**Redis集群部署**

**本例共6台虚拟机（192.168.91.[11-16]）3主3从**

# **1、下载Redis4.0.2安装包**

#cd /usr/local/src/

#wget http://download.redis.io/releases/redis-4.0.2.tar.gz



# **2、解压、编译、安装redis-4.0.2**

#tar -zxvf redis-4.0.2.tar.gz -C /usr/local/

#ln -sv /usr/local/redis-4.0.2 /usr/local/redis-cluster

#cd /usr/local/redis-cluster

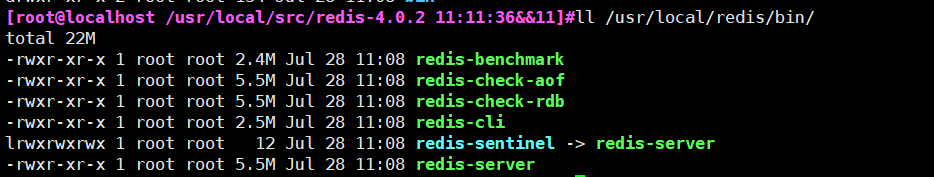
#make && make install USE\_SYSTEMD=yes

通过编译安装的Redis需要设置启动脚本来实现systemd启动，从而可以开机自启编译的时候，需要加入对systemd的支持

# libsystemd-dev on Debian/Ubuntu or systemd-devel on CentOS

yum install -y systemd-devel

make install USE\_SYSTEMD=yes PREFIX=/usr/local/redis/



否则使用systemd启动后，Redis日志中会有如下的报错：

systemd supervision requested or auto-detected, but Redis is compiled without libsystemd support!

# **3、创建文件夹**

(集群中 Redis 节点的端口号为 9001-9006 ，端口号即集群下各实例文件夹。数据存放在 端口号/data 文件夹中)

#mkdir -pv /usr/local/redis-cluster/data

# **4、复制执行脚本**

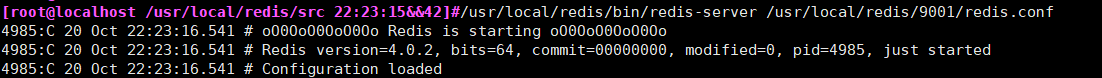
(在 /usr/local/redis-cluster 下创建 bin 文件夹，用来存放集群运行脚本，并把安装好的 Redis 的 src 路径下的运行脚本拷贝过来)

|  |
| --- |
| # mkdir /usr/local/redis-cluster/bin  # cd /usr/local/redis-cluster/src && cp mkreleasehdr.sh redis-sentinel redis-check-rdb redis-benchmark redis-check-aof redis-check-dump redis-cli redis-server redis-trib.rb /usr/local/redis-cluster/bin  # ll /usr/local/redis-cluster/bin/  #cp -a /usr/local/redis-cluster/\* /usr/local/redis-cluster/9001/ |

|  |
| --- |
| #vim /usr/local/redis-cluster/9001/redis.conf  port 9001  daemonize yes  bind 192.168.91.11  dir /usr/local/redis-cluster/9001/data/  pidfile /var/run/redis\_9001.pid  cluster-enabled yes  cluster-config-file redis.conf  cluster-node-timeout 15000  appendonly yes |

# **5、启动**

#/usr/local/redis/bin/redis-server /usr/local/redis-cluster/9001/redis.conf



查一下是否启动成功：ps -el | grep redis

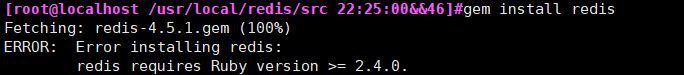
1. **安装集群所需软件**

由于 Redis 集群需要使用 ruby 命令，所以我们需要安装 ruby 和相关接口。

yum install ruby -y

yum install rubygems -y

gem install redis



**6.1（在线安装，速度快方便）**

6.1.1卸载当前ruby（当前使用centos系统）

|  |
| --- |
| #yum remove ruby |

6.1.2下载ruby稳定版

|  |
| --- |
| #cd /usr/local/src  #wget <https://cache.ruby-lang.org/pub/ruby/3.0/ruby-3.0.2.tar.gz>  #tar zxf ruby-3.0.2.tar.gz  #cd ruby-3.0.2  #./configure --prefix=/usr/local/ruby  #make  #make install |

6.1.3查看安装后的版本

|  |
| --- |
| #/usr/local/ruby/bin/ruby -v |

6.1.4 设置环境变量

|  |
| --- |
| #vi /etc/profile  export PATH=$PATH:/usr/local/ruby/bin:  #source /etc/profile  #echo $PATH |

