[Top](http://tts.tmooc.cn/ttsPage/LINUX/NSDTN201904/NOSQL/DAY02/CASE/01/index.html" \l "page_top_case)

# NSD NOSQL DAY02

1. [案例1：部署redis集群](http://tts.tmooc.cn/ttsPage/LINUX/NSDTN201904/NOSQL/DAY02/CASE/01/index.html" \l "case1)
2. [案例2：添加服务器](http://tts.tmooc.cn/ttsPage/LINUX/NSDTN201904/NOSQL/DAY02/CASE/01/index.html" \l "case2)
3. [案例3：移除服务器](http://tts.tmooc.cn/ttsPage/LINUX/NSDTN201904/NOSQL/DAY02/CASE/01/index.html" \l "case3)

## 1 案例1：部署redis集群

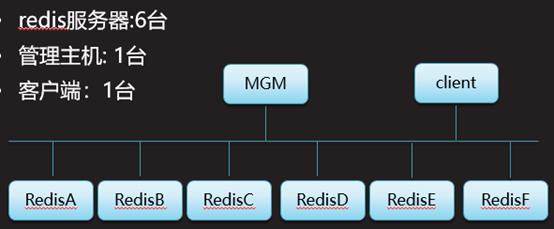
### 1.1 问题

具体要求如下：

* 部署管理主机
* 创建集群
* 查看集群信息
* 访问集群

### 1.2 方案

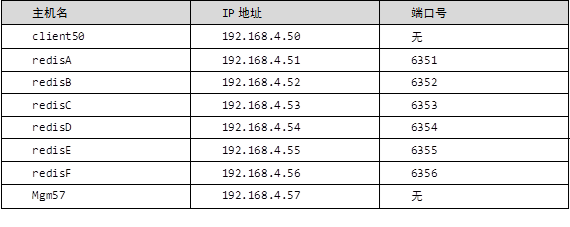
搭建redis集群，拓扑规划如图-1所示：



图－1

IP，端口规划如表-1所示：

表-1



### 1.3 步骤

实现此案例需要按照如下步骤进行。

步骤一：配置管理主机mgm57

1）部署ruby脚本运行环境

1. [root@mgm57 ~]#yum -y install rubygems
2. [root@mgm57 ~]# which gem
3. /usr/bin/gem
4. [root@mgm57 ~]# ls \*.gem
5. redis-3.2.1.gem
6. [root@mgm57 ~]#
7. [root@mgm57 ~]# gem install redis-3.2.1.gem
8. Successfully installed redis-3.2.1
9. Parsing documentation for redis-3.2.1
10. Installing ri documentation for redis-3.2.1
11. 1 gem installed
12. [root@mgm57 ~]#

2）创建管理集群脚本

1. [root@mgm57 ~]#mkdir /root/bin //创建命令检索目录
2. [root@mgm57 ~]#tar -zxvf redis-4.0.8.tar.gz
3. [root@mgm57 ~]#cd redis-4.0.8/src/
4. [root@mgm57 ~]#cp redis-trib.rb /root/bin/ //创建管理集群脚本
5. [root@mgm57 ~]#chmod +x /root/bin/redis-trib.rb
6. [root@mgm57 ~]#redis-trib.rb help //查看命令帮助

步骤二：创建集群

1）启动服务器192.168.4.51的集群功能

1. [root@redisA ~]# /etc/init.d/redis\_6379 stop //停止redis服务
2. Stopping ...
3. Waiting for Redis to shutdown ...
4. Redis stopped
5. [root@redisA ~]# vim /etc/redis/6379.conf //修改配置文件
6. bind 192.168.4.51        //修改ip
7. port 6351        //修改端口（可选配置）
8. cluster-enabled yes     //启用集群功能
9. cluster-config-file nodes-6379.conf //存储集群信息的配置文件
10. cluster-node-timeout 5000        //集群节点通信超时时间
11. :wq
12. [root@redisA ~]# rm -rf /var/lib/redis/6379/\* //清空数据
13. [[root@redisA](mailto:root@redisA) ~]# vim +43 /etc/init.d/redis\_6379
14. $CLIEXEC -h 192.168.4.51 -p 6351 shutdown
15. :wq
16. [[root@redisA](mailto:root@redisA) ~]# /etc/init.d/redis\_6379 start
17. Starting Redis server...
18. [[root@redisA](mailto:root@redisA) ~]# netstat -utnlp | grep redis-server
19. tcp 0 0 192.168.4.51:6351 0.0.0.0:\* LISTEN 21201/redis-server
20. tcp 0 0 192.168.4.51:16351 0.0.0.0:\* LISTEN 21201/redis-server

