

实 验 报 告

学 号	1902001 1038	姓 名	岳宇轩	专业班级	2019 级计算机科学与技术慧与 卓越工程师班
课程名称	大数据导论			学期	2022 年秋季学期
任课教师	刘洁 刘艳艳	完成日期	2022. 11. 21	上机课时间	周一 56 节（双周）
实 验 名 称	实验 6 NoSQL 数据库练习				

一、实验要求（10%）

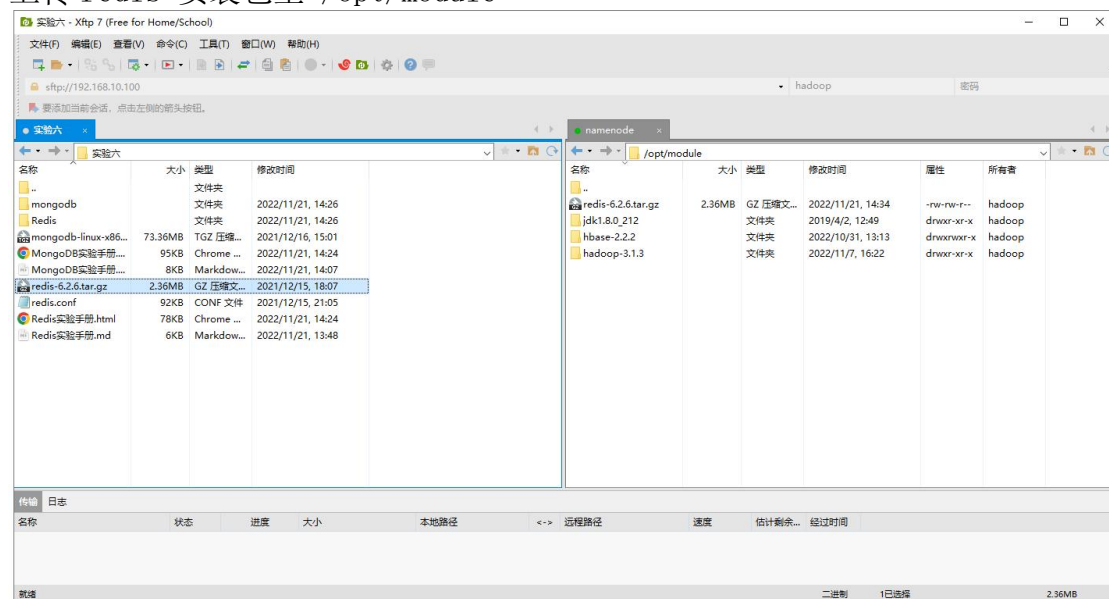
- （1）理解 Redis 和 MongoDB 的概念以及不同点；
- （2）熟练这两种数据库操作常用的 Shell 命令；
- （3）熟悉这两种数据库操作常用的 Java API。

（1）操作系统：Linux（建议 Ubuntu16.04 以上或者 CentOS 7 以上版本）；

- （2）Hadoop 版本：3.1.3 以上版本；
- （5）Redis 版本：5.0.5 以上版本；
- （6）MongoDB 版本：4.0.16 以上版本；
- （7）JDK 版本：1.8；
- （8）Java IDE: Eclipse;

二、实验内容及步骤（80%）

上传 redis 安装包至 /opt/module



修改权限:

```
[hadoop@namenode module]$ sudo chown hadoop:hadoop redis-6.2.6.tar.gz
[hadoop@namenode module]$ chmod 777 redis-6.2.6.tar.gz
[hadoop@namenode module]$ _
```