6.1.5镜像加速

|  |
| --- |
| #gem sources --add https://gems.ruby-china.com/ --remove https://rubygems.org/  #gem sources -l |

6.1.6安装gem-redis

|  |
| --- |
| #gem install redis |

**6.2（在线安装，很慢。）**

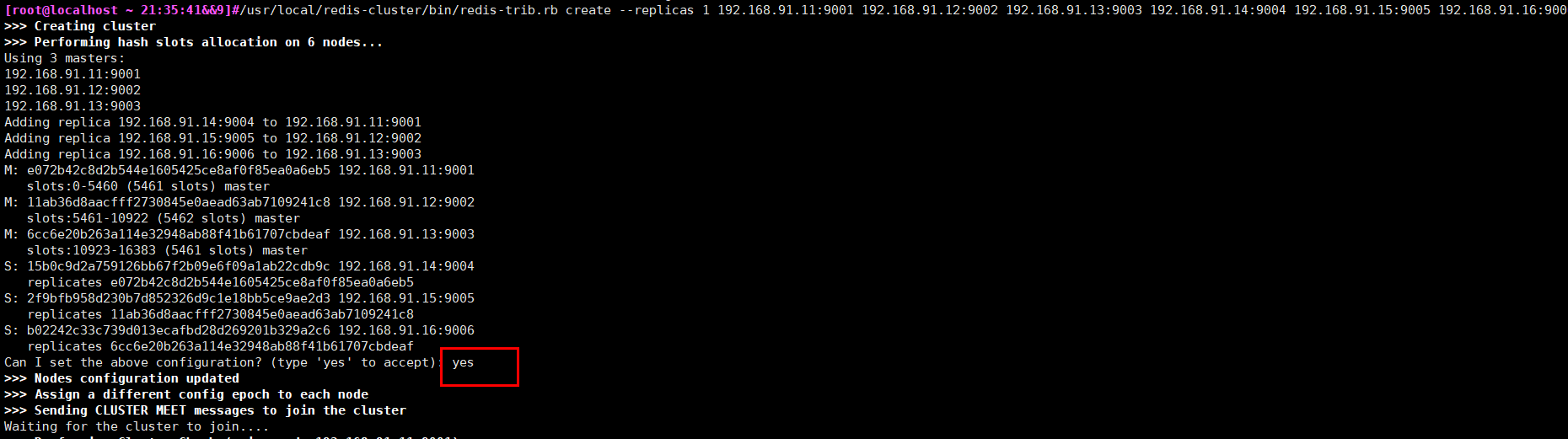
|  |
| --- |
| 解决方案：先安装rvm，再把ruby版本提升至2.3.0   1. 安装curl ：yum install curl 2. 安装RVM：curl -L get.rvm.io | bash -s stable     #gpg2 --keyserver hkp://pool.sks-keyservers.net --recv-keys 409B6B1796C275462A1703113804BB82D39DC0E3  **gpg --keyserver hkp://keys.gnupg.net --recv-keys 409B6B1796C275462A1703113804BB82D39DC0E3**  这个没试过  #curl -sSL https://rvm.io/mpapis.asc | gpg2 --import -  #curl -sSL https://rvm.io/pkuczynski.asc | gpg2 --import -  三个命令试着来  一般是最后一个成功    成功后再次运行 curl -L get.rvm.io | bash -s stable    支持curl成功  source /usr/local/rvm/scripts/rvm   1. 查看rvm库中已知的ruby版本：   rvm list known     1. 安装一个ruby版本：   rvm install 2.6.3   1. 使用一个ruby版本：rvm use 2.6.3 2. 设置默认版本：ruby --version 3. 卸载一个已知版本：rvm remove 2.0.0 4. 安装redis 5. [root@localhost ~ 21:33:20&&6]#gem install redis   **Fetching redis-4.5.1.gem**  **Successfully installed redis-4.5.1**  **Parsing documentation for redis-4.5.1**  **Installing ri documentation for redis-4.5.1**  **Done installing documentation for redis after 1 seconds**  **1 gem installed** |

