

TTS 11.0 COOKBOOK

(NSD NOSQL DAY01)

版本编号 11.0

2019-06

达内 IT 培训集团

NSD NOSQL DAY01

1. 案例 1：搭建 Redis 服务器

• 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 school, 值为 tarena
- 查看变量 school 的值

• 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) 安装源码 redis 软件

```
[root@redis1 redis]# yum -y install gcc
[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-sentinel src          utils
BUGS              COPYING      INSTALL      MANIFESTO    redis.conf    runtest-cluster
sentinel.conf     tests
[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]
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... //提示服务已经启动
Installation successful! //提示安装成功
```

2) 查看服务状态

```
[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 //查看端口
tcp        0      0 127.0.0.1:6379      0.0.0.0:*           LISTEN
15203/redis-server
[root@redis1 utils]# ps -C redis-server //查看进程
  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 ...
Redis stopped
```

5) 连接 redis

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

6) 存储变量 school, 值为 tarena, 查看变量 school 的值

常用指令操作:

```
set keyname keyvalue //存储
get keyname           //获取
```

```
127.0.0.1:6379> set school tarena
OK
127.0.0.1:6379> get school
"tarena"
127.0.0.1:6379>
```

2. 案例 2: 常用命令

• 问题

- 练习如下命令的使用:
- set mset get mget keys type
- exists ttl expire move 、select
- del flushdb flushall save shutdown

- **步骤**

实现此案例需要按照如下步骤进行。

步骤一：命令 set 、 mset 、 get 、 mget

具体操作如下

```
192.168.4.50:6350> set name bob
OK
192.168.4.50:6350>
192.168.4.50:6350> mset age 19 sex boy
OK
192.168.4.50:6350>
192.168.4.50:6350> get name
"bob"
192.168.4.50:6350>
192.168.4.50:6350> mget age sex
1) "19"
2) "boy"
192.168.4.50:6350>
192.168.4.50:6350>
```

步骤二：命令 keys 、 type 、 exists 、 del

具体操作如下

```
192.168.4.50:6350> keys *
1) "sex"
2) "age"
3) "name"
192.168.4.50:6350>
192.168.4.50:6350> keys ???
1) "sex"
2) "age"
192.168.4.50:6350> keys a*
1) "age"
192.168.4.50:6350>
192.168.4.50:6350> type age //使用 set 命令存储的变量都是字符类型
string
192.168.4.50:6350>
192.168.4.50:6350> del age
(integer) 1
192.168.4.50:6350>
192.168.4.50:6350> exists age //变量不存储返回值 0
(integer) 0
192.168.4.50:6350>
192.168.4.50:6350> exists sex //变量存在 返回值 1
(integer) 1
192.168.4.50:6350>
```

步骤三：命令 ttl 、 expire 、 move 、 flushdb 、 flushall 、 save、 shutdown 、 select

具体操作如下

```
192.168.4.50:6350> keys *
1) "sex"
2) "name"
192.168.4.50:6350> ttl sex //返回值-1 表示变量永不过期
(integer) -1
192.168.4.50:6350>
192.168.4.50:6350> expire sex 20 //设置变量过期时间为 20 秒
(integer) 1
192.168.4.50:6350>
192.168.4.50:6350> ttl sex //还剩 14 秒过期
(integer) 14
192.168.4.50:6350>
192.168.4.50:6350> ttl sex //返回值-2 表示已经过期
(integer) -2
192.168.4.50:6350> exists sex //变量已经不存在
(integer) 0
192.168.4.50:6350>
192.168.4.50:6350> move name 1 //把变量 name 移动到 1 号库里
(integer) 1
192.168.4.50:6350>
192.168.4.50:6350> select 1 //切换到 1 号库
OK
192.168.4.50:6350[1]> keys * //查看
1) "name"
192.168.4.50:6350[1]> select 0 //切换到 0 号库
OK
192.168.4.50:6350> keys * //查看
(empty list or set)
192.168.4.50:6350>
192.168.4.50:6350> select 1 //切换到 1 号库
OK
192.168.4.50:6350[1]>
192.168.4.50:6350[1]> keys *
1) "name"
192.168.4.50:6350[1]>
192.168.4.50:6350[1]> flushdb
OK
192.168.4.50:6350[1]>
192.168.4.50:6350[1]> keys *
(empty list or set)
192.168.4.50:6350[1]>
192.168.4.50:6350[1]> flushall
OK
192.168.4.50:6350[1]>
192.168.4.50:6350[1]> save
OK
192.168.4.50:6350[1]>
192.168.4.50:6350[1]> shutdown
not connected> //提示连接断开
not connected> exit //退出登录
[root@host50 ~]#
[root@host50 ~]# netstat -utnlp | grep redis-server //没有进程信息
[root@host50 ~]#
```

```
[root@host50 ~]# /etc/init.d/redis_6379 start //启动服务
Starting Redis server...
[root@host50 ~]#
[root@host50 ~]# netstat -utnlp | grep redis-server //查看进程信息
tcp        0      0 192.168.4.50:6350    0.0.0.0:*           LISTEN
11475/redis-server
[root@host50 ~]#
```

