**Assignment 2 - Socket programming**

**214101001 Abhijeet Padhy**

**214101002 Abhishek Pratap Singh**

**214101003 Aditya Kumar Sakre**

The project here by implements a simple Banking System. There are two C programs Bank Server and Client. To compile and execute the following commands are used:

gcc bank\_server.c -o server

./server <server\_port\_no>

gcc bank\_client.c -o client

./client <server\_ip\_address> <server\_port\_address>

Initially, the client will connect to the bank server using the server’s IP address and TCP port already known to the client. After successful connection, the server sends the banking dashboard to the client and asks for reply as shown:

------------------------- Banking Dashboard-------------------------

Available commands

------------------

1. LOGIN

2. QUIT

Enter the option number:

To continue to login, send 1 and the username and password is then asked for.

Enter the option number: 1

------------------------- User Authentication -------------------------

username: bank001

password: user@001

Authentication Successfull!

-------------------------Customer Panel-------------------------

Available commands

------------------

1. View Balance

2. Print Mini Statement

3. Log Out

Enter the option number:

There are three kinds of clients: Customer, Admin and Police. According to the kind of client, the appropriate panel menu is shown. The above is the panel for Customer and the below are the panels for Admin and Police:

-------------------------Admin Panel-------------------------

Available commands

------------------

1. Credit

2. Debit

3. QUIT

Enter the option number:

-------------------------Police Panel-------------------------

Available commands

------------------

1. Show Balance of All Customers

2. Print Mini Statement

3. Log Out

Enter the option number:

# Customer:

The customer is allowed two operations:

1. Check available balance

Enter the option number: 1

----------------------------------

Available Balance: 6000

----------------------------------

2. Print statement

------------------------- MINI STATEMENT -------------------------

|----------------------------------------------

|User: bank001

|----------------------------------------------

|Fri Apr 1 15:01:55 2022 CREDIT 2000

|Fri Apr 2 15:02:55 2022 DEBIT 1000

|Fri Apr 3 15:03:55 2022 CREDIT 5000

|Fri Apr 3 15:04:55 2022 CREDIT 4000

|Fri Apr 4 15:05:55 2022 DEBIT 2000

|Fri Apr 5 15:06:55 2022 DEBIT 2000

|----------------------------------------------

|Available Balance: 6000

|----------------------------------------------

# Admin

The Admin is allowed two operations:

1. Credit

Enter the option number: 1

Enter username: bank001

Enter the amount to be credited: 300

Amount successfully credited!

----------------------------------

Available Balance: 6300

----------------------------------

2. Debit

Enter the option number: 2

Enter username: bank002

Enter the amount to be debited: 200

Amount has been successfully debited to your account!

----------------------------------

Available Balance: 719

----------------------------------

The c**ustomer account balance underflow** is also taken care of:

Enter the option number: 2

Enter username: bank002

Enter the amount to be debited: 1000

**Transaction failed due to insufficient balance**

----------------------------------

Available Balance: 719

----------------------------------

# Police:

1. Check balance of all customers:

Enter the option number: 1

Available Balance of all the customers:

---------------------------------------

bank001 : 6000

bank002 : 6000

bank003 : 6000

bank004 : 6000

bank005 : 6000

bank006 : 6000

bank007 : 6000

bank008 : 6000

bank009 : 6000

bank010 : 6000

2. Check Mini Statement of a particular customer:

Enter the option number: 2

------------------------- MINI STATEMENT -------------------------

|----------------------------------------------

|User: bank001

|----------------------------------------------

|Fri Apr 1 15:01:55 2022 CREDIT 2000

|Fri Apr 2 15:02:55 2022 DEBIT 1000

|Fri Apr 3 15:03:55 2022 CREDIT 5000

|Fri Apr 3 15:04:55 2022 CREDIT 4000

|Fri Apr 4 15:05:55 2022 DEBIT 2000

|Fri Apr 5 15:06:55 2022 DEBIT 2000

|----------------------------------------------

|Available Balance: 6000

|----------------------------------------------

# Log File Entry Format

The login\_file.txt in database folder contains the login credentials which are as such:

bank001,user@001,C

bank002,user@002,C

bank003,user@003,C

bank004,user@004,C

bank005,user@005,C

bank006,user@006,C

bank007,user@007,C

bank008,user@008,C

bank009,user@009,C

bank010,user@010,C

admin01,admin@01,A

police0,police@0,P

# Customer Account files entry format

Fri Apr 1 15:01:55 2022,CREDIT,2000

Fri Apr 2 15:02:55 2022,DEBIT,1000

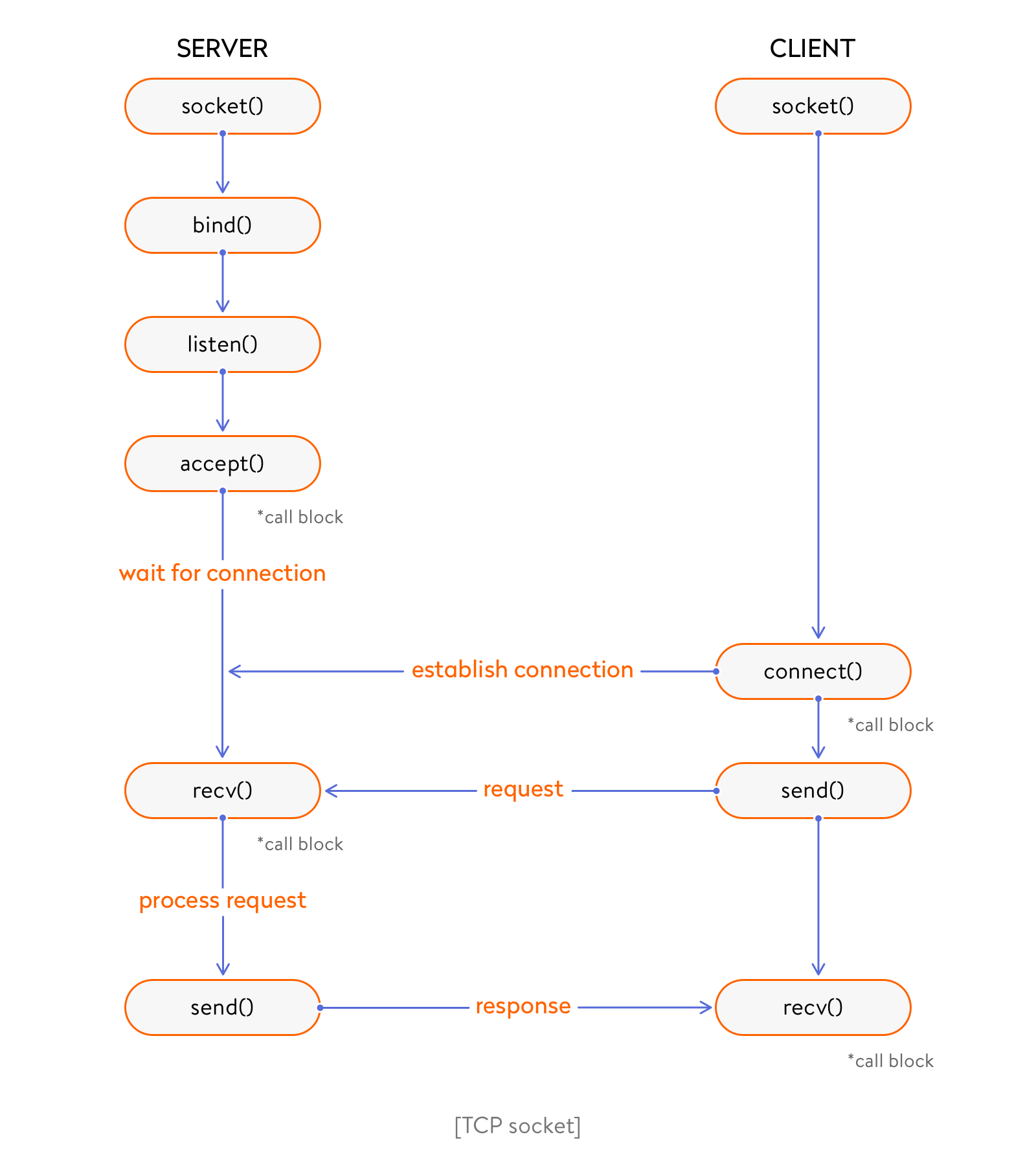
Fri Apr 3 15:03:55 2022,CREDIT,5000

Fri Apr 3 15:04:55 2022,CREDIT,4000

Fri Apr 4 15:05:55 2022,DEBIT,2000

Fri Apr 5 15:06:55 2022,DEBIT,2000

Sat Apr 9 11:47:20 2022,CREDIT,300

**TCP Socket Programming Model**

Functions used by both **Server** and **Client**s

**socket()**: Creates TCP socket for communication with connection-based byte streams (SOCK\_STREAM).

**read**() (***recv()*** can also be used): Read data from the data received from the other end of the socket connection.

**write()** (***send()*** can also be used): Write data to be sent to other end of the socket connection.

Server-side functions:

**bind()**: Assigns the address to the socket referred by the file descriptor and created by socket().

**listen()**: Set socket to the listening mode that will be used to accept incoming connection requests.

**accept()**: System call is used with connection-based socket. It extracts the first pending connection request on the listening socket.

Client-side function:

**connect()**: System call to connect the socket referred by the file descriptor, with address provided and protocol family.