# **7 创建集群**

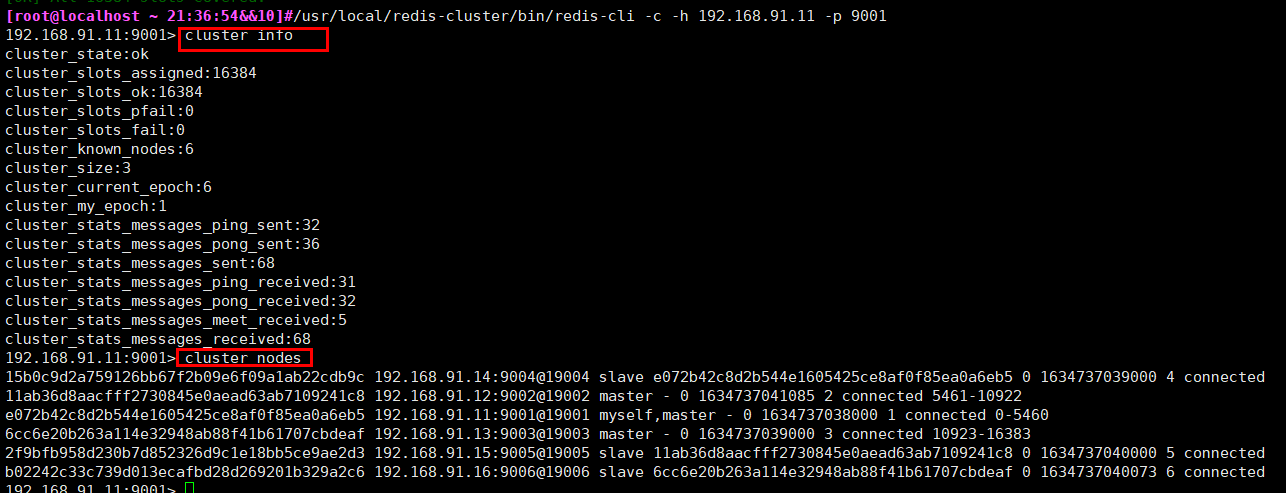
**以上步骤其他5个节点也是如此操作（注意修改IP和端口号）**

最后创建集群（任意一台。本次在11上）

#/usr/local/redis-cluster/bin/redis-trib.rb create --replicas 1 192.168.91.11:9001 192.168.91.12:9002 192.168.91.13:9003 192.168.91.14:9004 192.168.91.15:9005 192.168.91.16:9006



#/usr/local/redis/bin/redis-cli -c -h 192.168.91.11 -p 9001

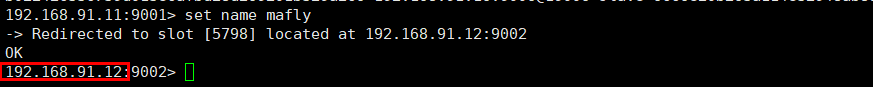


检查集群状态

#redis-trib.rb check 192.168.91.11:9001

**设置一个 mafly：**  
你会发现，当我们 set name mafly 时，出现了 Redirected to slot 信息并自动连接到了9002节点。这也是集群的一个数据分配特性，这里不详细说了。

redis-cli -p 9001 -h 192.168.91.11 -c



**参考url:**

<https://www.cnblogs.com/mafly/p/redis_cluster.html>

<https://blog.csdn.net/qq_30242987/article/details/99729702>

<https://www.icode9.com/content-2-330853.html>

<https://blog.whsir.com/post-2659.html>

<https://ruby-china.org/wiki/install_ruby_guide/>

# **8 Ansilbe 部署redis集群（redis 4版本）**

<https://www.cnblogs.com/ZFBG/p/13429733.html>

## **8.1运行方式**

ansible-playbook ./site.yml -i ./hosts

运行之前需要修改2个地方

1 hosts 修改对应的主机IP 以及密码

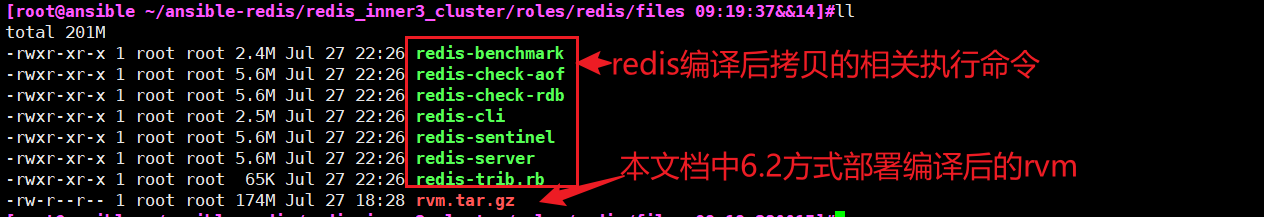
2 group\_vars/redis.yml 里面的 redis\_cluster\_ip