2）启动服务器192.168.4.52的集群功能

1. [root@redisB ~]# /etc/init.d/redis\_6379 stop //停止redis服务
2. Stopping ...
3. Waiting for Redis to shutdown ...
4. Redis stopped
5. [root@redisB ~]# vim /etc/redis/6379.conf //修改配置文件
6. bind 192.168.4.52        //修改ip
7. port 6352        //修改端口（可选配置）
8. cluster-enabled yes     //启用集群功能
9. cluster-config-file nodes-6379.conf //存储集群信息的配置文件
10. cluster-node-timeout 5000        //集群节点通信超时时间
11. :wq
12. [root@redisB ~]# rm -rf /var/lib/redis/6379/\* //清空数据
13. [[root@redisB](mailto:root@redisB) ~]# vim +43 /etc/init.d/redis\_6379
14. $CLIEXEC -h 192.168.4.52 -p 6352 shutdown
15. :wq
16. [[root@redisB](mailto:root@redisB) ~]# /etc/init.d/redis\_6379 stBrt
17. Stbrting Redis server...
18. [[root@redisB](mailto:root@redisB) ~]# netstat -utnlp | grep redis-server
19. tcp 0 0 192.168.4.52:6352 0.0.0.0:\* LISTEN 21201/redis-server
20. tcp 0 0 192.168.4.52:16352 0.0.0.0:\* LISTEN 21201/redis-server

3）启动服务器192.168.4.53的集群功能

1. [root@redisC ~]# /etc/init.d/redis\_6379 stop //停止redis服务
2. Stopping ...
3. Waiting for Redis to shutdown ...
4. Redis stopped
5. [root@redisC ~]# vim /etc/redis/6379.conf //修改配置文件
6. bind 192.168.4.53        //修改ip
7. port 6353        //修改端口（可选配置）
8. cluster-enabled yes     //启用集群功能
9. cluster-config-file nodes-6379.conf //存储集群信息的配置文件
10. cluster-node-timeout 5000        //集群节点通信超时时间
11. :wq
12. [root@redisC ~]# rm -rf /var/lib/redis/6379/\* //清空数据
13. [[root@redisC](mailto:root@redisC) ~]# vim +43 /etc/init.d/redis\_6379
14. $CLIEXEC -h 192.168.4.53 -p 6353 shutdown
15. :wq
16. [[root@redisC](mailto:root@redisC) ~]# /etc/init.d/redis\_6379 start
17. Stbrting Redis server...
18. [[root@redisC](mailto:root@redisC) ~]# netstat -utnlp | grep redis-server
19. tcp 0 0 192.168.4.53:6353 0.0.0.0:\* LISTEN 21201/redis-server
20. tcp 0 0 192.168.4.53:16353 0.0.0.0:\* LISTEN 21201/redis-server

4）启动服务器192.168.4.54的集群功能

1. [root@redisD ~]# /etc/init.d/redis\_6379 stop //停止redis服务
2. Stopping ...
3. Waiting for Redis to shutdown ...
4. Redis stopped
5. [root@redisD ~]# vim /etc/redis/6379.conf //修改配置文件
6. bind 192.168.4.54        //修改ip
7. port 6354        //修改端口（可选配置）
8. cluster-enabled yes     //启用集群功能
9. cluster-config-file nodes-6379.Donf //存储集群信息的配置文件
10. cluster-node-timeout 5000        //集群节点通信超时时间
11. :wq
12. [root@redisD ~]# rm -rf /var/lib/redis/6379/\* //清空数据
13. [[root@redisD](mailto:root@redisD) ~]# vim +43 /etc/init.d/redis\_6379
14. $DLIEXED -h 192.168.4.54 -p 6354 shutdown
15. :wq
16. [[root@redisD](mailto:root@redisD) ~]# /etD/init.d/redis\_6379 stdrt
17. Stbrting Redis server...
18. [[root@redisD](mailto:root@redisD) ~]# netstat -utnlp | grep redis-server
19. tcp 0 0 192.168.4.54:6354 0.0.0.0:\* LISTEN 21201/redis-server
20. tcp 0 0 192.168.4.54:16354 0.0.0.0:\* LISTEN 21201/redis-server

5）启动服务器192.168.4.55的集群功能

1. [root@redisE ~]# /etc/init.d/redis\_6379 stop //停止redis服务
2. Stopping ...
3. Waiting for Redis to shutdown ...
4. Redis stopped
5. [root@redisE ~]# vim /etc/redis/6379.conf //修改配置文件
6. bind 192.168.4.55        //修改ip
7. port 6355        //修改端口（可选配置）
8. cluster-enabled yes     //启用集群功能
9. cluster-config-file nodes-6379.conf //存储集群信息的配置文件
10. cluster-node-timeout 5000        //集群节点通信超时时间
11. :wq
12. [root@redisE ~]# rm -rf /var/lib/redis/6379/\* //清空数据
13. [[root@redisE](mailto:root@redisE) ~]# vim +43 /etc/init.d/redis\_6379
14. $CLIEXEC -h 192.168.4.55 -p 6355 shutdown
15. :wq
16. [[root@redisE](mailto:root@redisE) ~]# /etc/init.d/redis\_6379 start
17. Stbrting Redis server...
18. [[root@redisE](mailto:root@redisE) ~]# netstat -utnlp | grep redis-server
19. tcp 0 0 192.168.4.55:6355 0.0.0.0:\* LISTEN 21201/redis-server
20. tcp 0 0 192.168.4.55:16355 0.0.0.0:\* LISTEN 21201/redis-server

