

译 Redis监控方案

分类: redis

2013-01-30 09:46 4512人阅读 评论(0) 收藏 举报

目录(?)

[-]

1. 内存使用
2. 持久化
3. 主从复制
4. Fork 性能
5. 配置一致
6. 慢日志
7. 监控服务
 1. -Sentinel
 2. -Redis Live
 3. -Redis Faina
8. 数据分布
 1. -Redis-sampler
 2. -Redis-audit
 3. -Redis-rdb-tools

Redis 监控最直接的方法当然就是使用系统提供的 **info** 命令来做了，你只需要执行下面一条命令，就能获得 **Redis** 系统的状态报告。

```
redis-cli info
```

内存使用 **问题：**如果Redis使用的内存超出了可用的物理内存大小，那么Redis很可能会被系统OOM Killer杀掉

如果 **Redis** 使用的内存超出了可用的物理内存大小，那么 **Redis** 很可能会被系统 **OOM Killer** 杀掉。针对这一点，你可以通过 **info** 命令对 **used_memory** 和 **used_memory_peak** 进行监控，为使用内存量设定阈值，并设定相应的报警机制。当然，报警只是手段，重要的是你得预先计划好，当内存使用量过大后，你应该做些什么，是清除一些没用的冷数据，还是把 **Redis** 迁移到更强大的机器上去。

持久化

如果因为你的机器或 **Redis** 本身的问题导致 **Redis** 崩溃了，那么你唯一的救命稻草可能就是 **dump** 出来的 **rdb** 文件了。所以，对 **Redis dump** 文件进行监控也是很重要的。你可以通过对 **rdb_last_save_time** 进行监控，了解最近一次 **dump** 数据操作的时间，还可以通过对 **rdb_changes_since_last_save** 进行监控来知晓如果这时候出现故障，你会丢失多少数据。 **可能出现的问题**

主从复制

如果你设置了主从复制模式，那么你最好对复制的情况是否正常做一些监控，主要是对 **info** 输出中的 **master_link_status** 进行监控，如果这个值是 **up**，那么说明同步正常，如果是 **down**，那么你就要注意一下输出的其它一些诊断信息了。比如下面这些：

```
role:slave
master_host:192.168.1.128
master_port:6379
master_link_status:down
master_last_io_seconds_ago:-1
master_sync_in_progress:0
master_link_down_since_seconds:1356900595
```

Fork 性能 恐怖至极：fork 会造成主线程短时间的卡顿（停止所有读写操作）

当 Redis 持久化数据到磁盘上时，它会进行一次 fork 操作，通过 fork 对内存的 copy on write 机制最廉价的实现内存镜像。但是虽然内存是 copy on write 的，但是虚拟内存表是在 fork 的瞬间就需要分配，所以 fork 会造成主线程短时间的卡顿（停止所有读写操作），这个卡顿时间和当前 Redis 的内存使用量有关。通常 GB 量级的 Redis 进行 fork 操作的时间在毫秒级。你可以通过对 info 输出的 latest_fork_usec 进行监控来了解最近一次 fork 操作导致了多少时间的卡顿。

配置一致 问题：动态修改的配置是不会同步到你的配置文件中去的

Redis 支持使用 CONFIG SET 操作来实现运行时的配置修改，这很方便，但同时也会导致一个问题。就是通过这个命令动态修改的配置，是不会同步到你的配置文件中去的。所以当你因为某些原因重启 Redis 时，你使用 CONFIG SET 做的配置修改就会丢失掉，所以我们最好保证在每次使用 CONFIG SET 修改配置时，也把配置文件一起相应地改掉。为了防止人为的失误，所以我们最好对配置进行监控，使用 CONFIG GET 命令来获取当前运行时的配置，并与 redis.conf 中的配置值进行对比，如果发现两边对不上，就启动报警。

慢日志

Redis 提供了 SLOWLOG 指令来获取最近的慢日志，Redis 的慢日志是直接存在内存中的，所以它的慢日志开销并不大。在实际应用中，我们通过 crontab 任务执行 SLOWLOG 命令来获取慢日志，然后将慢日志存到文件中，并用 Kibana 生成实时的性能图表来实现性能监控。

值得一提的是，Redis 的慢日志记录的时间，仅仅包括 Redis 自身对一条命令的执行时间，不包括 IO 的时间，比如接收客户端数据和发送客户端数据这些时间。另外，Redis 的慢日志和其它数据库的慢日志有一点不同，其它数据库偶尔出现 100ms 的慢日志可能都比较正常，因为一般数据库都是多线程并发执行，某个线程执行某个命令的性能可能并不能代表整体性能，但是对 Redis 来说，它是单线程的，一旦出现慢日志，可能就需要马上得到重视，最好去查一下具体是什么原因了。

监控服务

-Sentinel

Sentinel 是 Redis 自带的工具，它可以对 Redis 主从复制进行监控，并实现主挂掉之后的自动故障转移。在转移的过程中，它还可以被配置去执行一个用户自定义的脚本，在脚本中我们就能够实现报警通知等功能。

-Redis Live

Redis Live 是一个更通用的 Redis 监控方案，它的原理是定时在 Redis 上执行 MONITOR 命令，来获取 Redis 当前正在执行的命令，并通过统计分析，生成 web 页面的可视化分析报表。

-Redis Faina

Redis Faina 是由著名的图片分享应用 instagram 开发的 Redis 监控服务，其原理和 Redis Live 类似，都是对通过 MONITOR 来做的。

数据分布

弄清 Redis 中数据存储分布是一件很难的，比如你想知道哪类型的 key 值占用内存最多。下面是一些工

具，可以帮助你对 Redis 的数据集进行分析。

-Redis-sampler

Redis-sampler 是 Redis 作者开发的工具，它通过采样的方法，能够让你了解到当前 Redis 中的数据的大致类型，数据及分布状况。

-Redis-audit

Redis-audit 是一个脚本，通过它，我们可以知道每一类 key 对内存的使用量。它可以提供的数据有：某一类 key 值的访问频率如何，有多少值设置了过期时间，某一类 key 值使用内存的大小，这很方便让我们能排查哪些 key 不常用或者压根不用。

-Redis-rdb-tools

Redis-rdb-tools 跟 Redis-audit 功能类似，不同的是它是通过对 rdb 文件进行分析来取得统计数据的。

翻译：<http://snmaynard.com/2013/01/22/redis-masterclass-part-two-monitoring-redis/>

⤴ 上一篇 高并发下的 Nginx 优化方案

⤵ 下一篇 Netty 优化技巧

顶

1

踩

1