## **8.2 关于升级**

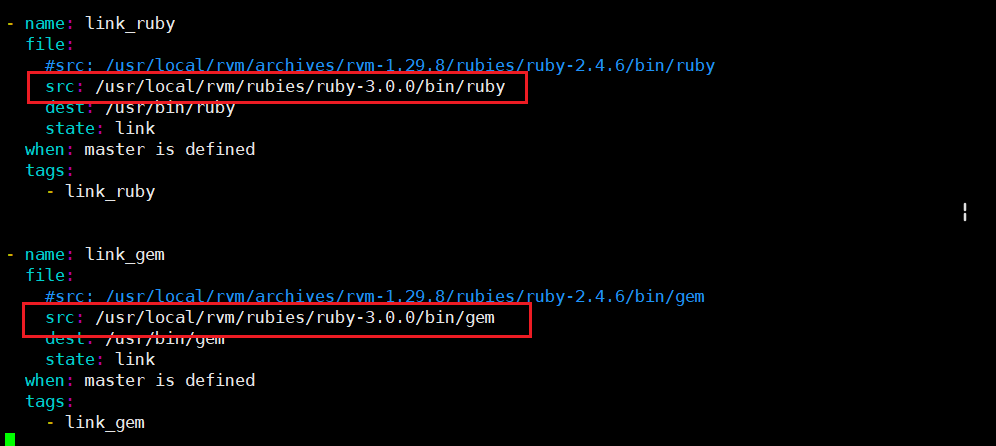
**8.2.1 Ansible-redis-v1**

需要注意files下的内容

~/ansible-redis/redis\_inner3\_cluster/roles/redis/files



再有就是修改tasks/ruby.yml文件



rvm只部署在标记为master上的服务器

安装包ansible-redis-v1.tar.gz

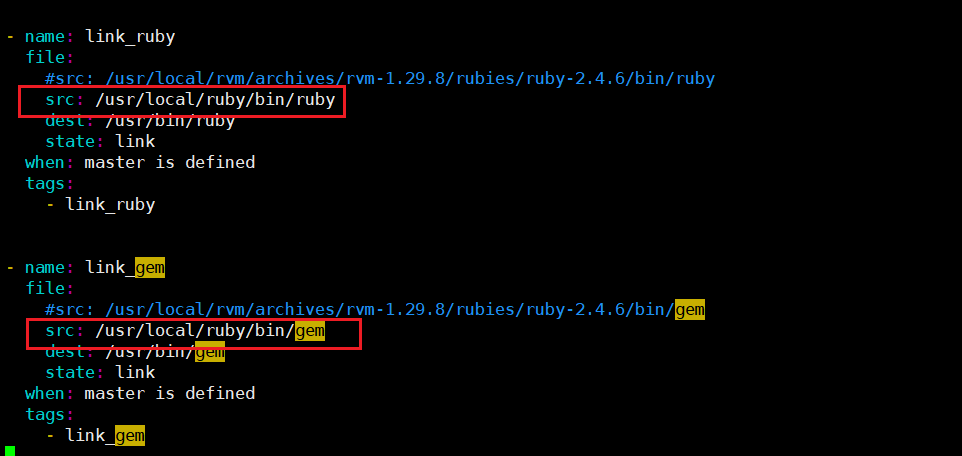
**8.2.3 Ansible-redis-v2**

v2的包比较小，采用v2



与Ansible-redis-v1的区别

tasks/ruby.yml



tasks/install\_redis.yml

备注1：redis5.0之前的版本创建集群需要用到ruby 命令，5.0之后不需要，我们的是5.0.13  所以直接跳过ruby，

备注2：redis5.0之前的版本创建集群使用redis-trib.rb命令，5.0之后的版本直接使用redis-cli命令



安装包ansible-redis-v2.tar.gz

# **9 Ansilbe 部署redis集群（redis 4以上版本）**

备注1：redis5.0之前的版本创建集群需要用到ruby 命令，5.0之后不需要，我们的是5.0.13  所以直接跳过ruby，

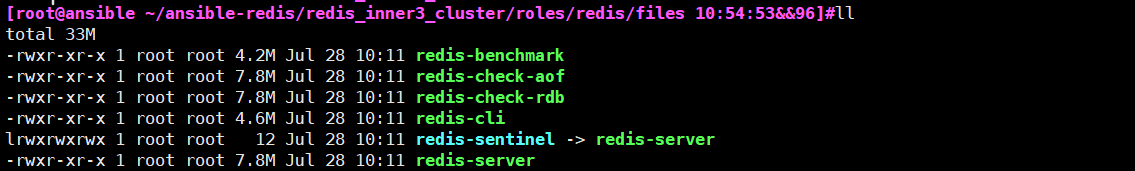
备注2：redis5.0之前的版本创建集群使用redis-trib.rb命令，5.0之后的版本直接使用redis-cli命令

