NSD NOSQL DAY01

1 案例1：搭建Redis服务器

1.1 问题

- 具体要求如下：
- 在主机 192.168.4.51 上安装并启用 redis 服务
- 设置变量test，值为123
- 查看变量test的值

1.2 步骤

实现此案例需要按照如下步骤进行。

步骤一：搭建redis服务器

1）安装redis服务器

```
[root@redis1 ~]# cd redis
[root@redis1 redis]# ls
lnmp_redis-4.0.8.tar.gz
[root@redis1 redis]# yum -y install gcc gcc-c++ make
[root@redis1 redis]# tar -zxf redis-4.0.8.tar.gz
[root@redis1 redis]# cd redis-4.0.8/
[root@redis1 redis-4.0.8]# ls
00-RELEASENOTES  CONTRIBUTING  deps    Makefile  README.md  runtest          runtest-s
BUGS             COPYING       INSTALL MANIFESTO redis.conf runtest-cluster sentinel.conf
[root@redis1 redis-4.0.8]# make
[root@redis1 redis-4.0.8]# make install
[root@redis1 redis-4.0.8]# cd utils/
[root@redis1 utils]# ./install_server.sh
Welcome to the redis service installer
This script will help you easily set up a running redis server

Please select the redis port for this instance: [6379]
Selecting default: 6379
Please select the redis config file name [/etc/redis/6379.conf]
Selected default - /etc/redis/6379.conf
Please select the redis log file name [/var/log/redis_6379.log]
```

Top

```
Selected default -  /var/log/redis_6379.log
Please select the data directory for this instance [ /var/lib/redis/6379]
Selected default -  /var/lib/redis/6379
Please select the redis executable path [ /usr/local/bin/redis- server]
Selected config:
Port           : 6379                 //端口号
Config file    : /etc/redis/6379.conf        //配置文件目录
Log file       : /var/log/redis_6379.log     //日志目录
Data dir       : /var/lib/redis/6379         //数据库目录
Executable     : /usr/local/bin/redis- server  //启动程序的目录
Cli Executable : /usr/local/bin/redis- cli     //命令行的连接工具
Is this ok? Then press ENTER to go on or Ctrl- C to abort.  //回车完成配置
Copied /tmp/6379.conf => /etc/init.d/redis_6379    //服务启动脚本
Installing service...
Successfully added to chkconfig!
Successfully added to runlevels 345!
Starting Redis server...
Installation successful!          //安装成功
```

2）查看状态

```
[ root@redis1 utils] #  /etc/init.d/redis_6379 status
Redis is running ( 15203)
```

3）查看监听的端口

```
[ root@redis1 utils] # netstat - antupl | grep :6379
tcp        0      0 127.0.0.1:6379          0.0.0.0:*               LISTEN      15203/redis- server
[ root@redis1 utils] # ps  - C redis- server
  PID TTY          TIME CMD
15203 ?        00:00:00 redis- server
```

4）停止服务

```
[ root@redis1 utils] # /etc/init.d/redis_6379 stop
Stopping ...
Waiting for Redis to shutdown ...
```

Top

```
Redis stopped
[root@redis1 utils]# /etc/init.d/redis_6379 status
//再次查看，显示 没有那个文件或目录
cat: /var/run/redis_6379.pid: No such file or directory
Redis is running ()
```

5）连接redis

```
[root@redis1 utils]# /etc/init.d/redis_6379 start
Starting Redis server...
[root@redis1 utils]# redis-cli
127.0.0.1:6379> ping
PONG          //PONG说明服务正常
```

6）设置变量test，值为123，查看变量test的值

常用指令操作：

set keyname keyvalue 存储

get keyname 获取

```
127.0.0.1:6379> set test 123
OK
127.0.0.1:6379> get test
"123"
```

del keyname 删除变量

```
127.0.0.1:6379> set k1 v1
OK
127.0.0.1:6379> get k1
"v1"
127.0.0.1:6379> del k1
(integer) 1
```

keys * 打印所有变量

Top

```
127.0.0.1:6379> keys *
```

```
02.     1) "test"
```

EXISTS keyname 测试是否存在

```
01.     127.0.0.1:6379> exists k1
02.     (integer) 0
```

type keyname 查看类型

```
01.     127.0.0.1:6379> set k2 v1
02.     OK
03.     127.0.0.1:6379> type k2
04.     string
```

move keyname dbname 移动变量

```
01.     127.0.0.1:6379> move k2 1          //移动k2到1库
02.     (integer) 1
```

select 数据库编号0-15 切换库

```
01.     127.0.0.1:6379> select 1       //切换到1库
02.     OK
03.     127.0.0.1:6379[1]> keys *          //查看有k2
04.     1) "k2"
```

expire keyname 10 设置有效时间

```
01.     127.0.0.1:6379[1]> EXPIRE k2 10
02.     (integer) 1
```

