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

```
01.    [root@redis1 ~]# cd redis
02.    [root@redis1 redis]# ls
03.    lnmp  redis-4.0.8.tar.gz
04.    [root@redis1 redis]# yum -y install gcc gcc-c++ make
05.    [root@redis1 redis]# tar -zxf redis-4.0.8.tar.gz
06.    [root@redis1 redis]# cd redis-4.0.8/
07.    [root@redis1 redis-4.0.8]# ls
08.    00-RELEASENOTES  CONTRIBUTING deps   Makefile  README.md  runtest        runtest-
09.    BUGS        COPYING     INSTALL  MANIFESTO redis.conf  runtest-cluster  sentinel.conf
10.    [root@redis1 redis-4.0.8]# make
11.    [root@redis1 redis-4.0.8]# make install
12.    [root@redis1 redis-4.0.8]# cd utils/
13.    [root@redis1 utils]# ./install_server.sh
14.    Welcome to the redis service installer
15.    This script will help you easily set up a running redis server
16.
17.    Please select the redis port for this instance: [6379]
18.    Selecting default: 6379
19.    Please select the redis config file name [/etc/redis/6379.conf]
20.    Selected default - /etc/redis/6379.conf
21.    Please select the redis log file name [/var/log/redis_6379.log]
```

Top

22.   Selected default - /var/log/redis_6379.log

23.   Please select the data directory for this instance [/var/lib/redis/6379]

24.   Selected default - /var/lib/redis/6379

25.   Please select the redis executable path [/usr/local/bin/redis-server]

26.   Selected config:

27.   Port   : 6379   //端口号

28.   Config file   : /etc/redis/6379.conf   //配置文件目录

29.   Log file   : /var/log/redis_6379.log   //日志目录

30.   Data dir   : /var/lib/redis/6379   //数据库目录

31.   Executable   : /usr/local/bin/redis-server //启动程序的目录

32.   Cli Executable : /usr/local/bin/redis-cli   //命令行的连接工具

33.   Is this ok? Then press ENTER to go on or Ctrl-C to abort. //回车完成配置

34.   Copied /tmp/6379.conf => /etc/init.d/redis_6379   //服务启动脚本

35.   Installing service...

36.   Successfully added to chkconfig!

37.   Successfully added to runlevels 345!

38.   Starting Redis server...

39.   Installation successful!   //安装成功

2）查看状态

01.   [root@redis1 utils] # /etc/init.d/redis_6379 status

02.   Redis is running (15203)

3）查看监听的端口

01.   [root@redis1 utils] # netstat -antupl | grep : 6379

02.   tcp   0   0 127.0.0.1: 6379   0.0.0.0: *   LISTEN   15203/redis-server

03.   [root@redis1 utils] # ps -C redis-server

04.   PID TTY   TIME CMD

05.   15203 ?   00: 00: 00 redis-server

4）停止服务

01.   [root@redis1 utils] # /etc/init.d/redis_6379 stop   **Top**

02.   Stopping ...

03.   Waiting for Redis to shutdown ...

04.   Redis stopped

05.   [root@redis1 utils] # /etc/init.d/redis_6379 status

06.   *//再次查看，显示 没有那个文件或目录*

07.   cat: /var/run/redis_6379.pid: No such file or directory

08.   Redis is running ( )

5）连接redis

01.   [root@redis1 utils] # /etc/init.d/redis_6379 start

02.   Starting Redis server...

03.   [root@redis1 utils] # redis-cli

04.   127.0.0.1: 6379> ping

05.   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

01.   127.0.0.1: 6379> set test 123

02.   OK

03.   127.0.0.1: 6379> get test

04.   "123"

del keyname 删除变量

01.   127.0.0.1: 6379> set k1 v1

02.   OK

03.   127.0.0.1: 6379> get k1

04.   "v1"

05.   127.0.0.1: 6379> del k1

06.   (integer) 1

keys * 打印所有变量

01.   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

```
01.     127.0.0.1:6379[1] > ttl k2
```

flushall 删除所有变量

```
01.     127.0.0.1:6379[1] > FLUSHALL
02.     OK
```

save 保存所有变量

```
01.     127.0.0.1:6379[1] > save
02.     OK
```

shutdown 关闭redis服务

```
01.     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）