完成解压步骤:

```
redis-6.2.6/utils/graphs/commits-over-time/README.md
redis-6.2.6/utils/graphs/commits-over-time/genhtml.tcl
redis-6.2.6/utils/hashtable/
redis-6.2.6/utils/hashtable/README
redis-6.2.6/utils/hashtable/rehashing.c
redis-6.2.6/utils/hyperloglog/
redis-6.2.6/utils/hyperloglog/.gitignore
redis-6.2.6/utils/hyperloglog/hll-err.rb
redis-6.2.6/utils/hyperloglog/hll-gnuplot-graph.rb
redis-6.2.6/utils/install_server.sh
redis-6.2.6/utils/lru/
redis-6.2.6/utils/lru/README
redis-6.2.6/utils/lru/lfu-simulation.c
redis-6.2.6/utils/lru/test-lru.rb
redis-6.2.6/utils/redis-copy.rb
redis-6.2.6/utils/redis-sha1.rb
redis-6.2.6/utils/redis_init_script
redis-6.2.6/utils/redis_init_script.tpl
redis-6.2.6/utils/releasetools/
redis-6.2.6/utils/releasetools/01_create_tarball.sh
redis-6.2.6/utils/releasetools/02_upload_tarball.sh
redis-6.2.6/utils/releasetools/03_test_release.sh
redis-6.2.6/utils/releasetools/04_release_hash.sh
redis-6.2.6/utils/releasetools/changelog.tcl
redis-6.2.6/utils/speed-regression.tcl
redis-6.2.6/utils/srandmember/
redis-6.2.6/utils/srandmember/README.md
redis-6.2.6/utils/srandmember/showdist.rb
redis-6.2.6/utils/srandmember/showfreq.rb
redis-6.2.6/utils/systemd-redis_multiple_servers@.service
redis-6.2.6/utils/systemd-redis_server.service
redis-6.2.6/utils/tracking_collisions.c
redis-6.2.6/utils/whatisdoing.sh
[hadoop@namenode module]$ chmod -R 777 redis-6.2.6
[hadoop@namenode module]$ cd redis-6.2.6
[hadoop@namenode redis-6.2.6]$ _
```

安装基本运行环境, 并查看版本信息:

```
Complete!
[hadoop@namenode redis-6.2.6]$ gcc -v
Using built-in specs.
COLLECT_GCC=gcc
COLLECT_LTO_WRAPPER=/usr/libexec/gcc/x86_64-redhat-linux/4.8.5/lto-wrapper
Target: x86_64-redhat-linux
Configured with: ../configure --prefix=/usr --mandir=/usr/share/man --infodir=/usr/share/info --with-bugurl=http://bugzilla.redhat.com/bugzilla --enable-bootstrap --enable-shared --enable-threads=posix --enable-checking=release --with-system-zlib --enable-__cxa_atexit --disable-libunwind-exception --enable-gnu-unique-object --enable-linker-build-id --with-linker-hash-style=gnu --enable-language=c,c++,objc,obj-c++,java,fortran,ada,go,lto --enable-plugin --enable-initfini-array --disable-libgc --with-isl=/builddir/build/BUILD/gcc-4.8.5-20150702/obj-x86_64-redhat-linux/isl-install --with-cloog=/builddir/build/BUILD/gcc-4.8.5-20150702/obj-x86_64-redhat-linux/cloog-install --enable-gnu-indirect-function --with-tune=generic --with-arch_32=x86-64 --build=x86_64-redhat-linux
Thread model: posix
gcc version 4.8.5 20150623 (Red Hat 4.8.5-44) (GCC)
[hadoop@namenode redis-6.2.6]$
```

执行 make 命令后显示:

```
Hint: It's a good idea to run 'make test' ;)

make[1]: Leaving directory `/opt/module/redis-6.2.6/src'
[hadoop@namenode redis-6.2.6]$ _
```

执行 sudo make install

```
[hadoop@namenode redis-6.2.6]$ sudo make install
cd src && make install
make[1]: Entering directory `/opt/module/redis-6.2.6/src'
CC Makefile.dep
make[1]: Leaving directory `/opt/module/redis-6.2.6/src'
make[1]: Entering directory `/opt/module/redis-6.2.6/src'

Hint: It's a good idea to run 'make test' ;)

INSTALL redis-server
INSTALL redis-benchmark
INSTALL redis-cli
make[1]: Leaving directory `/opt/module/redis-6.2.6/src'
[hadoop@namenode redis-6.2.6]$ _
```

在 usr/local/bin 中查看 redis 的安装:

namenode					
/usr/local/bin					
名称	大小	类型	修改时间	属性	所有者
..					
redis-server	9.08MB	文件	2022/11/21, 14:42	-rwxr-xr-x	root
redis-sentinel	9.08MB	文件	2022/11/21, 14:42	lrwxr-xr-x	0
redis-cli	4.77MB	文件	2022/11/21, 14:42	-rwxr-xr-x	root
redis-check-rdb	9.08MB	文件	2022/11/21, 14:42	lrwxr-xr-x	0
redis-check-aof	9.08MB	文件	2022/11/21, 14:42	lrwxr-xr-x	0
redis-benchmark	4.61MB	文件	2022/11/21, 14:42	-rwxr-xr-x	root

