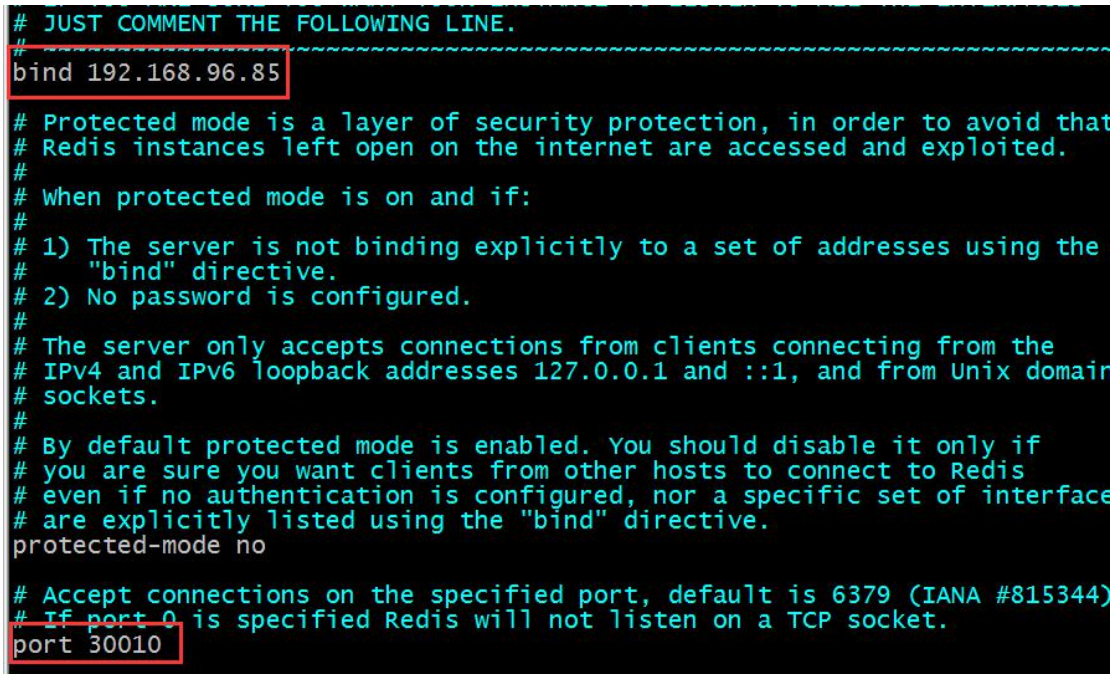
redis 集群部署手册

redis 集群部署有三种方式：redis 单机部署、伪集群部署（一台服务器上部署集群，启用不用端口）、集群部署（不同服务器搭建集群）

一、单机部署

解压 redis 单机安装包 redis-3.2.0.tar.gz 到 redis 安装目录，修改 IP、端口后启动 redis 即可。修改 IP 端口如下图：



或者用系统安装包中的 redis 安装脚本一键安装（~/Thinkit_Offline_SystemE/System_Offline_Server/tools/AutomaticInstall/redis_install.sh）.修改完 IP 端口之后用 nohup ./redis-server ./redis.conf >/dev/null 2>&1 &　启动 redis。

二、集群部署在一台机器上

1.进入~/local/redis/redis-3.2.0 目录；

2.修改 cluster/7000.conf 等文件中的 IP 和 port，如下图：

```
# JUST COMMENT THE FOLLOWING LINE.
# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
bind 192.168.96.85

# Protected mode is a layer of security protection, in order to avoid that
# Redis instances left open on the internet are accessed and exploited.
#
# When protected mode is on and if:
#
# 1) The server is not binding explicitly to a set of addresses using the
#    "bind" directive.
# 2) No password is configured.
#
# The server only accepts connections from clients connecting from the
# IPv4 and IPv6 loopback addresses 127.0.0.1 and ::1, and from Unix domain
# sockets.
#
# By default protected mode is enabled. You should disable it only if
# you are sure you want clients from other hosts to connect to Redis
# even if no authentication is configured, nor a specific set of interface
# are explicitly listed using the "bind" directive.
protected-mode no

# Accept connections on the specified port, default is 6379 (IANA #815344)
# If port 0 is specified Redis will not listen on a TCP socket.
port 30010
```

