主从复制

初始状态每一台都是主库

原理:

slave向master发送**sync命令**

master启动后台存盘进程,并收集**所有修改数据命令**

master完成后台存盘后,传送整个数据文件到slave

slave接受数据文件,加载到内存中完成首次完全同步

后续有新数据产生时,master继续收集数据修改命令依次传给slave

注意:

默认都是master

从: 只读

查看

> info replication

role: [master|slave] //角色

master\_host: //主库IP地址

master\_port: //主库端口

master\_link\_status: [up|down] //与主库连接状态

指定主服务器

• 命令行临时: > slaveof 主库ip 端口 //没有”:”号

• 配置文件永久: 282 slaveof 主库ip 端口

临时不做从(还原)

> slaveof no one

带密码的主从配置

主: 设置连接密码

501 requirepass 123456

从:

289 masterauth 123456 #指定主的密码

一主多从

主从从

无高可用: 即使主挂了,中间的从不会生成主,就是down状态

###############################################################################

哨兵服务

*监视master服务器,发现master宕机后,将从服务器升级为主服务器*

选的原则: 数据最接近主的从

1.安装redis软件

不用初始化配置, 关闭redis服务

1. 修改哨兵服务主配置文件

模板文件:

]# vim /etc/sentinel.conf

bind 0.0.0.0

port 26379

sentinel monitor redis

1. 启动哨兵服务

redis-sentinel 配置文件路径

##################################################################################

持久化

两种方式: RDB(默认) AOF

备份类型: **数据** **命令**

优先级: AOF > RDB

----------------------------------------------------------------------------------------------

RDB

redis数据库文件,全称: Redis DataBase

文件: dump.rdb

原理: 按照指定时间间隔,将内存中的数据集**快照**写入硬盘

新开一个进程处理持久化(创建一个子进程来执行持久化,现将数据写入临时文件,持

久化结束后,用这个临时文件替换上次持久化好的文件;过程中主进程不做任何IO

操作)

优缺点:

比较适合大规模数据恢复,且对数据完整性要求不是非常高的场合

意外宕机时,丢失最后一次持久化所有数据

阻塞: 不允许向内存里写数据

save 阻塞写存盘

bgsave 不阻塞写存盘

备份&恢复

• 直接把dump.rdb 拷贝给其他主机

优化配置:

• 数据文件名&路径

254 dbfilename dump.rdb

264 dir /var/lib/redis/6379

• 存盘时间

219 save 900 1 #900秒内有1个key改变存到硬盘

220 save 300 10 #300秒内有10个key改变存到硬盘

221 save 60 10000 #60秒内.........................

######################################################################################

AOF

默认没启用

原理: 追加方式记录所有**写**操作(**命令**)的**文件**

优先级: AOF > RDB (若dump.rdb 和app...同时存在,优先AOF,不会rdb)

若AOF开启了, 不会再读rdb文件

优缺点:

可以灵活设置持久化方式

出现意外宕机时,仅可能丢失1秒的数据

持久化文件的体积通常会大于RDB方式

执行fsync策略时的速度可能会比RDB方式慢

使用cat,vim等命令可查看文件内容

文件: appendonly.aof

• 启用AOF

> config set appendonly yes #开启AOF文件

> config rewrite #写入配置文件(永久开启)

• AOF备份

需恢复的主机需开启AOF,不然不识别AOF文件

直接把AOF文件拷贝给其他主机

• 优化配置

1.文件

673 appendonly yes #开启AOF

677 appendfilename "appendonly.aof" #AOF文件名

702 appendfsync always #实时记录,并完成磁盘同步

703 appendfsync everysec #每秒记录一次,并完成磁盘同步

704 appendfsync no #写入aof,不执行磁盘同步

日志重写

744 auto-aof-rewrite-percentage 100 #再次重写,增长百分比

745 auto-aof-rewrite-min-size 64mb #首次重写触发值

压缩方式: 重复的类似的合并

首次瘦身后, 增长的大小=瘦身后文件大小[数字]%时 再次瘦身

2.修复AOF文件

–把文件恢复到最后一次的正确操作

]#redis-check-aof --fix aof文件名

Continue?[y/N]: y