```
01.     [root@redis1 utils]# cp /etc/redis/6379.conf  /root/6379.conf
02.     //可以先备份一份，防止修改错误没法还原
03.     [root@redis1 utils]# /etc/init.d/redis_6379 stop
04.     [root@redis1 utils]# vim /etc/redis/6379.conf
05.     ...
```

06.   bind 192.168.4.51   //设置服务使用的ip
07.   port 6351   //更改端口号
08.   requirepass 123456   //设置密码
09.   [root@redis1 utils]# /etc/init.d/redis_6379 start
10.   Starting Redis server...
11.   [root@redis1 utils]# ss -antul | grep 6351   //查看有端口6351
12.   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

01.   [root@redis1 utils]# redis-cli -h 192.168.4.51 -p 6351
02.   192.168.4.51:6351> ping
03.   (error) NOAUTH Authentication required.
04.   192.168.4.51:6351> auth 123456   //输入密码才能操作（因为之前设置过密码）
05.   OK
06.   192.168.4.51:6351> ping
07.   PONG

还可以直接在命令行输入密码连接

01.   [root@redis1 utils]# redis-cli -h 192.168.4.51 -p 6351 -a 123456
02.   192.168.4.51:6351> ping
03.   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

01.   [root@redis1 utils]# /etc/init.d/redis_6379 stop   //停止失败
02.   Stopping ...
03.   Could not connect to Redis at 127.0.0.1:6379: Connection refused
04.   Waiting for Redis to shutdown ...
05.   Waiting for Redis to shutdown ...
06.   Waiting for Redis to shutdown ...
07.   Waiting for Redis to shutdown ...
08.   ...

Top

09.
10.

11.　　[root@redis1 utils] # redis- cli - h 192.168.4.51 - p 6351   - a 123456 shutdown

12.　　//停止成功

13.　　[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）

01.　　[root@nginx utils] # which php

02.　　/usr/bin/which: no php in (/usr/local/sbin: /usr/local/bin: /usr/sbin: /usr/bin: /root/bin)

03.　　[root@nginx utils] # php - m

04.　　bash: php: command not found...

05.　　[root@nginx utils] # yum - y install php- cli

06.　　[root@nginx utils] # which php

07.　　/usr/bin/php

08.　　[root@nginx utils] # php - m

09.　　[PHP Modules]

10.　　bz2

11.　　calendar

12.　　Core

13.　　ctype

14.　　curl

15.　　date

16.　　ereg

17.　　exif

18.　　fileinfo

19.　　filter

20.　　ftp

21.　　gettext   **Top**

22.　　gmp

23.　　hash

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   **Top**

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

16. [root@nginx phpredis-2.2.4]# cd ..

17. [root@nginx lnmp]# rpm -ivh php-devel-5.4.16-42.el7.x86_64.rpm

18. [root@nginx lnmp]# cd phpredis-2.2.4/

19. [root@nginx phpredis-2.2.4]# phpize   //生成一个php的文件

20. Configuring for:

21. PHP Api Version: 20100412

22. Zend Module Api No: 20100525

23. Zend Extension Api No: 220100525

24. [root@nginx phpredis-2.2.4]# find / -name "php-config"

25. /usr/bin/php-config

26. [root@nginx phpredis-2.2.4]# ./configure --with-php-config=/usr/bin/php-config

27. //指定模块编译的路径

28. [root@nginx phpredis-2.2.4]# make && make install

29. …

30. Installing shared extensions: /usr/lib64/php/modules/ //模块文件存放的路径

31. [root@nginx phpredis-2.2.4]# ls /usr/lib64/php/modules/

