix     | String | Prefix applied to the names of the bean's properties. |
| contexts.*.beans.*.properties | Object | Properties of the bean as name-value pairs.           |

| Path                                   | Type                | Description                                                                    |
|----------------------------------------|---------------------|--------------------------------------------------------------------------------|
| <code>contexts.*.beans.*.inputs</code> | <code>Object</code> | Origin and value of the configuration property used when binding to this bean. |
| <code>contexts.*.parentId</code>       | <code>String</code> | Id of the parent application context, if any.                                  |

## 6.2. Retrieving @ConfigurationProperties Beans By Prefix

To retrieve the `@ConfigurationProperties` beans mapped under a certain prefix, make a `GET` request to `/actuator/configprops/{prefix}`, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/configprops/spring.jackson' -i -X GET
```

The resulting response is similar to the following:

```

HTTP/1.1 200 OK
Content-Disposition: inline;filename=f.txt
Content-Type: application/vnd.spring-boot.actuator.v3+json
Content-Length: 676

{
  "contexts" : {
    "application" : {
      "beans" : {
        "spring.jackson-
org.springframework.boot.autoconfigure.jackson.JacksonProperties" : {
          "prefix" : "spring.jackson",
          "properties" : {
            "serialization" : { },
            "visibility" : { },
            "parser" : { },
            "deserialization" : { },
            "generator" : { },
            "mapper" : { }
          },
          "inputs" : {
            "serialization" : { },
            "visibility" : { },
            "parser" : { },
            "deserialization" : { },
            "generator" : { },
            "mapper" : { }
          }
        }
      }
    }
  }
}

```



The `{prefix}` does not need to be exact, a more general prefix will return all beans mapped under that prefix stem.

### 6.2.1. Response Structure

The response contains details of the application's `@ConfigurationProperties` beans. The following table describes the structure of the response:

| Path                            | Type                | Description                                                     |
|---------------------------------|---------------------|-----------------------------------------------------------------|
| <code>contexts</code>           | <code>Object</code> | Application contexts keyed by id.                               |
| <code>contexts.*.beans.*</code> | <code>Object</code> | <code>@ConfigurationProperties</code> beans keyed by bean name. |

| Path                                       | Type   | Description                                                                    |
|--------------------------------------------|--------|--------------------------------------------------------------------------------|
| <code>contexts.*.beans.*.prefix</code>     | String | Prefix applied to the names of the bean's properties.                          |
| <code>contexts.*.beans.*.properties</code> | Object | Properties of the bean as name-value pairs.                                    |
| <code>contexts.*.beans.*.inputs</code>     | Object | Origin and value of the configuration property used when binding to this bean. |
| <code>contexts.*.parentId</code>           | String | Id of the parent application context, if any.                                  |

# Chapter 7. Environment (env)

The `env` endpoint provides information about the application's `Environment`.

## 7.1. Retrieving the Entire Environment

To retrieve the entire environment, make a `GET` request to `/actuator/env`, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/env' -i -X GET
```

The resulting response is similar to the following:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.spring-boot.actuator.v3+json
Content-Length: 71108

