













































Nosql\_day01

1 搭建Redis服务器 （192.168.4.50）

1.1 相关概念

关系型数据库系统 RDBMS 服务软件mysql Oracle SQL SERVER

必须事先创建表结构，根据表结构存储数据。数据按照固定格式存储。

银行开户信息： 姓名 性别 年龄 身份证

公安局的户籍信息：姓名 性别 年龄 身份证 身份 家庭地址

非关系型数据库系统 NoSQL 服务软件 memcached redis MongoDB

存储数据的格式不固定，比较随机，也不需要创建表结构，存储数据通常格式 变量key/values值 通常数据都存储的内存了 。

redis软件介绍？

搭建redis服务器192.168.4.50 ，步骤如下

1 安装软件

# rpm -q gcc || yum -y install gcc

# tar -zxvf redis-4.0.8.tar.gz

# cd redis-4.0.8

# make

# make install

2 初始配置

]# ./utils/install\_server.sh

3 查看服务状态

]# netstat -utnlp | grep 6379

tcp 0 0 127.0.0.1:6379 0.0.0.0:\* LISTEN 14869/redis-server

[root@host50 redis-4.0.8]#

4 连接Redis服务

[root@host50 ~]# redis-cli

127.0.0.1:6379> exit 断开连接

[root@host50 ~]#

5 存取数据

[root@host50 redis-4.0.8]# redis-cli

127.0.0.1:6379> set x 9

OK

127.0.0.1:6379> set school tarena

OK

127.0.0.1:6379> get x

"9"

127.0.0.1:6379> get school

"tarena"

127.0.0.1:6379>

127.0.0.1:6379> keys \*

1) "school"

2) "x"

127.0.0.1:6379> exit

[root@host50 redis-4.0.8]#

停止redis服务

[root@host50 redis-4.0.8]# /etc/init.d/redis\_6379 stop

Stopping ...

Redis stopped

[root@host50 redis-4.0.8]# netstat -utnlp | grep 6379

[root@host50 redis-4.0.8]#

启动Redis服务

[root@host50 redis-4.0.8]# /etc/init.d/redis\_6379 start

Starting Redis server...

[root@host50 redis-4.0.8]#

[root@host50 redis-4.0.8]# netstat -utnlp | grep 6379

tcp 0 0 127.0.0.1:6379 0.0.0.0:\* LISTEN 5604/redis-server 1

[root@host50 redis-4.0.8]#

redis常用命令的使用（10分钟）

127.0.0.1:6379> mset i 99 j 88 k 77

127.0.0.1:6379> mget i j k

127.0.0.1:6379> select 1

OK

127.0.0.1:6379[1]>

127.0.0.1:6379[1]> keys ???? #查看变量名是4个字母的变量名

127.0.0.1:6379[1]> keys a?? #查看变量名是3个字母的且首字母必须是a

127.0.0.1:6379[1]> keys \* # 查看所有变量名

127.0.0.1:6379[1]> keys age # 查看是否名变量名叫age 的变量

127.0.0.1:6379[1]> EXISTS age

(integer) 0 表示变量不存储

127.0.0.1:6379[1]> EXISTS a1

(integer) 1 表示变量存储

127.0.0.1:6379[1]> ttl a1

(integer) -1

127.0.0.1:6379[1]> EXPIRE a1 20 #设置过期时间是20秒

127.0.0.1:6379[1]> ttl a1

(integer) -2 变量过期已经被删除的变量

127.0.0.1:6379[1]>

127.0.0.1:6379[1]> set i 99

OK

127.0.0.1:6379[1]> type i

string

127.0.0.1:6379[1]> lpush tea nb wk dmy plj

127.0.0.1:6379[1]> type tea

list

127.0.0.1:6379[1]> LRANGE tea 0 -1

127.0.0.1:6379[1]> keys \*

1) "tea"

2) "i"

3) "name"

127.0.0.1:6379[1]> MOVE tea 0 #把变量tea 移动到 0 号库里

127.0.0.1:6379> del tea 删除一个变量

127.0.0.1:6379> flushdb 删除当前所在库的所有数据

127.0.0.1:6379> FLUSHALL 清空内存

save //立刻把内存的数据保存到硬盘

shutdown 停止redis服务

not connected> exit

[root@host50 redis-4.0.8]# netstat -utnlp | grep 6379

[root@host50 redis-4.0.8]# /etc/init.d/redis\_6379 start

Starting Redis server...

