Guide for Setting Up Redis Cluster

12/2/2017

Contents

[1. 硬件参数 3](#_Toc479691253)

[2. 信息收集 3](#_Toc479691254)

[3. 开始之前 3](#_Toc479691255)

[3.1. 所需文件 3](#_Toc479691256)

[3.2. 所需工具 4](#_Toc479691257)

[4. 安装前置 4](#_Toc479691258)

[4.1. 创建redis用户 4](#_Toc479691259)

[4.2. 更新Hostname 4](#_Toc479691260)

[4.3. 安装前置(非必要，生产用户不允许用root用户运行) 4](#_Toc479691261)

[5. 安装Redis 5](#_Toc479691262)

[5.1. 准备 Tarball 5](#_Toc479691263)

[5.2. 编译Redis前准备 5](#_Toc479691264)

[5.3. 解压Tarball 编译Redis可执行文件 5](#_Toc479691265)

[Troubleshooting 6](#_Toc479691266)

[5.4. 建立Redis文件夹结构 6](#_Toc479691267)

[6. 配置Redis 6](#_Toc479691268)

[6.1. 配置Redis启动脚本 6](#_Toc479691269)

[6.2. 配置 Redis master节点 7](#_Toc479691270)

[6.3. 配置Redis slave节点 7](#_Toc479691271)

[6.4. 将Redis添加到开机启动服务 8](#_Toc479691272)

[7. 建立防火墙 8](#_Toc479691273)

[7.1. 更新防火墙配置 8](#_Toc479691274)

[7.2. 检查防火墙状态 8](#_Toc479691275)

[8. 验证Redis安装 9](#_Toc479691276)

[9. 配置 Sentinel 10](#_Toc479691277)

[9.1. 配置sentinel配置文件 10](#_Toc479691278)

[9.2. 将 Sentinel添加到开机启动服务 10](#_Toc479691279)

[9.3. 启动 Sentinel服务 11](#_Toc479691280)

[9.4. 检查Sentinel状态 11](#_Toc479691281)

[9.5. Redis Security 12](#_Toc479691282)

[10. Troubleshooting 13](#_Toc479691283)

# 硬件参数

|  |  |  |
| --- | --- | --- |
| **Item** | **Specification** | **Comments** |
| RAM | 64GB |  |
| CPU | 8 Cores |  |
| OS Disk | 20 GB |  |
| Log Disk | 127 GB | /var/log/redis |
| Data Disk | 200 GB | /var/lib/redis |
| Swap Disk | 128 GB |  |

# 信息收集

Fill the form below

|  |  |  |  |
| --- | --- | --- | --- |
| **Item** | **Key** | **Value** | **Comments** |
| Redis Node 1 | IP Address |  | Master node |
| Redis Node 2 | IP Address |  | Slave node |
| Redis Node 3 | IP Address |  | Slave node |
| App Server 1 | IP Address |  |  |
| App Server 2 | IP Address |  |  |

# 开始之前

## 所需文件

请确保你获得以下文件:

* ubuntu-16.04.1-server-amd64.iso
* redis-3.2.6.tar.gz
* redis-chmod-chown.sh
* 6379.conf
* sentinel.conf
* sentinel
* sentinel-start.sh
* redis
* redis-start.sh
* redis-stop.sh
* del-redis-pid

## 所需工具

请准备好以下工具:

* WinSCP (<http://winscp.net/> )
* PuTTY (<http://www.putty.org/>)
* Notepad ++ (可选, <https://notepad-plus-plus.org/>) (WinSCP和Notepad++ 的集成可参见 <https://winscp.net/eng/docs/integration_editor>)

# 安装前置

本说明以以下假设为前提: 服务器满足硬件参数条件; Linux HyperV虚拟机集成服务或者VmWare虚拟机的VmWare Tools已安装.

关于Linux集成服务的更多信息请参阅:

<http://blogs.technet.com/b/virtualization/archive/2015/05/01/linux-integration-services-4-0-announcement.aspx>

## 创建redis用户

sudo adduser **redis**

## 更新Hostname

hostnamectl set-hostname Redis\_Node\_1

例如:

hostnamectl set-hostname **Redis**

## 安装前置(非必要，生产用户不允许用root用户运行)

Update apt-get：sudo apt-get update

安装 open ssh：sudo apt-get install openssh-server

辑配置文件，允许以 root 用户通过 ssh 登录：

sudo vi /etc/ssh/sshd\_config

找到：PermitRootLogin prohibit-password禁用

添加：PermitRootLogin yes

sudo service ssh restart

安装后弹出DVD

# 安装Redis

## 准备 Tarball

使用WinSCP上传“redis-3.2.6.tar.gz” 到路径**~/home/deploy（）**

## 编译Redis前准备

如果系统没安装build-essential：

sudo apt-get update

sudo apt-get install build-essential

安装**tcl**

wget http://downloads.sourceforge.net/tcl/tcl8.6.1-src.tar.gz

sudo tar xzvf tcl8.6.1-src.tar.gz -C /usr/local/

cd /usr/local/tcl8.6.1/unix/

sudo ./configure

sudo make

sudo make install

## 解压Tarball 编译Redis可执行文件

切换到redis用户：su redis

cd ~

tar zxvf redis-3.2.6.tar.gz

cd redis-3.2.6

cd deps

make hiredis lua jemalloc linenoise

cd ..

make

make test

test运行过程中不应该出现任何错误.