ttl keyname 查看生存时间

Top

```
127.0.0.1:6379[1]> ttl k2
```

flushall 删除所有变量

```
127.0.0.1:6379[1]> FLUSHALL
OK
```

save 保存所有变量

```
127.0.0.1:6379[1]> save
OK
```

shutdown 关闭redis服务

```
127.0.0.1:6379[1]> SHUTDOWN
```

2 案例2：修改Redis服务运行参数

2.1 问题

- 具体要求如下：
- 端口号 6351
- IP地址 192.168.4.51
- 连接密码 123456
- 客户端连接Redis服务

2.2 步骤

实现此案例需要按照如下步骤进行。

步骤一：修改redis运行参数

1）

```
[root@redis1 utils]# cp /etc/redis/6379.conf  /root/6379.conf
//可以先备份一份，防止修改错误没法还原
[root@redis1 utils]# /etc/init.d/redis_6379 stop
[root@redis1 utils]# vim /etc/redis/6379.conf
...
```

Top

```
bind 192.168.4.51              //设置服务使用的ip
port 6351                      //更改端口号
requirepass 123456             //设置密码
[root@redis1 utils]# /etc/init.d/redis_6379 start
Starting Redis server...
[root@redis1 utils]# ss -antul | grep 6351      //查看有端口6351
tcp    LISTEN     0      128    192.168.4.51:6351                 *:*
```

由于修改了配置文件所以在连接的时候需要加上ip和端口

```
[root@redis1 utils]# redis-cli -h 192.168.4.51 -p 6351
192.168.4.51:6351> ping
(error) NOAUTH Authentication required.
192.168.4.51:6351> auth 123456          //输入密码才能操作（因为之前设置过密码）
OK
192.168.4.51:6351> ping
PONG
```

还可以直接在命令行输入密码连接

```
[root@redis1 utils]# redis-cli -h 192.168.4.51 -p 6351 -a 123456
192.168.4.51:6351> ping
PONG
```

2）停止服务

由于修改Redis服务运行参数，所以在停止服务的时候也不能用默认的方法停止

```
[root@redis1 utils]# /etc/init.d/redis_6379 stop       //停止失败
Stopping ...
Could not connect to Redis at 127.0.0.1:6379: Connection refused
Waiting for Redis to shutdown ...
Waiting for Redis to shutdown ...
Waiting for Redis to shutdown ...
Waiting for Redis to shutdown ...
...
```

Top

```
[root@redis1 utils]# redis-cli -h 192.168.4.51 -p 6351 -a 123456 shutdown
//停止成功
[root@redis1 utils]# ss -antul | grep 6351      //查看没有端口
```

3 案例3：部署LNMP+Redis

3.1 问题

- 具体要求如下：
- 在主机 192.168.4.52 上部署LNMP 环境
- 把数据存储到本机的redis服务中

3.2 步骤

实现此案例需要按照如下步骤进行。

步骤一：部署LNMP+Redis

1）安装redis，（不会搭建的请参考案例1）

2）安装php支持的功能模块（52上面操作）

```
[root@nginx utils]# which php
/usr/bin/which: no php in (/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/root/bin)
[root@nginx utils]# php -m
bash: php: command not found...
[root@nginx utils]# yum -y install php-cli
[root@nginx utils]# which php
/usr/bin/php
[root@nginx utils]# php -m
[PHP Modules]
bz2
calendar
Core
ctype
curl
date
ereg
exif
fileinfo
filter
ftp
gettext
gmp
hash
```

Top

```
24.  iconv
25.  json
26.  libxml
27.  mhash
28.  openssl
29.  pcntl
30.  pcre
31.  Phar
32.  readline
33.  Reflection
34.  session
35.  shmop
36.  SimpleXML
37.  sockets
38.  SPL
39.  standard
40.  tokenizer
41.  xml
42.  zip
43.  zlib
44.
45.  [Zend Modules]
```

3）安装连接redis的功能模块

```
01.  [root@nginx utils]# php -m | grep -i redis      //没有redis模块
02.  [root@nginx redis]# cd lnmp/
03.  [root@nginx lnmp]# ls
04.  nginx-1.12.2.tar.gz
05.  php-devel-5.4.16-42.el7.x86_64.rpm
06.  php-fpm-5.4.16-42.el7.x86_64.rpm
07.  php-redis-2.2.4.tar.gz
08.  [root@nginx lnmp]# tar -zxf php-redis-2.2.4.tar.gz
09.  [root@nginx lnmp]# cd phpredis-2.2.4/
10.  [root@nginx phpredis-2.2.4]# which phpize
11.  /usr/bin/phpize
12.  [root@nginx phpredis-2.2.4]# phpize
13.  Can't find PHP headers in /usr/include/php
14.  The php-devel package is required for use of this command.
15.  [root@nginx phpredis-2.2.4]# yum -y install autoconf automake  pcre-devel
```

