Redis Cluster集群安装部署

董仕应

## 单机安装

**redis依赖库包**：gcc，tcl，tcl-devel（yum install 全部安装即可）

cd /usr/local

wget <http://download.redis.io/releases/redis-3.2.10.tar.gz>

tar -zxvf redis-3.2.10.tar.gz

cd redis-3.2.10

**make**

**make install**

此时启动文件会跑到 /usr/local/bin 路径下

[在redis-3.2.10目录下新建data和bin目录

复制/usr/local/bin 路径下的启动文件到/usr/local/redis-3.2.10/bin下]

[以上两个步骤可以忽略，对应配置文件即可]

**修改配置文件**

vim /usr/local/redis-3.2.10/redis.conf  一般有几个参数要改

**bind** 192.168.18.128 #监听在哪个ip上，默认是127.0.0.1，如果要做集群的话，改成主机的IP,否则创建集群会失败。

**port** 7001      #监听在哪个端口，默认是6379，如果一台机器装了多个redis，那么彼此间不能冲突

**daemonize** no    #redis的守护进程，默认是no，no状态下，启动redis

需要添加 **&** 让其在后台运行（redis-server

/usr/local/redis-3.2.10/redis.conf **&**），

最好改成yes，不需要加&启动

**cluster-enabled** no  #redis集群模式，默认是no，做集群的话要改成yes

**pidfile** /var/run/redis\_7001.pid

**cluster-config-file**

/usr/local/redis-3.2.10/data/nodes-7001.conf #集群配置文件

**dir** /usr/local/redis-3.2.10/data 数据保存路径

**cluster-node-timeout** 5000    #集群超时时间

**appendonly** yes   #默认no，是否在每次更新操作后进行日志记录，如果不开启，可能会在断电时导致一段时间内的数据丢失

**启动redis**

**redis-server /usr/local/redis-3.2.10/redis.conf** 【**&】**

**停止redis**

**redis-cli shutdown**

使用redis，测试是否正常

redis-cli -h 192.168.18.128 -p 7001

192.168.18.128:7001>ping

PONG

## 集群安装

在阿里云中需要使用私网IP启动redis

集群正常工作至少需要3个主节点，在这里我们创建6个redis节点，其中三个为主节点，三个为从节点，对应的redis节点的ip和端口对应关系如下,创建集群时会自动分配对应关系。

创建相关目录，复制/usr/local/redis-3.2.10目录并改名到这些节点目录

/usr/local/redis\_cluster/7001/

192.168.146.129:7001

192.168.146.130:7002

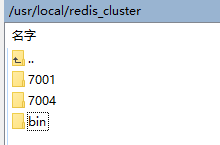
192.168.146.131:7003

192.168.146.129:7004

192.168.146.130:7005

192.168.146.131:7006

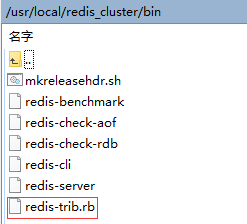
其中前三个为主节点，后三个为从节点（也可以分开6台服务器安装）



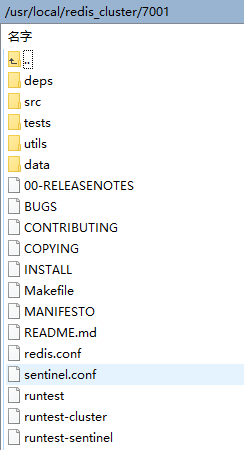
创建bin目录，并将启动文件以及集群创建文件从节点目录下的src目录中复制过来(如果有，重点是集群创建文件)

cd /usr/local/redis-3.2.10/src

cp mkreleasehdr.sh redis-benchmark redis-check-aof redis-check-dump redis-cli redis-server **redis-trib.rb** /usr/local/redis\_cluster/bin