**Troubleshooting**

机器性能如果不足，会测试失败，解决方法如下：

更改 tests/integration/replication-psync.tcl 文件：

vi tests/integration/replication-psync.tcl

把\*\* [err]: Test replication partial resync: ok psync (diskless: yes, reconnect:

的 after后面的数字，从100改成 500。

再次执行就不会有这样的问题了。

## 建立Redis文件夹结构

su deploy

sudo cp src/redis-server /usr/local/bin

sudo cp src/redis-cli /usr/local/bin

sudo cp src/redis-sentinel /usr/local/bin

sudo mkdir /etc/redis

sudo mkdir /var/log/redis/

sudo mkdir /var/lib/redis

sudo cd ..

sudo rm -v -f redis-3.2.6.tar.gz

sudo rm -v -r -f redis-3.2.6

# 配置Redis

## 配置Redis启动脚本

将如下文件拷贝到/etc/init.d下：

* sentinel
* sentinel-start.sh
* redis
* redis-start.sh
* redis-stop.sh
* del-redis-pid

将如下文件拷贝到/etc/redis下：

* 6379.conf
* sentinel.conf

注意：文件由windows 移动到ubuntu会无法执行，解决方法：

vi该文件 在命令模式下输入 :set ff=unix 回车

例如 a.sh

vi a.sh

进入输入 :set ff=unix 回车

输入:wq 回车

执行以下文件：

* redis-chmod-chown.sh

注意：给redis-chmod-chown.sh执行权限：sudo chmod 700 /etc/redis/ redis-chmod-chown.sh；使用wincp可能会导致文件格式改变，打开文件编辑，对比copy前后，确保文件没有变化

## 配置 Redis master节点

在master节点上,编辑 “/etc/redis/6379.conf”

daemonize yes

pidfile /var/run/redis\_6379.pid

tcp-keepalive 60

logfile "/var/log/redis/redis.log"

dir /var/lib/redis

maxmemory-policy noeviction

appendonly yes

## 配置Redis slave节点

在slave节点上, 编辑“/etc/redis/6379.conf”

daemonize yes

pidfile /var/run/redis\_6379.pid

tcp-keepalive 60

logfile "/var/log/redis/redis.log"

dir /var/lib/redis

slaveof master\_node\_ip\_addr 6379

slave-serve-stale-data no

maxmemory-policy noeviction

appendonly yes

# 配置 Sentinel

使用WinSCP上传“sentinel”到路径“/etc/init.d/”

使用WinSCP上传“sentinel.conf”到路径“/etc/redis/”

## 配置sentinel配置文件

在所有的redis服务器上, 按需要编辑“/etc/redis/sentinel.conf”

**sentinel monitor mymaster redis\_node\_1\_ip\_addr 6379 2**

**sentinel down-after-milliseconds mymaster 5000**

**sentinel config-epoch mymaster 0**

**sentinel leader-epoch mymaster 0**

**min-slaves-to-write 1**

**min-slaves-max-lag 10**

**daemonize yes**

**pidfile "/var/run/redis-sentinel.pid"**

**loglevel verbose**

**logfile "/var/log/redis/sentinel.log"**

**sentinel parallel-syncs mymaster 1**

## 配置开机启动服务

## 将Redis添加到开机启动服务

cd /etc/init.d/

添加脚本的执行权限  
sudo chmod a+x /etc/init.d/redis  
设置开机自动启动  
sudo update-rc.d redis defaults

## 将sentinel添加到开机启动服务

cd /etc/init.d

添加脚本的执行权限  
chmod a+x sentinel  
设置开机自动启动  
sudo update-rc.d sentinel defaults

## 将del-redis-pid添加到开机启动服务

cd /etc/init.d/

添加脚本的执行权限  
sudo chmod a+x /etc/init.d/del-redis-pid  
设置开机自动启动  
sudo update-rc.d del-redis-pid defaults

查看当前run level： runlevel

例如返回N 5

cd /etc/rc5.d

sudo mv S02redis S03redis

sudo mv S02sentinel S03sentinel

## 启动 Sentinel服务

service sentinel start

## 检查Sentinel状态

redis-cli –p 26379

在命令行中输入:

sentinel master mymaster

输入应该如下, 并且没有错误和警告.