{
  "activeProfiles" : [ ],
  "propertySources" : [ {
    "name" : "servletContextInitParams",
    "properties" : { }
  }, {
    "name" : "systemProperties",
    "properties" : {
      "java.specification.version" : {
        "value" : "17"
      },
      "java.class.path" : {
        "value" : "/root/.gradle/caches/7.6.2/workerMain/gradle-
worker.jar:/tmp/build/6da1d659/git-repo/spring-boot-project/spring-boot-actuator-
autoconfigure/build/classes/java/test:/tmp/build/6da1d659/git-repo/spring-boot-
project/spring-boot-actuator-
autoconfigure/build/resources/test:/tmp/build/6da1d659/git-repo/spring-boot-
project/spring-boot-actuator-
autoconfigure/build/classes/java/main:/tmp/build/6da1d659/git-repo/spring-boot-
project/spring-boot-actuator-
autoconfigure/build/resources/main:/tmp/build/6da1d659/git-repo/spring-boot-
project/spring-boot-tools/spring-boot-test-support/build/libs/spring-boot-test-
support-3.0.11-SNAPSHOT.jar:/tmp/build/6da1d659/git-repo/spring-boot-project/spring-
boot-actuator/build/libs/spring-boot-actuator-3.0.11-
SNAPSHOT.jar:/tmp/build/6da1d659/git-repo/spring-boot-project/spring-boot-
autoconfigure/build/libs/spring-boot-autoconfigure-3.0.11-
SNAPSHOT.jar:/tmp/build/6da1d659/git-repo/spring-boot-project/spring-boot-
test/build/libs/spring-boot-test-3.0.11-SNAPSHOT.jar:/tmp/build/6da1d659/git-
repo/spring-boot-project/spring-boot/build/libs/spring-boot-3.0.11-
SNAPSHOT.jar:/root/.gradle/caches/modules-2/files-
2.1/com.fasterxml.jackson.dataformat/jackson-dataformat-
```

xml/2.14.3/7e2bda2238924f21187718ed8183a6de1ffc909/jackson-dataformat-xml-  
 2.14.3.jar:/root/.gradle/caches/modules-2/files-  
 2.1/com.fasterxml.jackson.datatype/jackson-datatype-  
 jsr310/2.14.3/35f7f751c3555dadccb6f173f9ac1e035a648da3/jackson-datatype-jsr310-  
 2.14.3.jar:/root/.gradle/caches/modules-2/files-2.1/org.springframework.data/spring-  
 data-cassandra/4.0.10/fa5cd5336aa340280a628073edc2cdd78570c2a5/spring-data-cassandra-  
 4.0.10.jar:/root/.gradle/caches/modules-2/files-2.1/com.datastax.oss/java-driver-  
 query-builder/4.15.0/308cebab44e688c05ab61390005500d8db231ecd/java-driver-query-  
 builder-4.15.0.jar:/root/.gradle/caches/modules-2/files-2.1/com.datastax.oss/java-  
 driver-core/4.15.0/aedf50875f5be8589770957735d37d2580ca5ac5/java-driver-core-  
 4.15.0.jar:/root/.gradle/caches/modules-2/files-2.1/org.springframework.data/spring-  
 data-elasticsearch/5.0.10/940b0e1bd09e8ed523fdb155060a119b2f3bcc0/spring-data-  
 elasticsearch-5.0.10.jar:/root/.gradle/caches/modules-2/files-  
 2.1/org.glassfish.jersey.media/jersey-media-json-  
 jackson/3.1.3/c08467ee1da7f9c7cf0b28c7f3c526e183802bd7/jersey-media-json-jackson-  
 3.1.3.jar:/root/.gradle/caches/modules-2/files-  
 2.1/com.fasterxml.jackson.module/jackson-module-jakarta-xmlbind-  
 annotations/2.14.3/3cfc7899a420d5449288b3ea883bfb61fbfb5d4b/jackson-module-jakarta-  
 xmlbind-annotations-2.14.3.jar:/root/.gradle/caches/modules-2/files-  
 2.1/io.micrometer/micrometer-tracing-reporter-  
 wavefront/1.0.10/534b3f49e3a6b349c1669a100bc86ff6907e0467/micrometer-tracing-reporter-  
 wavefront-1.0.10.jar:/root/.gradle/caches/modules-2/files-2.1/com.wavefront/wavefront-  
 internal-reporter-java/1.7.16/135f9be9f149a6f0614db155ddf51e216d0d2891/wavefront-  
 internal-reporter-java-1.7.16.jar:/root/.gradle/caches/modules-2/files-  
 2.1/com.fasterxml.jackson.dataformat/jackson-dataformat-  
 yaml/2.14.3/cf8550a0c965aa3a4d08c3857e3f44dae9db9f8b/jackson-dataformat-yaml-  
 2.14.3.jar:/root/.gradle/caches/modules-2/files-2.1/org.springframework.data/spring-  
 data-rest-webmvc/4.0.10/8ecfed6982a390f7962a8a34ba11fa518277a0ba/spring-data-rest-  
 webmvc-4.0.10.jar:/root/.gradle/caches/modules-2/files-  
 2.1/org.springframework.data/spring-data-rest-  
 core/4.0.10/482b8f43abf5c47b964c5ad8ebab253533c9c9f5/spring-data-rest-core-  
 4.0.10.jar:/root/.gradle/caches/modules-2/files-  
 2.1/com.fasterxml.jackson.datatype/jackson-datatype-  
 jdk8/2.14.3/9474ab543a65867461f8326c8fa0494b3efc761/jackson-datatype-jdk8-  
 2.14.3.jar:/root/.gradle/caches/modules-2/files-  
 2.1/com.fasterxml.jackson.core/jackson-  
 core/2.14.3/4fa101f81504ed6811bf1823270adb51a9b45ff3/jackson-core-  
 2.14.3.jar:/root/.gradle/caches/modules-2/files-2.1/org.apache.kafka/kafka-  
 streams/3.3.2/446ef91b2cddb442b66f7e536f3dd1cbf8729b8/kafka-streams-  
 3.3.2.jar:/root/.gradle/caches/modules-2/files-2.1/com.fasterxml.jackson.core/jackson-  
 annotations/2.14.3/b49701acb02b8e95ae1967821cfbafc220cf5795/jackson-annotations-  
 2.14.3.jar:/root/.gradle/caches/modules-2/files-2.1/org.springframework.data/spring-  
 data-couchbase/5.0.10/1c7a2c7b14be002257b30d5b48faf4390b3e20e1/spring-data-couchbase-  
 5.0.10.jar:/root/.gradle/caches/modules-2/files-2.1/io.micrometer/micrometer-registry-  
 wavefront/1.10.11/646f95374da13ed39993cc7bb92e5f6b463518aa/micrometer-registry-  
 wavefront-1.10.11.jar:/root/.gradle/caches/modules-2/files-  
 2.1/com.wavefront/wavefront-sdk-  
 java/3.0.4/370dc7ed21a55b4d49c37fad6fc6343402873587/wavefront-sdk-java-  
 3.0.4.jar:/root/.gradle/caches/modules-2/files-  
 2.1/org.springframework.restdocs/spring-restdocs-  
 mockmvc/3.0.0/d6b6ac3817f68f2dcf9ad4db4e00836776abb1ea/spring-restdocs-mockmvc-

3.0.0.jar:/root/.gradle/caches/modules-2/files-  
2.1/org.springframework.restdocs/spring-restdocs-  
webtestclient/3.0.0/671307a91b382e690191eb40dcc44250092d373a/spring-restdocs-  
webtestclient-3.0.0.jar:/root/.gradle/caches/modules-2/files-  
2.1/org.springframework.restdocs/spring-restdocs-  
core/3.0.0/82e4c7cf63a478e71272662c511f3485f4a5af7b/spring-restdocs-core-  
3.0.0.jar:/root/.gradle/caches/modules-2/files-2.1/com.fasterxml.jackson.core/jackson-  
databind/2.14.3/ba0373b04bf0f03b0cd268cd2e5e8444aaaf9208/jackson-databind-  
2.14.3.jar:/root/.gradle/caches/modules-2/files-2.1/ch.qos.logback/logback-  
classic/1.4.11/54450c0c783e896a1a6d88c043bd2f1daba1c382/logback-classic-  
1.4.11.jar:/root/.gradle/caches/modules-2/files-2.1/com.github.ben-  
manes.caffeine/caffeine/3.1.8/24795585df8afaf70a2cd534786904ea5889c047/caffeine-  
3.1.8.jar:/root/.gradle/caches/modules-2/files-2.1/com.hazelcast/hazelcast-  
spring/5.1.7/687fbb49234610a21fc25db4d9be3960b055511f/hazelcast-spring-  
5.1.7.jar:/root/.gradle/caches/modules-2/files-  
2.1/com.hazelcast/hazelcast/5.1.7/cd2036e9af6e604d63188275ca457202d85e99d1/hazelcast-  
5.1.7.jar:/root/.gradle/caches/modules-2/files-  
2.1/com.zaxxer/HikariCP/5.0.1/a74c7f0a37046846e88d54f7cb6ea6d565c65f9c/HikariCP-  
5.0.1.jar:/root/.gradle/caches/modules-2/files-2.1/io.micrometer/micrometer-registry-  
jmx/1.10.11/decc764003f2a624e84347a8e17616678c2e9a9/micrometer-registry-jmx-  
1.10.11.jar:/root/.gradle/caches/modules-2/files-2.1/io.dropwizard.metrics/metrics-  
jmx/4.2.19/474c1c764df01a0d93fc268da1f779ab97fe99ae/metrics-jmx-  
4.2.19.jar:/root/.gradle/caches/modules-2/files-2.1/io.lettuce/lettuce-  
core/6.2.6.RELEASE/4d1c1700a271903b15384bf90c28471d750420ec/lettuce-core-  
6.2.6.RELEASE.jar:/root/.gradle/caches/modules-2/files-  
2.1/org.springframework.kafka/spring-  
kafka/3.0.11/8b93bea478da5f83629a054f51db38e50c7758d0/spring-kafka-  
3.0.11.jar:/root/.gradle/caches/modules-2/files-2.1/org.cache2k/cache2k-  
micrometer/2.6.1.Final/58731da010cc872915e012325027954bdc3122c8/cache2k-micrometer-  
2.6.1.Final.jar:/root/.gradle/caches/modules-2/files-2.1/org.hibernate.orm/hibernate-  
micrometer/6.1.7.Final/87261a3761c7245d6d2bd382583eea93e7e06063/hibernate-micrometer-  
6.1.7.Final.jar:/root/.gradle/caches/modules-2/files-2.1/io.micrometer/micrometer-  
registry-stackdriver/1.10.11/46e9f06c3d382c33bd3742e81b069f9dc66e1fdd/micrometer-  
registry-stackdriver-1.10.11.jar:/root/.gradle/caches/modules-2/files-  
2.1/io.micrometer/micrometer-registry-  
appoptics/1.10.11/b2f766a8b9df803de101e61eab565f47b2cbd558/micrometer-registry-  
appoptics-1.10.11.jar:/root/.gradle/caches/modules-2/files-  
2.1/io.micrometer/micrometer-registry-  
atlas/1.10.11/deac7d64165862b043110a2c0d2a1407cd04826b/micrometer-registry-atlas-  
1.10.11.jar:/root/.gradle/caches/modules-2/files-2.1/io.micrometer/micrometer-  
registry-datadog/1.10.11/f2c2841fdc13790b845d9335fe705d6a3c48386a/micrometer-registry-  
datadog-1.10.11.jar:/root/.gradle/caches/modules-2/files-2.1/io.micrometer/micrometer-  
registry-dynatrace/1.10.11/3ab470e6b810b245f87bb87778297134ee28e696/micrometer-  
registry-dynatrace-1.10.11.jar:/root/.gradle/caches/modules-2/files-  
2.1/io.micrometer/micrometer-registry-  
elastic/1.10.11/d30c003c9b6b443f0f8747c641329ee44b6d4aeb/micrometer-registry-elastic-  
1.10.11.jar:/root/.gradle/caches/modules-2/files-2.1/io.micrometer/micrometer-  
registry-ganglia/1.10.11/2305967ac5633608f774c21e52ab4b30d7e226cb/micrometer-registry-  
ganglia-1.10.11.jar:/root/.gradle/caches/modules-2/files-2.1/io.micrometer/micrometer-  
registry-graphite/1.10.11/8ac70ffd9edf5a4defcf907e5478f5ae2e28e3f1/micrometer-  
registry-graphite-1.10.11.jar:/root/.gradle/caches/modules-2/files-

2.1/io.micrometer/micrometer-registry-humio/1.10.11/85659bcd46e8da22d4105e79af55b864869824e5/micrometer-registry-humio-1.10.11.jar:/root/.gradle/caches/modules-2/files-2.1/io.micrometer/micrometer-registry-influx/1.10.11/8b81fed67b125134ef49b53144334e29a1de58e/micrometer-registry-influx-1.10.11.jar:/root/.gradle/caches/modules-2/files-2.1/io.micrometer/micrometer-registry-kairos/1.10.11/2d925e1673f726e9c0f03f8e423e8b9055d3e83e/micrometer-registry-kairos-1.10.11.jar:/root/.gradle/caches/modules-2/files-2.1/io.micrometer/micrometer-registry-new-relic/1.10.11/dec51ebe0ad3035e45f70a7b98fdccb4711d2d14/micrometer-registry-new-relic-1.10.11.jar:/root/.gradle/caches/modules-2/files-2.1/io.micrometer/micrometer-registry-otlp/1.10.11/5c03a8c86041f9da18401dc099d1fd6cf5c18277/micrometer-registry-otlp-1.10.11.jar:/root/.gradle/caches/modules-2/files-2.1/io.micrometer/micrometer-registry-prometheus/1.10.11/64c4538fcfa1a2a1b2cd373da1a115edb542a007/micrometer-registry-prometheus-1.10.11.jar:/root/.gradle/caches/modules-2/files-2.1/io.micrometer/micrometer-registry-signalfx/1.10.11/1d336dffe83be3abb0e5d38a4934455f8f41b0bc/micrometer-registry-signalfx-1.10.11.jar:/root/.gradle/caches/modules-2/files-2.1/io.micrometer/micrometer-registry-statsd/1.10.11/13fba2aa5c59675adad289e219235bae8ac4787b/micrometer-registry-statsd-1.10.11.jar:/root/.gradle/caches/modules-2/files-2.1/org.springframework.batch/spring-batch-core/5.0.3/95e97bf0a05b04f1758d8136be99bc08ab6e1a28/spring-batch-core-5.0.3.jar:/root/.gradle/caches/modules-2/files-2.1/org.apache.activemq/artemis-jakarta-server/2.26.0/189c6bc4db41d7911a02eb9ffaed21d0f73c753a/artemis-jakarta-server-2.26.0.jar:/root/.gradle/caches/modules-2/files-2.1/org.apache.activemq/artemis-server/2.26.0/5f8913832cb75a348aece1c6c44982fd92acc5d2/artemis-server-2.26.0.jar:/root/.gradle/caches/modules-2/files-2.1/io.micrometer/micrometer-core/1.10.11/adbe9b99be2e5cb4f835dd2134c5f66c69f23b3a/micrometer-core-1.10.11.jar:/root/.gradle/caches/modules-2/files-2.1/io.micrometer/micrometer-observation-test/1.10.11/a3ab2324d86645d101195898f10a85eb812851d8/micrometer-observation-test-1.10.11.jar:/root/.gradle/caches/modules-2/files-2.1/io.micrometer/micrometer-tracing-bridge-brave/1.0.10/c8bc94060a44de50fe4361f33d25203f61661296/micrometer-tracing-bridge-brave-1.0.10.jar:/root/.gradle/caches/modules-2/files-2.1/io.micrometer/micrometer-tracing-bridge-otel/1.0.10/de6b7c6f883d25c478893129355294ae09266eed/micrometer-tracing-bridge-otel-1.0.10.jar:/root/.gradle/caches/modules-2/files-2.1/io.micrometer/micrometer-tracing/1.0.10/7fdc5e755fd5cb99ab671891534eec1cfe88f102/micrometer-tracing-1.0.10.jar:/root/.gradle/caches/modules-2/files-2.1/org.springframework.amqp/spring-rabbit/3.0.9/42aece71d12d8c7957c62ecf4bb792407cc10e1d/spring-rabbit-3.0.9.jar:/root/.gradle/caches/modules-2/files-2.1/org.springframework.integration/spring-integration-jmx/6.0.7/85dbe686b3e7b2d7cc0ac85579bc8a31afce71cb/spring-integration-jmx-6.0.7.jar:/root/.gradle/caches/modules-2/files-2.1/org.springframework.integration/spring-integration-core/6.0.7/457c15897f190eb524c7b1d26e46d7baea15dc26/spring-integration-core-6.0.7.jar:/root/.gradle/caches/modules-2/files-2.1/org.springframework/spring-webflux/6.0.12/d218ccd0f6c1f656a33064ee3bd0bdb841535e6a/spring-webflux-6.0.12.jar:/root/.gradle/caches/modules-2/files-2.1/org.springframework/spring-webmvc/6.0.12/c7cf897d23da555a20e570d170f8f1541b04564d/spring-webmvc-6.0.12.jar:/root/.gradle/caches/modules-2/files-2.1/org.springframework.security/spring-security-oauth2-resource-server/6.0.7/6903eeb775645d55a02b7f98c32534fd271a7801/spring-security-oauth2-resource-

server-6.0.7.jar:/root/.gradle/caches/modules-2/files-  
2.1/org.springframework.security/spring-security-saml2-service-  
provider/6.0.7/4181de79b5e57f93e8058e284fa134b243b28b37/spring-security-saml2-service-  
provider-6.0.7.jar:/root/.gradle/caches/modules-2/files-  
2.1/org.springframework.security/spring-security-  
test/6.0.7/69de313b12214c399ecce783a523c541736a1d25/spring-security-test-  
6.0.7.jar:/root/.gradle/caches/modules-2/files-  
2.1/org.springframework.security/spring-security-  
web/6.0.7/526d3799d2076a0202506ab9d497da80b6b14fd0/spring-security-web-  
6.0.7.jar:/root/.gradle/caches/modules-2/files-  
2.1/org.springframework.security/spring-security-oauth2-  
jose/6.0.7/e7f8fdfae940b839f2abcde50b7a09b5150b93af/spring-security-oauth2-jose-  
6.0.7.jar:/root/.gradle/caches/modules-2/files-  
2.1/org.springframework.security/spring-security-oauth2-  
core/6.0.7/606a4bf621165c6a907b50e63bb5a42c8a003cd3/spring-security-oauth2-core-  
6.0.7.jar:/root/.gradle/caches/modules-2/files-2.1/org.springframework.hateoas/spring-  
hateoas/2.0.6/d9f113cd537513fa09e13f73fc67d45230c6f3e1/spring-hateoas-  
2.0.6.jar:/root/.gradle/caches/modules-2/files-2.1/org.springframework/spring-  
web/6.0.12/89a20bbd7c1f973dc246b1d790b34e0b3e28e74d/spring-web-  
6.0.12.jar:/root/.gradle/caches/modules-2/files-  
2.1/org.springframework.security/spring-security-  
config/6.0.7/4e7af44dc553f165a32352b01ef77e90e7b4850e/spring-security-config-  
6.0.7.jar:/root/.gradle/caches/modules-2/files-  
2.1/org.springframework.security/spring-security-  
core/6.0.7/f434936409601e7ca926ee60b713083b3b091c83/spring-security-core-  
6.0.7.jar:/root/.gradle/caches/modules-2/files-2.1/io.micrometer/micrometer-  
observation/1.10.11/884b41886e44f2f109ea628330aeeaf0a21549e7/micrometer-observation-  
1.10.11.jar:/root/.gradle/caches/modules-2/files-  
2.1/io.prometheus/simpleclient\_pushgateway/0.16.0/65935d9855ece6f85c21ad38634703d0917b  
f88c/simpleclient\_pushgateway-0.16.0.jar:/root/.gradle/caches/modules-2/files-  
2.1/io.zipkin.reporter2/zipkin-sender-  
urlconnection/2.16.3/8ea6090584755a4040491e9407b8858241259430/zipkin-sender-  
urlconnection-2.16.3.jar:/root/.gradle/caches/modules-2/files-  
2.1/io.opentelemetry/opentelemetry-exporter-  
zipkin/1.19.0/511b67a4cad80618944c2edad33ed5f58750f19e/opentelemetry-exporter-zipkin-  
1.19.0.jar:/root/.gradle/caches/modules-2/files-2.1/io.projectreactor.netty/reactor-  
netty-http/1.1.11/c9397212cfbfd4755026dcc5d048adf48819464/reactor-netty-http-  
1.1.11.jar:/root/.gradle/caches/modules-2/files-2.1/io.r2dbc/r2dbc-  
pool/1.0.1.RELEASE/41ec552a21fc0884dfb85335bd3f4c757a269ab1/r2dbc-pool-  
1.0.1.RELEASE.jar:/root/.gradle/caches/modules-2/files-2.1/io.r2dbc/r2dbc-  
h2/1.0.0.RELEASE/b99b52c87e7f32136f58131ad7b7a3e2eb168f75/r2dbc-h2-  
1.0.0.RELEASE.jar:/root/.gradle/caches/modules-2/files-2.1/io.r2dbc/r2dbc-  
spi/1.0.0.RELEASE/e3d15b2d27fdb8fdb76a181b21d5c752adf1d165/r2dbc-spi-  
1.0.0.RELEASE.jar:/root/.gradle/caches/modules-2/files-  
2.1/org.apache.activemq/artemis-jakarta-service-  
extensions/2.26.0/6953213dfc129dd12264b658687577a16f49870b/artemis-jakarta-service-  
extensions-2.26.0.jar:/root/.gradle/caches/modules-2/files-  
2.1/org.apache.activemq/artemis-jakarta-  
client/2.26.0/e4d3f9ebf7f0db83d9e28c316194e7efc75196ca/artemis-jakarta-client-  
2.26.0.jar:/root/.gradle/caches/modules-2/files-2.1/jakarta.jms/jakarta.jms-  
api/3.1.0/e194cf91a3f908e4846542849ac11a8e0b3c68ad/jakarta.jms-api-

3.1.0.jar:/root/.gradle/caches/modules-2/files-2.1/org.hibernate.orm/hibernate-core/6.1.7.Final/343f47b34c96fe9c44bf9b219a7b3c5d6d2fc90e/hibernate-core-6.1.7.Final.jar:/root/.gradle/caches/modules-2/files-2.1/jakarta.persistence/jakarta.persistence-api/3.1.0/66901fa1c373c6aff65c13791cc11da72060a8d6/jakarta.persistence-api-3.1.0.jar:/root/.gradle/caches/modules-2/files-2.1/io.undertow/undertow-servlet/2.3.8.Final/449114338a6d490df019f2c883794dc27bb5fde3/undertow-servlet-2.3.8.Final.jar:/root/.gradle/caches/modules-2/files-2.1/jakarta.servlet/jakarta.servlet-api/6.0.0/abecc699286e65035ebba9844c03931357a6a963/jakarta.servlet-api-6.0.0.jar:/root/.gradle/caches/modules-2/files-2.1/javax.cache/cache-api/1.1.1/c56fb980eb5208bfee29a9a5b9d951aba076bd91/cache-api-1.1.1.jar:/root/.gradle/caches/modules-2/files-2.1/org.apache.activemq/artemis-jdbc-store/2.26.0/690cebda6f22a0b7a29141fd6f151bbb61cc38fd/artemis-jdbc-store-2.26.0.jar:/root/.gradle/caches/modules-2/files-2.1/org.apache.commons/commons-dbcp2/2.9.0/16d808749cf3dac900c073dd834b5e288562a59c/commons-dbcp2-2.9.0.jar:/root/.gradle/caches/modules-2/files-2.1/org.apache.kafka/kafka-clients/3.3.2/be7a3c50a4026822671bdf6b53e3987149f951f/kafka-clients-3.3.2.jar:/root/.gradle/caches/modules-2/files-2.1/org.apache.logging.log4j/log4j-to-slf4j/2.19.0/30f4812e43172ecca5041da2cb6b965cc4777c19/log4j-to-slf4j-2.19.0.jar:/root/.gradle/caches/modules-2/files-2.1/org.apache.logging.log4j/log4j-api/2.19.0/ea1b37f38c327596b216542bc636cfdc0b8036fa/log4j-api-2.19.0.jar:/root/.gradle/caches/modules-2/files-2.1/org.apache.tomcat.embed/tomcat-embed-core/10.1.13/6909967f2ed6c323108c2cc7f20586d6f7eb6455/tomcat-embed-core-10.1.13.jar:/root/.gradle/caches/modules-2/files-2.1/org.apache.tomcat.embed/tomcat-embed-el/10.1.13/22c8845f7528334905c582e48a3eeefab616b0a5/tomcat-embed-el-10.1.13.jar:/root/.gradle/caches/modules-2/files-2.1/org.apache.tomcat/tomcat-jdbc/10.1.13/4b1924609bbabd1ad22141f6837b5ca3ece028f4/tomcat-jdbc-10.1.13.jar:/root/.gradle/caches/modules-2/files-2.1/org.springframework/spring-aspects/6.0.12/f9f903318add53a20c0e33ead1b16abd669543f7/spring-aspects-6.0.12.jar:/root/.gradle/caches/modules-2/files-2.1/org.aspectj/aspectjweaver/1.9.20/da562407e43f74c0f8f5f5df4065d85ec1736d01/aspectjweaver-1.9.20.jar:/root/.gradle/caches/modules-2/files-2.1/org.cache2k/cache2k-spring/2.6.1.Final/cd312efb1645de91bdd8571b98273cb0acf52d15/cache2k-spring-2.6.1.Final.jar:/root/.gradle/caches/modules-2/files-2.1/org.eclipse.angus/angus-mail/1.0.0/fa0a22df6dae0a1f081ba56a60704627b86d1f22/angus-mail-1.0.0.jar:/root/.gradle/caches/modules-2/files-2.1/org.eclipse.jetty/jetty-webapp/11.0.16/577eaca3b51ece5ebfb52731e0cc16e25eeb6a3b/jetty-webapp-11.0.16.jar:/root/.gradle/caches/modules-2/files-2.1/org.eclipse.jetty/jetty-servlet/11.0.16/fdd0839910e7df7e6890bf6d0383c71f803bd1bc/jetty-servlet-11.0.16.jar:/root/.gradle/caches/modules-2/files-2.1/org.eclipse.jetty/jetty-security/11.0.16/56a82424ae86c777897c7de5bdfd99264fc28935/jetty-security-11.0.16.jar:/root/.gradle/caches/modules-2/files-2.1/org.eclipse.jetty/jetty-server/11.0.16/a3807b87c1e513a9b8bf0e1ce449cad4c9354bf3/jetty-server-11.0.16.jar:/root/.gradle/caches/modules-2/files-2.1/co.elastic.clients/elasticsearch-java/8.5.3/fb0816b37e9b701fc81d8f9963c9d6e5a7280fee/elasticsearch-java-8.5.3.jar:/root/.gradle/caches/modules-2/files-2.1/org.elasticsearch.client/elasticsearch-rest-client/8.5.3/b6ea52ce5ea514aafae553fc8e10ac1138c600d/elasticsearch-rest-client-8.5.3.jar:/root/.gradle/caches/modules-2/files-2.1/org.flywaydb/flyway-core/9.5.1/31b91966d200d2a827182ba14cfb8917e8dabb38/flyway-core-

9.5.1.jar:/root/.gradle/caches/modules-2/files-2.1/org.glassfish.jersey.ext/jersey-spring6/3.1.3/1b4ba8c9f9d43a201f0514290672cd59dae8132/jersey-spring6-3.1.3.jar:/root/.gradle/caches/modules-2/files-2.1/org.glassfish.jersey.containers/jersey-container-servlet-core/3.1.3/b583f9ac773bdbc87a8a3ac03fa407f1175b185f/jersey-container-servlet-core-3.1.3.jar:/root/.gradle/caches/modules-2/files-2.1/org.glassfish.jersey.core/jersey-server/3.1.3/3108bec121e3c564cad7e8c7cb53835331bd654f/jersey-server-3.1.3.jar:/root/.gradle/caches/modules-2/files-2.1/org.hibernate.validator/hibernate-validator/8.0.1.Final/e49e116b3d3928060599b176b3538bb848718e95/hibernate-validator-8.0.1.Final.jar:/root/.gradle/caches/modules-2/files-2.1/org.influxdb/influxdb-java/2.23/5fd74d5ef7cc5c731f82d5fb9c7e0b6b6a94da77/influxdb-java-2.23.jar:/root/.gradle/caches/modules-2/files-2.1/org.liquibase/liquibase-core/4.17.2/fcc477e2ad657843424aef37d2ae86fba8804431/liquibase-core-4.17.2.jar:/root/.gradle/caches/modules-2/files-2.1/org.mongodb/mongodb-driver-reactivestreams/4.8.2/5d58a5bebbfd879306227fc9d6a66dfd260dedc2/mongodb-driver-reactivestreams-4.8.2.jar:/root/.gradle/caches/modules-2/files-2.1/org.mongodb/mongodb-driver-sync/4.8.2/342f6c34a7323f6f16fd5f4f41e156d47b6807f0/mongodb-driver-sync-4.8.2.jar:/root/.gradle/caches/modules-2/files-2.1/org.neo4j.driver/neo4j-java-driver/5.12.0/80cf488db020ca5cf8edadb384c1a652e80d33a/neo4j-java-driver-5.12.0.jar:/root/.gradle/caches/modules-2/files-2.1/org.quartz-scheduler/quartz/2.3.2/18a6d6b5a40b77bd060b34cb9f2acadc4bae7c8a/quartz-2.3.2.jar:/root/.gradle/caches/modules-2/files-2.1/org.springframework.data/spring-data-jpa/3.0.10/18fb82ea3208f7481be5abe3718fa0537b9710d0/spring-data-jpa-3.0.10.jar:/root/.gradle/caches/modules-2/files-2.1/org.springframework/spring-orm/6.0.12/174d73f69d878fd7ddeef9a25213d8932692a768/spring-orm-6.0.12.jar:/root/.gradle/caches/modules-2/files-2.1/org.springframework/spring-jdbc/6.0.12/cf14ce5078ab4b69b760ac91b2b9f23dab1e4457/spring-jdbc-6.0.12.jar:/root/.gradle/caches/modules-2/files-2.1/org.springframework/spring-jms/6.0.12/fc5244012f7e58802f42ab8693ddd76ac2191dc3/spring-jms-6.0.12.jar:/root/.gradle/caches/modules-2/files-2.1/org.springframework/spring-messaging/6.0.12/cef9bb726474ac699762aadcbbf0e8da81bec48f/spring-messaging-6.0.12.jar:/root/.gradle/caches/modules-2/files-2.1/org.springframework.data/spring-data-ldap/3.0.10/45cc29a062122765e639b1fbbab54298d20fbeb2/spring-data-ldap-3.0.10.jar:/root/.gradle/caches/modules-2/files-2.1/org.springframework.data/spring-data-mongodb/4.0.10/dc243e2a303be794ecfc1cfcc480d2e8fe9293a/spring-data-mongodb-4.0.10.jar:/root/.gradle/caches/modules-2/files-2.1/org.springframework.data/spring-data-redis/3.0.10/9e6a88ba9ab98bd44dab41e0031ce99a33b009c6/spring-data-redis-3.0.10.jar:/root/.gradle/caches/modules-2/files-2.1/org.springframework.graphql/spring-graphql/1.1.7/c7cdc211ae3173188a8027559ebb42431855f5fa/spring-graphql-1.1.7.jar:/root/.gradle/caches/modules-2/files-2.1/org.springframework.session/spring-session-core/3.0.3/f93d77e32978357be4ef5fe5f4b7d1e9c0efe9fd/spring-session-core-3.0.3.jar:/root/.gradle/caches/modules-2/files-2.1/redis.clients/jedis/4.3.2/3978900911b0b16bc30153d60d5f32d2cf04a464/jedis-4.3.2.jar:/root/.gradle/caches/modules-2/files-2.1/io.projectreactor/reactor-test/3.5.10/922b5f60425d9d5bfc427928be86b2c5b18d2c30/reactor-test-3.5.10.jar:/root/.gradle/caches/modules-2/files-2.1/com.squareup.okhttp3/mockwebserver/4.10.0/59da7fa7d338bfccee2ae831bee9ada9a1027363/mockwebserver-4.10.0.jar:/root/.gradle/caches/modules-2/files-2.1/com.jayway.jsonpath/json-path/2.7.0/f9d7d9659f2694e61142046ff8a216c047f263e8/json-

path-2.7.0.jar:/root/.gradle/caches/modules-2/files-2.1/io.undertow/undertow-  
 core/2.3.8.Final/c53e0a5a30a3619f5895e60898f8ce79e5666f9/undertow-core-  
 2.3.8.Final.jar:/root/.gradle/caches/modules-2/files-2.1/org.glassfish.jaxb/jaxb-  
 runtime/4.0.3/93af25be25b2c92c83e0ce61cb8b3ed23568f316/jaxb-runtime-  
 4.0.3.jar:/root/.gradle/caches/modules-2/files-2.1/org.glassfish.jaxb/jaxb-  
 core/4.0.3/e9093b4a82069a1d78ee9a3233ca387bca88861f/jaxb-core-  
 4.0.3.jar:/root/.gradle/caches/modules-2/files-2.1/jakarta.xml.bind/jakarta.xml.bind-  
 api/4.0.1/ca2330866cbc624c7e5ce982e121db1125d23e15/jakarta.xml.bind-api-  
 4.0.1.jar:/root/.gradle/caches/modules-2/files-  
 2.1/org.aspectj/aspectjrt/1.9.20/a950c7baef908470d0dff23a52cc805b87abf3a0/aspectjrt-  
 1.9.20.jar:/root/.gradle/caches/modules-2/files-2.1/org.springframework/spring-core-  
 test/6.0.12/18e3cb3ee62fb1df86422f81c9849c58d0473ad3/spring-core-test-  
 6.0.12.jar:/root/.gradle/caches/modules-2/files-2.1/org.assertj/assertj-  
 core/3.23.1/d2bb60570f5b3d7ffa8f8000118c9c07b86eca93/assertj-core-  
 3.23.1.jar:/root/.gradle/caches/modules-2/files-  
 2.1/org.awaitility/awaitility/4.2.0/2c39784846001a9cffd6c6b89c78de62c0d80fb8/awaitilit  
 y-4.2.0.jar:/root/.gradle/caches/modules-2/files-2.1/org.cache2k/cache2k-  
 core/2.6.1.Final/7e333caaafa2bf4e489b58537b5c4218a4e6505/cache2k-core-  
 2.6.1.Final.jar:/root/.gradle/caches/modules-2/files-2.1/org.cache2k/cache2k-  
 api/2.6.1.Final/6afd23d7897fff56515f0cefa6a846bdebe10bce/cache2k-api-  
 2.6.1.Final.jar:/root/.gradle/caches/modules-2/files-2.1/org.hamcrest/hamcrest-  
 library/2.2/cf530c8a0bc993487c64e940ae639bb4a6104dc6/hamcrest-library-  
 2.2.jar:/root/.gradle/caches/modules-2/files-  
 2.1/junit/junit/4.13.1/cdd00374f1fee76b11e2a9d127405aa3f6be5b6a/junit-  
 4.13.1.jar:/root/.gradle/caches/modules-2/files-2.1/org.hamcrest/hamcrest-  
 core/2.2/3f2bd07716a31c395e2837254f37f21f0f0ab24b/hamcrest-core-  
 2.2.jar:/root/.gradle/caches/modules-2/files-  
 2.1/org.hamcrest/hamcrest/2.2/1820c0968dba3a11a1b30669bb1f01978a91dedc/hamcrest-  
 2.2.jar:/root/.gradle/caches/modules-2/files-  
 2.1/org.hsqldb/hsqldb/2.7.2/d92d4d2aa515714da2165c9d640d584c2896c9df/hsqldb-  
 2.7.2.jar:/root/.gradle/caches/modules-2/files-2.1/org.junit.platform/junit-platform-  
 launcher/1.9.3/485650bfe2a2f39b606a6ca013285afda2eaae79/junit-platform-launcher-  
 1.9.3.jar:/root/.gradle/caches/modules-2/files-2.1/org.mockito/mockito-junit-  
 jupiter/4.8.1/e393aa62eca2244a535b03842843f2f199343d1f/mockito-junit-jupiter-  
 4.8.1.jar:/root/.gradle/caches/modules-2/files-2.1/org.junit.jupiter/junit-jupiter-  
 params/5.9.3/9e2a4bf6016a1975f408a73523392875cff7c26f/junit-jupiter-params-  
 5.9.3.jar:/root/.gradle/caches/modules-2/files-2.1/org.junit.jupiter/junit-jupiter-  
 engine/5.9.3/355322b03bf39306a183162cd06626c206f0286b/junit-jupiter-engine-  
 5.9.3.jar:/root/.gradle/caches/modules-2/files-2.1/org.junit.jupiter/junit-jupiter-  
 api/5.9.3/815818ad6ffcc8d320d8fbd3d748c753cf83201/junit-jupiter-api-  
 5.9.3.jar:/root/.gradle/caches/modules-2/files-2.1/org.junit.platform/junit-platform-  
 engine/1.9.3/8616734a190f8d307376aeb7353dba0a2c037a09/junit-platform-engine-  
 1.9.3.jar:/root/.gradle/caches/modules-2/files-2.1/org.junit.platform/junit-platform-  
 commons/1.9.3/36b2e26a90c41603be7f0094bee80e3f8a2cd4d4/junit-platform-commons-  
 1.9.3.jar:/root/.gradle/caches/modules-2/files-2.1/org.junit.jupiter/junit-  
 jupiter/5.9.3/72e840501e1550e9799c9a5cc9483d7d6b29e0ba/junit-jupiter-  
 5.9.3.jar:/root/.gradle/caches/modules-2/files-2.1/org.mockito/mockito-  
 core/4.8.1/d8eb9dec8747d08645347bb8c69088ac83197975/mockito-core-  
 4.8.1.jar:/root/.gradle/caches/modules-2/files-  
 2.1/org.skyscreamer/jsonassert/1.5.1/6d842d0faf4cf6725c509a5e5347d319ee0431c3/jsonasse  
 rt-1.5.1.jar:/root/.gradle/caches/modules-2/files-

2.1/org.yaml/snakeyaml/1.33/2cd0a87ff7df953f810c344bdf2fe3340b954c69/snakeyaml-  
 1.33.jar:/root/.gradle/caches/modules-2/files-  
 2.1/jakarta.management.j2ee/jakarta.management.j2ee-  
 api/1.1.4/dbbe7575f97efd0b04f3a8455cf82c256c853055/jakarta.management.j2ee-api-  
 1.1.4.jar:/root/.gradle/caches/modules-2/files-  
 2.1/jakarta.transaction/jakarta.transaction-  
 api/2.0.1/51a520e3fae406abb84e2e1148e6746ce3f80a1a/jakarta.transaction-api-  
 2.0.1.jar:/root/.gradle/caches/modules-2/files-2.1/org.opensaml/opensaml-saml-  
 impl/4.0.1/995986fd848ede1443469f3aff1f82b740224262/opensaml-saml-impl-  
 4.0.1.jar:/root/.gradle/caches/modules-2/files-2.1/org.opensaml/opensaml-saml-  
 api/4.0.1/2205aba935f4da468382a3dc5f32c3821ec1564c/opensaml-saml-api-  
 4.0.1.jar:/root/.gradle/caches/modules-2/files-2.1/org.opensaml/opensaml-soap-  
 impl/4.0.1/38bfaf5fc189774e94ead218bd1c754da295c226/opensaml-soap-impl-  
 4.0.1.jar:/root/.gradle/caches/modules-2/files-2.1/org.opensaml/opensaml-profile-  
 api/4.0.1/bece5f6d30d4051e6eeaf2b88dd1e5a13f6b28b7/opensaml-profile-api-  
 4.0.1.jar:/root/.gradle/caches/modules-2/files-2.1/org.opensaml/opensaml-soap-  
 api/4.0.1/d8e11e31cb5164788a530478e1831969e94a38b6/opensaml-soap-api-  
 4.0.1.jar:/root/.gradle/caches/modules-2/files-2.1/org.opensaml/opensaml-xmlsec-  
 impl/4.0.1/efa15ba85127ac3b20c75b8d4f04c7e92325a00a/opensaml-xmlsec-impl-  
 4.0.1.jar:/root/.gradle/caches/modules-2/files-2.1/org.opensaml/opensaml-xmlsec-  
 api/4.0.1/edb4365d3d183933cf0d0b31966ea352b8d20c60/opensaml-xmlsec-api-  
 4.0.1.jar:/root/.gradle/caches/modules-2/files-2.1/org.opensaml/opensaml-security-  
 impl/4.0.1/64568e9aa8bd7bcd76983e462f9eb2c3dcacbdce/opensaml-security-impl-  
 4.0.1.jar:/root/.gradle/caches/modules-2/files-2.1/org.opensaml/opensaml-security-  
 api/4.0.1/f3d33ca18cde2a7c7e3643aeca9f03974be9577d/opensaml-security-api-  
 4.0.1.jar:/root/.gradle/caches/modules-2/files-2.1/org.opensaml/opensaml-messaging-  
 api/4.0.1/eb9c9971f6bd2a6681a2a692a1f29a35874de389/opensaml-messaging-api-  
 4.0.1.jar:/root/.gradle/caches/modules-2/files-2.1/org.opensaml/opensaml-  
 core/4.0.1/ec3d1734137d6ccabba7d6d5e149f571beaa673/opensaml-core-  
 4.0.1.jar:/root/.gradle/caches/modules-2/files-2.1/org.springframework/spring-context-  
 support/6.0.12/6225f907b13d5a2240af91e354d71163d41b0307/spring-context-support-  
 6.0.12.jar:/root/.gradle/caches/modules-2/files-2.1/org.springframework.data/spring-  
 data-keyvalue/3.0.10/c654e47757af0dbe8970047d0bed685cb61b8ff0/spring-data-keyvalue-  
 3.0.10.jar:/root/.gradle/caches/modules-2/files-2.1/org.springframework.plugin/spring-  
 plugin-core/3.0.0/d56aa02dd7272dca30aa598dc8b72e823227046a/spring-plugin-core-  
 3.0.0.jar:/root/.gradle/caches/modules-2/files-2.1/org.springframework/spring-  
 context/6.0.12/f1964518e07796b2fdc1b9be2108485ffe1b4566/spring-context-  
 6.0.12.jar:/root/.gradle/caches/modules-2/files-2.1/org.springframework/spring-  
 test/6.0.12/45ba109a7e19561dbbad64c9d033f2fecb434e85/spring-test-  
 6.0.12.jar:/root/.gradle/caches/modules-2/files-2.1/org.springframework.ldap/spring-  
 ldap-core/3.0.5/c3bd3edd735ea7a65272d666de44a13933e2ba22/spring-ldap-core-  
 3.0.5.jar:/root/.gradle/caches/modules-2/files-2.1/org.springframework/spring-  
 tx/6.0.12/b4e997baac7dd14593f3e4f815e3646279e9a63f/spring-tx-  
 6.0.12.jar:/root/.gradle/caches/modules-2/files-2.1/org.springframework.amqp/spring-  
 amqp/3.0.9/d25b1c1e3af5de24c1afb327cb196981d0aa6756/spring-amqp-  
 3.0.9.jar:/root/.gradle/caches/modules-2/files-2.1/org.springframework/spring-  
 aop/6.0.12/ba7ecdbdc2abaf172d49692b8110f65ecd2d250c/spring-aop-  
 6.0.12.jar:/root/.gradle/caches/modules-2/files-2.1/org.springframework.data/spring-  
 data-commons/3.0.10/50a12ab567c19c34ce02f0d2180f81bc7872789/spring-data-commons-  
 3.0.10.jar:/root/.gradle/caches/modules-2/files-2.1/org.springframework/spring-  
 oxml/6.0.12/e5f6a5169284a237f19091e8acf100fba97eec07/spring-oxm-

6.0.12.jar:/root/.gradle/caches/modules-2/files-2.1/org.springframework/spring-beans/6.0.12/dfa53afbf1ab65f36c60333443ed1109ed331504/spring-beans-6.0.12.jar:/root/.gradle/caches/modules-2/files-2.1/org.springframework.batch/spring-batch-infrastructure/5.0.3/551c4bfc2c79259eff67e3e03ad7c40f5ce80ec1/spring-batch-infrastructure-5.0.3.jar:/root/.gradle/caches/modules-2/files-2.1/org.springframework/spring-expression/6.0.12/9283ad418f06f918e834b1981998a5f62a05b355/spring-expression-6.0.12.jar:/root/.gradle/caches/modules-2/files-2.1/org.springframework/spring-core/6.0.12/a55e70359371b03ee29dacd75c3b40d0baf3a656/spring-core-6.0.12.jar:/root/.gradle/caches/modules-2/files-2.1/com.vaadin.external.google/android-json/0.0.20131108.vaadin1/fa26d351fe62a6a17f5cda1287c1c6110dec413f/android-json-0.0.20131108.vaadin1.jar:/root/.gradle/caches/modules-2/files-2.1/com.mchange/c3p0/0.9.5.5/37dfc3021e5589d65ff2ae0becf811510b87ab01/c3p0-0.9.5.5.jar:/root/.gradle/caches/modules-2/files-2.1/org.glassfish.jersey.core/jersey-client/3.1.3/d701564333d25b3cc9ac95756996dab21a8cae43/jersey-client-3.1.3.jar:/root/.gradle/caches/modules-2/files-2.1/org.glassfish.jersey.inject/jersey-hk2/3.1.3/594b23af6a09f621c16e94b85901aa3738455f9c/jersey-hk2-3.1.3.jar:/root/.gradle/caches/modules-2/files-2.1/org.glassfish.jersey.core/jersey-common/3.1.3/b4a091d46ec7d0235177338419f5c5d2d2d89039/jersey-common-3.1.3.jar:/root/.gradle/caches/modules-2/files-2.1/org.glassfish.hk2/spring-bridge/3.0.4/88d29bb7b2a8ef666c8a9dd27432ac4407d4c14b/spring-bridge-3.0.4.jar:/root/.gradle/caches/modules-2/files-2.1/org.glassfish.hk2/hk2/3.0.4/5d9052b9f58fc5606b9d2106e359b306ed946a3a/hk2-3.0.4.jar:/root/.gradle/caches/modules-2/files-2.1/org.glassfish.hk2/hk2-core/3.0.4/a4e36a6578b50ab4d8efa5ebe8c365c03b76330c/hk2-core-3.0.4.jar:/root/.gradle/caches/modules-2/files-2.1/org.glassfish.hk2/hk2-runlevel/3.0.4/e22a729485e7b4c30e25dd6f963a87f5502f9077/hk2-runlevel-3.0.4.jar:/root/.gradle/caches/modules-2/files-2.1/org.glassfish.hk2/hk2-locator/3.0.4/8ac6800f4102d2961b2d4380a483a0374a5a32dc/hk2-locator-3.0.4.jar:/root/.gradle/caches/modules-2/files-2.1/org.glassfish.hk2/hk2-api/3.0.4/ec406de7356ace192ba536294b7424194a7f44ab/hk2-api-3.0.4.jar:/root/.gradle/caches/modules-2/files-2.1/org.glassfish.hk2/hk2-utils/3.0.4/d33f91938dbbb239ab026e37cb75afbd16090895/hk2-utils-3.0.4.jar:/root/.gradle/caches/modules-2/files-2.1/jakarta.inject/jakarta.inject-api/2.0.1/4c28afe1991a941d7702fe1362c365f0a8641d1e/jakarta.inject-api-2.0.1.jar:/root/.gradle/caches/modules-2/files-2.1/org.apache.maven/maven-resolver-provider/3.9.4/d5e38634c1251e000f7718437973a730346bc336/maven-resolver-provider-3.9.4.jar:/root/.gradle/caches/modules-2/files-2.1/org.apache.maven/maven-model-builder/3.9.4/b12e45736ddfb51f648e1579877a293659e5a96e/maven-model-builder-3.9.4.jar:/root/.gradle/caches/modules-2/files-2.1/org.apache.maven.resolver/maven-resolver-connector-basic/1.9.14/ffeb368f5587513b29ef9771514ec36d246e65f5/maven-resolver-connector-basic-1.9.14.jar:/root/.gradle/caches/modules-2/files-2.1/org.apache.maven.resolver/maven-resolver-impl/1.9.14/2aa090aca572f7061e5c57f714ef0445bbb4555f/maven-resolver-impl-1.9.14.jar:/root/.gradle/caches/modules-2/files-2.1/org.apache.maven.resolver/maven-resolver-transport-http/1.9.14/1eb4acdce7f21526bd06654b3d7b1edf80370ce8/maven-resolver-transport-http-1.9.14.jar:/root/.gradle/caches/modules-2/files-2.1/org.apache.maven.resolver/maven-resolver-spi/1.9.14/a190c3d49791240c87799f29848505fc9fd8be11/maven-resolver-spi-1.9.14.jar:/root/.gradle/caches/modules-2/files-2.1/org.apache.maven.resolver/maven-

resolver-util/1.9.14/ea6fd2997903700191711dee4448746ed30fe4b9/maven-resolver-util-1.9.14.jar:/root/.gradle/caches/modules-2/files-2.1/org.apache.maven.resolver/maven-resolver-api/1.9.14/bd5b7cd9d15c2efcab4d8c8215e769523e8843e4/maven-resolver-api-1.9.14.jar:/root/.gradle/caches/modules-2/files-2.1/org.opensaml/opensaml-storage-api/4.0.1/4e46a7f965ac9f91976b0f298fd4d4e69e9056db/opensaml-storage-api-4.0.1.jar:/root/.gradle/caches/modules-2/files-2.1/net.shibboleth.utilities/java-support/8.0.0/298f946e93922d789b6231599a446cea9dbbe80e/java-support-8.0.0.jar:/root/.gradle/caches/modules-2/files-2.1/org.apache.santuario/xmlsec/2.1.4/cb43326f02e3e77526c24269c8b5d3cc3f7f6653/xmlsec-2.1.4.jar:/root/.gradle/caches/modules-2/files-2.1/com.google.cloud/google-cloud-monitoring/3.6.0/b77ce9d450026df14c83fdb05117af8fb189d712/google-cloud-monitoring-3.6.0.jar:/root/.gradle/caches/modules-2/files-2.1/com.google.auth/google-auth-library-oauth2-http/1.12.2/7df532f9c0309ed794f54dbf15f3ab4ad12cae0e/google-auth-library-oauth2-http-1.12.2.jar:/root/.gradle/caches/modules-2/files-2.1/com.google.http-client/google-http-client-gson/1.42.2/863fbfbdb35c203c43cefd9e3cc1e3a7c8e9f51e/google-http-client-gson-1.42.2.jar:/root/.gradle/caches/modules-2/files-2.1/com.google.http-client/google-http-client/1.42.2/6d6ab266972f0afc6987fe3b88ce321179515fa3/google-http-client-1.42.2.jar:/root/.gradle/caches/modules-2/files-2.1/org.apache.httpcomponents/httpclient/4.5.14/1194890e6f56ec29177673f2f12d0b8e627dec98/httpclient-4.5.14.jar:/root/.gradle/caches/modules-2/files-2.1/commons-codec/commons-codec/1.15/49d94806b6e3dc933dadb8acbf0fdbab8ebd1e5d/commons-codec-1.15.jar:/root/.gradle/caches/modules-2/files-2.1/com.github.spotbugs/spotbugs-annotations/3.1.12/ba2c77a05091820668987292f245f3b089387bfa/spotbugs-annotations-3.1.12.jar:/root/.gradle/caches/modules-2/files-2.1/io.opencensus/opencensus-contrib-http-util/0.31.1/3c13fc5715231fadb16a9b74a44d9d59c460cfa8/opencensus-contrib-http-util-0.31.1.jar:/root/.gradle/caches/modules-2/files-2.1/com.google.guava/guava/31.1-jre/60458f877d055d0c9114d9e1a2efb737b4bc282c/guava-31.1-jre.jar:/root/.gradle/caches/modules-2/files-2.1/com.google.api.grpc/proto-google-cloud-monitoring-v3/3.6.0/1c58c69975d82f7df3bb35f5c98c6b5340791c5/proto-google-cloud-monitoring-v3-3.6.0.jar:/root/.gradle/caches/modules-2/files-2.1/com.google.code.findbugs/jsr305/3.0.2/25ea2e8b0c338a877313bd4672d3fe056ea78f0d/jsr305-3.0.2.jar:/root/.gradle/caches/modules-2/files-2.1/io.dropwizard.metrics/metrics-graphite/4.2.19/c8ed09f56bca464c136066bedbac86f451aa7f20/metrics-graphite-4.2.19.jar:/root/.gradle/caches/modules-2/files-2.1/io.dropwizard.metrics/metrics-core/4.2.19/d32b4c3f3e733bf4cb239ca4204fbed8464973a5/metrics-core-4.2.19.jar:/root/.gradle/caches/modules-2/files-2.1/org.apache.velocity/velocity-engine-core/2.2/68d899cb70cd27d495562fa808feb2da4926d38f/velocity-engine-core-2.2.jar:/root/.gradle/caches/modules-2/files-2.1/org.apache.maven.resolver/maven-resolver-named-locks/1.9.14/71cf74d91b4f6de1bfff8fc1fafb401d9ac42b1b8/maven-resolver-named-locks-1.9.14.jar:/root/.gradle/caches/modules-2/files-2.1/com.zaxxer/HikariCP-java7/2.4.13/3e441eddedb374d4de8e3abbb0c90997f51cc97b/HikariCP-java7-2.4.13.jar:/root/.gradle/caches/modules-2/files-2.1/com.graphql-java/graphql-java/19.5/54d25d19b2c8c542b57de972e6a06a0db38c14e2/graphql-java-19.5.jar:/root/.gradle/caches/modules-2/files-2.1/org.eclipse.jetty/jetty-http/11.0.16/fa8f92516b912989f13121cdea1f993ef225d420/jetty-http-11.0.16.jar:/root/.gradle/caches/modules-2/files-2.1/org.eclipse.jetty/jetty-io/11.0.16/e814235e08bda52f7adc2fb1a13ec3b6252f0a5a/jetty-io-11.0.16.jar:/root/.gradle/caches/modules-2/files-2.1/org.eclipse.jetty/jetty-xml/11.0.16/f34ff8b7dfaae846179dd0fb8a588a09037dce0e/jetty-xml-11.0.16.jar:/root/.gradle/caches/modules-2/files-2.1/com.netflix.spectator/spectator-

reg-atlas/1.3.10/315325879b62adef03cce2b12bf78354c44aaab/spectator-reg-atlas-1.3.10.jar:/root/.gradle/caches/modules-2/files-2.1/com.rabbitmq/amqp-client/5.16.1/7b483b06e74566b4fa2349a22b95197036058eaf/amqp-client-5.16.1.jar:/root/.gradle/caches/modules-2/files-2.1/com.graphql-java/java-dataloader/3.2.0/f45c53595cab4c23e35526cc122e2bd159a50516/java-dataloader-3.2.0.jar:/root/.gradle/caches/modules-2/files-2.1/org.eclipse.jetty/jetty-util/11.0.16/7b2c96bcf6f26386cfff6fc37292af379bbeb1e8/jetty-util-11.0.16.jar:/root/.gradle/caches/modules-2/files-2.1/com.netflix.spectator/spectator-ext-ipc/1.3.10/449587c0a2852e88e35bdf6d1793d593b834ad44/spectator-ext-ipc-1.3.10.jar:/root/.gradle/caches/modules-2/files-2.1/com.netflix.spectator/spectator-api/1.3.10/dd43510b2bcf15f2e54ce4b70d77916b2e4699e8/spectator-api-1.3.10.jar:/root/.gradle/caches/modules-2/files-2.1/org.slf4j/slf4j-api/2.0.9/7cf2726fdcfbc8610f9a71fb3ed639871f315340/slf4j-api-2.0.9.jar:/root/.gradle/caches/modules-2/files-2.1/org.apache.httpcomponents/httpcore/4.4.16/51cf043c87253c9f58b539c9f7e44c8894223850/httpcore-4.4.16.jar:/root/.gradle/caches/modules-2/files-2.1/org.apache.activemq/artemis-core-client/2.26.0/2a88861eca6c35f665f7ec1528c4b936121b7dd6/artemis-core-client-2.26.0.jar:/root/.gradle/caches/modules-2/files-2.1/org.apache.activemq/artemis-selector/2.26.0/51cd085ced60666a1394355311f9149a0c0d9d4/artemis-selector-2.26.0.jar:/root/.gradle/caches/modules-2/files-2.1/org.apache.activemq/artemis-journal/2.26.0/5699e3890907c4b1c6a6735d360cce4ab0b81446/artemis-journal-2.26.0.jar:/root/.gradle/caches/modules-2/files-2.1/org.apache.activemq/artemis-commons/2.26.0/c0c955255922c40570b6bf814ae16e980a09bd31/artemis-commons-2.26.0.jar:/root/.gradle/caches/modules-2/files-2.1/org.apache.activemq/artemis-quorum-api/2.26.0/f4a22b1902e49ecca4c68667fa610905eba04e37/artemis-quorum-api-2.26.0.jar:/root/.gradle/caches/modules-2/files-2.1/net.bytebuddy/byte-buddy/1.12.23/d470526e8c4566c04e9ae5d3ccb62d1a7aa58986/byte-buddy-1.12.23.jar:/root/.gradle/caches/modules-2/files-2.1/net.bytebuddy/byte-buddy-agent/1.12.23/1cba11fdb72c383edacb909f79ae6870efd275e4/byte-buddy-agent-1.12.23.jar:/root/.gradle/caches/modules-2/files-2.1/com.fasterxml/classmate/1.5.1/3fe0bed568c62df5e89f4f174c101eab25345b6c/classmate-1.5.1.jar:/root/.gradle/caches/modules-2/files-2.1/com.opencsv/opencsv/5.7.1/d707c095bc8c7c22fb3e377de774458a76229da4/opencsv-5.7.1.jar:/root/.gradle/caches/modules-2/files-2.1/org.apache.maven/maven-artifact/3.9.4/9738999a0459221e758ebe728449c43567d7fecc/maven-artifact-3.9.4.jar:/root/.gradle/caches/modules-2/files-2.1/org.apache.commons/commons-configuration2/2.8.0/6a76acbe14d2c01d4758a57171f3f6a150dbd462/commons-configuration2-2.8.0.jar:/root/.gradle/caches/modules-2/files-2.1/org.apache.commons/commons-text/1.10.0/3363381aef8cef2dbc1023b3e3a9433b08b64e01/commons-text-1.10.0.jar:/root/.gradle/caches/modules-2/files-2.1/org.apache.commons/commons-lang3/3.12.0/c6842c86792ff03b9f1d1fe2aab8dc23aa6c6f0e/commons-lang3-3.12.0.jar:/root/.gradle/caches/modules-2/files-2.1/org.apache.commons/commons-pool2/2.11.1/8970fd110c965f285ed4c6e40be7630c62db6f68/commons-pool2-2.11.1.jar:/root/.gradle/caches/modules-2/files-2.1/com.couchbase.client/java-client/3.4.10/458521190085ad0c25eeb6fc6357082be199512e/java-client-3.4.10.jar:/root/.gradle/caches/modules-2/files-2.1/com.google.code.gson/gson/2.9.1/2cc2131b98ebfb04e2b2c7dfb84431f4045096b/gson-2.9.1.jar:/root/.gradle/caches/modules-2/files-2.1/com.h2database/h2/2.1.214/d5c2005c9e3279201e12d4776c948578b16bf8b2/h2-2.1.214.jar:/root/.gradle/caches/modules-2/files-

2.1/org.apache.httpcomponents/httpasyncclient/4.1.5/cd18227f1eb8e9a263286c1d7362ceb24f6f9b32/httpasyncclient-4.1.5.jar:/root/.gradle/caches/modules-2/files-  
 2.1/org.apache.httpcomponents/httpcore-  
 nio/4.4.16/cd21c80a9956be48c4c1cfd2f594ba02857d0927/httpcore-nio-  
 4.4.16.jar:/root/.gradle/caches/modules-2/files-2.1/org.eclipse.angus/angus-  
 activation/2.0.1/eaafaf4eb71b400e4136fc3a286f50e34a68ecb7/angus-activation-  
 2.0.1.jar:/root/.gradle/caches/modules-2/files-2.1/jakarta.mail/jakarta.mail-  
 api/2.1.2/e2f24836f1720a6b53d98240a78b4860597428c8/jakarta.mail-api-  
 2.1.2.jar:/root/.gradle/caches/modules-2/files-  
 2.1/jakarta.activation/jakarta.activation-  
 api/2.1.2/640c0d5aff45dbff1e1a1bc09673ff3a02b1ba12/jakarta.activation-api-  
 2.1.2.jar:/root/.gradle/caches/modules-2/files-  
 2.1/jakarta.annotation/jakarta.annotation-  
 api/2.1.1/48b9bda22b091b1f48b13af03fe36db3be6e1ae3/jakarta.annotation-api-  
 2.1.1.jar:/root/.gradle/caches/modules-2/files-  
 2.1/org.eclipse.parsson/parsson/1.0.0/2a3b84d40ad99500d8032312bb6b03c35018dcca/parsson-  
 1.0.0.jar:/root/.gradle/caches/modules-2/files-2.1/jakarta.json/jakarta.json-  
 api/2.1.2/cf0d33a43f05d4d88f30a6972d95115f81c550be/jakarta.json-api-  
 2.1.2.jar:/root/.gradle/caches/modules-2/files-  
 -2.1/jakarta.validation/jakarta.validation-  
 api/3.0.2/92b6631659ba35ca09e44874d3eb936edfeee532/jakarta.validation-api-  
 3.0.2.jar:/root/.gradle/caches/modules-2/files-2.1/org.glassfish.jersey.ext/jersey-  
 entity-filtering/3.1.3/796dc903d14d68a8cf3a4f81d546d50777a4179c/jersey-entity-  
 filtering-3.1.3.jar:/root/.gradle/caches/modules-2/files-  
 -2.1/jakarta.ws.rs/jakarta.ws.rs-  
 api/3.1.0/15ce10d249a38865b58fc39521f10f29ab0e3363/jakarta.ws.rs-api-  
 3.1.0.jar:/root/.gradle/caches/modules-2/files-2.1/org.jboss.xnio/xnio-  
 nio/3.8.8.Final/a5d2faf8d02e0a0bb9e9eabca8e38173cb640331/xnio-nio-  
 3.8.8.Final.jar:/root/.gradle/caches/modules-2/files-2.1/org.jboss.xnio/xnio-  
 api/3.8.8.Final/1ba9c8b9a8dea1c6cd656155943e6d4c2c631fa7/xnio-api-  
 3.8.8.Final.jar:/root/.gradle/caches/modules-2/files-2.1/org.wildfly.client/wildfly-  
 client-config/1.0.1.Final/2a803b23c40a0de0f03a90d1fd3755747bc05f4b/wildfly-client-  
 config-1.0.1.Final.jar:/root/.gradle/caches/modules-2/files-  
 -2.1/org.apache.activemq/activemq-artemis-  
 native/1.0.2/b6dbe8cfbeecea4f33be0170d3bbe655f200a557/activemq-artemis-native-  
 1.0.2.jar:/root/.gradle/caches/modules-2/files-2.1/org.jboss.threads/jboss-  
 threads/3.5.0.Final/dd23d4788b3eafe9597ef3fe028e46ceb293ba8d/jboss-threads-  
 3.5.0.Final.jar:/root/.gradle/caches/modules-2/files-2.1/org.jboss.logging/jboss-  
 logging/3.5.3.Final/c88fc1d8a96d4c3491f55d4317458ccad53ca663/jboss-logging-  
 3.5.3.Final.jar:/root/.gradle/caches/modules-2/files-2.1/net.minidev/json-  
 smart/2.4.11/cc5888f14a5768f254b97baf8b9fd29b31e872e/json-smart-  
 2.4.11.jar:/root/.gradle/caches/modules-2/files-2.1/ch.qos.logback/logback-  
 core/1.4.11/2f9f280219a9922a74200eaf7138c4c17fb87c0f/logback-core-  
 1.4.11.jar:/root/.gradle/caches/modules-2/files-2.1/org.mongodb/mongodb-driver-  
 core/4.8.2/7eefb22e3cd5992778ea6cd6e991e0a4070917a5/mongodb-driver-core-  
 4.8.2.jar:/root/.gradle/caches/modules-2/files-2.1/org.mongodb/bson-record-  
 codec/4.8.2/5297c3ba51109d4fa3c01e65217344a873f4ffa0/bson-record-codec-  
 4.8.2.jar:/root/.gradle/caches/modules-2/files-  
 -2.1/org.mongodb/bson/4.8.2/301cb293127b2d2ee9c25307ae34b44f7f1e4964/bson-  
 4.8.2.jar:/root/.gradle/caches/modules-2/files-2.1/io.projectreactor.addons/reactor-  
 pool/1.0.2/bf1cadeb213c70e695a8de9994617d93b15387e8/reactor-pool

-1.0.2.jar:/root/.gradle/caches/modules-2/files-2.1/io.projectreactor.netty/reactor-netty-core/1.1.11/2cefe0af4adfb4044ada513a97858c8654ee3351/reactor-netty-core  
 -1.1.11.jar:/root/.gradle/caches/modules-2/files-2.1/com.couchbase.client/core-io/2.4.10/cc581365e1069ba6fd11f7a7f86dbe8199dee39b/core-io  
 -2.4.10.jar:/root/.gradle/caches/modules-2/files-2.1/io.projectreactor/reactor-core/3.5.10/b2d043508fbc0190bfb63eea1b33551f48a8c32b/reactor-core  
 -3.5.10.jar:/root/.gradle/caches/modules-2/files-2.1/org.reactivestreams/reactive-streams/1.0.4/3864a1320d97d7b045f729a326e1e077661f31b7/reactive-streams  
 -1.0.4.jar:/root/.gradle/caches/modules-2/files-2.1/org.springframework.retry/spring-retry/2.0.3/4bd76e84c4c5225f9313f80c1cbfef7c608128d3/spring-retry  
 -2.0.3.jar:/root/.gradle/caches/modules-2/files-2.1/org.apache.tomcat/tomcat-annotations-api/10.1.13/d7e006f8e64bc72f9c84b070b0823b4ba8e1da66/tomcat-annotations-api-10.1.13.jar:/root/.gradle/caches/modules-2/files-2.1/org.apache.maven/maven-model/3.9.4/5b79d873cf6d13b3fc4020eac04e2a62ebbfa0aa/maven-model  
 -3.9.4.jar:/root/.gradle/caches/modules-2/files-2.1/org.apache.maven/maven-repository-metadata/3.9.4/ef368d45a2caa54634de221540994ca2d810c4c9/maven-repository-metadata  
 -3.9.4.jar:/root/.gradle/caches/modules-2/files-2.1/org.codehaus.plexus/plexus-utils/3.5.1/c6bfb17c97ecc8863e88778ea301be742c62b06d/plexus-utils  
 -3.5.1.jar:/root/.gradle/caches/modules-2/files-2.1/com.google.guava/failureaccess/1.0.1/1dcf1de382a0bf95a3d8b0849546c88bac1292c9/failureaccess-1.0.1.jar:/root/.gradle/caches/modules-2/files-2.1/com.google.guava/listenablefuture/9999.0-empty-to-avoid-conflict-with-guava/b421526c5f297295adebf1c886e5246c39d4ac629/listenablefuture-9999.0-empty-to-avoid-conflict-with-guava.jar:/root/.gradle/caches/modules-2/files-2.1/org.checkerframework/checker-qual/3.37.0/ba74746d38026581c12166e164bb3c15e90cc4ea/checker-qual-3.37.0.jar:/root/.gradle/caches/modules-2/files-2.1/com.google.errorprone/error\_prone\_annotations/2.21.1/6d9b10773b5237df178a7b3c1b4208df7d0e7f94/error\_prone\_annotations-2.21.1.jar:/root/.gradle/caches/modules-2/files-2.1/com.google.j2objc/j2objc-annotations/1.3/ba035118bc8bac37d7eff77700720999acd9986d/j2objc-annotations-1.3.jar:/root/.gradle/caches/modules-2/files-2.1/org.cryptacular/cryptacular/1.2.4/4994c015d87886212683245d13e87f6fb903a760/cryptacular-1.2.4.jar:/root/.gradle/caches/modules-2/files-2.1/org.bouncycastle/bcpkix-jdk15on/1.64/3dac163e20110817d850d17e0444852a6d7d0bd7/bcpkix-jdk15on-1.64.jar:/root/.gradle/caches/modules-2/files-2.1/org.bouncycastle/bcprov-jdk15on/1.64/1467dac1b787b5ad2a18201c0c281df69882259e/bcprov-jdk15on-1.64.jar:/root/.gradle/caches/modules-2/files-2.1/com.fasterxml.wdostox/woodstox-core/6.5.0/cafa8aac5ddf104d28f172f19294d88d5e8c24c0/woodstox-core-6.5.0.jar:/root/.gradle/caches/modules-2/files-2.1/io.zipkin.reporter2/zipkin-sender-okhttp3/2.16.3/c98cff5bc2fa32914e613efc3cc95dde9906c01b/zipkin-sender-okhttp3-2.16.3.jar:/root/.gradle/caches/modules-2/files-2.1/io.zipkin.aws/brave-propagation-aws/0.23.5/94dafc941dac5a7bf686292d3a3aebd579dec45c/brave-propagation-aws-0.23.5.jar:/root/.gradle/caches/modules-2/files-2.1/io.zipkin.brave/brave-context-slf4j/5.14.1/91352ad48e41a76f421079d3534b18536c94731/brave-context-slf4j-5.14.1.jar:/root/.gradle/caches/modules-2/files-2.1/io.zipkin.brave/brave-instrumentation-http/5.14.1/d069f9ba02cff93e36ef80612dae60b57ed6451b/brave-instrumentation-http-5.14.1.jar:/root/.gradle/caches/modules-2/files-2.1/io.zipkin.brave/brave/5.14.1/6a6ecd4e59ca191ca10313e48d1e41ed3e83d851/brave-5.14.1.jar:/root/.gradle/caches/modules-2/files-2.1/io.zipkin.reporter2/zipkin-reporter-brave/2.16.3/4d5017d71e4de139b6a31612cfd837b7c71d288c/zipkin-reporter-brave-

2.16.3.jar:/root/.gradle/caches/modules-2/files-2.1/io.zipkin.reporter2/zipkin-reporter/2.16.3/7e43d8be3376d305c355d969e8b9f3a62221380/zipkin-reporter-2.16.3.jar:/root/.gradle/caches/modules-2/files-2.1/io.zipkin.zipkin2/zipkin/2.23.2/1c2c7f2e91a3749311f7f75d0535d14ba2e2f6/zipkin-2.23.2.jar:/root/.gradle/caches/modules-2/files-2.1/com.datastax.oss.native-protocol/1.5.1/97e812373a5fe7667384e7ad67819d2c71878bf8/native-protocol-1.5.1.jar:/root/.gradle/caches/modules-2/files-2.1/com.datastax.oss/java-driver-shaded-guava/25.1-jre-graal-sub-1/522771d14d6b7dba67056a39db33f205ffbed6a4/java-driver-shaded-guava-25.1-jre-graal-sub-1.jar:/root/.gradle/caches/modules-2/files-2.1/org.glassfish.jaxb/txw2/4.0.3/47b8fe31c6d1a3382203af919400527389e01e9c/txw2-4.0.3.jar:/root/.gradle/caches/modules-2/files-2.1/com.sun.istack/istack-commons-runtime/4.1.2/18ec117c85f3ba0ac65409136afa8e42bc74e739/istack-commons-runtime-4.1.2.jar:/root/.gradle/caches/modules-2/files-2.1/com.squareup.okhttp3/logging-interceptor/4.10.0/ae7524eec42d4ab0c3a7cb93da010cf9bcc5007c/logging-interceptor-4.10.0.jar:/root/.gradle/caches/modules-2/files-2.1/com.squareup.retrofit2/converter-moshi/2.9.0/db0979801926e6d39bc2478736145f9761c3e034/converter-moshi-2.9.0.jar:/root/.gradle/caches/modules-2/files-2.1/com.squareup.retrofit2/retrofit/2.9.0/d8fd9bd5da952141a665a403348b74538efc05ff/retrofit-2.9.0.jar:/root/.gradle/caches/modules-2/files-2.1/com.squareup.okhttp3/okhttp/4.10.0/cd63657ac15770ed1420647154c9f44645533bef/okhttp-4.10.0.jar:/root/.gradle/caches/modules-2/files-2.1/com.squareup.okio/okio-jvm/3.0.0/ab5a73fa2ccb4a36b0b5c69fe10b16d0255bcf8/okio-jvm-3.0.0.jar:/root/.gradle/caches/modules-2/files-2.1/org.jetbrains.kotlin/kotlin-stdlib-jdk8/1.7.22/b75c999616784d452613d670fec56a527f2bd3d1/kotlin-stdlib-jdk8-1.7.22.jar:/root/.gradle/caches/modules-2/files-2.1/org.jetbrains.kotlin/kotlin-stdlib-jdk7/1.7.22/6aa0a0e051d6b828dc25047025ec245758fab6d9/kotlin-stdlib-jdk7-1.7.22.jar:/root/.gradle/caches/modules-2/files-2.1/org.jetbrains.kotlin/kotlin-stdlib/1.7.22/9fb43bccb6cf8cfaaf3776c75f5df4f052f09b72/kotlin-stdlib-1.7.22.jar:/root/.gradle/caches/modules-2/files-2.1/org.jetbrains.kotlin/kotlin-stdlib-common/1.7.22/5dd2c9dcf64c94b46cf4bdb6fa4035b9bb726e2e/kotlin-stdlib-common-1.7.22.jar:/root/.gradle/caches/modules-2/files-2.1/io.micrometer/micrometer-commons/1.10.11/b8c88a741054458b78f6e30b7300fe485070bf07/micrometer-commons-1.10.11.jar:/root/.gradle/caches/modules-2/files-2.1/io.netty/netty-codec-http2/4.1.97.Final/893888d09a7bef0d0ba973d7471943e765d0fd08/netty-codec-http2-4.1.97.Final.jar:/root/.gradle/caches/modules-2/files-2.1/io.netty/netty-handler-proxy/4.1.97.Final/a99ecef0e1d86a92e40a7c89805c236d9cd7493e/netty-handler-proxy-4.1.97.Final.jar:/root/.gradle/caches/modules-2/files-2.1/io.netty/netty-codec-http/4.1.97.Final/af78acec783ffd77c63d8aecc21041fd39ac54f/netty-codec-http-4.1.97.Final.jar:/root/.gradle/caches/modules-2/files-2.1/io.netty/netty-resolver-dns-native-macos/4.1.97.Final/35a44e98cda1fec620dc336c6f8b3f44fa9b654d/netty-resolver-dns-native-macos-4.1.97.Final-osx-x86\_64.jar:/root/.gradle/caches/modules-2/files-2.1/io.netty/netty-resolver-dns-classes-macos/4.1.97.Final/2d42d18c6aca0774940a09a939f7f735b3d56e0a/netty-resolver-dns-classes-macos-4.1.97.Final.jar:/root/.gradle/caches/modules-2/files-2.1/io.netty/netty-resolver-dns/4.1.97.Final/2c50f835777ecd4535e15b552b5d9ccb26a2504f/netty-resolver-dns-4.1.97.Final.jar:/root/.gradle/caches/modules-2/files-2.1/io.netty/netty-handler/4.1.97.Final/abb86c6906bf512bf2b797a41cd7d2e8d3cd7c36/netty-handler-4.1.97.Final.jar:/root/.gradle/caches/modules-2/files-2.1/io.netty/netty-transport-native-epoll/4.1.97.Final/5514744c588190ffda076b35a9b8c9f24946a960/netty-transport-native-epoll-4.1.97.Final-linux-aarch\_64.jar:/root/.gradle/caches/modules-2/files

-2.1/io.netty/netty-transport-native  
 -epoll/4.1.97.Final/54188f271e388e7f313aea995e82f58ce2cdb809/netty-transport-native  
 -epoll-4.1.97.Final-linux-x86\_64.jar:/root/.gradle/caches/modules-2/files  
 -2.1/io.netty/netty-transport-native  
 -kqueue/4.1.97.Final/9952f04e1180b92b7d61febbc5e81914c76e37fb/netty-transport-native  
 -kqueue-4.1.97.Final-osx-x86\_64.jar:/root/.gradle/caches/modules-2/files  
 -2.1/io.netty/netty-codec  
 -socks/4.1.97.Final/30e8fa29a349db5a933225d61891b8802836bb79/netty-codec-socks  
 -4.1.97.Final.jar:/root/.gradle/caches/modules-2/files-2.1/io.netty/netty-codec  
 -dns/4.1.97.Final/d266d079ef33cf93a16b382d64dd15d562df1159/netty-codec-dns  
 -4.1.97.Final.jar:/root/.gradle/caches/modules-2/files-2.1/io.netty/netty  
 -codec/4.1.97.Final/384ba4d75670befbedb45c4d3b497a93639c206d/netty-codec  
 -4.1.97.Final.jar:/root/.gradle/caches/modules-2/files-2.1/io.netty/netty-transport  
 -classes-epoll/4.1.97.Final/795da37ded759e862457a82d9d92c4d39ce8ecee/netty-transport  
 -classes-epoll-4.1.97.Final.jar:/root/.gradle/caches/modules-2/files  
 -2.1/io.netty/netty-transport-classes  
 -kqueue/4.1.97.Final/29be6504ec6d9f5a173dfe562196998b2b365502/netty-transport-classes  
 -kqueue-4.1.97.Final.jar:/root/.gradle/caches/modules-2/files-2.1/io.netty/netty  
 -transport-native-unix  
 -common/4.1.97.Final/d469d84265ab70095b01b40886cabdd433b6e664/netty-transport-native  
 -unix-common-4.1.97.Final.jar:/root/.gradle/caches/modules-2/files-2.1/io.netty/netty  
 -transport/4.1.97.Final/f37380d23c9bb079bc702910833b2fd532c9abd0/netty-transport  
 -4.1.97.Final.jar:/root/.gradle/caches/modules-2/files-2.1/io.netty/netty  
 -buffer/4.1.97.Final/f8f3d8644afa5e6e1a40a3a6eb9d9aa970ecb4f/netty-buffer  
 -4.1.97.Final.jar:/root/.gradle/caches/modules-2/files-2.1/io.netty/netty  
 -resolver/4.1.97.Final/cec8348108dc76c47cf87c669d514be52c922144/netty-resolver  
 -4.1.97.Final.jar:/root/.gradle/caches/modules-2/files-2.1/io.netty/netty  
 -common/4.1.97.Final/7cceacaf11df8dc63f23d0fb58e9d4640fc88404/netty-common  
 -4.1.97.Final.jar:/root/.gradle/caches/modules-2/files-2.1/io.netty/netty-tcnative  
 -classes/2.0.61.Final/4c6ae851ed97921bc6c6b64e019c2b039f49131a/netty-tcnative-classes  
 -2.0.61.Final.jar:/root/.gradle/caches/modules-2/files  
 -2.1/io.opentelemetry.instrumentation/opentelemetry-instrumentation-api-semconv/1.19.2  
 -alpha/6c7aa07bb2f6ba396bfca39bee97d295f0921013/opentelemetry-instrumentation-api  
 -semconv-1.19.2-alpha.jar:/root/.gradle/caches/modules-2/files  
 -2.1/io.opentelemetry/opentelemetry  
 -sdk/1.19.0/a21c5b15550287f9d4782b18d82d944504f47e/opentelemetry-sdk  
 -1.19.0.jar:/root/.gradle/caches/modules-2/files-2.1/io.opentelemetry/opentelemetry  
 -sdk-trace/1.19.0/3683370ef425efe9cf794f3d57016f687e7ca5df/opentelemetry-sdk-trace  
 -1.19.0.jar:/root/.gradle/caches/modules-2/files-2.1/io.opentelemetry/opentelemetry  
 -sdk-logs/1.19.0-alpha/95e901fa9a6beb67ff83e353843556f490fec719/opentelemetry-sdk-logs  
 -1.19.0-alpha.jar:/root/.gradle/caches/modules-2/files  
 -2.1/io.opentelemetry/opentelemetry-sdk  
 -metrics/1.19.0/7a588f89f003fdf19e2a50bfd977c2a9075d4c98/opentelemetry-sdk-metrics  
 -1.19.0.jar:/root/.gradle/caches/modules-2/files-2.1/io.opentelemetry/opentelemetry  
 -sdk-common/1.19.0/72b3793dd29065e43a7b426f9b34ed6d23483f52/opentelemetry-sdk-common  
 -1.19.0.jar:/root/.gradle/caches/modules-2/files-2.1/io.opentelemetry/opentelemetry  
 -semconv/1.19.0-alpha/82f96bb99718c204a7775de298aabde2f6bffd2e/opentelemetry-semconv  
 -1.19.0-alpha.jar:/root/.gradle/caches/modules-2/files  
 -2.1/io.opentelemetry/opentelemetry-extension  
 -aws/1.19.0/671dd51db8fe155366a288f7f04190d0dfbbfac7/opentelemetry-extension-aws  
 -1.19.0.jar:/root/.gradle/caches/modules-2/files-2.1/io.opentelemetry/opentelemetry

