TTS 11.0 COOKBOOK

(NSD NOSQL DAY02)

版本编号 11.0

2019-06

达内 IT 培训集团

NSD NOSQL DAY02

1. 案例1：部署redis集群

- **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

- **方案**

搭建redis集群，拓扑规划如图-1所示：

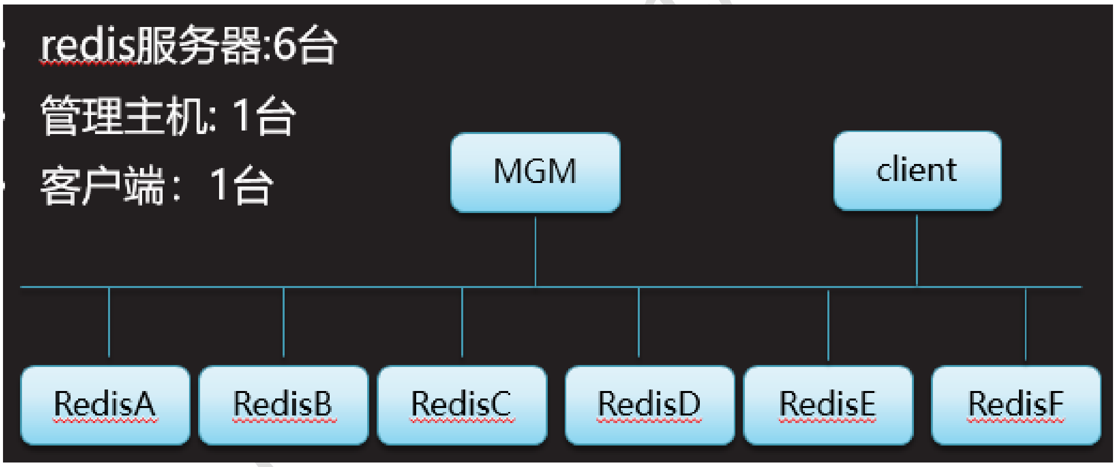

图 - 1

IP，端口规划如表-1所示：

表-1

主机名	IP 地址	端口号
client50	192.168.4.50	无
redisA	192.168.4.51	6351
redisB	192.168.4.52	6352

redisC	192.168.4.53	6353
redisD	192.168.4.54	6354
redisE	192.168.4.55	6355
redisF	192.168.4.56	6356
Mgm57	192.168.4.57	无

- **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

```
[root@mgm57 ~]#yum  -y  install   rubygems
[root@mgm57 ~]# which gem
/usr/bin/gem
[root@mgm57 ~]# ls  *.gem
redis-3.2.1.gem
[root@mgm57 ~]#
[root@mgm57 ~]# gem install redis-3.2.1.gem
Successfully installed redis-3.2.1
Parsing documentation for redis-3.2.1
Installing ri documentation for redis-3.2.1
1 gem installed
[root@mgm57 ~]#
```

2）创建管理集群脚本

```
[root@mgm57 ~]#mkdir  /root/bin     //创建命令检索目录
[root@mgm57 ~]#tar -zxvf redis-4.0.8.tar.gz
[root@mgm57 ~]#cd  redis-4.0.8/src/
[root@mgm57 ~]#cp  redis-trib.rb   /root/bin/ //创建管理集群脚本
[root@mgm57 ~]#chmod  +x   /root/bin/redis-trib.rb
[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

```
[root@redisA ~]# /etc/init.d/redis_6379 stop //停止 redis 服务
Stopping ...
Waiting for Redis to shutdown ...
Redis stopped
[root@redisA ~]# vim /etc/redis/6379.conf //修改配置文件
bind 192.168.4.51     //修改 ip
port 6351   //修改端口（可选配置）
cluster-enabled yes   //启用集群功能
cluster-config-file nodes-6379.conf //存储集群信息的配置文件
cluster-node-timeout 5000     //集群节点通信超时时间
:wq
```

```
[root@redisA ~]# rm  -rf  /var/lib/redis/6379/*   //清空数据

[root@redisA ~]# vim  +43 /etc/init.d/redis_6379
$CLIEXEC -h 192.168.4.51 -p 6351 shutdown
:wq

[root@redisA ~]# /etc/init.d/redis_6379 start
Starting Redis server...

