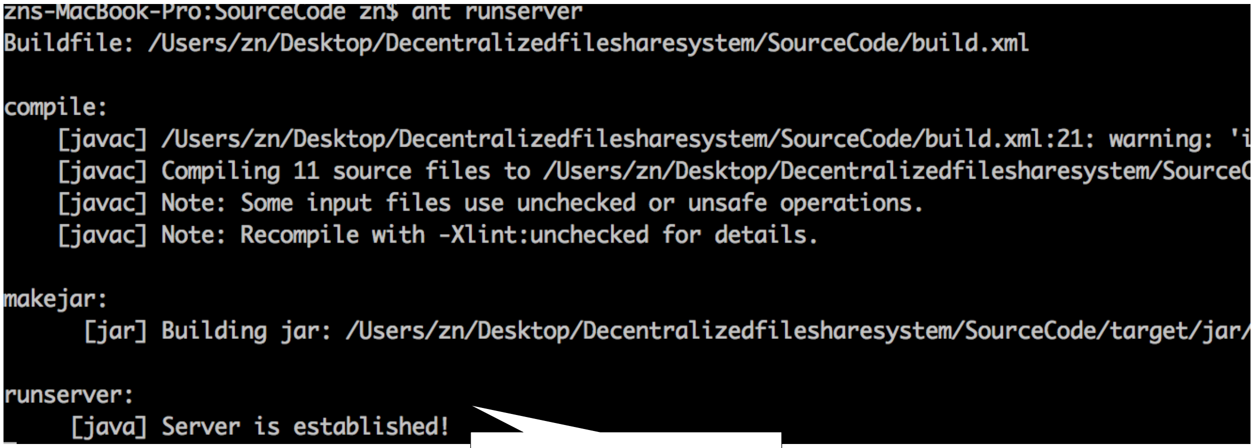
Output file

Name: Ning Zhang
CWID: A20336916
Department of Computer Science, Illinois Institute of Technology

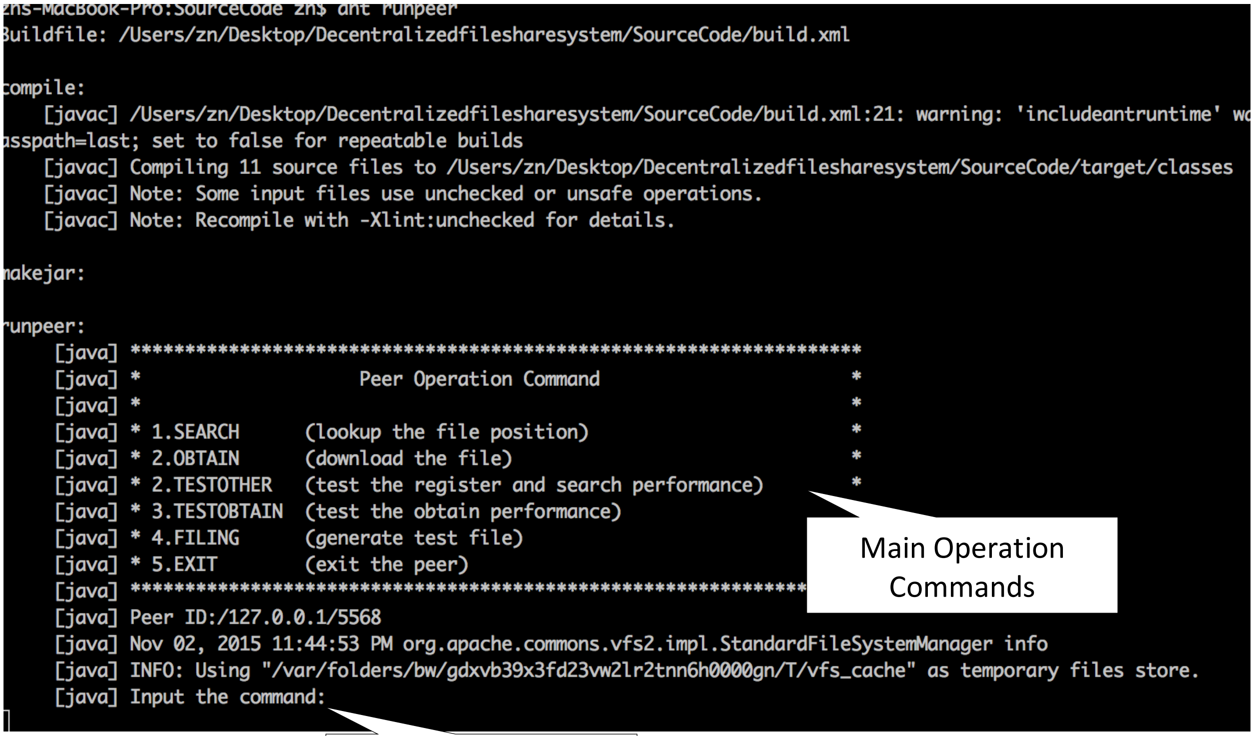
1.when the indexserver is running, it will output this results as following.



```
zns-MacBook-Pro:SourceCode zn$ ant runserver
Buildfile: /Users/zn/Desktop/Decentralizedfilesharesystem/SourceCode/build.xml

compile:
    [javac] /Users/zn/Desktop/Decentralizedfilesharesystem/SourceCode/build.xml:21: warning: 'i
    [javac] Compiling 11 source files to /Users/zn/Desktop/Decentralizedfilesharesystem/SourceC
    [javac] Note: Some input files use unchecked or unsafe operations.
    [javac] Note: Recompile with -Xlint:unchecked for details.

makejar:
      [jar] Building jar: /Users/zn/Desktop/Decentralizedfilesharesystem/SourceCode/target/jar/

runserver:
     [java] Server is established!
```