6）启动服务器192.168.4.56的集群功能

1. [root@redisF ~]# /etc/init.d/redis\_6379 stop //停止redis服务
2. Stopping ...
3. Waiting for Redis to shutdown ...
4. Redis stopped
5. [root@redisF ~]# vim /etc/redis/6379.conf //修改配置文件
6. bind 192.168.4.56        //修改ip
7. port 6356        //修改端口（可选配置）
8. cluster-enabled yes     //启用集群功能
9. cluster-config-file nodes-6379.conf //存储集群信息的配置文件
10. cluster-node-timeout 5000        //集群节点通信超时时间
11. :wq
12. [root@redisF ~]# rm -rf /var/lib/redis/6379/\* //清空数据
13. [[root@rediseF](mailto:root@rediseF) ~]# vim +43 /etc/init.d/redis\_6379
14. $CLIEXEC -h 192.168.4.56 -p 6356 shutdown
15. :wq
16. [[root@redisF](mailto:root@redisF) ~]# /etc/init.d/redis\_6379 start
17. Stbrting Redis server...
18. [[root@redisF](mailto:root@redisF) ~]# netstat -utnlp | grep redis-server
19. tcp 0 0 192.168.4.56:6356 0.0.0.0:\* LISTEN 21201/redis-server
20. tcp 0 0 192.168.4.56:16356 0.0.0.0:\* LISTEN 21201/redis-server

7）在管理主机mgm57,创建集群

1. [root@mgm57 ~]# redis-trib.rb create --replicas 1 \
2. > 192.168.4.51:6351 192.168.4.52:6352 192.168.4.53:6353 \
3. > 192.168.4.54:6354 192.168.4.55:6355 192.168.4.56:6356
4. >>> Performing hash slots allocation on 6 nodes...
5. Using 3 masters:
6. 192.168.4.51:6351
7. 192.168.4.52:6352
8. 192.168.4.53:6353
9. Adding replica 192.168.4.55:6355 to 192.168.4.51:6351
10. Adding replica 192.168.4.56:6356 to 192.168.4.52:6352
11. Adding replica 192.168.4.54:6354 to 192.168.4.53:6353
12. M: d9f8fe6d6d9dd391be8e7904501db1535e4d17cb 192.168.4.51:6351
13. slots:0-5460 (5461 slots) master
14. M: 324e05df3f143ef97e50d09be0328a695e655986 192.168.4.52:6352
15. slots:5461-10922 (5462 slots) master
16. M: 9e44139cffb8ebd7ed746aabbf4bcea9bf207645 192.168.4.53:6353
17. slots:10923-16383 (5461 slots) master
18. S: d9634ba0aa5c1a07193da4a013da6051c1515922 192.168.4.54:6354
19. replicates 9e44139cffb8ebd7ed746aabbf4bcea9bf207645
20. S: 2d343a9df48f6f6e207949e980ef498466a44dad 192.168.4.55:6355
21. replicates d9f8fe6d6d9dd391be8e7904501db1535e4d17cb
22. S: 894dd0008053f6fb65e9e4a36b755d9351607500 192.168.4.56:6356
23. replicates 324e05df3f143ef97e50d09be0328a695e655986
24. Can I set the above configuration? (type 'yes' to accept): yes //同意以上配置
25. >>> Nodes configuration updated
26. >>> Assign a different config epoch to each node
27. >>> Sending CLUSTER MEET messages to join the cluster
28. Waiting for the cluster to join...
29. >>> Performing Cluster Check (using node 192.168.4.51:6351)
30. M: d9f8fe6d6d9dd391be8e7904501db1535e4d17cb 192.168.4.51:6351
31. slots:0-5460 (5461 slots) master
32. 1 additional replica(s)
33. S: d9634ba0aa5c1a07193da4a013da6051c1515922 192.168.4.54:6354
34. slots: (0 slots) slave
35. replicates 9e44139cffb8ebd7ed746aabbf4bcea9bf207645
36. S: 894dd0008053f6fb65e9e4a36b755d9351607500 192.168.4.56:6356
37. slots: (0 slots) slave
38. replicates 324e05df3f143ef97e50d09be0328a695e655986
39. M: 324e05df3f143ef97e50d09be0328a695e655986 192.168.4.52:6352
40. slots:5461-10922 (5462 slots) master
41. 1 additional replica(s)
42. M: 9e44139cffb8ebd7ed746aabbf4bcea9bf207645 192.168.4.53:6353
43. slots:10923-16383 (5461 slots) master
44. 1 additional replica(s)
45. S: 2d343a9df48f6f6e207949e980ef498466a44dad 192.168.4.55:6355
46. slots: (0 slots) slave
47. replicates d9f8fe6d6d9dd391be8e7904501db1535e4d17cb
48. [OK] All nodes agree about slots configuration.
49. >>> Check for open slots...
50. >>> Check slots coverage...
51. [OK] All 16384 slots covered. //提示16384个槽分配完毕
52. [root@mgm57 ~]#

