

11_nosql01nosql 概述+部署 redis 服务+部署 lnmredis

一 nosql 概述

1.1 数据库类型

1.1.1 RDBMS: Relational DataBase Management System

关系型数据库

安装预先设置的组织结构,将数据存储于物理介质上

数据之间可以做关联操作

1.1.2 RDBMS 软件

MySQL MariaDB Oracle DB2 SQL Server 前 3 款为主流,前 2 款互斥

1.1.3 NoSQL: Not Only SQL(不仅仅是 SQL)

泛指非关系型数据库

不需要预先定义数据存储结构

每条记录可以有不同的数据类型和字段个数

1.1.4 nosql 软件

Redis Memcached MongoDB CouchDB Ne04j FlockDB 前 3 款为主流

二 部署 Redis 服务,搭建 Redis 服务器

2.1 Redis 介绍

Remote Dictionary Server,远程字典服务器

是一款高性能的(Key/VAalues)分布式内存数据库

支持数据持久化(定期把内存里数据存储在硬盘)

支持多种数据类型: string\list\hash ...

支持 master-slave 模式数据备份,支持集群

中文网站 www.redis.cn

2.2 安装软件

```
room9pc01 ~]$ scp -r /linux-soft/03/redis/
```

```
root@192.168.4.51:/root
```

```
mysql50 ~]# rpm -q gcc || yum -y install gcc
```

```
mysql50 ~]# cd redis
```

```
mysql50 redis]# tar -xf redis-4.0.8.tar.gz
```

```
mysql50 redis]# cd redis-4.0.8/
```

```
mysql50 redis-4.0.8]# make && make install
```

2.3 初始配置

执行源码目录下 `util/install_server.sh` 的 shell 脚本

```
mysql50 redis-4.0.8]# ./utils/install_server.sh #执行
```

一路回车,ctrl+c 取消,以下为默认设置

端口 **6379**

主配置文件 **/etc/redis/6379.conf**

日志文件 **/var/log/redis_6379.log**

数据库目录 **/var/lib/redis/6379** 目录下文件为.rdb

服务程序 /usr/local/bin/redis-server

命令行连接命令 /usr/local/bin/redis-cli

2.4 管理服务

停止服务: /etc/init.d/redis_6379 stop

启动服务: /etc/init.d/redis_6379 start

查看进程: ps -C redis-server

查看端口: netstat -antulp | grep :6379

2.5 连接访问服务

命令行输入命令: redis-cli [默认连接本机的 redis 服务]

2.6 常用命令

```
mysql50 ~]# redis-cli                      #登录 redis
127.0.0.1:6379> ping                        #连接测试: ping
PONG
127.0.0.1:6379> set name bob                #变量赋值: set 变量 值
OK                                            #存数据
127.0.0.1:6379> get name                    #查看变量值: get 变量
"bob"                                        #取数据
127.0.0.1:6379> exit                        #断开连接
```

运行中定期保存数据到硬盘,开启或关闭 redis 时保存数据到硬盘

常用数据管理命令

存储 1 个 key 值: **set key 名 key 值**

存储多个 key 值: **mset key 名 1 key 值 1 key 名 2 key 值 2 ...**

获取 key 值: **get key 名**

获取多个 key 值: **mget key 名 1 key 名 2 ...**

切换库: **select 数据库编号**

 #默认 16 个库,编号 0-15

显示所有变量: **keys ***

显示指定 key 名: **keys a 通配符**

 #显示 a 开头的变量,?匹配 1 个字符,*匹配多个字符

测试 key 名是否存在: **exists key 名**

 #存在返回 1,否则返回 0

显示 key 生存时间: **ttl key 名**

 #-1 表示永不过期 -2 表示过期或变量不存在 正数表示剩余生存时间

查看 key 类型: **tpye key 名**

 #set 命令设置的都是 string 类型

移动 key 到指定库: **move key 名 库编号**

 #成功返回 1,失败返回 0

设置 key 有效时间: **expire key 名 数字**

 #1 表示成功,0 表示失败或变量不存在

删除指定的 key: **del key 名**

#删除成功返回 1, 否则返回 0

删除内存里所有 key: **flushall**

#成功返回 OK

删除所在库的所有 key: **flushdb**

手动保存所有 key 到硬盘: **save**

手动停止服务: **shutdown**

三 配置文件/etc/redis/6379.conf 解析

3.1 配置模块分类

NETWORK	网络
GENERAL	常规
SNAPSHOTTING	快照
REPLICATION	复制
SECURITY	安全
CLIENTS	客户端
MEMORY MANAGEMENT	内存管理