修改配置文件:

在 bin 下新建 bconfig 文件夹, 并将助教给的 redis.conf 文件上传

namenode					
/usr/local/bin/bconfig					
名称	大小	类型	修改时间	属性	所有者
..					
redis.conf	92KB	CONF 文件	2022/11/21, 14:47	-rwxr-xr-x	root

修改权限:

```
[hadoop@namenode bin]$ sudo mkdir bconfig
[hadoop@namenode bin]$ sudo cp /opt/module/redis-6.2.6/redis.conf bconfig/
[hadoop@namenode bin]$ cd bconfig
[hadoop@namenode bconfig]$ sudo chmod 777 redis.conf
[hadoop@namenode bconfig]$ _
```

成功显示 PONG:

```
[hadoop@namenode bin]$ redis-server bconfig/redis.conf
[hadoop@namenode bin]$ redis-cli -p 6379
127.0.0.1:6379> ping
PONG
127.0.0.1:6379> _
```

用 redis 的哈希结构设计出学生表 Student

```
127.0.0.1:6379> ping
PONG
127.0.0.1:6379> hset student.zhangsan English 69
(integer) 1
127.0.0.1:6379> hset student.zhangsan Math 86
(integer) 1
127.0.0.1:6379> hset student.zhangsan Computer 77
(integer) 1
127.0.0.1:6379> hset student.lisi English 55
(integer) 1
127.0.0.1:6379> hset student.lisi Math 100
(integer) 1
127.0.0.1:6379> hset student.lisi Computer 88
(integer) 1
127.0.0.1:6379>
```

用 hgetall 命令分别输出 zhangsan 和 lisi 的成绩信息

```
127.0.0.1:6379> hgetall student.zhangsan
1) "English"
2) "69"
3) "Math"
4) "86"
5) "Computer"
6) "77"
127.0.0.1:6379> hgetall student.lisi
1) "English"
2) "55"
3) "Math"
4) "100"
5) "Computer"
6) "88"
127.0.0.1:6379>
```

用 hget 命令查询 zhangsan 的 Computer 成绩

```
127.0.0.1:6379> hget student.zhangsan Computer
"77"
```

修改 lisi 的 Math 成绩，改为 95

```
127.0.0.1:6379> hset student.lisi Math 95
(integer) 0
127.0.0.1:6379> hget student.lisi Math 95
(error) ERR wrong number of arguments for 'hget' command
127.0.0.1:6379> hget student.lisi Math
"95"
127.0.0.1:6379>
```

java api:

①


```

2021.3\plugins\maven\lib\maven-event-listener.jar" "-javaagent:D:\IntelliJ IDEA 202
-Dfile.encoding=UTF-8 -classpath D:\BIGDATA\apache-maven-3.5.4\boot\plexus-classwor
.version=2021.3 -s D:\BIGDATA\apache-maven-3.5.4\conf\settings.xml -Dmaven.repo.loc
\Redis "-Dexec.args=-classpath %classpath com.ben.redistest" "-Dexec.executable=C:\
[INFO] Scanning for projects...
[INFO]
[INFO] -----< org.example:Redis >-----
[INFO] Building Redis 1.0-SNAPSHOT
[INFO] -----[ jar ]-----
[INFO]
[INFO] --- exec-maven-plugin:3.1.0:exec (default-cli) @ Redis ---
Math:89
Computer:100
English:45
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 1.326 s
[INFO] Finished at: 2022-11-21T18:32:43+08:00
[INFO] -----

Process finished with exit code 0