```

-extension-trace
-propagators/1.19.0/ef9cd2988219f5d1874510d9f766810df4cee552/opentelemetry-extension
-trace-propagators-1.19.0.jar:/root/.gradle/caches/modules-2/files
-2.1/io.opentelemetry/opentelemetry-exporter
-common/1.19.0/868d4bbf56a7b7acc2a2f9bbd7de57872103d9d0/opentelemetry-exporter-common
-1.19.0.jar:/root/.gradle/caches/modules-2/files
-2.1/io.opentelemetry.instrumentation/opentelemetry-instrumentation
-api/1.19.2/817a9fb7e19b4f20716907612c5b0df121cf6868/opentelemetry-instrumentation-api
-1.19.2.jar:/root/.gradle/caches/modules-2/files-2.1/io.opentelemetry/opentelemetry
-api-logs/1.19.0-alpha/ed7f8c511cca3fe6e50d77f9cf740dd02483faef/opentelemetry-api-logs
-1.19.0-alpha.jar:/root/.gradle/caches/modules-2/files
-2.1/io.opentelemetry/opentelemetry
-api/1.19.0/95233814c9eb16486525e8e895a6ed326f3cb5f4/opentelemetry-api
-1.19.0.jar:/root/.gradle/caches/modules-2/files-2.1/io.opentelemetry/opentelemetry
-context/1.19.0/ae42d505048d264b08a3ad1bfa2aff532f244f54/opentelemetry-context
-1.19.0.jar:/root/.gradle/caches/modules-2/files
-2.1/io.prometheus/simpleclient_common/0.16.0/a09a8c790a20309b942a9fdbfe77da22407096e6
/simpleclient_common-0.16.0.jar:/root/.gradle/caches/modules-2/files-
2.1/io.prometheus/simpleclient/0.16.0/28b0eaf7c500c506976da8d0fc9cad6c278e8d87/simplec
lient-0.16.0.jar:/root/.gradle/caches/modules-2/files-
2.1/io.prometheus/simpleclient_tracer_otel/0.16.0/cc3d2b7b7cb6f077e3b1ee1d3e99eb54fddf
a151/simpleclient_tracer_otel-0.16.0.jar:/root/.gradle/caches/modules-2/files-
2.1/io.prometheus/simpleclient_tracer_otel_agent/0.16.0/9d724771e339ff7ec6cd7c0cc170d3
470904c5/simpleclient_tracer_otel_agent-0.16.0.jar:/root/.gradle/caches/modules-
2/files-
2.1/io.prometheus/simpleclient_tracer_common/0.16.0/dec00ef7c6155c4ca1109ec8248f7ff58d
8f6cd3/simpleclient_tracer_common-0.16.0.jar:/root/.gradle/caches/modules-2/files-
2.1/org.springframework/spring-
jcl/6.0.12/44a1d1d105063d4eb9c5dc705245ef6fa4162520/spring-jcl-
6.0.12.jar:/root/.gradle/caches/modules-2/files-
2.1/org.springframework.security/spring-security-
crypto/6.0.7/48d3f88377f5406daebcdd87545e83142eb6c3e5/spring-security-crypto-
6.0.7.jar:/root/.gradle/caches/modules-2/files-
2.1/com.typesafe/config/1.4.1/19058a07624a87f90d129af7cd9c68bee94535a9/config-
1.4.1.jar:/root/.gradle/caches/modules-2/files-2.1/com.github.jnr/jnr-
posix/3.1.15/f7d6737adcbd5925d625b8f99166de2cbf13caac/jnr-posix-
3.1.15.jar:/root/.gradle/caches/modules-2/files-
2.1/org.hdrhistogram/HdrHistogram/2.1.12/6eb7552156e0d517ae80cc2247be1427c8d90452/HdrH
istogram-2.1.12.jar:/root/.gradle/caches/modules-2/files-2.1/com.nimbusds/nimbus-jose-
jwt/9.24.4/29a1f6a00a4daa3e1873f6bf4f16ddf4d6fd6d37/nimbus-jose-jwt-
9.24.4.jar:/root/.gradle/caches/modules-2/files-2.1/com.github.stephenc.jcip/jcip-
annotations/1.0-1/ef31541dd28ae2cefdd17c7ebf352d93e9058c63/jcip-annotations-1.0-
1.jar:/root/.gradle/caches/modules-2/files-2.1/com.electronwill.night-
config/toml/3.6.6/f300021dfeece6a2907a0e7545574598220e4ef0/toml-
3.6.6.jar:/root/.gradle/caches/modules-2/files-2.1/org.hibernate.common/hibernate-
commons-annotations/6.0.6.Final/77a5f94b56d49508e0ee334751db5b78e5ccd50c/hibernate-
commons-annotations-6.0.6.Final.jar:/root/.gradle/caches/modules-2/files-
2.1/org.jboss.jandex/2.4.2.Final/1e1c385990b258ff1a24c801e84aebbacf70eb39/jandex-
2.4.2.Final.jar:/root/.gradle/caches/modules-2/files-2.1/org.antlr/antlr4-
runtime/4.10.1/10839f875928f59c622d675091d51a43ea0dc5f7/antlr4-runtime-
4.10.1.jar:/root/.gradle/caches/modules-2/files-2.1/org.msgpack/msgpack-

```

core/0.9.3/219f6fe7e9d4413a56bb1e819d4545282c85d700/msgpack-core-  
 0.9.3.jar:/root/.gradle/caches/modules-2/files-  
 2.1/org.json/json/20220320/6df2c050972619466f6dcef7654ef9bcc01dfd0/json-  
 20220320.jar:/root/.gradle/caches/modules-2/files-2.1/com.github.luben/zstd-jni/1.5.2-  
 1/fad786abc1d1b81570e8d9a2fc8a1ef479bc27b6/zstd-jni-1.5.2-  
 1.jar:/root/.gradle/caches/modules-2/files-2.1/org.lz4/lz4-  
 java/1.8.0/4b986a99445e49ea5fbf5d149c4b63f6ed6c6780/lz4-java-  
 1.8.0.jar:/root/.gradle/caches/modules-2/files-2.1/org.xerial.snappy/snappy-  
 java/1.1.8.4/66f0d56454509f6e36175f2331572e250e04a6cc/snappy-java-  
 1.1.8.4.jar:/root/.gradle/caches/modules-2/files-  
 2.1/org.rocksdb/rocksdbjni/7.1.2/ffe87d1c5d5b3a46d065cc4cf5311f18d8233a6b/rocksdbjni-  
 7.1.2.jar:/root/.gradle/caches/modules-2/files-2.1/com.mchange/mchange-commons-  
 java/0.2.19/7a4bee38ea02bd7dee776869b19fb3f6861d6acf/mchange-commons-java-  
 0.2.19.jar:/root/.gradle/caches/modules-2/files-2.1/io.micrometer/context-  
 propagation/1.0.5/c74e46568593def2bd6e63d67fddcced513c4e1d/context-propagation-  
 1.0.5.jar:/root/.gradle/caches/modules-2/files-2.1/org.apache.tomcat/tomcat-  
 juli/10.1.13/1c9b98f7b978afc7ce0ec51f2f6389b36f848d07/tomcat-juli-  
 10.1.13.jar:/root/.gradle/caches/modules-2/files-2.1/org.codehaus.plexus/plexus-  
 interpolation/1.26/25b919c664b79795ccde0ede5cee0fd68b544197/plexus-interpolation-  
 1.26.jar:/root/.gradle/caches/modules-2/files-2.1/org.apache.maven/maven-builder-  
 support/3.9.4/671110e56110d5c89eda904f198719a2b5fa4a21/maven-builder-support-  
 3.9.4.jar:/root/.gradle/caches/modules-2/files-  
 2.1/org.eclipse.sisu/org.eclipse.sisu.inject/0.3.5/d4265dd4f0f1d7a06d80df5a5f475d5ff9c  
 17140/org.eclipse.sisu.inject-0.3.5.jar:/root/.gradle/caches/modules-2/files-  
 2.1/org.codehaus.woodstox/stax2-  
 api/4.2.1/a3f7325c52240418c2ba257b103c3c550e140c83/stax2-api-  
 4.2.1.jar:/root/.gradle/caches/modules-2/files-  
 2.1/org.latencyutils/LatencyUtils/2.0.3/769c0b82cb2421c8256300e907298a9410a2a3d3/Laten  
 cyUtils-2.0.3.jar:/root/.gradle/caches/modules-2/files-  
 2.1/com.dynatrace.metric.util/dynatrace-metric-utils-  
 java/1.6.0/a988736aab51d86ab60ca834edf549597cc7ffb5/dynatrace-metric-utils-java-  
 1.6.0.jar:/root/.gradle/caches/modules-2/files-  
 2.1/info.ganglia/gmetric4j/gmetric4j/1.0.10/3d62003123b586adb86cb028cc0f8a8c3a701d81/g  
 metric4j-1.0.10.jar:/root/.gradle/caches/modules-2/files-  
 2.1/io.opentelemetry.proto/opentelemetry-proto/0.19.0-  
 alpha/940f1716e4ff81ee6cd797ec60c6f35ac3e5ccee/opentelemetry-proto-0.19.0-  
 alpha.jar:/root/.gradle/caches/modules-2/files-2.1/com.signalfx.public/signalfx-  
 java/1.0.34/7300192dec8f6ddcf23830f795cac2b9b69f32/signalfx-java-  
 1.0.34.jar:/root/.gradle/caches/modules-2/files-  
 2.1/aopalliance/aopalliance/1.0/235ba8b489512805ac13a8f9ea77a1ca5ebe3e8/aopalliance-  
 1.0.jar:/root/.gradle/caches/modules-2/files-  
 2.1/org.objenesis/objenesis/3.2/7fadf57620c8b8abdf7519533e5527367cb51f09/objenesis-  
 3.2.jar:/root/.gradle/caches/modules-2/files-  
 2.1/com.thoughtworks.qdox/qdox/2.0.3/d70143d2a58e7b16a8ec73a495508d43a085d83b/qdox-  
 2.0.3.jar:/root/.gradle/caches/modules-2/files-  
 2.1/org.jgroups/jgroups/5.2.0.Final/853b99ca3fce2e46a2d8872c1fa0418b99e6d68/jgroups-  
 5.2.0.Final.jar:/root/.gradle/caches/modules-2/files-2.1/commons-beanutils/commons-  
 beanutils/1.9.4/d52b9abcd97f38c81342bb7e7ae1eee9b73cba51/commons-beanutils-  
 1.9.4.jar:/root/.gradle/caches/modules-2/files-2.1/com.github.jnr/jnr-  
 ffi/2.2.11/bcf004ce358c87fc4cd2853b658d336348d0370f/jnr-ffi-  
 2.2.11.jar:/root/.gradle/caches/modules-2/files-2.1/com.github.jnr/jnr-

constants/0.10.3/af19ec7359dd72cbc7869c00bf7cdfa4c308e985/jnr-constants-  
0.10.3.jar:/root/.gradle/caches/modules-2/files-2.1/com.electronwill.night-  
config/core/3.6.6/1dccdc2c372ad0846561a2bd40c9db0543e55863/core-  
3.6.6.jar:/root/.gradle/caches/modules-2/files-  
2.1/com.squareup.moshi/moshi/1.8.0/752e7b187599d3ccb174d00ba7235e29add736be/moshi-  
1.8.0.jar:/root/.gradle/caches/modules-2/files-2.1/org.apache.commons/commons-  
collections4/4.4/62ebe7544cb7164d87e0637a2a6a2bdc981395e8/commons-collections4-  
4.4.jar:/root/.gradle/caches/modules-2/files-2.1/io.grpc/grpc-  
api/1.50.1/e16a030d5724a9070fdd6bf702bc2ed5230909b8/grpc-api-  
1.50.1.jar:/root/.gradle/caches/modules-2/files-2.1/io.opencensus/opencensus-  
api/0.31.1/66a60c7201c2b8b20ce495f0295b32bb0ccbbc57/opencensus-api-  
0.31.1.jar:/root/.gradle/caches/modules-2/files-2.1/io.grpc/grpc-  
context/1.50.1/9ba5b1fa99a9f9ae39319b496c6a2091b82d0c21/grpc-context-  
1.50.1.jar:/root/.gradle/caches/modules-2/files-2.1/io.grpc/grpc-  
stub/1.50.1/c578717d8445923f92c8eb06ec41c8def192be5c/grpc-stub-  
1.50.1.jar:/root/.gradle/caches/modules-2/files-2.1/io.grpc/grpc-  
protobuf/1.50.1/9fe96e868c741c194ae74b44c8196e3f43935e2c/grpc-protobuf-  
1.50.1.jar:/root/.gradle/caches/modules-2/files-2.1/io.grpc/grpc-protobuf-  
lite/1.50.1/2904b2d58711ec68850500dcaa4710bea42f8f30/grpc-protobuf-lite-  
1.50.1.jar:/root/.gradle/caches/modules-2/files-2.1/com.google.api/api-  
common/2.2.1/9eb62c522f96befccfbd8c92bafc952eed4417a8/api-common-  
2.2.1.jar:/root/.gradle/caches/modules-2/files-2.1/com.google.auto.value/auto-value-  
annotations/1.10/6327dc019fb316a8df3a202b68f38c909414eae8/auto-value-annotations-  
1.10.jar:/root/.gradle/caches/modules-2/files-2.1/com.google.protobuf/protobuf-  
java/3.21.8/2a1eebb74b844d9ccdf1d22eb2f57cec709698a9/protobuf-java-  
3.21.8.jar:/root/.gradle/caches/modules-2/files-2.1/com.google.api.grpc/proto-google-  
common-protos/2.9.6/ddb06641a75e3084fc91d680d1e7af90656434a4/proto-google-common-  
protos-2.9.6.jar:/root/.gradle/caches/modules-2/files-  
2.1/com.google.api/gax/2.19.4/af9e06d468490f0f9597cf8adb5dcd11b3da1bd5/gax-  
2.19.4.jar:/root/.gradle/caches/modules-2/files-2.1/com.google.auth/google-auth-  
library-credentials/1.12.2/f6dc543f34fc78dd0634340f37da782baabb8852/google-auth-  
library-credentials-1.12.2.jar:/root/.gradle/caches/modules-2/files-  
2.1/com.google.api/gax-grpc/2.19.4/3f99cd8b17ff56b45b352ab79e74159fedd0c682/gax-grpc-  
2.19.4.jar:/root/.gradle/caches/modules-2/files-2.1/io.grpc/grpc-  
alts/1.50.1/7e9ca160f061198d20811b05167b0f6f2c19199e/grpc-alts-  
1.50.1.jar:/root/.gradle/caches/modules-2/files-2.1/io.grpc/grpc-  
grpc-lb/1.50.1/fb2c33661e4666b924bdfb50dd8577db5023a0b6/grpc-grpclb-  
1.50.1.jar:/root/.gradle/caches/modules-2/files-2.1/com.google.protobuf/protobuf-java-  
util/3.21.8/9f9b51347bfd695278ba4eeb54003ae9cc6d6ab/protobuf-java-util-  
3.21.8.jar:/root/.gradle/caches/modules-2/files-2.1/org.conscrypt/conscrypt-openjdk-  
uber/2.5.2/d858f142ea189c62771c505a6548d8606ac098fe/conscrypt-openjdk-uber-  
2.5.2.jar:/root/.gradle/caches/modules-2/files-2.1/io.grpc/grpc-  
auth/1.50.1/176ee028d8d1fc23bb6194fb3cc40b4c8a4e59db/grpc-auth-  
1.50.1.jar:/root/.gradle/caches/modules-2/files-2.1/io.grpc/grpc-netty-  
shaded/1.50.1/8b56e8405141c7dbc389eaa9993a3a11784ed175/grpc-netty-shaded-  
1.50.1.jar:/root/.gradle/caches/modules-2/files-2.1/io.perfmark/perfmark-  
api/0.25.0/340a0c3d81cdcd9ecd7dc2ae00fb2633b469b157/perfmark-api-  
0.25.0.jar:/root/.gradle/caches/modules-2/files-2.1/io.grpc/grpc-  
core/1.50.1/6a41a894570c97c875d82cda13250dfee3fd3a93/grpc-core-  
1.50.1.jar:/root/.gradle/caches/modules-2/files-  
2.1/com.google.android/annotations/4.1.1.4/a1678ba907bf92691d879fef34e1a187038f9259/an

notations-4.1.1.4.jar:/root/.gradle/caches/modules-2/files-  
2.1/org.codehaus.mojo/animal-sniffer-  
annotations/1.22/911f763493163e03d750b369cc162085db09b46b/animal-sniffer-annotations-  
1.22.jar:/root/.gradle/caches/modules-2/files-2.1/io.grpc/grpc-  
googleapis/1.50.1/a1842155811366efc49961a573ac0161f7e3d3a9/grpc-googleapis-  
1.50.1.jar:/root/.gradle/caches/modules-2/files-2.1/io.grpc/grpc-  
xds/1.50.1/7b551b7069f69346270093f7838a1ad0ebae6714/grpc-xds-  
1.50.1.jar:/root/.gradle/caches/modules-2/files-2.1/io.opencensus/opencensus-  
proto/0.2.0/c05b6b32b69d5d9144087ea0ebc6fab183fb9151/opencensus-proto-  
0.2.0.jar:/root/.gradle/caches/modules-2/files-2.1/io.grpc/grpc-  
services/1.50.1/cf4d36f89ebb43685f5f9b0e405900c9c149a2a8/grpc-services-  
1.50.1.jar:/root/.gradle/caches/modules-2/files-  
2.1/com.google.re2j/re2j/1.6/a13e879fd7971738d06020fefeb108cc14e14169/re2j-  
1.6.jar:/root/.gradle/caches/modules-2/files-  
2.1/org.threeten/threetenbp/1.6.3/121bf72e1c110a982ee7724ae0b6f83b18382ac3/threetenbp-  
1.6.3.jar:/root/.gradle/caches/modules-2/files-2.1/org.jboss.logmanager/jboss-  
logmanager/2.1.18.Final/b43e1943e270ba4907b061f7f7f85d516ff4c92b/jboss-logmanager-  
2.1.18.Final.jar:/root/.gradle/caches/modules-2/files-2.1/org.wildfly.common/wildfly-  
common/1.5.4.Final/735ceee5616d5143bac1bc740e444697073c002f/wildfly-common-  
1.5.4.Final.jar:/root/.gradle/caches/modules-2/files-2.1/org.jctools/jctools-  
core/2.1.2/8ec46a6a26e7c1c7e57e2590a043238ffc462144/jctools-core-  
2.1.2.jar:/root/.gradle/caches/modules-2/files-2.1/net.minidev/accessors-  
smart/2.4.11/245ceca7bdf3190fbb977045c852d5f3c8efece1/accessors-smart-  
2.4.11.jar:/root/.gradle/caches/modules-2/files-2.1/org.glassfish.hk2/osgi-resource-  
locator/1.0.3/de3b21279df7e755e38275137539be5e2c80dd58/osgi-resource-locator-  
1.0.3.jar:/root/.gradle/caches/modules-2/files-2.1/org.glassfish.hk2/class-  
model/3.0.4/57f07924bcb9196afc7f5050a742b2a0f8b6dd2f/class-model-  
3.0.4.jar:/root/.gradle/caches/modules-2/files-2.1/org.javassist/javassist/3.29.2-  
GA/6c32028609e5dd4a1b78e10fbcd122b09b3928b1/javassist-3.29.2-  
GA.jar:/root/.gradle/caches/modules-2/files-  
2.1/org.opentest4j/opentest4j/1.2.0/28c11eb91f9b6d8e200631d46e20a7f407f2a046/opentest4j-  
1.2.0.jar:/root/.gradle/caches/modules-2/files-2.1/org.acplt.remotetea/remotetea-  
oncrpc/1.1.2/705c490ad22ff4627389853439f9decf5ee69be/remotetea-oncrpc-  
1.1.2.jar:/root/.gradle/caches/modules-2/files-2.1/com.tdunning/t-  
digest/3.2/2ab94758b0276a8a26102adf8d528cf6d0567b9a/t-digest-  
3.2.jar:/root/.gradle/caches/modules-2/files-2.1/org.atteo/evo-  
inflector/1.3/4cf8b5f363c60e63f8b7688ac053590460f2768e/evo-inflector-  
1.3.jar:/root/.gradle/caches/modules-2/files-2.1/commons-collections/commons-  
collections/3.2.2/8ad72fe39fa8c91eaaf12aadb21e0c3661fe26d5/commons-collections-  
3.2.2.jar:/root/.gradle/caches/modules-2/files-  
2.1/com.github.jnr/jffi/1.3.9/b776ea131fa693af2c943368b52acf94131cbd7/jffi-  
1.3.9.jar:/root/.gradle/caches/modules-2/files-  
2.1/com.github.jnr/jffi/1.3.9/163d683f80c06911f3e770c723d6e399e4c59448/jffi-1.3.9-  
native.jar:/root/.gradle/caches/modules-2/files-2.1/org.ow2.asm/asm-  
commons/9.5/19ab5b5800a3910d30d3a3e64fdb00fd0cb42de0/asm-commons-  
9.5.jar:/root/.gradle/caches/modules-2/files-2.1/org.ow2.asm/asm-  
util/9.5/64b5a1fc8c1b15ed2efd6a063e976bc8d3dc5ffe/asm-util-  
9.5.jar:/root/.gradle/caches/modules-2/files-2.1/org.ow2.asm/asm-  
analysis/9.5/490bacc77de7cbc0be1a30bb3471072d705be4a4/asm-analysis-  
9.5.jar:/root/.gradle/caches/modules-2/files-2.1/org.ow2.asm/asm-  
tree/9.5/fd33c8b6373abaa675be407082fdafa35021254a/asm-tree-

```

9.5.jar:/root/.gradle/caches/modules-2/files-
2.1/org.ow2.asm/asm/9.5/dc6ea1875f4d64fbc85e1691c95b96a3d8569c90/asm-
9.5.jar:/root/.gradle/caches/modules-2/files-2.1/com.github.jnr/jnr-
a64asm/1.0.0/a1cb8dbe71b5a6a0288043c3ba3ca64545be165/jnr-a64asm-
1.0.0.jar:/root/.gradle/caches/modules-2/files-2.1/com.github.jnr/jnr-
x86asm/1.0.2/6936bbd6c5b235665d87bd450f5e13b52d4b48/jnr-x86asm-
1.0.2.jar:/root/.gradle/caches/modules-2/files-
2.1/org.jetbrains/annotations/13.0/919f0dfe192fb4e063e7dacadee7f8bb9a2672a9/annotation
s-13.0.jar:/root/.gradle/caches/modules-2/files-
2.1/org.glassfish.hk2.external/aopalliance-
repackaged/3.0.4/b1d38a6e64c50f501503d7c4297445a346583870/aopalliance-repackaged-
3.0.4.jar"
    },
    "java.vm.vendor" : {
        "value" : "BellSoft"
    }
}
}, {
    "name" : "systemEnvironment",
    "properties" : {
        "JAVA_HOME" : {
            "value" : "/opt/openjdk",
            "origin" : "System Environment Property \"JAVA_HOME\""
        }
    }
}, {
    "name" : "Config resource 'class path resource [application.properties]' via
location 'classpath:/'",
    "properties" : {
        "com.example.cache.max-size" : {
            "value" : "1000",
            "origin" : "class path resource [application.properties] - 1:29"
        }
    }
}
} ]
}

```

### 7.1.1. Response Structure

The response contains details of the application's **Environment**. The following table describes the structure of the response:

| Path                                      | Type   | Description                                               |
|-------------------------------------------|--------|-----------------------------------------------------------|
| <code>activeProfiles</code>               | Array  | Names of the active profiles, if any.                     |
| <code>propertySources</code>              | Array  | Property sources in order of precedence.                  |
| <code>propertySources[].name</code>       | String | Name of the property source.                              |
| <code>propertySources[].properties</code> | Object | Properties in the property source keyed by property name. |

| Path                                               | Type   | Description                     |
|----------------------------------------------------|--------|---------------------------------|
| <code>propertySources[].properties.*.value</code>  | String | Value of the property.          |
| <code>propertySources[].properties.*.origin</code> | String | Origin of the property, if any. |

## 7.2. Retrieving a Single Property

To retrieve a single property, make a **GET** request to `/actuator/env/{property.name}`, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/env/com.example.cache.max-size' -i -X GET
```

The preceding example retrieves information about the property named `com.example.cache.max-size`. The resulting response is similar to the following:

```
HTTP/1.1 200 OK
Content-Disposition: inline;filename=f.txt
Content-Type: application/vnd.spring-boot.actuator.v3+json
Content-Length: 455

{"property":{"source":"Config resource 'class path resource [application.properties]' via location 'classpath:/'", "value":"1000"}, "activeProfiles":[], "propertySources":[{"name":"servlet ContextInitParams"}, {"name":"systemProperties"}, {"name":"systemEnvironment"}, {"name":"Config resource 'class path resource [application.properties]' via location 'classpath:/'", "property":{"value":"1000", "origin":"class path resource [application.properties] - 1:29"}}]}
```

### 7.2.1. Response Structure

The response contains details of the requested property. The following table describes the structure of the response:

| Path                                          | Type   | Description                              |
|-----------------------------------------------|--------|------------------------------------------|
| <code>property</code>                         | Object | Property from the environment, if found. |
| <code>property.source</code>                  | String | Name of the source of the property.      |
| <code>property.value</code>                   | String | Value of the property.                   |
| <code>activeProfiles</code>                   | Array  | Names of the active profiles, if any.    |
| <code>propertySources</code>                  | Array  | Property sources in order of precedence. |
| <code>propertySources[].name</code>           | String | Name of the property source.             |
| <code>propertySources[].property</code>       | Object | Property in the property source, if any. |
| <code>propertySources[].property.value</code> | Varies | Value of the property.                   |

| Path                                           | Type                | Description                     |
|------------------------------------------------|---------------------|---------------------------------|
| <code>propertySources[].property.origin</code> | <code>String</code> | Origin of the property, if any. |

# Chapter 8. Flyway (flyway)

The **flyway** endpoint provides information about database migrations performed by Flyway.

## 8.1. Retrieving the Migrations

To retrieve the migrations, make a **GET** request to **/actuator/flyway**, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/flyway' -i -X GET
```

The resulting response is similar to the following:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.spring-boot.actuator.v3+json
Content-Length: 515

{
  "contexts" : {
    "application" : {
      "flywayBeans" : {
        "flyway" : {
          "migrations" : [ {
            "type" : "SQL",
            "checksum" : -156244537,
            "version" : "1",
            "description" : "init",
            "script" : "V1__init.sql",
            "state" : "SUCCESS",
            "installedBy" : "SA",
            "installedOn" : "2023-09-21T00:38:52.598Z",
            "installedRank" : 1,
            "executionTime" : 2
          } ]
        }
      }
    }
  }
}
```

### 8.1.1. Response Structure

The response contains details of the application's Flyway migrations. The following table describes the structure of the response:

| Path                                                             | Type   | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|------------------------------------------------------------------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>contexts</code>                                            | Object | Application contexts keyed by id                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| <code>contexts.*.flywayBeans.*.migrations</code>                 | Array  | Migrations performed by the Flyway instance, keyed by Flyway bean name.                                                                                                                                                                                                                                                                                                                                                                                                                             |
| <code>contexts.*.flywayBeans.*.migrations[].checksum</code>      | Number | Checksum of the migration, if any.                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| <code>contexts.*.flywayBeans.*.migrations[].description</code>   | String | Description of the migration, if any.                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| <code>contexts.*.flywayBeans.*.migrations[].executionTime</code> | Number | Execution time in milliseconds of an applied migration.                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| <code>contexts.*.flywayBeans.*.migrations[].installedBy</code>   | String | User that installed the applied migration, if any.                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| <code>contexts.*.flywayBeans.*.migrations[].installedOn</code>   | String | Timestamp of when the applied migration was installed, if any.                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| <code>contexts.*.flywayBeans.*.migrations[].installedRank</code> | Number | Rank of the applied migration, if any. Later migrations have higher ranks.                                                                                                                                                                                                                                                                                                                                                                                                                          |
| <code>contexts.*.flywayBeans.*.migrations[].script</code>        | String | Name of the script used to execute the migration, if any.                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| <code>contexts.*.flywayBeans.*.migrations[].state</code>         | String | State of the migration. ( <code>PENDING</code> , <code>ABOVE_TARGET</code> , <code>BELOW_BASELINE</code> , <code>BASELINE_IGNORED</code> , <code>BASELINE_IGNORED</code> , <code>MISSING_SUCCESS</code> , <code>MISSING_FAILED</code> , <code>SUCCESS</code> , <code>UNDONE</code> , <code>AVAILABLE</code> , <code>FAILED</code> , <code>OUT_OF_ORDER</code> , <code>FUTURE_SUCCESS</code> , <code>FUTURE_FAILED</code> , <code>OUTDATED</code> , <code>SUPERSEDED</code> , <code>DELETED</code> ) |
| <code>contexts.*.flywayBeans.*.migrations[].type</code>          | String | Type of the migration.                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| <code>contexts.*.flywayBeans.*.migrations[].version</code>       | String | Version of the database after applying the migration, if any.                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| <code>contexts.*.parentId</code>                                 | String | Id of the parent application context, if any.                                                                                                                                                                                                                                                                                                                                                                                                                                                       |

# Chapter 9. Health (health)

The **health** endpoint provides detailed information about the health of the application.

## 9.1. Retrieving the Health of the Application

To retrieve the health of the application, make a **GET** request to **/actuator/health**, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/health' -i -X GET \  
-H 'Accept: application/json'
```

The resulting response is similar to the following:

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 810
```

```
{
  "status" : "UP",
  "components" : {
    "broker" : {
      "status" : "UP",
      "components" : {
        "us1" : {
          "status" : "UP",
          "details" : {
            "version" : "1.0.2"
          }
        },
        "us2" : {
          "status" : "UP",
          "details" : {
            "version" : "1.0.4"
          }
        }
      }
    },
    "db" : {
      "status" : "UP",
      "details" : {
        "database" : "H2",
        "validationQuery" : "isValid()"
      }
    },
    "diskSpace" : {
      "status" : "UP",
      "details" : {
        "total" : 325426569216,
        "free" : 192896565248,
        "threshold" : 10485760,
        "path" : "/tmp/build/6da1d659/git-repo/spring-boot-project/spring-boot-actuator-autoconfigure/.",
        "exists" : true
      }
    }
  }
}
```

### 9.1.1. Response Structure

The response contains details of the health of the application. The following table describes the structure of the response:

| Path                                 | Type                | Description                                                                                                                                   |
|--------------------------------------|---------------------|-----------------------------------------------------------------------------------------------------------------------------------------------|
| <code>status</code>                  | <code>String</code> | Overall status of the application.                                                                                                            |
| <code>components</code>              | <code>Object</code> | The components that make up the health.                                                                                                       |
| <code>components.*.status</code>     | <code>String</code> | Status of a specific part of the application.                                                                                                 |
| <code>components.*.components</code> | <code>Object</code> | The nested components that make up the health.                                                                                                |
| <code>components.*.details</code>    | <code>Object</code> | Details of the health of a specific part of the application. Presence is controlled by <code>management.endpoint.health.show-details</code> . |



The response fields above are for the V3 API. If you need to return V2 JSON you should use an accept header or `application/vnd.spring-boot.actuator.v2+json`

## 9.2. Retrieving the Health of a Component

To retrieve the health of a particular component of the application's health, make a `GET` request to `/actuator/health/{component}`, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/health/db' -i -X GET \
-H 'Accept: application/json'
```

The resulting response is similar to the following:

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 101

{
  "status" : "UP",
  "details" : {
    "database" : "H2",
    "validationQuery" : "isValid()"
  }
}
```

### 9.2.1. Response Structure

The response contains details of the health of a particular component of the application's health. The following table describes the structure of the response:

| Path                 | Type                | Description                                                  |
|----------------------|---------------------|--------------------------------------------------------------|
| <code>status</code>  | <code>String</code> | Status of a specific part of the application                 |
| <code>details</code> | <code>Object</code> | Details of the health of a specific part of the application. |

## 9.3. Retrieving the Health of a Nested Component

If a particular component contains other nested components (as the **broker** indicator in the example above), the health of such a nested component can be retrieved by issuing a **GET** request to `/actuator/health/{component}/{subcomponent}`, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/health/broker/us1' -i -X GET \
-H 'Accept: application/json'
```

The resulting response is similar to the following:

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 66

{
  "status" : "UP",
  "details" : {
    "version" : "1.0.2"
  }
}
```

Components of an application's health may be nested arbitrarily deep depending on the application's health indicators and how they have been grouped. The health endpoint supports any number of `/health/{component}` identifiers in the URL to allow the health of a component at any depth to be retrieved.

### 9.3.1. Response Structure

The response contains details of the health of an instance of a particular component of the application. The following table describes the structure of the response:

| Path                 | Type                | Description                                                  |
|----------------------|---------------------|--------------------------------------------------------------|
| <code>status</code>  | <code>String</code> | Status of a specific part of the application                 |
| <code>details</code> | <code>Object</code> | Details of the health of a specific part of the application. |

# Chapter 10. Heap Dump (heapdump)

The `heapdump` endpoint provides a heap dump from the application's JVM.

## 10.1. Retrieving the Heap Dump

To retrieve the heap dump, make a `GET` request to `/actuator/heapdump`. The response is binary data and can be large. Its format depends upon the JVM on which the application is running. When running on a HotSpot JVM the format is `HPROF` and on OpenJ9 it is `PHD`. Typically, you should save the response to disk for subsequent analysis. When using curl, this can be achieved by using the `-O` option, as shown in the following example:

```
$ curl 'http://localhost:8080/actuator/heapdump' -O
```

The preceding example results in a file named `heapdump` being written to the current working directory.

# Chapter 11. HTTP Exchanges (**httpexchanges**)

The **httpexchanges** endpoint provides information about HTTP request-response exchanges.

## 11.1. Retrieving the HTTP Exchanges

To retrieve the HTTP exchanges, make a **GET** request to **/actuator/httpexchanges**, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/httpexchanges' -i -X GET
```

The resulting response is similar to the following:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.spring-boot.actuator.v3+json
Content-Length: 511

{
  "exchanges" : [ {
    "timestamp" : "2022-12-22T13:43:41Z",
    "request" : {
      "uri" : "https://api.example.com",
      "method" : "GET",
      "headers" : {
        "Accept" : [ "application/json" ]
      }
    },
    "response" : {
      "status" : 200,
      "headers" : {
        "Content-Type" : [ "application/json" ]
      }
    },
    "principal" : {
      "name" : "alice"
    },
    "session" : {
      "id" : "fd7a61a9-9d85-4503-8420-528ac65de459"
    },
    "timeTaken" : "PT0.023S"
  } ]
}
```

### 11.1.1. Response Structure

The response contains details of the traced HTTP request-response exchanges. The following table describes the structure of the response:

| Path                                           | Type   | Description                                                   |
|------------------------------------------------|--------|---------------------------------------------------------------|
| <code>exchanges</code>                         | Array  | An array of HTTP request-response exchanges.                  |
| <code>exchanges[].timestamp</code>             | String | Timestamp of when the exchange occurred.                      |
| <code>exchanges[].principal</code>             | Object | Principal of the exchange, if any.                            |
| <code>exchanges[].principal.name</code>        | String | Name of the principal.                                        |
| <code>exchanges[].request.method</code>        | String | HTTP method of the request.                                   |
| <code>exchanges[].request.remoteAddress</code> | String | Remote address from which the request was received, if known. |
| <code>exchanges[].request.uri</code>           | String | URI of the request.                                           |
| <code>exchanges[].request.headers</code>       | Object | Headers of the request, keyed by header name.                 |
| <code>exchanges[].request.headers.*.[]</code>  | Array  | Values of the header                                          |
| <code>exchanges[].response.status</code>       | Number | Status of the response                                        |
| <code>exchanges[].response.headers</code>      | Object | Headers of the response, keyed by header name.                |
| <code>exchanges[].response.headers.*.[]</code> | Array  | Values of the header                                          |
| <code>exchanges[].session</code>               | Object | Session associated with the exchange, if any.                 |
| <code>exchanges[].session.id</code>            | String | ID of the session.                                            |
| <code>exchanges[].timeTaken</code>             | String | Time taken to handle the exchange.                            |

# Chapter 12. Info (info)

The **info** endpoint provides general information about the application.

## 12.1. Retrieving the Info

To retrieve the information about the application, make a **GET** request to **/actuator/info**, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/info' -i -X GET
```

The resulting response is similar to the following:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.spring-boot.actuator.v3+json
Content-Length: 231

{
  "git" : {
    "branch" : "main",
    "commit" : {
      "id" : "df027cf",
      "time" : "2023-09-21T00:38:53Z"
    }
  },
  "build" : {
    "artifact" : "application",
    "version" : "1.0.3",
    "group" : "com.example"
  }
}
```

### 12.1.1. Response Structure

The response contains general information about the application. Each section of the response is contributed by an **InfoContributor**. Spring Boot provides several contributors that are described below.

#### Build Response Structure

The following table describe the structure of the **build** section of the response:

| Path            | Type          | Description                             |
|-----------------|---------------|-----------------------------------------|
| <b>artifact</b> | <b>String</b> | Artifact ID of the application, if any. |
| <b>group</b>    | <b>String</b> | Group ID of the application, if any.    |

| Path    | Type   | Description                                          |
|---------|--------|------------------------------------------------------|
| name    | String | Name of the application, if any.                     |
| version | String | Version of the application, if any.                  |
| time    | Varies | Timestamp of when the application was built, if any. |

## Git Response Structure

The following table describes the structure of the `git` section of the response:

| Path        | Type   | Description                        |
|-------------|--------|------------------------------------|
| branch      | String | Name of the Git branch, if any.    |
| commit      | Object | Details of the Git commit, if any. |
| commit.time | Varies | Timestamp of the commit, if any.   |
| commit.id   | String | ID of the commit, if any.          |



This is the "simple" output. The contributor can also be configured to output all available data.

# Chapter 13. Spring Integration graph (*integrationgraph*)

The *integrationgraph* endpoint exposes a graph containing all Spring Integration components.

## 13.1. Retrieving the Spring Integration Graph

To retrieve the information about the application, make a *GET* request to */actuator/integrationgraph*, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/integrationgraph' -i -X GET
```

The resulting response is similar to the following:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.spring-boot.actuator.v3+json
Content-Length: 1033
```

```
{
  "contentDescriptor" : {
    "providerVersion" : "6.0.7",
    "providerFormatVersion" : 1.2,
    "provider" : "spring-integration"
  },
  "nodes" : [ {
    "nodeId" : 1,
    "componentType" : "null-channel",
    "integrationPatternType" : "null_channel",
    "integrationPatternCategory" : "messaging_channel",
    "properties" : { },
    "observed" : false,
    "name" : "nullChannel"
  }, {
    "nodeId" : 2,
    "componentType" : "publish-subscribe-channel",
    "integrationPatternType" : "publish_subscribe_channel",
    "integrationPatternCategory" : "messaging_channel",
    "properties" : { },
    "observed" : false,
    "name" : "errorChannel"
  }, {
    "nodeId" : 3,
    "componentType" : "logging-channel-adapter",
    "integrationPatternType" : "outbound_channel_adapter",
    "integrationPatternCategory" : "messaging_endpoint",
    "properties" : { },
    "observed" : false,
    "input" : "errorChannel",
    "name" : "errorLogger"
  } ],
  "links" : [ {
    "from" : 2,
    "to" : 3,
    "type" : "input"
  } ]
}
```

### 13.1.1. Response Structure

The response contains all Spring Integration components used within the application, as well as the links between them. More information about the structure can be found in the [reference documentation](#).

