# [Redis集群搭建与使用](https://www.cnblogs.com/wuxl360/p/5920330.html)

一、什么是Redis集群

Redis 集群是一个分布式（distributed）、容错（fault-tolerant）的 Redis 实现， 集群可以使用的功能是普通单机 Redis 所能使用的功能的一个子集（subset）。

Redis 集群中不存在中心（central）节点或者代理（proxy）节点， 集群的其中一个主要设计目标是达到线性可扩展性（linear scalability）。

Redis 集群提供了一种运行 Redis 的方式，其中数据在多个 Redis 节点间自动分区。Redis 集群还在分区期间提供一定程度的可用性，即在实际情况下能够在某些节点发生故障或无法通信时继续运行。但是，如果发生较大故障（例如，大多数主站不可用时），集群会停止运行。



## 一．介绍安装环境与版本

用两台虚拟机模拟6个节点，一台机器3个节点，创建出3 master、3 salve 环境。

redis 采用 redis-3.2.4 版本。

两台虚拟机都是 CentOS ，一台 CentOS7 （IP:192.168.12.10），一台 CentOS7（IP:192.168.12.11） 。

安装过程

**1. 下载并解压**

|  |
| --- |
| #cd /root/software  #wget http://download.redis.io/releases/redis-3.2.4.tar.gz  tar -zxvf redis-3.2.4.tar.gz |

**2. 编译安装**

|  |
| --- |
| #cd redis-3.2.4  #make && make install |

**3. 将 redis-trib.rb 复制到 /usr/local/bin 目录下**

|  |
| --- |
| #cd src  #cp redis-trib.rb /usr/local/bin/ |

**4. 创建 Redis 节点**

首先在 192.168.12.10 机器上 /root/software/redis-3.2.4 目录下创建 redis\_cluster 目录；

|  |
| --- |
| #mkdir redis\_cluster |

在 redis\_cluster 目录下，创建名为7000、7001、7002的目录，并将 redis.conf 拷贝到这三个目录中

|  |
| --- |
| #mkdir 7000 7001 7002  cp redis.conf redis\_cluster/7000  cp redis.conf redis\_cluster/7001  cp redis.conf redis\_cluster/7002 |

分别修改这三个配置文件，修改如下内容

|  |
| --- |
| #端口7000,7002,7003  port 7000  #默认ip为127.0.0.1 需要改为其他节点机器可访问的ip 否则创建集群时无法访问对应的端口，无法创建集群  bind 本机ip  #redis后台运行  daemonize yes  #pidfile文件对应7000,7001,7002  pidfile /var/run/redis\_7000.pid  #开启集群  cluster-enabled yes  #集群的配置 配置文件首次启动自动生成 7000,7001,7002  cluster-config-file nodes\_7000.conf  #请求超时 默认15秒，可自行设置  cluster-node-timeout 15000  #aof日志开启 有需要就开启，它会每次写操作都记录一条日志  appendonly yes |

接着在另外一台机器上（192.168.12.11），的操作重复以上三步，只是把目录改为7003、7004、7005，对应的配置文件也按照这个规则修改即可

1. **启动各个节点**

进入到 /root/software/redis-3.2.4目录

|  |
| --- |
| 第一台机器上执行  #redis-server redis\_cluster/7000/redis.conf  #redis-server redis\_cluster/7001/redis.conf  #redis-server redis\_cluster/7002/redis.conf    另外一台机器上执行  #redis-server redis\_cluster/7003/redis.conf  #redis-server redis\_cluster/7004/redis.conf  #redis-server redis\_cluster/7005/redis.conf |

