



## Red Hat Data Grid 8.4

# Using the RESP protocol endpoint with Data Grid

Use the Data Grid RESP endpoint to interact with remote caches



## Red Hat Data Grid 8.4 Using the RESP protocol endpoint with Data Grid

---

Use the Data Grid RESP endpoint to interact with remote caches

## Legal Notice

Copyright © 2024 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, the Red Hat logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux<sup>®</sup> is the registered trademark of Linus Torvalds in the United States and other countries.

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

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

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

Node.js<sup>®</sup> is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack<sup>®</sup> Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

## Abstract

The RESP endpoint allows Redis clients to connect to one or more Data Grid backed RESP servers and perform cache operations.

---

## Table of Contents

RED HAT DATA GRID .....	3
DATA GRID DOCUMENTATION .....	4
DATA GRID DOWNLOADS .....	5
MAKING OPEN SOURCE MORE INCLUSIVE .....	6
CHAPTER 1. USING THE RESP ENDPOINT .....	7
1.1. TECHNOLOGY PREVIEWS .....	7
1.2. ENABLING THE RESP ENDPOINT .....	7
RESP endpoint configuration .....	8
1.3. REDIS COMMANDS .....	9



# RED HAT DATA GRID

Data Grid is a high-performance, distributed in-memory data store.

## **Schemaless data structure**

Flexibility to store different objects as key-value pairs.

## **Grid-based data storage**

Designed to distribute and replicate data across clusters.

## **Elastic scaling**

Dynamically adjust the number of nodes to meet demand without service disruption.

## **Data interoperability**

Store, retrieve, and query data in the grid from different endpoints.

## DATA GRID DOCUMENTATION

Documentation for Data Grid is available on the Red Hat customer portal.

- [Data Grid 8.4 Documentation](#)
- [Data Grid 8.4 Component Details](#)
- [Supported Configurations for Data Grid 8.4](#)
- [Data Grid 8 Feature Support](#)
- [Data Grid Deprecated Features and Functionality](#)



## DATA GRID DOWNLOADS

Access the [Data Grid Software Downloads](#) on the Red Hat customer portal.



### NOTE

You must have a Red Hat account to access and download Data Grid software.

## MAKING OPEN SOURCE MORE INCLUSIVE

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

# CHAPTER 1. USING THE RESP ENDPOINT

Data Grid Server includes an experimental module that implements the [RESP3 protocol](#). The RESP endpoint allows Redis clients to connect to one or several Data Grid-backed RESP servers and perform cache operations.



## IMPORTANT

RESP protocol endpoint is available as a technology preview feature.

## 1.1. TECHNOLOGY PREVIEWS

Technology Preview features or capabilities are not supported with Red Hat production service-level agreements (SLAs) and might not be functionally complete.

Red Hat does not recommend using Technology Preview features or capabilities for production. These features provide early access to upcoming product features, which enables you to test functionality and provide feedback during the development process.

For more information, see [Red Hat Technology Preview Features Support Scope](#).

## 1.2. ENABLING THE RESP ENDPOINT

Add the **resp-connector** to Data Grid Server configuration to enable the RESP endpoint. You can enable the RESP endpoint with:

- Standalone Data Grid Server deployments, exactly like standalone Redis, where each server instance runs independently of each other.
- Clustered Data Grid Server deployments, where server instances replicate or distribute data between each other. Clustered deployments provides clients with failover capabilities.

### Prerequisites

- Install Data Grid Server.

### Procedure

1. Open your Data Grid Server configuration for editing.
2. Add cache configuration to the **cache-container** section if required.  
Cache configuration cannot enable capabilities that violate the RESP protocol. For example, specifying expiration values in a cache for the RESP endpoint results in a fatal error at startup.

### TIP

Configure your cache with Protobuf encoding if you want to view cache entries in the Data Grid Console (**encoding media-type="application/x-protostream"**).

3. Add an **endpoint** declaration to your configuration.
4. Add the **resp-connector** element and specify the name of the cache to use with the RESP connector with the **cache** attribute.

You can use only one cache with the RESP endpoint.

5. Declare the security realm to use with the RESP endpoint with the **security-realm** attribute, if required.
6. Ensure that the endpoint declaration also adds a Hot Rod and REST connector.
7. Save the changes to your configuration.

## Verification

When you start Data Grid Server check for the following log message:

```
[org.infinispan.SERVER] ISPN080018: Started connector Resp (internal)
```

You can now connect to the RESP endpoint with a Redis client. For example, with the Redis CLI you can do the following to add an entry to the cache:

```
redis-cli -p 11222 --user username --pass password
```

```
127.0.0.1:11222> SET k v
OK
127.0.0.1:11222> GET k
"v"
127.0.0.1:11222> quit
```

## RESP endpoint configuration

### XML

```
<endpoints>
  <endpoint socket-binding="default" security-realm="default">
    <resp-connector cache="mycache" />
    <hotrod-connector />
    <rest-connector/>
  </endpoint>
</endpoints>
```

### JSON

```
{
  "server": {
    "endpoints": {
      "endpoint": {
        "socket-binding": "default",
        "security-realm": "default",
        "resp-connector": {
          "cache": "mycache"
        },
        "hotrod-connector": {},
        "rest-connector": {}
      }
    }
  }
}
```

```
}  
}  
}
```

## YAML

```
server:  
  endpoints:  
    endpoint:  
      socketBinding: "default"  
      securityRealm: "default"  
      respConnector:  
        cache: "mycache"  
      hotrodConnector: ~  
      restConnector: ~
```

## 1.3. REDIS COMMANDS

The Data Grid RESP endpoint implements the following Redis commands:

- [AUTH](#)
- [DECR](#)
- [DEL](#)
- [ECHO](#)
- [GET](#)
- [HELLO](#)
- [INCR](#)
- [MGET](#)
- [MSET](#)
- [PING](#)
- [PUBLISH](#)
- [QUIT](#)
- [RESET](#)
- [SET](#)
- [SUBSCRIBE](#)
- [UNSUBSCRIBE](#)