其中一个节点7001的目录结构如下：



安装步骤：

1、三台机器分别安装redis

2、主节点安装集群依赖包**ruby （必须）**

**yum -y install ruby ruby-devel rubygems rpm-build**

安装与redis的关联

**gem install redis**

没网情况下，安装依赖包为

rpm -ivh compat-readline5-5.2-17.1.el6.x86\_64.rpm

rpm -ivh ruby-libs-1.8.7.374-4.el6\_6.x86\_64.rpm

rpm -ivh ruby-1.8.7.374-4.el6\_6.x86\_64.rpm

rpm -ivh ruby-irb-1.8.7.374-4.el6\_6.x86\_64.rpm

rpm -ivh ruby-rdoc-1.8.7.374-4.el6\_6.x86\_64.rpm

rpm -ivh rubygems-1.3.7-5.el6.noarch.rpm

gem install -l /rpm/redis-3.2.0.gem

以上默认安装**ruby**安装版本(1.8.7)过低，不能关联redis

[root@localhost ~]# gem install redis

ERROR: Error installing redis:

redis requires Ruby version >= 2.2.2.

[root@localhost ~]#

解决办法是 先安装**rvm**，再把ruby版本提升至2.3.3

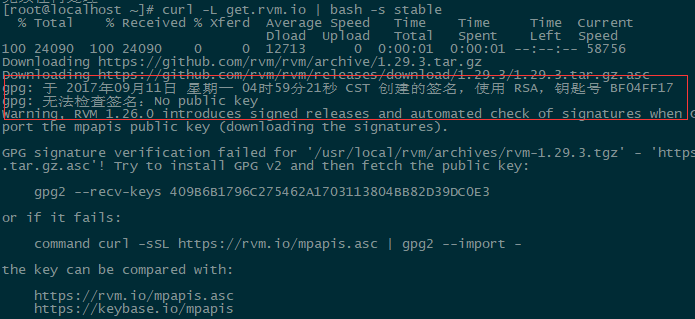
1.安装curl

**sudo yum install curl**

2.安装rvm

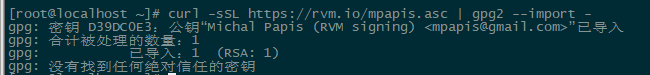
**curl -sSL get.rvm.io | bash -s stable**

可能安装失败，因为签名在解密的时候，不能使用gpg2加密算法进行解密：



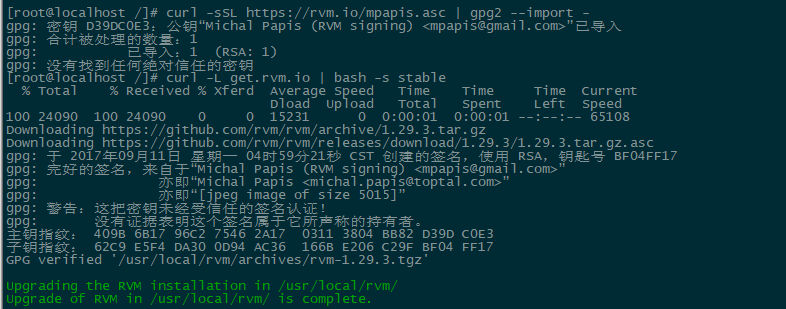
需要导入加密算法：

**curl -sSL https://rvm.io/mpapis.asc | gpg2 --import -**



再次安装RVM，此时应该只剩下警告：

**curl -sSL get.rvm.io | bash -s stable**



命令窗口激活rvm：

**source /etc/profile.d/rvm.sh**

查看rvm版本，看安装是否成功

**rvm -v**

查看网络上rvm库中已知的ruby版本

rvm list known

选择安装其中一个版本

rvm install 2.2[.7]