3. 后台启动 redis，执行 nohup ./redis-server cluster/7000.conf >/dev/null 2>&1 &启动不同端口。7001-7005 端口启动方法同 7000.

4. 使用 cluster meet ip port 与各节点握手：

1）使用 ./redis-cli -h 192.168.96.88(ip) -p 7000(port) -c -a foobared(redis 密码) 进入 redis，如下：

```
[houshasha@localhost redis-3.2.0]$ ./redis-cli -h 192.168.96.88 -p 7000 -c -a foobared
192.168.96.88:7000>
```

2）输入 cluster meet ip port，如：cluster meet 192.168.96.88 7001; cluster meet 192.168.96.88 7002 等等，如下图：

```
[houshasha@localhost redis-3.2.0]$ ./redis-cli -h 192.168.96.88 -p 7000 -c -a foobared
192.168.96.88:7000> cluster meet 192.168.96.88 7001
OK
192.168.96.88:7000> 15658:M 21 May 17:54:01.184 # IP address for this node updated to 192.168.96.88
                                                                                15655:M 21 May 17:54:01.285 # IP address for
 this node updated to 192.168.96.88
192.168.96.88:7000>
192.168.96.88:7000> cluster meet 192.168.96.88 7002
OK
192.168.96.88:7000> 15661:M 21 May 17:54:04.991 # IP address for this node updated to 192.168.96.88
192.168.96.88:7000>
192.168.96.88:7000> cluster meet 192.168.96.88 7003
OK
192.168.96.88:7000> 15664:M 21 May 17:54:07.094 # IP address for this node updated to 192.168.96.88
192.168.96.88:7000>
192.168.96.88:7000> cluster meet 192.168.96.88 7004
OK
192.168.96.88:7000> 15669:M 21 May 17:54:08.896 # IP address for this node updated to 192.168.96.88
192.168.96.88:7000>
192.168.96.88:7000> cluster meet 192.168.96.88 7005
OK
192.168.96.88:7000> 15672:M 21 May 17:54:10.997 # IP address for this node updated to 192.168.96.88
```

3）查看节点信息，输入 cluster nodes，如下图：

```
192.168.96.88:7000> cluster nodes
cfbc44bf1569c349ed2733ecd950ca3d57b7aae2 192.168.96.88:7004 master - 0 1558433713614 0 connected
7cbcbd7840389a8c532ecd8dc7c79f79088ccc83 192.168.96.88:7000 myself,master - 0 0 1 connected
8e9b51c36f137a94e16a8d6c95b81653f9f1618b 192.168.96.88:7001 master - 0 1558433712612 4 connected
2c0625d400d86baae60b9f1e6e61e4b590c276b6 192.168.96.88:7002 master - 0 1558433710608 3 connected
a653d99577df95ccfa87ffa1eefd29eb653a4e26 192.168.96.88:7005 master - 0 1558433711610 5 connected
6703cf83557ccf9dda05e783d7353ce22c7084ae 192.168.96.88:7003 master - 0 1558433709606 2 connected
```

