

redis配置文件详解

redis.conf配置选项如下：

daemonize 默认情况下，redis不是在后台运行的，如果需要在后台运行，把该项的值更改为yes

pidfile 如以后台进程运行，则需指定一个pid，默认为 /var/run/redis.pid，可以配置其他地址，当运行多个redis服务时，需要指定不同的pid文件和端口

bind 绑定主机IP，默认值为127.0.0.1（注释），生产环境可以设置。

port 监听端口，默认为6379

timeout 超时时间，设置客户端连接时的超时时间，单位为秒。当客户端在这段时间内没有发出任何指令，那么关闭该连接默认为300（秒），0为关闭此设置

loglevel 日志记录等级，有4个可选值，debug，verbose（默认值），notice，warning

debug 记录很多信息，用于开发和测试

verbose 有用的信息，不像debug会记录那么多

notice 普通的verbose，常用于生产环境

warning 只有非常重要或者严重的信息会记录到日志

logfile 日志记录方式，默认值为stdout，标准输出，若后台模式会输出到/dev/null

databases 可用数据库数，默认值为16，默认数据库为0，数据库范围在0-（database-1）之间

save <seconds> <changes> 指出在多长时间之内，有多少次更新操作，就将数据同步到数据文件。这个可以多个条件配合，比如默认配置文件中的设置，就设置了三个条件。

save 900 1 900秒（15分钟）内至少有1个key被改变

save 300 10 300秒（5分钟）内至少有300个key被改变

save 60 10000 60秒内至少有10000个key被改变

rdbcompression 存储至本地数据库（持久化到rdb文件）时是否压缩数据，默认为yes

rdbchecksum yes rdb 文件的校验，如果校验将避免文件格式坏掉，如果不校验将在每次操作文件时要付出校验过程的资源新能，将此参数设置为no，将跳过校验

dbfilename 本地持久化数据库文件名，默认值为dump.rdb

dir redis的工作目录，它会将转储文件存储到这个目录下，并生成一个附加文件，本地数据库存放路径，默认值为 ./ ----->数据库镜像备份的文件放置的路径（持久化文件）

stop-writes-on-bgsave-error yes #后台存储错误停止写。

slaveof <masterip> <masterport> 当本机为从服务时，设置主服务的IP及端口（注释）

masterauth <master-password> 当本机为从服务时，设置主服务的连接密码（注释），当master服务设置了密码保护时(用requirepass制定的密码)

requirepass 连接密码（注释）

slave-serve-stale-data yes 当从主机脱离主的链接时，如果此值为yes当客户端查询从时，回响应客户端，如果是第一次同步回返回一个日期数据或这空值，如果设置为no，则返回“SYNC with master in progress”到INFO and SLAVEOF

slave-read-only yes 从服务器只读（默认）

repl-ping-slave-period 10 从发送ping到主的时间间隔(单位：秒)

repl-timeout 60 批量传输I/O超时和主数据或ping响应超时 默认60s 必须大于repl-ping-slave-period值

repl-disable-tcp-nodelay no 此选项如果是“yes”那么Redis的使用数量较少的TCP数据包和更少的带宽将数据发送到，在从主机上延迟40毫秒（linux kernel中的40毫秒）出现。如果是no将在slave中减少延迟，但是流量使用回相对多一些，如果用多个从主机，此处建议设置成yes

slave-priority 100 从主机的优先级，如果当主主机挂了的时候，将从从主机中选取一个作为其他从机的主，首先优先级的数字最低的将成为主，0是一个特殊的级别，0将永远不会成为主。默认值是100。

maxclients 最大客户端连接数，默认不限制（注释）

maxmemory <bytes> 设置最大内存，达到最大内存设置后，Redis会先尝试清除已到期或即将到期的Key，当此方法处理后，任到达最大内存设置，将无法再进行写入操作。（注释）

appendonly 是否在每次更新操作后进行日志记录, 如果不开启, 可能会在断电时导致一段时间内的数据丢失。因为redis本身同步数据文件是按上面save条件来同步的, 所以所有的数据会在一段时间内只存在于内存中。默认值为no