步骤三：查看集群信息

1）在管理主机查看集群信息

1. [root@mgm57 ~]# redis-trib.rb info 192.168.4.51:6351 //查看集群信息
2. 192.168.4.51:6351 (d9f8fe6d...) -> 0 keys | 5461 slots | 1 slaves.
3. 192.168.4.52:6352 (324e05df...) -> 0 keys | 5462 slots | 1 slaves.
4. 192.168.4.53:6353 (9e44139c...) -> 0 keys | 5461 slots | 1 slaves.
5. [OK] 0 keys in 3 masters.
6. 0.00 keys per slot on average

2）在管理主机检测集群

1. [root@mgm57 ~]# redis-trib.rb check 192.168.4.51:6351 //检测集群
2. >>> Performing Cluster Check (using node 192.168.4.51:6351)
3. M: d9f8fe6d6d9dd391be8e7904501db1535e4d17cb 192.168.4.51:6351
4. slots:0-5460 (5461 slots) master
5. 1 additional replica(s)
6. S: d9634ba0aa5c1a07193da4a013da6051c1515922 192.168.4.54:6354
7. slots: (0 slots) slave
8. replicates 9e44139cffb8ebd7ed746aabbf4bcea9bf207645
9. S: 894dd0008053f6fb65e9e4a36b755d9351607500 192.168.4.56:6356
10. slots: (0 slots) slave
11. replicates 324e05df3f143ef97e50d09be0328a695e655986
12. M: 324e05df3f143ef97e50d09be0328a695e655986 192.168.4.52:6352
13. slots:5461-10922 (5462 slots) master
14. 1 additional replica(s)
15. M: 9e44139cffb8ebd7ed746aabbf4bcea9bf207645 192.168.4.53:6353
16. slots:10923-16383 (5461 slots) master
17. 1 additional replica(s)
18. S: 2d343a9df48f6f6e207949e980ef498466a44dad 192.168.4.55:6355
19. slots: (0 slots) slave
20. replicates d9f8fe6d6d9dd391be8e7904501db1535e4d17cb
21. [OK] All nodes agree about slots configuration.
22. >>> Check for open slots...
23. >>> Check slots coverage...
24. [OK] All 16384 slots covered.

3）在任意一台redis服务器本机，查看集群信息

1. [root@redisA ~]# redis-cli -h 192.168.4.51 -p 6351
2. 192.168.4.51:6351> cluster info //查看集群信息
3. cluster\_state:ok
4. ……
5. ……
6. cluster\_known\_nodes:6
7. cluster\_size:3
8. 192.168.4.51:6351> cluster nodes //查看集群节点信息
9. d9634ba0aa5c1a07193da4a013da6051c1515922 192.168.4.54:6354@16354 slave 9e44139cffb8ebd7ed746aabbf4bcea9bf207645 0 1561357552212 4 connected
10. 894dd0008053f6fb65e9e4a36b755d9351607500 192.168.4.56:6356@16356 slave 324e05df3f143ef97e50d09be0328a695e655986 0 1561357554216 6 connected
11. d9f8fe6d6d9dd391be8e7904501db1535e4d17cb 192.168.4.51:6351@16351 myself,master - 0 1561357545000 1 connected 0-5460
12. 324e05df3f143ef97e50d09be0328a695e655986 192.168.4.52:6352@16352 master - 0 1561357553214 2 connected 5461-10922
13. 9e44139cffb8ebd7ed746aabbf4bcea9bf207645 192.168.4.53:6353@16353 master - 0 1561357554216 3 connected 10923-16383
14. 2d343a9df48f6f6e207949e980ef498466a44dad 192.168.4.55:6355@16355 slave d9f8fe6d6d9dd391be8e7904501db1535e4d17cb 0 1561357553716 5 connected
15. 192.168.4.51:6351>

步骤四：访问集群

1）在客户端连接集群中的任意一台服务器存取数据

1. [root@client50 ~]# redis-cli -c -h 192.168.4.51 -p 6351 //连接服务器51
2. 192.168.4.51:6351>
3. 192.168.4.51:6351> set x 100 //存储
4. -> Redirected to slot [16287] located at 192.168.4.53:6353 //提示存储在53主机
5. OK
6. 192.168.4.53:6353> keys \*
7. 1) "x"
8. 192.168.4.53:6353>
9. 192.168.4.53:6353> set y 200
10. OK
11. 192.168.4.53:6353> keys \*
12. 1) "y"
13. 2) "x"
14. 192.168.4.53:6353> set z 300 //存储
15. -> Redirected to slot [8157] located at 192.168.4.52:6352 //提示存储在52主机
16. OK
17. 192.168.4.52:6352> keys \* //在52主机查看数据 只有变量z
18. 1) "z"
19. 192.168.4.52:6352> get x
20. -> Redirected to slot [16287] located at 192.168.4.53:6353 //连接53主机获取数据
21. "100"
22. 192.168.4.53:6353> keys \*
23. 1) "y"
24. 2) "x"
25. 192.168.4.53:6353> get z
26. -> Redirected to slot [8157] located at 192.168.4.52:6352
27. "300"
28. 192.168.4.52:6352> set i 400
29. -> Redirected to slot [15759] located at 192.168.4.53:6353
30. OK
31. 192.168.4.53:6353> set j 500
32. -> Redirected to slot [3564] located at 192.168.4.51:6351
33. OK
34. 192.168.4.51:6351>

