1 Check for valid Number as Input

```
Server IP is(Local): 192.168.1.204
ATM run
valid input
decrypted String at Server: start Handshake
”account”:”roy”,”initial balance”:1000.11
ATM closed after successful action
finished -n message
ATM run
valid input
decrypted String at Server:
start Handshake
”Handshake achieved successfully”

”account”:”roy”,”deposit”:500.02
”account”:”roy”,”initial balance”:500.02
ATM closed after successful action
```

2 Check for invalid Number as Input (Amount = 001000.11) and Only Account creation parameter is passed.

Test Results : Pass. CardFile is also not created

```
created
Server IP is(Local): 192.168.1.204
ATM run
SimpleSec: closes connection
Exception in thread ”Thread-1” java.lang.NullPointerException
at SimpleSec.closeConnection(SimpleSec.java:49)
at Bank1.run(Bank.java:43)
```

3 Check by passing amount and balance as String

Balances and currency amounts are specified as a number indicating a whole amount and a fractional input separated by a period.

Test Results : Pass. CardFile is also not created

```
created
Server IP is(Local): 192.168.1.204
ATM run
SimpleSec: closes connection
Exception in thread ”Thread-1” java.lang.NullPointerException
at SimpleSec.closeConnection(SimpleSec.java:49)
at Bank1.run(Bank.java:42)
```

4 Check by passing file name as . Or ..

Test Results : Pass. CardFile is also not created. Now resolved

```
created
Server IP is(Local): 192.168.1.204
ATM run
SimpleSec: closes connection
Exception in thread ”Thread-1” java.lang.NullPointerException
at SimpleSec.closeConnection(SimpleSec.java:49)
at Bank1.run(Bank.java:42)
```

5 Check for the length of the File names and Account names

File names are to be between 1 and 127 characters long.Account names are restricted to same characters as file names but they are inclusively between 1 and 122 characters of length, and ”.” and ”..” are valid account names.

Test Results : Pass. CardFile is also not created

```
created
Server IP is(Local): 192.168.1.204
ATM run
SimpleSec: closes connection
Exception in thread ”Thread-1” java.lang.NullPointerException
at SimpleSec.closeConnection(SimpleSec.java:49)
at Bank1.run(Bank.java:42)
```

6 Check IP Address by passing number less than 0 and greater than 255.

IP addresses are restricted to IPv4 32-bit addresses and are provided on the command line in dotted decimal notation, i.e., four numbers between 0 and 255 separated by periods.

Test Results : Pass. CardFile is also not created

```
created
Server IP is(Local): 192.168.1.204
ATM run
SimpleSec: closes connection
Exception in thread ”Thread-1” java.lang.NullPointerException
at SimpleSec.closeConnection(SimpleSec.java:49)
at Bank1.run(Bank.java:42)
```

7 Check Port numbers by passing a value not in between 1024 and 65535 inclusively and also by port numbers as string.

Ports are specified as numbers between 1024 and 65535 inclusively.

Test Results: Pass. CardFile is also not created

```
Exception in thread ”main” java.lang.NumberFormatException: For input
string: ”3000343545”
at java.base/java.lang.NumberFormatException.forInputString(NumberForma
tException.java:65)
at java.base/java.lang.Integer.parseInt(Integer.java:652)
at java.base/java.lang.Integer.valueOf(Integer.java:983)
at Bank.handleArguments(Bank.java:126)
at Bank.main(Bank.java:47)
```

8 A timeout should occur if the other program does not respond within 10 seconds.

If the atm observes the timeout, it should exit with return code 63, while if the bank observes it, it should print ”protocol error” to stdout (followed by a newline)and rollback the current transaction.

Test Results: Pass. After Account creation since the thread is made to sleep for more than 10 sec the system exits and does not perform deposit

```
created
Server IP is(Local): 192.168.1.204
Input at ATM is valid
”Handshake is achieved successfully”
”account”:”musterman”,”initial balance”:1000.11
”account”:”musterman”,”initial balance”:1000.11
finished -n message
ATM closed after successful action
```

9 Check if Server has started by passing -p -s

On startup, bank will generate a auth file with the specified name. Existing auth files are not valid for new runs of bank if the specified file already exists, bank should exit with return code 255. Once the auth file is written completely, bank prints ”created” (followed by a newline) to stdout. bank will not change the auth file once ”created” has been printed.