使用安装的这个版本

rvm use 2.2.7

删除一个版本

rvm remove 1.8.7

最后关联redis

**gem install redis**

3、修改各节点配置文件redis.conf，主要修改以下几行

bind 192.168.18.128

port 7001

daemonize yes

cluster-enabled yes

pidfile /var/run/redis\_7001.pid

dir /usr/local/redis\_cluster/7001/data

cluster-config-file nodes-7001.conf

cluster-node-timeout 5000

appendonly yes

**#注意IP地址和端口，要改成对应的**

**#集群要求相应端口开放，并且相应集总端口也要开放（+10000**），如17001

#当然也可以关闭防火墙

4、分别启动这6个redis节点

/usr/local/bin/redis-server /usr/local/redis\_cluster/7001/redis.conf

/usr/local/bin/redis-server /usr/local/redis\_cluster/7002/redis.conf

/usr/local/bin/redis-server /usr/local/redis\_cluster/7003/redis.conf

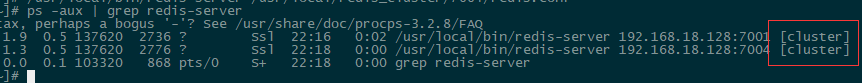
/usr/local/bin/redis-server /usr/local/redis\_cluster/7004/redis.conf

/usr/local/bin/redis-server /usr/local/redis\_cluster/7005/redis.conf

/usr/local/bin/redis-server /usr/local/redis\_cluster/7006/redis.conf

查看进程启动情况，看是否是redis-server ip:port [cluster]的进程

ps -aux | grep redis-server



#注意，进程必须带有[cluster]字样，否则就不是集群模式，查看redis.conf文件的cluster-enabled是否为yes

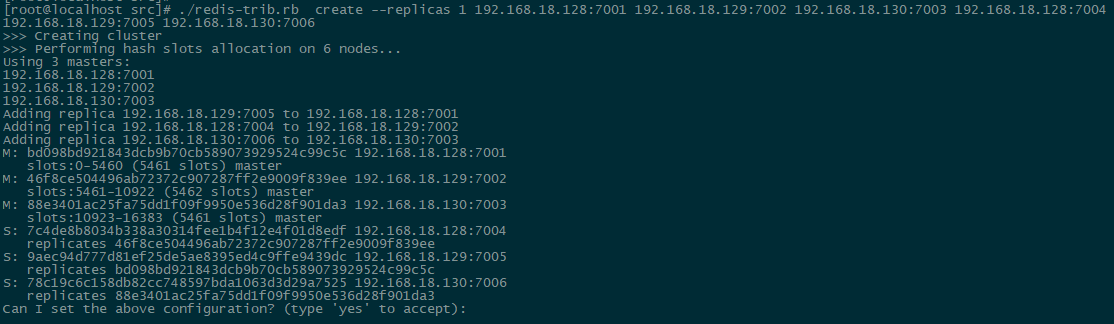
此时集群还未创建成功，测试报错

redis-cli **-c** -h 192.168.18.128 -p 7001

**(error) CLUSTERDOWN Hash slot not served（不提供集群的散列槽）**

5、创建集群（在任意节点）

/usr/local/redis\_cluster/bin/redis-trib.rb create --replicas 1 192.168.18.128:7001 192.168.18.129:7002 192.168.18.130:7003 192.168.18.128:7004 192.168.18.129:7005 192.168.18.130:7006



输入yes，没报错的话，集群创建成功

如果阻塞在 waiting for cluster join …

说明集群搭建不正确，或者端口不通，比如集总端口忘了打开

停止一个节点命令 **redis-cli -h 172.17.174.100 -p 7000 shutdown**

6、测试使用

redis-cli **-c** -h 192.168.18.128 -p 7001

192.168.18.128:7001>set jack 30

OK

在节点上查看是否同步数据过来

redis-cli **-c** -h 192.168.18.129 -p 7002

192.168.18.129:7002>get jack

#**-c**启动为cluster模式（必须，否则数据不会同步，**(error) MOVED 5798 192.168.18.129:7002**），-h指定host，-p指定端口 #不一定要在那台服务器上使用那个redis，可以在一台服务器上使用所有redis，在192.168.146.129上也可以redis-cli -c -h 192.168.146.130 -p 7002