# 1、环境

192.168.0.11

192.168.0.12

192.168.0.13

每台服务器1主1从，共3主3从

相关安装包存储路径：/root/svr/

# 2、部署

### 2.1下载

wget <http://download.redis.io/releases/redis-3.2.9.tar.gz>

tar xvf redis-3.2.9.tar.gz

cd redis-3.2.9

### 2.2安装

make install PREFIX=/root/svr/redis-3.2.9 安装

### 2.3配置

cd /usr/local/redis-3.2.9

创建集群配置文件夹：mkdir cluster-conf

cd cluster-conf

创建集群端口文件夹：mkdir 7001 mkdir 7002

cd 7001

复制配置文件：cp /root/svr/redis-3.2.9/redis.conf ./

Redis的log及持久化文件建议存储到磁盘空间较大的目录，本次存储路径：/root/svr/redis-cluster/

修改配置文件：vi redis.conf

|  |  |
| --- | --- |
| port 7001  logfile "/root/svr/redis-3.2.9/cluster-conf/7001/redis.log"  dir /root/svr/redis-cluster/7001/ **#事先创建好**  cluster-enabled yes  cluster-config-file nodes.conf  cluster-node-timeout 5000  appendonly yes  bind 0.0.0.0 | #端口号  port  #指定了记录日志的文件。  logfile  #数据目录，数据库的写入会在这个目录。rdb、aof文件也会写在这个目录  dir  #是否开启集群  cluster-enabled  ##集群配置文件的名称，每个节点都有一个集群相关的配置文件，持久化保存集群的信息。这个文件并不需要手动配置，这个配置文件有Redis生成并更新，每个Redis集群节点需要一个单独的配置文件，请确保与实例运行的系统中配置文件名称不冲突  cluster-config-file nodes.conf  #节点互连超时的阀值。集群节点超时毫秒数  cluster-node-timeout  #默认redis使用的是rdb方式持久化，这种方式在许多应用中已经足够用了。但是redis如果中途宕机，会导致可能有几分钟的数据丢失，根据save来策略进行持久化，Append Only File是另一种持久化方式，可以提供更好的持久化特性。Redis会把每次写入的数据在接收后都写入 appendonly.aof 文件，每次启动时Redis都会先把这个文件的数据读入内存里，先忽略RDB文件。  appendonly |

复制redis.conf至7002并修改端口及存储路径

**scp其他服务器**

scp -r redis-3.2.9/ root@192.168.0.13:/root/svr/

### 2.4启动

/root/svr/redis-3.2.9/bin/redis-server /root/svr/redis-3.2.9/cluster-conf/7002/redis.conf &

### 2.5创建集群

./redis-trib.rb create --replicas 1 192.168.0.11:7001 192.168.0.12:7001 192.168.0.13:7001 192.168.0.11:7002 192.168.0.12:7002 192.168.0.13:7002