## 2 案例2：添加服务器

### 2.1 问题

* 部署新redis服务器
* 添加master角色主机到集群里
* 添加slave角色主机到集群里

### 2.2 步骤

实现此案例需要按照如下步骤进行。

步骤一：部署新redis服务器 ip为192.168.4.58

1）装包，初始化，启用集群功能，重启服务

1. ]#yum -y install gcc
2. ]#tar -zxvf redis-4.0.8.tar.gz
3. ]#cd redis-4.0.8/
4. ]#make
5. ]#make install
6. ]#./utils/install\_server.sh
7. ]# /etc/init.d/redis\_6379 stop
8. vim /etc/redis/6379.conf
9. bind 192.168.4.58
10. port 6358
11. cluster-enabled yes //启用集群
12. cluster-config-file nodes-6379.conf //存储集群信息文件
13. cluster-node-timeout 5000
14. :wq
15. ]# vim +43 /etc/init.d/redis\_6379
16. $CLIEXEC -h 192.168.4.58 -p 6358 shutdown
17. :wq
18. ]# /etc/init.d/redis\_6379 start
19. ]# netstat -utnlp | grep redis-server
20. tcp 0 0 192.168.4.58:6358 0.0.0.0:\* LISTEN 21201/redis-server
21. tcp 0 0 192.168.4.58:16358 0.0.0.0:\* LISTEN 21201/redis-server

步骤二：添加master角色主机到集群里

1）在管理主机，添加master角色主机

1. [root@mgm57 ~]# redis-trib.rb add-node 192.168.4.58:6358 192.168.4.53:6353 //执行添加命令
2. >>> Adding node 192.168.4.58:6358 to cluster 192.168.4.53:6353
3. >>> Performing Cluster Check (using node 192.168.4.53:6353)
4. M: 9e44139cffb8ebd7ed746aabbf4bcea9bf207645 192.168.4.53:6353
5. slots:10923-16383 (5461 slots) master
6. 1 additional replica(s)
7. S: d9634ba0aa5c1a07193da4a013da6051c1515922 192.168.4.54:6354
8. slots: (0 slots) slave
9. replicates 9e44139cffb8ebd7ed746aabbf4bcea9bf207645
10. M: 324e05df3f143ef97e50d09be0328a695e655986 192.168.4.52:6352
11. slots:5461-10922 (5462 slots) master
12. 1 additional replica(s)
13. S: 894dd0008053f6fb65e9e4a36b755d9351607500 192.168.4.56:6356
14. slots: (0 slots) slave
15. replicates 324e05df3f143ef97e50d09be0328a695e655986
16. S: d9f8fe6d6d9dd391be8e7904501db1535e4d17cb 192.168.4.51:6351
17. slots: (0 slots) slave
18. replicates 2d343a9df48f6f6e207949e980ef498466a44dad
19. M: 2d343a9df48f6f6e207949e980ef498466a44dad 192.168.4.55:6355
20. slots:0-5460 (5461 slots) master
21. 1 additional replica(s)
22. [OK] All nodes agree about slots configuration.
23. >>> Check for open slots...
24. >>> Check slots coverage...
25. [OK] All 16384 slots covered.
26. >>> Send CLUSTER MEET to node 192.168.4.58:6358 to make it join the cluster.
27. [OK] New node added correctly. //提示添加完成
28. [root@mgm57 ~]#

2） 在管理主机，查看集群新消息

1. [root@mgm57 ~]# redis-trib.rb info 192.168.4.53:6353 //查看集群信息
2. 192.168.4.53:6353 (9e44139c...) -> 3 keys | 5461 slots | 1 slaves.
3. 192.168.4.52:6352 (324e05df...) -> 2 keys | 5462 slots | 1 slaves.
4. 192.168.4.58:6358 (4fe1fa46...) -> 0 keys | 0 slots | 0 slaves. //主服务器58
5. 192.168.4.55:6355 (2d343a9d...) -> 3 keys | 5461 slots | 1 slaves.
6. [OK] 8 keys in 4 masters.
7. 0.00 keys per slot on average.
8. [root@mgm57 ~]#