说明握手成功。

5. 给 redis 主节点分配槽值，如下图：

./redis-cli -h 192.168.96.88 -p 7000 -a foobared cluster addslots {0..5460}

./redis-cli -h 192.168.96.88 -p 7001 -a foobared cluster addslots {5461..10922}

./redis-cli -h 192.168.96.88 -p 7002 -a foobared cluster addslots {10923..16383}

```
[houshasha@localhost redis-3.2.0]$ ./redis-cli -h 192.168.96.88 -p 7000 -a foobared cluster addslots {0..5460}
OK
[houshasha@localhost redis-3.2.0]$ ./redis-cli -h 192.168.96.88 -p 7001 -a foobared cluster addslots {5461..10922}
OK
[houshasha@localhost redis-3.2.0]$ ./redis-cli -h 192.168.96.88 -p 7002 -a foobared cluster addslots {10923..16383}
16307:M 21 May 18:15:51.948 # Cluster state changed: ok
OK
```

3个主节点　　每个主节点对应的槽值

分别给 3 个主节点分配槽值。

然后，进入 redis 查看节点信息：

```
[houshasha@localhost redis-3.2.0]$ ./redis-cli -h 192.168.96.88 -p 7000 -c -a foobared
192.168.96.88:7000> cluster nodes
cfbc44bf1569c349ed2733ecd950ca3d57b7aae2 192.168.96.88:7004 master - 0 1558436931054 0 connected
7cbcbd7840389a8c532ecd8dc7c79f79088ccc83 192.168.96.88:7000 myself,master - 0 0 1 connected 0-5460
8e9b51c36f137a94e16a8d6c95b81653f9f1618b 192.168.96.88:7001 master - 0 1558436932056 4 connected 5461-10922
2c0625d400d86baae60b9f1e6e61e4b590c276b6 192.168.96.88:7002 master - 0 1558436929053 3 connected 10923-16383
a653d99577df95ccfa87ffa1eefd29eb653a4e26 192.168.96.88:7005 master - 0 1558436933058 5 connected
6703cf83557ccf9dda05e783d7353ce22c7084ae 192.168.96.88:7003 master - 0 1558436934060 2 connected
```

每个主节点后面会有对应的槽值。

输入 cluster info，查看 cluster 状态，为 OK，如下图：

```
192.168.96.88:7000> cluster info
cluster_state:ok
cluster_slots_assigned:16384
cluster_slots_ok:16384
cluster_slots_pfail:0
cluster_slots_fail:0
cluster_known_nodes:6
cluster_size:3
cluster_current_epoch:5
cluster_my_epoch:1
cluster_stats_messages_sent:4025
cluster_stats_messages_received:4025
```

6. 使用 cluster replicate 进行集群映射

1）使用 ./redis-cli -h 192.168.96.88(ip) -p 7000(port) -c -a foobared(redis 密码) 进入 redis，如下：

```
[houshasha@localhost redis-3.2.0]$ ./redis-cli -h 192.168.96.88 -p 7000 -c -a foobared
192.168.96.88:7000>
```

2）输入 cluster nodes 查看节点信息：

```
[houshasha@localhost redis-3.2.0]$ ./redis-cli -h 192.168.96.88 -p 7000 -c -a foobared
192.168.96.88:7000> cluster nodes
cfbc44bf1569c349ed2733ecd950ca3d57b7aae2 192.168.96.88:7004 master - 0 1558436931054 0 connected
7cbcbd7840389a8c532ecd8dc7c79f79088ccc83 192.168.96.88:7000 myself,master - 0 0 1 connected 0-5460
8e9b51c36f137a94e16a8d6c95b81653f9f1618b 192.168.96.88:7001 master - 0 1558436932056 4 connected 5461-10922
2c0625d400d86baae60b9f1e6e61e4b590c276b6 192.168.96.88:7002 master - 0 1558436929053 3 connected 10923-16383
a653d99577df95ccfa87ffa1eefd29eb653a4e26 192.168.96.88:7005 master - 0 1558436933058 5 connected
6703cf83557ccf9dda05e783d7353ce22c7084ae 192.168.96.88:7003 master - 0 1558436934060 2 connected
192.168.96.88:7000>
```

