Redis Enterprise Software Release Notes 5.4.14 (February 2020)

Redis Enterprise Software (RS) 5.4.14 is now available. This release bundles OSS Redis 5.0.7 and includes new Redis Modules versions, several enhancements, and bug fixes.

Overview

Follow these instructions for upgrading to RS 5.4.14 from RS 5.0.2 and above.

New features

- Version 5.0.7 of OSS Redis is merged into RS 5.4.14.
- The following GA releases of Redis Modules are bundled in RS 5.4.14:
  - RedisBloom, version 2.2.1 (updated, release notes)
  - RedisGraph, version 2.0.1 (updated, release notes)
  - RedisJSON, version 1.0.4 (update, release notes)
  - RediSearch, version 1.4.25 (updated, release notes)
  - RedisTimeSeries, version 1.2.3 (updated, release notes)

Additional capabilities

- Added the ability to retrieve license details with a REST API command. Now you can get your license details from the admin console (settings > general) or from the REST API command:
  
  ```
  GET https://localhost:9443/v1/license
  ```

  The REST API response includes:
  - license - The cluster's name (FQDN) in the key string
  - expired - If the cluster key is expired (True or False)
  - activation_date - The date of the cluster key activation
  - expiration_date - The date of the cluster key expiration
  - shards_limit - The number of shards allowed by the cluster key

- Added Flush command in the UI console for Active-Active databases. The command flushes the data from all participating clusters.

- Updated radmin upgrade module command so module arguments are optional. When you upgrade the module for a database, you can either:
  - Specify the module arguments to replace the existing arguments
  - Specify the keep_module_args flag to use the existing arguments

Examples:

1. To upgrade the version of RediSearch to 10017 and replace the module arguments:

   ```
   radmin upgrade module db_name MyAwesomeDB module_name ft version 10017 module_args "PARTITIONS AUTO"
   ```

2. To upgrade RedisBloom to version 10100 and remove the current module arguments:
3. To upgrade RedisJSON to 10002 and use the current module arguments:

```
rladmin upgrade module db_name MyDB module_name ReJSON version 10002 keep_module_args
```

- Upgraded js-yaml version to 3.13.1, and lodash version to 4.17.15 (RS31819)
- Added an alert for actions that run for an extended period of time. For example, a notification is sent when an action like updating database property runs for more than a configurable threshold. Default value is set to 24 hours.

**Information**

- End of Life (EOL) for Redis Enterprise Software 5.4, as well as for Redis Modules and previous RS versions, can be found here.
- Google Chrome browser on macOS Catalina requires self-signed certificate generated after June 2019 to include the extendedKeyUsage field in order to connect to the RS admin console. If you use a self-signed certificate that does not include this field, update the self-signed certificate.
- When you upgrade an Active-Active Redis with active AOF from version RS 5.4.2 or lower to version RS 5.4.4 or higher:
  - If replication is enabled, you must run the BGREWRITEAOF command on all slave shards after the upgrade.
  - If replication is not enabled, you must run the BGREWRITEAOF command on all shards after the upgrade.

**Important fixes**

- RS23396 - Improved the disconnection mechanism in case of inactive UI session
- RS27924 - Added descriptive error messages in the UI console when validating the license
- RS29968 - Improved internal mechanism to better support high scale of clients connections
- RS35675 - Updated the upgrade process of Active-Active Redis databases to save causal consistency and encryption flags
- RS36922 - Fixed an issue in a specific cluster upgrade scenario from versions earlier than RS 5.4.0 to RS 5.4.0 or higher

**Known limitations**

**Upgrade**

- When you upgrade a cluster from version 5.0.2-20 to version 5.4.14, you must first upgrade to version 5.2.2 and then to version 5.4.14.
- RS 5.4.2 introduced new Active-Active Redis (CRDB) capabilities that improve its compatibility with open source Redis. Now the string data-type in Active-Active Redis (CRDB) is implicitly and dynamically typed, just like open source Redis. To use the new capabilities on nodes that are upgraded from version RS 5.4.2 or lower, you must upgrade the CRDB protocol.
- Before you upgrade a database with RediSearch Module to Redis 5.0, you must upgrade the RediSearch Module to version 1.4.2 or above.
- Node upgrade fails if the SSL certificates were configured in version 5.0.2 or above by manually updating the certificates on the disk instead of updating them through the API. For assistance with this issue, contact Redis support.
- We recommend that you test module upgrade commands in a test environment before you upgrade modules in a production environment. The module upgrade arguments are not validated during the upgrade process and incorrect
arguments can cause unexpected downtime.

- Starting from RS 5.4.2, to preserve the current Redis major.minor version during database upgrade you must use the `keep_redis_version` option instead of `keep_current_version`.

**Cluster API**

- The API for removing a node is updated in RS 5.4.2 or higher. The API call must include json data and the "Content-Type: application/json" header. For example:

```bash
curl -X POST -H "Content-Type: application/json" -i -k -u user@redislabs.com:password https://localhost:9443/v1/nodes/3/actions/remove --data "{}"
```

**Discovery service**

- For Redis Sentinel (Discovery Service), every database name must be unique across the cluster.

**Redis commands**

- The capability of disabling specific Redis commands does not work on commands specific to Redis Modules.
- The CLIENT ID command cannot guarantee incremental IDs between clients that connect to different nodes under multi proxy policies.
- CLIENT UNBLOCK command is not supported in RS 5.4 and above
- Starting from RS 5.4.2 and after upgrading the CRDB, TYPE command for string data-type in CRDBs return "string" (OSS Redis standard).

---

**Updated:** March 11, 2022