**Redis集群搭建与简单使用**

**安装环境与版本：**

两台centos 7.2 服务器 一台机器3个节点，创建出3 master、3 salve 环境。

Redis 版本 redis-4.0.9

Redis01 ip：118.25.123.121

Redis02 ip : 122.152.221.147

**安装过程：**

1. 首先安装jdk，下载好jdk后解压并配置环境变量

Shell > tar zxf jdk-8u101-linux-x64.tar.gz

Shell > vim /etc/profile //添加如下参数

export JAVA\_HOME=/usr/local/java/jdk1.8.0\_101

export CLASSPATH=.:$JAVA\_HOME/lib/dt.jar:$JAVA\_USR/lib/tools.jar

export PATH=$PATH:$JAVA\_HOME/bin

# /usr/local/java/jdk1.8.0\_101 jdk所在路径

1. 下载redis并解压

Shell > cd /usr/local/src

Shell > wget <http://download.redis.io/releases/redis-4.0.9.tar.gz>

Shell > tar zxf redis-4.0.9.tar.gz

1. 编译安装

Shell > yum -y install gcc-c++

Shell > cd /usr/local/src/redis-4.0.9/src

Shell > make && make install

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

Shell > cd /usr/local/src/redis-4.0.9/src

Shell > cp redis-trib.rb /usr/local/bin

1. 创建redis 节点

* 首先在 118.25.123.121机器上 /usr/local/src/redis-4.0.9 目录下创建 redis\_cluster 目录；

Shell > cd /usr/local/src/redis-4.0.9

Shell > 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

Shell > mkdir -p redis\_cluster{7000,7001,7002}

Shell > cp redis.conf redis\_cluster/7000

Shell > cp redis.conf redis\_cluster/7001

Shell > cp redis.conf redis\_cluster/7002

* 分别修改这三个配置文件，修改如下内容:

port 7000 //端口7000,7001,7002

bind 172.17.0.46 //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

daemonize yes

pidfile /var/run/redis\_7000.pid //pidfile文件对应7000,7001，7002

cluster-enabled yes //开启集群 把注释#去掉

cluster-config-file nodes\_7000.conf //集群的配置

cluster-node-timeout 15000 //请求超时 默认15秒，可自行设置

appendonly yes //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

* 接着在另外一台机器上（122.152.221.147），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

第一台机器上执行:

Shell > cd /usr/local/src/redis-4.0.9

Shell > redis-server redis\_cluster/7000/redis.conf

Shell >redis-server redis\_cluster/7001/redis.conf

Shell >redis-server redis\_cluster/7002/redis.conf

另外一台机器上执行:

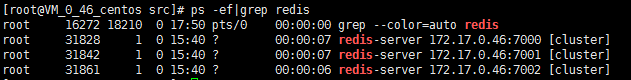
Shell >redis-server redis\_cluster/7003/redis.conf

Shell >redis-server redis\_cluster/7004/redis.conf

Shell >redis-server redis\_cluster/7005/redis.conf

1. 检查 redis 启动情况

Shell > ps -ef|grep redis



1. 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。

Shell > redis-trib.rb create --replicas 1 172.17.0.46:7000 172.17.0.46:7001 172.17.0.46:7002 172.17.0.33:7003 172.17.0.33:7004 172.17.0.33:7005

其中，前三个 ip:port 为第一台机器的节点，剩下三个为第二台机器。

等等，出错了。这个工具是用 ruby 实现的，所以需要安装 ruby。安装命令如下：

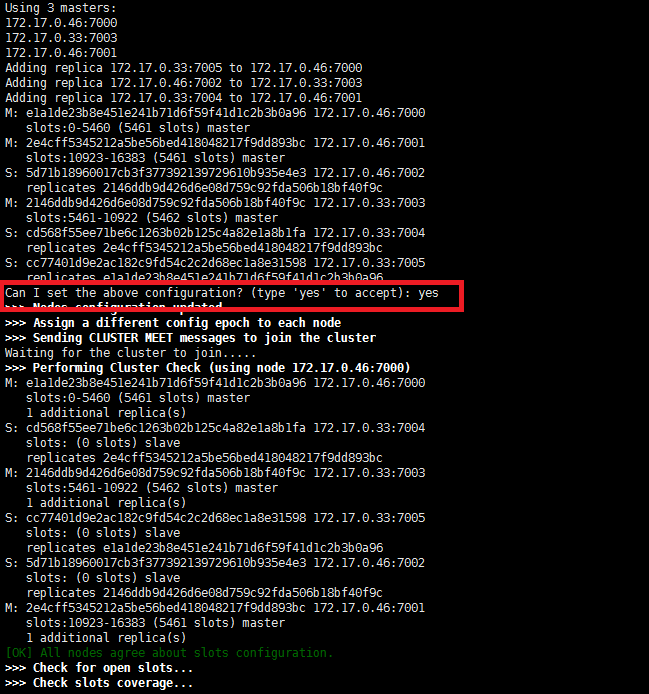
Shell > yum -y install ruby ruby-devel rubygems

Shell > gem install redis

提示有报错 ruby版本需要2.2.2以上 解决方法：

<https://www.cnblogs.com/ding2016/p/7903147.html>

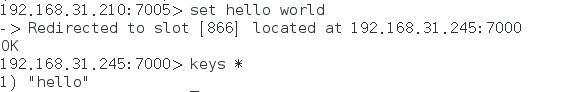
之后再运行 redis-trib.rb 命令，会出现如下提示： //如图，中间输入yes



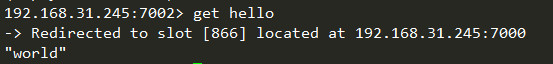
8. 集群验证

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 172.17.0.46 -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：



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。

**相关参考链接：**

**安装redis集群:**

<https://www.cnblogs.com/wuxl360/p/5920330.html>

<http://www.redis.cn/topics/cluster-tutorial.html>

**卸载或重新安装Redis集群以及报错解决方法:**

<https://www.cnblogs.com/tommy-huang/p/6733967.html>

<https://blog.csdn.net/wudalang_gd/article/details/52153571>

<https://blog.csdn.net/qiushisoftware/article/details/78837855>