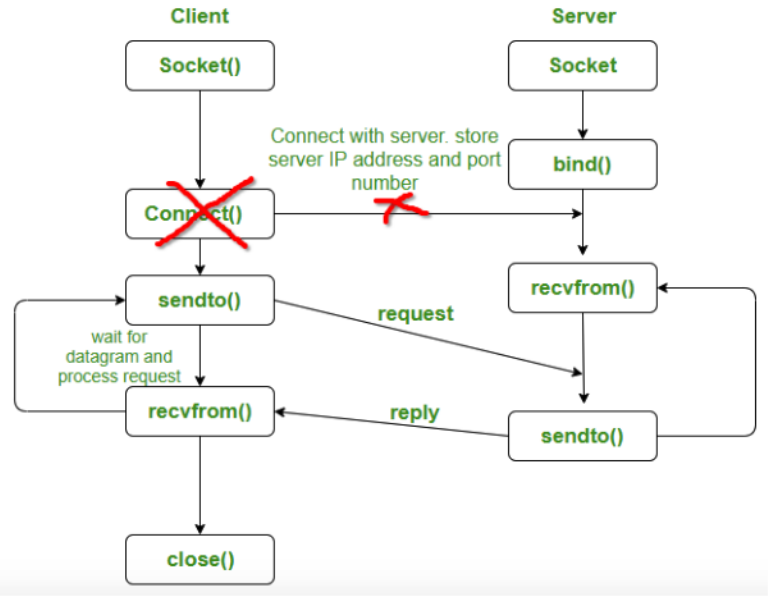
CISC 4615 — Lab 2

In the Lab 1, we learned how to programming with sockets and TCP. In the lab 2, we will UPD as our protocol and integrate CRC into the tranmissions.Different from TCP, UDP is a connection less protocol. There is no connection is established between client and server.

UDP Protocol Workflow

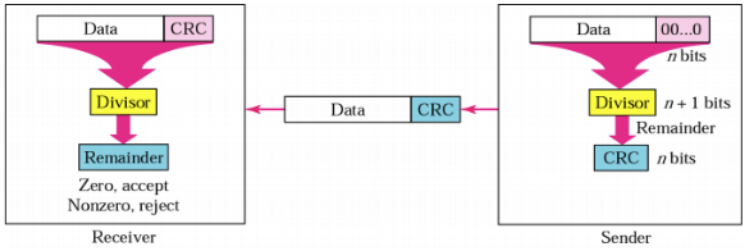


In UDP, the client does not form a **connection** with the server like in TCP and instead, it just sends a datagram. Similarly, the server need not to **accept** a connection and just waits for datagrams to arrive. We can call a function called connect() in UDP but it does not result anything like it does in TCP. There is no **3 way handshake**(we will study this topic later). It just checks for any immediate errors and store the peer's IP address and port number. connect() is storing peers address so no need to pass server address and server address length arguments in sendto().

CRC Generator and Checker

Cyclic Redundancy Check (CRC) is an error detection mechanism in which a special number is appended to a block of data in order to detect any changes introduced during storage (or transmission).

Transmitted messages are divided into predetermined lengths that are divided by a fixed **divisor**. According to the calculation, the **remainder** number is appended onto and sent with the



message. When the message is received, the receiver recalculates the remainder and compares it to the transmitted remainder. If the numbers do not match, an error is detected.

Study the sample codes

Please study the sample code to understand the code style and logic of the samples.

`https://github.com/yingmao/CISC4615-Lab2-Base`

1. : In sample 1, client sends a message to server by using UDP protocol. Client will break out of the loop by typing "bye".
2. : In sample 2, client inputs a message, the program will calculate the CRC32 code for this message.
3. : In sample 3, client accepts a message from the users, and calculates a CRC32 code for this messgae. Then, it sends the message along with the CRC32 code to the server.

After compiling the samples (c++), you can run the samples as shown below (using **python2.7**)

Sample 1:

- CPP: ./server 9000(port number) and ./client 127.0.0.1 9000(port number)
- Python: python2.7 client.py 127.0.0.1 9000 and python server.py 9000

Sample 2:

- CPP: ./crc
- Python: python2.7 crc.py

Sample 3:

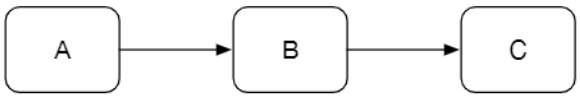
- CPP: ./server 9000(port number) and ./client 127.0.0.1 9000(port number)
- Python: python2.7 client.py 127.0.0.1 9000 and python server.py 9000

I have added a python 3 version of the samples.

Lab 1 Assignment: Part 1 and 2

Lab 1 consists of the following two parts.

1. Base on sample 1, create a "forwarder" for your program as shown below.



   - A sends a message, B receives it and forwards it to C, C receives it and print it out.
   - They all use UDP protocol.

2. Based on sample 3, integrate CRC into your 3-node program.
   - A sends a message to B along with the CRC32 code.
   - B receives this message and CRC32 code.
   - B follows a 40% probability to change the message.
   - B sends the message along with the original CRC32 code to C.
   - C receives the message and CRC32 code and check whether it is correct or not.

Grading Rubric

We will try to complete the part 1 in class and you have two weeks to complete the whole lab. In addition, you should prepare a detailed report with screenshots of your results about what you have learned and what are the challenges.

(70%) Part 1;
(20%) Part 2;
(5%) Report;
(5%) Submission format;

Submission

On Thursday, Feb 28th, each group has to demonstrate the lab with part 1 to me in class.

On Thursday, March 7th, each group has to demonstrate the lab with part 2 to me in class.

By the end of March 7th, each group has to email me with a subject line **CISC4615 Lab2 Submission** and a zip file that contains two folders,

- Lab2Part1: your programs of A.py (or A.cpp), B.py (or B.cpp) and C.py (or C.pp) as well as your report.

- Lab2Part2: your programs of A.py (or A.cpp), B.py (or B.cpp) and C.py (or C.pp) as well as your report.

If you submit both part 1 and 2 by Feb 28th, you will earn 5% bonus.