Running indexserver successfully

2.when the client is running, it will output this results as following.

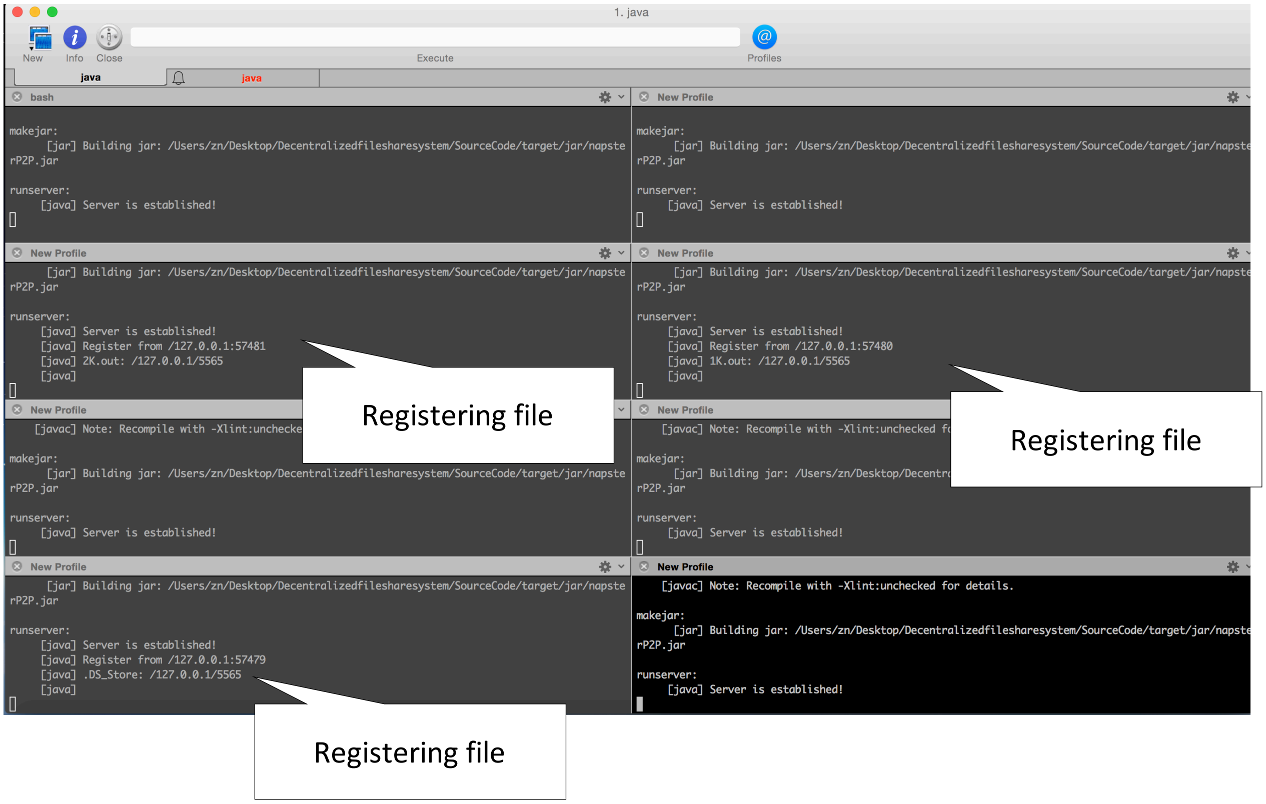


```
zns-MacBook-Pro:SourceCode zn$ ant runpeer
Buildfile: /Users/zn/Desktop/Decentralizedfilesharesystem/SourceCode/build.xml

compile:
    [javac] /Users/zn/Desktop/Decentralizedfilesharesystem/SourceCode/build.xml:21: warning: 'includeantruntime' wa
asspath=last; set to false for repeatable builds
    [javac] Compiling 11 source files to /Users/zn/Desktop/Decentralizedfilesharesystem/SourceCode/target/classes
    [javac] Note: Some input files use unchecked or unsafe operations.
    [javac] Note: Recompile with -Xlint:unchecked for details.

makejar:

runpeer:
     [java] *******************************************************************
     [java] *                    Peer Operation Command                       *
     [java] *                                                                 *
     [java] * 1.SEARCH      (lookup the file position)                        *
     [java] * 2.OBTAIN      (download the file)                               *
     [java] * 2.TESTOTHER   (test the register and search performance)        *
     [java] * 3.TESTOBTAIN  (test the obtain performance)
     [java] * 4.FILING      (generate test file)
     [java] * 5.EXIT        (exit the peer)
     [java] *************************************************************
     [java] Peer ID:/127.0.0.1/5568
     [java] Nov 02, 2015 11:44:53 PM org.apache.commons.vfs2.impl.StandardFileSystemManager info
     [java] INFO: Using "/var/folders/bw/gdxvb39x3fd23vw2lr2tnn6h0000gn/T/vfs_cache" as temporary files store.
     [java] Input the command:
```