32. curl.so fileinfo.so json.so phar.so redis.so zip.so

33. [root@nginx phpredis-2.2.4]# vim /etc/php.ini

34. 728 extension_dir = "/usr/lib64/php/modules/"

35. 729 ; On windows:

36. 730 extension = "redis.so"

37. [root@nginx phpredis-2.2.4]# php -m | grep -i redis

38. redis   //出现redis

4）安装nginx（52上面操作）

01. [root@nginx ~]# cd redis/lnmp/

02. [root@nginx lnmp]# ls

03. nginx-1.12.2.tar.gz

04. [root@nginx lnmp]# tar -xf nginx-1.12.2.tar.gz

05. [root@nginx lnmp]# cd nginx-1.12.2/

06. [root@nginx nginx-1.12.2]# yum -y install gcc pcre-devel openssl-devel

07. [root@nginx nginx-1.12.2]# useradd -s /sbin/nologin nginx

08. [root@nginx nginx-1.12.2]# ./configure --user=nginx --group=nginx --with-http_ssl_modu

09. [root@nginx nginx-1.12.2]# make && make install

10. [root@nginx nginx-1.12.2]# ln -s /usr/local/nginx/sbin/nginx /sbin/

11. [root@nginx nginx-1.12.2]# cd /usr/local/nginx/html/

12. [root@nginx html]# echo "aa" > text.html

Top

13. [root@nginx html]# yum -y install mariadb mariadb-server mariadb-devel php php-mysql

14. [root@nginx html]# cd /root/redis/lnmp/

15.　　[root@nginx lnmp] # rpm - iv h php- f pm- 5. 4. 16- 42. el7. x86_64. rpm　　　//安装php

16.　　[root@nginx lnmp] # cd /usr/local/nginx/html/

17.　　[root@nginx html] # v im test. php

18.　　<?php

19.　　$i=33;

20.　　$j=44;

21.　　if ($i<$j) {

22.　　echo "oK";

23.　　}

24.　　else{

25.　　echo "error";

26.　　}

27.　　#echo $i;

28.　　?>

29.　　[root@nginx html] # php test. php　　　//在命令行测试

30.　　oK

31.　　[root@nginx html] # sy stemctl restart mariadb

32.　　[root@nginx html] # sy stemctl restart php- f pm

33.　　[root@nginx html] # v im /usr/local/nginx/conf /nginx. conf

34.　　…

35.　　　　location ~ \. php$ {

36.　　　　　root　　　html;

37.　　　　　f astcgi_pass　127. 0. 0. 1: 9000;

38.　　　　　f astcgi_index index. php;

39.　　　　　#f astcgi_param SCRIPT_FILENAME /scripts$fastcgi_script_name;

40.　　　　　include　　f astcgi. conf ;

41.　　　　}

42.　　…

43.　　[root@nginx html] # nginx　　//启动nginx

44.　　客户端用火狐浏览器访问：

45.　　[root@room9pc01 ~] # f iref ox 192. 168. 4. 56/text. html　　　//成功

46.　　[root@room9pc01 ~] # f iref ox 192. 168. 4. 56/test. php　　　//成功

5）连接redis测试

01.　　[root@nginx html] # v im lkredis. php

02.　　<?php

03.　　$redis = new redis( ) ;

04.　　$redis- >connect('192. 168. 4. 51', 6351) ;

Top

05. $redis->auth("123456");

06. $redis->set("redistest","666666");

07.   echo $redis->get("redistest");

08.   ?>

09.   [root@nginx html]# php lkredis.php   //命令行测试

10.   666666

火狐浏览器测试，如图-1所示：
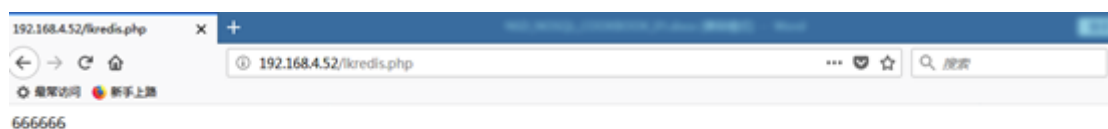

图-1

在51上面查看，有数据存入

01.   [root@redis1 lnmp]# redis-cli -h 192.168.4.51 -p 6351 -a 123456

02.   192.168.4.51:6351> ping

03.   PONG

04.   192.168.4.51:6351> keys *

05.   1) "redistest"

06.   192.168.4.51:6351> get redistest

07.   "666666"

08.   192.168.4.51:6351>

Top