[root@host50 redis-4.0.8]# netstat -utnlp | grep 6379

tcp 0 0 127.0.0.1:6379 0.0.0.0:\* LISTEN 15435/redis-server

[root@host50 redis-4.0.8]#

[root@host50 redis-4.0.8]#

[root@host50 redis-4.0.8]# redis-cli

127.0.0.1:6379> select 1

OK

127.0.0.1:6379[1]>

配置文件解析 /etc/redis/6379.conf

vim /etc/redis/6379.conf

配置项 值

：wq

重启redis服务

]# /etc/init.d/redis\_6379 stop

]# /etc/init.d/redis\_6379 start

守护进程

非守护进程

常驻部队（无论有没有敌人入侵部队都驻扎在边境不动）

驻守部队

机动部队（发现敌人入侵时，才派兵抵御）

配置文件解析之内存清除策略 (30分钟 存理论) 休息10分钟到 10:20

LRU 删除最近最少使用的（allkeys-lru volatile-lru）

LFU 删除最少使用的(volatile-lfu allkeys-lfu)

random 随机删除 (volatile-random allkeys-random)

volatile-ttl 删除快要过期 900M 1000变量 400个设置了ttl时间

noeviction 不删除

相关的配置项

maxmemory //最大内存

maxmemory-policy //定义使用策略

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

修改服务使用的地址 端口 和 连接密码

vim /etc/redis/6379.conf

bind 192.168.4.50

port 6350

requirepass 123456

：wq

重启redis服务

]# /etc/init.d/redis\_6379 stop

]# /etc/init.d/redis\_6379 start

]# redis-cli -h 192.168.4.50 -p 6350 -a 123456

> ping

> keys \*

> exit

修改服务使用的地址 端口 和 连接密码 练习到 16:52

vim /etc/redis/6379.conf

bind 192.168.4.50

port 6350

requirepass 123456

：wq

重启redis服务

]# /etc/init.d/redis\_6379 stop

]# /etc/init.d/redis\_6379 start

]# redis-cli -h 192.168.4.50 -p 6350 -a 123456

> ping

> keys \*

> exit

修改ip 端口 密码之后停止服务方法

[root@host50 ~]# vim +43 /etc/init.d/redis\_6379

43 $CLIEXEC -h 192.168.4.50 -p 6350 -a 123456 shutdown

:wq

或者

执行停止命令

[root@host50 ~]# redis-cli -h 192.168.4.50 -p 6350 -a 123456 shutdown

2 部署LNMP+redis

2.1 部署LNMP环境 192.168.4.51

249 yum -y install gcc pcre-devel zlib-devel

251 tar -zxvf nginx-1.12.2.tar.gz

253 cd nginx-1.12.2

254 ./configure

255 make && make install

261 yum -y install php php-devel php-fpm

[root@localhost nginx-1.12.2]# systemctl start php-fpm

[root@localhost nginx-1.12.2]# netstat -utnlp | grep 9000

tcp 0 0 127.0.0.1:9000 0.0.0.0:\* LISTEN 10355/php-fpm: mast

[root@localhost nginx-1.12.2]#

[root@localhost nginx-1.12.2]# 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

[root@localhost nginx-1.12.2]# /usr/local/nginx/sbin/nginx -t

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

[root@localhost nginx-1.12.2]#

[root@localhost nginx-1.12.2]# /usr/local/nginx/sbin/nginx

[root@localhost nginx-1.12.2]#

[root@localhost nginx-1.12.2]# netstat -utnlp | grep 80

tcp 0 0 0.0.0.0:80 0.0.0.0:\* LISTEN 12334/nginx: master

[root@localhost nginx-1.12.2]#

[root@localhost nginx-1.12.2]# vim /usr/local/nginx/html/test.php

<?php

$i = 99 ;

echo $i ;

