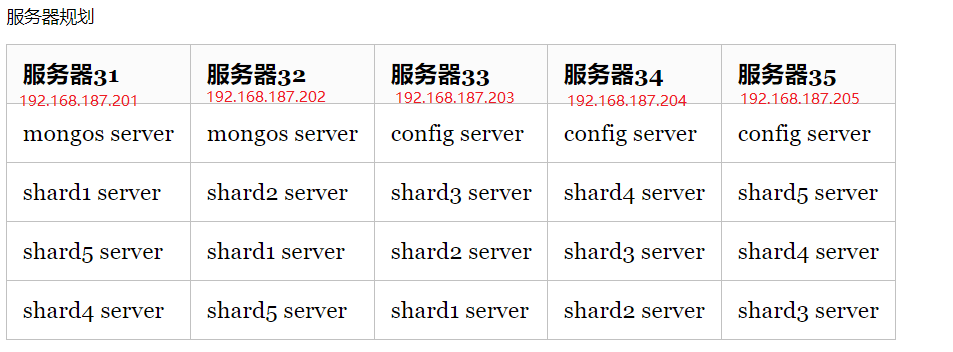
# MongoDB一键化部署

# mongodb节点分配规划

这里使用的mongodb版本为3.6.3,如图所示，在5台服务器的规格为：



在4台服务器的规格为：



# 系统基础配置

下面的命令默认都使用root用户进行操作

1. **修改系统配置文件/etc/security/limits.conf和/etc/profile**

修改/etc/security/limits.conf 持久化设置允许用户/进程打开文件句柄数，这一步需要重启系统，不然不起作用

*\* soft nofile 1048576*

*\* hard nofile 1048576*

*\* soft nproc 524288*

*\* hard nproc 524288*

修改/etc/profile，在最后添加*ulimit -s 1024*，然后保存并*source /etc/profile*

1. **关闭每台机器的防火墙：**

chkconfig iptables off && service iptables stop

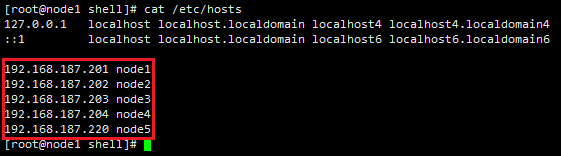
使用命令查看chkconfig –list 是否设置自动启动为关闭



使用命令查看防火墙是否关闭service iptables status



1. **每台机器修改/etc/hosts和hostname，并在/etc/hosts添加集群所有的ip及对应的hostname，如图：**



1. **5台机器之间配置ssh免密登录**

每台机器分别执行下面命令

ssh-keygen -t rsa -P "" -f /root/.ssh/id\_rsa

ssh-copy-id node1

ssh-copy-id node2

ssh-copy-id node3

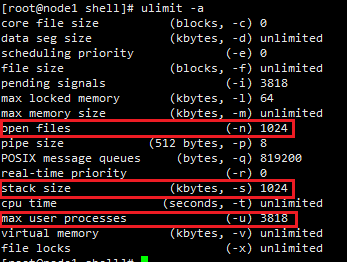
ssh-copy-id node4

ssh-copy-id node5

1. **检查自动化部署所需的命令是否可用**

**ifconfig、ssh需要可用，需要安装numactl，使用rpm -qa | grep numactl查看numactl是否已安装。**

注意：完成上面5步操作后重启每台机器。重启完毕，使用命令ulimit -a 查看



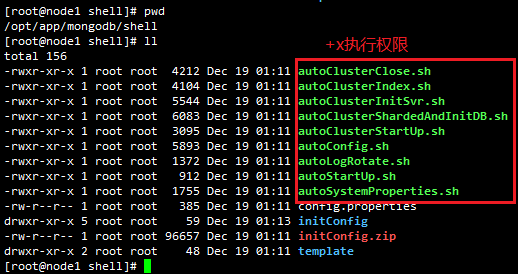
要保证open files对应的值为*1048576*

Stack size对应的值为1024

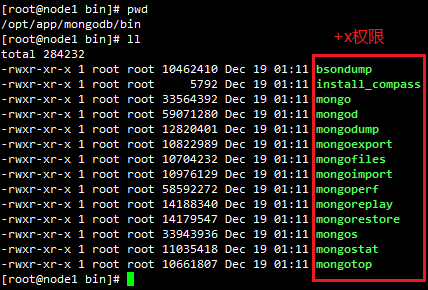
Max user processes对应的值为*524288*

1. **检查mongodb安装下bin目录和shell目录执行权限**

**进入到shell目录，执行chmod +x auto\*加执行权限**

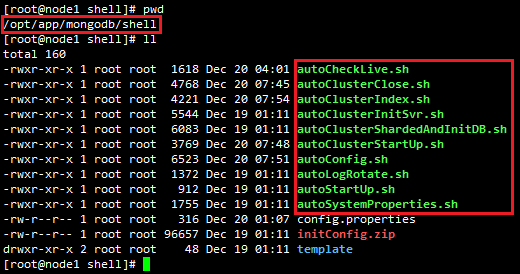


**进入mongodb的bin目录，执行chmod +x ./\*加执行权限**



# MongoDB一键化部署

## 一键化部署shell文件介绍



Shell脚本文件目前放置在mongodb的安装目录下，