Main Operation Commands

Input the command

3.when we copy several files into fileserver which is also the client, we can get the outputs from the indexservers as following.



```
makejar:
      [jar] Building jar: /Users/zn/Desktop/Decentralizedfilesharesystem/SourceCode/target/jar/napste
rP2P.jar

runserver:
     [java] Server is established!
```

```
makejar:
      [jar] Building jar: /Users/zn/Desktop/Decentralizedfilesharesystem/SourceCode/target/jar/napste
rP2P.jar

runserver:
     [java] Server is established!
```

```
      [jar] Building jar: /Users/zn/Desktop/Decentralizedfilesharesystem/SourceCode/target/jar/napste
rP2P.jar

runserver:
     [java] Server is established!
     [java] Register from /127.0.0.1:57481
     [java] 2K.out: /127.0.0.1/5565
     [java]
```

```
      [jar] Building jar: /Users/zn/Desktop/Decentralizedfilesharesystem/SourceCode/target/jar/napste
rP2P.jar

runserver:
     [java] Server is established!
     [java] Register from /127.0.0.1:57480
     [java] 1K.out: /127.0.0.1/5565
     [java]
```

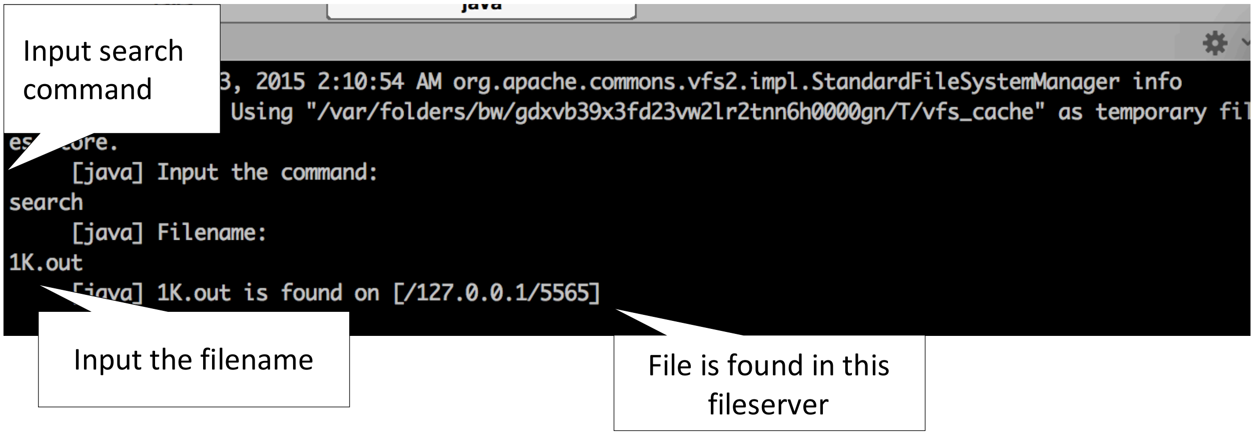
```
    [javac] Note: Recompile with -Xlint:unchecke

makejar:
      [jar] Building jar: /Users/zn/Desktop/Decentralizedfilesharesystem/SourceCode/target/jar/napste