--cluster-replicas 1  ：表示主从复制比例为 1:1，即一个主节点对应一个从节点，我们有6个节点，即默认前3个未主，后3个为从

-a 123456 ：redis密码参数，若redis无密码则不需要此参数

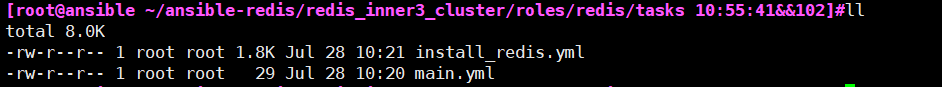
**相对8 做了如下修改**

1. **~/ansible-redis/redis\_inner3\_cluster/roles/redis/files**



只留下了redis编译后的命令文件，同时版本5也没有redis-trib.rb用不到。

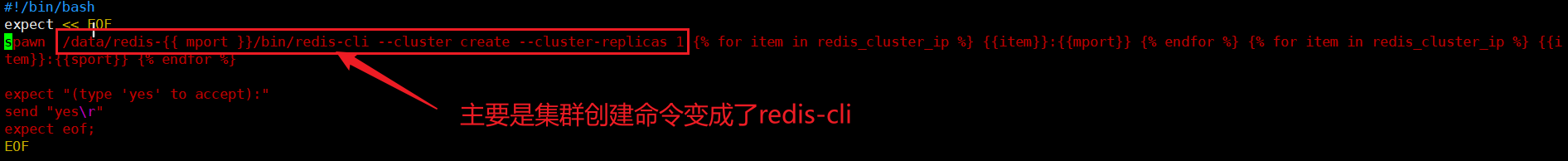
1. **~/ansible-redis/redis\_inner3\_cluster/roles/redis/tasks**



1. **删除了ruby.yaml文件**
2. **install\_redis.yml 删除了安装gem的配置**



1. **~/ansible-redis/redis\_inner3\_cluster/roles/redis/templates/add\_cluster.sh**



**检查集群状态**

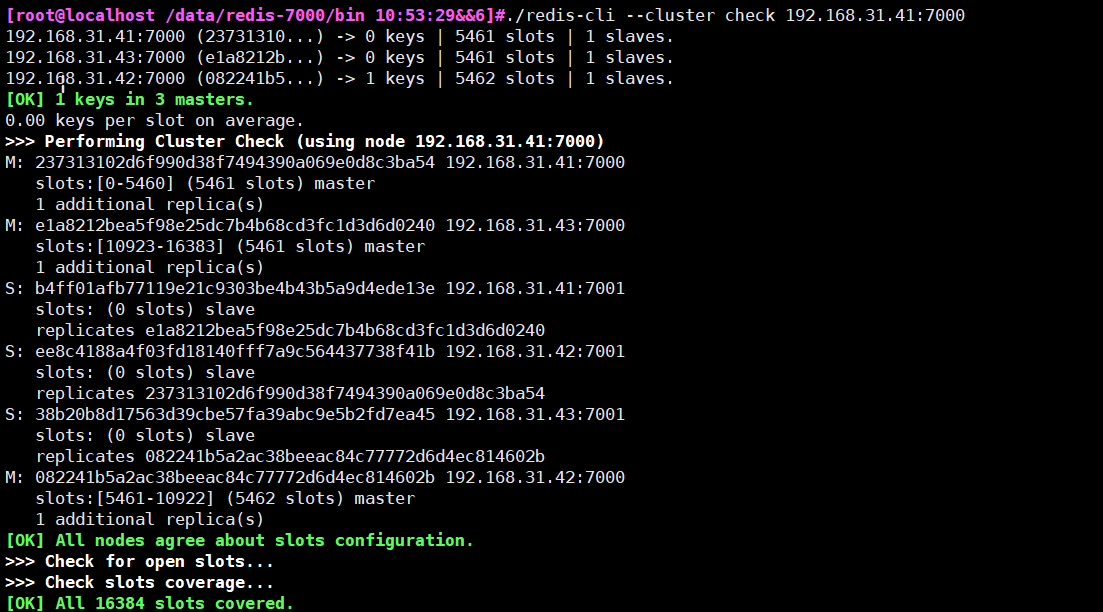
redis-cli -c -h 192.168.31.41 -p 7000 cluster info

redis-cli -c -h 192.168.31.41 -p 7000 cluster nodes

-c 是进入集群状态

**高版本检查集群健康状态**

./redis-cli --cluster check 192.168.31.41:7000



**安装包：ansible-redis-v3.tar.gz**

<https://blog.csdn.net/u010976445/article/details/119331006>