**6. 检查 redis 启动情况**

|  |
| --- |
| ##一台机器<br>ps -ef | grep redis  root 61020 1 0 02:14 ? 00:00:01 redis-server 127.0.0.1:7000 [cluster]  root 61024 1 0 02:14 ? 00:00:01 redis-server 127.0.0.1:7001 [cluster]  root 61029 1 0 02:14 ? 00:00:01 redis-server 127.0.0.1:7002 [cluster]    netstat -tnlp | grep redis  tcp 0 0 127.0.0.1:17000 0.0.0.0:\* LISTEN 61020/redis-server  tcp 0 0 127.0.0.1:17001 0.0.0.0:\* LISTEN 61024/redis-server  tcp 0 0 127.0.0.1:17002 0.0.0.0:\* LISTEN 61029/redis-server  tcp 0 0 127.0.0.1:7000 0.0.0.0:\* LISTEN 61020/redis-server  tcp 0 0 127.0.0.1:7001 0.0.0.0:\* LISTEN 61024/redis-server  tcp 0 0 127.0.0.1:7002 0.0.0.0:\* LISTEN 61029/redis-server  ----------------------------------------------------------------------------------------  ##另外一台机器  ps -ef | grep redis  root 9957 1 0 02:32 ? 00:00:01 redis-server 127.0.0.1:7003 [cluster]  root 9964 1 0 02:32 ? 00:00:01 redis-server 127.0.0.1:7004 [cluster]  root 9971 1 0 02:32 ? 00:00:01 redis-server 127.0.0.1:7005 [cluster]  root 10065 4744 0 02:38 pts/0 00:00:00 grep --color=auto redis  netstat -tlnp | grep redis  tcp 0 0 127.0.0.1:17003 0.0.0.0:\* LISTEN 9957/redis-server 1  tcp 0 0 127.0.0.1:17004 0.0.0.0:\* LISTEN 9964/redis-server 1  tcp 0 0 127.0.0.1:17005 0.0.0.0:\* LISTEN 9971/redis-server 1  tcp 0 0 127.0.0.1:7003 0.0.0.0:\* LISTEN 9957/redis-server 1  tcp 0 0 127.0.0.1:7004 0.0.0.0:\* LISTEN 9964/redis-server 1  tcp 0 0 127.0.0.1:7005 0.0.0.0:\* LISTEN 9971/redis-server 1 |

安装 ruby

|  |
| --- |
| #yum install ruby  #yum install rubygems  安装 ruby的包（先下载redis-3.3.5.gem文件）  #gem install redis-3.3.5.gem |

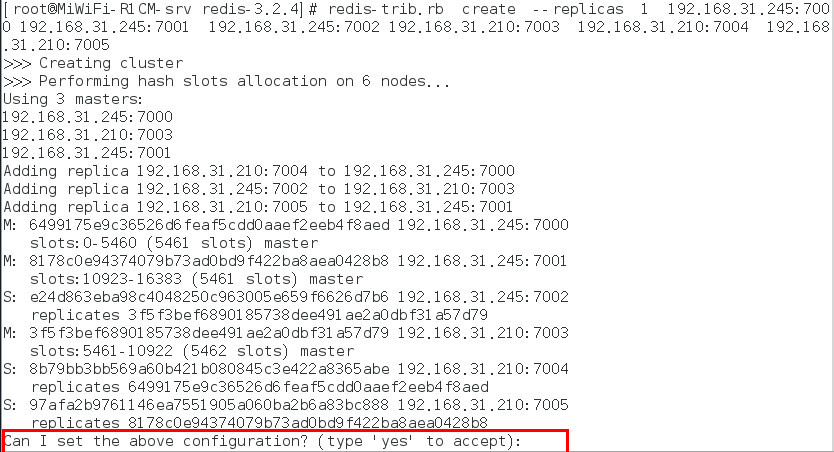
**7.创建集群**

Redis 官方提供了 redis-trib.rb 这个工具，就在解压目录的 src 目录中，第三步中已将它复制到 /usr/local/bin 目录中，可以直接在命令行中使用了。使用下面这个命令即可完成安装。

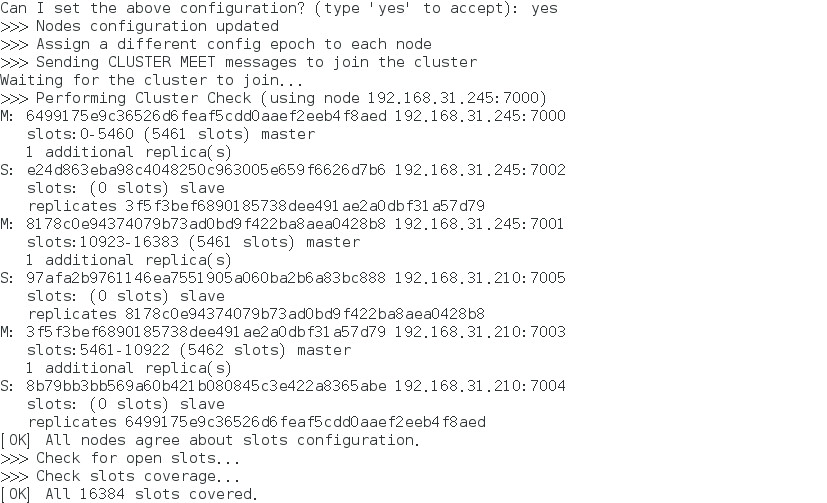
**\*切记不要忘记 【第五步】启动各个节点**

|  |
| --- |
| #redis-trib.rb create --replicas 1 192.168.12.10:6000 192.168.12.10:6001 192.168.12.10:6002 192.168.12.11:6003 192.168.12.11:6004 192.168.12.11:6005 |