## 13.2. Rebuilding the Spring Integration Graph

To rebuild the exposed graph, make a **POST** request to `/actuator/integrationgraph`, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/integrationgraph' -i -X POST
```

This will result in a **204 - No Content** response:

```
HTTP/1.1 204 No Content
```

# Chapter 14. Liquibase (liquibase)

The **liquibase** endpoint provides information about database change sets applied by Liquibase.

## 14.1. Retrieving the Changes

To retrieve the changes, make a **GET** request to **/actuator/liquibase**, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/liquibase' -i -X GET
```

The resulting response is similar to the following:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.spring-boot.actuator.v3+json
Content-Length: 677

{
  "contexts" : {
    "application" : {
      "liquibaseBeans" : {
        "liquibase" : {
          "changeSets" : [ {
            "author" : "marceloverdijk",
            "changeLog" : "db/changelog/db.changelog-master.yaml",
            "comments" : "",
            "contexts" : [ ],
            "dateExecuted" : "2023-09-21T00:38:55.347Z",
            "deploymentId" : "5256735020",
            "description" : "createTable tableName=customer",
            "execType" : "EXECUTED",
            "id" : "1",
            "labels" : [ ],
            "checksum" : "8:46debf252cce6d7b25e28ddeb9fc4bf6",
            "orderExecuted" : 1
          } ]
        }
      }
    }
  }
}
```

### 14.1.1. Response Structure

The response contains details of the application's Liquibase change sets. The following table describes the structure of the response:

| Path                                                   | Type   | Description                                                                                                             |
|--------------------------------------------------------|--------|-------------------------------------------------------------------------------------------------------------------------|
| contexts                                               | Object | Application contexts keyed by id                                                                                        |
| contexts.*.liquibaseBeans.*.changeSets                 | Array  | Change sets made by the Liquibase beans, keyed by bean name.                                                            |
| contexts.*.liquibaseBeans.*.changeSets[].author        | String | Author of the change set.                                                                                               |
| contexts.*.liquibaseBeans.*.changeSets[].changelog     | String | Change log that contains the change set.                                                                                |
| contexts.*.liquibaseBeans.*.changeSets[].comments      | String | Comments on the change set.                                                                                             |
| contexts.*.liquibaseBeans.*.changeSets[].contexts      | Array  | Contexts of the change set.                                                                                             |
| contexts.*.liquibaseBeans.*.changeSets[].dateExecuted  | String | Timestamp of when the change set was executed.                                                                          |
| contexts.*.liquibaseBeans.*.changeSets[].deploymentId  | String | ID of the deployment that ran the change set.                                                                           |
| contexts.*.liquibaseBeans.*.changeSets[].description   | String | Description of the change set.                                                                                          |
| contexts.*.liquibaseBeans.*.changeSets[].execType      | String | Execution type of the change set ( <b>EXECUTED</b> , <b>FAILED</b> , <b>SKIPPED</b> , <b>RERAN</b> , <b>MARK_RAN</b> ). |
| contexts.*.liquibaseBeans.*.changeSets[].id            | String | ID of the change set.                                                                                                   |
| contexts.*.liquibaseBeans.*.changeSets[].labels        | Array  | Labels associated with the change set.                                                                                  |
| contexts.*.liquibaseBeans.*.changeSets[].checksum      | String | Checksum of the change set.                                                                                             |
| contexts.*.liquibaseBeans.*.changeSets[].orderExecuted | Number | Order of the execution of the change set.                                                                               |
| contexts.*.liquibaseBeans.*.changeSets[].tag           | String | Tag associated with the change set, if any.                                                                             |
| contexts.*.parentId                                    | String | Id of the parent application context, if any.                                                                           |

## Chapter 15. Log File (logfile)

The **logfile** endpoint provides access to the contents of the application's log file.

## 15.1. Retrieving the Log File

To retrieve the log file, make a **GET** request to `/actuator/logfile`, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/logfile' -i -X GET
```

The resulting response is similar to the following:

```

HTTP/1.1 200 OK
Accept-Ranges: bytes
Content-Type: text/plain;charset=UTF-8
Content-Length: 4715

      .      _ _ _ _ _
     /\ /  _ _ _ ' _ _ _ _ _ ( _ ) _ _ _ _ _ \ \ \ \ \
    ( ( ) \ _ _ _ | ' _ | ' _ | | ' _ \ / ' _ | \ \ \ \ \
     \ \ /  _ _ _ | | _ | | | | | | | ( _ | | ) ) ) )
      '  | _ _ _ | . _ _ | _ | | _ | _ | _ \ _ _ , | / / / /
     =====|_|=====|_| _ _ / = / _ _ / _ /
:: Spring Boot ::

```

```

2017-08-08 17:12:30.910 INFO 19866 --- [          main]
s.f.SampleWebFreeMarkerApplication      : Starting SampleWebFreeMarkerApplication
with PID 19866
2017-08-08 17:12:30.913 INFO 19866 --- [          main]
s.f.SampleWebFreeMarkerApplication      : No active profile set, falling back to
default profiles: default
2017-08-08 17:12:30.952 INFO 19866 --- [          main]
ConfigServletWebServerApplicationContext : Refreshing
org.springframework.boot.web.servlet.context.AnnotationConfigServletWebServerApplicati
onContext@76b10754: startup date [Tue Aug 08 17:12:30 BST 2017]; root of context
hierarchy
2017-08-08 17:12:31.878 INFO 19866 --- [          main]
o.s.b.w.embedded.tomcat.TomcatWebServer  : Tomcat initialized with port(s): 8080
(http)
2017-08-08 17:12:31.889 INFO 19866 --- [          main]
o.apache.catalina.core.StandardService   : Starting service [Tomcat]
2017-08-08 17:12:31.890 INFO 19866 --- [          main]
org.apache.catalina.core.StandardEngine   : Starting Servlet Engine: Apache
Tomcat/8.5.16
2017-08-08 17:12:31.978 INFO 19866 --- [ost-startStop-1]
o.a.c.c.C.[Tomcat].[localhost].[/]       : Initializing Spring embedded
WebApplicationContext

```

```

2017-08-08 17:12:31.978 INFO 19866 --- [ost-startStop-1]
o.s.web.context.ContextLoader      : Root WebApplicationContext: initialization
completed in 1028 ms
2017-08-08 17:12:32.080 INFO 19866 --- [ost-startStop-1]
o.s.b.w.servlet.ServletRegistrationBean : Mapping servlet: 'dispatcherServlet' to [/]
2017-08-08 17:12:32.084 INFO 19866 --- [ost-startStop-1]
o.s.b.w.servlet.FilterRegistrationBean : Mapping filter: 'characterEncodingFilter'
to: [/]
2017-08-08 17:12:32.084 INFO 19866 --- [ost-startStop-1]
o.s.b.w.servlet.FilterRegistrationBean : Mapping filter: 'hiddenHttpMethodFilter'
to: [/]
2017-08-08 17:12:32.084 INFO 19866 --- [ost-startStop-1]
o.s.b.w.servlet.FilterRegistrationBean : Mapping filter: 'httpPutFormContentFilter'
to: [/]
2017-08-08 17:12:32.084 INFO 19866 --- [ost-startStop-1]
o.s.b.w.servlet.FilterRegistrationBean : Mapping filter: 'requestContextFilter' to:
[/]
2017-08-08 17:12:32.349 INFO 19866 --- [          main]
s.w.s.m.m.a.RequestMappingHandlerAdapter : Looking for @ControllerAdvice:
org.springframework.boot.web.servlet.context.AnnotationConfigServletWebServerApplicati
onContext@76b10754: startup date [Tue Aug 08 17:12:30 BST 2017]; root of context
hierarchy
2017-08-08 17:12:32.420 INFO 19866 --- [          main]
s.w.s.m.m.a.RequestMappingHandlerMapping : Mapped "{[/error]}" onto public
org.springframework.http.ResponseEntity<java.util.Map<java.lang.String,
java.lang.Object>>
org.springframework.boot.autoconfigure.web.servlet.error.BasicErrorController.error(ja
karta.servlet.http.HttpServletRequest)
2017-08-08 17:12:32.421 INFO 19866 --- [          main]
s.w.s.m.m.a.RequestMappingHandlerMapping : Mapped "{[/error],produces=[text/html]}"
onto public org.springframework.web.servlet.ModelAndView
org.springframework.boot.autoconfigure.web.servlet.error.BasicErrorController.errorHtm
l(jakarta.servlet.http.HttpServletRequest,jakarta.servlet.http.HttpServletResponse)
2017-08-08 17:12:32.444 INFO 19866 --- [          main]
o.s.w.s.handler.SimpleUrlHandlerMapping : Mapped URL path [/webjars/**] onto handler
of type [class org.springframework.web.servlet.resource.ResourceHttpRequestHandler]
2017-08-08 17:12:32.444 INFO 19866 --- [          main]
o.s.w.s.handler.SimpleUrlHandlerMapping : Mapped URL path [/**] onto handler of type
[class org.springframework.web.servlet.resource.ResourceHttpRequestHandler]
2017-08-08 17:12:32.471 INFO 19866 --- [          main]
o.s.w.s.handler.SimpleUrlHandlerMapping : Mapped URL path [/**/favicon.ico] onto
handler of type [class
org.springframework.web.servlet.resource.ResourceHttpRequestHandler]
2017-08-08 17:12:32.600 INFO 19866 --- [          main]
o.s.w.s.v.f.FreeMarkerConfigurer    : ClassTemplateLoader for Spring macros added
to FreeMarker configuration
2017-08-08 17:12:32.681 INFO 19866 --- [          main]
o.s.j.e.a.AnnotationMBeanExporter    : Registering beans for JMX exposure on
startup
2017-08-08 17:12:32.744 INFO 19866 --- [          main]
o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port(s): 8080 (http)

```

```
2017-08-08 17:12:32.750 INFO 19866 --- [          main]
s.f.SampleWebFreeMarkerApplication : Started SampleWebFreeMarkerApplication in
2.172 seconds (JVM running for 2.479)
```

## 15.2. Retrieving Part of the Log File



Retrieving part of the log file is not supported when using Jersey.

To retrieve part of the log file, make a **GET** request to **/actuator/logfile** by using the **Range** header, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/logfile' -i -X GET \
-H 'Range: bytes=0-1023'
```

The preceding example retrieves the first 1024 bytes of the log file. The resulting response is similar to the following:

```
HTTP/1.1 206 Partial Content
Accept-Ranges: bytes
Content-Type: text/plain;charset=UTF-8
Content-Range: bytes 0-1023/4715
Content-Length: 1024
```

```
  .
 /\ / ___ ' _ _ _ _ ( _ ) _ _ _ _ \ \ \ \
( ( ) \ _ _ | ' _ | ' _ | ' _ \ _ | \ \ \ \
 \ \ _ _ ) | | _ | | | | | | ( _ | | ) ) )
 ' | _ _ | . _ | | | | | \ _ , | / / / /
=====|_|=====|___/=/_/_/_/
:: Spring Boot ::
```

```
2017-08-08 17:12:30.910 INFO 19866 --- [          main]
s.f.SampleWebFreeMarkerApplication : Starting SampleWebFreeMarkerApplication
with PID 19866
2017-08-08 17:12:30.913 INFO 19866 --- [          main]
s.f.SampleWebFreeMarkerApplication : No active profile set, falling back to
default profiles: default
2017-08-08 17:12:30.952 INFO 19866 --- [          main]
ConfigServletWebServerApplicationContext : Refreshing
org.springframework.boot.web.servlet.context.AnnotationConfigServletWebServerApplicati
onContext@76b10754: startup date [Tue Aug 08 17:12:30 BST 2017]; root of context
hierarchy
2017-08-08 17:12:31.878 INFO 19866 --- [          main]
o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port(s): 8080 (http
```

# Chapter 16. Loggers (loggers)

The **loggers** endpoint provides access to the application's loggers and the configuration of their levels.

## 16.1. Retrieving All Loggers

To retrieve the application's loggers, make a **GET** request to **/actuator/loggers**, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/loggers' -i -X GET
```

The resulting response is similar to the following:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.spring-boot.actuator.v3+json
Content-Length: 791

{
  "levels" : [ "OFF", "FATAL", "ERROR", "WARN", "INFO", "DEBUG", "TRACE" ],
  "loggers" : {
    "ROOT" : {
      "configuredLevel" : "INFO",
      "effectiveLevel" : "INFO"
    },
    "com.example" : {
      "configuredLevel" : "DEBUG",
      "effectiveLevel" : "DEBUG"
    }
  },
  "groups" : {
    "test" : {
      "configuredLevel" : "INFO",
      "members" : [ "test.member1", "test.member2" ]
    },
    "web" : {
      "members" : [ "org.springframework.core.codec", "org.springframework.http",
"org.springframework.web", "org.springframework.boot.actuate.endpoint.web",
"org.springframework.boot.web.servlet.ServletContextInitializerBeans" ]
    },
    "sql" : {
      "members" : [ "org.springframework.jdbc.core", "org.hibernate.SQL",
"org.jooq.tools.LoggerListener" ]
    }
  }
}
```

### 16.1.1. Response Structure

The response contains details of the application's loggers. The following table describes the structure of the response:

| Path                      | Type   | Description                                   |
|---------------------------|--------|-----------------------------------------------|
| levels                    | Array  | Levels support by the logging system.         |
| loggers                   | Object | Loggers keyed by name.                        |
| groups                    | Object | Logger groups keyed by name                   |
| loggers.*.configuredLevel | String | Configured level of the logger, if any.       |
| loggers.*.effectiveLevel  | String | Effective level of the logger.                |
| groups.*.configuredLevel  | String | Configured level of the logger group, if any. |
| groups.*.members          | Array  | Loggers that are part of this group           |

## 16.2. Retrieving a Single Logger

To retrieve a single logger, make a **GET** request to `/actuator/loggers/{logger.name}`, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/loggers/com.example' -i -X GET
```

The preceding example retrieves information about the logger named `com.example`. The resulting response is similar to the following:

```
HTTP/1.1 200 OK
Content-Disposition: inline;filename=f.txt
Content-Type: application/vnd.spring-boot.actuator.v3+json
Content-Length: 61

{
  "configuredLevel" : "INFO",
  "effectiveLevel" : "INFO"
}
```

### 16.2.1. Response Structure

The response contains details of the requested logger. The following table describes the structure of the response:

| Path            | Type   | Description                             |
|-----------------|--------|-----------------------------------------|
| configuredLevel | String | Configured level of the logger, if any. |

| Path                        | Type                | Description                    |
|-----------------------------|---------------------|--------------------------------|
| <code>effectiveLevel</code> | <code>String</code> | Effective level of the logger. |

## 16.3. Retrieving a Single Group

To retrieve a single group, make a `GET` request to `/actuator/loggers/{group.name}`, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/loggers/test' -i -X GET
```

The preceding example retrieves information about the logger group named `test`. The resulting response is similar to the following:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.spring-boot.actuator.v3+json
Content-Length: 82

{
  "configuredLevel" : "INFO",
  "members" : [ "test.member1", "test.member2" ]
}
```

### 16.3.1. Response Structure

The response contains details of the requested group. The following table describes the structure of the response:

| Path                         | Type                | Description                                   |
|------------------------------|---------------------|-----------------------------------------------|
| <code>configuredLevel</code> | <code>String</code> | Configured level of the logger group, if any. |
| <code>members</code>         | <code>Array</code>  | Loggers that are part of this group           |

## 16.4. Setting a Log Level

To set the level of a logger, make a `POST` request to `/actuator/loggers/{logger.name}` with a JSON body that specifies the configured level for the logger, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/loggers/com.example' -i -X POST \
  -H 'Content-Type: application/json' \
  -d '{"configuredLevel":"debug"}'
```

The preceding example sets the `configuredLevel` of the `com.example` logger to `DEBUG`.

### 16.4.1. Request Structure

The request specifies the desired level of the logger. The following table describes the structure of the request:

| Path                         | Type                | Description                                              |
|------------------------------|---------------------|----------------------------------------------------------|
| <code>configuredLevel</code> | <code>String</code> | Level for the logger. May be omitted to clear the level. |

## 16.5. Setting a Log Level for a Group

To set the level of a logger, make a `POST` request to `/actuator/loggers/{group.name}` with a JSON body that specifies the configured level for the logger group, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/loggers/test' -i -X POST \
  -H 'Content-Type: application/json' \
  -d '{"configuredLevel":"debug"}'
```

The preceding example sets the `configuredLevel` of the `test` logger group to `DEBUG`.

### 16.5.1. Request Structure

The request specifies the desired level of the logger group. The following table describes the structure of the request:

| Path                         | Type                | Description                                              |
|------------------------------|---------------------|----------------------------------------------------------|
| <code>configuredLevel</code> | <code>String</code> | Level for the logger. May be omitted to clear the level. |

## 16.6. Clearing a Log Level

To clear the level of a logger, make a `POST` request to `/actuator/loggers/{logger.name}` with a JSON body containing an empty object, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/loggers/com.example' -i -X POST \
  -H 'Content-Type: application/json' \
  -d '{}'
```

The preceding example clears the configured level of the `com.example` logger.

# Chapter 17. Mappings (mappings)

The `mappings` endpoint provides information about the application's request mappings.

## 17.1. Retrieving the Mappings

To retrieve the mappings, make a `GET` request to `/actuator/mappings`, as shown in the following curl-based example:

```
$ curl 'http://localhost:36251/actuator/mappings' -i -X GET \
-H 'accept-encoding: gzip' \
-H 'user-agent: ReactorNetty/1.1.11' \
-H 'accept: */*' \
```

The resulting response is similar to the following:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.spring-boot.actuator.v3+json
Transfer-Encoding: chunked
Date: Thu, 21 Sep 2023 00:38:58 GMT
Content-Length: 5342

{
  "contexts" : {
    "application" : {
      "mappings" : {
        "dispatcherServlets" : {
          "dispatcherServlet" : [ {
            "handler" : "Actuator root web endpoint",
            "predicate" : "{GET [/actuator], produces [application/vnd.spring-
boot.actuator.v3+json || application/vnd.spring-boot.actuator.v2+json ||
application/json]}" ,
            "details" : {
              "handlerMethod" : {
                "className" :
"org.springframework.boot.actuate.endpoint.web.servlet.WebMvcEndpointHandlerMapping.Web
MvcLinksHandler",
                "name" : "links",
                "descriptor" :
"(Ljakarta/servlet/http/HttpServletRequest;Ljakarta/servlet/http/HttpServletResponse;)
Ljava/util/Map;"
              },
              "requestMappingConditions" : {
                "consumes" : [ ],
                "headers" : [ ],
                "methods" : [ "GET" ],
                "params" : [ ],
                "patterns" : [ "/actuator" ],
```

```

        "produces" : [ {
            "mediaType" : "application/vnd.spring-boot.actuator.v3+json",
            "negated" : false
        }, {
            "mediaType" : "application/vnd.spring-boot.actuator.v2+json",
            "negated" : false
        }, {
            "mediaType" : "application/json",
            "negated" : false
        } ]
    }
}
}, {
    "handler" : "Actuator web endpoint 'mappings'",
    "predicate" : "{GET [/actuator/mappings], produces
[application/vnd.spring-boot.actuator.v3+json || application/vnd.spring-
boot.actuator.v2+json || application/json]}",
    "details" : {
        "handlerMethod" : {
            "className" :
"org.springframework.boot.actuate.endpoint.web.servlet.AbstractWebMvcEndpointHandlerMa
pping.OperationHandler",
            "name" : "handle",
            "descriptor" :
"(Ljakarta/servlet/http/HttpServletRequest;Ljava/util/Map;)Ljava/lang/Object;"
        },
        "requestMappingConditions" : {
            "consumes" : [ ],
            "headers" : [ ],
            "methods" : [ "GET" ],
            "params" : [ ],
            "patterns" : [ "/actuator/mappings" ],
            "produces" : [ {
                "mediaType" : "application/vnd.spring-boot.actuator.v3+json",
                "negated" : false
            }, {
                "mediaType" : "application/vnd.spring-boot.actuator.v2+json",
                "negated" : false
            }, {
                "mediaType" : "application/json",
                "negated" : false
            } ]
        }
    }
}
}, {
    "handler" :
"org.springframework.boot.actuate.autoconfigure.endpoint.web.documentation.MappingsEnd
pointServletDocumentationTests$ExampleController#example()",
    "predicate" : "{POST [/], params [a!=alpha], headers [X-Custom=Foo],
consumes [application/json || !application/xml], produces [text/plain]}",
    "details" : {

```

```

        "handlerMethod" : {
            "className" :
"org.springframework.boot.actuate.autoconfigure.endpoint.web.documentation.MappingsEnd
pointServletDocumentationTests.ExampleController",
            "name" : "example",
            "descriptor" : "()Ljava/lang/String;"
        },
        "requestMappingConditions" : {
            "consumes" : [ {
                "mediaType" : "application/json",
                "negated" : false
            }, {
                "mediaType" : "application/xml",
                "negated" : true
            } ],
            "headers" : [ {
                "name" : "X-Custom",
                "value" : "Foo",
                "negated" : false
            } ],
            "methods" : [ "POST" ],
            "params" : [ {
                "name" : "a",
                "value" : "alpha",
                "negated" : true
            } ],
            "patterns" : [ "/" ],
            "produces" : [ {
                "mediaType" : "text/plain",
                "negated" : false
            } ]
        }
    }, {
        "handler" : "ResourceHttpRequestHandler [classpath [META-
INF/resources/webjars/]]",
        "predicate" : "/webjars/**"
    }, {
        "handler" : "ResourceHttpRequestHandler [classpath [META-INF/resources/],
classpath [resources/], classpath [static/], classpath [public/], ServletContext
[/]]",
        "predicate" : "/*"
    } ]
}, {
    "servletFilters" : [ {
        "servletNameMappings" : [ ],
        "urlPatternMappings" : [ "/*" ],
        "name" : "requestContextFilter",
        "className" :
"org.springframework.boot.web.servlet.filter.OrderedRequestContextFilter"
    }, {

```

```

        "servletNameMappings" : [ ],
        "urlPatternMappings" : [ "/"* ],
        "name" : "formContentFilter",
        "className" :
"org.springframework.boot.web.servlet.filter.OrderedFormContentFilter"
    } ],
    "servlets" : [ {
        "mappings" : [ "/" ],
        "name" : "dispatcherServlet",
        "className" : "org.springframework.web.servlet.DispatcherServlet"
    } ]
}
}
}
}
}

```

### 17.1.1. Response Structure

The response contains details of the application's mappings. The items found in the response depend on the type of web application (reactive or Servlet-based). The following table describes the structure of the common elements of the response:

| Path                                                | Type   | Description                                     |
|-----------------------------------------------------|--------|-------------------------------------------------|
| <code>contexts</code>                               | Object | Application contexts keyed by id.               |
| <code>contexts.*.mappings</code>                    | Object | Mappings in the context, keyed by mapping type. |
| <code>contexts.*.mappings.dispatcherServlets</code> | Object | Dispatcher servlet mappings, if any.            |
| <code>contexts.*.mappings.servletFilters</code>     | Array  | Servlet filter mappings, if any.                |
| <code>contexts.*.mappings.servlets</code>           | Array  | Servlet mappings, if any.                       |
| <code>contexts.*.mappings.dispatcherHandlers</code> | Object | Dispatcher handler mappings, if any.            |
| <code>contexts.*.parentId</code>                    | String | Id of the parent application context, if any.   |

The entries that may be found in `contexts.*.mappings` are described in the following sections.

### 17.1.2. Dispatcher Servlets Response Structure

When using Spring MVC, the response contains details of any `DispatcherServlet` request mappings beneath `contexts.*.mappings.dispatcherServlets`. The following table describes the structure of this section of the response:

| Path                                                                       | Type   | Description                                                                 |
|----------------------------------------------------------------------------|--------|-----------------------------------------------------------------------------|
| *                                                                          | Array  | Dispatcher servlet mappings, if any, keyed by dispatcher servlet bean name. |
| *.[] <b>.details</b>                                                       | Object | Additional implementation-specific details about the mapping. Optional.     |
| *.[] <b>.handler</b>                                                       | String | Handler for the mapping.                                                    |
| *.[] <b>.predicate</b>                                                     | String | Predicate for the mapping.                                                  |
| *.[] <b>.details.handlerMethod</b>                                         | Object | Details of the method, if any, that will handle requests to this mapping.   |
| *.[] <b>.details.handlerMethod.className</b>                               | Varies | Fully qualified name of the class of the method.                            |
| *.[] <b>.details.handlerMethod.name</b>                                    | Varies | Name of the method.                                                         |
| *.[] <b>.details.handlerMethod.descriptor</b>                              | Varies | Descriptor of the method as specified in the Java Language Specification.   |
| *.[] <b>.details.requestMappingConditions</b>                              | Object | Details of the request mapping conditions.                                  |
| *.[] <b>.details.requestMappingConditions.consumes</b>                     | Varies | Details of the consumes condition                                           |
| *.[] <b>.details.requestMappingConditions.consumes.[]<b>.mediaType</b></b> | Varies | Consumed media type.                                                        |
| *.[] <b>.details.requestMappingConditions.consumes.[]<b>.negated</b></b>   | Varies | Whether the media type is negated.                                          |
| *.[] <b>.details.requestMappingConditions.headers</b>                      | Varies | Details of the headers condition.                                           |
| *.[] <b>.details.requestMappingConditions.headers.[]<b>.name</b></b>       | Varies | Name of the header.                                                         |
| *.[] <b>.details.requestMappingConditions.headers.[]<b>.value</b></b>      | Varies | Required value of the header, if any.                                       |
| *.[] <b>.details.requestMappingConditions.headers.[]<b>.negated</b></b>    | Varies | Whether the value is negated.                                               |
| *.[] <b>.details.requestMappingConditions.methods</b>                      | Varies | HTTP methods that are handled.                                              |
| *.[] <b>.details.requestMappingConditions.params</b>                       | Varies | Details of the params condition.                                            |
| *.[] <b>.details.requestMappingConditions.params.[]<b>.name</b></b>        | Varies | Name of the parameter.                                                      |

| Path                                                        | Type   | Description                                            |
|-------------------------------------------------------------|--------|--------------------------------------------------------|
| *.[] details.requestMappingConditions.params.[] value       | Varies | Required value of the parameter, if any.               |
| *.[] details.requestMappingConditions.params.[] negated     | Varies | Whether the value is negated.                          |
| *.[] details.requestMappingConditions.patterns              | Varies | Patterns identifying the paths handled by the mapping. |
| *.[] details.requestMappingConditions.produces              | Varies | Details of the produces condition.                     |
| *.[] details.requestMappingConditions.produces.[] mediaType | Varies | Produced media type.                                   |
| *.[] details.requestMappingConditions.produces.[] negated   | Varies | Whether the media type is negated.                     |

### 17.1.3. Servlets Response Structure

When using the Servlet stack, the response contains details of any **Servlet** mappings beneath `contexts.*.mappings.servlets`. The following table describes the structure of this section of the response:

| Path         | Type   | Description               |
|--------------|--------|---------------------------|
| [] mappings  | Array  | Mappings of the servlet.  |
| [] name      | String | Name of the servlet.      |
| [] className | String | Class name of the servlet |

### 17.1.4. Servlet Filters Response Structure

When using the Servlet stack, the response contains details of any **Filter** mappings beneath `contexts.*.mappings.servletFilters`. The following table describes the structure of this section of the response:

| Path                   | Type   | Description                                          |
|------------------------|--------|------------------------------------------------------|
| [] servletNameMappings | Array  | Names of the servlets to which the filter is mapped. |
| [] urlPatternMappings  | Array  | URL pattern to which the filter is mapped.           |
| [] name                | String | Name of the filter.                                  |
| [] className           | String | Class name of the filter                             |

### 17.1.5. Dispatcher Handlers Response Structure

When using Spring WebFlux, the response contains details of any **DispatcherHandler** request mappings beneath `contexts.*.mappings.dispatcherHandlers`. The following table describes the

structure of this section of the response:

| Path                                                        | Type   | Description                                                                 |
|-------------------------------------------------------------|--------|-----------------------------------------------------------------------------|
| *                                                           | Array  | Dispatcher handler mappings, if any, keyed by dispatcher handler bean name. |
| *.[].details                                                | Object | Additional implementation-specific details about the mapping. Optional.     |
| *.[].handler                                                | String | Handler for the mapping.                                                    |
| *.[].predicate                                              | String | Predicate for the mapping.                                                  |
| *.[].details.requestMappingConditions                       | Object | Details of the request mapping conditions.                                  |
| *.[].details.requestMappingConditions.consumes              | Varies | Details of the consumes condition                                           |
| *.[].details.requestMappingConditions.consumes.[].mediaType | Varies | Consumed media type.                                                        |
| *.[].details.requestMappingConditions.consumes.[].negated   | Varies | Whether the media type is negated.                                          |
| *.[].details.requestMappingConditions.headers               | Varies | Details of the headers condition.                                           |
| *.[].details.requestMappingConditions.headers.[].name       | Varies | Name of the header.                                                         |
| *.[].details.requestMappingConditions.headers.[].value      | Varies | Required value of the header, if any.                                       |
| *.[].details.requestMappingConditions.headers.[].negated    | Varies | Whether the value is negated.                                               |
| *.[].details.requestMappingConditions.methods               | Varies | HTTP methods that are handled.                                              |
| *.[].details.requestMappingConditions.params                | Varies | Details of the params condition.                                            |
| *.[].details.requestMappingConditions.params.[].name        | Varies | Name of the parameter.                                                      |
| *.[].details.requestMappingConditions.params.[].value       | Varies | Required value of the parameter, if any.                                    |
| *.[].details.requestMappingConditions.params.[].negated     | Varies | Whether the value is negated.                                               |
| *.[].details.requestMappingConditions.patterns              | Varies | Patterns identifying the paths handled by the mapping.                      |

| Path                                                          | Type   | Description                                                                 |
|---------------------------------------------------------------|--------|-----------------------------------------------------------------------------|
| *. [].details.requestMappingConditions.produces               | Varies | Details of the produces condition.                                          |
| *. [].details.requestMappingConditions.produces. [].mediaType | Varies | Produced media type.                                                        |
| *. [].details.requestMappingConditions.produces. [].negated   | Varies | Whether the media type is negated.                                          |
| *. [].details.handlerMethod                                   | Object | Details of the method, if any, that will handle requests to this mapping.   |
| *. [].details.handlerMethod.className                         | String | Fully qualified name of the class of the method.                            |
| *. [].details.handlerMethod.name                              | String | Name of the method.                                                         |
| *. [].details.handlerMethod.descriptor                        | String | Descriptor of the method as specified in the Java Language Specification.   |
| *. [].details.handlerFunction                                 | Object | Details of the function, if any, that will handle requests to this mapping. |
| *. [].details.handlerFunction.className                       | String | Fully qualified name of the class of the function.                          |

# Chapter 18. Metrics (metrics)

The **metrics** endpoint provides access to application metrics.

## 18.1. Retrieving Metric Names

To retrieve the names of the available metrics, make a **GET** request to **/actuator/metrics**, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/metrics' -i -X GET
```

The resulting response is similar to the following:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.spring-boot.actuator.v3+json
Content-Length: 154

{
  "names" : [ "jvm.buffer.count", "jvm.buffer.memory.used",
    "jvm.buffer.total.capacity", "jvm.memory.committed", "jvm.memory.max",
    "jvm.memory.used" ]
}
```

### 18.1.1. Response Structure

The response contains details of the metric names. The following table describes the structure of the response:

| Path         | Type         | Description                 |
|--------------|--------------|-----------------------------|
| <b>names</b> | <b>Array</b> | Names of the known metrics. |

## 18.2. Retrieving a Metric

To retrieve a metric, make a **GET** request to **/actuator/metrics/{metric.name}**, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/metrics/jvm.memory.max' -i -X GET
```

The preceding example retrieves information about the metric named **jvm.memory.max**. The resulting response is similar to the following:

```
HTTP/1.1 200 OK
Content-Disposition: inline;filename=f.txt
Content-Type: application/vnd.spring-boot.actuator.v3+json
Content-Length: 555

{
  "name" : "jvm.memory.max",
  "description" : "The maximum amount of memory in bytes that can be used for memory
management",
  "baseUnit" : "bytes",
  "measurements" : [ {
    "statistic" : "VALUE",
    "value" : 2.399141885E9
  } ],
  "availableTags" : [ {
    "tag" : "area",
    "values" : [ "heap", "nonheap" ]
  }, {
    "tag" : "id",
    "values" : [ "CodeHeap 'profiled nmethods'", "G1 Old Gen", "CodeHeap 'non-profiled
nmethods'", "G1 Survivor Space", "Compressed Class Space", "Metaspace", "G1 Eden
Space", "CodeHeap 'non-nmethods'" ]
  } ]
}
```

### 18.2.1. Query Parameters

The endpoint uses query parameters to [drill down](#) into a metric by using its tags. The following table shows the single supported query parameter:

| Parameter        | Description                                                       |
|------------------|-------------------------------------------------------------------|
| <code>tag</code> | A tag to use for drill-down in the form <code>name:value</code> . |

### 18.2.2. Response structure

The response contains details of the metric. The following table describes the structure of the response:

| Path                      | Type                | Description                |
|---------------------------|---------------------|----------------------------|
| <code>name</code>         | <code>String</code> | Name of the metric         |
| <code>description</code>  | <code>String</code> | Description of the metric  |
| <code>baseUnit</code>     | <code>String</code> | Base unit of the metric    |
| <code>measurements</code> | <code>Array</code>  | Measurements of the metric |

| Path                                  | Type   | Description                                                                                            |
|---------------------------------------|--------|--------------------------------------------------------------------------------------------------------|
| <code>measurements[].statistic</code> | String | Statistic of the measurement. (TOTAL, TOTAL_TIME, COUNT, MAX, VALUE, UNKNOWN, ACTIVE_TASKS, DURATION). |
| <code>measurements[].value</code>     | Number | Value of the measurement.                                                                              |
| <code>availableTags</code>            | Array  | Tags that are available for drill-down.                                                                |
| <code>availableTags[].tag</code>      | String | Name of the tag.                                                                                       |
| <code>availableTags[].values</code>   | Array  | Possible values of the tag.                                                                            |

## 18.3. Drilling Down

To drill down into a metric, make a **GET** request to `/actuator/metrics/{metric.name}` using the **tag** query parameter, as shown in the following curl-based example:

```
$ curl
'http://localhost:8080/actuator/metrics/jvm.memory.max?tag=area%3Aanonheap&tag=id%3ACompressed+Class+Space' -i -X GET
```

The preceding example retrieves the `jvm.memory.max` metric, where the `area` tag has a value of `nonheap` and the `id` attribute has a value of `Compressed Class Space`. The resulting response is similar to the following:

```
HTTP/1.1 200 OK
Content-Disposition: inline;filename=f.txt
Content-Type: application/vnd.spring-boot.actuator.v3+json
Content-Length: 263

{
  "name" : "jvm.memory.max",
  "description" : "The maximum amount of memory in bytes that can be used for memory management",
  "baseUnit" : "bytes",
  "measurements" : [ {
    "statistic" : "VALUE",
    "value" : 1.073741824E9
  } ],
  "availableTags" : [ ]
}
```

# Chapter 19. Prometheus (prometheus)

The `prometheus` endpoint provides Spring Boot application's metrics in the format required for scraping by a Prometheus server.

## 19.1. Retrieving All Metrics

To retrieve all metrics, make a `GET` request to `/actuator/prometheus`, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/prometheus' -i -X GET
```

The resulting response is similar to the following:

```
HTTP/1.1 200 OK
Content-Type: text/plain;version=0.0.4;charset=utf-8
Content-Length: 3129

# HELP jvm_buffer_memory_used_bytes An estimate of the memory that the Java virtual
machine is using for this buffer pool
# TYPE jvm_buffer_memory_used_bytes gauge
jvm_buffer_memory_used_bytes{id="mapped - 'non-volatile memory'",} 0.0
jvm_buffer_memory_used_bytes{id="mapped",} 0.0
jvm_buffer_memory_used_bytes{id="direct",} 1.841972E7
# HELP jvm_memory_used_bytes The amount of used memory
# TYPE jvm_memory_used_bytes gauge
jvm_memory_used_bytes{area="nonheap",id="CodeHeap 'profiled nmethods'",} 1.9067776E7
jvm_memory_used_bytes{area="heap",id="G1 Survivor Space",} 3200888.0
jvm_memory_used_bytes{area="heap",id="G1 Old Gen",} 7.6013568E7
jvm_memory_used_bytes{area="nonheap",id="Metaspace",} 8.273708E7
jvm_memory_used_bytes{area="nonheap",id="CodeHeap 'non-nmethods'",} 3720320.0
jvm_memory_used_bytes{area="heap",id="G1 Eden Space",} 2.2020096E7
jvm_memory_used_bytes{area="nonheap",id="Compressed Class Space",} 1.0969408E7
jvm_memory_used_bytes{area="nonheap",id="CodeHeap 'non-profiled nmethods'",} 7473280.0
# HELP jvm_buffer_total_capacity_bytes An estimate of the total capacity of the
buffers in this pool
# TYPE jvm_buffer_total_capacity_bytes gauge
jvm_buffer_total_capacity_bytes{id="mapped - 'non-volatile memory'",} 0.0
jvm_buffer_total_capacity_bytes{id="mapped",} 0.0
jvm_buffer_total_capacity_bytes{id="direct",} 1.8419719E7
# HELP jvm_memory_committed_bytes The amount of memory in bytes that is committed for
the Java virtual machine to use
# TYPE jvm_memory_committed_bytes gauge
jvm_memory_committed_bytes{area="nonheap",id="CodeHeap 'profiled nmethods'",}
1.9791872E7
jvm_memory_committed_bytes{area="heap",id="G1 Survivor Space",} 4194304.0
jvm_memory_committed_bytes{area="heap",id="G1 Old Gen",} 1.03809024E8
jvm_memory_committed_bytes{area="nonheap",id="Metaspace",} 8.3492864E7
```

```
jvm_memory_committed_bytes{area="nonheap",id="CodeHeap 'non-nmethods'",} 3866624.0
jvm_memory_committed_bytes{area="heap",id="G1 Eden Space",} 3.2505856E7
jvm_memory_committed_bytes{area="nonheap",id="Compressed Class Space",} 1.1337728E7
jvm_memory_committed_bytes{area="nonheap",id="CodeHeap 'non-profiled nmethods'",}
7733248.0
# HELP jvm_buffer_count_buffers An estimate of the number of buffers in the pool
# TYPE jvm_buffer_count_buffers gauge
jvm_buffer_count_buffers{id="mapped - 'non-volatile memory'",} 0.0
jvm_buffer_count_buffers{id="mapped",} 0.0
jvm_buffer_count_buffers{id="direct",} 44.0
# HELP jvm_memory_max_bytes The maximum amount of memory in bytes that can be used for
memory management
# TYPE jvm_memory_max_bytes gauge
jvm_memory_max_bytes{area="nonheap",id="CodeHeap 'profiled nmethods'",} 1.21729024E8
jvm_memory_max_bytes{area="heap",id="G1 Survivor Space",} -1.0
jvm_memory_max_bytes{area="heap",id="G1 Old Gen",} 1.073741824E9
jvm_memory_max_bytes{area="nonheap",id="Metaspace",} -1.0
jvm_memory_max_bytes{area="nonheap",id="CodeHeap 'non-nmethods'",} 8196096.0
jvm_memory_max_bytes{area="heap",id="G1 Eden Space",} -1.0
jvm_memory_max_bytes{area="nonheap",id="Compressed Class Space",} 1.073741824E9
jvm_memory_max_bytes{area="nonheap",id="CodeHeap 'non-profiled nmethods'",}
1.2173312E8
```

The default response content type is `text/plain;version=0.0.4`. The endpoint can also produce `application/openmetrics-text;version=1.0.0` when called with an appropriate `Accept` header, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/prometheus' -i -X GET \
-H 'Accept: application/openmetrics-text; version=1.0.0; charset=utf-8'
```

The resulting response is similar to the following:

```
HTTP/1.1 200 OK
Content-Type: application/openmetrics-text;version=1.0.0;charset=utf-8
Content-Length: 3102

# TYPE jvm_buffer_memory_used_bytes gauge
# HELP jvm_buffer_memory_used_bytes An estimate of the memory that the Java virtual
machine is using for this buffer pool
jvm_buffer_memory_used_bytes{id="mapped - 'non-volatile memory'",} 0.0
jvm_buffer_memory_used_bytes{id="mapped",} 0.0
jvm_buffer_memory_used_bytes{id="direct",} 1.841972E7
# TYPE jvm_memory_used_bytes gauge
# HELP jvm_memory_used_bytes The amount of used memory
jvm_memory_used_bytes{area="nonheap",id="CodeHeap 'profiled nmethods'",} 1.9022976E7
jvm_memory_used_bytes{area="heap",id="G1 Survivor Space",} 3200888.0
jvm_memory_used_bytes{area="heap",id="G1 Old Gen",} 7.6013568E7
jvm_memory_used_bytes{area="nonheap",id="Metaspace",} 8.2706328E7
jvm_memory_used_bytes{area="nonheap",id="CodeHeap 'non-nmethods'",} 3783424.0
```

```

jvm_memory_used_bytes{area="heap",id="G1 Eden Space"} 2.097152E7
jvm_memory_used_bytes{area="nonheap",id="Compressed Class Space"} 1.0965488E7
jvm_memory_used_bytes{area="nonheap",id="CodeHeap 'non-profiled nmethods'"} 7425536.0
# TYPE jvm_buffer_total_capacity_bytes gauge
# HELP jvm_buffer_total_capacity_bytes An estimate of the total capacity of the
buffers in this pool
jvm_buffer_total_capacity_bytes{id="mapped - 'non-volatile memory'"} 0.0
jvm_buffer_total_capacity_bytes{id="mapped"} 0.0
jvm_buffer_total_capacity_bytes{id="direct"} 1.8419719E7
# TYPE jvm_memory_committed_bytes gauge
# HELP jvm_memory_committed_bytes The amount of memory in bytes that is committed for
the Java virtual machine to use
jvm_memory_committed_bytes{area="nonheap",id="CodeHeap 'profiled nmethods'"}
1.9791872E7
jvm_memory_committed_bytes{area="heap",id="G1 Survivor Space"} 4194304.0
jvm_memory_committed_bytes{area="heap",id="G1 Old Gen"} 1.03809024E8
jvm_memory_committed_bytes{area="nonheap",id="Metaspace"} 8.3492864E7
jvm_memory_committed_bytes{area="nonheap",id="CodeHeap 'non-nmethods'"} 3866624.0
jvm_memory_committed_bytes{area="heap",id="G1 Eden Space"} 3.2505856E7
jvm_memory_committed_bytes{area="nonheap",id="Compressed Class Space"} 1.1337728E7
jvm_memory_committed_bytes{area="nonheap",id="CodeHeap 'non-profiled nmethods'"}
7733248.0
# TYPE jvm_buffer_count_buffers gauge
# HELP jvm_buffer_count_buffers An estimate of the number of buffers in the pool
jvm_buffer_count_buffers{id="mapped - 'non-volatile memory'"} 0.0
jvm_buffer_count_buffers{id="mapped"} 0.0
jvm_buffer_count_buffers{id="direct"} 44.0
# TYPE jvm_memory_max_bytes gauge
# HELP jvm_memory_max_bytes The maximum amount of memory in bytes that can be used for
memory management
jvm_memory_max_bytes{area="nonheap",id="CodeHeap 'profiled nmethods'"} 1.21729024E8
jvm_memory_max_bytes{area="heap",id="G1 Survivor Space"} -1.0
jvm_memory_max_bytes{area="heap",id="G1 Old Gen"} 1.073741824E9
jvm_memory_max_bytes{area="nonheap",id="Metaspace"} -1.0
jvm_memory_max_bytes{area="nonheap",id="CodeHeap 'non-nmethods'"} 8196096.0
jvm_memory_max_bytes{area="heap",id="G1 Eden Space"} -1.0
jvm_memory_max_bytes{area="nonheap",id="Compressed Class Space"} 1.073741824E9
jvm_memory_max_bytes{area="nonheap",id="CodeHeap 'non-profiled nmethods'"} 1.2173312E8
# EOF

```

### 19.1.1. Query Parameters

The endpoint uses query parameters to limit the samples that it returns. The following table shows the supported query parameters:

| Parameter                  | Description                                                    |
|----------------------------|----------------------------------------------------------------|
| <code>includedNames</code> | Restricts the samples to those that match the names. Optional. |

## 19.2. Retrieving Filtered Metrics

To retrieve metrics matching specific names, make a **GET** request to `/actuator/prometheus` with the `includedNames` query parameter, as shown in the following curl-based example:

```
$ curl
'http://localhost:8080/actuator/prometheus?includedNames=jvm_memory_used_bytes%2Cjvm_m
emory_committed_bytes' -i -X GET
```

The resulting response is similar to the following:

```
HTTP/1.1 200 OK
Content-Type: text/plain;version=0.0.4;charset=utf-8
Content-Length: 1475

# HELP jvm_memory_used_bytes The amount of used memory
# TYPE jvm_memory_used_bytes gauge
jvm_memory_used_bytes{area="nonheap",id="CodeHeap 'profiled nmethods'",} 1.908096E7
jvm_memory_used_bytes{area="heap",id="G1 Survivor Space",} 3200888.0
jvm_memory_used_bytes{area="heap",id="G1 Old Gen",} 7.6013568E7
jvm_memory_used_bytes{area="nonheap",id="Metaspace",} 8.2754904E7
jvm_memory_used_bytes{area="nonheap",id="CodeHeap 'non-nmethods'",} 3720320.0
jvm_memory_used_bytes{area="heap",id="G1 Eden Space",} 2.3068672E7
jvm_memory_used_bytes{area="nonheap",id="Compressed Class Space",} 1.0971176E7
jvm_memory_used_bytes{area="nonheap",id="CodeHeap 'non-profiled nmethods'",} 7474048.0
# HELP jvm_memory_committed_bytes The amount of memory in bytes that is committed for
the Java virtual machine to use
# TYPE jvm_memory_committed_bytes gauge
jvm_memory_committed_bytes{area="nonheap",id="CodeHeap 'profiled nmethods'",}
1.9791872E7
jvm_memory_committed_bytes{area="heap",id="G1 Survivor Space",} 4194304.0
jvm_memory_committed_bytes{area="heap",id="G1 Old Gen",} 1.048576E8
jvm_memory_committed_bytes{area="nonheap",id="Metaspace",} 8.3623936E7
jvm_memory_committed_bytes{area="nonheap",id="CodeHeap 'non-nmethods'",} 3866624.0
jvm_memory_committed_bytes{area="heap",id="G1 Eden Space",} 3.145728E7
jvm_memory_committed_bytes{area="nonheap",id="Compressed Class Space",} 1.1403264E7
jvm_memory_committed_bytes{area="nonheap",id="CodeHeap 'non-profiled nmethods'",}
7733248.0
```

# Chapter 20. Quartz (quartz)

The **quartz** endpoint provides information about jobs and triggers that are managed by the Quartz Scheduler.

## 20.1. Retrieving Registered Groups

Jobs and triggers are managed in groups. To retrieve the list of registered job and trigger groups, make a **GET** request to **/actuator/quartz**, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/quartz' -i -X GET
```