3）分别进入 3 个从节点的 redis，然后使用命令：cluster replicate 主节点 id。

如进入 7003 节点，然后 cluster replicate 7cbcbd7840389a8c532ecd8dc7c79f79088ccc83，则 7003 节点就是 7000 节点的从节点。如下图：

```
[houshasha@localhost redis-3.2.0]$ ./redis-cli -h 192.168.96.88 -p 7003 -c -a foobared
192.168.96.88:7003> cluster replicate 7cbcbd7840389a8c532ecd8dc7c79f79088ccc83
OK
```

4）查看节点信息：

```
192.168.96.88:7003> cluster nodes
cfbc44bf1569c349ed2733ecd950ca3d57b7aae2 192.168.96.88:7004 master - 0 1558437064642 0 connected
6703cf83557ccf9dda05e783d7353ce22c7084ae 192.168.96.88:7003 myself,slave 7cbcbd7840389a8c532ecd8dc7c79f79088ccc83 0 0 2 connected
7cbcbd7840389a8c532ecd8dc7c79f79088ccc83 192.168.96.88:7000 master - 0 1558437065643 1 connected 0-5460
a653d99577df95ccfa87ffa1eefd29eb653a4e26 192.168.96.88:7005 master - 0 1558437066645 5 connected
2c0625d400d86baae60b9f1e6e61e4b590c276b6 192.168.96.88:7002 master - 0 1558437059633 3 connected 10923-16383
8e9b51c36f137a94e16a8d6c95b81653f9f1618b 192.168.96.88:7001 master - 0 1558437063641 4 connected 5461-10922
```

7003 节点是 7000 的从节点。

然后进入 7004，设置为 7001 的从节点，进入 7005，设置为 7002 的从节点，如下图：

```
192.168.96.88:7005> cluster nodes
2c0625d400d86baae60b9f1e6e61e4b590c276b6 192.168.96.88:7002 master - 0 1558437204796 3 connected 10923-16383
a653d99577df95ccfa87ffa1eefd29eb653a4e26 192.168.96.88:7005 myself,slave 2c0625d400d86baae60b9f1e6e61e4b590c276b6 0 0 5 connected
7cbcbd7840389a8c532ecd8dc7c79f79088ccc83 192.168.96.88:7000 master - 0 1558437206800 1 connected 0-5460
6703cf83557ccf9dda05e783d7353ce22c7084ae 192.168.96.88:7003 slave 7cbcbd7840389a8c532ecd8dc7c79f79088ccc83 0 1558437207800 2 connected
cfbc44bf1569c349ed2733ecd950ca3d57b7aae2 192.168.96.88:7004 slave 8e9b51c36f137a94e16a8d6c95b81653f9f1618b 0 1558437205798 4 connected
8e9b51c36f137a94e16a8d6c95b81653f9f1618b 192.168.96.88:7001 master - 0 1558437203794 4 connected 5461-10922
```

7. 查看 redis 是否启动：

ps -ef|grep redis

如有以下内容，说明启动成功。

```
[houshasha@localhost redis-3.2.0]$ ps -ef|grep redis
houshas+  16301  40418  0 18:14 pts/3    00:00:01 ./redis-server 192.168.96.88:7000 [cluster]
houshas+  16304  40418  0 18:14 pts/3    00:00:01 ./redis-server 192.168.96.88:7001 [cluster]
houshas+  16307  40418  0 18:14 pts/3    00:00:01 ./redis-server 192.168.96.88:7002 [cluster]
houshas+  16310  40418  0 18:14 pts/3    00:00:01 ./redis-server 192.168.96.88:7003 [cluster]
houshas+  16314  40418  0 18:14 pts/3    00:00:01 ./redis-server 192.168.96.88:7004 [cluster]
houshas+  16317  40418  0 18:14 pts/3    00:00:01 ./redis-server 192.168.96.88:7005 [cluster]
houshas+  16921  40418  0 18:46 pts/3    00:00:00 grep --color=auto redis
```