appendfilename 更新日志文件名, 默认值为appendonly.aof (注释)

appendfsync 更新日志条件, 共有3个可选值。no表示等操作系统进行数据缓存同步到磁盘, always表示每次更新操作后手动调用fsync()将数据写到磁盘, everysec表示每秒同步一次 (默认值)。

no-appendfsync-on-rewrite no #如果为yes, 当BGSAVE或BGREWRITEAOF指令运行时, 即把AOF文件转写到RDB文件中时, 会阻止调用fsync()。

auto-aof-rewrite-percentage 100

auto-aof-rewrite-min-size 64mb #Redis会将AOF文件最初的大小记录下来, 如果当前的AOF文件的大小增加100%并且超过64mb时, 就会自动触发Redis改写AOF文件到RDB文件中, 如果auto-aof-rewrite-percentage为0表示取消自动rewrite功能。

lua-time-limit 5000 #一个Lua脚本最长的执行时间为5000毫秒 (5秒), 如果为0或负数表示无限执行时间。

vm-enabled 是否使用虚拟内存, 默认值为no

vm-swap-file 虚拟内存文件路径, 默认值为/tmp/redis.swap, 不可多个Redis实例共享

vm-max-memory 将所有大于vm-max-memory的数据存入虚拟内存, 无论vm-max-memory设置多小, 所有索引数据都是内存存储的(Redis的索引数据就是keys), 也就是说, 当vm-max-memory设置为0的时候, 其实是所有value都存在于磁盘。默认值为0。

SLOW LOG

slowlog-log-slower-than 10000 #当某个请求执行时间 (不包括IO时间) 超过10000微妙 (10毫秒), 把请求记录在慢日志中, 如果为负数不使用慢日志, 如果为0强制记录每个指令。

slowlog-max-len 128 #慢日志的最大长度是128, 当慢日志超过128时, 最先进入队列的记录会被踢出来, 慢日志会消耗内存, 你可以使用SLOWLOG RESET清空队列回收这些内存。

ADVANCED CONFIG

hash-max-ziplist-entries 512

hash-max-ziplist-value 64 #较小的hash可以通过某种特殊的方式进行编码, 以节省大量的内存空间, 我们指定最大的条目数为512, 每个条目的最大长度为64。

list-max-ziplist-entries 512

list-max-ziplist-value 64 #同上。

zset-max-ziplist-entries 128

zset-max-ziplist-value 64 #同上。

activeremhashing yes #重新哈希the main Redis hash table(the one mapping top-level keys to values), 这样会节省更多的空间。

client-output-buffer-limit normal 0 0 0 #对客户端输出缓冲进行限制可以强迫那些就不从服务器读取数据的客户端断开连接。对于normal client, 第一个0表示取消hard limit, 第二个0和第三个0表示取消soft limit, normal client默认取消限制, 因为如果没有寻问, 他们是不会接收数据的。

client-output-buffer-limit slave 256mb 64mb 60 #对于slave client和MONITOR client, 如果client-output-buffer一旦超过256mb, 又或者超过64mb持续60秒, 那么服务器就会立即断开客户端连接。

client-output-buffer-limit pubsub 32mb 8mb 60 #对于pubsub client, 如果client-output-buffer一旦超过32mb, 又或者超过8mb持续60秒, 那么服务器就会立即断开客户端连接。

Redis官方文档对VM的使用提出了一些建议: 当你的key很小而value很大时, 使用VM的效果会比较好。因为这样节约的内存比较大。当你的key不小时, 可以考虑使用一些非常方法将很大的key变成很大的value, 比如你可以考虑将key,value组合成一个新的value。最好使用linux ext3 等对稀疏文件支持比较好的文件系统保存你的swap文件。vm-max-threads这个参数, 可以设置访问swap文件的线程数, 设置最好不要超过机器的核数。如果设置为0, 那么所有对swap文件的操作都是串行的。可能会造成比较长时间的延迟, 但是对数据完整性有很好的保证