之后再运行 redis-trib.rb 命令，会出现如下提示：



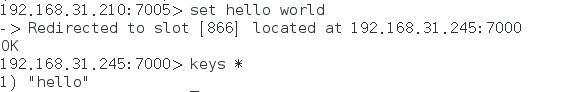
输入 yes 即可，然后出现如下内容，说明安装成功。



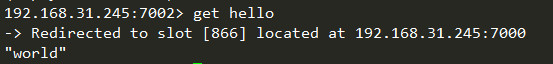
**8. 集群验证**

在第一台机器上连接集群的7002端口的节点，在另外一台连接7005节点，连接方式为 redis-cli -h 192.168.12.10 -c -p 7002  ,加参数 -C 可连接到集群，因为上面 redis.conf 将 bind 改为了ip地址，所以 -h 参数不可以省略。

在7005节点执行命令  set hello world ，执行结果如下：

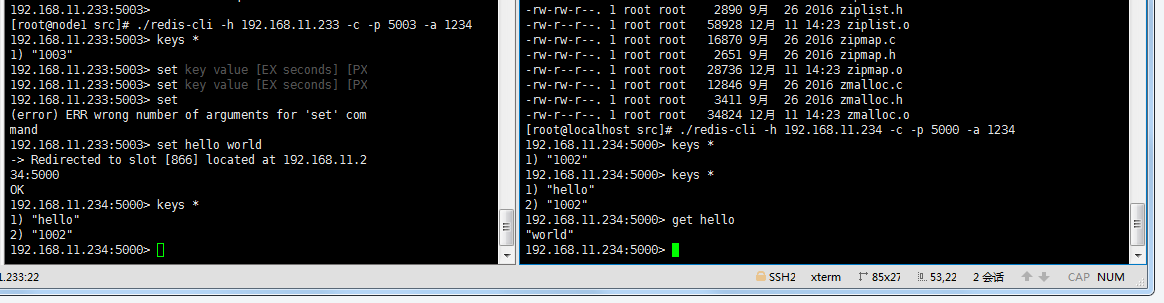


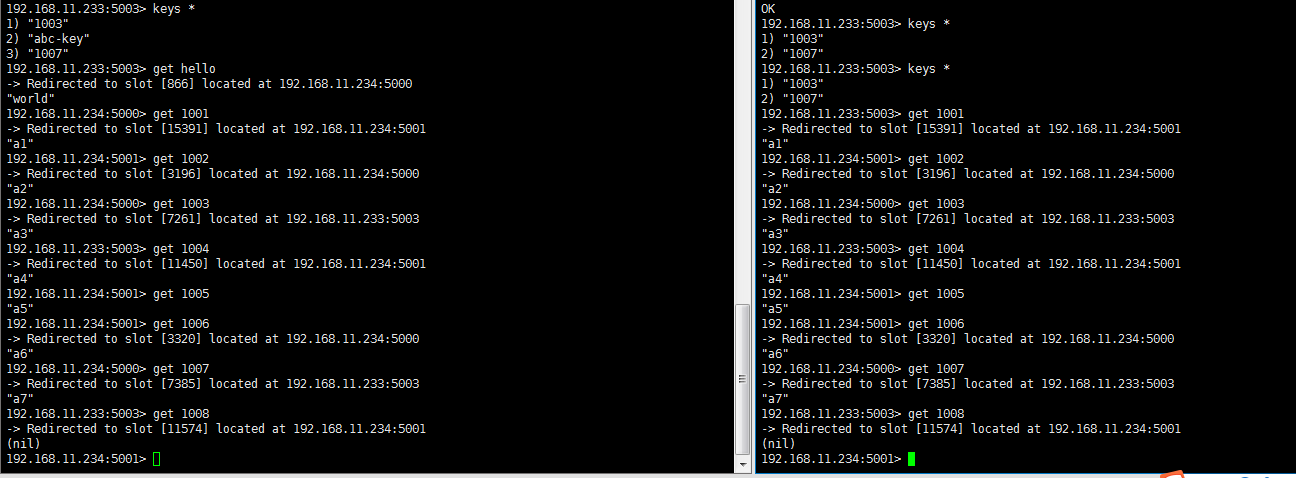
然后在另外一台7002端口，查看 key 为 hello 的内容， get hello  ，执行结果如下：





