



链滴

Redis 哨兵模式

作者: [yscxy](#)

原文链接: <https://ld246.com/article/1603503277119>

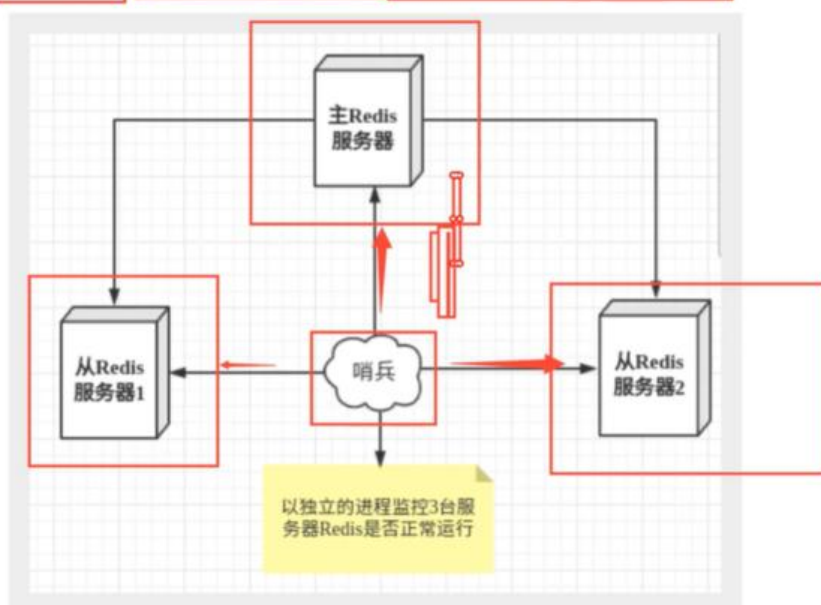
来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)



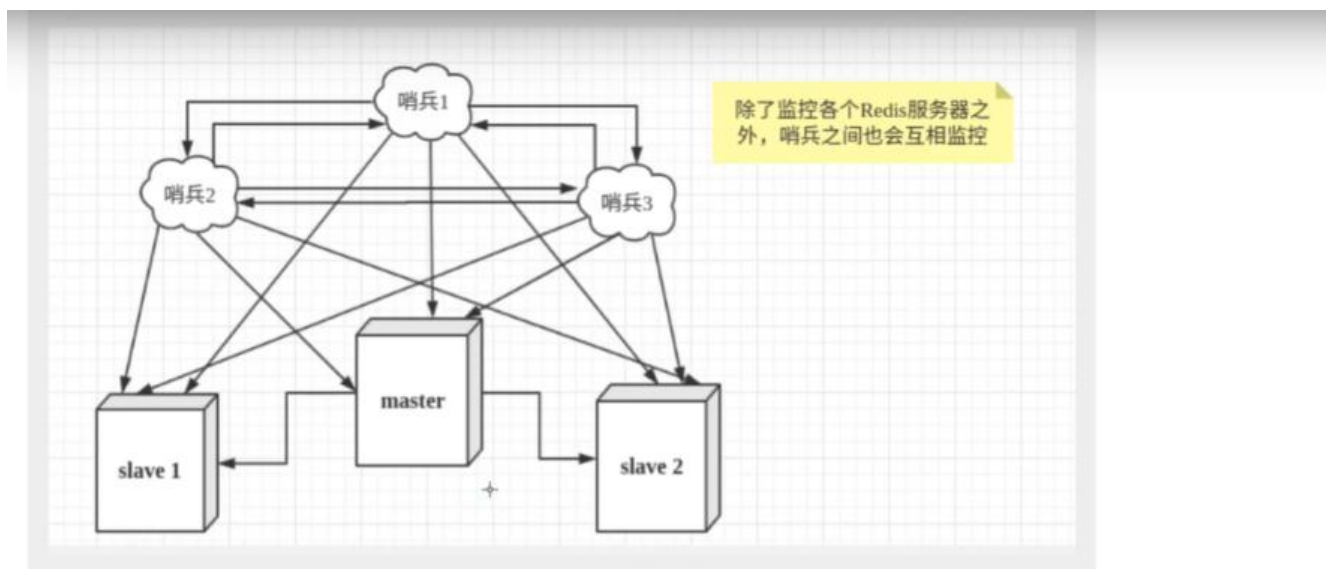
哨兵模式概述

哨兵模式是一种特殊的模式，首先Redis提供了哨兵的命令，哨兵是一个独立的进程，作为进程，它会独立运行。其原理是哨兵通过发送命令等待Redis服务器响应从而监控运行的多个Redis实例。



有时候这种方案也不能满足我们的需求，于是就出现了一种多哨兵的模式

多哨兵模式



假设主服务器宕机，哨兵1先检测到这个结果，系统并不会马上进行failover过程，仅仅是哨兵1主观的认为主服务器不可用，这个现象成为**主观下线**。当后面的哨兵也检测到主服务器不可用，并且数量达到一定值时，那么哨兵之间就会进行一次投票，投票的结果由一个哨兵发起，进行failover[故障转移]操作。切换成功后，就会通过发布订阅模式，让各个哨兵把自己监控的从服务器实现切换主机，这个过程称为**客观下线**。

测试！

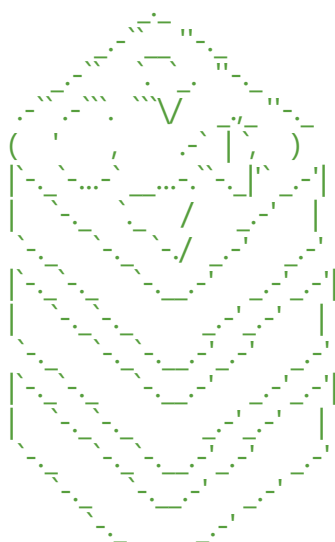
1. 配置哨兵配置文件sentinel.conf

```
sentinel monitor myredis 127.0.0.1 6379 1
```