rP2P.jar

runserver:
     [java] Server is established!
```

```
    [javac] Note: Recompile with -Xlint:unchecked fo

makejar:
      [jar] Building jar: /Users/zn/Desktop/Decentra
rP2P.jar

runserver:
     [java] Server is established!
```

```
      [jar] Building jar: /Users/zn/Desktop/Decentralizedfilesharesystem/SourceCode/target/jar/napste
rP2P.jar

runserver:
     [java] Server is established!
     [java] Register from /127.0.0.1:57479
     [java] .DS_Store: /127.0.0.1/5565
     [java]
```

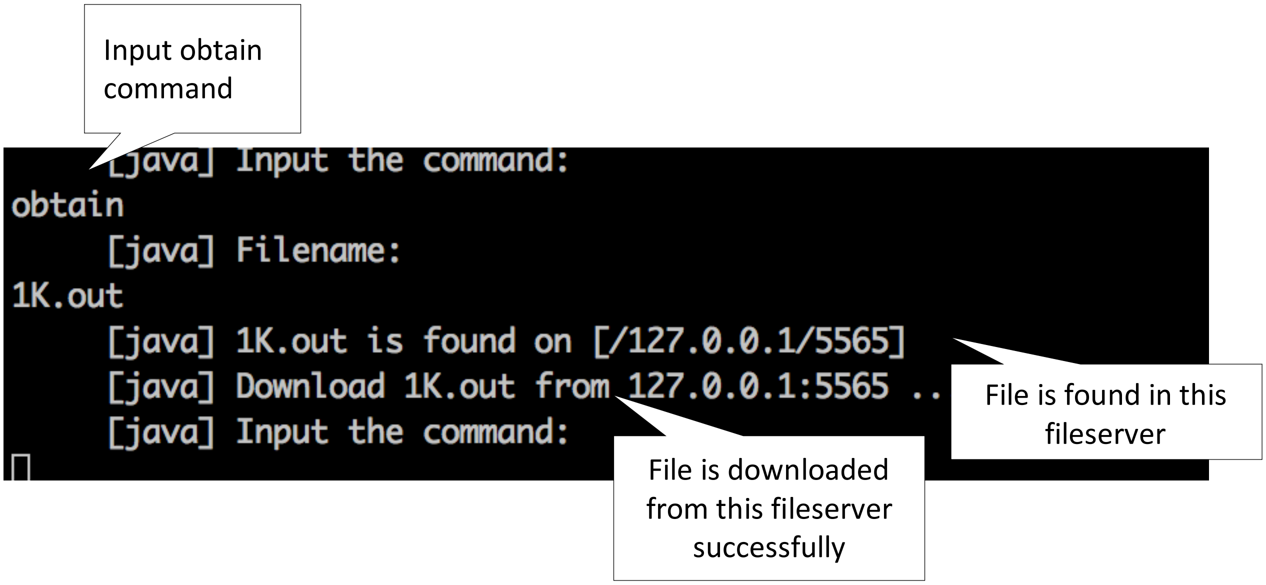
```
    [javac] Note: Recompile with -Xlint:unchecked for details.

makejar:
      [jar] Building jar: /Users/zn/Desktop/Decentralizedfilesharesystem/SourceCode/target/jar/napste
rP2P.jar

runserver:
     [java] Server is established!
```

4.when it executes the search command, we can get outputs in the client as following.



```
3, 2015 2:10:54 AM org.apache.commons.vfs2.impl.StandardFileSystemManager info
 Using "/var/folders/bw/gdxvb39x3fd23vw2lr2tnn6h0000gn/T/vfs_cache" as temporary fi
es   ore.
     [java] Input the command:
search
     [java] Filename:
1K.out
     [java] 1K.out is found on [/127.0.0.1/5565]
```

5.when it executes the obtain command, we can get outputs in the client as following.