```

②

```

"C:\Program Files\Java\jdk1.8.0_202\bin\java.exe" -Dmaven.multiModuleProjectDirectory=C:\Users\vyx\Desktop\实验六\Redis -Dmaven
.home=D:\BIGDATA\apache-maven-3.5.4 -Dclassworlds.conf=D:\BIGDATA\apache-maven-3.5.4\bin\m2.conf "-Dmaven.ext.class.path=D:\IntelliJ IDEA
2021.3\plugins\maven\lib\maven-event-listener.jar" "-javaagent:D:\IntelliJ IDEA 2021.3\lib\idea_rt.jar=53379:D:\IntelliJ IDEA 2021.3\bin"
-Dfile.encoding=UTF-8 -classpath D:\BIGDATA\apache-maven-3.5.4\boot\plexus-classworlds-2.5.2.jar org.codehaus.classworlds.Launcher -Didea
.version=2021.3 -s D:\BIGDATA\apache-maven-3.5.4\conf\settings.xml -Dmaven.repo.local=D:\maven3\repo -Dexec.workingdir=C:\Users\vyx\Desktop\实验六
\Redis "-Dexec.args=-classpath %classpath com.ben.redistest" "-Dexec.executable=C:\Program Files\Java\jdk1.8.0_202\bin\java.exe" exec:exec
[INFO] Scanning for projects...
[INFO]
[INFO] -----< org.example:Redis >-----
[INFO] Building Redis 1.0-SNAPSHOT
[INFO] -----[ jar ]-----
[INFO]
[INFO] --- exec-maven-plugin:3.1.0:exec (default-cli) @ Redis ---
scofield's English score is: 45
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 0.670 s
[INFO] Finished at: 2022-11-21T18:34:32+08:00
[INFO] -----

Process finished with exit code 0

```

上传 mogodb 的压缩包到/opt/software

namenode					
/opt/software					
名称	大小	类型	修改时间	属性	所有者
..					
mongodb-linux-x86...	73.36MB	TGZ 压缩...	2022/11/21, 18:42	-rw-rw-r--	hadoop
jdk-8u212-linux-x64...	185.98MB	GZ 压缩文...	2022/9/14, 15:35	-rw-rw-r--	hadoop
hbase-2.2.2.7z	205.71MB	7Z 压缩文件	2022/10/31, 12:40	-rw-rw-r--	hadoop
hadoop-3.1.3.tar.gz	322.41MB	GZ 压缩文...	2022/9/16, 14:02	-rw-rw-r--	hadoop

修改权限:

```
[hadoop@namenode opt]$ cd software
[hadoop@namenode software]$ sudo chown hadoop:hadoop mongodb-linux-x86_64-3.2.12.tgz
[hadoop@namenode software]$ chmod 777 mongodb-linux-x86_64-3.2.12.tgz
[hadoop@namenode software]$
```

使用 vim 配置系统文件 profile

```
HOSTNAME=`/usr/bin/hostname 2>/dev/null`
HISTSIZE=1000
if [ "$HISTCONTROL" = "ignorespace" ] ; then
    export HISTCONTROL=ignoreboth
else
    export HISTCONTROL=ignoredups
fi
export PATH USER LOGNAME MAIL HOSTNAME HISTSIZE HISTCONTROL

# By default, we want umask to get set. This sets it for login shell
# Current threshold for system reserved uid/gids is 200
# You could check uidgid reservation validity in
# /usr/share/doc/setup-*/uidgid file
if [ $UID -gt 199 ] && [ "`/usr/bin/id -gn`" = "`/usr/bin/id -un`" ]; then
    umask 002
else
    umask 022
fi
for i in /etc/profile.d/*.sh /etc/profile.d/sh.local ; do
    if [ -r "$i" ] ; then
        if [ "${-#*i}" != "$-" ] ; then
            . "$i"
        else
            . "$i" >>/dev/null
        fi
    fi
done
unset i
unset -f pathmunge

export MONGODB_HOME=/opt/module/mongodb-linux-x86_64-3.2.12
export PATH=$PATH:$MONGODB_HOME/bin
:wq!
```

配置部署:

```
[hadoop@namenode mongodb-linux-x86_64-3.2.12]$ sudo mkdir -p data/db
[hadoop@namenode mongodb-linux-x86_64-3.2.12]$ sudo chown -R hadoop:hadoop data/
[hadoop@namenode mongodb-linux-x86_64-3.2.12]$ sudo chmod -R 777 data/db
[hadoop@namenode mongodb-linux-x86_64-3.2.12]$ sudo mkdir logs
[hadoop@namenode mongodb-linux-x86_64-3.2.12]$ cd logs
[hadoop@namenode logs]$ sudo touch mongod.log
[hadoop@namenode logs]$ sudo chown -R hadoop:hadoop logs
chown: cannot access 'logs': No such file or directory
[hadoop@namenode logs]$ cd ..
[hadoop@namenode mongodb-linux-x86_64-3.2.12]$ sudo chown -R hadoop:hadoop logs
[hadoop@namenode mongodb-linux-x86_64-3.2.12]$
```

mongodb 启动配置

```
dbpath=/opt/module/mongodb-linux-x86_64-3.2.12/data/db
logpath=/opt/module/mongodb-linux-x86_64-3.2.12/logs/mongod.log
port=27017
fork=true
nohttpinterface=true
```