The resulting response is similar to the following:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.spring-boot.actuator.v3+json
Content-Length: 120

{
  "jobs" : {
    "groups" : [ "samples", "tests" ]
  },
  "triggers" : {
    "groups" : [ "samples", "DEFAULT" ]
  }
}
```

### 20.1.1. Response Structure

The response contains the groups names for registered jobs and triggers. The following table describes the structure of the response:

| Path                   | Type  | Description                      |
|------------------------|-------|----------------------------------|
| <b>jobs.groups</b>     | Array | An array of job group names.     |
| <b>triggers.groups</b> | Array | An array of trigger group names. |

## 20.2. Retrieving Registered Job Names

To retrieve the list of registered job names, make a **GET** request to **/actuator/quartz/jobs**, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/quartz/jobs' -i -X GET
```

The resulting response is similar to the following:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.spring-boot.actuator.v3+json
Content-Length: 137
```

```
{
  "groups" : {
    "samples" : {
      "jobs" : [ "jobOne", "jobTwo" ]
    },
    "tests" : {
      "jobs" : [ "jobThree" ]
    }
  }
}
```

### 20.2.1. Response Structure

The response contains the registered job names for each group. The following table describes the structure of the response:

| Path                       | Type                | Description               |
|----------------------------|---------------------|---------------------------|
| <code>groups</code>        | <code>Object</code> | Job groups keyed by name. |
| <code>groups.*.jobs</code> | <code>Array</code>  | An array of job names.    |

## 20.3. Retrieving Registered Trigger Names

To retrieve the list of registered trigger names, make a `GET` request to `/actuator/quartz/triggers`, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/quartz/triggers' -i -X GET
```

The resulting response is similar to the following:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.spring-boot.actuator.v3+json
Content-Length: 229
```

```
{
  "groups" : {
    "samples" : {
      "paused" : false,
      "triggers" : [ "3am-weekdays", "every-day", "once-a-week" ]
    },
    "DEFAULT" : {
      "paused" : false,
      "triggers" : [ "every-hour-tue-thu" ]
    }
  }
}
```

### 20.3.1. Response Structure

The response contains the registered trigger names for each group. The following table describes the structure of the response:

| Path                           | Type                 | Description                           |
|--------------------------------|----------------------|---------------------------------------|
| <code>groups</code>            | <code>Object</code>  | Trigger groups keyed by name.         |
| <code>groups.*.paused</code>   | <code>Boolean</code> | Whether this trigger group is paused. |
| <code>groups.*.triggers</code> | <code>Array</code>   | An array of trigger names.            |

## 20.4. Retrieving Overview of a Job Group

To retrieve an overview of the jobs in a particular group, make a `GET` request to `/actuator/quartz/jobs/{groupName}`, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/quartz/jobs/samples' -i -X GET
```

The preceding example retrieves the summary for jobs in the `samples` group. The resulting response is similar to the following:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.spring-boot.actuator.v3+json
Content-Length: 201

{
  "group" : "samples",
  "jobs" : {
    "jobOne" : {
      "className" : "org.springframework.scheduling.quartz.DelegatingJob"
    },
    "jobTwo" : {
      "className" : "org.quartz.Job"
    }
  }
}
```

### 20.4.1. Response Structure

The response contains an overview of jobs in a particular group. The following table describes the structure of the response:

| Path                          | Type                | Description                                     |
|-------------------------------|---------------------|-------------------------------------------------|
| <code>group</code>            | <code>String</code> | Name of the group.                              |
| <code>jobs</code>             | <code>Object</code> | Job details keyed by name.                      |
| <code>jobs.*.className</code> | <code>String</code> | Fully qualified name of the job implementation. |

## 20.5. Retrieving Overview of a Trigger Group

To retrieve an overview of the triggers in a particular group, make a `GET` request to `/actuator/quartz/triggers/{groupName}`, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/quartz/triggers/tests' -i -X GET
```

The preceding example retrieves the summary for triggers in the `tests` group. The resulting response is similar to the following:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.spring-boot.actuator.v3+json
Content-Length: 1268

{
  "group" : "tests",
  "paused" : false,
  "triggers" : {
```

```

"cron" : {
  "3am-week" : {
    "previousFireTime" : "2020-12-04T03:00:00.000+00:00",
    "nextFireTime" : "2020-12-07T03:00:00.000+00:00",
    "priority" : 3,
    "expression" : "0 0 3 ? * 1,2,3,4,5",
    "timeZone" : "Europe/Paris"
  }
},
"simple" : {
  "every-day" : {
    "nextFireTime" : "2020-12-04T12:00:00.000+00:00",
    "priority" : 7,
    "interval" : 86400000
  }
},
"dailyTimeInterval" : {
  "tue-thu" : {
    "priority" : 5,
    "interval" : 3600000,
    "daysOfWeek" : [ 3, 5 ],
    "startTimeOfDay" : "09:00:00",
    "endTimeOfDay" : "18:00:00"
  }
},
"calendarInterval" : {
  "once-a-week" : {
    "previousFireTime" : "2020-12-02T14:00:00.000+00:00",
    "nextFireTime" : "2020-12-08T14:00:00.000+00:00",
    "priority" : 5,
    "interval" : 604800000,
    "timeZone" : "Etc/UTC"
  }
},
"custom" : {
  "once-a-year-custom" : {
    "previousFireTime" : "2020-07-14T16:00:00.000+00:00",
    "nextFireTime" : "2021-07-14T16:00:00.000+00:00",
    "priority" : 10,
    "trigger" : "com.example.CustomTrigger@fdsfsd"
  }
}
}
}

```

### 20.5.1. Response Structure

The response contains an overview of triggers in a particular group. Trigger implementation specific details are available. The following table describes the structure of the response:

| Path                                          | Type    | Description                                                                                                  |
|-----------------------------------------------|---------|--------------------------------------------------------------------------------------------------------------|
| group                                         | String  | Name of the group.                                                                                           |
| paused                                        | Boolean | Whether the group is paused.                                                                                 |
| triggers.cron                                 | Object  | Cron triggers keyed by name, if any.                                                                         |
| triggers.simple                               | Object  | Simple triggers keyed by name, if any.                                                                       |
| triggers.dailyTimeInterval                    | Object  | Daily time interval triggers keyed by name, if any.                                                          |
| triggers.calendarInterval                     | Object  | Calendar interval triggers keyed by name, if any.                                                            |
| triggers.custom                               | Object  | Any other triggers keyed by name, if any.                                                                    |
| triggers.cron.*.previousFireTime              | String  | Last time the trigger fired, if any.                                                                         |
| triggers.cron.*.nextFireTime                  | String  | Next time at which the Trigger is scheduled to fire, if any.                                                 |
| triggers.cron.*.priority                      | Number  | Priority to use if two triggers have the same scheduled fire time.                                           |
| triggers.cron.*.expression                    | String  | Cron expression to use.                                                                                      |
| triggers.cron.*.timeZone                      | String  | Time zone for which the expression will be resolved, if any.                                                 |
| triggers.simple.*.previousFireTime            | String  | Last time the trigger fired, if any.                                                                         |
| triggers.simple.*.nextFireTime                | String  | Next time at which the Trigger is scheduled to fire, if any.                                                 |
| triggers.simple.*.priority                    | Number  | Priority to use if two triggers have the same scheduled fire time.                                           |
| triggers.simple.*.interval                    | Number  | Interval, in milliseconds, between two executions.                                                           |
| triggers.dailyTimeInterval.*.previousFireTime | String  | Last time the trigger fired, if any.                                                                         |
| triggers.dailyTimeInterval.*.nextFireTime     | String  | Next time at which the Trigger is scheduled to fire, if any.                                                 |
| triggers.dailyTimeInterval.*.priority         | Number  | Priority to use if two triggers have the same scheduled fire time.                                           |
| triggers.dailyTimeInterval.*.interval         | Number  | Interval, in milliseconds, added to the fire time in order to calculate the time of the next trigger repeat. |
| triggers.dailyTimeInterval.*.daysOfWeek       | Array   | An array of days of the week upon which to fire.                                                             |
| triggers.dailyTimeInterval.*.startTimeOfDay   | String  | Time of day to start firing at the given interval, if any.                                                   |

| Path                                                      | Type   | Description                                                                                                  |
|-----------------------------------------------------------|--------|--------------------------------------------------------------------------------------------------------------|
| <code>triggers.dailyTimeInterval.*.endTimeOfDay</code>    | String | Time of day to complete firing at the given interval, if any.                                                |
| <code>triggers.calendarInterval.*.previousFireTime</code> | String | Last time the trigger fired, if any.                                                                         |
| <code>triggers.calendarInterval.*.nextFireTime</code>     | String | Next time at which the Trigger is scheduled to fire, if any.                                                 |
| <code>triggers.calendarInterval.*.priority</code>         | Number | Priority to use if two triggers have the same scheduled fire time.                                           |
| <code>triggers.calendarInterval.*.interval</code>         | Number | Interval, in milliseconds, added to the fire time in order to calculate the time of the next trigger repeat. |
| <code>triggers.calendarInterval.*.timeZone</code>         | String | Time zone within which time calculations will be performed, if any.                                          |
| <code>triggers.custom.*.previousFireTime</code>           | String | Last time the trigger fired, if any.                                                                         |
| <code>triggers.custom.*.nextFireTime</code>               | String | Next time at which the Trigger is scheduled to fire, if any.                                                 |
| <code>triggers.custom.*.priority</code>                   | Number | Priority to use if two triggers have the same scheduled fire time.                                           |
| <code>triggers.custom.*.trigger</code>                    | String | A toString representation of the custom trigger instance.                                                    |

## 20.6. Retrieving Details of a Job

To retrieve the details about a particular job, make a **GET** request to `/actuator/quartz/jobs/{groupName}/{jobName}`, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/quartz/jobs/samples/jobOne' -i -X GET
```

The preceding example retrieves the details of the job identified by the `samples` group and `jobOne` name. The resulting response is similar to the following:

```

HTTP/1.1 200 OK
Content-Type: application/vnd.spring-boot.actuator.v3+json
Content-Length: 609

{
  "group" : "samples",
  "name" : "jobOne",
  "description" : "A sample job",
  "className" : "org.springframework.scheduling.quartz.DelegatingJob",
  "durable" : false,
  "requestRecovery" : false,
  "data" : {
    "password" : "secret",
    "user" : "admin"
  },
  "triggers" : [ {
    "group" : "samples",
    "name" : "every-day",
    "previousFireTime" : "2020-12-04T03:00:00.000+00:00",
    "nextFireTime" : "2020-12-04T12:00:00.000+00:00",
    "priority" : 7
  }, {
    "group" : "samples",
    "name" : "3am-weekdays",
    "nextFireTime" : "2020-12-07T03:00:00.000+00:00",
    "priority" : 3
  } ]
}

```

If a key in the data map is identified as sensitive, its value is sanitized.

### 20.6.1. Response Structure

The response contains the full details of a job including a summary of the triggers associated with it, if any. The triggers are sorted by next fire time and priority. The following table describes the structure of the response:

| Path                         | Type                 | Description                                                                                    |
|------------------------------|----------------------|------------------------------------------------------------------------------------------------|
| <code>group</code>           | <code>String</code>  | Name of the group.                                                                             |
| <code>name</code>            | <code>String</code>  | Name of the job.                                                                               |
| <code>description</code>     | <code>String</code>  | Description of the job, if any.                                                                |
| <code>className</code>       | <code>String</code>  | Fully qualified name of the job implementation.                                                |
| <code>durable</code>         | <code>Boolean</code> | Whether the job should remain stored after it is orphaned.                                     |
| <code>requestRecovery</code> | <code>Boolean</code> | Whether the job should be re-executed if a 'recovery' or 'fail-over' situation is encountered. |

| Path                                     | Type   | Description                                                        |
|------------------------------------------|--------|--------------------------------------------------------------------|
| <code>data.*</code>                      | String | Job data map as key/value pairs, if any.                           |
| <code>triggers</code>                    | Array  | An array of triggers associated to the job, if any.                |
| <code>triggers[].group</code>            | String | Name of the trigger group.                                         |
| <code>triggers[].name</code>             | String | Name of the trigger.                                               |
| <code>triggers[].previousFireTime</code> | String | Last time the trigger fired, if any.                               |
| <code>triggers[].nextFireTime</code>     | String | Next time at which the Trigger is scheduled to fire, if any.       |
| <code>triggers[].priority</code>         | Number | Priority to use if two triggers have the same scheduled fire time. |

## 20.7. Retrieving Details of a Trigger

To retrieve the details about a particular trigger, make a `GET` request to `/actuator/quartz/triggers/{groupName}/{triggerName}`, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/quartz/triggers/samples/example' -i -X GET
```

The preceding example retrieves the details of trigger identified by the `samples` group and `example` name.

### 20.7.1. Common Response Structure

The response has a common structure and an additional object that is specific to the trigger's type. There are five supported types:

- `cron` for `CronTrigger`
- `simple` for `SimpleTrigger`
- `dailyTimeInterval` for `DailyTimeIntervalTrigger`
- `calendarInterval` for `CalendarIntervalTrigger`
- `custom` for any other trigger implementations

The following table describes the structure of the common elements of the response:

| Path                     | Type   | Description                                                                                                                                                 |
|--------------------------|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>group</code>       | String | Name of the group.                                                                                                                                          |
| <code>name</code>        | String | Name of the trigger.                                                                                                                                        |
| <code>description</code> | String | Description of the trigger, if any.                                                                                                                         |
| <code>state</code>       | String | State of the trigger ( <code>NONE</code> , <code>NORMAL</code> , <code>PAUSED</code> , <code>COMPLETE</code> , <code>ERROR</code> , <code>BLOCKED</code> ). |

| Path                           | Type   | Description                                                                                                                                                                                                                |
|--------------------------------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>type</code>              | String | Type of the trigger ( <code>calendarInterval</code> , <code>cron</code> , <code>custom</code> , <code>dailyTimeInterval</code> , <code>simple</code> ). Determines the key of the object containing type-specific details. |
| <code>calendarName</code>      | String | Name of the Calendar associated with this Trigger, if any.                                                                                                                                                                 |
| <code>startTime</code>         | String | Time at which the Trigger should take effect, if any.                                                                                                                                                                      |
| <code>endTime</code>           | String | Time at which the Trigger should quit repeating, regardless of any remaining repeats, if any.                                                                                                                              |
| <code>previousFireTime</code>  | String | Last time the trigger fired, if any.                                                                                                                                                                                       |
| <code>nextFireTime</code>      | String | Next time at which the Trigger is scheduled to fire, if any.                                                                                                                                                               |
| <code>priority</code>          | Number | Priority to use if two triggers have the same scheduled fire time.                                                                                                                                                         |
| <code>finalFireTime</code>     | String | Last time at which the Trigger will fire, if any.                                                                                                                                                                          |
| <code>data</code>              | Object | Job data map keyed by name, if any.                                                                                                                                                                                        |
| <code>calendarInterval</code>  | Object | Calendar time interval trigger details, if any. Present when <code>type</code> is <code>calendarInterval</code> .                                                                                                          |
| <code>custom</code>            | Object | Custom trigger details, if any. Present when <code>type</code> is <code>custom</code> .                                                                                                                                    |
| <code>cron</code>              | Object | Cron trigger details, if any. Present when <code>type</code> is <code>cron</code> .                                                                                                                                        |
| <code>dailyTimeInterval</code> | Object | Daily time interval trigger details, if any. Present when <code>type</code> is <code>dailyTimeInterval</code> .                                                                                                            |
| <code>simple</code>            | Object | Simple trigger details, if any. Present when <code>type</code> is <code>simple</code> .                                                                                                                                    |

### 20.7.2. Cron Trigger Response Structure

A cron trigger defines the cron expression that is used to determine when it has to fire. The resulting response for such a trigger implementation is similar to the following:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.spring-boot.actuator.v3+json
Content-Length: 490
```

```
{
  "group" : "samples",
  "name" : "example",
  "description" : "Example trigger",
  "state" : "NORMAL",
  "type" : "cron",
  "calendarName" : "bankHolidays",
  "startTime" : "2020-11-30T17:00:00.000+00:00",
  "endTime" : "2020-12-30T03:00:00.000+00:00",
  "previousFireTime" : "2020-12-04T03:00:00.000+00:00",
  "nextFireTime" : "2020-12-07T03:00:00.000+00:00",
  "priority" : 3,
  "data" : { },
  "cron" : {
    "expression" : "0 0 3 ? * 1,2,3,4,5",
    "timeZone" : "Europe/Paris"
  }
}
```

Much of the response is common to all trigger types. The structure of the common elements of the response was [described previously](#). The following table describes the structure of the parts of the response that are specific to cron triggers:

| Path                         | Type                | Description                                                  |
|------------------------------|---------------------|--------------------------------------------------------------|
| <code>cron</code>            | <code>Object</code> | Cron trigger specific details.                               |
| <code>cron.expression</code> | <code>String</code> | Cron expression to use.                                      |
| <code>cron.timeZone</code>   | <code>String</code> | Time zone for which the expression will be resolved, if any. |

### 20.7.3. Simple Trigger Response Structure

A simple trigger is used to fire a Job at a given moment in time, and optionally repeated at a specified interval. The resulting response for such a trigger implementation is similar to the following:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.spring-boot.actuator.v3+json
Content-Length: 549
```

```
{
  "group" : "samples",
  "name" : "example",
  "description" : "Example trigger",
  "state" : "NORMAL",
  "type" : "simple",
  "calendarName" : "bankHolidays",
  "startTime" : "2020-11-30T17:00:00.000+00:00",
  "endTime" : "2020-12-30T03:00:00.000+00:00",
  "previousFireTime" : "2020-12-04T03:00:00.000+00:00",
  "nextFireTime" : "2020-12-07T03:00:00.000+00:00",
  "priority" : 7,
  "finalFireTime" : "2020-12-29T17:00:00.000+00:00",
  "data" : { },
  "simple" : {
    "interval" : 86400000,
    "repeatCount" : -1,
    "timesTriggered" : 0
  }
}
```

Much of the response is common to all trigger types. The structure of the common elements of the response was [described previously](#). The following table describes the structure of the parts of the response that are specific to simple triggers:

| Path                               | Type   | Description                                                              |
|------------------------------------|--------|--------------------------------------------------------------------------|
| <code>simple</code>                | Object | Simple trigger specific details.                                         |
| <code>simple.interval</code>       | Number | Interval, in milliseconds, between two executions.                       |
| <code>simple.repeatCount</code>    | Number | Number of times the trigger should repeat, or -1 to repeat indefinitely. |
| <code>simple.timesTriggered</code> | Number | Number of times the trigger has already fired.                           |

#### 20.7.4. Daily Time Interval Trigger Response Structure

A daily time interval trigger is used to fire a Job based upon daily repeating time intervals. The resulting response for such a trigger implementation is similar to the following:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.spring-boot.actuator.v3+json
Content-Length: 667
```

```
{
  "group" : "samples",
  "name" : "example",
  "description" : "Example trigger",
  "state" : "PAUSED",
  "type" : "dailyTimeInterval",
  "calendarName" : "bankHolidays",
  "startTime" : "2020-11-30T17:00:00.000+00:00",
  "endTime" : "2020-12-30T03:00:00.000+00:00",
  "previousFireTime" : "2020-12-04T03:00:00.000+00:00",
  "nextFireTime" : "2020-12-07T03:00:00.000+00:00",
  "priority" : 5,
  "finalFireTime" : "2020-12-30T18:00:00.000+00:00",
  "data" : { },
  "dailyTimeInterval" : {
    "interval" : 3600000,
    "daysOfWeek" : [ 3, 5 ],
    "startTimeOfDay" : "09:00:00",
    "endTimeOfDay" : "18:00:00",
    "repeatCount" : -1,
    "timesTriggered" : 0
  }
}
```

Much of the response is common to all trigger types. The structure of the common elements of the response was [described previously](#). The following table describes the structure of the parts of the response that are specific to daily time interval triggers:

| Path                                          | Type   | Description                                                                                                  |
|-----------------------------------------------|--------|--------------------------------------------------------------------------------------------------------------|
| <code>dailyTimeInterval</code>                | Object | Daily time interval trigger specific details.                                                                |
| <code>dailyTimeInterval.interval</code>       | Number | Interval, in milliseconds, added to the fire time in order to calculate the time of the next trigger repeat. |
| <code>dailyTimeInterval.daysOfWeek</code>     | Array  | An array of days of the week upon which to fire.                                                             |
| <code>dailyTimeInterval.startTimeOfDay</code> | String | Time of day to start firing at the given interval, if any.                                                   |
| <code>dailyTimeInterval.endTimeOfDay</code>   | String | Time of day to complete firing at the given interval, if any.                                                |
| <code>dailyTimeInterval.repeatCount</code>    | Number | Number of times the trigger should repeat, or -1 to repeat indefinitely.                                     |
| <code>dailyTimeInterval.timesTriggered</code> | Number | Number of times the trigger has already fired.                                                               |

## 20.7.5. Calendar Interval Trigger Response Structure

A calendar interval trigger is used to fire a Job based upon repeating calendar time intervals. The resulting response for such a trigger implementation is similar to the following:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.spring-boot.actuator.v3+json
Content-Length: 669

{
  "group" : "samples",
  "name" : "example",
  "description" : "Example trigger",
  "state" : "NORMAL",
  "type" : "calendarInterval",
  "calendarName" : "bankHolidays",
  "startTime" : "2020-11-30T17:00:00.000+00:00",
  "endTime" : "2020-12-30T03:00:00.000+00:00",
  "previousFireTime" : "2020-12-04T03:00:00.000+00:00",
  "nextFireTime" : "2020-12-07T03:00:00.000+00:00",
  "priority" : 5,
  "finalFireTime" : "2020-12-28T17:00:00.000+00:00",
  "data" : { },
  "calendarInterval" : {
    "interval" : 604800000,
    "timeZone" : "Etc/UTC",
    "timesTriggered" : 0,
    "preserveHourOfDayAcrossDaylightSavings" : false,
    "skipDayIfHourDoesNotExist" : false
  }
}
```

Much of the response is common to all trigger types. The structure of the common elements of the response was [described previously](#). The following table describes the structure of the parts of the response that are specific to calendar interval triggers:

| Path                                         | Type   | Description                                                                                                  |
|----------------------------------------------|--------|--------------------------------------------------------------------------------------------------------------|
| <code>calendarInterval</code>                | Object | Calendar interval trigger specific details.                                                                  |
| <code>calendarInterval.interval</code>       | Number | Interval, in milliseconds, added to the fire time in order to calculate the time of the next trigger repeat. |
| <code>calendarInterval.timeZone</code>       | String | Time zone within which time calculations will be performed, if any.                                          |
| <code>calendarInterval.timesTriggered</code> | Number | Number of times the trigger has already fired.                                                               |

| Path                                                                 | Type    | Description                                                                                          |
|----------------------------------------------------------------------|---------|------------------------------------------------------------------------------------------------------|
| <code>calendarInterval.preserveHourOfDayAcrossDaylightSavings</code> | Boolean | Whether to fire the trigger at the same time of day, regardless of daylight saving time transitions. |
| <code>calendarInterval.skipDayIfHourDoesNotExist</code>              | Boolean | Whether to skip if the hour of the day does not exist on a given day.                                |

### 20.7.6. Custom Trigger Response Structure

A custom trigger is any other implementation. The resulting response for such a trigger implementation is similar to the following:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.spring-boot.actuator.v3+json
Content-Length: 457

{
  "group" : "samples",
  "name" : "example",
  "description" : "Example trigger.",
  "state" : "NORMAL",
  "type" : "custom",
  "calendarName" : "bankHolidays",
  "startTime" : "2020-11-30T17:00:00.000+00:00",
  "endTime" : "2020-12-30T03:00:00.000+00:00",
  "previousFireTime" : "2020-12-04T03:00:00.000+00:00",
  "nextFireTime" : "2020-12-07T03:00:00.000+00:00",
  "priority" : 10,
  "custom" : {
    "trigger" : "com.example.CustomTrigger@fdsfsd"
  }
}
```

Much of the response is common to all trigger types. The structure of the common elements of the response was [described previously](#). The following table describes the structure of the parts of the response that are specific to custom triggers:

| Path                        | Type   | Description                                               |
|-----------------------------|--------|-----------------------------------------------------------|
| <code>custom</code>         | Object | Custom trigger specific details.                          |
| <code>custom.trigger</code> | String | A toString representation of the custom trigger instance. |

# Chapter 21. Scheduled Tasks (scheduledtasks)

The `scheduledtasks` endpoint provides information about the application's scheduled tasks.

## 21.1. Retrieving the Scheduled Tasks

To retrieve the scheduled tasks, make a `GET` request to `/actuator/scheduledtasks`, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/scheduledtasks' -i -X GET
```

The resulting response is similar to the following:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.spring-boot.actuator.v3+json
Content-Length: 629

{
  "cron" : [ {
    "runnable" : {
      "target" : "com.example.Processor.processOrders"
    },
    "expression" : "0 0 0/3 1/1 * ?"
  } ],
  "fixedDelay" : [ {
    "runnable" : {
      "target" : "com.example.Processor.purge"
    },
    "initialDelay" : 5000,
    "interval" : 5000
  } ],
  "fixedRate" : [ {
    "runnable" : {
      "target" : "com.example.Processor.retrieveIssues"
    },
    "initialDelay" : 10000,
    "interval" : 3000
  } ],
  "custom" : [ {
    "runnable" : {
      "target" : "com.example.Processor$CustomTriggeredRunnable"
    },
    "trigger" : "com.example.Processor$CustomTrigger@27b5c8d0"
  } ]
}
```

### 21.1.1. Response Structure

The response contains details of the application's scheduled tasks. The following table describes the structure of the response:

| Path                                      | Type   | Description                                                                                 |
|-------------------------------------------|--------|---------------------------------------------------------------------------------------------|
| <code>cron</code>                         | Array  | Cron tasks, if any.                                                                         |
| <code>cron[].runnable.target</code>       | String | Target that will be executed.                                                               |
| <code>cron[].expression</code>            | String | Cron expression.                                                                            |
| <code>fixedDelay</code>                   | Array  | Fixed delay tasks, if any.                                                                  |
| <code>fixedDelay[].runnable.target</code> | String | Target that will be executed.                                                               |
| <code>fixedDelay[].initialDelay</code>    | Number | Delay, in milliseconds, before first execution.                                             |
| <code>fixedDelay[].interval</code>        | Number | Interval, in milliseconds, between the end of the last execution and the start of the next. |
| <code>fixedRate</code>                    | Array  | Fixed rate tasks, if any.                                                                   |
| <code>fixedRate[].runnable.target</code>  | String | Target that will be executed.                                                               |
| <code>fixedRate[].interval</code>         | Number | Interval, in milliseconds, between the start of each execution.                             |
| <code>fixedRate[].initialDelay</code>     | Number | Delay, in milliseconds, before first execution.                                             |
| <code>custom</code>                       | Array  | Tasks with custom triggers, if any.                                                         |
| <code>custom[].runnable.target</code>     | String | Target that will be executed.                                                               |
| <code>custom[].trigger</code>             | String | Trigger for the task.                                                                       |

# Chapter 22. Sessions (sessions)

The `sessions` endpoint provides information about the application's HTTP sessions that are managed by Spring Session.

## 22.1. Retrieving Sessions

To retrieve the sessions, make a `GET` request to `/actuator/sessions`, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/sessions?username=alice' -i -X GET
```

The preceding examples retrieves all of the sessions for the user whose username is `alice`. The resulting response is similar to the following:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.spring-boot.actuator.v3+json
Content-Length: 789

{
  "sessions" : [ {
    "id" : "0b969caa-304a-4e9a-9717-7d863457bd8b",
    "attributeNames" : [ ],
    "creationTime" : "2023-09-20T12:39:00.474276709Z",
    "lastAccessedTime" : "2023-09-21T00:38:15.474285185Z",
    "maxInactiveInterval" : 1800,
    "expired" : false
  }, {
    "id" : "15cdc3bd-09ea-4c36-b50c-1067ed7f6e85",
    "attributeNames" : [ ],
    "creationTime" : "2023-09-20T22:39:00.474709774Z",
    "lastAccessedTime" : "2023-09-21T00:38:48.474710304Z",
    "maxInactiveInterval" : 1800,
    "expired" : false
  }, {
    "id" : "4db5efcc-99cb-4d05-a52c-b49acfbb7ea9",
    "attributeNames" : [ ],
    "creationTime" : "2023-09-20T19:39:00.474703909Z",
    "lastAccessedTime" : "2023-09-21T00:38:23.474706951Z",
    "maxInactiveInterval" : 1800,
    "expired" : false
  } ]
}
```

### 22.1.1. Query Parameters

The endpoint uses query parameters to limit the sessions that it returns. The following table shows the single required query parameter:

| Parameter             | Description       |
|-----------------------|-------------------|
| <code>username</code> | Name of the user. |

### 22.1.2. Response Structure

The response contains details of the matching sessions. The following table describes the structure of the response:

| Path                                        | Type    | Description                                                                         |
|---------------------------------------------|---------|-------------------------------------------------------------------------------------|
| <code>sessions</code>                       | Array   | Sessions for the given username.                                                    |
| <code>sessions[].id</code>                  | String  | ID of the session.                                                                  |
| <code>sessions[].attributeNames</code>      | Array   | Names of the attributes stored in the session.                                      |
| <code>sessions[].creationTime</code>        | String  | Timestamp of when the session was created.                                          |
| <code>sessions[].lastAccessedTime</code>    | String  | Timestamp of when the session was last accessed.                                    |
| <code>sessions[].maxInactiveInterval</code> | Number  | Maximum permitted period of inactivity, in seconds, before the session will expire. |
| <code>sessions[].expired</code>             | Boolean | Whether the session has expired.                                                    |

## 22.2. Retrieving a Single Session

To retrieve a single session, make a `GET` request to `/actuator/sessions/{id}`, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/sessions/4db5efcc-99cb-4d05-a52c-b49acfb7ea9'
-i -X GET
```

The preceding example retrieves the session with the `id` of `4db5efcc-99cb-4d05-a52c-b49acfb7ea9`. The resulting response is similar to the following:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.spring-boot.actuator.v3+json
Content-Length: 240

{
  "id" : "4db5efcc-99cb-4d05-a52c-b49acfbb7ea9",
  "attributeNames" : [ ],
  "creationTime" : "2023-09-20T19:39:00.474703909Z",
  "lastAccessedTime" : "2023-09-21T00:38:23.474706951Z",
  "maxInactiveInterval" : 1800,
  "expired" : false
}
```

### 22.2.1. Response Structure

The response contains details of the requested session. The following table describes the structure of the response:

| Path                             | Type                 | Description                                                                         |
|----------------------------------|----------------------|-------------------------------------------------------------------------------------|
| <code>id</code>                  | <code>String</code>  | ID of the session.                                                                  |
| <code>attributeNames</code>      | <code>Array</code>   | Names of the attributes stored in the session.                                      |
| <code>creationTime</code>        | <code>String</code>  | Timestamp of when the session was created.                                          |
| <code>lastAccessedTime</code>    | <code>String</code>  | Timestamp of when the session was last accessed.                                    |
| <code>maxInactiveInterval</code> | <code>Number</code>  | Maximum permitted period of inactivity, in seconds, before the session will expire. |
| <code>expired</code>             | <code>Boolean</code> | Whether the session has expired.                                                    |

## 22.3. Deleting a Session

To delete a session, make a **DELETE** request to `/actuator/sessions/{id}`, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/sessions/4db5efcc-99cb-4d05-a52c-b49acfbb7ea9'
-i -X DELETE
```

The preceding example deletes the session with the `id` of `4db5efcc-99cb-4d05-a52c-b49acfbb7ea9`.

# Chapter 23. Shutdown (shutdown)

The **shutdown** endpoint is used to shut down the application.

## 23.1. Shutting Down the Application

To shut down the application, make a **POST** request to **/actuator/shutdown**, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/shutdown' -i -X POST
```

A response similar to the following is produced:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.spring-boot.actuator.v3+json
Content-Length: 41

{
  "message" : "Shutting down, bye..."
}
```

### 23.1.1. Response Structure

The response contains details of the result of the shutdown request. The following table describes the structure of the response:

| Path           | Type          | Description                                   |
|----------------|---------------|-----------------------------------------------|
| <b>message</b> | <b>String</b> | Message describing the result of the request. |

# Chapter 24. Application Startup (startup)

The `startup` endpoint provides information about the application's startup sequence.

## 24.1. Retrieving the Application Startup Steps

The application startup steps can either be retrieved as a snapshot (`GET`) or drained from the buffer (`POST`).

### 24.1.1. Retrieving a snapshot of the Application Startup Steps

To retrieve the steps recorded so far during the application startup phase, make a `GET` request to `/actuator/startup`, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/startup' -i -X GET
```

The resulting response is similar to the following:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.spring-boot.actuator.v3+json
Content-Length: 899

{
  "springBootVersion" : "3.0.11-SNAPSHOT",
  "timeline" : {
    "startTime" : "2023-09-21T00:39:01.104362159Z",
    "events" : [ {
      "endTime" : "2023-09-21T00:39:01.202019023Z",
      "duration" : "PT0.000007276S",
      "startTime" : "2023-09-21T00:39:01.202011747Z",
      "startupStep" : {
        "name" : "spring.beans.instantiate",
        "id" : 3,
        "tags" : [ {
          "key" : "beanName",
          "value" : "homeController"
        } ],
        "parentId" : 2
      }
    }, {
      "endTime" : "2023-09-21T00:39:01.202029473Z",
      "duration" : "PT0.000029306S",
      "startTime" : "2023-09-21T00:39:01.202000167Z",
      "startupStep" : {
        "name" : "spring.boot.application.starting",
        "id" : 2,
        "tags" : [ {
          "key" : "mainApplicationClass",
          "value" : "com.example.startup.StartupApplication"
        } ]
      }
    } ]
  }
}
```

### 24.1.2. Draining the Application Startup Steps

To drain and return the steps recorded so far during the application startup phase, make a **POST** request to `/actuator/startup`, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/startup' -i -X POST
```

The resulting response is similar to the following:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.spring-boot.actuator.v3+json
Content-Length: 899
```

```
{
  "springBootVersion" : "3.0.11-SNAPSHOT",
  "timeline" : {
    "startTime" : "2023-09-21T00:39:01.104362159Z",
    "events" : [ {
      "endTime" : "2023-09-21T00:39:01.181759236Z",
      "duration" : "PT0.000390961S",
      "startTime" : "2023-09-21T00:39:01.181368275Z",
      "startupStep" : {
        "name" : "spring.beans.instantiate",
        "id" : 1,
        "tags" : [ {
          "key" : "beanName",
          "value" : "homeController"
        } ],
        "parentId" : 0
      }
    }, {
      "endTime" : "2023-09-21T00:39:01.181836160Z",
      "duration" : "PT0.001659208S",
      "startTime" : "2023-09-21T00:39:01.180176952Z",
      "startupStep" : {
        "name" : "spring.boot.application.starting",
        "id" : 0,
        "tags" : [ {
          "key" : "mainApplicationClass",
          "value" : "com.example.startup.StartupApplication"
        } ]
      }
    }
  ]
}
```

### 24.1.3. Response Structure

The response contains details of the application startup steps. The following table describes the structure of the response:

| Path                            | Type                | Description                                                    |
|---------------------------------|---------------------|----------------------------------------------------------------|
| <code>springBootVersion</code>  | <code>String</code> | Spring Boot version for this application.                      |
| <code>timeline.startTime</code> | <code>String</code> | Start time of the application.                                 |
| <code>timeline.events</code>    | <code>Array</code>  | An array of steps collected during application startup so far. |

| Path                                                    | Type   | Description                                            |
|---------------------------------------------------------|--------|--------------------------------------------------------|
| <code>timeline.events[].startTime</code>                | String | The timestamp of the start of this event.              |
| <code>timeline.events[].endTime</code>                  | String | The timestamp of the end of this event.                |
| <code>timeline.events[].duration</code>                 | String | The precise duration of this event.                    |
| <code>timeline.events[].startupStep.name</code>         | String | The name of the StartupStep.                           |
| <code>timeline.events[].startupStep.id</code>           | Number | The id of this StartupStep.                            |
| <code>timeline.events[].startupStep.parentId</code>     | Number | The parent id for this StartupStep.                    |
| <code>timeline.events[].startupStep.tags</code>         | Array  | An array of key/value pairs with additional step info. |
| <code>timeline.events[].startupStep.tags[].key</code>   | String | The key of the StartupStep Tag.                        |
| <code>timeline.events[].startupStep.tags[].value</code> | String | The value of the StartupStep Tag.                      |

# Chapter 25. Thread Dump (threaddump)

The `threaddump` endpoint provides a thread dump from the application's JVM.

## 25.1. Retrieving the Thread Dump as JSON

To retrieve the thread dump as JSON, make a `GET` request to `/actuator/threaddump` with an appropriate `Accept` header, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/threaddump' -i -X GET \
-H 'Accept: application/json'
```

The resulting response is similar to the following:

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 39004

{
  "threads" : [ {
    "threadName" : "Test worker",
    "threadId" : 1,
    "blockedTime" : -1,
    "blockedCount" : 49,
    "waitedTime" : -1,
    "waitedCount" : 46,
    "lockOwnerId" : -1,
    "daemon" : false,
    "inNative" : false,
    "suspended" : false,
    "threadState" : "RUNNABLE",
    "priority" : 5,
    "stackTrace" : [ {
      "moduleName" : "java.management",
      "moduleVersion" : "17.0.8.1",
      "methodName" : "dumpThreads0",
      "fileName" : "ThreadImpl.java",
      "lineNumber" : -2,
      "nativeMethod" : true,
      "className" : "sun.management.ThreadImpl"
    }, {
      "moduleName" : "java.management",
      "moduleVersion" : "17.0.8.1",
      "methodName" : "dumpAllThreads",
      "fileName" : "ThreadImpl.java",
      "lineNumber" : 521,
      "nativeMethod" : false,
      "className" : "sun.management.ThreadImpl"
    }
  ]
}
```

```

}, {
  "moduleName" : "java.management",
  "moduleVersion" : "17.0.8.1",
  "methodName" : "dumpAllThreads",
  "fileName" : "ThreadImpl.java",
  "lineNumber" : 509,
  "nativeMethod" : false,
  "className" : "sun.management.ThreadImpl"
}, {
  "classLoaderName" : "app",
  "methodName" : "getFormattedThreadDump",
  "fileName" : "ThreadDumpEndpoint.java",
  "lineNumber" : 52,
  "nativeMethod" : false,
  "className" : "org.springframework.boot.actuate.management.ThreadDumpEndpoint"
}, {
  "classLoaderName" : "app",
  "methodName" : "threadDump",
  "fileName" : "ThreadDumpEndpoint.java",
  "lineNumber" : 43,
  "nativeMethod" : false,
  "className" : "org.springframework.boot.actuate.management.ThreadDumpEndpoint"
}, {
  "moduleName" : "java.base",
  "moduleVersion" : "17.0.8.1",
  "methodName" : "invoke0",
  "fileName" : "NativeMethodAccessorImpl.java",
  "lineNumber" : -2,
  "nativeMethod" : true,
  "className" : "jdk.internal.reflect.NativeMethodAccessorImpl"
}, {
  "moduleName" : "java.base",
  "moduleVersion" : "17.0.8.1",
  "methodName" : "invoke",
  "fileName" : "NativeMethodAccessorImpl.java",
  "lineNumber" : 77,
  "nativeMethod" : false,
  "className" : "jdk.internal.reflect.NativeMethodAccessorImpl"
}, {
  "moduleName" : "java.base",
  "moduleVersion" : "17.0.8.1",
  "methodName" : "invoke",
  "fileName" : "DelegatingMethodAccessorImpl.java",
  "lineNumber" : 43,
  "nativeMethod" : false,
  "className" : "jdk.internal.reflect.DelegatingMethodAccessorImpl"
}, {
  "moduleName" : "java.base",
  "moduleVersion" : "17.0.8.1",
  "methodName" : "invoke",
  "fileName" : "Method.java",

```

```

        "lineNumber" : 568,
        "nativeMethod" : false,
        "className" : "java.lang.reflect.Method"
    }, {
        "classLoaderName" : "app",
        "methodName" : "invokeMethod",
        "fileName" : "ReflectionUtils.java",
        "lineNumber" : 281,
        "nativeMethod" : false,
        "className" : "org.springframework.util.ReflectionUtils"
    }, {
        "classLoaderName" : "app",
        "methodName" : "invoke",
        "fileName" : "ReflectiveOperationInvoker.java",
        "lineNumber" : 74,
        "nativeMethod" : false,
        "className" :
"org.springframework.boot.actuate.endpoint.invoke.reflect.ReflectiveOperationInvoker"
    }, {
        "classLoaderName" : "app",
        "methodName" : "invoke",
        "fileName" : "AbstractDiscoveredOperation.java",
        "lineNumber" : 60,
        "nativeMethod" : false,
        "className" :
"org.springframework.boot.actuate.endpoint.annotation.AbstractDiscoveredOperation"
    }, {
        "classLoaderName" : "app",
        "methodName" : "handle",
        "fileName" : "AbstractWebMvcEndpointHandlerMapping.java",
        "lineNumber" : 321,
        "nativeMethod" : false,
        "className" :
"org.springframework.boot.actuate.endpoint.web.servlet.AbstractWebMvcEndpointHandlerMa
pping$ServletWebOperationAdapter"
    }, {
        "classLoaderName" : "app",
        "methodName" : "handle",
        "fileName" : "AbstractWebMvcEndpointHandlerMapping.java",
        "lineNumber" : 428,
        "nativeMethod" : false,
        "className" :
"org.springframework.boot.actuate.endpoint.web.servlet.AbstractWebMvcEndpointHandlerMa
pping$OperationHandler"
    }, {
        "methodName" : "invoke",
        "lineNumber" : -1,
        "nativeMethod" : false,
        "className" : "jdk.internal.reflect.GeneratedMethodAccessor35"
    }, {
        "moduleName" : "java.base",

```