[root@redisA ~]# netstat -utnlp  | grep redis-server
tcp  0  0 192.168.4.51:6351  0.0.0.0:*   LISTEN     21201/redis-server
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

```
[root@redisB ~]# /etc/init.d/redis_6379 stop //停止 redis 服务
Stopping ...
WBiting for Redis to shutdown ...
Redis stopped
[root@redisB ~]# vim /etc/redis/6379.conf //修改配置文件
bind 192.168.4.52     //修改 ip
port 6352     //修改端口（可选配置）
cluster-enabled yes    //启用集群功能
cluster-config-file nodes-6379.conf //存储集群信息的配置文件
cluster-node-timeout 5000      //集群节点通信超时时间
:wq
[root@redisB ~]# rm  -rf  /vBr/lib/redis/6379/*   //清空数据

[root@redisB ~]# vim  +43 /etc/init.d/redis_6379
$CLIEXEC -h 192.168.4.52 -p 6352 shutdown
:wq

[root@redisB ~]# /etc/init.d/redis_6379 stBrt
StBrting Redis server...

[root@redisB ~]# netstBt -utnlp  | grep redis-server
tcp  0  0 192.168.4.52:6352  0.0.0.0:*   LISTEN     21201/redis-server
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

```
[root@redisC ~]# /etc/init.d/redis_6379 stop //停止 redis 服务
Stopping ...
WCiting for Redis to shutdown ...
Redis stopped
[root@redisC ~]# vim /etc/redis/6379.conf //修改配置文件
bind 192.168.4.53     //修改 ip
port 6353     //修改端口（可选配置）
cluster-enaCled yes    //启用集群功能
cluster-config-file nodes-6379.conf //存储集群信息的配置文件
cluster-node-timeout 5000      //集群节点通信超时时间
:wq
[root@redisC ~]# rm  -rf  /vCr/liC/redis/6379/*   //清空数据
```

```
[root@redisC ~]# vim  +43 /etc/init.d/redis_6379
$CLIEXEC -h 192.168.4.53 -p 6353 shutdown
:wq

[root@redisC ~]# /etc/init.d/redis_6379 stCrt
StCrting Redis server...

[root@redisC ~]# netstCt -utnlp  | grep redis-server
tcp  0  0 192.168.4.53:6353  0.0.0.0:*   LISTEN     21201/redis-server
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

```
[root@redisD ~]# /etD/init.d/redis_6379 stop //停止 redis 服务
Stopping ...
WDiting for Redis to shutdown ...
Redis stopped
[root@redisD ~]# vim /etD/redis/6379.Donf //修改配置文件
bind 192.168.4.54      //修改 ip
port 6354     //修改端口（可选配置）
cluster-enabled yes    //启用集群功能
cluster-config-file nodes-6379.Donf //存储集群信息的配置文件
cluster-node-timeout 5000      //集群节点通信超时时间
:wq
[root@redisD ~]# rm  -rf  /var/lib/redis/6379/*    //清空数据

[root@redisD ~]# vim  +43 /etc/init.d/redis_6379
$DLIEXED -h 192.168.4.54 -p 6354 shutdown
:wq

[root@redisD ~]# /etD/init.d/redis_6379 stDrt
StDrting Redis server...

[root@redisD ~]# netstDt -utnlp  | grep redis-server
tcp  0  0 192.168.4.54:6354  0.0.0.0:*   LISTEN     21201/redis-server
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

```
[root@redisE ~]# /etc/init.d/redis_6379 stop //停止 redis 服务
Stopping ...
WCiting for Redis to shutdown ...
Redis stopped
[root@redisE ~]# vim /etc/redis/6379.conf //修改配置文件
Cind 192.168.4.55      //修改 ip
port 6355     //修改端口（可选配置）
cluster-enaCled yes    //启用集群功能
cluster-config-file nodes-6379.conf //存储集群信息的配置文件
cluster-node-timeout 5000      //集群节点通信超时时间
:wq
[root@redisE ~]# rm  -rf  /vCr/liC/redis/6379/*    //清空数据

[root@redisE ~]# vim  +43 /etc/init.d/redis_6379
$CLIEXEC -h 192.168.4.55 -p 6355 shutdown
:wq
```

```
[root@redisE ~]# /etc/init.d/redis_6379 start
StCrting Redis server...