①autoClusterClose.sh脚本为mongodb集群关闭脚本，用法：./autoClusterClose.sh shutdown或者./autoClusterClose.sh kill，这里建议使用shutdown方式关闭。

②autoClusterIndex.sh脚本为mongodb集群初始化配置文件和修改系统配置、config和shard副本集以及mongos路由启动并初始化、mongodb集群数据库表分片和基础数据初始化的脚本、添加定时任务。用法：./ autoClusterIndex.sh。使用该脚本，可以全自动化安装mongodb。

③autoClusterInitSvr.sh脚本为mongodb集群config、shard和mongos初始化脚本。用法：./autoClusterInitSvr.sh cs，初始化config和shard副本集配置；./autoClusterInitSvr.sh mongos，初始化mongos路由配置。

④autoClusterShardedAndInitDB.sh 脚本为指定数据库meerkat-test-v104、face\_manager、face\_manager及其数据库表分片。

⑤autoClusterStartUp.sh 脚本为config和shard副本集、mongos路由启动脚本，用法：./ autoClusterStartUp.sh，全部启动，包括config和shard副本集、mongos路由；./ autoClusterStartUp.sh notmongos，只启动config和shard副本集。

⑥autoConfig.sh脚本为创建所在服务器对应的mongodb的配置文件如config.conf、shard1.conf、mongos.conf，以及添加日志切割和mongo进程存活监控定时任务，用法：./autoConfig.sh addLogRotateCronTask。

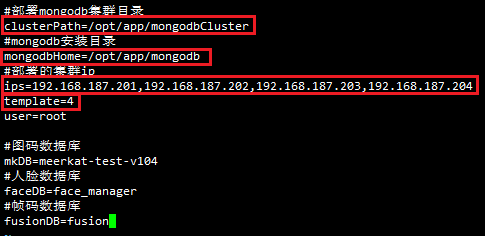
⑦autoLogRotate.sh脚本为创建所在服务器对应的mongodb的日志切割脚本，并添加定时任务，默认为每天凌晨0点0分，对mongodb数据库日志进行切割并删除7天前的日志。

⑧autoCheckLive.sh脚本为创建所在服务器对应的mongodb的进程存活监控，不存在则自动拉起。

⑨autoStartUp.sh脚本为启动所在服务器的config、shard、mongos节点。

⑩autoSystemProperties.sh 脚本为修改服务器的系统配置，TCP配置等。

⑪config.properties，配置文件，包括mongodb安装目录、mongoCluster集群安装目录、所需主机、数据库等等。如图：



上图四个参数可能需要随环境不同而修改，其他属性不建议修改。

**说明：**

ips=192.168.187.201, 192.168.187.202, 192.168.187.203, 192.168.187.204

这里默认192.168.187.201为mongdb集群的节点1，192.168.187.202为mongdb集群的节点2，192.168.187.203为mongdb集群的节点3，192.168.187.204为mongdb集群的节点4。