三、集群部署在多台机器上（3 台 6 节点）

1.进入~/local/redis/redis-3.2.0 目录；

2.修改 cluster/7000.conf 7001.conf 文件中的 IP 和 port，如下图：

```
bind 192.168.96.85

# Protected mode is a layer of security protection, in order to avoid that
# Redis instances left open on the internet are accessed and exploited.
#
# When protected mode is on and if:
#
# 1) The server is not binding explicitly to a set of addresses using the
#    "bind" directive.
# 2) No password is configured.
#
# The server only accepts connections from clients connecting from the
# IPv4 and IPv6 loopback addresses 127.0.0.1 and ::1, and from Unix domain
# sockets.
#
# By default protected mode is enabled. You should disable it only if
# you are sure you want clients from other hosts to connect to Redis
# even if no authentication is configured, nor a specific set of interfaces
# are explicitly listed using the "bind" directive.
protected-mode no

# Accept connections on the specified port, default is 6379 (IANA #815344).
# If port 0 is specified Redis will not listen on a TCP socket.
port 8888
```

3. 85 后台启动 redis，执行

nohup ./redis-server cluster/7000.conf >/dev/null 2>&1 &

nohup ./redis-server cluster/7000.conf >/dev/null 2>&1 &

```
[wangchaofeng@localhost redis-3.2.0]$ nohup ./redis-server cluster/7000.conf  >/dev/null 2>&1 &
[1] 67413
[wangchaofeng@localhost redis-3.2.0]$ nohup ./redis-server cluster/7001.conf  >/dev/null 2>&1 &
[2] 67719
[wangchaofeng@localhost redis-3.2.0]$ ps -ef|grep redis
wangcha+  67413  88071  0 18:24 pts/1    00:00:00 ./redis-server 192.168.96.86:8888 [cluster]
wangcha+  67719  88071  0 18:24 pts/1    00:00:00 ./redis-server 192.168.96.86:8889 [cluster]
wangcha+  68114  88071  0 18:24 pts/1    00:00:00 grep --color=auto redis
```

4. 86/87 服务器上一次按照 1.2.3 步骤分别后台启动 redis

5.使用 cluster meet ip port 与各节点握手：

1）使用 ./redis-cli -h 192.168.96.85(ip) -p 8888(port) -c -a foobared(redis 密码) 进入 redis，如下：

```
[wangchaofeng@localhost redis-3.2.0]$ ./redis-cli -h 192.168.96.85 -p 8888 -c -a foobared
192.168.96.85:8888>
```

2）输入 cluster meet ip port，如：cluster meet 192.168.96.86 8888; cluster meet 192.168.96.87 8888; cluster meet 192.168.96.85 8889;cluster meet 192.168.96.86 8889; cluster meet 192.168.96.87 8889;等等，如下图：

```
[wangchaofeng@localhost redis-3.2.0]$ ./redis-cli -h 192.168.96.85 -p 8888 -c -a foobared
192.168.96.85:8888> cluster meet 192.168.96.86 8888
OK
192.168.96.85:8888> cluster meet 192.168.96.87 8888
OK
192.168.96.85:8888> cluster meet 192.168.96.85 8889
OK
192.168.96.85:8888> cluster meet 192.168.96.86 8889
OK
192.168.96.85:8888> cluster meet 192.168.96.87 8889
OK
192.168.96.85:8888>
```

3）查看节点信息，输入 cluster nodes，如下图：

```
192.168.96.85:8888> cluster nodes
b019abb47ec7b1c323b7d2427fde1f3489954e50 192.168.96.86:8889 master - 0 1591009136666 5 connected
13c5f80dc42e3645996ecc0a929ae057881dc0a0 192.168.96.87:8889 master - 0 1591009136967 6 connected
3fd830664b7b4e07393b052b5cd8d3db34475926 192.168.96.87:8888 master - 0 1591009136967 2 connected
eea8559e0fe80e7b33c938fb9c6fc119833c1364 192.168.96.86:8888 master - 0 1591009136967 1 connected
8c16f803ba79d9c754a36f41361efd8c631dd137 192.168.96.85:8889 master - 0 1591009136666 4 connected
f71a51cb2f8733ccc2b49d046fcb6337789fa3ac 192.168.96.85:8888 myself,master - 0 0 0 connected
```