3. 案例 3：修改 Redis 服务运行参数

• 问题

- 对 Redis 服务器 192.168.4.50 做如下配置：
- 端口号 6350
- IP 地址 192.168.4.50
- 连接密码 123456
- 测试配置

• 步骤

实现此案例需要按照如下步骤进行。

步骤一：修改主配置文件

1) 修改配置文件

```
[root@host50 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@host50 utils]# /etc/init.d/redis_6379 stop

[root@host50 utils]# vim /etc/redis/6379.conf
...
bind 192.168.4.50           //设置服务使用的 ip
port 6350                   //更改端口号
requirepass 123456          //设置密码
:wq
```

2) 修改启动脚本

```
[root@host50 ~]# vim +43 /etc/init.d/redis_6379
$CLIXEC -h 192.168.4.50 -p 6350 -a 123456 shutdown
:wq
```

3) 启动服务

```
[root@host50 ~]# /etc/init.d/redis_6379 start
Starting Redis server...
[root@host50 ~]#
[root@host50 ~]# netstat -utnlp | grep redis-server
tcp        0      0 192.168.4.50:6350    0.0.0.0:*           LISTEN
11523/redis-server
```

4) 测试配置

访问服务存取数据

```
[root@host50 ~]# redis-cli -h 192.168.4.50 -p 6350 -a 123456 //访问服务
192.168.4.50:6350> ping
PONG
192.168.4.50:6350> keys *
(empty list or set)
192.168.4.50:6350>
192.168.4.50:6350> set x 99
OK
192.168.4.50:6350>
192.168.4.50:6350> exit
[root@host50 ~]#
```

4. 案例 4: 部署 LNMP+Redis

• 问题

- 具体要求如下:
- 在主机 192.168.4.57 部署 LNMP 环境
- 配置 PHP 支持 redis
- 编写网站脚本, 把数据存储到 redis 服务器 192.168.4.50

• 步骤

实现此案例需要按照如下步骤进行。

步骤一: 在主机 192.168.4.57 部署 LNMP 环境

1) 安装源码 nginx 软件及 php-fpm

```
]#yum -y install gcc pcre-devel zlib-devel //安装依赖
]#tar -zxvf nginx-1.12.2.tar.gz //解压
]#cd nginx-1.12.2 //进源码目录
]#./configure //配置
.....
Configuration summary
+ using system PCRE library
+ OpenSSL library is not used
+ using system zlib library

nginx path prefix: "/usr/local/nginx"
nginx binary file: "/usr/local/nginx/sbin/nginx"
nginx modules path: "/usr/local/nginx/modules"
nginx configuration prefix: "/usr/local/nginx/conf"
nginx configuration file: "/usr/local/nginx/conf/nginx.conf"
nginx pid file: "/usr/local/nginx/logs/nginx.pid"
```

```

nginx error log file: "/usr/local/nginx/logs/error.log"
nginx http access log file: "/usr/local/nginx/logs/access.log"
nginx http client request body temporary files: "client_body_temp"
nginx http proxy temporary files: "proxy_temp"
nginx http fastcgi temporary files: "fastcgi_temp"
nginx http uwsgi temporary files: "uwsgi_temp"
nginx http scgi temporary files: "scgi_temp"

[root@localhost nginx-1.12.2]# make //编译
.....
objs/src/http/modules/ngx_http_upstream_zone_module.o \
objs/ngx_modules.o \
-lldl -lpthread -lcrypt -lpcrc -lz \
-Wl,-E
sed -e "s|%%PREFIX%|/usr/local/nginx|" \
    -e "s|%%PID_PATH%|/usr/local/nginx/logs/nginx.pid|" \
    -e "s|%%CONF_PATH%|/usr/local/nginx/conf/nginx.conf|" \
    -e "s|%%ERROR_LOG_PATH%|/usr/local/nginx/logs/error.log|" \
    < man/nginx.8 > objs/nginx.8
make[1]: 离开目录 "/root/lnmp/nginx-1.12.2"

[root@localhost nginx-1.12.2]#make install //安装
.....
test -d '/usr/local/nginx/logs' \
    || mkdir -p '/usr/local/nginx/logs'
test -d '/usr/local/nginx/html' \
    || cp -R html '/usr/local/nginx'
test -d '/usr/local/nginx/logs' \
    || mkdir -p '/usr/local/nginx/logs'
make[1]: 离开目录 "/root/lnmp/nginx-1.12.2"

[root@localhost nginx-1.12.2]# ls /usr/local //查看安装目录
bin etc games include lib lib64 libexec nginx sbin share src
[root@localhost nginx-1.12.2]#
[root@localhost nginx-1.12.2]# ls /usr/local/nginx //查看目录列表
conf html logs sbin
[root@localhost nginx-1.12.2]#

]#yum -y install php-fpm //安装 php-fpm
.....
已安装:
    php-fpm.x86_64 0:5.4.16-45.el7

作为依赖被安装:
    libzip.x86_64 0:0.10.1-8.el7          php-common.x86_64 0:5.4.16-45.el7

完毕!