template与ips的ip数量对应，ips的ip数为4，则意味着采用4台机器的模板，这里默认为4。

⑫initConfig.zip 为数据库meerkat-test-v104、face\_manager、face\_manager及其数据库表的初始化文件。

⑬template 目录包括3.conf、4.conf和5.conf，为mongodb3台机器、4台机器和5台机器安装模板，强烈建议不要动或者修改。

## 全自动化部署

1. 把mongodb安装目录放置在4台机器中的任意一台上面即可。

2. 进入mongodb/shell目录



根据需要修改config.properties文件的clusterPath、mongodbHome、ips三个属性

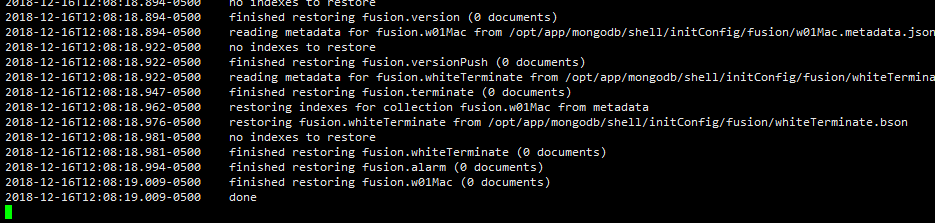
1. autoClusterIndex.sh脚本所在的目录执行如下命令，否则脚本会不起作用

执行命令

./ autoClusterIndex.sh > log.txt 2>&1



执行上面命令，会一键自动化部署，并把日志输入到log.txt里面，另开一个窗口并进入到shell目录，使用tail -f log.txt可以查看自动化部署执行的进度。



至此，完成部署。

**一键自动化部署正确完整日志，以供参考：**

**5台机器的日志：**

**4台机器的日志：**

## 半自动化部署

如果全自动化出现问题，可以使用半自动化部署，需要把mongodbCluster这个目录删除，并把mongodb安装目录拷贝到每台机器上面（如果已有则不用拷贝），再执行下面操作，否则会出错。