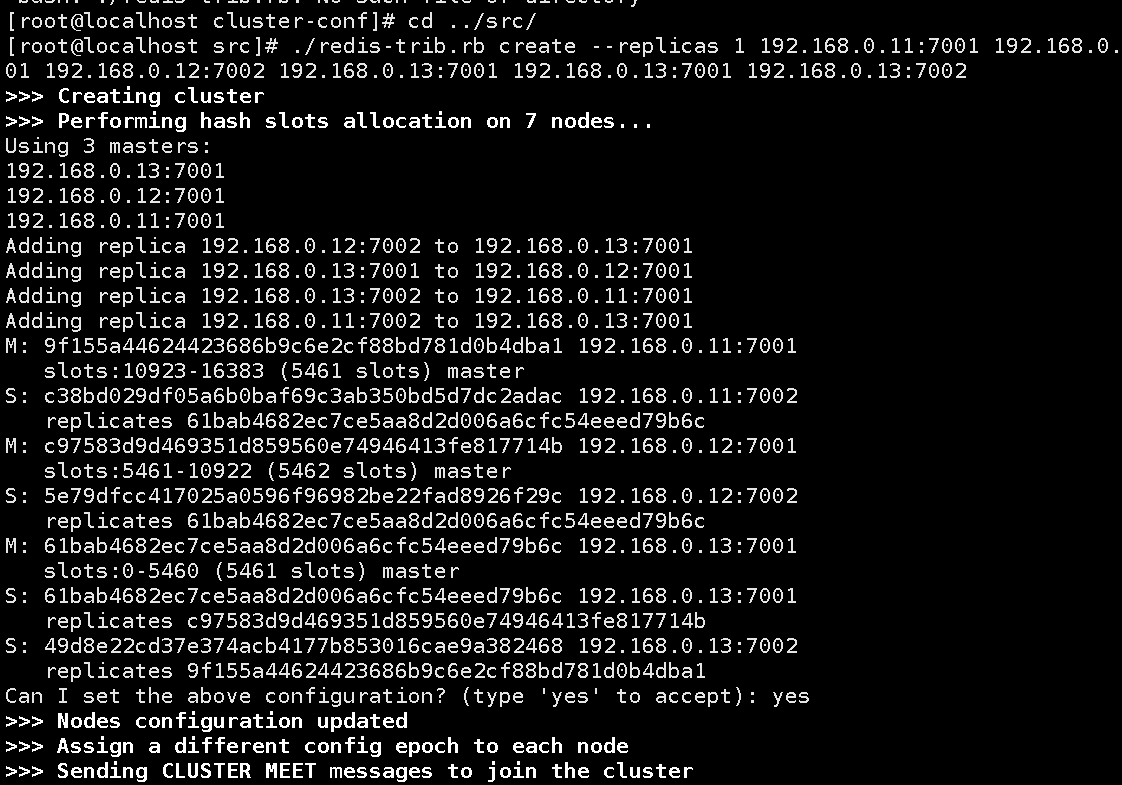
以上命令的意思就是让 redis-trib 程序创建一个包含三个主节点和三个从节点的集群。

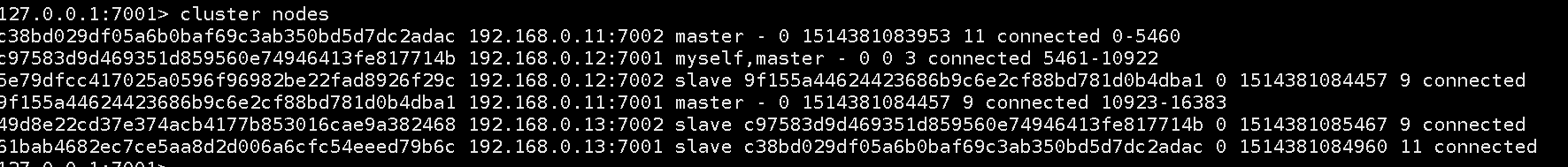
命令的意义如下：

1、给定 redis-trib.rb 程序的命令是 create ， 这表示我们希望创建一个新的集群。

2、选项 --replicas 1 表示我们希望为集群中的每个主节点创建一个从节点（百分比 选举master按先后顺序）。

3之后跟着的其他参数则是实例的地址列表， 我们希望程序使用这些地址所指示的实例来创建新集群。





含义如下:

节点ID

IP:端口

标志: master, slave, myself, fail

如果是个从节点, 这里是它的主节点的NODE ID

集群最近一次向节点发送 PING 命令之后， 过去了多长时间还没接到回复。.