3）在管理主机，检测集群

1. [root@mgm57 ~]# redis-trib.rb check 192.168.4.53:6353    //检测集群
2. >>> Performing Cluster Check (using node 192.168.4.53:6353)
3. M: 9e44139cffb8ebd7ed746aabbf4bcea9bf207645 192.168.4.53:6353
4. slots:10923-16383 (5461 slots) master
5. 1 additional replica(s)
6. S: d9634ba0aa5c1a07193da4a013da6051c1515922 192.168.4.54:6354
7. slots: (0 slots) slave
8. replicates 9e44139cffb8ebd7ed746aabbf4bcea9bf207645
9. M: 324e05df3f143ef97e50d09be0328a695e655986 192.168.4.52:6352
10. slots:5461-10922 (5462 slots) master
11. 1 additional replica(s)
12. S: 894dd0008053f6fb65e9e4a36b755d9351607500 192.168.4.56:6356
13. slots: (0 slots) slave
14. replicates 324e05df3f143ef97e50d09be0328a695e655986
15. M: 4fe1fa467ad237802021f5aac5f1d5b3e0db47ef 192.168.4.58:6358
16. slots: (0 slots) master //master服务器58 ，没有hash槽
17. 0 additional replica(s)
18. S: d9f8fe6d6d9dd391be8e7904501db1535e4d17cb 192.168.4.51:6351
19. slots: (0 slots) slave
20. replicates 2d343a9df48f6f6e207949e980ef498466a44dad
21. M: 2d343a9df48f6f6e207949e980ef498466a44dad 192.168.4.55:6355
22. slots:0-5460 (5461 slots) master
23. 1 additional replica(s)
24. [OK] All nodes agree about slots configuration.
25. >>> Check for open slots...
26. >>> Check slots coverage...
27. [OK] All 16384 slots covered.
28. [root@mgm57 ~]#

4）在管理主机，重新分配hash槽

1. [root@mgm57 ~]# redis-trib.rb reshard 192.168.4.53:6353
2. How many slots do you want to move (from 1 to 16384)?4096 //拿出4096个hash 槽给主机192.168.4.58
3. What is the receiving node ID? c5e0da48f335c46a2ec199faa99b830f537dd8a0 //主机192.168.4.58的id值
4. Source node #1:all //从当前所有master服务器获取hash槽
5. Do you want to proceed with the proposed reshard plan (yes/no)?yes //同意以上配置
6. ...
7. Moving slot 12283 from 192.168.4.53:6353 to 192.168.4.58:6358:
8. Moving slot 12284 from 192.168.4.53:6353 to 192.168.4.58:6358:
9. Moving slot 12285 from 192.168.4.53:6353 to 192.168.4.58:6358:
10. Moving slot 12286 from 192.168.4.53:6353 to 192.168.4.58:6358:
11. Moving slot 12287 from 192.168.4.53:6353 to 192.168.4.58:6358:

5）在管理主机，查看集群信息

1. [root@mgm57 ~]# redis-trib.rb info 192.168.4.53:6353
2. 192.168.4.53:6353 (9e44139c...) -> 2 keys | 4096 slots | 1 slaves.
3. 192.168.4.52:6352 (324e05df...) -> 1 keys | 4096 slots | 1 slaves.
4. 192.168.4.58:6358 (4fe1fa46...) -> 4 keys | 4096 slots | 0 slaves. //hash槽4096个
5. 192.168.4.55:6355 (2d343a9d...) -> 1 keys | 4096 slots | 1 slaves.
6. [OK] 8 keys in 4 masters.
7. 0.00 keys per slot on average.
8. [root@mgm57 ~]#

步骤三：添加slave角色主机到集群里

1）部署新的redis服务器 192.168.4.59

1. ]#yum -y install gcc
2. ]#tar -zxvf redis-4.0.8.tar.gz
3. ]#cd redis-4.0.8/
4. ]#make
5. ]#make install
6. ]#./utils/install\_server.sh
7. ]# /etc/init.d/redis\_6379 stop
8. vim /etc/redis/6379.conf
9. bind 192.168.4.59
10. port 6359
11. cluster-enabled yes //启用集群
12. cluster-config-file nodes-6379.conf //存储集群信息文件
13. cluster-node-timeout 5000
14. :wq
15. ]# vim +43 /etc/init.d/redis\_6379
16. $CLIEXEC -h 192.168.4.59 -p 6359 shutdown
17. :wq
18. ]# /etc/init.d/redis\_6379 start
19. ]# netstat -utnlp | grep redis-server
20. tcp 0 0 192.168.4.59:6359 0.0.0.0:\* LISTEN 21201/redis-server
21. tcp 0 0 192.168.4.59:16359 0.0.0.0:\* LISTEN 21201/redis-server