```

    "moduleName" : "java.base",
    "moduleVersion" : "17.0.8.1",
    "methodName" : "invoke",
    "fileName" : "DelegatingMethodAccessorImpl.java",
    "lineNumber" : 43,
    "nativeMethod" : false,
    "className" : "jdk.internal.reflect.DelegatingMethodAccessorImpl"
  }, {
    "moduleName" : "java.base",
    "moduleVersion" : "17.0.8.1",
    "methodName" : "invoke",
    "fileName" : "Method.java",
    "lineNumber" : 568,
    "nativeMethod" : false,
    "className" : "java.lang.reflect.Method"
  }, {
    "className" : "app",
    "methodName" : "doInvoke",
    "fileName" : "InvocableHandlerMethod.java",
    "lineNumber" : 205,
    "nativeMethod" : false,
    "className" : "org.springframework.web.method.support.InvocableHandlerMethod"
  }, {
    "className" : "app",
    "methodName" : "invokeForRequest",
    "fileName" : "InvocableHandlerMethod.java",
    "lineNumber" : 150,
    "nativeMethod" : false,
    "className" : "org.springframework.web.method.support.InvocableHandlerMethod"
  }, {
    "className" : "app",
    "methodName" : "invokeAndHandle",
    "fileName" : "ServletInvocableHandlerMethod.java",
    "lineNumber" : 118,
    "nativeMethod" : false,
    "className" :
"org.springframework.web.servlet.mvc.method.annotation.ServletInvocableHandlerMethod"
  }, {
    "className" : "app",
    "methodName" : "invokeHandlerMethod",
    "fileName" : "RequestMappingHandlerAdapter.java",
    "lineNumber" : 884,
    "nativeMethod" : false,
    "className" :
"org.springframework.web.servlet.mvc.method.annotation.RequestMappingHandlerAdapter"
  }, {
    "className" : "app",
    "methodName" : "handleInternal",
    "fileName" : "RequestMappingHandlerAdapter.java",
    "lineNumber" : 797,
    "nativeMethod" : false,
    "className" :

```

```

"org.springframework.web.servlet.mvc.method.annotation.RequestMappingHandlerAdapter"
  }, {
    "className" : "app",
    "methodName" : "handle",
    "fileName" : "AbstractHandlerMethodAdapter.java",
    "lineNumber" : 87,
    "nativeMethod" : false,
    "className" :
"org.springframework.web.servlet.mvc.method.AbstractHandlerMethodAdapter"
  }, {
    "className" : "app",
    "methodName" : "doDispatch",
    "fileName" : "DispatcherServlet.java",
    "lineNumber" : 1081,
    "nativeMethod" : false,
    "className" : "org.springframework.web.servlet.DispatcherServlet"
  }, {
    "className" : "app",
    "methodName" : "doService",
    "fileName" : "DispatcherServlet.java",
    "lineNumber" : 974,
    "nativeMethod" : false,
    "className" : "org.springframework.web.servlet.DispatcherServlet"
  }, {
    "className" : "app",
    "methodName" : "processRequest",
    "fileName" : "FrameworkServlet.java",
    "lineNumber" : 1011,
    "nativeMethod" : false,
    "className" : "org.springframework.web.servlet.FrameworkServlet"
  }, {
    "className" : "app",
    "methodName" : "doGet",
    "fileName" : "FrameworkServlet.java",
    "lineNumber" : 903,
    "nativeMethod" : false,
    "className" : "org.springframework.web.servlet.FrameworkServlet"
  }, {
    "className" : "app",
    "methodName" : "service",
    "fileName" : "HttpServlet.java",
    "lineNumber" : 527,
    "nativeMethod" : false,
    "className" : "jakarta.servlet.http.HttpServlet"
  }, {
    "className" : "app",
    "methodName" : "service",
    "fileName" : "FrameworkServlet.java",
    "lineNumber" : 885,
    "nativeMethod" : false,
    "className" : "org.springframework.web.servlet.FrameworkServlet"
  }

```

```

}, {
  "className" : "org.springframework.test.web.servlet.TestDispatcherServlet"
}, {
  "className" : "jakarta.servlet.http.HttpServlet"
}, {
  "className" : "org.springframework.mock.web.MockFilterChain$ServletFilterProxy"
}, {
  "className" : "org.springframework.mock.web.MockFilterChain"
}, {
  "className" : "org.springframework.test.web.servlet.MockMvc"
}, {
  "className" :
"org.springframework.boot.actuate.autoconfigure.endpoint.web.documentation.ThreadDumpE
ndpointDocumentationTests"
}, {
  "moduleName" : "java.base",
  "moduleVersion" : "17.0.8.1",
  "methodName" : "invoke0",
  "fileName" : "NativeMethodAccessorImpl.java",
  "lineNumber" : -2,
  "nativeMethod" : true,

```

```

    "className" : "jdk.internal.reflect.NativeMethodAccessorImpl"
  }, {
    "moduleName" : "java.base",
    "moduleVersion" : "17.0.8.1",
    "methodName" : "invoke",
    "fileName" : "NativeMethodAccessorImpl.java",
    "lineNumber" : 77,
    "nativeMethod" : false,
    "className" : "jdk.internal.reflect.NativeMethodAccessorImpl"
  }, {
    "moduleName" : "java.base",
    "moduleVersion" : "17.0.8.1",
    "methodName" : "invoke",
    "fileName" : "DelegatingMethodAccessorImpl.java",
    "lineNumber" : 43,
    "nativeMethod" : false,
    "className" : "jdk.internal.reflect.DelegatingMethodAccessorImpl"
  }, {
    "moduleName" : "java.base",
    "moduleVersion" : "17.0.8.1",
    "methodName" : "invoke",
    "fileName" : "Method.java",
    "lineNumber" : 568,
    "nativeMethod" : false,
    "className" : "java.lang.reflect.Method"
  }, {
    "classLoaderName" : "app",
    "methodName" : "invokeMethod",
    "fileName" : "ReflectionUtils.java",
    "lineNumber" : 727,
    "nativeMethod" : false,
    "className" : "org.junit.platform.commons.util.ReflectionUtils"
  }, {
    "classLoaderName" : "app",
    "methodName" : "proceed",
    "fileName" : "MethodInvocation.java",
    "lineNumber" : 60,
    "nativeMethod" : false,
    "className" : "org.junit.jupiter.engine.execution.MethodInvocation"
  }, {
    "classLoaderName" : "app",
    "methodName" : "proceed",
    "fileName" : "InvocationInterceptorChain.java",
    "lineNumber" : 131,
    "nativeMethod" : false,
    "className" :
"org.junit.jupiter.engine.execution.InvocationInterceptorChain$ValidatingInvocation"
  }, {
    "classLoaderName" : "app",
    "methodName" : "intercept",
    "fileName" : "TimeoutExtension.java",

```

```

    "lineNumber" : 156,
    "nativeMethod" : false,
    "className" : "org.junit.jupiter.engine.extension.TimeoutExtension"
  }, {
    "className" : "app",
    "methodName" : "interceptTestableMethod",
    "fileName" : "TimeoutExtension.java",
    "lineNumber" : 147,
    "nativeMethod" : false,
    "className" : "org.junit.jupiter.engine.extension.TimeoutExtension"
  }, {
    "className" : "app",
    "methodName" : "interceptTestMethod",
    "fileName" : "TimeoutExtension.java",
    "lineNumber" : 86,
    "nativeMethod" : false,
    "className" : "org.junit.jupiter.engine.extension.TimeoutExtension"
  }, {
    "className" : "app",
    "methodName" : "apply",
    "lineNumber" : -1,
    "nativeMethod" : false,
    "className" :
"org.junit.jupiter.engine.descriptor.TestMethodTestDescriptor$$Lambda$196/0x00007ff0e8136d18"
  }, {
    "className" : "app",
    "methodName" : "lambda$ofVoidMethod$0",
    "fileName" : "InterceptingExecutableInvoker.java",
    "lineNumber" : 103,
    "nativeMethod" : false,
    "className" :
"org.junit.jupiter.engine.execution.InterceptingExecutableInvoker$ReflectiveInterceptorCall"
  }, {
    "className" : "app",
    "methodName" : "apply",
    "lineNumber" : -1,
    "nativeMethod" : false,
    "className" :
"org.junit.jupiter.engine.execution.InterceptingExecutableInvoker$ReflectiveInterceptorCall$$Lambda$197/0x00007ff0e8137138"
  }, {
    "className" : "app",
    "methodName" : "lambda$invoke$0",
    "fileName" : "InterceptingExecutableInvoker.java",
    "lineNumber" : 93,
    "nativeMethod" : false,
    "className" : "org.junit.jupiter.engine.execution.InterceptingExecutableInvoker"
  }, {
    "className" : "app",

```

```

        "methodName" : "apply",
        "lineNumber" : -1,
        "nativeMethod" : false,
        "className" :
"org.junit.jupiter.engine.execution.InterceptingExecutableInvoker$$Lambda$464/0x00007f
f0e81df500"
    }, {
        "classLoaderName" : "app",
        "methodName" : "proceed",
        "fileName" : "InvocationInterceptorChain.java",
        "lineNumber" : 106,
        "nativeMethod" : false,
        "className" :
"org.junit.jupiter.engine.execution.InvocationInterceptorChain$InterceptedInvocation"
    }, {
        "classLoaderName" : "app",
        "methodName" : "proceed",
        "fileName" : "InvocationInterceptorChain.java",
        "lineNumber" : 64,
        "nativeMethod" : false,
        "className" : "org.junit.jupiter.engine.execution.InvocationInterceptorChain"
    }, {
        "classLoaderName" : "app",
        "methodName" : "chainAndInvoke",
        "fileName" : "InvocationInterceptorChain.java",
        "lineNumber" : 45,
        "nativeMethod" : false,
        "className" : "org.junit.jupiter.engine.execution.InvocationInterceptorChain"
    }, {
        "classLoaderName" : "app",
        "methodName" : "invoke",
        "fileName" : "InvocationInterceptorChain.java",
        "lineNumber" : 37,
        "nativeMethod" : false,
        "className" : "org.junit.jupiter.engine.execution.InvocationInterceptorChain"
    }, {
        "classLoaderName" : "app",
        "methodName" : "invoke",
        "fileName" : "InterceptingExecutableInvoker.java",
        "lineNumber" : 92,
        "nativeMethod" : false,
        "className" : "org.junit.jupiter.engine.execution.InterceptingExecutableInvoker"
    }, {
        "classLoaderName" : "app",
        "methodName" : "invoke",
        "fileName" : "InterceptingExecutableInvoker.java",
        "lineNumber" : 86,
        "nativeMethod" : false,
        "className" : "org.junit.jupiter.engine.execution.InterceptingExecutableInvoker"
    }, {
        "classLoaderName" : "app",

```

```

    "methodName" : "lambda$invokeTestMethod$7",
    "fileName" : "TestMethodTestDescriptor.java",
    "lineNumber" : 217,
    "nativeMethod" : false,
    "className" : "org.junit.jupiter.engine.descriptor.TestMethodTestDescriptor"
  }, {
    "classLoaderName" : "app",
    "methodName" : "execute",
    "lineNumber" : -1,
    "nativeMethod" : false,
    "className" :
"org.junit.jupiter.engine.descriptor.TestMethodTestDescriptor$$Lambda$949/0x00007ff0e8
599478"
  }, {
    "classLoaderName" : "app",
    "methodName" : "execute",
    "fileName" : "ThrowableCollector.java",
    "lineNumber" : 73,
    "nativeMethod" : false,
    "className" :
"org.junit.platform.engine.support.hierarchical.ThrowableCollector"
  }, {
    "classLoaderName" : "app",
    "methodName" : "invokeTestMethod",
    "fileName" : "TestMethodTestDescriptor.java",
    "lineNumber" : 213,
    "nativeMethod" : false,
    "className" : "org.junit.jupiter.engine.descriptor.TestMethodTestDescriptor"
  }, {
    "classLoaderName" : "app",
    "methodName" : "execute",
    "fileName" : "TestMethodTestDescriptor.java",
    "lineNumber" : 138,
    "nativeMethod" : false,
    "className" : "org.junit.jupiter.engine.descriptor.TestMethodTestDescriptor"
  }, {
    "classLoaderName" : "app",
    "methodName" : "execute",
    "fileName" : "TestMethodTestDescriptor.java",
    "lineNumber" : 68,
    "nativeMethod" : false,
    "className" : "org.junit.jupiter.engine.descriptor.TestMethodTestDescriptor"
  }, {
    "classLoaderName" : "app",
    "methodName" : "lambda$executeRecursively$6",
    "fileName" : "NodeTestTask.java",
    "lineNumber" : 151,
    "nativeMethod" : false,
    "className" : "org.junit.platform.engine.support.hierarchical.NodeTestTask"
  }, {
    "classLoaderName" : "app",

```

```

    "methodName" : "execute",
    "lineNumber" : -1,
    "nativeMethod" : false,
    "className" :
"org.junit.platform.engine.support.hierarchical.NodeTestTask$$Lambda$299/0x00007ff0e81
547c0"
  }, {
    "classLoaderName" : "app",
    "methodName" : "execute",
    "fileName" : "ThrowableCollector.java",
    "lineNumber" : 73,
    "nativeMethod" : false,
    "className" :
"org.junit.platform.engine.support.hierarchical.ThrowableCollector"
  }, {
    "classLoaderName" : "app",
    "methodName" : "lambda$executeRecursively$8",
    "fileName" : "NodeTestTask.java",
    "lineNumber" : 141,
    "nativeMethod" : false,
    "className" : "org.junit.platform.engine.support.hierarchical.NodeTestTask"
  }, {
    "classLoaderName" : "app",
    "methodName" : "invoke",
    "lineNumber" : -1,
    "nativeMethod" : false,
    "className" :
"org.junit.platform.engine.support.hierarchical.NodeTestTask$$Lambda$298/0x00007ff0e81
54598"
  }, {
    "classLoaderName" : "app",
    "methodName" : "around",
    "fileName" : "Node.java",
    "lineNumber" : 137,
    "nativeMethod" : false,
    "className" : "org.junit.platform.engine.support.hierarchical.Node"
  }, {
    "classLoaderName" : "app",
    "methodName" : "lambda$executeRecursively$9",
    "fileName" : "NodeTestTask.java",
    "lineNumber" : 139,
    "nativeMethod" : false,
    "className" : "org.junit.platform.engine.support.hierarchical.NodeTestTask"
  }, {
    "classLoaderName" : "app",
    "methodName" : "execute",
    "lineNumber" : -1,
    "nativeMethod" : false,
    "className" :
"org.junit.platform.engine.support.hierarchical.NodeTestTask$$Lambda$297/0x00007ff0e81
54170"

```

```

}, {
  "classNameName" : "app",
  "methodName" : "execute",
  "fileName" : "ThrowableCollector.java",
  "lineNumber" : 73,
  "nativeMethod" : false,
  "className" :
"org.junit.platform.engine.support.hierarchical.ThrowableCollector"
}, {
  "classNameName" : "app",
  "methodName" : "executeRecursively",
  "fileName" : "NodeTestTask.java",
  "lineNumber" : 138,
  "nativeMethod" : false,
  "className" : "org.junit.platform.engine.support.hierarchical.NodeTestTask"
}, {
  "classNameName" : "app",
  "methodName" : "execute",
  "fileName" : "NodeTestTask.java",
  "lineNumber" : 95,
  "nativeMethod" : false,
  "className" : "org.junit.platform.engine.support.hierarchical.NodeTestTask"
}, {
  "classNameName" : "app",
  "methodName" : "accept",
  "lineNumber" : -1,
  "nativeMethod" : false,
  "className" :
"org.junit.platform.engine.support.hierarchical.SameThreadHierarchicalTestExecutorService$$Lambda$303/0x00007ff0e81552d8"
}, {
  "moduleName" : "java.base",
  "moduleVersion" : "17.0.8.1",
  "methodName" : "forEach",
  "fileName" : "ArrayList.java",
  "lineNumber" : 1511,
  "nativeMethod" : false,
  "className" : "java.util.ArrayList"
}, {
  "classNameName" : "app",
  "methodName" : "invokeAll",
  "fileName" : "SameThreadHierarchicalTestExecutorService.java",
  "lineNumber" : 41,
  "nativeMethod" : false,
  "className" :
"org.junit.platform.engine.support.hierarchical.SameThreadHierarchicalTestExecutorService"
}, {
  "classNameName" : "app",
  "methodName" : "lambda$executeRecursively$6",
  "fileName" : "NodeTestTask.java",

```

```

    "lineNumber" : 155,
    "nativeMethod" : false,
    "className" : "org.junit.platform.engine.support.hierarchical.NodeTestTask"
  }, {
    "classLoaderName" : "app",
    "methodName" : "execute",
    "lineNumber" : -1,
    "nativeMethod" : false,
    "className" :
"org.junit.platform.engine.support.hierarchical.NodeTestTask$$Lambda$299/0x00007ff0e81
547c0"
  }, {
    "classLoaderName" : "app",
    "methodName" : "execute",
    "fileName" : "ThrowableCollector.java",
    "lineNumber" : 73,
    "nativeMethod" : false,
    "className" :
"org.junit.platform.engine.support.hierarchical.ThrowableCollector"
  }, {
    "classLoaderName" : "app",
    "methodName" : "lambda$executeRecursively$8",
    "fileName" : "NodeTestTask.java",
    "lineNumber" : 141,
    "nativeMethod" : false,
    "className" : "org.junit.platform.engine.support.hierarchical.NodeTestTask"
  }, {
    "classLoaderName" : "app",
    "methodName" : "invoke",
    "lineNumber" : -1,
    "nativeMethod" : false,
    "className" :
"org.junit.platform.engine.support.hierarchical.NodeTestTask$$Lambda$298/0x00007ff0e81
54598"
  }, {
    "classLoaderName" : "app",
    "methodName" : "around",
    "fileName" : "Node.java",
    "lineNumber" : 137,
    "nativeMethod" : false,
    "className" : "org.junit.platform.engine.support.hierarchical.Node"
  }, {
    "classLoaderName" : "app",
    "methodName" : "lambda$executeRecursively$9",
    "fileName" : "NodeTestTask.java",
    "lineNumber" : 139,
    "nativeMethod" : false,
    "className" : "org.junit.platform.engine.support.hierarchical.NodeTestTask"
  }, {
    "classLoaderName" : "app",
    "methodName" : "execute",

```

```

        "lineNumber" : -1,
        "nativeMethod" : false,
        "className" :
"org.junit.platform.engine.support.hierarchical.NodeTestTask$$Lambda$297/0x00007ff0e81
54170"
    }, {
        "classLoaderName" : "app",
        "methodName" : "execute",
        "fileName" : "ThrowableCollector.java",
        "lineNumber" : 73,
        "nativeMethod" : false,
        "className" :
"org.junit.platform.engine.support.hierarchical.ThrowableCollector"
    }, {
        "classLoaderName" : "app",
        "methodName" : "executeRecursively",
        "fileName" : "NodeTestTask.java",
        "lineNumber" : 138,
        "nativeMethod" : false,
        "className" : "org.junit.platform.engine.support.hierarchical.NodeTestTask"
    }, {
        "classLoaderName" : "app",
        "methodName" : "execute",
        "fileName" : "NodeTestTask.java",
        "lineNumber" : 95,
        "nativeMethod" : false,
        "className" : "org.junit.platform.engine.support.hierarchical.NodeTestTask"
    }, {
        "classLoaderName" : "app",
        "methodName" : "accept",
        "lineNumber" : -1,
        "nativeMethod" : false,
        "className" :
"org.junit.platform.engine.support.hierarchical.SameThreadHierarchicalTestExecutorServ
ice$$Lambda$303/0x00007ff0e81552d8"
    }, {
        "moduleName" : "java.base",
        "moduleVersion" : "17.0.8.1",
        "methodName" : "forEach",
        "fileName" : "ArrayList.java",
        "lineNumber" : 1511,
        "nativeMethod" : false,
        "className" : "java.util.ArrayList"
    }, {
        "classLoaderName" : "app",
        "methodName" : "invokeAll",
        "fileName" : "SameThreadHierarchicalTestExecutorService.java",
        "lineNumber" : 41,
        "nativeMethod" : false,
        "className" :
"org.junit.platform.engine.support.hierarchical.SameThreadHierarchicalTestExecutorServ

```

```

ice"
    }, {
        "className" : "app",
        "methodName" : "lambda$executeRecursively$6",
        "fileName" : "NodeTestTask.java",
        "lineNumber" : 155,
        "nativeMethod" : false,
        "className" : "org.junit.platform.engine.support.hierarchical.NodeTestTask"
    }, {
        "className" : "app",
        "methodName" : "execute",
        "lineNumber" : -1,
        "nativeMethod" : false,
        "className" :
"org.junit.platform.engine.support.hierarchical.NodeTestTask$$Lambda$299/0x00007ff0e81
547c0"
    }, {
        "className" : "app",
        "methodName" : "execute",
        "fileName" : "ThrowableCollector.java",
        "lineNumber" : 73,
        "nativeMethod" : false,
        "className" :
"org.junit.platform.engine.support.hierarchical.ThrowableCollector"
    }, {
        "className" : "app",
        "methodName" : "lambda$executeRecursively$8",
        "fileName" : "NodeTestTask.java",
        "lineNumber" : 141,
        "nativeMethod" : false,
        "className" : "org.junit.platform.engine.support.hierarchical.NodeTestTask"
    }, {
        "className" : "app",
        "methodName" : "invoke",
        "lineNumber" : -1,
        "nativeMethod" : false,
        "className" :
"org.junit.platform.engine.support.hierarchical.NodeTestTask$$Lambda$298/0x00007ff0e81
54598"
    }, {
        "className" : "app",
        "methodName" : "around",
        "fileName" : "Node.java",
        "lineNumber" : 137,
        "nativeMethod" : false,
        "className" : "org.junit.platform.engine.support.hierarchical.Node"
    }, {
        "className" : "app",
        "methodName" : "lambda$executeRecursively$9",
        "fileName" : "NodeTestTask.java",
        "lineNumber" : 139,

```

```

    "nativeMethod" : false,
    "className" : "org.junit.platform.engine.support.hierarchical.NodeTestTask"
  }, {
    "classLoaderName" : "app",
    "methodName" : "execute",
    "lineNumber" : -1,
    "nativeMethod" : false,
    "className" :
"org.junit.platform.engine.support.hierarchical.NodeTestTask$$Lambda$297/0x00007ff0e81
54170"
  }, {
    "classLoaderName" : "app",
    "methodName" : "execute",
    "fileName" : "ThrowableCollector.java",
    "lineNumber" : 73,
    "nativeMethod" : false,
    "className" :
"org.junit.platform.engine.support.hierarchical.ThrowableCollector"
  }, {
    "classLoaderName" : "app",
    "methodName" : "executeRecursively",
    "fileName" : "NodeTestTask.java",
    "lineNumber" : 138,
    "nativeMethod" : false,
    "className" : "org.junit.platform.engine.support.hierarchical.NodeTestTask"
  }, {
    "classLoaderName" : "app",
    "methodName" : "execute",
    "fileName" : "NodeTestTask.java",
    "lineNumber" : 95,
    "nativeMethod" : false,
    "className" : "org.junit.platform.engine.support.hierarchical.NodeTestTask"
  }, {
    "classLoaderName" : "app",
    "methodName" : "submit",
    "fileName" : "SameThreadHierarchicalTestExecutorService.java",
    "lineNumber" : 35,
    "nativeMethod" : false,
    "className" :
"org.junit.platform.engine.support.hierarchical.SameThreadHierarchicalTestExecutorServ
ice"
  }, {
    "classLoaderName" : "app",
    "methodName" : "execute",
    "fileName" : "HierarchicalTestExecutor.java",
    "lineNumber" : 57,
    "nativeMethod" : false,
    "className" :
"org.junit.platform.engine.support.hierarchical.HierarchicalTestExecutor"
  }, {
    "classLoaderName" : "app",

```

```

        "methodName" : "execute",
        "fileName" : "HierarchicalTestEngine.java",
        "lineNumber" : 54,
        "nativeMethod" : false,
        "className" :
"org.junit.platform.engine.support.hierarchical.HierarchicalTestEngine"
    }, {
        "classLoaderName" : "app",
        "methodName" : "execute",
        "fileName" : "EngineExecutionOrchestrator.java",
        "lineNumber" : 147,
        "nativeMethod" : false,
        "className" : "org.junit.platform.launcher.core.EngineExecutionOrchestrator"
    }, {
        "classLoaderName" : "app",
        "methodName" : "execute",
        "fileName" : "EngineExecutionOrchestrator.java",
        "lineNumber" : 127,
        "nativeMethod" : false,
        "className" : "org.junit.platform.launcher.core.EngineExecutionOrchestrator"
    }, {
        "classLoaderName" : "app",
        "methodName" : "execute",
        "fileName" : "EngineExecutionOrchestrator.java",
        "lineNumber" : 90,
        "nativeMethod" : false,
        "className" : "org.junit.platform.launcher.core.EngineExecutionOrchestrator"
    }, {
        "classLoaderName" : "app",
        "methodName" : "lambda$execute$0",
        "fileName" : "EngineExecutionOrchestrator.java",
        "lineNumber" : 55,
        "nativeMethod" : false,
        "className" : "org.junit.platform.launcher.core.EngineExecutionOrchestrator"
    }, {
        "classLoaderName" : "app",
        "methodName" : "accept",
        "lineNumber" : -1,
        "nativeMethod" : false,
        "className" :
"org.junit.platform.launcher.core.EngineExecutionOrchestrator$$Lambda$251/0x00007ff0e8
141eb8"
    }, {
        "classLoaderName" : "app",
        "methodName" : "withInterceptedStreams",
        "fileName" : "EngineExecutionOrchestrator.java",
        "lineNumber" : 102,
        "nativeMethod" : false,
        "className" : "org.junit.platform.launcher.core.EngineExecutionOrchestrator"
    }, {
        "classLoaderName" : "app",

```

```

        "methodName" : "execute",
        "fileName" : "EngineExecutionOrchestrator.java",
        "lineNumber" : 54,
        "nativeMethod" : false,
        "className" : "org.junit.platform.launcher.core.EngineExecutionOrchestrator"
    }, {
        "classLoaderName" : "app",
        "methodName" : "execute",
        "fileName" : "DefaultLauncher.java",
        "lineNumber" : 114,
        "nativeMethod" : false,
        "className" : "org.junit.platform.launcher.core.DefaultLauncher"
    }, {
        "classLoaderName" : "app",
        "methodName" : "execute",
        "fileName" : "DefaultLauncher.java",
        "lineNumber" : 86,
        "nativeMethod" : false,
        "className" : "org.junit.platform.launcher.core.DefaultLauncher"
    }, {
        "classLoaderName" : "app",
        "methodName" : "execute",
        "fileName" : "DefaultLauncherSession.java",
        "lineNumber" : 86,
        "nativeMethod" : false,
        "className" :
"org.junit.platform.launcher.core.DefaultLauncherSession$DelegatingLauncher"
    }, {
        "classLoaderName" : "app",
        "methodName" : "execute",
        "fileName" : "SessionPerRequestLauncher.java",
        "lineNumber" : 53,
        "nativeMethod" : false,
        "className" : "org.junit.platform.launcher.core.SessionPerRequestLauncher"
    }, {
        "methodName" : "processAllTestClasses",
        "fileName" : "JUnitPlatformTestClassProcessor.java",
        "lineNumber" : 99,
        "nativeMethod" : false,
        "className" :
"org.gradle.api.internal.tasks.testing.junitplatform.JUnitPlatformTestClassProcessor$C
ollectAllTestClassesExecutor"
    }, {
        "methodName" : "access$000",
        "fileName" : "JUnitPlatformTestClassProcessor.java",
        "lineNumber" : 79,
        "nativeMethod" : false,
        "className" :
"org.gradle.api.internal.tasks.testing.junitplatform.JUnitPlatformTestClassProcessor$C
ollectAllTestClassesExecutor"
    }, {

```

```

        "methodName" : "stop",
        "fileName" : "JUnitPlatformTestClassProcessor.java",
        "lineNumber" : 75,
        "nativeMethod" : false,
        "className" :
"org.gradle.api.internal.tasks.testing.junitplatform.JUnitPlatformTestClassProcessor"
    }, {
        "methodName" : "stop",
        "fileName" : "SuiteTestClassProcessor.java",
        "lineNumber" : 62,
        "nativeMethod" : false,
        "className" : "org.gradle.api.internal.tasks.testing.SuiteTestClassProcessor"
    }, {
        "moduleName" : "java.base",
        "moduleVersion" : "17.0.8.1",
        "methodName" : "invoke0",
        "fileName" : "NativeMethodAccessorImpl.java",
        "lineNumber" : -2,
        "nativeMethod" : true,
        "className" : "jdk.internal.reflect.NativeMethodAccessorImpl"
    }, {
        "moduleName" : "java.base",
        "moduleVersion" : "17.0.8.1",
        "methodName" : "invoke",
        "fileName" : "NativeMethodAccessorImpl.java",
        "lineNumber" : 77,
        "nativeMethod" : false,
        "className" : "jdk.internal.reflect.NativeMethodAccessorImpl"
    }, {
        "moduleName" : "java.base",
        "moduleVersion" : "17.0.8.1",
        "methodName" : "invoke",
        "fileName" : "DelegatingMethodAccessorImpl.java",
        "lineNumber" : 43,
        "nativeMethod" : false,
        "className" : "jdk.internal.reflect.DelegatingMethodAccessorImpl"
    }, {
        "moduleName" : "java.base",
        "moduleVersion" : "17.0.8.1",
        "methodName" : "invoke",
        "fileName" : "Method.java",
        "lineNumber" : 568,
        "nativeMethod" : false,
        "className" : "java.lang.reflect.Method"
    }, {
        "methodName" : "dispatch",
        "fileName" : "ReflectionDispatch.java",
        "lineNumber" : 36,
        "nativeMethod" : false,
        "className" : "org.gradle.internal.dispatch.ReflectionDispatch"
    }, {

```

```

    "methodName" : "dispatch",
    "fileName" : "ReflectionDispatch.java",
    "lineNumber" : 24,
    "nativeMethod" : false,
    "className" : "org.gradle.internal.dispatch.ReflectionDispatch"
}, {
    "methodName" : "dispatch",
    "fileName" : "ContextClassLoaderDispatch.java",
    "lineNumber" : 33,
    "nativeMethod" : false,
    "className" : "org.gradle.internal.dispatch.ContextClassLoaderDispatch"
}, {
    "methodName" : "invoke",
    "fileName" : "ProxyDispatchAdapter.java",
    "lineNumber" : 94,
    "nativeMethod" : false,
    "className" :
"org.gradle.internal.dispatch.ProxyDispatchAdapter$DispatchingInvocationHandler"
}, {
    "moduleName" : "jdk.proxy1",
    "methodName" : "stop",
    "lineNumber" : -1,
    "nativeMethod" : false,
    "className" : "jdk.proxy1.$Proxy2"
}, {
    "methodName" : "run",
    "fileName" : "TestWorker.java",
    "lineNumber" : 193,
    "nativeMethod" : false,
    "className" : "org.gradle.api.internal.tasks.testing.worker.TestWorker$3"
}, {
    "methodName" : "executeAndMaintainThreadName",
    "fileName" : "TestWorker.java",
    "lineNumber" : 129,
    "nativeMethod" : false,
    "className" : "org.gradle.api.internal.tasks.testing.worker.TestWorker"
}, {
    "methodName" : "execute",
    "fileName" : "TestWorker.java",
    "lineNumber" : 100,
    "nativeMethod" : false,
    "className" : "org.gradle.api.internal.tasks.testing.worker.TestWorker"
}, {
    "methodName" : "execute",
    "fileName" : "TestWorker.java",
    "lineNumber" : 60,
    "nativeMethod" : false,
    "className" : "org.gradle.api.internal.tasks.testing.worker.TestWorker"
}, {
    "methodName" : "execute",
    "fileName" : "ActionExecutionWorker.java",

```

```

        "lineNumber" : 56,
        "nativeMethod" : false,
        "className" : "org.gradle.process.internal.worker.child.ActionExecutionWorker"
    }, {
        "methodName" : "call",
        "fileName" : "SystemApplicationClassLoaderWorker.java",
        "lineNumber" : 113,
        "nativeMethod" : false,
        "className" :
"org.gradle.process.internal.worker.child.SystemApplicationClassLoaderWorker"
    }, {
        "methodName" : "call",
        "fileName" : "SystemApplicationClassLoaderWorker.java",
        "lineNumber" : 65,
        "nativeMethod" : false,
        "className" :
"org.gradle.process.internal.worker.child.SystemApplicationClassLoaderWorker"
    }, {
        "classLoaderName" : "app",
        "methodName" : "run",
        "fileName" : "GradleWorkerMain.java",
        "lineNumber" : 69,
        "nativeMethod" : false,
        "className" : "worker.org.gradle.process.internal.worker.GradleWorkerMain"
    }, {
        "classLoaderName" : "app",
        "methodName" : "main",
        "fileName" : "GradleWorkerMain.java",
        "lineNumber" : 74,
        "nativeMethod" : false,
        "className" : "worker.org.gradle.process.internal.worker.GradleWorkerMain"
    } ],
    "lockedMonitors" : [ ],
    "lockedSynchronizers" : [ ]
}, {
    "threadName" : "Reference Handler",
    "threadId" : 2,
    "blockedTime" : -1,
    "blockedCount" : 12,
    "waitedTime" : -1,
    "waitedCount" : 0,
    "lockOwnerId" : -1,
    "daemon" : true,
    "inNative" : false,
    "suspended" : false,
    "threadState" : "RUNNABLE",
    "priority" : 10,
    "stackTrace" : [ {
        "moduleName" : "java.base",
        "moduleVersion" : "17.0.8.1",
        "methodName" : "waitForReferencePendingList",

```

```

    "fileName" : "Reference.java",
    "lineNumber" : -2,
    "nativeMethod" : true,
    "className" : "java.lang.ref.Reference"
  }, {
    "moduleName" : "java.base",
    "moduleVersion" : "17.0.8.1",
    "methodName" : "processPendingReferences",
    "fileName" : "Reference.java",
    "lineNumber" : 253,
    "nativeMethod" : false,
    "className" : "java.lang.ref.Reference"
  }, {
    "moduleName" : "java.base",
    "moduleVersion" : "17.0.8.1",
    "methodName" : "run",
    "fileName" : "Reference.java",
    "lineNumber" : 215,
    "nativeMethod" : false,
    "className" : "java.lang.ref.Reference$ReferenceHandler"
  } ],
  "lockedMonitors" : [ ],
  "lockedSynchronizers" : [ ]
}, {
  "threadName" : "Finalizer",
  "threadId" : 3,
  "blockedTime" : -1,
  "blockedCount" : 4,
  "waitedTime" : -1,
  "waitedCount" : 5,
  "lockName" : "java.lang.ref.ReferenceQueue$Lock@434d5eb9",
  "lockOwnerId" : -1,
  "daemon" : true,
  "inNative" : false,
  "suspended" : false,
  "threadState" : "WAITING",
  "priority" : 8,
  "stackTrace" : [ {
    "moduleName" : "java.base",
    "moduleVersion" : "17.0.8.1",
    "methodName" : "wait",
    "fileName" : "Object.java",
    "lineNumber" : -2,
    "nativeMethod" : true,
    "className" : "java.lang.Object"
  } ], {
    "moduleName" : "java.base",
    "moduleVersion" : "17.0.8.1",
    "methodName" : "remove",
    "fileName" : "ReferenceQueue.java",
    "lineNumber" : 155,

```

```

    "nativeMethod" : false,
    "className" : "java.lang.ref.ReferenceQueue"
  }, {
    "moduleName" : "java.base",
    "moduleVersion" : "17.0.8.1",
    "methodName" : "remove",
    "fileName" : "ReferenceQueue.java",
    "lineNumber" : 176,
    "nativeMethod" : false,
    "className" : "java.lang.ref.ReferenceQueue"
  }, {
    "moduleName" : "java.base",
    "moduleVersion" : "17.0.8.1",
    "methodName" : "run",
    "fileName" : "Finalizer.java",
    "lineNumber" : 172,
    "nativeMethod" : false,
    "className" : "java.lang.ref.Finalizer$FinalizerThread"
  } ],
  "lockedMonitors" : [ ],
  "lockedSynchronizers" : [ ],
  "lockInfo" : {
    "className" : "java.lang.ref.ReferenceQueue$Lock",
    "identityHashCode" : 1129143993
  }
} ]
}

```

### 25.1.1. Response Structure

The response contains details of the JVM's threads. The following table describes the structure of the response:

| Path                                | Type    | Description                                                                                             |
|-------------------------------------|---------|---------------------------------------------------------------------------------------------------------|
| <code>threads</code>                | Array   | JVM's threads.                                                                                          |
| <code>threads[].blockedCount</code> | Number  | Total number of times that the thread has been blocked.                                                 |
| <code>threads[].blockedTime</code>  | Number  | Time in milliseconds that the thread has spent blocked. -1 if thread contention monitoring is disabled. |
| <code>threads[].daemon</code>       | Boolean | Whether the thread is a daemon thread. Only available on Java 9 or later.                               |
| <code>threads[].inNative</code>     | Boolean | Whether the thread is executing native code.                                                            |

| Path                                                          | Type   | Description                                                                                                                                  |
|---------------------------------------------------------------|--------|----------------------------------------------------------------------------------------------------------------------------------------------|
| <code>threads[].lockName</code>                               | String | Description of the object on which the thread is blocked, if any.                                                                            |
| <code>threads[].lockInfo</code>                               | Object | Object for which the thread is blocked waiting.                                                                                              |
| <code>threads[].lockInfo.className</code>                     | String | Fully qualified class name of the lock object.                                                                                               |
| <code>threads[].lockInfo.identityHashCode</code>              | Number | Identity hash code of the lock object.                                                                                                       |
| <code>threads[].lockedMonitors</code>                         | Array  | Monitors locked by this thread, if any                                                                                                       |
| <code>threads[].lockedMonitors[].className</code>             | String | Class name of the lock object.                                                                                                               |
| <code>threads[].lockedMonitors[].identityHashCode</code>      | Number | Identity hash code of the lock object.                                                                                                       |
| <code>threads[].lockedMonitors[].lockedStackDepth</code>      | Number | Stack depth where the monitor was locked.                                                                                                    |
| <code>threads[].lockedMonitors[].lockedStackFrame</code>      | Object | Stack frame that locked the monitor.                                                                                                         |
| <code>threads[].lockedSynchronizers</code>                    | Array  | Synchronizers locked by this thread.                                                                                                         |
| <code>threads[].lockedSynchronizers[].className</code>        | String | Class name of the locked synchronizer.                                                                                                       |
| <code>threads[].lockedSynchronizers[].identityHashCode</code> | Number | Identity hash code of the locked synchronizer.                                                                                               |
| <code>threads[].lockOwnerId</code>                            | Number | ID of the thread that owns the object on which the thread is blocked. <code>-1</code> if the thread is not blocked.                          |
| <code>threads[].lockOwnerName</code>                          | String | Name of the thread that owns the object on which the thread is blocked, if any.                                                              |
| <code>threads[].priority</code>                               | Number | Priority of the thread. Only available on Java 9 or later.                                                                                   |
| <code>threads[].stackTrace</code>                             | Array  | Stack trace of the thread.                                                                                                                   |
| <code>threads[].stackTrace[].classLoaderName</code>           | String | Name of the class loader of the class that contains the execution point identified by this entry, if any. Only available on Java 9 or later. |

| Path                                              | Type    | Description                                                                                                                  |
|---------------------------------------------------|---------|------------------------------------------------------------------------------------------------------------------------------|
| <code>threads[].stackTrace[].className</code>     | String  | Name of the class that contains the execution point identified by this entry.                                                |
| <code>threads[].stackTrace[].fileName</code>      | String  | Name of the source file that contains the execution point identified by this entry, if any.                                  |
| <code>threads[].stackTrace[].lineNumber</code>    | Number  | Line number of the execution point identified by this entry. Negative if unknown.                                            |
| <code>threads[].stackTrace[].methodName</code>    | String  | Name of the method.                                                                                                          |
| <code>threads[].stackTrace[].moduleName</code>    | String  | Name of the module that contains the execution point identified by this entry, if any. Only available on Java 9 or later.    |
| <code>threads[].stackTrace[].moduleVersion</code> | String  | Version of the module that contains the execution point identified by this entry, if any. Only available on Java 9 or later. |
| <code>threads[].stackTrace[].nativeMethod</code>  | Boolean | Whether the execution point is a native method.                                                                              |
| <code>threads[].suspended</code>                  | Boolean | Whether the thread is suspended.                                                                                             |
| <code>threads[].threadId</code>                   | Number  | ID of the thread.                                                                                                            |
| <code>threads[].threadName</code>                 | String  | Name of the thread.                                                                                                          |
| <code>threads[].threadState</code>                | String  | State of the thread (NEW, RUNNABLE, BLOCKED, WAITING, TIMED_WAITING, TERMINATED).                                            |
| <code>threads[].waitedCount</code>                | Number  | Total number of times that the thread has waited for notification.                                                           |
| <code>threads[].waitedTime</code>                 | Number  | Time in milliseconds that the thread has spent waiting. -1 if thread contention monitoring is disabled                       |

## 25.2. Retrieving the Thread Dump as Text

To retrieve the thread dump as text, make a `GET` request to `/actuator/threaddump` that accepts `text/plain`, as shown in the following curl-based example:

```
$ curl 'http://localhost:8080/actuator/threaddump' -i -X GET \
-H 'Accept: text/plain'
```

The resulting response is similar to the following:

```
HTTP/1.1 200 OK
Content-Type: text/plain;charset=UTF-8
Content-Length: 75199

2023-09-21 00:39:01
Full thread dump OpenJDK 64-Bit Server VM (17.0.8.1+1-LTS mixed mode, sharing):

"Test worker" - Thread t@1
  java.lang.Thread.State: RUNNABLE
    at java.management@17.0.8.1/sun.management.ThreadImpl.dumpThreads0(Native Method)
    at
  java.management@17.0.8.1/sun.management.ThreadImpl.dumpAllThreads(ThreadImpl.java:521)
    at
  java.management@17.0.8.1/sun.management.ThreadImpl.dumpAllThreads(ThreadImpl.java:509)
    at
  app//org.springframework.boot.actuate.management.ThreadDumpEndpoint.getFormattedThread
  Dump(ThreadDumpEndpoint.java:52)
    at
  app//org.springframework.boot.actuate.management.ThreadDumpEndpoint.textThreadDump(Thr
  eadDumpEndpoint.java:48)
    at java.base@17.0.8.1/jdk.internal.reflect.NativeMethodAccessorImpl.invoke0(Native
  Method)
    at
  java.base@17.0.8.1/jdk.internal.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAc
  cessorImpl.java:77)
    at
  java.base@17.0.8.1/jdk.internal.reflect.DelegatingMethodAccessorImpl.invoke(Delegating
  MethodAccessorImpl.java:43)
    at java.base@17.0.8.1/java.lang.reflect.Method.invoke(Method.java:568)
    at
  app//org.springframework.util.ReflectionUtils.invokeMethod(ReflectionUtils.java:281)
    at
  app//org.springframework.boot.actuate.endpoint.invoke.reflect.ReflectiveOperationInvok
  er.invoke(ReflectiveOperationInvoker.java:74)
    at
  app//org.springframework.boot.actuate.endpoint.annotation.AbstractDiscoveredOperation.
  invoke(AbstractDiscoveredOperation.java:60)
    at
  app//org.springframework.boot.actuate.endpoint.web.servlet.AbstractWebMvcEndpointHandl
  erMapping$ServletWebOperationAdapter.handle(AbstractWebMvcEndpointHandlerMapping.java:
  321)
    at
  app//org.springframework.boot.actuate.endpoint.web.servlet.AbstractWebMvcEndpointHandl
  erMapping$OperationHandler.handle(AbstractWebMvcEndpointHandlerMapping.java:428)
```

