

Redis 从入门到精通

黄健宏(huangz)



服务器配置

通过调整服务器来适应你的需要



Redis 服务器提供了一些配置选项(configuration option), 通过修改这些选项的值, 可以改变选项对应功能的行为。

举个例子, 前面介绍 SELECT 命令时曾经说过, Redis 服务器默认会创建 0 号至 15 号共十六个数据库以供用户使用。

但 Redis 服务器的数据库数量并不是一成不变的, Redis 提供了 databases 选项, 它的默认值为 16, 通过修改这个选项的值, 我们可以让服务器创建指定数量的数据库, 比如 5 个、10 个、32 个、100 个, 诸如此类。

再举一个例子, 在介绍 Lua 脚本的时候, 我们曾说过, 如果一个脚本的运行时间过长, 用户可以使用 SCRIPT KILL 命令来强制停止脚本, 其中 lua-time-limit 选项的值就决定了脚本可以不被打扰地运行的最大毫秒数, 如果这个选项的值是 5000 的话, 那么只有在脚本运行时间超过 5000 毫秒之后, 服务器才会开始接受 SCRIPT KILL 命令, 允许用户终结正在运行的脚本。如果有需要的话, 用户也可以把这个选项的值调小或者调大。



修改和获取配置选项的值



修改选项的值(1):通过给定参数

Redis 提供了三种方法来修改配置 选项的值。

第一种方法是在启动服务器时, 通过给定参数的方式来为配置选项设置值, 格式为:

```
$ redis-server --<option1> <value1> --<option2> <value2> --<option3> <value3> ...
```

比如给定以下参数来启动 Redis 服务器的话, 服务器就会创建 32 个数据库:

```
$ redis-server --databases 32
```

而给定以下参数可以让服务器创建 100 个数据库, 并将服务器的端口设置为 10086:

```
$ redis-server --databases 100 --port 10086
```



修改选项的值(2):通过给定配置文件

修改配置选项值的第二种方法是, 将要修改的配置 选项以及选项的值记录到一个配置文件里面, 并在启动服务器时, 让服务器载入该配置文件。格式为:

```
$ redis-server <path-to-config-file>
```

举个例子, 我们可以创建一个包含以下内容的配置文件 redis.conf :

```
databases 128  
port 10086
```

并在启动服务器时让服务器载入该文件:

```
$ redis-server redis.conf
```

那么启动后的服务器将创建 128 个数据库, 并使用端口 10086 来监听客户端的连接请求。



修改选项的值(3):使用 CONFIG 系列命令

以上两种方法都 **只能在服务器启动时**修改配置选项的值, 通过使用 CONFIG 系列命令, 用户可以在 **服务器运行时**动态修改选项的值, 也可以通过命令获取选项当前的值, 等等。

使用 CONFIG GET 命令可以获取选项当前的值, 该命令的格式为:

CONFIG GET **<option>**

举个例子, 执行以下命令可以返回服务器目前设置的 Lua 脚本的最大正常执行时间:

```
redis> CONFIG GET lua-time-limit
```

```
1) "lua-time-limit"
```

```
2) "5000"
```



修改选项的值(3):使用 CONFIG 系列命令

而通过 CONFIG SET 命令则可以修改配置选项的值, 该命令的格式为:

```
CONFIG SET <option> <value>
```

举个例子, 执行以下命令, 可以将 Lua 脚本的最大正常执行时间从原来的 5000 改为 3000 :

```
redis> CONFIG SET lua-time-limit 3000  
OK
```

执行 CONFIG GET 命令可以检查设置是否已经成功:

```
redis> CONFIG GET lua-time-limit  
1) "lua-time-limit"  
2) "3000"
```



CONFIG SET 的注意事项(1/2)

使用 CONFIG SET 需要注意的一点是，并不是所有配置选项都可以在服务器运行时动态地设置的，有一些配置选项必须在服务器启动时才能设置。

举个例子，因为创建数据库的工作是在服务器启动时进行的，所以数据库的数量必须在启动服务器时指定，在服务器运行的过程中，尝试使用 CONFIG SET 去修改数据库的数量是不可行的：

```
redis> CONFIG SET databases 100  
(error) ERR Unsupported CONFIG parameter: databases
```