启动 mongod 服务, 以配置文件的方式启动

```
[hadoop@namenode bin]$ ./mongod -f mongod.conf
about to fork child process, waiting until server is ready for connections.
forked process: 1347
child process started successfully, parent exiting
[hadoop@namenode bin]$
```

连接 mongodb 数据库:

```
[hadoop@namenode bin]$ ./mongo
MongoDB shell version: 3.2.12
connecting to: test
Welcome to the MongoDB shell.
For interactive help, type "help".
For more comprehensive documentation, see
  http://docs.mongodb.org/
Questions? Try the support group
  http://groups.google.com/group/mongodb-user
Server has startup warnings:
2022-11-21T19:11:50.844+0800 I CONTROL [initandlisten]
2022-11-21T19:11:50.844+0800 I CONTROL [initandlisten] ** WARNING: /sys/kernel/mm/transparent_hugep
age/enabled is 'always'.
2022-11-21T19:11:50.844+0800 I CONTROL [initandlisten] **          We suggest setting it to 'never'
2022-11-21T19:11:50.844+0800 I CONTROL [initandlisten]
2022-11-21T19:11:50.844+0800 I CONTROL [initandlisten] ** WARNING: /sys/kernel/mm/transparent_hugep
age/defrag is 'always'.
2022-11-21T19:11:50.844+0800 I CONTROL [initandlisten] **          We suggest setting it to 'never'
2022-11-21T19:11:50.844+0800 I CONTROL [initandlisten]
> _
```

首先, 切换到 student 集合, 命令如下:


```

> use student
switched to db student
> _

```

其次，定义包含上述两个文档的数组，命令如下：

最后调用如下命令插入数据库：

```

> var stus=[{"name":"zhangsan","scores":{"English":69,"Math":86,"Computer":77}},{"name":"lisi","score":{"English":55,"Math":100,"Computer":88}}]
> db.student.insert(stus)
BulkWriteResult({
  "writeErrors" : [ ],
  "writeConcernErrors" : [ ],
  "nInserted" : 2,
  "nUpserted" : 0,
  "nMatched" : 0,
  "nModified" : 0,
  "nRemoved" : 0,
  "upserted" : [ ]
})
>

```

用 find()方法输出两个学生信息的命令如下：

```

> db.student.find().pretty()
{
  "_id" : ObjectId("637b5f9821ccdceacdd57bb0"),
  "name" : "zhangsan",
  "scores" : {
    "English" : 69,
    "Math" : 86,
    "Computer" : 77
  }
}
{
  "_id" : ObjectId("637b5f9821ccdceacdd57bb1"),
  "name" : "lisi",
  "score" : {
    "English" : 55,
    "Math" : 100,
    "Computer" : 88
  }
}
>

```

可以看到有自动生成的 id

用 find 函数查询 zhangsan 的所有成绩的命令如下：

```

> db.student.find({"name":"zhangsan"},{"_id":0,"name":0})
{ "scores" : { "English" : 69, "Math" : 86, "Computer" : 77 } }
>

```

修改 lisi 的 Math 成绩，改为 95：

```

> db.student.update({"name":"lisi"},{"$set":{"score.Math":95}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
>

```

java api 实验：

①

```

state=CONNECTED, ok=true, version=ServerVersion{versionList=[3, 2, 12]}, minWireVersion=0,
roundTripTimeNanos=568500}
uh 21, 2022 7:35:22 com.mongodb.diagnostics.logging.JULLogger log
  21: Opened connection [connectionId{localValue:2, serverValue:9}] to 192.168.10.100:27017
Document inserted successfully
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 2.375 s
[INFO] Finished at: 2022-11-21T19:35:22+08:00
[INFO] -----

Process finished with exit code 0