格式: 配置项 值

3.2 配置文件/etc/redis/6379.conf 数据单位

1k => 1000 bytes

1kb => 1024 bytes

1m => 1000000 bytes

1mb => 1024*1024 bytes

1g => 1000000000 bytes

1gb => 1024*1024*1024 bytes

3.3 配置文件/etc/redis/6379.conf 常用配置

端口: 93 port 6379

IP 地址: 70 bind 127.0.0.1

守护进程运行方式: 137 daemonize yes #默认状态

#yes 表示守护进程在内存中一直处于活动状态, no 表示处于休眠状态

数据库个数: 187 databases 16

日志文件: 172 logfile /var/log/redis_6379.log

并发连接量: 533 # maxclients 10000 #默认没开启

数据库目录: 264 dir /var/lib/redis/6379

3.4 /etc/redis/6379.conf 内存管理

560 # maxmemory <bytes> #默认没开启, 剩余多少内存使用多少

3.4.1 /etc/redis/6379.conf 内存清除策略

volatile-lru 565 最近最少使用 (针对设置了 TTL 值的 key)

allkeys-lru 566 最近最少使用 (针对所有的 key)

volatile-lfu 567 从所有配置了过期时间的 key 中清楚使用频率最少的

allkeys-lfu 568 从所有 key 中清楚使用频率最少的

volatile-random 569 在设置了 TTL 的 key 里随机移除

allkeys-random 570 随机移除 key

volatile-ttl (minor TTL) 571 移除最近过期的 key

noeviction 572 不删除,写满时报错

3.4.2 /etc/redis/6379.conf 优化设置,默认被注释没开启

maxmemory 560 最大内存

maxmemory-policy 591 赋值使用 3.4.1 的内存清除策略

maxmemory-samples 602 选取 key 模板的个数(针对 lru 和 ttl 策略)

3.5 设置密码,IP 地址,端口

案例: 修改 Redis 服务运行参数

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

步骤一: 修改主配置文件

1) 修改配置文件

```
mysql50 ~]tils[# 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

```
mysql50 ~]tils[# /etc/init.d/redis_6379 stop
```

```
mysql50 ~]tils[# vim /etc/redis/6379.conf
```

```
...
```

```
70 bind 192.168.4.50 //设置服务使用的 ip
```

```
93 port 6350 //更改端口号
```

```
501 requirepass 123456 //设置密码
```

```
:wq
```

2) 修改启动脚本

```
mysql50 ~]# vim +43 /etc/init.d/redis_6379
```

```
$CLIEEXEC -h 192.168.4.50 -p 6350 -a 123456 shutdown
```

```
:wq
```

3) 启动服务

```
mysql50 ~]# /etc/init.d/redis_6379 start
```

```
Starting Redis server...
```

```
mysql50 ~]#
```

```
mysql50 ~]# 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

```
mysql50 ~]# 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
```

```
mysql50 ~]#
```

四 部署 LNMP+Reids

4.1 部署 LNMP

4.1.1 安装软件

安装源码 nginx 及 php-fpm

```
mysql51 ~]# yum -y install gcc pcre-devel php-fpm zlib-devel
```

```
mysql51 ~]# cd nginx-1.12.2
```

```
mysql51 ~]# ./configure
```

```
mysql51 ~]# make &&make install
```

4.1.2 修改配置

```
mysql51 ~]# vim /usr/local/nginx/conf/nginx.conf
```

```
location ~ \.php$ {                                #解除此段的注释并修改

    root html;

    fastcgi_pass    127.0.0.1:9000;

    fastcgi_index   index.php;

    #fastcgi_param..._script_name; #此句不解除注释

    include    fastcgi.conf;    #修改为 fastcgi.conf

}
```

```
mysql51 ~]# /usr/local/nginx/sbin/nginx -t #测试配置文件的修改
```