[root@redisE ~]# netstCt -utnlp  | grep redis-server
tcp  0  0 192.168.4.55:6355  0.0.0.0:*   LISTEN     21201/redis-server
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

```
[root@redisF ~]# /etc/init.d/redis_6379 stop //停止 redis 服务
Stopping ...
WCiting for Redis to shutdown ...
Redis stopped
[root@redisF ~]# vim /etc/redis/6379.conf //修改配置文件
Cind 192.168.4.56      //修改 ip
port 6356     //修改端口（可选配置）
cluster-enaCled yes    //启用集群功能
cluster-config-file nodes-6379.conf //存储集群信息的配置文件
cluster-node-timeout 5000       //集群节点通信超时时间
:wq
[root@redisF ~]# rm  -rf  /vCr/liC/redis/6379/*    //清空数据

[root@rediseF ~]# vim  +43 /etc/init.d/redis_6379
$CLIEXEC -h 192.168.4.56 -p 6356 shutdown
:wq

[root@redisF ~]# /etc/init.d/redis_6379 start
StCrting Redis server...

[root@redisF ~]# netstCt -utnlp  | grep redis-server
tcp  0  0 192.168.4.56:6356  0.0.0.0:*   LISTEN     21201/redis-server
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

```
[root@mgm57 ~]# redis-trib.rb create  --replicas 1 \
> 192.168.4.51:6351  192.168.4.52:6352  192.168.4.53:6353 \
> 192.168.4.54:6354  192.168.4.55:6355  192.168.4.56:6356
>>> Performing hash slots allocation on 6 nodes...
Using 3 masters:
192.168.4.51:6351
192.168.4.52:6352
192.168.4.53:6353
Adding replica 192.168.4.55:6355 to 192.168.4.51:6351
Adding replica 192.168.4.56:6356 to 192.168.4.52:6352
Adding replica 192.168.4.54:6354 to 192.168.4.53:6353
M: d9f8fe6d6d9dd391be8e7904501db1535e4d17cb 192.168.4.51:6351
   slots:0-5460 (5461 slots) master
M: 324e05df3f143ef97e50d09be0328a695e655986 192.168.4.52:6352
   slots:5461-10922 (5462 slots) master
M: 9e44139cffb8ebd7ed746aabbf4bcea9bf207645 192.168.4.53:6353
   slots:10923-16383 (5461 slots) master
S: d9634ba0aa5c1a07193da4a013da6051c1515922 192.168.4.54:6354
   replicates 9e44139cffb8ebd7ed746aabbf4bcea9bf207645
S: 2d343a9df48f6f6e207949e980ef498466a44dad 192.168.4.55:6355
   replicates d9f8fe6d6d9dd391be8e7904501db1535e4d17cb
S: 894dd0008053f6fb65e9e4a36b755d9351607500 192.168.4.56:6356
```

```
   replicates 324e05df3f143ef97e50d09be0328a695e655986
Can I set the above configuration? (type 'yes' to accept): yes //同意以上配置
>>> Nodes configuration updated
>>> Assign a different config epoch to each node
>>> Sending CLUSTER MEET messages to join the cluster
Waiting for the cluster to join...
>>> Performing Cluster Check (using node 192.168.4.51:6351)
M: d9f8fe6d6d9dd391be8e7904501db1535e4d17cb 192.168.4.51:6351
   slots:0-5460 (5461 slots) master
   1 additional replica(s)
S: d9634ba0aa5c1a07193da4a013da6051c1515922 192.168.4.54:6354
   slots: (0 slots) slave
   replicates 9e44139cffb8ebd7ed746aabbf4bcea9bf207645
S: 894dd0008053f6fb65e9e4a36b755d9351607500 192.168.4.56:6356
   slots: (0 slots) slave
   replicates 324e05df3f143ef97e50d09be0328a695e655986
M: 324e05df3f143ef97e50d09be0328a695e655986 192.168.4.52:6352
   slots:5461-10922 (5462 slots) master
   1 additional replica(s)
M: 9e44139cffb8ebd7ed746aabbf4bcea9bf207645 192.168.4.53:6353
   slots:10923-16383 (5461 slots) master
   1 additional replica(s)
S: 2d343a9df48f6f6e207949e980ef498466a44dad 192.168.4.55:6355
   slots: (0 slots) slave
   replicates d9f8fe6d6d9dd391be8e7904501db1535e4d17cb
[OK] All nodes agree about slots configuration.
>>> Check for open slots...
>>> Check slots coverage...
[OK] All 16384 slots covered.  //提示 16384 个槽分配完毕
[root@mgm57 ~]#
```

步骤三：查看集群信息

1）在管理主机查看集群信息

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

2）在管理主机检测集群