```

    at jdk.internal.reflect.GeneratedMethodAccessor35.invoke(Unknown Source)
    at
java.base@17.0.8.1/jdk.internal.reflect.DelegatingMethodAccessorImpl.invoke(Delegating
MethodAccessorImpl.java:43)
    at java.base@17.0.8.1/java.lang.reflect.Method.invoke(Method.java:568)
    at
app//org.springframework.web.method.support.InvocableHandlerMethod.doInvoke(InvocableH
andlerMethod.java:205)
    at
app//org.springframework.web.method.support.InvocableHandlerMethod.invokeForRequest(In
vocableHandlerMethod.java:150)
    at
app//org.springframework.web.servlet.mvc.method.annotation.ServletInvocableHandlerMeth
od.invokeAndHandle(ServletInvocableHandlerMethod.java:118)
    at
app//org.springframework.web.servlet.mvc.method.annotation.RequestMappingHandlerAdapte
r.invokeHandlerMethod(RequestMappingHandlerAdapter.java:884)
    at
app//org.springframework.web.servlet.mvc.method.annotation.RequestMappingHandlerAdapte
r.handleInternal(RequestMappingHandlerAdapter.java:797)
    at
app//org.springframework.web.servlet.mvc.method.AbstractHandlerMethodAdapter.handle(Ab
stractHandlerMethodAdapter.java:87)
    at
app//org.springframework.web.servlet.DispatcherServlet.doDispatch(DispatcherServlet.ja
va:1081)
    at
app//org.springframework.web.servlet.DispatcherServlet.doService(DispatcherServlet.jav
a:974)
    at
app//org.springframework.web.servlet.FrameworkServlet.processRequest(FrameworkServlet.
java:1011)
    at
app//org.springframework.web.servlet.FrameworkServlet.doGet(FrameworkServlet.java:903)
    at app//jakarta.servlet.http.HttpServlet.service(HttpServlet.java:527)
    at
app//org.springframework.web.servlet.FrameworkServlet.service(FrameworkServlet.java:88
5)
    at
app//org.springframework.test.web.servlet.TestDispatcherServlet.service(TestDispatcher
Servlet.java:72)
    at app//jakarta.servlet.http.HttpServlet.service(HttpServlet.java:614)
    at
app//org.springframework.mock.web.MockFilterChain$ServletFilterProxy.doFilter(MockFilt
erChain.java:165)
    at
app//org.springframework.mock.web.MockFilterChain.doFilter(MockFilterChain.java:132)
    at app//org.springframework.test.web.servlet.MockMvc.perform(MockMvc.java:201)
    at
app//org.springframework.boot.actuate.autoconfigure.endpoint.web.documentation.ThreadD
umpEndpointDocumentationTests.textThreadDump(ThreadDumpEndpointDocumentationTests.java

```

```

:182)
    at java.base@17.0.8.1/jdk.internal.reflect.NativeMethodAccessorImpl.invoke0(Native
Method)
    at
java.base@17.0.8.1/jdk.internal.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAc
cessorImpl.java:77)
    at
java.base@17.0.8.1/jdk.internal.reflect.DelegatingMethodAccessorImpl.invoke(Delegating
MethodAccessorImpl.java:43)
    at java.base@17.0.8.1/java.lang.reflect.Method.invoke(Method.java:568)
    at
app//org.junit.platform.commons.util.ReflectionUtils.invokeMethod(ReflectionUtils.java
:727)
    at
app//org.junit.jupiter.engine.execution.MethodInvocation.proceed(MethodInvocation.java
:60)
    at
app//org.junit.jupiter.engine.execution.InvocationInterceptorChain$ValidatingInvocatio
n.proceed(InvocationInterceptorChain.java:131)
    at
app//org.junit.jupiter.engine.extension.TimeoutExtension.intercept(TimeoutExtension.ja
va:156)
    at
app//org.junit.jupiter.engine.extension.TimeoutExtension.interceptTestableMethod(Timeo
utExtension.java:147)
    at
app//org.junit.jupiter.engine.extension.TimeoutExtension.interceptTestMethod(TimeoutEx
tension.java:86)
    at
app//org.junit.jupiter.engine.descriptor.TestMethodTestDescriptor$$Lambda$196/0x00007f
f0e8136d18.apply(Unknown Source)
    at
app//org.junit.jupiter.engine.execution.InterceptingExecutableInvoker$ReflectiveInterc
eptorCall.lambda$ofVoidMethod$0(InterceptingExecutableInvoker.java:103)
    at
app//org.junit.jupiter.engine.execution.InterceptingExecutableInvoker$ReflectiveInterc
eptorCall$$Lambda$197/0x00007ff0e8137138.apply(Unknown Source)
    at
app//org.junit.jupiter.engine.execution.InterceptingExecutableInvoker.lambda$invoke$0(
InterceptingExecutableInvoker.java:93)
    at
app//org.junit.jupiter.engine.execution.InterceptingExecutableInvoker$$Lambda$464/0x00
007ff0e81df500.apply(Unknown Source)
    at
app//org.junit.jupiter.engine.execution.InvocationInterceptorChain$InterceptedInvocati
on.proceed(InvocationInterceptorChain.java:106)
    at
app//org.junit.jupiter.engine.execution.InvocationInterceptorChain.proceed(Invocatio
nInterceptorChain.java:64)
    at
app//org.junit.jupiter.engine.execution.InvocationInterceptorChain.chainAndInvoke(Invo

```

```

cationInterceptorChain.java:45)
    at
app//org.junit.jupiter.engine.execution.InvocationInterceptorChain.invoke(InvocationIn
terceptorChain.java:37)
    at
app//org.junit.jupiter.engine.execution.InterceptingExecutableInvoker.invoke(Intercept
ingExecutableInvoker.java:92)
    at
app//org.junit.jupiter.engine.execution.InterceptingExecutableInvoker.invoke(Intercept
ingExecutableInvoker.java:86)
    at
app//org.junit.jupiter.engine.descriptor.TestMethodTestDescriptor.lambda$invokeTestMet
hod$7(TestMethodTestDescriptor.java:217)
    at
app//org.junit.jupiter.engine.descriptor.TestMethodTestDescriptor$$Lambda$949/0x00007f
f0e8599478.execute(Unknown Source)
    at
app//org.junit.platform.engine.support.hierarchical.ThrowableCollector.execute(Throwab
leCollector.java:73)
    at
app//org.junit.jupiter.engine.descriptor.TestMethodTestDescriptor.invokeTestMethod(Tes
tMethodTestDescriptor.java:213)
    at
app//org.junit.jupiter.engine.descriptor.TestMethodTestDescriptor.execute(TestMethodTe
stDescriptor.java:138)
    at
app//org.junit.jupiter.engine.descriptor.TestMethodTestDescriptor.execute(TestMethodTe
stDescriptor.java:68)
    at
app//org.junit.platform.engine.support.hierarchical.NodeTestTask.lambda$executeRecursi
vely$6(NodeTestTask.java:151)
    at
app//org.junit.platform.engine.support.hierarchical.NodeTestTask$$Lambda$299/0x00007ff
0e81547c0.execute(Unknown Source)
    at
app//org.junit.platform.engine.support.hierarchical.ThrowableCollector.execute(Throwab
leCollector.java:73)
    at
app//org.junit.platform.engine.support.hierarchical.NodeTestTask.lambda$executeRecursi
vely$8(NodeTestTask.java:141)
    at
app//org.junit.platform.engine.support.hierarchical.NodeTestTask$$Lambda$298/0x00007ff
0e8154598.invoke(Unknown Source)
    at app//org.junit.platform.engine.support.hierarchical.Node.around(Node.java:137)
    at
app//org.junit.platform.engine.support.hierarchical.NodeTestTask.lambda$executeRecursi
vely$9(NodeTestTask.java:139)
    at
app//org.junit.platform.engine.support.hierarchical.NodeTestTask$$Lambda$297/0x00007ff
0e8154170.execute(Unknown Source)
    at

```

```

app//org.junit.platform.engine.support.hierarchical.ThrowableCollector.execute(Throwab
leCollector.java:73)
    at
app//org.junit.platform.engine.support.hierarchical.NodeTestTask.executeRecursively(No
deTestTask.java:138)
    at
app//org.junit.platform.engine.support.hierarchical.NodeTestTask.execute(NodeTestTask.
java:95)
    at
app//org.junit.platform.engine.support.hierarchical.SameThreadHierarchicalTestExecuto
rService$$Lambda$303/0x00007ff0e81552d8.accept(Unknown Source)
    at java.base@17.0.8.1/java.util.ArrayList.forEach(ArrayList.java:1511)
    at
app//org.junit.platform.engine.support.hierarchical.SameThreadHierarchicalTestExecuto
rService.invokeAll(SameThreadHierarchicalTestExecutorService.java:41)
    at
app//org.junit.platform.engine.support.hierarchical.NodeTestTask.lambda$executeRecursi
vely$6(NodeTestTask.java:155)
    at
app//org.junit.platform.engine.support.hierarchical.NodeTestTask$$Lambda$299/0x00007ff
0e81547c0.execute(Unknown Source)
    at
app//org.junit.platform.engine.support.hierarchical.ThrowableCollector.execute(Throwab
leCollector.java:73)
    at
app//org.junit.platform.engine.support.hierarchical.NodeTestTask.lambda$executeRecursi
vely$8(NodeTestTask.java:141)
    at
app//org.junit.platform.engine.support.hierarchical.NodeTestTask$$Lambda$298/0x00007ff
0e8154598.invoke(Unknown Source)
    at app//org.junit.platform.engine.support.hierarchical.Node.around(Node.java:137)
    at
app//org.junit.platform.engine.support.hierarchical.NodeTestTask.lambda$executeRecursi
vely$9(NodeTestTask.java:139)
    at
app//org.junit.platform.engine.support.hierarchical.NodeTestTask$$Lambda$297/0x00007ff
0e8154170.execute(Unknown Source)
    at
app//org.junit.platform.engine.support.hierarchical.ThrowableCollector.execute(Throwab
leCollector.java:73)
    at
app//org.junit.platform.engine.support.hierarchical.NodeTestTask.executeRecursively(No
deTestTask.java:138)
    at
app//org.junit.platform.engine.support.hierarchical.NodeTestTask.execute(NodeTestTask.
java:95)
    at
app//org.junit.platform.engine.support.hierarchical.SameThreadHierarchicalTestExecuto
rService$$Lambda$303/0x00007ff0e81552d8.accept(Unknown Source)
    at java.base@17.0.8.1/java.util.ArrayList.forEach(ArrayList.java:1511)
    at

```

```
app//org.junit.platform.engine.support.hierarchical.SameThreadHierarchicalTestExecutor
Service.invokeAll(SameThreadHierarchicalTestExecutorService.java:41)
    at
app//org.junit.platform.engine.support.hierarchical.NodeTestTask.lambda$executeRecursi
vely$6(NodeTestTask.java:155)
    at
app//org.junit.platform.engine.support.hierarchical.NodeTestTask$$Lambda$299/0x00007ff
0e81547c0.execute(Unknown Source)
    at
app//org.junit.platform.engine.support.hierarchical.ThrowableCollector.execute(Throwab
leCollector.java:73)
    at
app//org.junit.platform.engine.support.hierarchical.NodeTestTask.lambda$executeRecursi
vely$8(NodeTestTask.java:141)
    at
app//org.junit.platform.engine.support.hierarchical.NodeTestTask$$Lambda$298/0x00007ff
0e8154598.invoke(Unknown Source)
        at app//org.junit.platform.engine.support.hierarchical.Node.around(Node.java:137)
    at
app//org.junit.platform.engine.support.hierarchical.NodeTestTask.lambda$executeRecursi
vely$9(NodeTestTask.java:139)
    at
app//org.junit.platform.engine.support.hierarchical.NodeTestTask$$Lambda$297/0x00007ff
0e8154170.execute(Unknown Source)
    at
app//org.junit.platform.engine.support.hierarchical.ThrowableCollector.execute(Throwab
leCollector.java:73)
    at
app//org.junit.platform.engine.support.hierarchical.NodeTestTask.executeRecursively(N
odeTestTask.java:138)
    at
app//org.junit.platform.engine.support.hierarchical.NodeTestTask.execute(NodeTestTask.
java:95)
    at
app//org.junit.platform.engine.support.hierarchical.SameThreadHierarchicalTestExecuto
rService.submit(SameThreadHierarchicalTestExecutorService.java:35)
    at
app//org.junit.platform.engine.support.hierarchical.HierarchicalTestExecutor.execute(H
ierarchicalTestExecutor.java:57)
    at
app//org.junit.platform.engine.support.hierarchical.HierarchicalTestEngine.execute(Hie
rarchicalTestEngine.java:54)
    at
app//org.junit.platform.launcher.core.EngineExecutionOrchestrator.execute(EngineExecut
ionOrchestrator.java:147)
    at
app//org.junit.platform.launcher.core.EngineExecutionOrchestrator.execute(EngineExecut
ionOrchestrator.java:127)
    at
app//org.junit.platform.launcher.core.EngineExecutionOrchestrator.execute(EngineExecut
ionOrchestrator.java:90)
```

```

    at
app//org.junit.platform.launcher.core.EngineExecutionOrchestrator.lambda$execute$0(Eng
ineExecutionOrchestrator.java:55)
    at
app//org.junit.platform.launcher.core.EngineExecutionOrchestrator$$Lambda$251/0x00007f
f0e8141eb8.accept(Unknown Source)
    at
app//org.junit.platform.launcher.core.EngineExecutionOrchestrator.withInterceptedStrea
ms(EngineExecutionOrchestrator.java:102)
    at
app//org.junit.platform.launcher.core.EngineExecutionOrchestrator.execute(EngineExecut
ionOrchestrator.java:54)
    at
app//org.junit.platform.launcher.core.DefaultLauncher.execute(DefaultLauncher.java:114
)
    at
app//org.junit.platform.launcher.core.DefaultLauncher.execute(DefaultLauncher.java:86)
    at
app//org.junit.platform.launcher.core.DefaultLauncherSession$DelegatingLauncher.execut
e(DefaultLauncherSession.java:86)
    at
app//org.junit.platform.launcher.core.SessionPerRequestLauncher.execute(SessionPerRequ
estLauncher.java:53)
    at
org.gradle.api.internal.tasks.testing.junitplatform.JUnitPlatformTestClassProcessor$Co
llectAllTestClassesExecutor.processAllTestClasses(JUnitPlatformTestClassProcessor.java
:99)
    at
org.gradle.api.internal.tasks.testing.junitplatform.JUnitPlatformTestClassProcessor$Co
llectAllTestClassesExecutor.access$000(JUnitPlatformTestClassProcessor.java:79)
    at
org.gradle.api.internal.tasks.testing.junitplatform.JUnitPlatformTestClassProcessor.st
op(JUnitPlatformTestClassProcessor.java:75)
    at
org.gradle.api.internal.tasks.testing.SuiteTestClassProcessor.stop(SuiteTestClassProce
ssor.java:62)
    at java.base@17.0.8.1/jdk.internal.reflect.NativeMethodAccessorImpl.invoke0(Native
Method)
    at
java.base@17.0.8.1/jdk.internal.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAc
cessorImpl.java:77)
    at
java.base@17.0.8.1/jdk.internal.reflect.DelegatingMethodAccessorImpl.invoke(Delegating
MethodAccessorImpl.java:43)
    at java.base@17.0.8.1/java.lang.reflect.Method.invoke(Method.java:568)
    at
org.gradle.internal.dispatch.ReflectionDispatch.dispatch(ReflectionDispatch.java:36)
    at
org.gradle.internal.dispatch.ReflectionDispatch.dispatch(ReflectionDispatch.java:24)
    at
org.gradle.internal.dispatch.ContextClassLoaderDispatch.dispatch(ContextClassLoaderDis

```

```

patch.java:33)
    at
org.gradle.internal.dispatch.ProxyDispatchAdapter$DispatchingInvocationHandler.invoke(
ProxyDispatchAdapter.java:94)
    at jdk.proxy1/jdk.proxy1.$Proxy2.stop(Unknown Source)
    at
org.gradle.api.internal.tasks.testing.worker.TestWorker$3.run(TestWorker.java:193)
    at
org.gradle.api.internal.tasks.testing.worker.TestWorker.executeAndMaintainThreadName(T
estWorker.java:129)
    at
org.gradle.api.internal.tasks.testing.worker.TestWorker.execute(TestWorker.java:100)
    at
org.gradle.api.internal.tasks.testing.worker.TestWorker.execute(TestWorker.java:60)
    at
org.gradle.process.internal.worker.child.ActionExecutionWorker.execute(ActionExecution
Worker.java:56)
    at
org.gradle.process.internal.worker.child.SystemApplicationClassLoaderWorker.call(Syste
mApplicationClassLoaderWorker.java:113)
    at
org.gradle.process.internal.worker.child.SystemApplicationClassLoaderWorker.call(Syste
mApplicationClassLoaderWorker.java:65)
    at
app//worker.org.gradle.process.internal.worker.GradleWorkerMain.run(GradleWorkerMain.j
ava:69)
    at
app//worker.org.gradle.process.internal.worker.GradleWorkerMain.main(GradleWorkerMain.
java:74)

```

Locked ownable synchronizers:

- None

"Reference Handler" - Thread t@2

java.lang.Thread.State: RUNNABLE

at java.base@17.0.8.1/java.lang.ref.Reference.waitForReferencePendingList(Native  
Method)

at

java.base@17.0.8.1/java.lang.ref.Reference.processPendingReferences(Reference.java:253  
)

at

java.base@17.0.8.1/java.lang.ref.Reference\$ReferenceHandler.run(Reference.java:215)

Locked ownable synchronizers:

- None

"Finalizer" - Thread t@3

java.lang.Thread.State: WAITING

at java.base@17.0.8.1/java.lang.Object.wait(Native Method)

- waiting on <434d5eb9> (a java.lang.ref.ReferenceQueue\$Lock)

at java.base@17.0.8.1/java.lang.ref.ReferenceQueue.remove(ReferenceQueue.java:155)

```

    at java.base@17.0.8.1/java.lang.ref.ReferenceQueue.remove(ReferenceQueue.java:176)
    at
java.base@17.0.8.1/java.lang.ref.Finalizer$FinalizerThread.run(Finalizer.java:172)

    Locked ownable synchronizers:
      - None

"Signal Dispatcher" - Thread t@4
  java.lang.Thread.State: RUNNABLE

    Locked ownable synchronizers:
      - None

"Common-Cleaner" - Thread t@23
  java.lang.Thread.State: TIMED_WAITING
    at java.base@17.0.8.1/java.lang.Object.wait(Native Method)
    - waiting on <5914a346> (a java.lang.ref.ReferenceQueue$Lock)
    at java.base@17.0.8.1/java.lang.ref.ReferenceQueue.remove(ReferenceQueue.java:155)
    at java.base@17.0.8.1/jdk.internal.ref.CleanerImpl.run(CleanerImpl.java:140)
    at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)
    at
java.base@17.0.8.1/jdk.internal.misc.InnocuousThread.run(InnocuousThread.java:162)

    Locked ownable synchronizers:
      - None

"Notification Thread" - Thread t@24
  java.lang.Thread.State: RUNNABLE

    Locked ownable synchronizers:
      - None

"/127.0.0.1:49196 to /127.0.0.1:37527 workers" - Thread t@26
  java.lang.Thread.State: WAITING
    at java.base@17.0.8.1/jdk.internal.misc.Unsafe.park(Native Method)
    - parking to wait for <5242079e> (a
java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionObject)
    at
java.base@17.0.8.1/java.util.concurrent.locks.LockSupport.park(LockSupport.java:341)
    at
java.base@17.0.8.1/java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionNode
.block(AbstractQueuedSynchronizer.java:506)
    at
java.base@17.0.8.1/java.util.concurrent.ForkJoinPool.unmanagedBlock(ForkJoinPool.java:
3465)
    at
java.base@17.0.8.1/java.util.concurrent.ForkJoinPool.managedBlock(ForkJoinPool.java:34
36)
    at
java.base@17.0.8.1/java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionObje
ct.await(AbstractQueuedSynchronizer.java:1623)

```

```

    at
org.gradle.internal.remote.internal.hub.queue.EndPointQueue.take(EndPointQueue.java:49
)
    at
org.gradle.internal.remote.internal.hub.MessageHub$Handler.run(MessageHub.java:403)
    at
org.gradle.internal.concurrent.ExecutorPolicy$CatchAndRecordFailures.onExecute(Executo
rPolicy.java:64)
    at
org.gradle.internal.concurrent.ManagedExecutorImpl$1.run(ManagedExecutorImpl.java:49)
    at
java.base@17.0.8.1/java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecuto
r.java:1136)
    at
java.base@17.0.8.1/java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecut
or.java:635)
    at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)

Locked ownable synchronizers:
- Locked <1cbb87f3> (a java.util.concurrent.ThreadPoolExecutor$Worker)

"/127.0.0.1:49196 to /127.0.0.1:37527 workers Thread 2" - Thread t@27
java.lang.Thread.State: WAITING
    at java.base@17.0.8.1/jdk.internal.misc.Unsafe.park(Native Method)
    - parking to wait for <3109f8ae> (a
java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionObject)
    at
java.base@17.0.8.1/java.util.concurrent.locks.LockSupport.park(LockSupport.java:341)
    at
java.base@17.0.8.1/java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionNode
.block(AbstractQueuedSynchronizer.java:506)
    at
java.base@17.0.8.1/java.util.concurrent.ForkJoinPool.unmanagedBlock(ForkJoinPool.java:
3465)
    at
java.base@17.0.8.1/java.util.concurrent.ForkJoinPool.managedBlock(ForkJoinPool.java:34
36)
    at
java.base@17.0.8.1/java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionObje
ct.await(AbstractQueuedSynchronizer.java:1623)
    at
org.gradle.internal.remote.internal.hub.queue.EndPointQueue.take(EndPointQueue.java:49
)
    at
org.gradle.internal.remote.internal.hub.MessageHub$ConnectionDispatch.run(MessageHub.j
ava:322)
    at
org.gradle.internal.concurrent.ExecutorPolicy$CatchAndRecordFailures.onExecute(Executo
rPolicy.java:64)
    at
org.gradle.internal.concurrent.ManagedExecutorImpl$1.run(ManagedExecutorImpl.java:49)

```

```

    at
java.base@17.0.8.1/java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecuto
r.java:1136)
    at
java.base@17.0.8.1/java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecut
or.java:635)
    at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)

Locked ownable synchronizers:
- Locked <4738a206> (a java.util.concurrent.ThreadPoolExecutor$Worker)

"/127.0.0.1:49196 to /127.0.0.1:37527 workers Thread 3" - Thread t@28
java.lang.Thread.State: RUNNABLE
    at java.base@17.0.8.1/sun.nio.ch.EPoll.wait(Native Method)
    at
java.base@17.0.8.1/sun.nio.ch.EPollSelectorImpl.doSelect(EPollSelectorImpl.java:118)
    at
java.base@17.0.8.1/sun.nio.ch.SelectorImpl.lockAndDoSelect(SelectorImpl.java:129)
    - locked <1818c0a0> (a sun.nio.ch.Util$2)
    - locked <2656db92> (a sun.nio.ch.EPollSelectorImpl)
    at java.base@17.0.8.1/sun.nio.ch.SelectorImpl.select(SelectorImpl.java:146)
    at
org.gradle.internal.remote.internal.inet.SocketConnection$SocketInputStream.read(Socket
Connection.java:185)
    at com.esotericsoftware.kryo.io.Input.fill(Input.java:146)
    at com.esotericsoftware.kryo.io.Input.require(Input.java:178)
    at com.esotericsoftware.kryo.io.Input.readByte(Input.java:295)
    at
org.gradle.internal.serialize.kryo.KryoBackedDecoder.readByte(KryoBackedDecoder.java:8
8)
    at
org.gradle.internal.remote.internal.hub.InterHubMessageSerializer$MessageReader.read(I
nterHubMessageSerializer.java:64)
    at
org.gradle.internal.remote.internal.hub.InterHubMessageSerializer$MessageReader.read(I
nterHubMessageSerializer.java:52)
    at
org.gradle.internal.remote.internal.inet.SocketConnection.receive(SocketConnection.jav
a:81)
    at
org.gradle.internal.remote.internal.hub.MessageHub$ConnectionReceive.run(MessageHub.ja
va:270)
    at
org.gradle.internal.concurrent.ExecutorPolicy$CatchAndRecordFailures.onExecute(Executo
rPolicy.java:64)
    at
org.gradle.internal.concurrent.ManagedExecutorImpl$1.run(ManagedExecutorImpl.java:49)
    at
java.base@17.0.8.1/java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecuto
r.java:1136)
    at

```

```

java.base@17.0.8.1/java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecut
or.java:635)
    at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)

Locked ownable synchronizers:
  - Locked <18d87d80> (a java.util.concurrent.ThreadPoolExecutor$Worker)

"HikariPool-1 housekeeper" - Thread t@39
  java.lang.Thread.State: TIMED_WAITING
    at java.base@17.0.8.1/jdk.internal.misc.Unsafe.park(Native Method)
      - parking to wait for <58649b7e> (a
java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionObject)
    at
java.base@17.0.8.1/java.util.concurrent.locks.LockSupport.parkNanos(LockSupport.java:2
52)
    at
java.base@17.0.8.1/java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionObje
ct.awaitNanos(AbstractQueuedSynchronizer.java:1672)
    at
java.base@17.0.8.1/java.util.concurrent.ScheduledThreadPoolExecutor$DelayedWorkQueue.t
ake(ScheduledThreadPoolExecutor.java:1182)
    at
java.base@17.0.8.1/java.util.concurrent.ScheduledThreadPoolExecutor$DelayedWorkQueue.t
ake(ScheduledThreadPoolExecutor.java:899)
    at
java.base@17.0.8.1/java.util.concurrent.ThreadPoolExecutor.getTask(ThreadPoolExecuto
r.java:1062)
    at
java.base@17.0.8.1/java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecuto
r.java:1122)
    at
java.base@17.0.8.1/java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecut
or.java:635)
    at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)

Locked ownable synchronizers:
  - None

"boundedElastic-evictor-1" - Thread t@50
  java.lang.Thread.State: TIMED_WAITING
    at java.base@17.0.8.1/jdk.internal.misc.Unsafe.park(Native Method)
      - parking to wait for <6a75d12> (a
java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionObject)
    at
java.base@17.0.8.1/java.util.concurrent.locks.LockSupport.parkNanos(LockSupport.java:2
52)
    at
java.base@17.0.8.1/java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionObje
ct.awaitNanos(AbstractQueuedSynchronizer.java:1672)
    at
java.base@17.0.8.1/java.util.concurrent.ScheduledThreadPoolExecutor$DelayedWorkQueue.t

```

```

ake(ScheduledThreadPoolExecutor.java:1182)
    at
java.base@17.0.8.1/java.util.concurrent.ScheduledThreadPoolExecutor$DelayedWorkQueue.t
ake(ScheduledThreadPoolExecutor.java:899)
    at
java.base@17.0.8.1/java.util.concurrent.ThreadPoolExecutor.getTask(ThreadPoolExecutor.
java:1062)
    at
java.base@17.0.8.1/java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecuto
r.java:1122)
    at
java.base@17.0.8.1/java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecut
or.java:635)
    at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)

```

Locked ownable synchronizers:  
- None

```

"reactor-http-epoll-1" - Thread t@51
java.lang.Thread.State: RUNNABLE
    at app//io.netty.channel.epoll.Native.epollWait(Native Method)
    at app//io.netty.channel.epoll.Native.epollWait(Native.java:209)
    at app//io.netty.channel.epoll.Native.epollWait(Native.java:202)
    at
app//io.netty.channel.epoll.EpollEventLoop.epollWaitNoTimerChange(EpollEventLoop.java:
316)
    at app//io.netty.channel.epoll.EpollEventLoop.run(EpollEventLoop.java:373)
    at
app//io.netty.util.concurrent.SingleThreadEventExecutor$4.run(SingleThreadEventExecuto
r.java:997)
    at app//io.netty.util.internal.ThreadExecutorMap$2.run(ThreadExecutorMap.java:74)
    at
app//io.netty.util.concurrent.FastThreadLocalRunnable.run(FastThreadLocalRunnable.java
:30)
    at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)

```

Locked ownable synchronizers:  
- None

```

"server" - Thread t@52
java.lang.Thread.State: WAITING
    at java.base@17.0.8.1/jdk.internal.misc.Unsafe.park(Native Method)
    - parking to wait for <30082a5e> (a java.util.concurrent.CountDownLatch$Sync)
    at
java.base@17.0.8.1/java.util.concurrent.locks.LockSupport.park(LockSupport.java:211)
    at
java.base@17.0.8.1/java.util.concurrent.locks.AbstractQueuedSynchronizer.acquire(Abstr
actQueuedSynchronizer.java:715)
    at
java.base@17.0.8.1/java.util.concurrent.locks.AbstractQueuedSynchronizer.acquireShared
Interruptibly(AbstractQueuedSynchronizer.java:1047)

```

```
at
java.base@17.0.8.1/java.util.concurrent.CountDownLatch.await(CountDownLatch.java:230)
at
app//reactor.core.publisher.BlockingSingleSubscriber.blockingGet(BlockingSingleSubscriber.java:90)
at app//reactor.core.publisher.Mono.block(Mono.java:1712)
at
app//org.springframework.boot.web.embedded.netty.NettyWebServer$1.run(NettyWebServer.java:180)
```

Locked ownable synchronizers:

- None

"reactor-http-epoll-2" - Thread t@53

```
java.lang.Thread.State: RUNNABLE
at app//io.netty.channel.epoll.Native.epollWait(Native Method)
at app//io.netty.channel.epoll.Native.epollWait(Native.java:209)
at app//io.netty.channel.epoll.Native.epollWait(Native.java:202)
at
app//io.netty.channel.epoll.EpollEventLoop.epollWaitNoTimerChange(EpollEventLoop.java:316)
at app//io.netty.channel.epoll.EpollEventLoop.run(EpollEventLoop.java:373)
at
app//io.netty.util.concurrent.SingleThreadEventExecutor$4.run(SingleThreadEventExecutor.java:997)
at app//io.netty.util.internal.ThreadExecutorMap$2.run(ThreadExecutorMap.java:74)
at
app//io.netty.util.concurrent.FastThreadLocalRunnable.run(FastThreadLocalRunnable.java:30)
at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)
```

Locked ownable synchronizers:

- None

"reactor-http-epoll-3" - Thread t@54

```
java.lang.Thread.State: RUNNABLE
at app//io.netty.channel.epoll.Native.epollWait(Native Method)
at app//io.netty.channel.epoll.Native.epollWait(Native.java:209)
at app//io.netty.channel.epoll.Native.epollWait(Native.java:202)
at
app//io.netty.channel.epoll.EpollEventLoop.epollWaitNoTimerChange(EpollEventLoop.java:316)
at app//io.netty.channel.epoll.EpollEventLoop.run(EpollEventLoop.java:373)
at
app//io.netty.util.concurrent.SingleThreadEventExecutor$4.run(SingleThreadEventExecutor.java:997)
at app//io.netty.util.internal.ThreadExecutorMap$2.run(ThreadExecutorMap.java:74)
at
app//io.netty.util.concurrent.FastThreadLocalRunnable.run(FastThreadLocalRunnable.java:30)
at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)
```

Locked ownable synchronizers:

- None

"reactor-http-epoll-4" - Thread t@55

java.lang.Thread.State: RUNNABLE

at app//io.netty.channel.epoll.Native.epollWait(Native Method)

at app//io.netty.channel.epoll.Native.epollWait(Native.java:209)

at app//io.netty.channel.epoll.Native.epollWait(Native.java:202)

at

app//io.netty.channel.epoll.EpollEventLoop.epollWaitNoTimerChange(EpollEventLoop.java:316)

at app//io.netty.channel.epoll.EpollEventLoop.run(EpollEventLoop.java:373)

at

app//io.netty.util.concurrent.SingleThreadEventExecutor\$4.run(SingleThreadEventExecutor.java:997)

at app//io.netty.util.internal.ThreadExecutorMap\$2.run(ThreadExecutorMap.java:74)

at

app//io.netty.util.concurrent.FastThreadLocalRunnable.run(FastThreadLocalRunnable.java:30)

at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)

Locked ownable synchronizers:

- None

"reactor-http-epoll-5" - Thread t@56

java.lang.Thread.State: RUNNABLE

at app//io.netty.channel.epoll.Native.epollWait(Native Method)

at app//io.netty.channel.epoll.Native.epollWait(Native.java:209)

at app//io.netty.channel.epoll.Native.epollWait(Native.java:202)

at

app//io.netty.channel.epoll.EpollEventLoop.epollWaitNoTimerChange(EpollEventLoop.java:316)

at app//io.netty.channel.epoll.EpollEventLoop.run(EpollEventLoop.java:373)

at

app//io.netty.util.concurrent.SingleThreadEventExecutor\$4.run(SingleThreadEventExecutor.java:997)

at app//io.netty.util.internal.ThreadExecutorMap\$2.run(ThreadExecutorMap.java:74)

at

app//io.netty.util.concurrent.FastThreadLocalRunnable.run(FastThreadLocalRunnable.java:30)

at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)

Locked ownable synchronizers:

- None

"reactor-http-epoll-6" - Thread t@57

java.lang.Thread.State: RUNNABLE

at app//io.netty.channel.epoll.Native.epollWait(Native Method)

at app//io.netty.channel.epoll.Native.epollWait(Native.java:209)

at app//io.netty.channel.epoll.Native.epollWait(Native.java:202)

```
    at
app//io.netty.channel.epoll.EpollEventLoop.epollWaitNoTimerChange(EpollEventLoop.java:
316)
    at app//io.netty.channel.epoll.EpollEventLoop.run(EpollEventLoop.java:373)
    at
app//io.netty.util.concurrent.SingleThreadEventExecutor$4.run(SingleThreadEventExecuto
r.java:997)
    at app//io.netty.util.internal.ThreadExecutorMap$2.run(ThreadExecutorMap.java:74)
    at
app//io.netty.util.concurrent.FastThreadLocalRunnable.run(FastThreadLocalRunnable.java
:30)
    at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)
```

Locked ownable synchronizers:

- None

"reactor-http-epoll-7" - Thread t@58

java.lang.Thread.State: RUNNABLE

```
    at app//io.netty.channel.epoll.Native.epollWait(Native Method)
    at app//io.netty.channel.epoll.Native.epollWait(Native.java:209)
    at app//io.netty.channel.epoll.Native.epollWait(Native.java:202)
    at
app//io.netty.channel.epoll.EpollEventLoop.epollWaitNoTimerChange(EpollEventLoop.java:
316)
    at app//io.netty.channel.epoll.EpollEventLoop.run(EpollEventLoop.java:373)
    at
app//io.netty.util.concurrent.SingleThreadEventExecutor$4.run(SingleThreadEventExecuto
r.java:997)
    at app//io.netty.util.internal.ThreadExecutorMap$2.run(ThreadExecutorMap.java:74)
    at
app//io.netty.util.concurrent.FastThreadLocalRunnable.run(FastThreadLocalRunnable.java
:30)
    at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)
```

Locked ownable synchronizers:

- None

"reactor-http-epoll-8" - Thread t@59

java.lang.Thread.State: RUNNABLE

```
    at app//io.netty.channel.epoll.Native.epollWait(Native Method)
    at app//io.netty.channel.epoll.Native.epollWait(Native.java:209)
    at app//io.netty.channel.epoll.Native.epollWait(Native.java:202)
    at
app//io.netty.channel.epoll.EpollEventLoop.epollWaitNoTimerChange(EpollEventLoop.java:
316)
    at app//io.netty.channel.epoll.EpollEventLoop.run(EpollEventLoop.java:373)
    at
app//io.netty.util.concurrent.SingleThreadEventExecutor$4.run(SingleThreadEventExecuto
r.java:997)
    at app//io.netty.util.internal.ThreadExecutorMap$2.run(ThreadExecutorMap.java:74)
    at
```

```

app//io.netty.util.concurrent.FastThreadLocalRunnable.run(FastThreadLocalRunnable.java:30)
    at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)

Locked ownable synchronizers:
- None

"reactor-http-epoll-9" - Thread t@60
java.lang.Thread.State: RUNNABLE
    at app//io.netty.channel.epoll.Native.epollWait(Native Method)
    at app//io.netty.channel.epoll.Native.epollWait(Native.java:209)
    at app//io.netty.channel.epoll.Native.epollWait(Native.java:202)
    at
app//io.netty.channel.epoll.EpollEventLoop.epollWaitNoTimerChange(EpollEventLoop.java:316)
    at app//io.netty.channel.epoll.EpollEventLoop.run(EpollEventLoop.java:373)
    at
app//io.netty.util.concurrent.SingleThreadEventExecutor$4.run(SingleThreadEventExecutor.java:997)
    at app//io.netty.util.internal.ThreadExecutorMap$2.run(ThreadExecutorMap.java:74)
    at
app//io.netty.util.concurrent.FastThreadLocalRunnable.run(FastThreadLocalRunnable.java:30)
    at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)

Locked ownable synchronizers:
- None

"reactor-http-epoll-10" - Thread t@61
java.lang.Thread.State: RUNNABLE
    at app//io.netty.channel.epoll.Native.epollWait(Native Method)
    at app//io.netty.channel.epoll.Native.epollWait(Native.java:209)
    at app//io.netty.channel.epoll.Native.epollWait(Native.java:202)
    at
app//io.netty.channel.epoll.EpollEventLoop.epollWaitNoTimerChange(EpollEventLoop.java:316)
    at app//io.netty.channel.epoll.EpollEventLoop.run(EpollEventLoop.java:373)
    at
app//io.netty.util.concurrent.SingleThreadEventExecutor$4.run(SingleThreadEventExecutor.java:997)
    at app//io.netty.util.internal.ThreadExecutorMap$2.run(ThreadExecutorMap.java:74)
    at
app//io.netty.util.concurrent.FastThreadLocalRunnable.run(FastThreadLocalRunnable.java:30)
    at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)

Locked ownable synchronizers:
- None

"reactor-http-epoll-11" - Thread t@62
java.lang.Thread.State: RUNNABLE

```

```
at app//io.netty.channel.epoll.Native.epollWait(Native Method)
at app//io.netty.channel.epoll.Native.epollWait(Native.java:209)
at app//io.netty.channel.epoll.Native.epollWait(Native.java:202)
at
app//io.netty.channel.epoll.EpollEventLoop.epollWaitNoTimerChange(EpollEventLoop.java:
316)
at app//io.netty.channel.epoll.EpollEventLoop.run(EpollEventLoop.java:373)
at
app//io.netty.util.concurrent.SingleThreadEventExecutor$4.run(SingleThreadEventExecuto
r.java:997)
at app//io.netty.util.internal.ThreadExecutorMap$2.run(ThreadExecutorMap.java:74)
at
app//io.netty.util.concurrent.FastThreadLocalRunnable.run(FastThreadLocalRunnable.java
:30)
at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)
```

Locked ownable synchronizers:  
- None

"reactor-http-epoll-12" - Thread t@63

```
java.lang.Thread.State: RUNNABLE
at app//io.netty.channel.epoll.Native.epollWait(Native Method)
at app//io.netty.channel.epoll.Native.epollWait(Native.java:209)
at app//io.netty.channel.epoll.Native.epollWait(Native.java:202)
at
app//io.netty.channel.epoll.EpollEventLoop.epollWaitNoTimerChange(EpollEventLoop.java:
316)
at app//io.netty.channel.epoll.EpollEventLoop.run(EpollEventLoop.java:373)
at
app//io.netty.util.concurrent.SingleThreadEventExecutor$4.run(SingleThreadEventExecuto
r.java:997)
at app//io.netty.util.internal.ThreadExecutorMap$2.run(ThreadExecutorMap.java:74)
at
app//io.netty.util.concurrent.FastThreadLocalRunnable.run(FastThreadLocalRunnable.java
:30)
at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)
```

Locked ownable synchronizers:  
- None

"reactor-http-epoll-13" - Thread t@64

```
java.lang.Thread.State: RUNNABLE
at app//io.netty.channel.epoll.Native.epollWait(Native Method)
at app//io.netty.channel.epoll.Native.epollWait(Native.java:209)
at app//io.netty.channel.epoll.Native.epollWait(Native.java:202)
at
app//io.netty.channel.epoll.EpollEventLoop.epollWaitNoTimerChange(EpollEventLoop.java:
316)
at app//io.netty.channel.epoll.EpollEventLoop.run(EpollEventLoop.java:373)
at
app//io.netty.util.concurrent.SingleThreadEventExecutor$4.run(SingleThreadEventExecuto
```

```

r.java:997)
    at app//io.netty.util.internal.ThreadExecutorMap$2.run(ThreadExecutorMap.java:74)
    at
app//io.netty.util.concurrent.FastThreadLocalRunnable.run(FastThreadLocalRunnable.java
:30)
    at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)

Locked ownable synchronizers:
- None

"reactor-http-epoll-14" - Thread t@65
java.lang.Thread.State: RUNNABLE
    at app//io.netty.channel.epoll.Native.epollWait(Native Method)
    at app//io.netty.channel.epoll.Native.epollWait(Native.java:209)
    at app//io.netty.channel.epoll.Native.epollWait(Native.java:202)
    at
app//io.netty.channel.epoll.EpollEventLoop.epollWaitNoTimerChange(EpollEventLoop.java:
316)
    at app//io.netty.channel.epoll.EpollEventLoop.run(EpollEventLoop.java:373)
    at
app//io.netty.util.concurrent.SingleThreadEventExecutor$4.run(SingleThreadEventExecuto
r.java:997)
    at app//io.netty.util.internal.ThreadExecutorMap$2.run(ThreadExecutorMap.java:74)
    at
app//io.netty.util.concurrent.FastThreadLocalRunnable.run(FastThreadLocalRunnable.java
:30)
    at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)