?>

:wq

[root@localhost nginx-1.12.2]# setenforce 0

[root@localhost nginx-1.12.2]# systemctl stop firewalld

[root@localhost nginx-1.12.2]# curl http://localhost/test.php

2.2 配置php支持Redis服务器 192.168.4.51

安装软件提供模块

279 tar -zxvf php-redis-2.2.4.tar.gz

281 cd phpredis-2.2.4/

283 which phpize

284 phpize

286 ./configure --with-php-config=/usr/bin/php-config

287 make

288 make install

[root@localhost phpredis-2.2.4]# make install

Installing shared extensions: /usr/lib64/php/modules/

[root@localhost phpredis-2.2.4]#

[root@localhost phpredis-2.2.4]# ls /usr/lib64/php/modules/

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

[root@localhost phpredis-2.2.4]#

调用模块

]# vim /etc/php.ini

728 extension\_dir = "/usr/lib64/php/modules/"

730 extension = "redis.so"

:wq

]# systemctl restart php-fpm

查看模块

[root@localhost ~]# php -m | grep -i redis

redis

[root@localhost ~]#

编写存储数据的PHP脚本

vim /usr/local/nginx/html/set.php

<?php

$redis = new redis(); 定义连接命令

$redis->connect("192.168.4.50","6350");

$redis->auth("123456");

$redis->set("school","tarena"); //存数据

$redis->set("class","nsd2005");

echo "data save ok ";

?>

:wq

编写获取数据的PHP脚本

vim /usr/local/nginx/html/get.php

<?php

$redis = new redis();

$redis->connect("192.168.4.50","6350");

$redis->auth("123456");

echo $redis->get("school");

echo $redis->get("class");

?>

[root@localhost ~]# curl http://localhost/set.php

data save ok[root@localhost ~]#

验证：在Redis服务器本机查看数据 192.168.4.50

[root@host50 ~]# redis-cli -h 192.168.4.50 -p 6350 -a 123456

192.168.4.50:6350> keys \*

1) "class"

2) "school"

192.168.4.50:6350> get class

"nsd2005"

192.168.4.50:6350> get school

"tarena"

192.168.4.50:6350>

客户端访问网站服务器的php脚本取数据

[root@localhost ~]# curl http://localhost/get.php

tarena nsd2005[root@localhost ~]#

1. [案例1：搭建Redis服务器](http://tts.tmooc.cn/ttsPage/LINUX/NSDTN202001/NOSQL/DAY01/CASE/01/index.html#case1)
2. [案例2：常用命令](http://tts.tmooc.cn/ttsPage/LINUX/NSDTN202001/NOSQL/DAY01/CASE/01/index.html#case2)
3. [案例3：修改Redis服务运行参数](http://tts.tmooc.cn/ttsPage/LINUX/NSDTN202001/NOSQL/DAY01/CASE/01/index.html#case3)
4. [案例4：部署LNMP+Redis](http://tts.tmooc.cn/ttsPage/LINUX/NSDTN202001/NOSQL/DAY01/CASE/01/index.html#case4)

**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school，值为tarena
* 查看变量school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）安装源码redis软件

1. **[**root@redis1 redis**]**# yum **-**y install gcc
2. **[**root@redis1 redis**]**# tar **-**zxf redis**-4.0.8.**tar**.**gz
3. **[**root@redis1 redis**]**# cd redis**-4.0.8/**
4. **[**root@redis1 redis**-4.0.8]**# ls
5. **00-**RELEASENOTES CONTRIBUTING deps Makefile README**.**md runtest runtest**-**sentinel src utils
6. BUGS COPYING INSTALL MANIFESTO redis**.**conf runtest**-**cluster sentinel**.**conf tests
7. **[**root@redis1 redis**-4.0.8]**# make
8. **[**root@redis1 redis**-4.0.8]**# make install
9. **[**root@redis1 redis**-4.0.8]**# cd utils**/**
10. **[**root@redis1 utils**]**# **./**install\_server**.**sh
11. Welcome to the redis service installer
12. This script will help you easily set up a running redis server
13. Please select the redis port **for** **this** instance**:** **[6379]**
14. Selecting **default:** **6379**
15. Please select the redis config file name **[**/etc/redis**/6379.**conf**]**
16. Selected **default** **-** /etc/redis**/6379.**conf
17. Please select the redis log file name **[**/var/log**/**redis\_6379**.**log**]**
18. Selected **default** **-** /var/log**/**redis\_6379**.**log
19. Please select the data directory **for** **this** instance **[**/var/lib**/**redis**/6379]**
20. Selected **default** **-** /var/lib**/**redis**/6379**
21. Please select the redis executable path **[**/usr/local**/**bin**/**redis**-**server**]**
22. Selected config**:**
23. Port **:** **6379**                  //端口号
24. Config file **:** /etc/redis**/6379.**conf //配置文件目录
25. Log file **:** /var/log**/**redis\_6379**.**log //日志目录
26. Data dir **:** /var/lib**/**redis**/6379** //数据库目录
27. Executable **:** /usr/local**/**bin**/**redis**-**server //启动程序的目录
28. Cli Executable **:** /usr/local**/**bin**/**redis**-**cli //命令行的连接工具
29. Is **this** ok**?** Then press ENTER to go on or Ctrl**-**C to abort**.** //回车完成配置
30. Copied **/**tmp**/6379.**conf **=>** /etc/init**.**d**/**redis\_6379 //服务启动脚本
31. Installing service**...**
32. Successfully added to chkconfig**!**
33. Successfully added to runlevels **345!**
34. Starting Redis server**...** //提示服务已经启动
35. Installation successful**!**        //提示安装成功