另一个例子是服务器使用的监听端口号，这个选项也不能在服务器已经运行的情况下设置：

```
redis> CONFIG SET port 10086  
(error) ERR Unsupported CONFIG parameter: port
```



CONFIG SET 的注意事项(2/2)

CONFIG SET 另外一个需要注意的地方是，CONFIG SET 设置的选项值只会在服务器运行的过程中生效，一旦服务器关机，CONFIG SET 设置的选项值就会丢失。

举个例子，lua-time-limit 选项的默认值为 5000，虽然通过 CONFIG SET lua-time-limit 3000 可以将选项的值改为 3000，但这个修改只会在服务器的本次运行中有效，一旦服务器关闭并重启的话，lua-time-limit 选项的值就会变回默认值 5000。

```
redis> CONFIG SET lua-time-limit 3000
```

```
OK
```

```
redis> CONFIG GET lua-time-limit
```

```
1) "lua-time-limit"
```

```
2) "3000"
```

```
redis> CONFIG GET lua-time-limit # 重启服务器之后执行
```

```
1) "lua-time-limit"
```

```
2) "5000"
```



CONFIG REWRITE 命令



如果服务器在启动时载入了配置文件, 并且在服务器运行的过程中使用 CONFIG SET 修改了配置选项的值, 那么执行 CONFIG REWRITE 命令可以将被修改的配置选项以及它的值写入到配置文件里面。

举个例子, 如果服务器启动时载入了包含以下内容的配置文件:

```
databases 32  
lua-time-limit 5000
```

如果用户在服务器运行的过程中, 执行了 CONFIG SET lua-time-limit 3000 命令, 并且他打算将这一修改记录到配置文件里面的话, 那么他可以执行 CONFIG REWRITE 命令, 将配置文件的内容修改为:

```
databases 32  
lua-time-limit 3000
```

这样服务器在下次启动并载入配置文件时, 就会继续将 Lua 脚本的最大正常运行时间设置为 3000 毫秒。



一些基本的配置选项



配置选项示例

选项	作用	默认值	可在线修改?
port <num>	服务器的监听端口号。	6379	否
timeout <seconds>	在客户端处于空闲状态多久之后, 服务器才会自动断开它。	0 (不主动断开)	是
loglevel <level>	服务器记录日志的级别。	notice	是
databases <num>	数据库的数量。	16	否
requirepass <pswd>	客户端连接服务器的密码。	(空密码)	是
maxmemory <bytes>	可用的最大内存数量。	(不限制)	是
lua-time-limit <ms>	Lua 脚本正常运行的最大时间。	5000	是



关于配置选项的更多信息

Redis 提供了丰富的配置选项，前面列出的只是其中的一小部分，大多数配置选项都和某个特定的功能有关，在之后介绍 RDB 持久化、AOF 持久化、复制、Sentinel 和 集群 等功能时，我们会陆续介绍和这些功能有关的配置选项。

想了解更多可设置的配置选项，或者知道更多关于配置选项的信息，可以参考 Redis 源码附带的配置示例文件 [redis.conf](#)。



复习

回顾本节学习的内容



三种设置配置选项的方法：

1. 服务器启动时通过参数来设置，格式为 `$ redis-server --<option> <value>` 。
2. 服务器启动时通过指定配置文件来设置，格式为 `$ redis-server <path-to-config-file>`
3. 在服务器运行时，通过 `CONFIG SET <option> <value>` 命令来设置。

需要注意的是，并不是所有配置选项都可以使用 `CONFIG SET` 来修改，有些选项只能在服务器启动时设置，并且在不使用 `CONFIG REWRITE` 的情况下，`CONFIG SET` 设置的选项值在服务器关闭之后就会丢失。

使用 `CONFIG GET <option>` 命令可以获取配置选项的当前值。

如果服务器在启动时载入了配置文件，并且在服务器运行的过程中，用户使用 `CONFIG SET` 修改了某些选项的值，那么只要使用 `CONFIG REWRITE` 命令，就可以将 `CONFIG SET` 的修改记录到配置文件里面，以便服务器下次启动时使用。