Locked ownable synchronizers:
- None

"reactor-http-epoll-15" - Thread t@66
java.lang.Thread.State: RUNNABLE
    at app//io.netty.channel.epoll.Native.epollWait(Native Method)
    at app//io.netty.channel.epoll.Native.epollWait(Native.java:209)
    at app//io.netty.channel.epoll.Native.epollWait(Native.java:202)
    at
app//io.netty.channel.epoll.EpollEventLoop.epollWaitNoTimerChange(EpollEventLoop.java:
316)
    at app//io.netty.channel.epoll.EpollEventLoop.run(EpollEventLoop.java:373)
    at
app//io.netty.util.concurrent.SingleThreadEventExecutor$4.run(SingleThreadEventExecuto
r.java:997)
    at app//io.netty.util.internal.ThreadExecutorMap$2.run(ThreadExecutorMap.java:74)
    at
app//io.netty.util.concurrent.FastThreadLocalRunnable.run(FastThreadLocalRunnable.java
:30)
    at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)

Locked ownable synchronizers:
- None

```

```
"reactor-http-epoll-16" - Thread t@67
  java.lang.Thread.State: RUNNABLE
    at app//io.netty.channel.epoll.Native.epollWait(Native Method)
    at app//io.netty.channel.epoll.Native.epollWait(Native.java:209)
    at app//io.netty.channel.epoll.Native.epollWait(Native.java:202)
    at
app//io.netty.channel.epoll.EpollEventLoop.epollWaitNoTimerChange(EpollEventLoop.java:
316)
    at app//io.netty.channel.epoll.EpollEventLoop.run(EpollEventLoop.java:373)
    at
app//io.netty.util.concurrent.SingleThreadEventExecutor$4.run(SingleThreadEventExecuto
r.java:997)
    at app//io.netty.util.internal.ThreadExecutorMap$2.run(ThreadExecutorMap.java:74)
    at
app//io.netty.util.concurrent.FastThreadLocalRunnable.run(FastThreadLocalRunnable.java
:30)
    at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)
```

Locked ownable synchronizers:  
- None

```
"reactor-http-epoll-17" - Thread t@68
  java.lang.Thread.State: RUNNABLE
    at app//io.netty.channel.epoll.Native.epollWait(Native Method)
    at app//io.netty.channel.epoll.Native.epollWait(Native.java:209)
    at app//io.netty.channel.epoll.Native.epollWait(Native.java:202)
    at
app//io.netty.channel.epoll.EpollEventLoop.epollWaitNoTimerChange(EpollEventLoop.java:
316)
    at app//io.netty.channel.epoll.EpollEventLoop.run(EpollEventLoop.java:373)
    at
app//io.netty.util.concurrent.SingleThreadEventExecutor$4.run(SingleThreadEventExecuto
r.java:997)
    at app//io.netty.util.internal.ThreadExecutorMap$2.run(ThreadExecutorMap.java:74)
    at
app//io.netty.util.concurrent.FastThreadLocalRunnable.run(FastThreadLocalRunnable.java
:30)
    at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)
```

Locked ownable synchronizers:  
- None

```
"reactor-http-epoll-18" - Thread t@69
  java.lang.Thread.State: RUNNABLE
    at app//io.netty.channel.epoll.Native.epollWait(Native Method)
    at app//io.netty.channel.epoll.Native.epollWait(Native.java:209)
    at app//io.netty.channel.epoll.Native.epollWait(Native.java:202)
    at
app//io.netty.channel.epoll.EpollEventLoop.epollWaitNoTimerChange(EpollEventLoop.java:
316)
```

```
    at app//io.netty.channel.epoll.EpollEventLoop.run(EpollEventLoop.java:373)
    at
app//io.netty.util.concurrent.SingleThreadEventExecutor$4.run(SingleThreadEventExecuto
r.java:997)
    at app//io.netty.util.internal.ThreadExecutorMap$2.run(ThreadExecutorMap.java:74)
    at
app//io.netty.util.concurrent.FastThreadLocalRunnable.run(FastThreadLocalRunnable.java
:30)
    at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)
```

Locked ownable synchronizers:  
- None

"reactor-http-epoll-19" - Thread t@70

```
java.lang.Thread.State: RUNNABLE
    at app//io.netty.channel.epoll.Native.epollWait(Native Method)
    at app//io.netty.channel.epoll.Native.epollWait(Native.java:209)
    at app//io.netty.channel.epoll.Native.epollWait(Native.java:202)
    at
app//io.netty.channel.epoll.EpollEventLoop.epollWaitNoTimerChange(EpollEventLoop.java:
316)
    at app//io.netty.channel.epoll.EpollEventLoop.run(EpollEventLoop.java:373)
    at
app//io.netty.util.concurrent.SingleThreadEventExecutor$4.run(SingleThreadEventExecuto
r.java:997)
    at app//io.netty.util.internal.ThreadExecutorMap$2.run(ThreadExecutorMap.java:74)
    at
app//io.netty.util.concurrent.FastThreadLocalRunnable.run(FastThreadLocalRunnable.java
:30)
    at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)
```

Locked ownable synchronizers:  
- None

"reactor-http-epoll-20" - Thread t@71

```
java.lang.Thread.State: RUNNABLE
    at app//io.netty.channel.epoll.Native.epollWait(Native Method)
    at app//io.netty.channel.epoll.Native.epollWait(Native.java:209)
    at app//io.netty.channel.epoll.Native.epollWait(Native.java:202)
    at
app//io.netty.channel.epoll.EpollEventLoop.epollWaitNoTimerChange(EpollEventLoop.java:
316)
    at app//io.netty.channel.epoll.EpollEventLoop.run(EpollEventLoop.java:373)
    at
app//io.netty.util.concurrent.SingleThreadEventExecutor$4.run(SingleThreadEventExecuto
r.java:997)
    at app//io.netty.util.internal.ThreadExecutorMap$2.run(ThreadExecutorMap.java:74)
    at
app//io.netty.util.concurrent.FastThreadLocalRunnable.run(FastThreadLocalRunnable.java
:30)
    at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)
```

Locked ownable synchronizers:

- None

"reactor-http-epoll-21" - Thread t@72

java.lang.Thread.State: RUNNABLE

at app//io.netty.channel.epoll.Native.epollWait(Native Method)

at app//io.netty.channel.epoll.Native.epollWait(Native.java:209)

at app//io.netty.channel.epoll.Native.epollWait(Native.java:202)

at

app//io.netty.channel.epoll.EpollEventLoop.epollWaitNoTimerChange(EpollEventLoop.java:316)

at app//io.netty.channel.epoll.EpollEventLoop.run(EpollEventLoop.java:373)

at

app//io.netty.util.concurrent.SingleThreadEventExecutor\$4.run(SingleThreadEventExecutor.java:997)

at app//io.netty.util.internal.ThreadExecutorMap\$2.run(ThreadExecutorMap.java:74)

at

app//io.netty.util.concurrent.FastThreadLocalRunnable.run(FastThreadLocalRunnable.java:30)

at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)

Locked ownable synchronizers:

- None

"reactor-http-epoll-22" - Thread t@73

java.lang.Thread.State: RUNNABLE

at app//io.netty.channel.epoll.Native.epollWait(Native Method)

at app//io.netty.channel.epoll.Native.epollWait(Native.java:209)

at app//io.netty.channel.epoll.Native.epollWait(Native.java:202)

at

app//io.netty.channel.epoll.EpollEventLoop.epollWaitNoTimerChange(EpollEventLoop.java:316)

at app//io.netty.channel.epoll.EpollEventLoop.run(EpollEventLoop.java:373)

at

app//io.netty.util.concurrent.SingleThreadEventExecutor\$4.run(SingleThreadEventExecutor.java:997)

at app//io.netty.util.internal.ThreadExecutorMap\$2.run(ThreadExecutorMap.java:74)

at

app//io.netty.util.concurrent.FastThreadLocalRunnable.run(FastThreadLocalRunnable.java:30)

at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)

Locked ownable synchronizers:

- None

"reactor-http-epoll-23" - Thread t@74

java.lang.Thread.State: RUNNABLE

at app//io.netty.channel.epoll.Native.epollWait(Native Method)

at app//io.netty.channel.epoll.Native.epollWait(Native.java:209)

at app//io.netty.channel.epoll.Native.epollWait(Native.java:202)

```
at
app//io.netty.channel.epoll.EpollEventLoop.epollWaitNoTimerChange(EpollEventLoop.java:
316)
at app//io.netty.channel.epoll.EpollEventLoop.run(EpollEventLoop.java:373)
at
app//io.netty.util.concurrent.SingleThreadEventExecutor$4.run(SingleThreadEventExecuto
r.java:997)
at app//io.netty.util.internal.ThreadExecutorMap$2.run(ThreadExecutorMap.java:74)
at
app//io.netty.util.concurrent.FastThreadLocalRunnable.run(FastThreadLocalRunnable.java
:30)
at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)
```

Locked ownable synchronizers:

- None

"reactor-http-epoll-24" - Thread t@75

java.lang.Thread.State: RUNNABLE

```
at app//io.netty.channel.epoll.Native.epollWait(Native Method)
at app//io.netty.channel.epoll.Native.epollWait(Native.java:209)
at app//io.netty.channel.epoll.Native.epollWait(Native.java:202)
at
app//io.netty.channel.epoll.EpollEventLoop.epollWaitNoTimerChange(EpollEventLoop.java:
316)
at app//io.netty.channel.epoll.EpollEventLoop.run(EpollEventLoop.java:373)
at
app//io.netty.util.concurrent.SingleThreadEventExecutor$4.run(SingleThreadEventExecuto
r.java:997)
at app//io.netty.util.internal.ThreadExecutorMap$2.run(ThreadExecutorMap.java:74)
at
app//io.netty.util.concurrent.FastThreadLocalRunnable.run(FastThreadLocalRunnable.java
:30)
at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)
```

Locked ownable synchronizers:

- None

"reactor-http-epoll-25" - Thread t@76

java.lang.Thread.State: RUNNABLE

```
at app//io.netty.channel.epoll.Native.epollWait(Native Method)
at app//io.netty.channel.epoll.Native.epollWait(Native.java:209)
at app//io.netty.channel.epoll.Native.epollWait(Native.java:202)
at
app//io.netty.channel.epoll.EpollEventLoop.epollWaitNoTimerChange(EpollEventLoop.java:
316)
at app//io.netty.channel.epoll.EpollEventLoop.run(EpollEventLoop.java:373)
at
app//io.netty.util.concurrent.SingleThreadEventExecutor$4.run(SingleThreadEventExecuto
r.java:997)
at app//io.netty.util.internal.ThreadExecutorMap$2.run(ThreadExecutorMap.java:74)
at
```

```
app//io.netty.util.concurrent.FastThreadLocalRunnable.run(FastThreadLocalRunnable.java:30)
```

```
at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)
```

Locked ownable synchronizers:

- None

"reactor-http-epoll-26" - Thread t@77

java.lang.Thread.State: RUNNABLE

```
at app//io.netty.channel.epoll.Native.epollWait(Native Method)
```

```
at app//io.netty.channel.epoll.Native.epollWait(Native.java:209)
```

```
at app//io.netty.channel.epoll.Native.epollWait(Native.java:202)
```

```
at
```

```
app//io.netty.channel.epoll.EpollEventLoop.epollWaitNoTimerChange(EpollEventLoop.java:316)
```

```
at app//io.netty.channel.epoll.EpollEventLoop.run(EpollEventLoop.java:373)
```

```
at
```

```
app//io.netty.util.concurrent.SingleThreadEventExecutor$4.run(SingleThreadEventExecutor.java:997)
```

```
at app//io.netty.util.internal.ThreadExecutorMap$2.run(ThreadExecutorMap.java:74)
```

```
at
```

```
app//io.netty.util.concurrent.FastThreadLocalRunnable.run(FastThreadLocalRunnable.java:30)
```

```
at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)
```

Locked ownable synchronizers:

- None

"reactor-http-epoll-27" - Thread t@78

java.lang.Thread.State: RUNNABLE

```
at app//io.netty.channel.epoll.Native.epollWait(Native Method)
```

```
at app//io.netty.channel.epoll.Native.epollWait(Native.java:209)
```

```
at app//io.netty.channel.epoll.Native.epollWait(Native.java:202)
```

```
at
```

```
app//io.netty.channel.epoll.EpollEventLoop.epollWaitNoTimerChange(EpollEventLoop.java:316)
```

```
at app//io.netty.channel.epoll.EpollEventLoop.run(EpollEventLoop.java:373)
```

```
at
```

```
app//io.netty.util.concurrent.SingleThreadEventExecutor$4.run(SingleThreadEventExecutor.java:997)
```

```
at app//io.netty.util.internal.ThreadExecutorMap$2.run(ThreadExecutorMap.java:74)
```

```
at
```

```
app//io.netty.util.concurrent.FastThreadLocalRunnable.run(FastThreadLocalRunnable.java:30)
```

```
at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)
```

Locked ownable synchronizers:

- None

"reactor-http-epoll-28" - Thread t@79

java.lang.Thread.State: RUNNABLE

```
at app//io.netty.channel.epoll.Native.epollWait(Native Method)
at app//io.netty.channel.epoll.Native.epollWait(Native.java:209)
at app//io.netty.channel.epoll.Native.epollWait(Native.java:202)
at
app//io.netty.channel.epoll.EpollEventLoop.epollWaitNoTimerChange(EpollEventLoop.java:
316)
at app//io.netty.channel.epoll.EpollEventLoop.run(EpollEventLoop.java:373)
at
app//io.netty.util.concurrent.SingleThreadEventExecutor$4.run(SingleThreadEventExecuto
r.java:997)
at app//io.netty.util.internal.ThreadExecutorMap$2.run(ThreadExecutorMap.java:74)
at
app//io.netty.util.concurrent.FastThreadLocalRunnable.run(FastThreadLocalRunnable.java
:30)
at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)
```

Locked ownable synchronizers:

- None

"reactor-http-epoll-29" - Thread t@80

java.lang.Thread.State: RUNNABLE

```
at app//io.netty.channel.epoll.Native.epollWait(Native Method)
at app//io.netty.channel.epoll.Native.epollWait(Native.java:209)
at app//io.netty.channel.epoll.Native.epollWait(Native.java:202)
at
app//io.netty.channel.epoll.EpollEventLoop.epollWaitNoTimerChange(EpollEventLoop.java:
316)
at app//io.netty.channel.epoll.EpollEventLoop.run(EpollEventLoop.java:373)
at
app//io.netty.util.concurrent.SingleThreadEventExecutor$4.run(SingleThreadEventExecuto
r.java:997)
at app//io.netty.util.internal.ThreadExecutorMap$2.run(ThreadExecutorMap.java:74)
at
app//io.netty.util.concurrent.FastThreadLocalRunnable.run(FastThreadLocalRunnable.java
:30)
at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)
```

Locked ownable synchronizers:

- None

"reactor-http-epoll-30" - Thread t@81

java.lang.Thread.State: RUNNABLE

```
at app//io.netty.channel.epoll.Native.epollWait(Native Method)
at app//io.netty.channel.epoll.Native.epollWait(Native.java:209)
at app//io.netty.channel.epoll.Native.epollWait(Native.java:202)
at
app//io.netty.channel.epoll.EpollEventLoop.epollWaitNoTimerChange(EpollEventLoop.java:
316)
at app//io.netty.channel.epoll.EpollEventLoop.run(EpollEventLoop.java:373)
at
app//io.netty.util.concurrent.SingleThreadEventExecutor$4.run(SingleThreadEventExecuto
```

```

r.java:997)
    at app//io.netty.util.internal.ThreadExecutorMap$2.run(ThreadExecutorMap.java:74)
    at
app//io.netty.util.concurrent.FastThreadLocalRunnable.run(FastThreadLocalRunnable.java
:30)
    at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)

Locked ownable synchronizers:
- None

"reactor-http-epoll-31" - Thread t@82
java.lang.Thread.State: RUNNABLE
    at app//io.netty.channel.epoll.Native.epollWait(Native Method)
    at app//io.netty.channel.epoll.Native.epollWait(Native.java:209)
    at app//io.netty.channel.epoll.Native.epollWait(Native.java:202)
    at
app//io.netty.channel.epoll.EpollEventLoop.epollWaitNoTimerChange(EpollEventLoop.java:
316)
    at app//io.netty.channel.epoll.EpollEventLoop.run(EpollEventLoop.java:373)
    at
app//io.netty.util.concurrent.SingleThreadEventExecutor$4.run(SingleThreadEventExecuto
r.java:997)
    at app//io.netty.util.internal.ThreadExecutorMap$2.run(ThreadExecutorMap.java:74)
    at
app//io.netty.util.concurrent.FastThreadLocalRunnable.run(FastThreadLocalRunnable.java
:30)
    at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)

Locked ownable synchronizers:
- None

"reactor-http-epoll-32" - Thread t@83
java.lang.Thread.State: RUNNABLE
    at app//io.netty.channel.epoll.Native.epollWait(Native Method)
    at app//io.netty.channel.epoll.Native.epollWait(Native.java:209)
    at app//io.netty.channel.epoll.Native.epollWait(Native.java:202)
    at
app//io.netty.channel.epoll.EpollEventLoop.epollWaitNoTimerChange(EpollEventLoop.java:
316)
    at app//io.netty.channel.epoll.EpollEventLoop.run(EpollEventLoop.java:373)
    at
app//io.netty.util.concurrent.SingleThreadEventExecutor$4.run(SingleThreadEventExecuto
r.java:997)
    at app//io.netty.util.internal.ThreadExecutorMap$2.run(ThreadExecutorMap.java:74)
    at
app//io.netty.util.concurrent.FastThreadLocalRunnable.run(FastThreadLocalRunnable.java
:30)
    at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)

Locked ownable synchronizers:
- None

```

```

"boundedElastic-1" - Thread t@84
  java.lang.Thread.State: WAITING
    at java.base@17.0.8.1/jdk.internal.misc.Unsafe.park(Native Method)
      - parking to wait for <2012ed86> (a
java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionObject)
    at
java.base@17.0.8.1/java.util.concurrent.locks.LockSupport.park(LockSupport.java:341)
    at
java.base@17.0.8.1/java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionNode
.block(AbstractQueuedSynchronizer.java:506)
    at
java.base@17.0.8.1/java.util.concurrent.ForkJoinPool.unmanagedBlock(ForkJoinPool.java:
3465)
    at
java.base@17.0.8.1/java.util.concurrent.ForkJoinPool.managedBlock(ForkJoinPool.java:34
36)
    at
java.base@17.0.8.1/java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionObje
ct.await(AbstractQueuedSynchronizer.java:1623)
    at
java.base@17.0.8.1/java.util.concurrent.ScheduledThreadPoolExecutor$DelayedWorkQueue.t
ake(ScheduledThreadPoolExecutor.java:1170)
    at
java.base@17.0.8.1/java.util.concurrent.ScheduledThreadPoolExecutor$DelayedWorkQueue.t
ake(ScheduledThreadPoolExecutor.java:899)
    at
java.base@17.0.8.1/java.util.concurrent.ThreadPoolExecutor.getTask(ThreadPoolExecutor.
java:1062)
    at
java.base@17.0.8.1/java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecuto
r.java:1122)
    at
java.base@17.0.8.1/java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecut
or.java:635)
      at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)

```

Locked ownable synchronizers:

- None

```

"Catalina-utility-1" - Thread t@86
  java.lang.Thread.State: WAITING
    at java.base@17.0.8.1/jdk.internal.misc.Unsafe.park(Native Method)
      - parking to wait for <5c512d91> (a
java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionObject)
    at
java.base@17.0.8.1/java.util.concurrent.locks.LockSupport.park(LockSupport.java:341)
    at
java.base@17.0.8.1/java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionNode
.block(AbstractQueuedSynchronizer.java:506)
    at

```

```

java.base@17.0.8.1/java.util.concurrent.ForkJoinPool.unmanagedBlock(ForkJoinPool.java:
3465)
    at
java.base@17.0.8.1/java.util.concurrent.ForkJoinPool.managedBlock(ForkJoinPool.java:34
36)
    at
java.base@17.0.8.1/java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionObjec
t.await(AbstractQueuedSynchronizer.java:1623)
    at
java.base@17.0.8.1/java.util.concurrent.ScheduledThreadPoolExecutor$DelayedWorkQueue.t
ake(ScheduledThreadPoolExecutor.java:1177)
    at
java.base@17.0.8.1/java.util.concurrent.ScheduledThreadPoolExecutor$DelayedWorkQueue.t
ake(ScheduledThreadPoolExecutor.java:899)
    at
java.base@17.0.8.1/java.util.concurrent.ThreadPoolExecutor.getTask(ThreadPoolExecutor.
java:1062)
    at
java.base@17.0.8.1/java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecuto
r.java:1122)
    at
java.base@17.0.8.1/java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecut
or.java:635)
    at
app//org.apache.tomcat.util.threads.TaskThread$WrappingRunnable.run(TaskThread.java:61
)
    at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)

```

Locked ownable synchronizers:

- None

"Catalina-utility-2" - Thread t@87

```

java.lang.Thread.State: TIMED_WAITING
    at java.base@17.0.8.1/jdk.internal.misc.Unsafe.park(Native Method)
    - parking to wait for <5c512d91> (a
java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionObject)
    at
java.base@17.0.8.1/java.util.concurrent.locks.LockSupport.parkNanos(LockSupport.java:2
52)
    at
java.base@17.0.8.1/java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionObjec
t.awaitNanos(AbstractQueuedSynchronizer.java:1672)
    at
java.base@17.0.8.1/java.util.concurrent.ScheduledThreadPoolExecutor$DelayedWorkQueue.t
ake(ScheduledThreadPoolExecutor.java:1182)
    at
java.base@17.0.8.1/java.util.concurrent.ScheduledThreadPoolExecutor$DelayedWorkQueue.t
ake(ScheduledThreadPoolExecutor.java:899)
    at
java.base@17.0.8.1/java.util.concurrent.ThreadPoolExecutor.getTask(ThreadPoolExecutor.
java:1062)

```

```

    at
java.base@17.0.8.1/java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecuto
r.java:1122)
    at
java.base@17.0.8.1/java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecut
or.java:635)
    at
app//org.apache.tomcat.util.threads.TaskThread$WrappingRunnable.run(TaskThread.java:61
)
    at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)

Locked ownable synchronizers:
- None

"container-0" - Thread t@88
java.lang.Thread.State: TIMED_WAITING
    at java.base@17.0.8.1/java.lang.Thread.sleep(Native Method)
    at app//org.apache.catalina.core.StandardServer.await(StandardServer.java:565)
    at
app//org.springframework.boot.web.embedded.tomcat.TomcatWebServer$1.run(TomcatWebServe
r.java:197)

Locked ownable synchronizers:
- None

"http-nio-auto-1-exec-1" - Thread t@89
java.lang.Thread.State: WAITING
    at java.base@17.0.8.1/jdk.internal.misc.Unsafe.park(Native Method)
    - parking to wait for <6022a4c4> (a
java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionObject)
    at
java.base@17.0.8.1/java.util.concurrent.locks.LockSupport.park(LockSupport.java:341)
    at
java.base@17.0.8.1/java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionNode
.block(AbstractQueuedSynchronizer.java:506)
    at
java.base@17.0.8.1/java.util.concurrent.ForkJoinPool.unmanagedBlock(ForkJoinPool.java:
3465)
    at
java.base@17.0.8.1/java.util.concurrent.ForkJoinPool.managedBlock(ForkJoinPool.java:34
36)
    at
java.base@17.0.8.1/java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionObje
ct.await(AbstractQueuedSynchronizer.java:1623)
    at
java.base@17.0.8.1/java.util.concurrent.LinkedBlockingQueue.take(LinkedBlockingQueue.j
ava:435)
    at app//org.apache.tomcat.util.threads.TaskQueue.take(TaskQueue.java:117)
    at app//org.apache.tomcat.util.threads.TaskQueue.take(TaskQueue.java:33)
    at
app//org.apache.tomcat.util.threads.ThreadPoolExecutor.getTask(ThreadPoolExecutor.java

```

```

:1114)
    at
app//org.apache.tomcat.util.threads.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1176)
    at
app//org.apache.tomcat.util.threads.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:659)
    at
app//org.apache.tomcat.util.threads.TaskThread$WrappingRunnable.run(TaskThread.java:61)
)
    at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)

Locked ownable synchronizers:
- None

"http-nio-auto-1-exec-2" - Thread t@90
java.lang.Thread.State: WAITING
    at java.base@17.0.8.1/jdk.internal.misc.Unsafe.park(Native Method)
    - parking to wait for <6022a4c4> (a
java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionObject)
    at
java.base@17.0.8.1/java.util.concurrent.locks.LockSupport.park(LockSupport.java:341)
    at
java.base@17.0.8.1/java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionNode
.block(AbstractQueuedSynchronizer.java:506)
    at
java.base@17.0.8.1/java.util.concurrent.ForkJoinPool.unmanagedBlock(ForkJoinPool.java:
3465)
    at
java.base@17.0.8.1/java.util.concurrent.ForkJoinPool.managedBlock(ForkJoinPool.java:34
36)
    at
java.base@17.0.8.1/java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionObje
ct.await(AbstractQueuedSynchronizer.java:1623)
    at
java.base@17.0.8.1/java.util.concurrent.LinkedBlockingQueue.take(LinkedBlockingQueue.j
ava:435)
    at app//org.apache.tomcat.util.threads.TaskQueue.take(TaskQueue.java:117)
    at app//org.apache.tomcat.util.threads.TaskQueue.take(TaskQueue.java:33)
    at
app//org.apache.tomcat.util.threads.ThreadPoolExecutor.getTask(ThreadPoolExecutor.java
:1114)
    at
app//org.apache.tomcat.util.threads.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.ja
va:1176)
    at
app//org.apache.tomcat.util.threads.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.j
ava:659)
    at
app//org.apache.tomcat.util.threads.TaskThread$WrappingRunnable.run(TaskThread.java:61
)

```

```
at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)
```

Locked ownable synchronizers:

- None

"http-nio-auto-1-exec-3" - Thread t@91

java.lang.Thread.State: WAITING

at java.base@17.0.8.1/jdk.internal.misc.Unsafe.park(Native Method)

- parking to wait for <6022a4c4> (a

java.util.concurrent.locks.AbstractQueuedSynchronizer\$ConditionObject)

at

java.base@17.0.8.1/java.util.concurrent.locks.LockSupport.park(LockSupport.java:341)

at

java.base@17.0.8.1/java.util.concurrent.locks.AbstractQueuedSynchronizer\$ConditionNode.  
.block(AbstractQueuedSynchronizer.java:506)

at

java.base@17.0.8.1/java.util.concurrent.ForkJoinPool.unmanagedBlock(ForkJoinPool.java:  
3465)

at

java.base@17.0.8.1/java.util.concurrent.ForkJoinPool.managedBlock(ForkJoinPool.java:34  
36)

at

java.base@17.0.8.1/java.util.concurrent.locks.AbstractQueuedSynchronizer\$ConditionObje  
ct.await(AbstractQueuedSynchronizer.java:1623)

at

java.base@17.0.8.1/java.util.concurrent.LinkedBlockingQueue.take(LinkedBlockingQueue.j  
ava:435)

at app//org.apache.tomcat.util.threads.TaskQueue.take(TaskQueue.java:117)

at app//org.apache.tomcat.util.threads.TaskQueue.take(TaskQueue.java:33)

at

app//org.apache.tomcat.util.threads.ThreadPoolExecutor.getTask(ThreadPoolExecutor.java  
:1114)

at

app//org.apache.tomcat.util.threads.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.ja  
va:1176)

at

app//org.apache.tomcat.util.threads.ThreadPoolExecutor\$Worker.run(ThreadPoolExecutor.j  
ava:659)

at

app//org.apache.tomcat.util.threads.TaskThread\$WrappingRunnable.run(TaskThread.java:61  
)

at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)

Locked ownable synchronizers:

- None

"http-nio-auto-1-exec-4" - Thread t@92

java.lang.Thread.State: WAITING

at java.base@17.0.8.1/jdk.internal.misc.Unsafe.park(Native Method)

- parking to wait for <6022a4c4> (a

java.util.concurrent.locks.AbstractQueuedSynchronizer\$ConditionObject)

```

    at
java.base@17.0.8.1/java.util.concurrent.locks.LockSupport.park(LockSupport.java:341)
    at
java.base@17.0.8.1/java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionNode
.block(AbstractQueuedSynchronizer.java:506)
    at
java.base@17.0.8.1/java.util.concurrent.ForkJoinPool.unmanagedBlock(ForkJoinPool.java:
3465)
    at
java.base@17.0.8.1/java.util.concurrent.ForkJoinPool.managedBlock(ForkJoinPool.java:34
36)
    at
java.base@17.0.8.1/java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionObje
ct.await(AbstractQueuedSynchronizer.java:1623)
    at
java.base@17.0.8.1/java.util.concurrent.LinkedBlockingQueue.take(LinkedBlockingQueue.j
ava:435)
        at app//org.apache.tomcat.util.threads.TaskQueue.take(TaskQueue.java:117)
        at app//org.apache.tomcat.util.threads.TaskQueue.take(TaskQueue.java:33)
    at
app//org.apache.tomcat.util.threads.ThreadPoolExecutor.getTask(ThreadPoolExecutor.java
:1114)
    at
app//org.apache.tomcat.util.threads.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.ja
va:1176)
    at
app//org.apache.tomcat.util.threads.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.j
ava:659)
    at
app//org.apache.tomcat.util.threads.TaskThread$WrappingRunnable.run(TaskThread.java:61
)
        at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)

Locked ownable synchronizers:
- None

"http-nio-auto-1-exec-5" - Thread t@93
java.lang.Thread.State: WAITING
    at java.base@17.0.8.1/jdk.internal.misc.Unsafe.park(Native Method)
        - parking to wait for <6022a4c4> (a
java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionObject)
    at
java.base@17.0.8.1/java.util.concurrent.locks.LockSupport.park(LockSupport.java:341)
    at
java.base@17.0.8.1/java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionNode
.block(AbstractQueuedSynchronizer.java:506)
    at
java.base@17.0.8.1/java.util.concurrent.ForkJoinPool.unmanagedBlock(ForkJoinPool.java:
3465)
    at
java.base@17.0.8.1/java.util.concurrent.ForkJoinPool.managedBlock(ForkJoinPool.java:34

```

```

36)
    at
java.base@17.0.8.1/java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionObject.await(AbstractQueuedSynchronizer.java:1623)
    at
java.base@17.0.8.1/java.util.concurrent.LinkedBlockingQueue.take(LinkedBlockingQueue.java:435)
    at app//org.apache.tomcat.util.threads.TaskQueue.take(TaskQueue.java:117)
    at app//org.apache.tomcat.util.threads.TaskQueue.take(TaskQueue.java:33)
    at
app//org.apache.tomcat.util.threads.ThreadPoolExecutor.getTask(ThreadPoolExecutor.java:1114)
    at
app//org.apache.tomcat.util.threads.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1176)
    at
app//org.apache.tomcat.util.threads.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:659)
    at
app//org.apache.tomcat.util.threads.TaskThread$WrappingRunnable.run(TaskThread.java:61)
    )
    at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)

Locked ownable synchronizers:
- None

"http-nio-auto-1-exec-6" - Thread t@94
java.lang.Thread.State: WAITING
    at java.base@17.0.8.1/jdk.internal.misc.Unsafe.park(Native Method)
    - parking to wait for <6022a4c4> (a
java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionObject)
    at
java.base@17.0.8.1/java.util.concurrent.locks.LockSupport.park(LockSupport.java:341)
    at
java.base@17.0.8.1/java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionNode.block(AbstractQueuedSynchronizer.java:506)
    at
java.base@17.0.8.1/java.util.concurrent.ForkJoinPool.unmanagedBlock(ForkJoinPool.java:3465)
    at
java.base@17.0.8.1/java.util.concurrent.ForkJoinPool.managedBlock(ForkJoinPool.java:3436)
    at
java.base@17.0.8.1/java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionObject.await(AbstractQueuedSynchronizer.java:1623)
    at
java.base@17.0.8.1/java.util.concurrent.LinkedBlockingQueue.take(LinkedBlockingQueue.java:435)
    at app//org.apache.tomcat.util.threads.TaskQueue.take(TaskQueue.java:117)
    at app//org.apache.tomcat.util.threads.TaskQueue.take(TaskQueue.java:33)
    at

```

```

app//org.apache.tomcat.util.threads.ThreadPoolExecutor.getTask(ThreadPoolExecutor.java:1114)
    at
app//org.apache.tomcat.util.threads.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1176)
    at
app//org.apache.tomcat.util.threads.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:659)
    at
app//org.apache.tomcat.util.threads.TaskThread$WrappingRunnable.run(TaskThread.java:61)
    at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)

Locked ownable synchronizers:
- None

"http-nio-auto-1-exec-7" - Thread t@95
  java.lang.Thread.State: WAITING
    at java.base@17.0.8.1/jdk.internal.misc.Unsafe.park(Native Method)
    - parking to wait for <6022a4c4> (a
java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionObject)
    at
java.base@17.0.8.1/java.util.concurrent.locks.LockSupport.park(LockSupport.java:341)
    at
java.base@17.0.8.1/java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionNode.block(AbstractQueuedSynchronizer.java:506)
    at
java.base@17.0.8.1/java.util.concurrent.ForkJoinPool.unmanagedBlock(ForkJoinPool.java:3465)
    at
java.base@17.0.8.1/java.util.concurrent.ForkJoinPool.managedBlock(ForkJoinPool.java:3436)
    at
java.base@17.0.8.1/java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionObject.await(AbstractQueuedSynchronizer.java:1623)
    at
java.base@17.0.8.1/java.util.concurrent.LinkedBlockingQueue.take(LinkedBlockingQueue.java:435)
    at app//org.apache.tomcat.util.threads.TaskQueue.take(TaskQueue.java:117)
    at app//org.apache.tomcat.util.threads.TaskQueue.take(TaskQueue.java:33)
    at
app//org.apache.tomcat.util.threads.ThreadPoolExecutor.getTask(ThreadPoolExecutor.java:1114)
    at
app//org.apache.tomcat.util.threads.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1176)
    at
app//org.apache.tomcat.util.threads.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:659)
    at
app//org.apache.tomcat.util.threads.TaskThread$WrappingRunnable.run(TaskThread.java:61)

```

```

)
  at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)

Locked ownable synchronizers:
- None

"http-nio-auto-1-exec-8" - Thread t@96
  java.lang.Thread.State: WAITING
    at java.base@17.0.8.1/jdk.internal.misc.Unsafe.park(Native Method)
    - parking to wait for <6022a4c4> (a
java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionObject)
    at
java.base@17.0.8.1/java.util.concurrent.locks.LockSupport.park(LockSupport.java:341)
    at
java.base@17.0.8.1/java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionNode
.block(AbstractQueuedSynchronizer.java:506)
    at
java.base@17.0.8.1/java.util.concurrent.ForkJoinPool.unmanagedBlock(ForkJoinPool.java:
3465)
    at
java.base@17.0.8.1/java.util.concurrent.ForkJoinPool.managedBlock(ForkJoinPool.java:34
36)
    at
java.base@17.0.8.1/java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionObje
ct.await(AbstractQueuedSynchronizer.java:1623)
    at
java.base@17.0.8.1/java.util.concurrent.LinkedBlockingQueue.take(LinkedBlockingQueue.j
ava:435)
    at app//org.apache.tomcat.util.threads.TaskQueue.take(TaskQueue.java:117)
    at app//org.apache.tomcat.util.threads.TaskQueue.take(TaskQueue.java:33)
    at
app//org.apache.tomcat.util.threads.ThreadPoolExecutor.getTask(ThreadPoolExecutor.java
:1114)
    at
app//org.apache.tomcat.util.threads.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.ja
va:1176)
    at
app//org.apache.tomcat.util.threads.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.j
ava:659)
    at
app//org.apache.tomcat.util.threads.TaskThread$WrappingRunnable.run(TaskThread.java:61
)
  at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)

Locked ownable synchronizers:
- None

"http-nio-auto-1-exec-9" - Thread t@97
  java.lang.Thread.State: WAITING
    at java.base@17.0.8.1/jdk.internal.misc.Unsafe.park(Native Method)
    - parking to wait for <6022a4c4> (a

```

```

java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionObject)
    at
java.base@17.0.8.1/java.util.concurrent.locks.LockSupport.park(LockSupport.java:341)
    at
java.base@17.0.8.1/java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionNode
.block(AbstractQueuedSynchronizer.java:506)
    at
java.base@17.0.8.1/java.util.concurrent.ForkJoinPool.unmanagedBlock(ForkJoinPool.java:
3465)
    at
java.base@17.0.8.1/java.util.concurrent.ForkJoinPool.managedBlock(ForkJoinPool.java:34
36)
    at
java.base@17.0.8.1/java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionObje
ct.await(AbstractQueuedSynchronizer.java:1623)
    at
java.base@17.0.8.1/java.util.concurrent.LinkedBlockingQueue.take(LinkedBlockingQueue.j
ava:435)
    at app//org.apache.tomcat.util.threads.TaskQueue.take(TaskQueue.java:117)
    at app//org.apache.tomcat.util.threads.TaskQueue.take(TaskQueue.java:33)
    at
app//org.apache.tomcat.util.threads.ThreadPoolExecutor.getTask(ThreadPoolExecutor.java
:1114)
    at
app//org.apache.tomcat.util.threads.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.ja
va:1176)
    at
app//org.apache.tomcat.util.threads.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.j
ava:659)
    at
app//org.apache.tomcat.util.threads.TaskThread$WrappingRunnable.run(TaskThread.java:61
)
    at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)

Locked ownable synchronizers:
- None

"http-nio-auto-1-exec-10" - Thread t@98
java.lang.Thread.State: WAITING
    at java.base@17.0.8.1/jdk.internal.misc.Unsafe.park(Native Method)
    - parking to wait for <6022a4c4> (a
java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionObject)
    at
java.base@17.0.8.1/java.util.concurrent.locks.LockSupport.park(LockSupport.java:341)
    at
java.base@17.0.8.1/java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionNode
.block(AbstractQueuedSynchronizer.java:506)
    at
java.base@17.0.8.1/java.util.concurrent.ForkJoinPool.unmanagedBlock(ForkJoinPool.java:
3465)
    at

```

```

java.base@17.0.8.1/java.util.concurrent.ForkJoinPool.managedBlock(ForkJoinPool.java:3436)
    at
java.base@17.0.8.1/java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionObject.await(AbstractQueuedSynchronizer.java:1623)
    at
java.base@17.0.8.1/java.util.concurrent.LinkedBlockingQueue.take(LinkedBlockingQueue.java:435)
    at app//org.apache.tomcat.util.threads.TaskQueue.take(TaskQueue.java:117)
    at app//org.apache.tomcat.util.threads.TaskQueue.take(TaskQueue.java:33)
    at
app//org.apache.tomcat.util.threads.ThreadPoolExecutor.getTask(ThreadPoolExecutor.java:1114)
    at
app//org.apache.tomcat.util.threads.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1176)
    at
app//org.apache.tomcat.util.threads.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:659)
    at
app//org.apache.tomcat.util.threads.TaskThread$WrappingRunnable.run(TaskThread.java:61)
    at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)

```

Locked ownable synchronizers:

- None

"http-nio-auto-1-Poller" - Thread t@99

java.lang.Thread.State: RUNNABLE

at java.base@17.0.8.1/sun.nio.ch.EPoll.wait(Native Method)

at

java.base@17.0.8.1/sun.nio.ch.EPollSelectorImpl.doSelect(EPollSelectorImpl.java:118)

at

java.base@17.0.8.1/sun.nio.ch.SelectorImpl.lockAndDoSelect(SelectorImpl.java:129)

- locked <11092844> (a sun.nio.ch.Util\$2)

- locked <2640893e> (a sun.nio.ch.EPollSelectorImpl)

at java.base@17.0.8.1/sun.nio.ch.SelectorImpl.select(SelectorImpl.java:141)

at app//org.apache.tomcat.util.net.NioEndpoint\$Poller.run(NioEndpoint.java:758)

at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)

Locked ownable synchronizers:

- None

"http-nio-auto-1-Acceptor" - Thread t@100

java.lang.Thread.State: RUNNABLE

at java.base@17.0.8.1/sun.nio.ch.Net.accept(Native Method)

at

java.base@17.0.8.1/sun.nio.ch.ServerSocketChannelImpl.implAccept(ServerSocketChannelImpl.java:425)

at

java.base@17.0.8.1/sun.nio.ch.ServerSocketChannelImpl.accept(ServerSocketChannelImpl.j

```

ava:391)
    at
app//org.apache.tomcat.util.net.NioEndpoint.serverSocketAccept(NioEndpoint.java:521)
    at
app//org.apache.tomcat.util.net.NioEndpoint.serverSocketAccept(NioEndpoint.java:79)
    at app//org.apache.tomcat.util.net.Acceptor.run(Acceptor.java:128)
    at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)

Locked ownable synchronizers:
- Locked <73a815c2> (a java.util.concurrent.locks.ReentrantLock$NonfairSync)

"pool-4-thread-1" - Thread t@105
  java.lang.Thread.State: RUNNABLE
    at java.base@17.0.8.1/jdk.internal.misc.VM.getNanoTimeAdjustment(Native Method)
    at java.base@17.0.8.1/java.time.Clock.currentInstant(Clock.java:498)
    at java.base@17.0.8.1/java.time.Clock$SystemClock.instant(Clock.java:614)
    at
app//org.springframework.scheduling.concurrent.ReschedulingRunnable.run(ReschedulingRunnable.java:97)
    at
java.base@17.0.8.1/java.util.concurrent.Executors$RunnableAdapter.call(Executors.java:539)
    at java.base@17.0.8.1/java.util.concurrent.FutureTask.run(FutureTask.java:264)
    at
java.base@17.0.8.1/java.util.concurrent.ScheduledThreadPoolExecutor$ScheduledFutureTask.run(ScheduledThreadPoolExecutor.java:304)
    at
java.base@17.0.8.1/java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1136)
    at
java.base@17.0.8.1/java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:635)
    at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)

Locked ownable synchronizers:
- Locked <3cc135a9> (a java.util.concurrent.ThreadPoolExecutor$Worker)

"Thread-58" - Thread t@108
  java.lang.Thread.State: TIMED_WAITING
    at java.base@17.0.8.1/java.lang.Thread.sleep(Native Method)
    at
app//org.springframework.boot.actuate.context.ShutdownEndpoint.performShutdown(ShutdownEndpoint.java:57)
    at
app//org.springframework.boot.actuate.context.ShutdownEndpoint$$Lambda$1920/0x00007ff0e8b10000.run(Unknown Source)
    at java.base@17.0.8.1/java.lang.Thread.run(Thread.java:833)

Locked ownable synchronizers:
- None

```