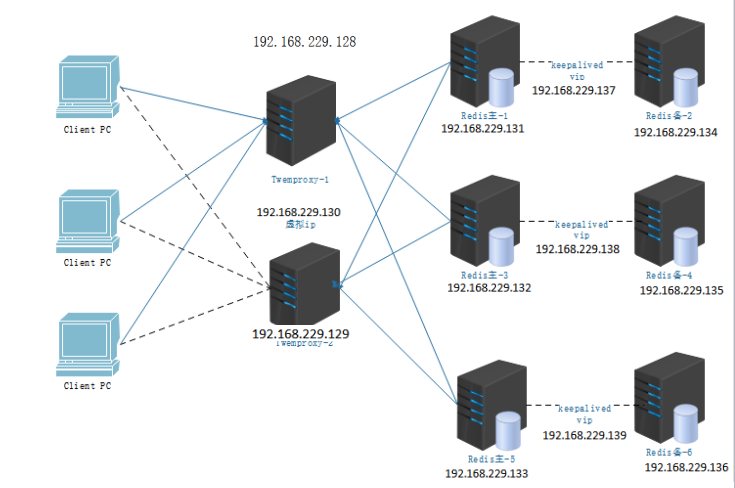
## Redis+TwemProxy（nutcracker）集群

* Twemproxy 又称nutcracker ，是一个memcache、Redis协议的轻量级代理，一个用于sharding 的中间件。有了Twemproxy，客户端不直接访问Redis服务器，而是通过twemproxy 代理中间件间接访问。 Twemproxy 为 Twitter 开源产品，简单来说，Twemproxy是Twitter开发的一个redis代理proxy，类似于nginx的反向代理或者mysql的代理工具，如amoeba。Twemproxy通过引入一个代理层，可以将其后端的多台Redis或Memcached实例进行统一管理与分配，使应用程序只需要在Twemproxy上进行操作，而不用关心后面具体有多少个真实的Redis或Memcached存储。

### 架构图



### 环境说明

**系统环境和软件版本号：**

|  |  |
| --- | --- |
| 项目 | 参数 |
| 系统版本 | Redhat 7.2 64位 |
| 内核版本 | 3.10.0-327.el7.x86\_64 |
| Redis版本 | redis-4.0.0 |
| Keepalived版本 | Keepalived v1.3.5 |
| Twemproxy | nutcracker-0.4.0 |

使用redis+nutcracker+keepalive实现内存高可用解决方案

一 搭建redis环境. 操作在两台机器上完成相同的工作 (memcached)

### 主机环境配置

#### 主机名解析

* 在各主机/etc/hosts中配置主机名解析，两台均执行：

#vim /etc/hosts

127.0.0.1 localhost localhost.localdomain localhost4 localhost4.localdomain4

::1 localhost localhost.localdomain localhost6 localhost6.localdomain6

192.168.229.128 node1

192.168.229.129 node2

#### 安全策略

* 两台主机均selinux/iptables设置:

sed -i 's/SELINUX=.\*/SELINUX=disabled/' /etc/selinux/config

setenforce 0

systemctl stop firewalld

systemctl disable firewalld

### Redis主从搭建

* 可参考3.1 Redis主从配置及keepalived实现Redis高可用

### nutcracker 安装部署

#### 安装包下载

* 将安装包分别拷贝到192.168.229.128 和 192.168.229.129 两台机器中的/usr/local/soft/ 下

#编译安装

cd /usr/local/soft/

tar -xzvf nutcracker-0.4.0.tar.gz

cd nutcracker-0.4.0

./configure

make && make install

* 复制配置文件：

cd /usr/local/soft/nutcracker-0.4.0/conf

cp nutcracker.yml /etc/nutcracker

* 编辑配置文件：

#vim /etc/nutcracker/nutcracker.yml (配置文件只需要写这些就行)

alpha:

listen: 0.0.0.0:22121

hash: fnv1a\_64

hash\_tag: "{}"