2）查看服务状态

1. **[**root@redis1 utils**]**# /etc/init**.**d**/**redis\_6379 status
2. Redis is running **(15203)**

3）查看监听的端口

1. **[**root@redis1 utils**]**# netstat **-**antupl **|**grep **:6379** //查看端口
2. tcp **0** **0** **127.0.0.1:6379** **0.0.0.0:\*** LISTEN **15203/**redis**-**server
3. **[**root@redis1 utils**]**# ps **-**C redis**-**server //查看进程
4. PID TTY TIME CMD
5. **15203** **?** **00:00:00** redis**-**server

4）停止服务

1. **[**root@redis1 utils**]**# /etc/init**.**d**/**redis\_6379 stop
2. Stopping **...**
3. Waiting **for** Redis to shutdown **...**
4. Redis stopped

5）连接redis

1. **[**root@redis1 utils**]**# /etc/init**.**d**/**redis\_6379 start
2. Starting Redis server**...**
3. **[**root@redis1 utils**]**# redis**-**cli //默认连接127.0.0.1地址的 6379端口
4. **127.0.0.1:6379>** ping
5. 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

1. **127.0.0.1:6379>** set school tarena
2. OK
3. **127.0.0.1:6379>** get school
4. "tarena"
5. **127.0.0.1:6379>**

**2 案例2：常用命令**

**2.1 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

**2.2 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

1. **192.168.4.50:6350>** set name bob
2. OK
3. **192.168.4.50:6350>**
4. **192.168.4.50:6350>** mset age **19** sex boy
5. OK
6. **192.168.4.50:6350>**
7. **192.168.4.50:6350>** get name
8. "bob"
9. **192.168.4.50:6350>**
10. **192.168.4.50:6350>** mget age sex
11. **1)** "19"
12. **2)** "boy"
13. **192.168.4.50:6350>**
14. **192.168.4.50:6350>**

**步骤二：命令keys 、 type 、 exists 、 del**

具体操作如下

1. **192.168.4.50:6350>** keys **\***
2. **1)** "sex"
3. **2)** "age"
4. **3)** "name"
5. **192.168.4.50:6350>**
6. **192.168.4.50:6350>** keys **???**
7. **1)** "sex"
8. **2)** "age"
9. **192.168.4.50:6350>** keys a**\***
10. **1)** "age"
11. **192.168.4.50:6350>**
12. **192.168.4.50:6350>** type age //使用set命令存储的变量都是字符类型
13. string
14. **192.168.4.50:6350>**
15. **192.168.4.50:6350>** del age
16. **(**integer**)** **1**
17. **192.168.4.50:6350>**
18. **192.168.4.50:6350>** exists age //变量不存储返回值0
19. **(**integer**)** **0**
20. **192.168.4.50:6350>**
21. **192.168.4.50:6350>** exists sex //变量存在 返回值1
22. **(**integer**)** **1**
23. **192.168.4.50:6350>**