节点最近一次返回 PONG 回复的时间。

本节点的网络连接情况

节点目前包含的槽：例如 127.0.0.1:7001 目前包含号码为 5960 至 10921 的哈希槽(master)。

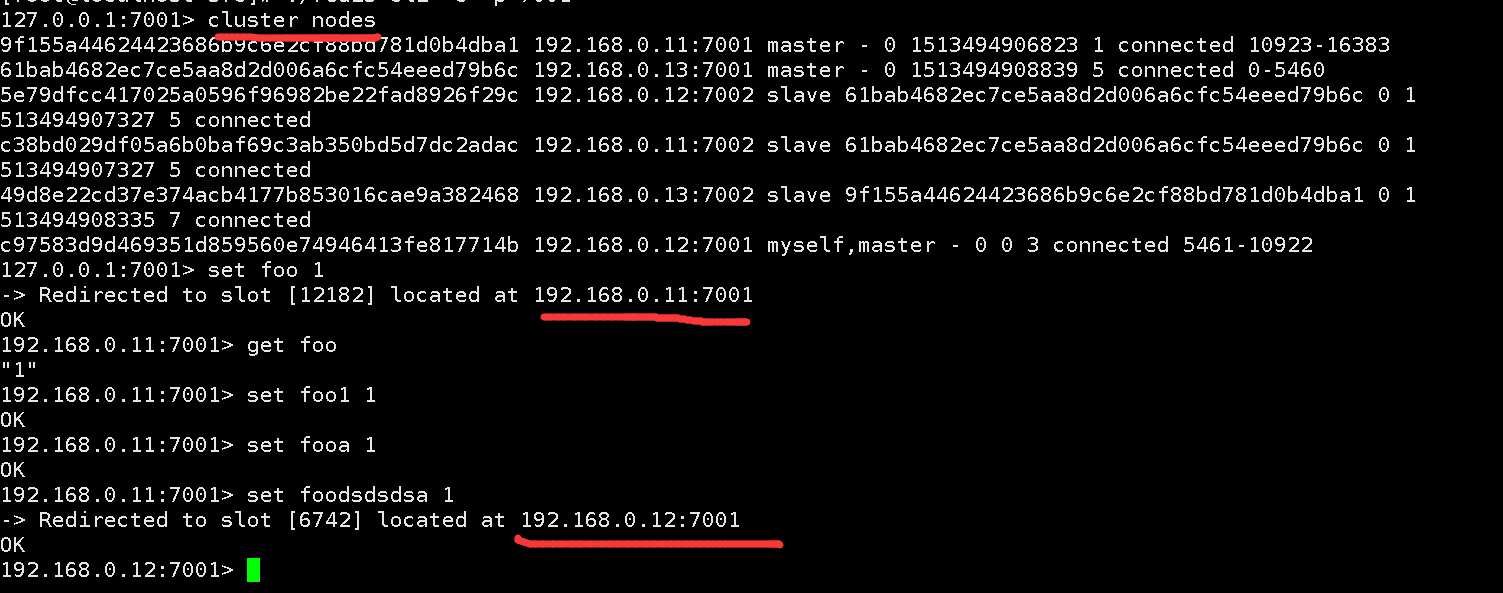
### 2.6查看

ps -ef |grep redis

或者netsta -tnlp |grep redis

./redis-cli -c -p 7001

表示安装成功了



### 2.7其他命令（安装过程不需要执行）：

//增加

./redis-trib.rb add-node 127.0.0.1:7006 127.0.0.1:7000

//从节点（masterid 和被加的节点）

./redis-trib.rb add-node --slave masterid 192.168.0.11:7002

//移除

./redis-trib del-node 127.0.0.1:7000 `<node-id>`

关闭服务：./redis-cli -h 192.168.0.11 -p 7001 shutdown

删除：rm -rf /root/svr/redis-cluster/7001/\*

# 问题

## 一、在执行./redis-trib.rb create报

/usr/bin/env: ruby: No such file or directory

解决：yum install ruby

## 在执行./redis-trib.rb create报

./redis-trib.rb:24:in `require': no such file to load -- rubygems (LoadError)

from ./redis-trib.rb:24

解决：yum install rubygems

## 三、在执行./redis-trib.rb create报

/usr/lib/ruby/site\_ruby/1.8/rubygems/custom\_require.rb:31:in `gem\_original\_require': no such file to load -- redis (LoadError)

from /usr/lib/ruby/site\_ruby/1.8/rubygems/custom\_require.rb:31:in `require'

解决：gem install redis

## 如果telnet 192.168.0.xx 7001

如果telnet不通，但是ps -ef|grep redis可以看到服务或者本地客户端可以连接，远程客户端连接不了。这个时候就要修改redis.conf参数了

vim 7001(2)/redis.conf

注释bind或者改为bind 0.0.0.0



关掉受保护模式