Test Results : Pass

```
Exception in thread ”Thread-0” java.lang.NullPointerException
at Bank1.run(Bank.java:44)
```

Check if Server has started by passing –s -p

Test Results : Pass

```
created
Bank: Server IO exeption
java.net.BindException: Address already in use: NET Bind
at java.base/java.net.DualStackPlainSocketImpl.bind0(Native Method)
at java.base/java.net.DualStackPlainSocketImpl.socketBind(DualStackPlainS
ocketImpl.java:106)
at java.base/java.net.AbstractPlainSocketImpl.bind(AbstractPlainSocketImp
l.java:437)
at java.base/java.net.PlainSocketImpl.bind(PlainSocketImpl.java:174)
at java.base/java.net.ServerSocket.bind(ServerSocket.java:381)
at java.base/java.net.ServerSocket.¡init¿(ServerSocket.java:243)
at java.base/java.net.ServerSocket.¡init¿(ServerSocket.java:135)
at SimpleSecServer.¡init¿(SimpleSecServer.java:22)
at Bank.genAuthfile(Bank.java:77)
at Bank.main(Bank.java:50)
```

Issue resolved now. After resolving issue we get following:

```
created
Server IP is(Local): 192.168.1.204
```

10 Check for Account creation, Deposit and card file creation.

11 Check that an account is created only after successful 3-way handshake is established.

Test Results : Pass

```
created
Server IP is(Local): 192.168.1.204
Input at ATM is valid
”Handshake is achieved successfully”

Bank does Account creation
start Handshake
”Handshake is achieved successfully”

”account”:”musterman”,”initial balance”:1000.11
ATM closed after successful action

”account”:”musterman”,”initial balance”:1000.11
finished -n message
Input at ATM is valid

decrypted String at Server:
start Handshake
Bank does Deposit
”account”:”musterman”,”deposit”:500.02

”account”:”musterman”,”deposit”:500.02
ATM closed after successful action
finished -d message
```

12 Check that server should still be running inspite of passing invalid client parameters.Invalid data from a client should not cause the server to exit and thereby deny access to other clients.

Test Results : Fail

```
created
Server IP is(Local): 192.168.1.204
```

13 Create an new account ’bob’ with balance 1000.00

Test Results : Pass

```
created
Server IP is(Local): 192.168.1.204
Input at ATM is valid
”Handshake is achieved successfully”

”account”:”bob”,”initial balance”:1000
”account”:”bob”,”initial balance”:1000
ATM closed after successful action
finished -n message
```

14 Create Account and perform deposit.

Test Results : Pass

```
created
Server IP is(Local): 192.168.1.204
Input at ATM is valid
”Handshake is achieved successfully”
”account”:”musterman”,”initial balance”:1000.11
”account”:”musterman”,”initial balance”:1000.11
```

```
ATM closed after successful action
finished -n message
Input at ATM is valid
decrypted String at Server:
”Handshake is achieved successfully”

”account”:”musterman”,”deposit”:500.02
”account”:”musterman”,”deposit”:500.02
ATM closed after successful action
finished -d message
```

15 First created account then deposit and then Checking Balance

Test Result : Pass

```
created
Server IP is(Local): 192.168.1.204
Input at ATM is valid
”Handshake is achieved successfully”

”account”:”musterman”,”initial balance”:1000.11
”account”:”musterman”,”initial balance”:1000.11
ATM closed after successful action
finished -n message
Input at ATM is valid
decrypted String at Server:
”Handshake is achieved successfully”

”account”:”musterman”,”deposit”:500.02
”account”:”musterman”,”deposit”:500.02
ATM closed after successful action
finished -d message

decrypted String at Server:
”account”:”musterman”,”balance”:1500.13
Handshake Log : Object Hash: 1596609309 Log : B,1500.13
```

```
Time out occurs because the wait time was more than 10 sec
java.net.SocketTimeoutException: Read timed out
at java.base/java.net.SocketInputStream.socketRead0(Native Method)
at java.base/java.net.SocketInputStream.socketRead(SocketInputStream.java:116)
at java.base/java.net.SocketInputStream.read(SocketInputStream.java:171)
at java.base/java.net.SocketInputStream.read(SocketInputStream.java:141)
at java.base/sun.nio.cs.StreamDecoder.readBytes(StreamDecoder.java:284)
at java.base/sun.nio.cs.StreamDecoder.implRead(StreamDecoder.java:326)
at java.base/sun.nio.cs.StreamDecoder.read(StreamDecoder.java:178)
at java.base/java.io.InputStreamReader.read(InputStreamReader.java:185)
at java.base/java.io.BufferedReader.fill(BufferedReader.java:161)
at java.base/java.io.BufferedReader.readLine(BufferedReader.java:326)
at java.base/java.io.BufferedReader.readLine(BufferedReader.java:392)
at SimpleSecClient.listen(SimpleSecClient.java:82)
at ATM.handleArguments(ATM.java:117)
at ATM.main(ATM.java:93)
SimpleSec: closes connection
```

16 Attempt to withdraw 2000, which fails since musterman does not have a sufficient balance.

Test Results: Pass

```
created
Server IP is(Local): 192.168.1.204
Input at ATM is valid
”Handshake is achieved successfully”

”account”:”musterman”,”initial balance”:1000.11
”account”:”musterman”,”initial balance”:1000.11
ATM closed after successful action
finished -n message
SimpleSec: closes connection
SimpleSec: closes connection
```

17 Attempt to create another account ’bob’, which fails since the account ’bob’ already exists.

Test Results-Pass. System Terminates

18 First created a new account with name musterman then did deposit and then Checked Balance. Then created a new account with name muster.