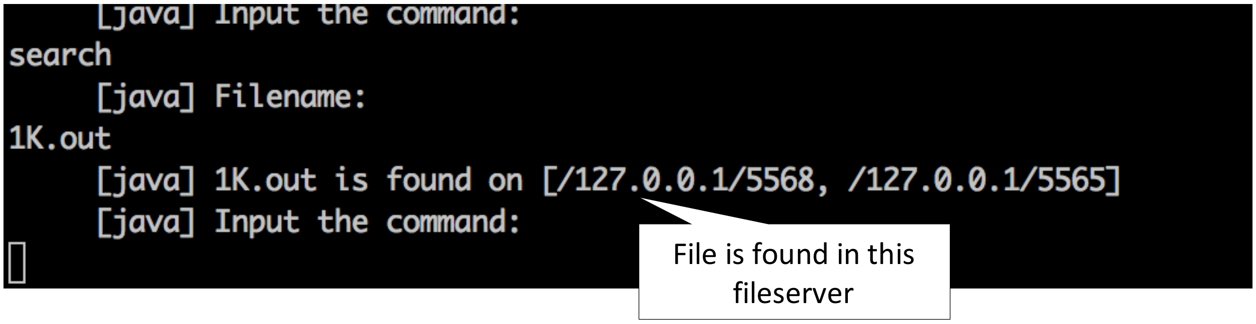
Input obtain command

```
     [java] Input the command:
obtain
     [java] Filename:
1K.out
     [java] 1K.out is found on [/127.0.0.1/5565]
     [java] Download 1K.out from 127.0.0.1:5565 ..
     [java] Input the command:
```

File is found in this fileserver

File is downloaded from this fileserver successfully

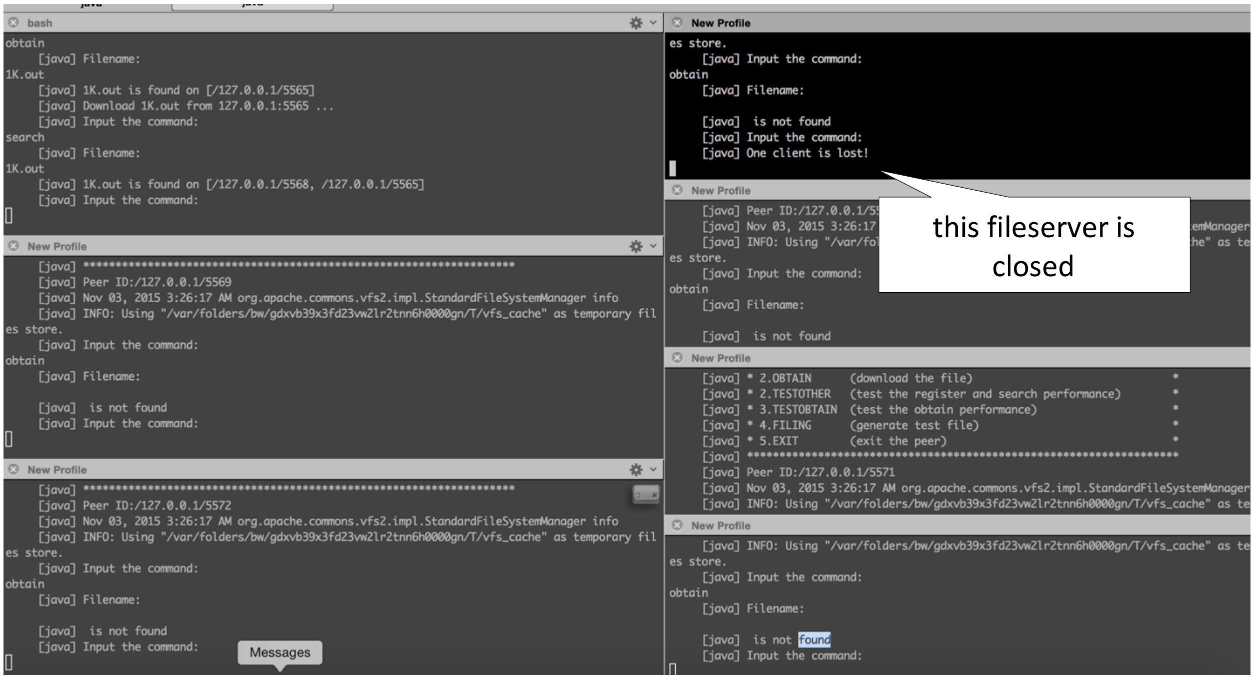
6.In order to prove the correctness of the obtain command, we use the search command to lookup if the file is downloaded into this fileserver executing the obtain command and which ID is /127.0.0.1/5568. Then we get the outputs in this fileserver as following.



```
     [java] Input the command:
search
     [java] Filename:
1K.out
     [java] 1K.out is found on [/127.0.0.1/5568, /127.0.0.1/5565]
     [java] Input the command:
```