后面的这个数字1，代表主机挂了，slave投票看让谁阶梯成为主机，票数最多的就成为主机

`/www/server/redis/src/redis-sentinel /www/server/redis/kconfig/sentinel` #启动监控

```
[root@iz2zeclwyjl1sk1uesu6z1z src]# /www/server/redis/src/redis-sentinel /www/server/redis/kconfig/sentinel
3383:X 23 Oct 2020 17:35:07.017 # oO0OoO0OoO0Oo Redis is starting oO0OoO0OoO0Oo
3383:X 23 Oct 2020 17:35:07.017 # Redis version=6.0.8, bits=64, commit=00000000, modif=0, pid=3383, just started
3383:X 23 Oct 2020 17:35:07.017 # Configuration loaded
```



Redis 6.0.8 (00000000/0) 64 bit

Running in sentinel mode
Port: 26379
PID: 3383

<http://redis.io>



```
3383:X 23 Oct 2020 17:35:07.018 # WARNING: The TCP backlog setting of 511 cannot be enforced because /proc/sys/net/core/somaxconn is set to the lower value of 128.
3383:X 23 Oct 2020 17:35:07.022 # Sentinel ID is ee42d4ff53ff3617a6287c1946b44e62ca62183
3383:X 23 Oct 2020 17:35:07.022 # +monitor master myredis 127.0.0.1 6379 quorum 1
3383:X 23 Oct 2020 17:35:07.023 * +slave slave 127.0.0.1:6381 127.0.0.1 6381 @ myredis 127.0.0.1 6379
3383:X 23 Oct 2020 17:35:07.027 * +slave slave 127.0.0.1:6380 127.0.0.1 6380 @ myredis 127.0.0.1 6379
3383:X 23 Oct 2020 17:35:07.027 * +slave slave 127.0.0.1:6380 127.0.0.1 6380 @ myredis 127.0.0.1 6379
```

如果主节点宕机了，这是偶就会从从机中选举出来一个新的主机

```
3383:X 23 Oct 2020 17:35:07.022 # +monitor master myredis 127.0.0.1 6379 quorum 1
3383:X 23 Oct 2020 17:35:07.023 * +slave slave 127.0.0.1:6381 127.0.0.1 6381 @ myredis 127.0.0.1 6379
3383:X 23 Oct 2020 17:35:07.027 * +slave slave 127.0.0.1:6380 127.0.0.1 6380 @ myredis 127.0.0.1 6379
3383:X 23 Oct 2020 17:38:14.757 # +sdown master myredis 127.0.0.1 6379
3383:X 23 Oct 2020 17:38:14.757 # +odown master myredis 127.0.0.1 6379 #quorum 1/1
3383:X 23 Oct 2020 17:38:14.757 # +new-epoch 1
3383:X 23 Oct 2020 17:38:14.757 # +try-failover master myredis 127.0.0.1 6379
3383:X 23 Oct 2020 17:38:14.759 # +vote-for-leader ee42d4ff53ff3617a6287c1946b44e62ca621873 1
3383:X 23 Oct 2020 17:38:14.759 # +elected-leader master myredis 127.0.0.1 6379
3383:X 23 Oct 2020 17:38:14.759 # +failover-state-select-slave master myredis 127.0.0.1 6379
3383:X 23 Oct 2020 17:38:14.842 # +selected-slave slave 127.0.0.1:6381 127.0.0.1 6381 @ myredis 127.0.0.1 6379
3383:X 23 Oct 2020 17:38:14.842 * +failover-state-send-slaveof-noone slave 127.0.0.1:6381 127.0.0.1 6381 @ myredis 127.0.0.1 6379
3383:X 23 Oct 2020 17:38:14.895 * +failover-state-wait-promotion slave 127.0.0.1:6381 127.0.0.1 6381 @ myredis 127.0.0.1 6379
3383:X 23 Oct 2020 17:38:15.772 # +promoted-slave slave 127.0.0.1:6381 127.0.0.1 6381 @ myredis 127.0.0.1 6379
3383:X 23 Oct 2020 17:38:15.772 # +failover-state-reconf-slaves master myredis 127.0.0.1 6379
3383:X 23 Oct 2020 17:38:15.846 * +slave-reconf-sent slave 127.0.0.1:6380 127.0.0.1 6380 @ myredis 127.0.0.1 6379
3383:X 23 Oct 2020 17:38:16.853 * +slave-reconf-inprog slave 127.0.0.1:6380 127.0.0.1 6380 @ myredis 127.0.0.1 6379
3383:X 23 Oct 2020 17:38:16.853 * +slave-reconf-done slave 127.0.0.1:6380 127.0.0.1 6380 @ myredis 127.0.0.1 6379
3383:X 23 Oct 2020 17:38:16.904 # +failover-end master myredis 127.0.0.1 6379
3383:X 23 Oct 2020 17:38:16.904 # +switch-master myredis 127.0.0.1 6379 127.0.0.1 6381
3383:X 23 Oct 2020 17:38:16.904 * +slave slave 127.0.0.1:6380 127.0.0.1 6380 @ myredis 127.0.0.1 6381
3383:X 23 Oct 2020 17:38:16.904 * +slave slave 127.0.0.1:6379 127.0.0.1 6379 @ myredis 127.0.0.1 6381
3383:X 23 Oct 2020 17:38:46.909 # +sdown slave 127.0.0.1:6379 127.0.0.1 6379 @ myredis 127.0.0.1 6381
```

如果刚开始的主机复活了，那也只能当新选举的主机的从机，这就是哨兵模式的规则！