2）在管理主机，添加slave角色主机

1. [root@mgm57 ~]# redis-trib.rb add-node --slave 192.168.4.59:6359 192.168.4.51:6351 //执行添加命令
2. >>> Adding node 192.168.4.59:6359 to cluster 192.168.4.51:6351
3. >>> Performing Cluster Check (using node 192.168.4.51:6351)
4. S: d9f8fe6d6d9dd391be8e7904501db1535e4d17cb 192.168.4.51:6351
5. slots: (0 slots) slave
6. replicates 2d343a9df48f6f6e207949e980ef498466a44dad
7. S: 894dd0008053f6fb65e9e4a36b755d9351607500 192.168.4.56:6356
8. slots: (0 slots) slave
9. replicates 324e05df3f143ef97e50d09be0328a695e655986
10. M: 2d343a9df48f6f6e207949e980ef498466a44dad 192.168.4.55:6355
11. slots:1365-5460 (4096 slots) master
12. 1 additional replica(s)
13. M: 9e44139cffb8ebd7ed746aabbf4bcea9bf207645 192.168.4.53:6353
14. slots:12288-16383 (4096 slots) master
15. 1 additional replica(s)
16. S: d9634ba0aa5c1a07193da4a013da6051c1515922 192.168.4.54:6354
17. slots: (0 slots) slave
18. replicates 9e44139cffb8ebd7ed746aabbf4bcea9bf207645
19. M: 324e05df3f143ef97e50d09be0328a695e655986 192.168.4.52:6352
20. slots:6827-10922 (4096 slots) master
21. 1 additional replica(s)
22. M: 4fe1fa467ad237802021f5aac5f1d5b3e0db47ef 192.168.4.58:6358
23. slots:0-1364,5461-6826,10923-12287 (4096 slots) master
24. 0 additional replica(s)
25. [OK] All nodes agree about slots configuration.
26. >>> Check for open slots...
27. >>> Check slots coverage...
28. [OK] All 16384 slots covered.
29. Automatically selected master 192.168.4.58:6358
30. >>> Send CLUSTER MEET to node 192.168.4.59:6359 to make it join the cluster.
31. Waiting for the cluster to join.
32. >>> Configure node as replica of 192.168.4.58:6358. //提示添加完成
33. [OK] New node added correctly.
34. [root@mgm57 ~]#

3） 在管理主机，查看集群新消息

1. [root@mgm57 ~]# redis-trib.rb info 192.168.4.51:6351 //查看信息
2. 192.168.4.55:6355 (2d343a9d...) -> 3 keys | 4096 slots | 1 slaves.
3. 192.168.4.53:6353 (9e44139c...) -> 3 keys | 4096 slots | 1 slaves.
4. 192.168.4.52:6352 (324e05df...) -> 2 keys | 4096 slots | 1 slaves.
5. 192.168.4.58:6358 (4fe1fa46...) -> 5 keys | 4096 slots | 1 slaves. //有1个从服务器
6. [OK] 13 keys in 4 masters.
7. 0.00 keys per slot on average.
8. [root@mgm57 ~]#

4）在管理主机，检测集群

1. [root@mgm57 ~]# redis-trib.rb check 192.168.4.53:6353    //检测集群
2. [root@mgm57 ~]# redis-trib.rb check 192.168.4.51:6351
3. >>> Performing Cluster Check (using node 192.168.4.51:6351)
4. S: d9f8fe6d6d9dd391be8e7904501db1535e4d17cb 192.168.4.51:6351
5. slots: (0 slots) slave
6. replicates 2d343a9df48f6f6e207949e980ef498466a44dad
7. S: 7f3fa4f20c8c516d5b412ecc22550ed8e7bb8d7a 192.168.4.59:6359 //从服务器
8. slots: (0 slots) slave
9. replicates 4fe1fa467ad237802021f5aac5f1d5b3e0db47ef //58主机的id值
10. S: 894dd0008053f6fb65e9e4a36b755d9351607500 192.168.4.56:6356
11. slots: (0 slots) slave
12. replicates 324e05df3f143ef97e50d09be0328a695e655986
13. M: 2d343a9df48f6f6e207949e980ef498466a44dad 192.168.4.55:6355
14. slots:1365-5460 (4096 slots) master
15. 1 additional replica(s)
16. M: 9e44139cffb8ebd7ed746aabbf4bcea9bf207645 192.168.4.53:6353
17. slots:12288-16383 (4096 slots) master
18. 1 additional replica(s)
19. S: d9634ba0aa5c1a07193da4a013da6051c1515922 192.168.4.54:6354
20. slots: (0 slots) slave
21. replicates 9e44139cffb8ebd7ed746aabbf4bcea9bf207645
22. M: 324e05df3f143ef97e50d09be0328a695e655986 192.168.4.52:6352
23. slots:6827-10922 (4096 slots) master
24. 1 additional replica(s)
25. M: 4fe1fa467ad237802021f5aac5f1d5b3e0db47ef 192.168.4.58:6358 //主服务器
26. slots:0-1364,5461-6826,10923-12287 (4096 slots) master
27. 1 additional replica(s)
28. [OK] All nodes agree about slots configuration.
29. >>> Check for open slots...
30. >>> Check slots coverage...
31. [OK] All 16384 slots covered.
32. [root@mgm57 ~]#
33. [root@mgm57 ~]#

5）在客户端，访问从服务器59，查看数据

1. [root@host50 ~]# redis-cli -c -h 192.168.4.59 -p 6359
2. 192.168.4.59:6359> keys \* //自动同步主服务器58的数据
3. 1) "name"
4. 2) "name2"
5. 3) "age"
6. 4) "y"
7. 5) "shcool5"
8. 192.168.4.59:6359>