每台机器进入mongodb安装目录shell目录下，config.properties的修改同上。

1. 创建mongodbCluster和修改系统属性配置，执行如下命令：

./autoConfig.sh && ./ autoSystemProperties.sh

1. 然后分别启动config、shard1、shard2、shard3、shard4

服务器32/33/34(即node2/node3/node4)启动config副本集执行如下命令：

./autoStartUp.sh config\_numa

服务器31/32/33(即node1/node2/node3)启动shard1副本集执行如下命令：

./autoStartUp.sh shard1\_numa

服务器32/33/34(即node2/node3/node4)启动shard2副本集执行如下命令：

./autoStartUp.sh shard2\_numa

服务器33/34/31(即node3/node4/node1)启动shard3副本集执行如下命令：

./autoStartUp.sh shard3\_numa

服务器34/31/32(即node4/node1/node2)启动shard4副本集执行如下命令：

./autoStartUp.sh shard4\_numa

1. 初始化config、shard1、shard2、shard3、shard4的配置，执行如下命令：

./autoClusterInitSvr.sh cs

1. 启动服务器31（node1）、服务器32（node2）、服务器33（node3）的mongos，执行如下命令：

./autoStartUp.sh mongos

1. Mongos初始化，执行如下命令：

./autoClusterInitSvr.sh mongos

1. mongodb集群数据库表分片和初始化，执行如下命令：

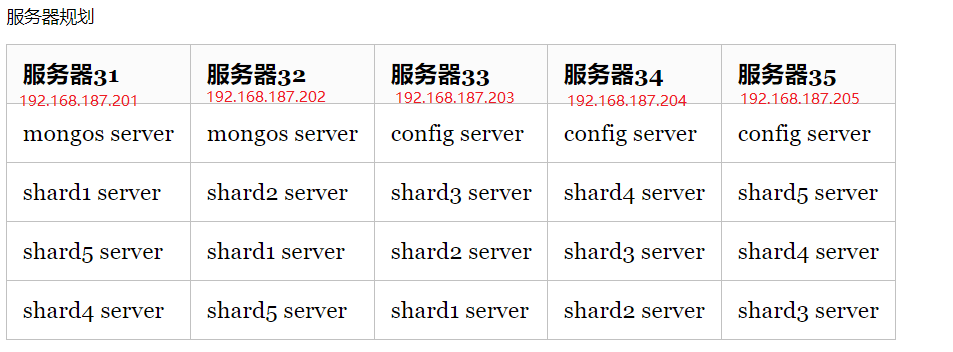
./autoClusterShardedAndInitDB.sh

1. mongodb集群添加定时任务，定时日志切割和删除，mongo进程存活监控，执行如下命令(每台机器都执行)：

./autoConfig.sh addCronTask

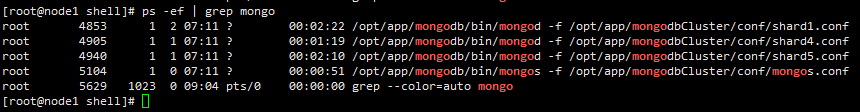
## 查看每台机器启动的mongod实例是否正确

**5台机器查看：**

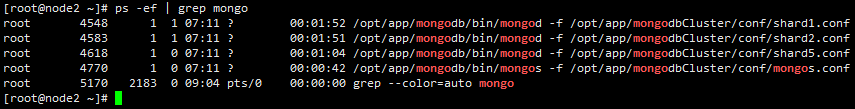


对比上面每台机器应该启动的mongo实例，查看每台机器启动的是否正确：

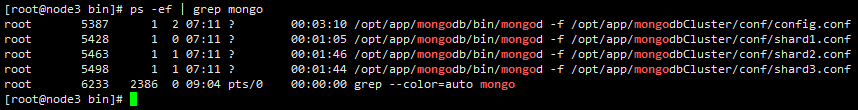
服务器31（192.168.187.201）：



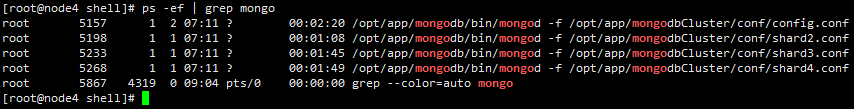
服务器32（192.168.187.202）：



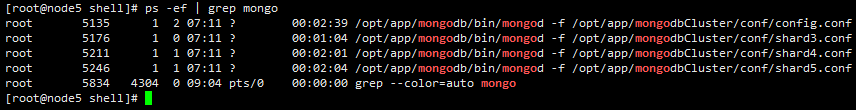
服务器33（192.168.187.203）：



服务器34（192.168.187.204）：



服务器35（192.168.187.205）：



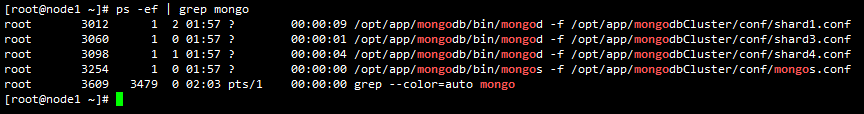
注意：如果不成功，可以使用./autoClusterClose.sh kill把集群已启动的mongodb实例kill掉，把mongodbCluster目录删除，然后重新来过

**4台机器查看：**

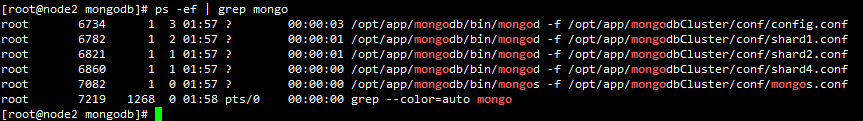


对比上面每台机器应该启动的mongo实例，查看每台机器启动的是否正确：

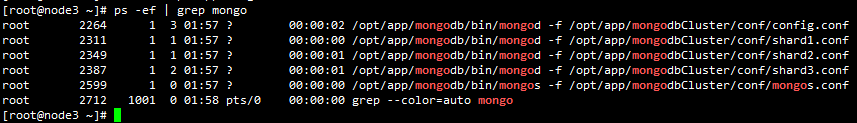
服务器31（192.168.187.201）：



服务器32（192.168.187.202）：



服务器33（192.168.187.203）：



服务器34（192.168.187.204）：