4.1.3 启动服务

```
mysql51 ~]# /usr/local/nginx/sbin/nginx #启动 nginx
```

```
mysql51 ~]# ss -antulp | grep :80    #查看 nginx 端口是否开启
```

```
mysql51 ~]# systemctl restart php-fpm #启动 php-fpm
```

```
mysql51 ~]# netstat -antulp | grep :9000 #查看 php-fpm 端口
```

4.1.4 测试配置

编写配置文件

```
mysql51 ~]# vim /usr/local/nginx/html/test.php
```

```
<?php
```

```
    echo "hello world!!!";
```

```
?>
```

访问 nginx 服务

```
mysql51 ~]# curl http://localhost/test.php  
hello world!!!
```

4.2 配置支持 Redis

4.2.1 运行 Redis 服务

```
mysql51 ~]# cd redis/  
mysql51 redis]# tar -xf redis-4.0.8.tar.gz  
mysql51 redis]# cd redis-4.0.8/  
mysql51 redis-4.0.8]# make && make install  
mysql51 redis-4.0.8]# ./utils/install_server.sh
```

一路回车

```
mysql51 redis-4.0.8]# /etc/init.d/redis_6379 status #查看状态  
Redis is running (5494)
```

4.2.2 配置支持 Redis

安装 php 扩展

```
mysql51 ~]# yum -y install php php-devel  
mysql51 ~]# rpm -q autoconf automake  
autoconf-2.69-11.el7.noarch  
automake-1.13.4-3.el7.noarch
```

```
mysql51 ~]# cd /root/redis/lnmp
mysql51 lnmp]# tar -xf php-redis-2.2.4.tar.gz
mysql51 lnmp]# cd phpredis-2.2.4/
mysql51 phpredis-2.2.4]# phpize
```

#生成配置文件 php-configure 及 configure 命令

```
mysql51 phpredis-2.2.4]# ./configure \
--with-php-configure=/usr/bin/php-config
mysql51 phpredis-2.2.4]# make && make install
mysql51 ~]# ls /usr/lib64/php/modules/ #php 模块路径
```

```
mysql51 ~]# vim /etc/php.ini
```

```
728 extension_dir = "./" #模块目录
```

```
730 extension_dir = "ext" #模块名
```

解除注释并修改为

```
extension_dir = "./"
```

```
extension = "redis.so"
```

```
mysql51 ~]# systemctl restart php-fpm #启动服务
```

4.2.3 测试配置

```
mysql51 ~]# php -m | grep -i redis #检查是否支持模块
redis
```

```
mysql51 ~]# vim /usr/local/nginx/html/set.php #编写测试脚本
```

```
<?php
```

```
$redis = new redis();
```

```
$redis->connect('192.168.4.50',6350);
```

```
$redis->auth('123456');
```

```
$redis->set('redistest','666666');
```

```
?>
```

```
mysql51 html]# curl http://localhost/set.php #执行脚本
```

50 登录 redis 查看值

```
mysql50 ~]# 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> get linux #获取值
```

```
"666666"
```

```
mysql51 ~]# vim /usr/local/nginx/html/get.php
```

```
<?php
```

```
$redis = new redis();
```

```
$redis->connect('192.168.4.50',6350);
```

```
$redis->auth('123456');
```

```
echo $redis->get('redistest');
```

```
?>
```

```
mysql51 ~]# curl http://localhost/get.php
```

```
666666
```

```
mysql51 ~]# vim /usr/local/nginx/html/setget.php
```

```
<?php
```

```
$redis = new redis();
```

```
$redis->connect('192.168.4.50',6350);
```

```
$redis->auth('123456');
```

```
$redis->set('site','www.tedu.cn');
```

```
echo $redis->get('site');
```

```
?>
```

```
mysql51 ~]# curl http://localhost/setget.php
```

```
www.tedu.cn
```

```
50 192.168.4.50:6350> keys *
```

```
1) "site"
```

```
50 192.168.4.50:6350> get site
```

```
"www.tedu.cn"
```