```

2) 修改配置 nginx.conf

```

] # vim +65 /usr/local/nginx/conf/nginx.conf
    location ~ /\.php$ {
        root            html;
        fastcgi_pass    127.0.0.1:9000;
        fastcgi_index   index.php;
        include         fastcgi.conf;
    }

```



```
:wq
```

```
]# /usr/local/nginx/sbin/nginx -t //测试修改
nginx: the configuration file /usr/local/nginx/conf/nginx.conf syntax is ok
nginx: configuration file /usr/local/nginx/conf/nginx.conf test is successful
```

3) 启动服务

启动 php-fpm 服务

```
]# systemctl start php-fpm //启动服务
]# netstat -utnlp | grep :9000 //查看端口
```

启动 nginx 服务

```
]# /usr/local/nginx/sbin/nginx
]# netstat -utnlp | grep :80
tcp        0      0 0.0.0.0:80      0.0.0.0:*        LISTEN
23505/nginx: master
```

4) 测试配置

```
]# vim /usr/local/nginx/html/test.php //编写 php 文件
<?php
    echo "hello world!!!";
?>
:wq

]# curl http://localhost/test.php //访问 nginx 服务
hello world!!!
```

步骤二：配置 PHP 支持 redis

1) 安装 php 扩展

```
[root@host71 ~]# rpm -q php php-devel
未安装软件包 php
未安装软件包 php-devel
[root@host71 ~]#
[root@host71 ~]# rpm -q automake autoconf
未安装软件包 automack
未安装软件包 autoconf
[root@host71 ~]#
[root@host71 ~]# yum -y install php php-devel automake autoconf //安装依赖

]# tar -zxvf php-redis-2.2.4.tar.gz //安装扩展包
]# cd phpredis-2.2.4/
]# phpize //生成配置文件 php-config 及 configure 命令
Configuring for:
PHP Api Version:      20100412
Zend Module Api No:   20100525
Zend Extension Api No: 220100525

]# ./configure --with-php-config=/usr/bin/php-config //配置
```

```
]# make //编译  
]# make install //安装
```

2) 修改 php.ini 文件

```
]#vim /etc/php.ini  
728 extension_dir = "/usr/lib64/php/modules/" //模块文件目录  
730 extension = "redis.so" //模块文件名  
:wq  
  
]# systemctl restart php-fpm //重启 php-fpm 服务  
]# php -m | grep -i redis //查看已加载的模块  
redis
```

步骤三：测试配置：编写网站脚本，把数据存储到 redis 服务器 192.168.4.50

1) 查看 192.168.4.50 主机的 redis 服务是否运行

```
[root@host50 ~]# netstat -utnlp | grep redis-server  
tcp        0      0 192.168.4.50:6350 0.0.0.0:*        LISTEN  
11523/redis-server  
[root@host50 ~]#  
  
[root@host50 ~]# redis-cli -h 192.168.4.50 -p 6350 -a 123456 //访问服务  
192.168.4.50:6350> ping  
PONG  
192.168.4.50:6350> exit
```

2) 编写网站脚本

```
]# vim /usr/local/nginx/html/linkredis.php  
<?php  
$redis = new redis();  
$redis->connect("192.168.4.50","6350");  
$redis->auth("123456");  
$redis->set("linux","redhat");  
echo $redis->get("linux");  
?>  
:wq
```

3) 访问网站脚本

```
]#curl http://localhost/linkredis.php //访问 nginx 服务  
redhat
```

4) 在 192.168.4.50 服务器，查看数据

```
[root@host50 ~]# redis-cli -h 192.168.4.50 -p 6350 -a 123456 //连接 redis 服务  
192.168.4.50:6350> keys * //查看变量  
1) "linux"  
192.168.4.50:6350>  
192.168.4.50:6350> get linux //获取值  
"redhat"
```

达内IT培训