```

②

```

uh 21, 2022 7:33:44 com.mongodb.diagnostics.logging.JULLogger log
  21: No server chosen by ReadPreferenceServerSelector{readPreference=primary} from cluster des
connectionMode=SINGLE, all=[ServerDescription{address=192.168.10.100:27017, type=UNKNOWN, statu
out
uh 21, 2022 7:33:44 com.mongodb.diagnostics.logging.JULLogger log
  21: Opened connection [connectionId{localValue:1, serverValue:6}] to 192.168.10.100:27017
uh 21, 2022 7:33:44 com.mongodb.diagnostics.logging.JULLogger log
  21: Monitor thread successfully connected to server with description ServerDescription{addres
state=CONNECTED, ok=true, version=ServerVersion{versionList=[3, 2, 12]}, minWireVersion=0, maxR
roundTripTimeNanos=720600}
uh 21, 2022 7:33:44 com.mongodb.diagnostics.logging.JULLogger log
  21: Opened connection [connectionId{localValue:2, serverValue:7}] to 192.168.10.100:27017
{ "score" : { "English" : 45, "Math" : 89, "Computer" : 100 } }
{ "score" : { "English" : 45, "Math" : 89, "Computer" : 100 } }
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 2.407 s
[INFO] Finished at: 2022-11-21T19:33:44+08:00
[INFO] -----

Process finished with exit code 0

```

三、心得总结（写出自己在完成实验过程中遇到的问题、解决方法，以及体会、收获等）（10%）

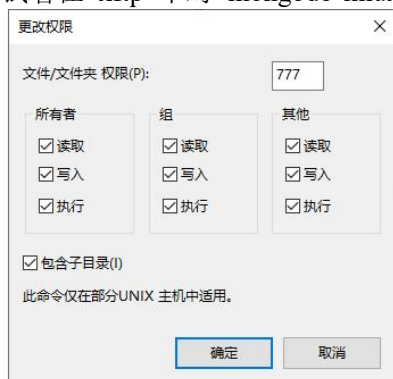
在做 redis 实验的时候，我第一遍直接用 xftp 上传助教的 config 文件，但是没有覆盖成功，还是用了原来的 conf 文件，导致起 redis 的时候第一遍没有起来。后来修改之后，遇到了报错的问题显示没有访问日志的权限 permission denied.

```
# Specify the log file name. Also the empty string can be used to force
# Redis to log on the standard output. Note that if you use standard
# output for logging but daemonize, logs will be sent to /dev/null
logfile /usr/local/bin/redis-log.log
```

助教讲实验的时候说这个实验经常遇到权限问题，但是我按照前面的实验步骤全部又执行了一遍赋权操作之后，依然没能解决。最后询问助教，助教帮我给 hadoop 用户增加了可读可写的权限之后问题得以解决：

```
[hadoop@namenode bin]$ sudo chmod 777 -R ./
[hadoop@namenode bin]$ ll
total 18904
drwxrwxrwx. 2 hadoop hadoop    24 Nov 21 14:47 bconfig
-rwxrwxrwx. 1 hadoop hadoop 4829584 Nov 21 14:42 redis-benchmark
lrwxrwxrwx. 1 hadoop hadoop    12 Nov 21 14:42 redis-check-aof -> redis-server
lrwxrwxrwx. 1 hadoop hadoop    12 Nov 21 14:42 redis-check-rdb -> redis-server
-rwxrwxrwx. 1 hadoop hadoop 5003880 Nov 21 14:42 redis-cli
lrwxrwxrwx. 1 hadoop hadoop    12 Nov 21 14:42 redis-sentinel -> redis-server
-rwxrwxrwx. 1 hadoop hadoop 9519000 Nov 21 14:42 redis-server
[hadoop@namenode bin]$ cd bconfig/
[hadoop@namenode bconfig]$ ls
redis.conf
[hadoop@namenode bconfig]$ ll
total 92
-rwxrwxrwx. 1 hadoop hadoop 93781 Nov 21 14:59 redis.conf
[hadoop@namenode bconfig]$ cd ..
[hadoop@namenode bin]$ ls
bconfig redis-benchmark redis-check-aof redis-check-rdb redis-cli redis-sentinel redis-server
[hadoop@namenode bin]$ redis-server bconfig/redis.conf
[hadoop@namenode bin]$ _
```

在启动 mongodb 的时候我也遇到了 permission denied 的问题，我猜测可能是由于我没有直接在虚拟机中使用修改了权限的文件进行解压导致的权限问题，但是我对于使用 linux 的 chown 和 chmod 指令还不是很熟悉，也不知道该对哪几个文件修改为怎样的权限。但我回忆之前助教给我解答的内容，我尝试着在 xftp 中对 mongodb-linux-x86_64-3.2.12/bin/mongo 这个文件修改权限为 777：



这样它就变成 hadoop 用户可读可写可执行的了。最终成功运行出结果。