1) "name"

2) "mymaster"

3) "ip"

4) "127.0.0.1"

5) "port"

6) "6379"

7) "runid"

8) "953ae6a589449c13ddefaee3538d356d287f509b"

9) "flags"

10) "master"

11) "link-pending-commands"

12) "0"

13) "link-refcount"

14) "1"

15) "last-ping-sent"

16) "0"

17) "last-ok-ping-reply"

18) "735"

19) "last-ping-reply"

20) "735"

21) "down-after-milliseconds"

22) "5000"

23) "info-refresh"

24) "126"

25) "role-reported"

26) "master"

27) "role-reported-time"

28) "532439"

29) "config-epoch"

30) "1"

31) "num-slaves"

32) "1"

33) "num-other-sentinels"

34) "2"

35) "quorum"

36) "2"

37) "failover-timeout"

38) "60000"

39) "parallel-syncs"

40) "1"

# 验证Redis安装

reboot

系统重启后, 检查Redis log

cat /var/log/redis/redis.log

输出应该没有错误和警告.

1153:M 20 Nov 13:59:05.733 \* Increased maximum number of open files to 10032 (it was originally set to 1024).

\_.\_

\_.-``\_\_ ''-.\_

\_.-`` `. `\_. ''-.\_ Redis 3.0.5 (00000000/0) 64 bit

.-`` .-```. ```\/ \_.,\_ ''-.\_

( ' , .-` | `, ) Running in standalone mode

|`-.\_`-...-` \_\_...-.``-.\_|'` \_.-'| Port: 6379

| `-.\_ `.\_ / \_.-' | PID: 1153

`-.\_ `-.\_ `-./ \_.-' \_.-'

|`-.\_`-.\_ `-.\_\_.-' \_.-'\_.-'|

| `-.\_`-.\_ \_.-'\_.-' | http://redis.io

`-.\_ `-.\_`-.\_\_.-'\_.-' \_.-'

|`-.\_`-.\_ `-.\_\_.-' \_.-'\_.-'|

| `-.\_`-.\_ \_.-'\_.-' |

`-.\_ `-.\_`-.\_\_.-'\_.-' \_.-'

`-.\_ `-.\_\_.-' \_.-'

`-.\_ \_.-'

`-.\_\_.-'

1153:M 20 Nov 13:59:05.745 # Server started, Redis version 3.0.5

1153:M 20 Nov 13:59:05.745 \* The server is now ready to accept connections on port 6379

Slave节点上输出应为

1189:S 20 Nov 14:09:28.872 \* Connecting to MASTER 192.168.100.211:6379

1189:S 20 Nov 14:09:28.873 \* MASTER <-> SLAVE sync started

# 建立防火墙

## 更新防火墙配置

sudo apt install firewalld

sudo firewall-cmd --permanent --zone=internal --add-port=6379/tcp

sudo firewall-cmd --permanent --zone=internal --add-port=26379/tcp

sudo firewall-cmd --permanent --zone=internal --add-source=**redis\_node\_1\_ip\_addr**

sudo firewall-cmd --permanent --zone=internal --add-source=**redis\_node\_2\_ip\_addr**

sudo firewall-cmd --permanent --zone=internal --add-source=**redis\_node\_3\_ip\_addrk**

sudo firewall-cmd --permanent --zone=internal --add-source=**app\_server\_ip1**

sudo firewall-cmd --permanent --zone=internal --add-source=**app\_server\_ip2**

sudo firewall-cmd --reload

## 检查防火墙状态

firewall-cmd --zone=internal --list-all

输出应该如下:

internal

interfaces:

sources: ***ip\_addresses***

services: dhcpv6-client ipp-client mdns samba-client ssh

ports: **6379/tcp 26379/tcp**

masquerade: no

forward-ports:

icmp-blocks:

rich rules:

## Redis Security

Redis provides a tiny layer of authentication that is optionally turned on editing the **redis.conf** file.

When the authorization layer is enabled, Redis will refuse any query by unauthenticated clients. A client can authenticate itself by sending the **AUTH** command followed by the password.

The password is set by the system administrator in clear text inside the redis.conf file. It should be long enough to preent brute force attacks.

To set the Redis password, edit your redis.conf file, find this line:

# requirepass foobared

Then uncomment it and change foobared to your password. Mkae sure you choose something pretty long, 32 characters or so would probably be good:

requirepass 10d9a99851a411cdae8c3fa09d7290df192441a9

After setting the password, save the file, and restart Redis:

$ sudo service redis-server restart

To check that Redis is working, use the Redis command line. The **redis-cli** command is used to access the Redis command line:

redis-cli

If you already set a password for Redis, you have to auth after connecting:

127.0.0.1:6379> auth your\_redis\_password

# Troubleshooting

/bin/sh^M:bad interpreter: No such file or directory

这个错误发生在你在windows下编写文件上传到[linux服务器](http://www.wodianer.net/)去运行的时候。

错误原因：windows和linux的文件不一样。

解决办法:vi该文件 在命令模式下输入 :set ff=unix 回车

例如 a.sh

vi a.sh

进入输入 :set ff=unix 回车

输入:wq 回车

再次执行就不会有这样的问题了。