进入/root/software/redis-3.2.4/src





说明集群运作正常。

|  |
| --- |
| 简单说一下原理  redis cluster在设计的时候，就考虑到了去中心化，去中间件，也就是说，集群中的每个节点都是平等的关系，都是对等的，每个节点都保存各自的数据和整个集群的状态。每个节点都和其他所有节点连接，而且这些连接保持活跃，这样就保证了我们只需要连接集群中的任意一个节点，就可以获取到其他节点的数据。  Redis 集群没有并使用传统的一致性哈希来分配数据，而是采用另外一种叫做哈希槽 (hash slot)的方式来分配的。redis cluster 默认分配了 16384 个slot，当我们set一个key 时，会用CRC16算法来取模得到所属的slot，然后将这个key 分到哈希槽区间的节点上，具体算法就是：CRC16(key) % 16384。所以我们在测试的时候看到set 和 get 的时候，直接跳转到了7000端口的节点。  Redis 集群会把数据存在一个 master 节点，然后在这个 master 和其对应的salve 之间进行数据同步。当读取数据时，也根据一致性哈希算法到对应的 master 节点获取数据。只有当一个master 挂掉之后，才会启动一个对应的 salve 节点，充当 master 。  需要注意的是：必须要3个或以上的主节点，否则在创建集群时会失败，并且当存活的主节点数小于总节点数的一半时，整个集群就无法提供服务了。 |

参考资料为：

http://www.cnblogs.com/wuxl360/p/5920330.html

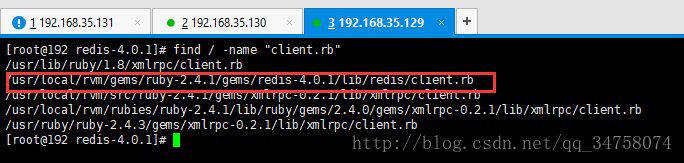
## 二．Redis集群密码设置

**一 修改所有redis集群中的redis.conf文件**   
masterauth 1234   
requirepass 1234

注意所有节点的密码都必须一致，masterauth也要加的。

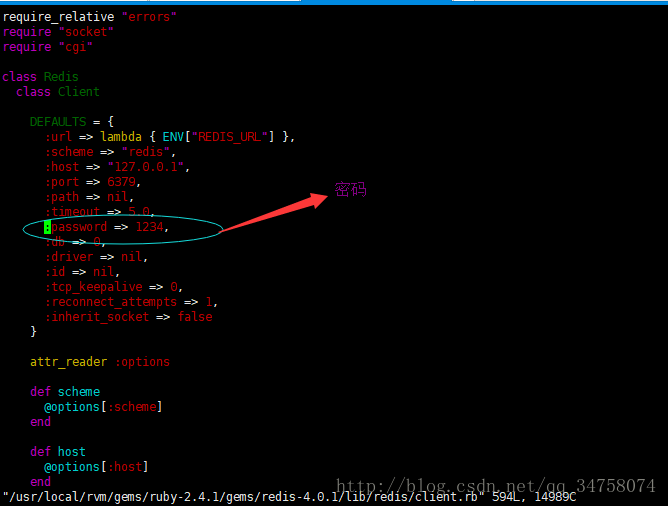
**二 通过指令找到安装的redis在ruby环境中的配置client.rb**

|  |
| --- |
| #find / -name client.rb |



编辑该文件

|  |
| --- |
| #vim /usr/local/rvm/gems/ruby-2.4.1/gems/redis-4.0.1/lib/redis/client.rb |



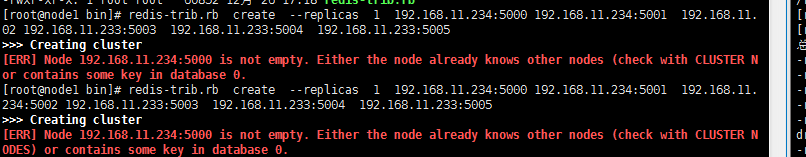
之后重启Redis。

参考资料为：

<https://blog.csdn.net/qq_34758074/article/details/78985283>

## 出错解决方法

**(1)报错截图为:**



进入 /usr/local/bin，执行命令

（每个节点）

|  |
| --- |
| redis-cli -h 192.168.11.234 -p 5001 -a 1234 #####单机  redis-cli -h 192.168.11.234 -c -p 5012 -a 1234 #####集群 |

flushdb #清空当前数据库

|  |
| --- |
| #redis-trib.rb create --replicas 1 192.168.12.10:6000 192.168.12.10:6001 192.168.12.10:6002 192.168.12.11:6003 192.168.12.11:6004 192.168.12.11:6005 |

**(2)启动还是报错，则看下面**



（每个节点）

执行2个命令

Flushall

Cluster reset