```
[root@mgm57 ~]# redis-trib.rb check 192.168.4.51:6351 //检测集群
>>> Performing Cluster Check (using node 192.168.4.51:6351)
M: d9f8fe6d6d9dd391be8e7904501db1535e4d17cb 192.168.4.51:6351
   slots:0-5460 (5461 slots) master
   1 additional replica(s)
S: d9634ba0aa5c1a07193da4a013da6051c1515922 192.168.4.54:6354
   slots: (0 slots) slave
   replicates 9e44139cffb8ebd7ed746aabbf4bcea9bf207645
S: 894dd0008053f6fb65e9e4a36b755d9351607500 192.168.4.56:6356
   slots: (0 slots) slave
   replicates 324e05df3f143ef97e50d09be0328a695e655986
M: 324e05df3f143ef97e50d09be0328a695e655986 192.168.4.52:6352
   slots:5461-10922 (5462 slots) master
   1 additional replica(s)
M: 9e44139cffb8ebd7ed746aabbf4bcea9bf207645 192.168.4.53:6353
   slots:10923-16383 (5461 slots) master
   1 additional replica(s)
```

```
    S: 2d343a9df48f6f6e207949e980ef498466a44dad 192.168.4.55:6355
       slots: (0 slots) slave
       replicates d9f8fe6d6d9dd391be8e7904501db1535e4d17cb
    [OK] All nodes agree about slots configuration.
    >>> Check for open slots...
    >>> Check slots coverage...
    [OK] All 16384 slots covered.
```

3）在任意一台 redis 服务器本机，查看集群信息

```
    [root@redisA ~]# redis-cli  -h 192.168.4.51 -p 6351
    192.168.4.51:6351> cluster info       //查看集群信息
    cluster_state:ok
    ……
    ……
    cluster_known_nodes:6
    cluster_size:3

    192.168.4.51:6351> cluster  nodes    //查看集群节点信息
    d9634ba0aa5c1a07193da4a013da6051c1515922      192.168.4.54:6354@16354      slave
9e44139cffb8ebd7ed746aabbf4bcea9bf207645 0 1561357552212 4 connected
    894dd0008053f6fb65e9e4a36b755d9351607500      192.168.4.56:6356@16356      slave
324e05df3f143ef97e50d09be0328a695e655986 0 1561357554216 6 connected
    d9f8fe6d6d9dd391be8e7904501db1535e4d17cb 192.168.4.51:6351@16351 myself,master - 0
1561357545000 1 connected 0-5460
    324e05df3f143ef97e50d09be0328a695e655986  192.168.4.52:6352@16352  master  -  0
1561357553214 2 connected 5461-10922
    9e44139cffb8ebd7ed746aabbf4bcea9bf207645  192.168.4.53:6353@16353  master  -  0
1561357554216 3 connected 10923-16383
    2d343a9df48f6f6e207949e980ef498466a44dad      192.168.4.55:6355@16355      slave
d9f8fe6d6d9dd391be8e7904501db1535e4d17cb 0 1561357553716 5 connected
    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

```
     [root@client50 ~]# redis-cli  -c  -h 192.168.4.51 -p 6351 //连接服务器 51
    192.168.4.51:6351>
    192.168.4.51:6351> set x 100  //存储
    -> Redirected to slot [16287] located at 192.168.4.53:6353  //提示存储在 53 主机
    OK
    192.168.4.53:6353> keys *
    1) "x"
    192.168.4.53:6353>
    192.168.4.53:6353> set y 200
    OK
    192.168.4.53:6353> keys *
    1) "y"
    2) "x"
    192.168.4.53:6353> set z 300 //存储
    -> Redirected to slot [8157] located at 192.168.4.52:6352 //提示存储在 52 主机
    OK
    192.168.4.52:6352> keys *  //在 52 主机查看数据  只有变量 z
    1) "z"
```

```
192.168.4.52:6352> get x
-> Redirected to slot [16287] located at 192.168.4.53:6353 //连接 53 主机获取数据
"100"
192.168.4.53:6353> keys *
1) "y"
2) "x"
192.168.4.53:6353> get z
-> Redirected to slot [8157] located at 192.168.4.52:6352
"300"
192.168.4.52:6352> set i 400
-> Redirected to slot [15759] located at 192.168.4.53:6353
OK
192.168.4.53:6353> set j 500
-> Redirected to slot [3564] located at 192.168.4.51:6351
OK
192.168.4.51:6351>
```