File is found in this fileserver

7. In order to prove that the file replication is working, we just close a fileserver whose ID is /127.0.0.1/5565. And if we obtain the file 2K.out successfully, it proves that the file replication is working. Because the file 2K.out is registered from the fileserver /127.0.0.1/5565, which means this file is not in other fileservers except the fileserver /127.0.0.1/5565. When we close this fileserver, we can get the outputs.



```
obtain
     [java] Filename:
1K.out
     [java] 1K.out is found on [/127.0.0.1/5565]
     [java] Download 1K.out from 127.0.0.1:5565 ...
     [java] Input the command:
search
     [java] Filename:
1K.out
     [java] 1K.out is found on [/127.0.0.1/5568, /127.0.0.1/5565]
     [java] Input the command:
```

```
     [java] ******************************************************************
     [java] Peer ID:/127.0.0.1/5569
     [java] Nov 03, 2015 3:26:17 AM org.apache.commons.vfs2.impl.StandardFileSystemManager info
     [java] INFO: Using "/var/folders/bw/gdxvb39x3fd23vw2lr2tnn6h0000gn/T/vfs_cache" as temporary fil
es store.
     [java] Input the command:
obtain
     [java] Filename:

     [java]  is not found
     [java] Input the command:
```

```
     [java] ******************************************************************
     [java] Peer ID:/127.0.0.1/5572
     [java] Nov 03, 2015 3:26:17 AM org.apache.commons.vfs2.impl.StandardFileSystemManager info
     [java] INFO: Using "/var/folders/bw/gdxvb39x3fd23vw2lr2tnn6h0000gn/T/vfs_cache" as temporary fil
es store.
     [java] Input the command:
obtain
     [java] Filename:

     [java]  is not found
     [java] Input the command:
```

```
es store.
     [java] Input the command:
obtain
     [java] Filename:

     [java]  is not found
     [java] Input the command:
     [java] One client is lost!
```