Top

```
[root@nginx phpredis-2.2.4]# cd ..
[root@nginx lnmp]# rpm -ivh php-devel-5.4.16-42.el7.x86_64.rpm
[root@nginx lnmp]# cd phpredis-2.2.4/
[root@nginx phpredis-2.2.4]# phpize     //生成一个php的文件
Configuring for:
PHP Api Version:         20100412
Zend Module Api No:      20100525
Zend Extension Api No:   220100525
[root@nginx phpredis-2.2.4]# find / -name "php-config"
/usr/bin/php-config
[root@nginx phpredis-2.2.4]#  ./configure  --with-php-config=/usr/bin/php-config
//指定模块编译的路径
[root@nginx phpredis-2.2.4]#  make && make install
...
Installing shared extensions:     /usr/lib64/php/modules/  //模块文件存放的路径
[root@nginx phpredis-2.2.4]#  ls /usr/lib64/php/modules/
curl.so  fileinfo.so  json.so  phar.so  redis.so  zip.so
[root@nginx phpredis-2.2.4]# vim /etc/php.ini
728   extension_dir = "/usr/lib64/php/modules/"
729 ; On windows:
730   extension = "redis.so"
[root@nginx phpredis-2.2.4]# php -m | grep -i redis
redis       //出现redis
```

4）安装nginx（52上面操作）

```
[root@nginx ~]# cd redis/lnmp/
[root@nginx lnmp]# ls
nginx-1.12.2.tar.gz
[root@nginx lnmp]# tar -xf nginx-1.12.2.tar.gz
[root@nginx lnmp]# cd nginx-1.12.2/
[root@nginx nginx-1.12.2]# yum -y  install  gcc pcre-devel openssl-devel
[root@nginx nginx-1.12.2]# useradd -s /sbin/nologin  nginx
[root@nginx nginx-1.12.2]# ./configure --user=nginx --group=nginx --with-http_ssl_modu
[root@nginx nginx-1.12.2]# make && make install
[root@nginx nginx-1.12.2]# ln -s /usr/local/nginx/sbin/nginx  /sbin/
[root@nginx nginx-1.12.2]#  cd /usr/local/nginx/html/
[root@nginx html]# echo "aa" > text.html
[root@nginx html]# yum -y  install mariadb mariadb-server mariadb-devel php php-mysql
[root@nginx html]# cd /root/redis/lnmp/
```

Top

```
[root@nginx lnmp]#  rpm -ivh php-fpm-5.4.16-42.el7.x86_64.rpm       //安装php
[root@nginx lnmp]# cd /usr/local/nginx/html/
[root@nginx html]# vim test.php
<?php
$i=33;
$j=44;
if ($i<$j) {
echo "oK";
}
else{
echo "error";
}
#echo $i;
?>
[root@nginx html]# php test.php        //在命令行测试
oK
[root@nginx html]# systemctl restart  mariadb
[root@nginx html]# systemctl restart  php-fpm
[root@nginx html]# vim /usr/local/nginx/conf/nginx.conf
...
        location ~ \.php$ {
            root           html;
            fastcgi_pass   127.0.0.1:9000;
            fastcgi_index  index.php;
            #fastcgi_param  SCRIPT_FILENAME  /scripts$fastcgi_script_name;
            include        fastcgi.conf;
        }
...
[root@nginx html]# nginx    //启动nginx
客户端用火狐浏览器访问：
[root@room9pc01 ~]# firefox  192.168.4.56/text.html       //成功
[root@room9pc01 ~]# firefox  192.168.4.56/test.php       //成功
```

5）连接redis测试

```
[root@nginx html]# vim lkredis.php
<?php
$redis = new redis();
$redis->connect('192.168.4.51',6351);
```

Top

```
$redis->auth("123456");
$redis->set("redistest","666666");
echo $redis->get("redistest");
?>
[root@nginx html]# php lkredis.php        //命令行测试
666666
```

火狐浏览器测试，如图-1所示：

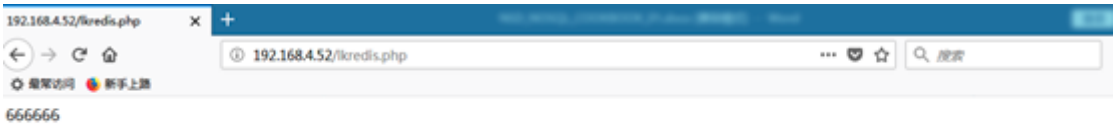

图-1

在51上面查看，有数据存入

```
[root@redis1 lnmp]# redis-cli -h 192.168.4.51 -p 6351 -a 123456
192.168.4.51:6351> ping
PONG
192.168.4.51:6351> keys *
1) "redistest"
192.168.4.51:6351> get redistest
"666666"
192.168.4.51:6351>
```

Top