Test Results-Pass for 1st Account creation. But Fails during second Account creation.The authentication file gets deleted and it says card file already exist.

```
created
Server IP is(Local): 192.168.1.204
Input at ATM is valid
”Handshake is achieved successfully”
”account”:”musterman”,”initial balance”:1000.11
”account”:”musterman”,”initial balance”:1000.11
finished -n message
ATM closed after successful action
Input at ATM is valid
decrypted String at Server:
”Handshake is achieved successfully”

”account”:”musterman”,”deposit”:500.02
ATM closed after successful action
”account”:”musterman”,”deposit”:500.02
finished -d message
Input at ATM is valid
java.io.IOException: cardfile does already exist
at ATM.handleNmessage(ATM.java:143)
at ATM.handleArguments(ATM.java:118)
at ATM.main(ATM.java:95)
at Test.main(Test.java:62)
SimpleSec: closes connection
Exception in thread ”Thread-0” java.lang.NullPointerException
at java.base/java.util.Base64Decoder.decode(Base64.java:549)
at AESCrpyter.decrypt(AESCrpyter.java:78)
at SimpleSecServer.handleMessage(SimpleSecServer.java:85)
at SimpleSecServer.mainLoop(SimpleSecServer.java:40)
at Bank.genAuthfile(Bank.java:80)
at Bank.main(Bank.java:51)
at Test1.run(Test.java:17)
SimpleSec: closes connection
SimpleSec: closes connection
SimpleSec: closes connection
```

19 Create a 2 new accounts alice and musterman with some balance. Musterman attempts to access alice’s balance with his card, which fails.

Test results: Pass

```
Created
Server IP is(Local): 192.168.1.204
Input at ATM is valid
”Handshake is achieved successfully”
”account”:”musterman”,”initial balance”:1000.11
”account”:”musterman”,”initial balance”:1000.11
finished -n message
ATM closed after successful action
Input at ATM is valid
”Handshake is achieved successfully”

”account”:”alice”,”initial balance”:500.02
```

”account”:”alice”,”initial balance”:500.02

```
ATM closed after successful action
Input at ATM is valid
ATM socket timed out
java.net.SocketTimeoutException: Read timed out
at java.base/java.net.SocketInputStream.socketRead0(Native Method)
at java.base/java.net.SocketInputStream.socketRead(SocketInputStream.java:116)
at java.base/java.net.SocketInputStream.read(SocketInputStream.java:171)
at java.base/java.net.SocketInputStream.read(SocketInputStream.java:141)
at java.base/sun.nio.cs.StreamDecoder.readBytes(StreamDecoder.java:284)
at java.base/sun.nio.cs.StreamDecoder.implRead(StreamDecoder.java:326)
at java.base/sun.nio.cs.StreamDecoder.read(StreamDecoder.java:178)
at java.base/java.io.InputStreamReader.read(InputStreamReader.java:185)
at java.base/java.io.BufferedReader.fill(BufferedReader.java:161)
at java.base/java.io.BufferedReader.readLine(BufferedReader.java:326)
at java.base/java.io.BufferedReader.readLine(BufferedReader.java:392)
at SimpleSecClient.listen(SimpleSecClient.java:87)
at ATM.handleArguments(ATM.java:125)
at ATM.main(ATM.java:96)
at Test.main(Test.java:59)
SimpleSec: closes connection
SimpleSec: closes connection
SimpleSec: closes connection
SimpleSec: closes connection
```

20 Pass the parameters in any random way like for example –n 1500.00 –a alice

```
created
Server IP is(Local): 192.168.1.204
SimpleSec: closes connection
Exception in thread ”Thread-1” java.lang.NullPointerException
at SimpleSec.closeConnection(SimpleSec.java:49)
at Bank1.run(Bank.java:43)
```

21 Deposit or withdraw negative amounts.

Test Results : Pass

```
created
Server IP is(Local): 192.168.1.204
SimpleSec: closes connection
Exception in thread ”Thread-1” java.lang.NullPointerException
at SimpleSec.closeConnection(SimpleSec.java:49)
at Bank1.run(Bank.java:43)
```

22 Check if the old card files are deleted before the ATM starts again.

Test Results : Pass

23 If I run bank and give command for only Account creation.

Test Results Fail.Issue resolved now

```
created
Server IP is(Local): 192.168.1.204
ATM run
SimpleSec: closes connection
Exception in thread ”Thread-1” java.lang.NullPointerException
at SimpleSec.closeConnection(SimpleSec.java:49)
at Bank1.run(Bank.java:43)
```

24 Check if handshake is successful before creating account.

Test Results : Pass

```
created
Server IP is(Local): 192.168.1.204
Input at ATM is valid
”Handshake is achieved successfully”
”account”:”account2”,”initial balance”:1000.11
ATM closed after successful action
”account”:”account2”,”initial balance”:1000.11
finished -n message
SimpleSec: closes connection
SimpleSec: closes connection
```

25 Reran the server again just to check if all the .card files are deleted of the 2 users created.

Test Results: Fail. Only 1 user file is deleted but not for another user