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

**、select**

具体操作如下

1. **192.168.4.50:6350>** keys **\***
2. **1)** "sex"
3. **2)** "name"
4. **192.168.4.50:6350>** ttl sex //返回值-1 表示变量永不过期
5. **(**integer**)** **-1**
6. **192.168.4.50:6350>**
7. **192.168.4.50:6350>** expire sex **20** //设置变量过期时间为 20 秒
8. **(**integer**)** **1**
9. **192.168.4.50:6350>**
10. **192.168.4.50:6350>** ttl sex //还剩14秒过期
11. **(**integer**)** **14**
12. **192.168.4.50:6350>**
13. **192.168.4.50:6350>** ttl sex //返回值-2 表示已经过期
14. **(**integer**)** **-2**
15. **192.168.4.50:6350>** exists sex //变量已经不存在
16. **(**integer**)** **0**
17. **192.168.4.50:6350>**
18. **192.168.4.50:6350>** move name **1** //把变量name移动到1号库里
19. **(**integer**)** **1**
20. **192.168.4.50:6350>**
21. **192.168.4.50:6350>** select **1** //切换到1号库
22. OK
23. **192.168.4.50:6350[1]>** keys **\*** //查看
24. **1)** "name"
25. **192.168.4.50:6350[1]>** select **0** //切换到0号库
26. OK
27. **192.168.4.50:6350>** keys **\*** //查看
28. **(**empty list or set**)**
29. **192.168.4.50:6350>**
30. **192.168.4.50:6350>** select **1** //切换到1号库
31. OK
32. **192.168.4.50:6350[1]>**
33. **192.168.4.50:6350[1]>** keys **\***
34. **1)** "name"
35. **192.168.4.50:6350[1]>**
36. **192.168.4.50:6350[1]>** flushdb
37. OK
38. **192.168.4.50:6350[1]>**
39. **192.168.4.50:6350[1]>** keys **\***
40. **(**empty list or set**)**
41. **192.168.4.50:6350[1]>**
42. **192.168.4.50:6350[1]>** flushall
43. OK
44. **192.168.4.50:6350[1]>**
45. **192.168.4.50:6350[1]>** save
46. OK
47. **192.168.4.50:6350[1]>**
48. **192.168.4.50:6350[1]>** shutdown
49. not connected**>** //提示连接断开
50. not connected**>** exit //退出登录
51. **[**root@host50 **~]**#
52. **[**root@host50 **~]**# netstat **-**utnlp **|** grep redis**-**server //没有进程信息
53. **[**root@host50 **~]**#
54. **[**root@host50 **~]**# /etc/init**.**d**/**redis\_6379 start //启动服务
55. Starting Redis server**...**
56. **[**root@host50 **~]**#
57. **[**root@host50 **~]**# netstat **-**utnlp **|** grep redis**-**server //查看进程信息
58. tcp **0** **0** **192.168.4.50:6350** **0.0.0.0:\*** LISTEN **11475/**redis**-**server
59. **[**root@host50 **~]**#

**3 案例3：修改Redis服务运行参数**

**3.1 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

**3.2 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

1. **[**root@host50 utils**]**# cp **/**etc**/**redis**/6379.**conf **/**root**/6379.**conf
2. //可以先备份一份，防止修改错误没法还原
3. **[**root@host50 utils**]**# /etc/init**.**d**/**redis\_6379 stop
4. **[**root@host50 utils**]**# vim **/**etc**/**redis**/6379.**conf
5. **...**
6. bind **192.168.4.50**                //设置服务使用的ip
7. port **6350**                            //更改端口号
8. requirepass **123456**                //设置密码
9. ：wq

2）修改启动脚本

1. **[**root@host50 **~]**# vim **+43** **/**etc**/**init**.**d**/**redis\_6379
2. $CLIEXEC **-**h **192.168.4.50** **-**p **6350** **-**a **123456** shutdown
3. **:**wq

3）启动服务

1. **[**root@host50 **~]**# /etc/init**.**d**/**redis\_6379 start
2. Starting Redis server**...**
3. **[**root@host50 **~]**#
4. **[**root@host50 **~]**# netstat **-**utnlp **|** grep redis**-**server
5. tcp **0** **0** **192.168.4.50:6350** **0.0.0.0:\*** LISTEN **11523/**redis**-**server
6. **[**root@host50 **~]**#

4）测试配置

访问服务存取数据

1. **[**root@host50 **~]**# redis**-**cli **-**h **192.168.4.50** **-**p **6350** **-**a **123456** //访问服务
2. **192.168.4.50:6350>** ping
3. PONG
4. **192.168.4.50:6350>** keys **\***
5. **(**empty list or set**)**
6. **192.168.4.50:6350>**
7. **192.168.4.50:6350>** set x **99**
8. OK
9. **192.168.4.50:6350>**
10. **192.168.4.50:6350>** exit
11. **[**root@host50 **~]**#