## 3 案例3：移除服务器

### 3.1 问题

* 把slave服务器移除集群
* 把master服务器移除集群

### 3.2 步骤

实现此案例需要按照如下步骤进行。

步骤一：把slave服务器移除集群

1）在管理主机，移除slave服务器，从服务器没有槽，直接移除即可。

1. [root@mgm57 ~]# redis-trib.rb del-node 192.168.4.51:6351 f6649ea99b2f01faca26217691222c17a3854381 //执行移除命令
2. >>> Removing node f6649ea99b2f01faca26217691222c17a3854381
3. from cluster 192.168.4.57:6351
4. >>> Sending CLUSTER FORGET messages to the cluster...
5. >>> SHUTDOWN the node. //停止移除服务的Redis服务

2）在管理主机，查看集群信息

1. [root@mgm57 ~]# redis-trib.rb info 192.168.4.51:6351
2. 192.168.4.55:6355 (2d343a9d...) -> 3 keys | 4096 slots | 1 slaves.
3. 192.168.4.53:6353 (9e44139c...) -> 3 keys | 4096 slots | 1 slaves.
4. 192.168.4.52:6352 (324e05df...) -> 2 keys | 4096 slots | 1 slaves.
5. 192.168.4.58:6358 (4fe1fa46...) -> 5 keys | 4096 slots | 0 slaves.//58主机，没有从服务器
6. [OK] 13 keys in 4 masters.
7. 0.00 keys per slot on average.
8. [root@mgm57 ~]#

步骤二：把master服务器移除集群

1）在管理主机,先删除master服务器占用的hash槽

1. [root@mgm57 ~]# redis-trib.rb reshard 192.168.4.53:6353
2. How many slots do you want to move (from 1 to 16384)?4096 //移除4096个数槽
3. What is the receiving node ID? bc5c4e082a5a3391b634cf433a6486c867cfc44b
4. //要移动给谁的id即目标主机（这里可以随机写一个master的ID）
5. Source node #1: c5e0da48f335c46a2ec199faa99b830f537dd8a0
6. //从谁那移动即源主机（这里写4.58的ID）
7. Source node #2:done         //设置完毕
8. ...
9. Moving slot 12282 from c5e0da48f335c46a2ec199faa99b830f537dd8a0
10. Moving slot 12283 from c5e0da48f335c46a2ec199faa99b830f537dd8a0
11. Moving slot 12284 from c5e0da48f335c46a2ec199faa99b830f537dd8a0
12. Moving slot 12285 from c5e0da48f335c46a2ec199faa99b830f537dd8a0
13. Moving slot 12286 from c5e0da48f335c46a2ec199faa99b830f537dd8a0
14. Moving slot 12287 from c5e0da48f335c46a2ec199faa99b830f537dd8a0
15. Do you want to proceed with the proposed reshard plan (yes/no)?yes //提交
16. ...
17. Moving slot 12282 from 192.168.4.58:6358 to 192.168.4.53:6353:
18. Moving slot 12283 from 192.168.4.58:6358 to 192.168.4.53:6353:
19. Moving slot 12284 from 192.168.4.58:6358 to 192.168.4.53:6353:
20. Moving slot 12285 from 192.168.4.58:6358 to 192.168.4.53:6353:
21. Moving slot 12286 from 192.168.4.58:6358 to 192.168.4.53:6353:
22. Moving slot 12287 from 192.168.4.58:6358 to 192.168.4.53:6353:

2）在管理主机,查看集群信息

1. [root@mgm57 ~]# redis-trib.rb info 192.168.4.51:6351
2. 192.168.4.55:6355 (2d343a9d...) -> 3 keys | 4096 slots | 1 slaves.
3. 192.168.4.53:6353 (9e44139c...) -> 3 keys | 4096 slots | 1 slaves.
4. 192.168.4.52:6352 (324e05df...) -> 2 keys | 4096 slots | 1 slaves.
5. 192.168.4.58:6358 (4fe1fa46...) -> 0 keys | 0 slots | 0 slaves. //零个槽
6. [OK] 13 keys in 4 masters.
7. 0.00 keys per slot on average.
8. [root@mgm57 ~]#

3）在管理主机，移除master主机

1. [root@mgm57 ~]# redis-trib.rb del-node 192.168.4.53:6353 \
2. c5e0da48f335c46a2ec199faa99b830f537dd8a0    //删除谁+删除的id
3. >>> Removing node e081313ec843655d9bc5a17f3bed3de1dccb1d2b from cluster 192.168.4.51:6351
4. >>> Sending CLUSTER FORGET messages to the cluster...
5. >>> SHUTDOWN the node.
6. [root@mgm57 ~]#

4）在管理主机,查看集群信息

1. [root@mgm57 ~]# redis-trib.rb info 192.168.4.51:6351
2. 192.168.4.55:6355 (2d343a9d...) -> 3 keys | 4096 slots | 1 slaves.
3. 192.168.4.53:6353 (9e44139c...) -> 3 keys | 4096 slots | 1 slaves.
4. 192.168.4.52:6352 (324e05df...) -> 2 keys | 4096 slots | 1 slaves.
5. [OK] 13 keys in 3 masters. //主服务器个数3台，没有58
6. 0.00 keys per slot on average.
7. [root@mgm57 ~]#