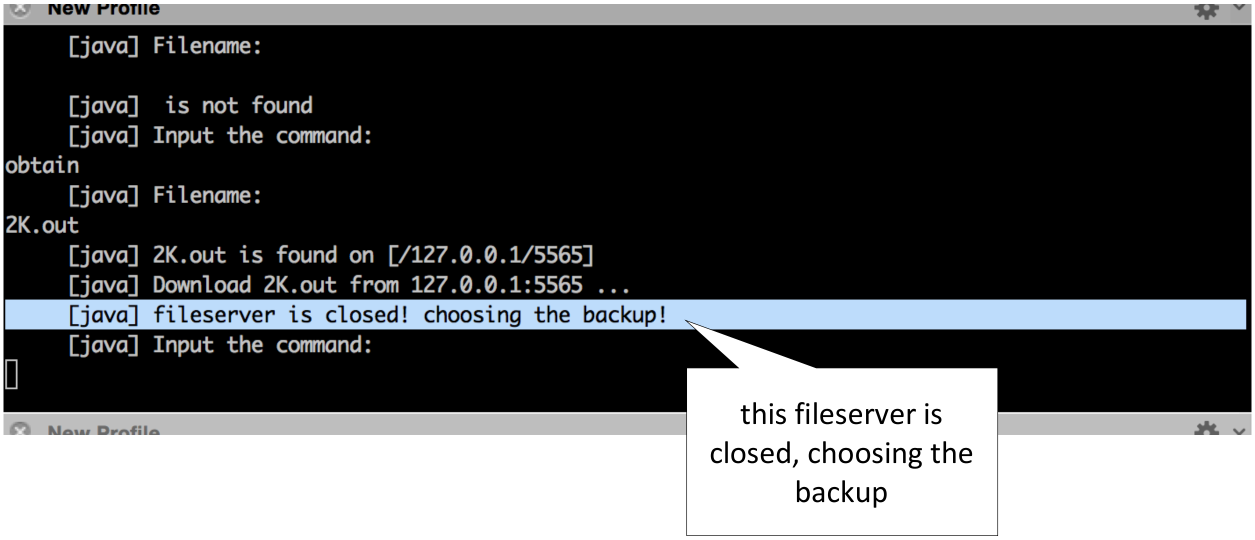
```
     [java] Peer ID:/127.0.0.1/55
     [java] Nov 03, 2015 3:26:17                                              emManager
     [java] INFO: Using "/var/fo                                              he" as te
es store.
     [java] Input the command:
obtain
     [java] Filename:

     [java]  is not found
```

```
     [java] * 2.OBTAIN      (download the file)                               *
     [java] * 2.TESTOTHER   (test the register and search performance)        *
     [java] * 3.TESTOBTAIN  (test the obtain performance)                     *
     [java] * 4.FILING      (generate test file)                              *
     [java] * 5.EXIT        (exit the peer)                                   *
     [java] ******************************************************************
     [java] Peer ID:/127.0.0.1/5571
     [java] Nov 03, 2015 3:26:17 AM org.apache.commons.vfs2.impl.StandardFileSystemManager
     [java] INFO: Using "/var/folders/bw/gdxvb39x3fd23vw2lr2tnn6h0000gn/T/vfs_cache" as te
```

```
     [java] INFO: Using "/var/folders/bw/gdxvb39x3fd23vw2lr2tnn6h0000gn/T/vfs_cache" as te
es store.
     [java] Input the command:
obtain
     [java] Filename:

     [java]  is not found
     [java] Input the command:
```

this fileserver is closed

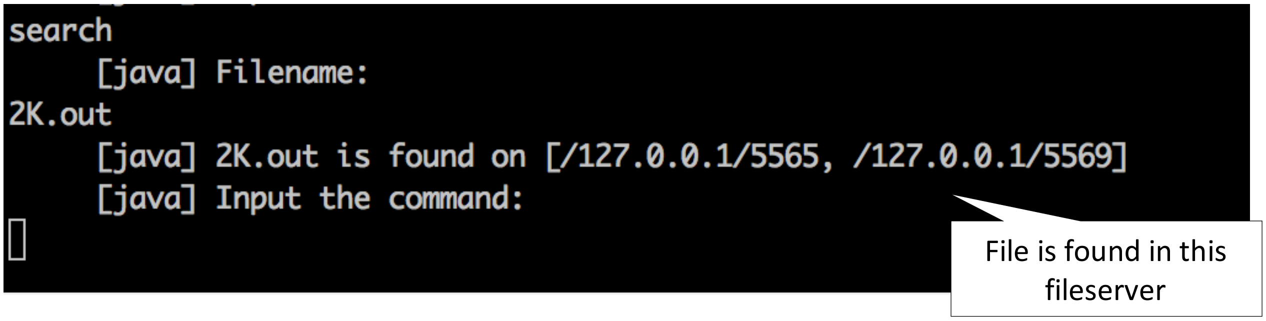
Then we obtain the file 2K.out in any client. We can get the outputs as following.



```
     [java] Filename:

     [java]  is not found
     [java] Input the command:
obtain
     [java] Filename:
2K.out
     [java] 2K.out is found on [/127.0.0.1/5565]
     [java] Download 2K.out from 127.0.0.1:5565 ...
     [java] fileserver is closed! choosing the backup!
     [java] Input the command:
```

this fileserver is closed, choosing the backup

Then we search this file to prove that it obtains this file successfully.



```
search
     [java] Filename:
2K.out
     [java] 2K.out is found on [/127.0.0.1/5565, /127.0.0.1/5569]
     [java] Input the command:
```

File is found in this fileserver

this file is in the fileserver /127.0.0.1/5569 which obtains this file. It proves that the fileserver /127.0.0.1/5569 obtains this file successfully. And it also proves that our replication for files works well.

8.In terms of the test command and filing command, I will present the outputs in the performance document.