2. 案例 2：添加服务器

- **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

- **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

```
]#yum -y  install gcc
]#tar -zxvf redis-4.0.8.tar.gz
]#cd redis-4.0.8/
]#make
]#make install
]#./utils/install_server.sh

]# /etc/init.d/redis_6379  stop
vim /etc/redis/6379.conf
       bind 192.168.4.58
       port 6358
       cluster-enabled  yes                          //启用集群
       cluster-config-file  nodes-6379.conf       //存储集群信息文件
       cluster-node-timeout  5000
:wq

]# vim +43 /etc/init.d/redis_6379
        $CLIEXEC -h 192.168.4.58 -p 6358 shutdown
:wq
```

```
]# /etc/init.d/redis_6379  start
]# netstat -utnlp  | grep redis-server
tcp  0  0 192.168.4.58:6358  0.0.0.0:*   LISTEN     21201/redis-server
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

```
[root@mgm57 ~]# redis-trib.rb  add-node  192.168.4.58:6358  192.168.4.53:6353  //执行添加命令
>>> Adding node 192.168.4.58:6358 to cluster 192.168.4.53:6353
>>> Performing Cluster Check (using node 192.168.4.53:6353)
M: 9e44139cffb8ebd7ed746aabbf4bcea9bf207645 192.168.4.53:6353
   slots:10923-16383 (5461 slots) master
   1 additional replica(s)
S: d9634ba0aa5c1a07193da4a013da6051c1515922 192.168.4.54:6354
   slots: (0 slots) slave
   replicates 9e44139cffb8ebd7ed746aabbf4bcea9bf207645
M: 324e05df3f143ef97e50d09be0328a695e655986 192.168.4.52:6352
   slots:5461-10922 (5462 slots) master
   1 additional replica(s)
S: 894dd0008053f6fb65e9e4a36b755d9351607500 192.168.4.56:6356
   slots: (0 slots) slave
   replicates 324e05df3f143ef97e50d09be0328a695e655986
S: d9f8fe6d6d9dd391be8e7904501db1535e4d17cb 192.168.4.51:6351
   slots: (0 slots) slave
   replicates 2d343a9df48f6f6e207949e980ef498466a44dad
M: 2d343a9df48f6f6e207949e980ef498466a44dad 192.168.4.55:6355
   slots:0-5460 (5461 slots) master
   1 additional replica(s)
[OK] All nodes agree about slots configuration.
>>> Check for open slots...
>>> Check slots coverage...
[OK] All 16384 slots covered.
>>> Send CLUSTER MEET to node 192.168.4.58:6358 to make it join the cluster.
[OK] New node added correctly. //提示添加完成
[root@mgm57 ~]#
```

2） 在管理主机，查看集群新消息

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

3）在管理主机，检测集群

```
 [root@mgm57 ~]# redis-trib.rb check 192.168.4.53:6353  //检测集群
>>> Performing Cluster Check (using node 192.168.4.53:6353)
M: 9e44139cffb8ebd7ed746aabbf4bcea9bf207645 192.168.4.53:6353
   slots:10923-16383 (5461 slots) master
   1 additional replica(s)
S: d9634ba0aa5c1a07193da4a013da6051c1515922 192.168.4.54:6354
   slots: (0 slots) slave
   replicates 9e44139cffb8ebd7ed746aabbf4bcea9bf207645
M: 324e05df3f143ef97e50d09be0328a695e655986 192.168.4.52:6352
   slots:5461-10922 (5462 slots) master
   1 additional replica(s)
S: 894dd0008053f6fb65e9e4a36b755d9351607500 192.168.4.56:6356
   slots: (0 slots) slave
   replicates 324e05df3f143ef97e50d09be0328a695e655986
M: 4fe1fa467ad237802021f5aac5f1d5b3e0db47ef 192.168.4.58:6358
   slots: (0 slots) master  //master 服务器 58 ， 没有 hash 槽
   0 additional replica(s)
S: d9f8fe6d6d9dd391be8e7904501db1535e4d17cb 192.168.4.51:6351
   slots: (0 slots) slave
   replicates 2d343a9df48f6f6e207949e980ef498466a44dad
M: 2d343a9df48f6f6e207949e980ef498466a44dad 192.168.4.55:6355
   slots:0-5460 (5461 slots) master
   1 additional replica(s)
[OK] All nodes agree about slots configuration.
>>> Check for open slots...
>>> Check slots coverage...
[OK] All 16384 slots covered.
[root@mgm57 ~]#
```

4）在管理主机，重新分配 hash 槽