**4 案例4：部署LNMP+Redis**

**4.1 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

**4.2 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

1. **]**#yum **-**y install gcc pcre**-**devel zlib**-**devel //安装依赖
2. **]**#tar **-**zxvf nginx**-1.12.2.**tar**.**gz //解压
3. **]**#cd nginx**-1.12.2** //进源码目录
4. **]**#**.**/configure /**/**配置
5. ……
6. ……
7. Configuration summary
8. **+** using system PCRE library
9. **+** OpenSSL library is not used
10. **+** using system zlib library
11. nginx path prefix**:** "/usr/local/nginx"
12. nginx binary file**:** "/usr/local/nginx/sbin/nginx"
13. nginx modules path**:** "/usr/local/nginx/modules"
14. nginx configuration prefix**:** "/usr/local/nginx/conf"
15. nginx configuration file**:** "/usr/local/nginx/conf/nginx.conf"
16. nginx pid file**:** "/usr/local/nginx/logs/nginx.pid"
17. nginx error log file**:** "/usr/local/nginx/logs/error.log"
18. nginx http access log file**:** "/usr/local/nginx/logs/access.log"
19. nginx http client request body temporary files**:** "client\_body\_temp"
20. nginx http proxy temporary files**:** "proxy\_temp"
21. nginx http fastcgi temporary files**:** "fastcgi\_temp"
22. nginx http uwsgi temporary files**:** "uwsgi\_temp"
23. nginx http scgi temporary files**:** "scgi\_temp"
24. **[**root@localhost nginx**-1.12.2]**# make //编译
25. ……
26. ……
27. objs**/**src**/**http**/**modules**/**ngx\_http\_upstream\_zone\_module**.**o **\**
28. objs**/**ngx\_modules**.**o **\**
29. **-**ldl **-**lpthread **-**lcrypt **-**lpcre **-**lz **\**
30. **-**Wl**,-**E
31. sed **-**e "s|%%PREFIX%%|/usr/local/nginx|" **\**
32. **-**e "s|%%PID\_PATH%%|/usr/local/nginx/logs/nginx.pid|" **\**
33. **-**e "s|%%CONF\_PATH%%|/usr/local/nginx/conf/nginx.conf|" **\**
34. **-**e "s|%%ERROR\_LOG\_PATH%%|/usr/local/nginx/logs/error.log|" **\**
35. **<** man**/**nginx**.8** **>** objs**/**nginx**.8**
36. make**[1]:** 离开目录“/root/lnmp**/**nginx**-1.12.2**”
37. **[**root@localhost nginx**-1.12.2]**#make install //安装
38. ……
39. ……
40. test **-**d '/usr/local/nginx/logs' **\**
41. **||** mkdir **-**p '/usr/local/nginx/logs'
42. test **-**d '/usr/local/nginx/html' **\**
43. **||** cp **-**R html '/usr/local/nginx'
44. test **-**d '/usr/local/nginx/logs' **\**
45. **||** mkdir **-**p '/usr/local/nginx/logs'
46. make**[1]:** 离开目录“/root/lnmp**/**nginx**-1.12.2**”
47. **[**root@localhost nginx**-1.12.2]**# ls **/**usr**/**local //查看安装目录
48. bin etc games include lib lib64 libexec nginx sbin share src
49. **[**root@localhost nginx**-1.12.2]**#
50. **[**root@localhost nginx**-1.12.2]**# ls **/**usr**/**local**/**nginx //查看目录列表
51. conf html logs sbin
52. **[**root@localhost nginx**-1.12.2]**#
53. **]**#yum **-**y install php**-**fpm //安装php-fpm
54. ……
55. ……
56. 已安装**:**
57. php**-**fpm**.**x86\_64 **0:5.4.16-45.**el7
58. 作为依赖被安装**:**
59. libzip**.**x86\_64 **0:0.10.1-8.**el7 php**-**common**.**x86\_64 **0:5.4.16-45.**el7
60. 完毕！

