REDB custom resource options

The database options are specified in the spec section of the database custom resource. These options include options that you can change and options that are created by the controller for applications or developers. Changes made to the REDB custom resource will override changes made to the database via the admin console or rladmin commands.

The most common options are listed below. For a complete list of options, see Redis Enterprise Database API.

**databasePort**
Manually sets the TCP port on which the database is available. If the port number is not specified, it will be automatically generated.

> **Warning** - databasePort cannot be added, removed, or changed after database creation.

If the admission controller is installed and configured, it will run validity checks before the REDB creation. If the port is not available, you'll get an error for "port is unavailable." If the port is not valid, you'll get an error for "change databasePort is not allowed."

**databaseSecretName**
A string containing the name of a secret that contains the desired database password.

If you specify a secret name, you must create an opaque secret before you create the database resource. The operator takes the password from the password key in the secret.

To disable authentication for the database, use an empty string as a value for the password key.

If you do not specify a secret name, the operator will create a secret for you with the name constructed from the database name (metadata.name) with "re<db>-" as a prefix.

When the database is created, the secret is updated to include the port and service name for the database, but the password does not change. If you did not create the secret, it is also updated with the generated database password.

**evictionPolicy**
An eviction policy (default: volatile-lru)

**memorySize**
The amount of memory to allocate that is at least 100MB. Values are an integer suffixed with a unit. For example, values like 1GB, 250MB, etc.

**persistence**
The value for the database persistence setting.

The value is a keyword with the values:

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>disabled</td>
<td>Data is not persisted (default)</td>
</tr>
<tr>
<td>Value</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>aofEverySecond</td>
<td>Data is synced to disk every second</td>
</tr>
<tr>
<td>aofAlways</td>
<td>Data is synced to disk with every write.</td>
</tr>
<tr>
<td>snapshotEvery1Hour</td>
<td>A snapshot of the database is created every hour</td>
</tr>
<tr>
<td>snapshotEvery6Hour</td>
<td>A snapshot of the database is created every 6 hours.</td>
</tr>
<tr>
<td>snapshotEvery12Hour</td>
<td>A snapshot of the database is created every 12 hours.</td>
</tr>
</tbody>
</table>

**rackAware**

A boolean that indicates whether the database is rack-zone aware (default: the cluster setting)

**redisEnterpriseCluster**

The name of the cluster to create the database on.

The value has a single name property. For example, to refer to the rec cluster:

```yaml
redisEnterpriseCluster:
  name: rec
```

**redisVersion**

The upgrade policy specific to an REDB: major or latest.

Specifying this field during REDB creation will determine if the database version is the latest available, or the latest major version available.

If it is not specified, the database will follow the upgrade policy set on the cluster. If the cluster version is set to major, only major will be accepted on REDBs. If the cluster is set to latest, both latest and major are valid for the REDB.

**replication**

A boolean that indicates whether in-memory database replication is enabled (default: false).

When enabled, the database has a replica shard for every master.

**shardCount**

The number of database shards (default: 1).

**tlsMode**

Controls SSL authentication and encryption for connections to the database.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>disabled</td>
<td>no incoming connection to the Database uses SSL (default)</td>
</tr>
<tr>
<td>enabled</td>
<td>all incoming connections to the Database must use SSL.</td>
</tr>
<tr>
<td>replica_ssl</td>
<td>databases that replicate from this database must use SSL.</td>
</tr>
</tbody>
</table>

Updated: May 11, 2022