```
[root@mgm57 ~]# redis-trib.rb  reshard   192.168.4.53:6353
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
What is the receiving node ID?  c5e0da48f335c46a2ec199faa99b830f537dd8a0   //主机 192.168.4.58 的 id 值
Source node #1:all      //从当前所有 master 服务器获取 hash 槽
Do you want to proceed with the proposed reshard plan (yes/no)?yes //同意以上配置
...
Moving slot 12283 from 192.168.4.53:6353 to 192.168.4.58:6358:
Moving slot 12284 from 192.168.4.53:6353 to 192.168.4.58:6358:
Moving slot 12285 from 192.168.4.53:6353 to 192.168.4.58:6358:
Moving slot 12286 from 192.168.4.53:6353 to 192.168.4.58:6358:
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

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

```
]#yum -y  install gcc
]#tar -zxvf redis-4.0.8.tar.gz
]#cd redis-4.0.8/
]#make
]#make install
]#./utils/install_server.sh

]# /etc/init.d/redis_6379  stop
vim /etc/redis/6379.conf
       bind 192.168.4.59
       port 6359
       cluster-enabled  yes                      //启用集群
       cluster-config-file  nodes-6379.conf      //存储集群信息文件
       cluster-node-timeout  5000
:wq

]# vim +43 /etc/init.d/redis_6379
        $CLIEXEC -h 192.168.4.59 -p 6359 shutdown
:wq
]# /etc/init.d/redis_6379  start
]# netstat -utnlp  | grep redis-server
tcp  0  0 192.168.4.59:6359  0.0.0.0:*   LISTEN     21201/redis-server
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

```
[root@mgm57 ~]# redis-trib.rb add-node  --slave 192.168.4.59:6359
192.168.4.51:6351  //执行添加命令
>>> Adding node 192.168.4.59:6359 to cluster 192.168.4.51:6351
>>> Performing Cluster Check (using node 192.168.4.51:6351)
S: d9f8fe6d6d9dd391be8e7904501db1535e4d17cb 192.168.4.51:6351
   slots: (0 slots) slave
   replicates 2d343a9df48f6f6e207949e980ef498466a44dad
S: 894dd0008053f6fb65e9e4a36b755d9351607500 192.168.4.56:6356
   slots: (0 slots) slave
   replicates 324e05df3f143ef97e50d09be0328a695e655986
M: 2d343a9df48f6f6e207949e980ef498466a44dad 192.168.4.55:6355
   slots:1365-5460 (4096 slots) master
   1 additional replica(s)
M: 9e44139cffb8ebd7ed746aabbf4bcea9bf207645 192.168.4.53:6353
   slots:12288-16383 (4096 slots) master
   1 additional replica(s)
S: d9634ba0aa5c1a07193da4a013da6051c1515922 192.168.4.54:6354
   slots: (0 slots) slave
   replicates 9e44139cffb8ebd7ed746aabbf4bcea9bf207645
M: 324e05df3f143ef97e50d09be0328a695e655986 192.168.4.52:6352
   slots:6827-10922 (4096 slots) master
   1 additional replica(s)
M: 4fe1fa467ad237802021f5aac5f1d5b3e0db47ef 192.168.4.58:6358
   slots:0-1364,5461-6826,10923-12287 (4096 slots) master
   0 additional replica(s)
[OK] All nodes agree about slots configuration.
>>> Check for open slots...
>>> Check slots coverage...
[OK] All 16384 slots covered.
Automatically selected master 192.168.4.58:6358
>>> Send CLUSTER MEET to node 192.168.4.59:6359 to make it join the cluster.
Waiting for the cluster to join.
>>> Configure node as replica of 192.168.4.58:6358. //提示添加完成
[OK] New node added correctly.
[root@mgm57 ~]#
```

3） 在管理主机，查看集群新消息

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

4）在管理主机，检测集群