2）修改配置nginx.conf

1. **]** # vim **+65** **/**usr**/**local**/**nginx**/**conf**/**nginx**.**conf
2. location **~** **\.**php$ **{**
3. root html**;**
4. fastcgi\_pass **127.0.0.1:9000;**
5. fastcgi\_index index**.**php**;**
6. include fastcgi**.**conf**;**
7. **}**
8. **:**wq
9. **]**# /usr/local**/**nginx**/**sbin**/**nginx **-**t     //测试修改
10. nginx**:** the configuration file **/**usr**/**local**/**nginx**/**conf**/**nginx**.**conf syntax is ok
11. nginx**:** configuration file **/**usr**/**local**/**nginx**/**conf**/**nginx**.**conf test is successful

3）启动服务

启动php-fpm服务

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

启动nginx服务

1. **]**# /usr/local**/**nginx**/**sbin**/**nginx
2. **]**# netstat **-**utnlp **|** grep **:80**
3. tcp **0** **0** **0.0.0.0:80** **0.0.0.0:\*** LISTEN **23505/**nginx**:** master

4）测试配置

1. **]**# vim **/**usr**/**local**/**nginx**/**html**/**test**.**php //编写php文件
2. **<?**php
3. echo "hello world!!!"**;**
4. **?>**
5. **:**wq
6. **]**# curl http**:**//localhost/test.php //访问nginx服务
7. hello world**!!!**

**步骤二：配置PHP支持redis**

1）安装php扩展

1. **[**root@host71 **~]**# rpm **-**q php php**-**devel
2. 未安装软件包 php
3. 未安装软件包 php**-**devel
4. **[**root@host71 **~]**#
5. **[**root@host71 **~]**# rpm **-**q automake autoconf
6. 未安装软件包 automack
7. 未安装软件包 autoconf
8. **[**root@host71 **~]**#
9. **[**root@host71 **~]**# yum **-**y install php php**-**devel automake autoconf //安装依赖
10. **]**# tar **-**zxf php**-**redis**-2.2.4.**tar**.**gz //安装扩展包
11. **]**# cd phpredis**-2.2.4/**
12. **]**# phpize //生成配置文件php-config及 configure命令
13. Configuring **for:**
14. PHP Api Version**:** **20100412**
15. Zend Module Api No**:** **20100525**
16. Zend Extension Api No**:** **220100525**
17. **]**# **.**/configure --with-php-config=/usr**/**bin**/**php**-**config //配置
18. **]**# make //编译
19. **]**# make install //安装

2）修改php.ini文件

1. **]**#vim **/**etc**/**php**.**ini
2. **728** extension\_dir **=** "/usr/lib64/php/modules/" //模块文件目录
3. **730** extension **=** "redis.so" //模块文件名
4. **:**wq
5. **]**# systemctl restart php**-**fpm //重启php-fpm服务
6. **]**# php **-**m **|** grep **-**i redis //查看已加载的模块
7. 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

1. **[**root@host50 **~]**# netstat **-**utnlp **|** grep redis**-**server
2. tcp **0** **0** **192.168.4.50:6350** **0.0.0.0:\*** LISTEN **11523/**redis**-**server
3. **[**root@host50 **~]**#
4. **[**root@host50 **~]**# redis**-**cli **-**h **192.168.4.50** **-**p **6350** **-**a **123456** //访问服务
5. **192.168.4.50:6350>** ping
6. PONG
7. **192.168.4.50:6350>** exit

2）编写网站脚本

1. **]**# vim **/**usr**/**local**/**nginx**/**html**/**linkredis**.**php
2. **<?**php
3. $redis **=** **new** redis**();**
4. $redis**->**connect**(**"192.168.4.50"**,**"6350"**);**
5. $redis**->**auth**(**"123456"**);**
6. $redis**->**set**(**"linux"**,**"redhat"**);**
7. echo $redis**->**get**(**"linux"**);**
8. **?>**
9. **:**wq

3）访问网站脚本

1. **]**#curl http**:**//localhost/linkredis.php     //访问nginx服务
2. redhat

4）在192.168.4.50 服务器，查看数据

1. **[**root@host50 **~]**# redis**-**cli **-**h **192.168.4.50** **-**p **6350** **-**a **123456** //连接redis服务
2. **192.168.4.50:6350>** keys **\*** //查看变量
3. **1)** "linux"
4. **192.168.4.50:6350>**
5. **192.168.4.50:6350>** get linux //获取值
6. "redhat"
7. **192.168.4.50:6350>**