distribution: ketama

auto\_eject\_hosts: true

timeout: 400

redis: true

server\_retry\_timeout: 300000

server\_failure\_limit: 3

servers: -----三台redis服务器的地址和端口

#### 测试nutcracker配置文件是否正常

nutcracker –t -c /etc/nutcracker/nutcracker.yml

* **Linux显示内容:**

nutcracker: configuration file 'conf/nutcracker.yml' syntax is ok

#显示以上内容表示负载成功！

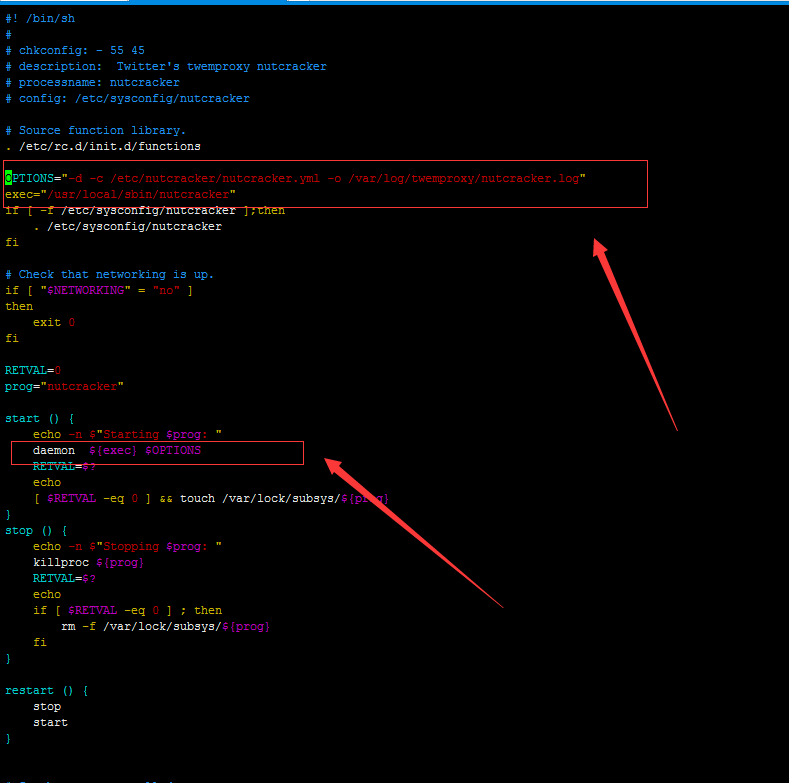
#### 配置chkconfig 以及service

cd /usr/local/soft/nutcracker-0.4.0/scripts

cp nutcracker.init /etc/init.d/nutcracker

chmod 777 /etc/init.d/nutcracker

vim /etc/init.d/nutcracker



修改成上述文件的样子，注意红色框内为改变的地方

第一个红色框：

OPTIONS="-d -c /etc/nutcracker/nutcracker.yml -o /var/log/twemproxy/nutcracker.log"

exec="/usr/local/sbin/nutcracker"

第二个红色框:

daemon ${exec} $OPTIONS

* 设置开机启动项

Chkconfig --add nutcracker

Chkconfig nutcracker on

Chkconfig --list nutcracker

* linux显示内容：

nutcracker 0:off 1:off 2:on  3:on  4:on 5:on 6:off

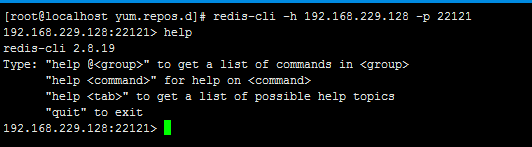
如显示以上内容，则证明nutcracker安装成功

#### 测试nutcracker是否成功为redis代理,

service nutcracker start

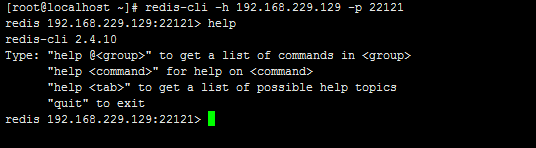
redis-cli  -h 192.168.229.128 -p 22121

#在192.168.229.128上执行的动作



redis-cli  -h 192.168.229.129 -p 22121

#在192.168.229.129上执行的动作



#### keepalived安装

* 以下操作在两台装有nutcracker的机器上执行相同的命令

**Keepalived安装参考安装部署规范**

#### 配置keepalived脚本

编写nutcracker监控脚本

mkdir /etc/keepalived/script

cd /etc/keepalived/script/

vim twemproxy\_check.sh

############################################

#!/bin/bash

#To install check twemproxy status automatically

#Made by liujun,2014/12/31

############################################

#Define system environment PATH

export PATH=$PATH

flag=$(pidof nutcracker)

if [ "$flag" == "" ];then

/etc/init.d/nutcracker start

sleep 3

flag\_wait=$(pidof nutcracker)

if [ "$flag\_wait" == "" ];then

/etc/init.d/keepalived stop

fi

fi

* 编写192.268.229.128上的keepalived配置（配置该台keepalive为Master）

vim /etc/keepalived/keepalived.conf

global\_defs {

router\_id LVS\_01

}

vrrp\_script twemproxy\_check {

script "/etc/keepalived/scripts/twemproxy\_check.sh"

interval 2

weight 2

}

vrrp\_instance VI\_1 {

state MASTER

interface eth0 --可能需要修改

virtual\_router\_id 51

priority 200

advert\_int 1

authentication {

auth\_type PASS

auth\_pass 1111

}

virtual\_ipaddress {

10.72.38.112 //vip 虚拟IP

}

track\_script {

twemproxy\_check

}

}

* 编写192.268.229.129上的keepalived配置（配置该台keepalive为slave）

global\_defs {

router\_id LVS\_02

}

vrrp\_script twemproxy\_check {

script "/etc/keepalived/scripts/twemproxy\_check.sh"

interval 2

weight 2

}

vrrp\_instance VI\_1 {

state BACKUP

interface eth0

virtual\_router\_id 51

priority 150

advert\_int 1

authentication {

auth\_type PASS

auth\_pass 1111

}

virtual\_ipaddress {

192.168.229.190

}

track\_script {

twemproxy\_check

}

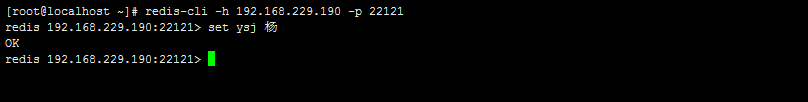
}

#### 启动两台机器上的keepalived

service keepalived start --这条语句分别在两台机器上执行

#### 测试集群环境下的redis是否能正常存储数据

redis-cli  -h 192.168.229.190 -p 22121 --注释192.168.229.190为VIP



**注意：本文中采用的192.168.229.128与192.168.229.129两个IP不做解释，实际安装，可按照不同的机器的IP做相应的更换。本文中的IP只帮助理解实际安装过程**