说明握手成功。

6.给 redis 主节点（85/86/87 位主节点）分配槽值，如下图：

分别进入 85/86/87 服务器执行：

./redis-cli -h 192.168.96.85 -p 8888 -a foobared cluster addslots {0..5460}

./redis-cli -h 192.168.96.86 -p 8888 -a foobared cluster addslots {5461..10922}

./redis-cli -h 192.168.96.87 -p 8888 -a foobared cluster addslots {10923..16383}

```
[wangchaofeng@localhost redis-3.2.0]$ ./redis-cli -h 192.168.96.85 -p 8888 -a foobared cluster addslots {0..5460}
OK
```

```
[wangchaofeng@localhost redis-3.2.0]$ ./redis-cli -h 192.168.96.86 -p 8888 -a foobared cluster addslots {5461..10922}
OK
```

```
[guoyonglei@localhost redis-3.2.0]$ ./redis-cli -h 192.168.96.87 -p 8888 -a foobared cluster addslots {10923..16383}
OK
```

分别给 3 个主节点分配槽值。

然后，进入 redis 查看节点信息：

```
192.168.96.85:8888> cluster nodes
b019abb47ec7b1c323b7d2427fde1f3489954e50 192.168.96.86:8889 master - 0 1591009333152 5 connected
13c5f80dc42e3645996ecc0a929ae057881dc0a0 192.168.96.87:8889 master - 0 1591009333252 6 connected
3fd830664b7b4e07393b052b5cd8d3db34475926 192.168.96.87:8888 master - 0 1591009333452 2 connected 10923-16383
eea8559e0fe80e7b33c938fb9c6fc119833c1364 192.168.96.86:8888 master - 0 1591009333152 1 connected 5461-10922
8c16f803ba79d9c754a36f41361efd8c631dd137 192.168.96.85:8889 master - 0 1591009333451 4 connected
f71a51cb2f8733ccc2b49d046fcb6337789fa3ac 192.168.96.85:8888 myself,master - 0 0 0 connected 0-5460
192.168.96.85:8888>
```

每个主节点后面会有对应的槽值。

输入 cluster info，查看 cluster 状态，为 OK，如下图：

```
192.168.96.85:8888> cluster info
cluster_state:ok
cluster_slots_assigned:16384
cluster_slots_ok:16384
cluster_slots_pfail:0
cluster_slots_fail:0
cluster_known_nodes:6
cluster_size:3
cluster_current_epoch:6
cluster_my_epoch:0
cluster_stats_messages_sent:61315
cluster_stats_messages_received:30855
```

7.使用 cluster replicate 进行集群映射

1）使用 ./redis-cli -h 192.168.96.85(ip) -p 8888(port) -c -a foobared(redis 密码) 进入 redis，如下：

```
[wangchaofeng@localhost redis-3.2.0]$ ./redis-cli -h 192.168.96.85 -p 8888 -c -a foobared
192.168.96.85:8888>
```

2）输入 cluster nodes 查看节点信息：

```
192.168.96.85:8888> cluster nodes
b019abb47ec7b1c323b7d2427fde1f3489954e50 192.168.96.86:8889 master - 0 1591009465157 5 connected
13c5f80dc42e3645996ecc0a929ae057881dc0a0 192.168.96.87:8889 master - 0 1591009465458 6 connected
3fd830664b7b4e07393b052b5cd8d3db34475926 192.168.96.87:8888 master - 0 1591009465258 2 connected 10923-16383
eea8559e0fe80e7b33c938fb9c6fc119833c1364 192.168.96.86:8888 master - 0 1591009465258 1 connected 5461-10922
8c16f803ba79d9c754a36f41361efd8c631dd137 192.168.96.85:8889 master - 0 1591009465458 4 connected
f71a51cb2f8733ccc2b49d046fcb6337789fa3ac 192.168.96.85:8888 myself,master - 0 0 0 connected 0-5460
```