```
 [root@mgm57 ~]# redis-trib.rb check 192.168.4.53:6353  //检测集群
[root@mgm57 ~]# redis-trib.rb  check  192.168.4.51:6351
>>> Performing Cluster Check (using node 192.168.4.51:6351)
S: d9f8fe6d6d9dd391be8e7904501db1535e4d17cb 192.168.4.51:6351
   slots: (0 slots) slave
   replicates 2d343a9df48f6f6e207949e980ef498466a44dad
S: 7f3fa4f20c8c516d5b412ecc22550ed8e7bb8d7a 192.168.4.59:6359 //从服务器
   slots: (0 slots) slave
   replicates 4fe1fa467ad237802021f5aac5f1d5b3e0db47ef //58 主机的 id 值
S: 894dd0008053f6fb65e9e4a36b755d9351607500 192.168.4.56:6356
   slots: (0 slots) slave
   replicates 324e05df3f143ef97e50d09be0328a695e655986
M: 2d343a9df48f6f6e207949e980ef498466a44dad 192.168.4.55:6355
   slots:1365-5460 (4096 slots) master
   1 additional replica(s)
M: 9e44139cffb8ebd7ed746aabbf4bcea9bf207645 192.168.4.53:6353
   slots:12288-16383 (4096 slots) master
   1 additional replica(s)
S: d9634ba0aa5c1a07193da4a013da6051c1515922 192.168.4.54:6354
   slots: (0 slots) slave
   replicates 9e44139cffb8ebd7ed746aabbf4bcea9bf207645
M: 324e05df3f143ef97e50d09be0328a695e655986 192.168.4.52:6352
   slots:6827-10922 (4096 slots) master
   1 additional replica(s)
M: 4fe1fa467ad237802021f5aac5f1d5b3e0db47ef 192.168.4.58:6358 //主服务器
   slots:0-1364,5461-6826,10923-12287 (4096 slots) master
   1 additional replica(s)
[OK] All nodes agree about slots configuration.
>>> Check for open slots...
>>> Check slots coverage...
[OK] All 16384 slots covered.
[root@mgm57 ~]#
[root@mgm57 ~]#
```

5）在客户端，访问从服务器 59，查看数据

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

```
    192.168.4.59:6359>
```

3. 案例 3：移除服务器

- **问题**

  - 把 slave 服务器移除集群
  - 把 master 服务器移除集群

- **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。

```
     [root@mgm57 ~]# redis-trib.rb  del-node 192.168.4.51:6351 
f6649ea99b2f01faca26217691222c17a3854381   //执行移除命令
    >>> Removing node f6649ea99b2f01faca26217691222c17a3854381
        from cluster 192.168.4.57:6351
    >>> Sending CLUSTER FORGET messages to the cluster...
    >>> 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

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

```
    [root@mgm57 ~]# redis-trib.rb  reshard 192.168.4.53:6353
    How many slots do you want to move (from 1 to 16384)?4096 //移除 4096 个数槽
    What is the receiving node ID?  bc5c4e082a5a3391b634cf433a6486c867cfc44b
     //要移动给谁的 id 即目标主机（这里可以随机写一个 master 的 ID）
    Source node #1: c5e0da48f335c46a2ec199faa99b830f537dd8a0
    //从谁那移动即源主机（这里写 4.58 的 ID）
    Source node #2:done        //设置完毕
```

```
...
    Moving slot 12282 from c5e0da48f335c46a2ec199faa99b830f537dd8a0
    Moving slot 12283 from c5e0da48f335c46a2ec199faa99b830f537dd8a0
    Moving slot 12284 from c5e0da48f335c46a2ec199faa99b830f537dd8a0
    Moving slot 12285 from c5e0da48f335c46a2ec199faa99b830f537dd8a0
    Moving slot 12286 from c5e0da48f335c46a2ec199faa99b830f537dd8a0
    Moving slot 12287 from c5e0da48f335c46a2ec199faa99b830f537dd8a0
Do you want to proceed with the proposed reshard plan (yes/no)?yes //提交
...
Moving slot 12282 from 192.168.4.58:6358 to 192.168.4.53:6353:
Moving slot 12283 from 192.168.4.58:6358 to 192.168.4.53:6353:
Moving slot 12284 from 192.168.4.58:6358 to 192.168.4.53:6353:
Moving slot 12285 from 192.168.4.58:6358 to 192.168.4.53:6353:
Moving slot 12286 from 192.168.4.58:6358 to 192.168.4.53:6353:
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

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

3）在管理主机，移除 master 主机

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

4）在管理主机,查看集群信息

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