3）分别进入 3 个从节点的 redis，然后使用命令：cluster replicate 主节点 id。

如进入 86：8889 节点，然后 cluster replicate f71a51cb2f8733ccc2b49d046fcb6337789fa3ac，则 86：8889 节点就是 85：8888 节点的从节点。如下图：

```
[wangchaofeng@localhost redis-3.2.0]$ ./redis-cli -h 192.168.96.86 -p 8889 -c -a foobared
192.168.96.86:8889> cluster replicate f71a51cb2f8733ccc2b49d046fcb6337789fa3ac
OK
192.168.96.86:8889>
```

5）查看节点信息：

```
192.168.96.85:8888> cluster nodes
b019abb47ec7b1c323b7d2427fde1f3489954e50 192.168.96.86:8889 slave f71a51cb2f8733ccc2b49d046fcb6337789fa3ac 0 1591009636809 5 connected
13c5f80dc42e3645996ecc0a929ae057881dc0a0 192.168.96.87:8889 master - 0 1591009636709 6 connected
3fd830664b7b4e07393b052b5cd8d3db34475926 192.168.96.87:8888 master - 0 1591009637110 2 connected 10923-16383
eea8559e0fe80e7b33c938fb9c6fc119833c1364 192.168.96.86:8888 master - 0 1591009637110 1 connected 5461-10922
8c16f803ba79d9c754a36f41361efd8c631dd137 192.168.96.85:8889 master - 0 1591009636709 4 connected
f71a51cb2f8733ccc2b49d046fcb6337789fa3ac 192.168.96.85:8888 myself,master - 0 0 0 connected 0-5460
192.168.96.85:8888>
```

86：8889 节点是 85：8888 的从节点。

然后进:87：8889，设置为 86:8888 的从节点，进入 85:8889，设置为 87:8888 的从节点，如下图：

```
192.168.96.85:8888> cluster nodes
b019abb47ec7b1c323b7d2427fde1f3489954e50 192.168.96.86:8889 slave f71a51cb2f8733ccc2b49d046fcb6337789fa3ac 0 1591009817487 5 connected
13c5f80dc42e3645996ecc0a929ae057881dc0a0 192.168.96.87:8889 slave eea8559e0fe80e7b33c938fb9c6fc119833c1364 0 1591009817387 6 connected
3fd830664b7b4e07393b052b5cd8d3db34475926 192.168.96.87:8888 master - 0 1591009817788 2 connected 10923-16383
eea8559e0fe80e7b33c938fb9c6fc119833c1364 192.168.96.86:8888 master - 0 1591009817688 1 connected 5461-10922
8c16f803ba79d9c754a36f41361efd8c631dd137 192.168.96.85:8889 slave 3fd830664b7b4e07393b052b5cd8d3db34475926 0 1591009817788 4 connected
f71a51cb2f8733ccc2b49d046fcb6337789fa3ac 192.168.96.85:8888 myself,master - 0 0 0 connected 0-5460
192.168.96.85:8888>
```

8. 查看 redis 是否启动：

三台机器上分别查看：ps -ef|grep redis

如有以下内容，说明启动成功。

```
[wangchaofeng@localhost redis-3.2.0]$ ps -ef|grep redis
wangcha+  67413  88071  0 18:24 pts/1    00:00:05 ./redis-server 192.168.96.86:8888 [cluster]
wangcha+  67719  88071  0 18:24 pts/1    00:00:04 ./redis-server 192.168.96.86:8889 [cluster]
```