Indian Institute of Technology, Mandi
Feb-Jun 2022
CS307 - System Practicum
Assignment 4

Course Instructor: Aditya Nigam
Deadline: 8th May 2022 EOD

Instructions

- Plagiarism is strictly prohibited. In case of violation, a zero will be awarded for this assignment as a warning and a quick F grade if repeated later.
- The demo must contain the walk-through of all functionalities by running them as well as a brief walk-through of code.
- In the moodle students need to submit 2 things:
    - A zip file containing code,report file which provides the report of the assignment, and readme file for compiling the code.
- Use tcp connections in each case of this assignment. Use socket() system calls for creating tcp connections.
- Each application should have proper termination criteria (CTRL-C should not be the only way to make it stop). Don't forget to close the ports before terminating the application.
- Contact Abhijeet Manhas or Daksh Thapar for any queries.

Question 1 (Exploring IP addresses):

A. Find out the IP address of your laptop using ifconfig. Now go to ip2location.com and check ip there. Do this on the laptop of each team member. Explain your observations.
B. Use traceroute on a server outside India and find out location of each intermediate hop using ip2location or similar services.
C. Create a new user named `remote` on your machine. Install `jupyter notebook` for that user. Run jupyter notebook on port `1234`. Forward this (remote's 1234) port to port `4321` of your team-mate's machine so that your teammate can listen and run a jupyter notebook on their browser on `localhost:4321` using the hardware of your machine. Your team mate should install the python package `platform` and show the output of `platform.uname()` on the jupyter notebook after importing the above-mentioned package.

Question 2 (Client-Server File transfer):

Design a file transfer application based on client-server architecture which supports multiple clients simultaneously. Implement it over one PC utilizing two or more VMs (Virtual Machines). Let one VM be the server. This VM will have all the files that can be transferred to clients. Other VM's will act as client and connect with server and request files.

- **Functionalities:**

- Client sends file name to the server which checks its local disk for the file. If found it will send the file to the requesting client.
- Clients can ask the server for its usage details (list of files client has downloaded so far, size of data transferred etc). Usage details are client specific.
- Different kinds of errors/messages should be reported to the client. Eg: If a file is transferred successfully then print the same on the console of the client. Similarly, do the same for failed transfers, or files not found. Report such errors to the client.
- Think how you will differentiate between different types of messages (request for file, file not found,request for usage details etc).
- Application should have proper termination criteria (CTRL-C should not be the only way to make it stop). Don't forget to close the ports before terminating the application

- **Demo and Viva requirements**
  - Show that the server is handling multiple clients simultaneously.(You can request 2 large files simultaneously from two different clients).
  - Show that the application can support different types of files: Eg .png, .pdf, .txt etc.
  - Show that the application can support different file sizes: From a few KBs to a few GBs.
  - Connection should terminate properly after completion of file transfer.
  - Make it a reliable transfer(no packet loss and in-order packet reception).

Question 3 (Error Control):

Error Control is important as it ensures that data transferred over the network doesn't get corrupted. In this task, you have to perform error detection. Error detection is a challenging task because the receiver has to decide if received data is correct or not, without having a copy of the original message. **Burst errors** are errors in which 2 or more bits in the data unit have changed. **Cyclic Redundancy Check**(CRC) is an algorithm that is used to detect such errors.

- **Functionalities:**
  - In this task, use a single pair of server and client.
  - Instead of actual bits, generate strings consisting of '0' and '1'. Apply CRC algorithm on these strings.
  - Agree upon any divisor of length 5(eg 11001).
  - Generate a message of length 8*N in server. Apply CRC on it and send it to the client.
  - Apply the algorithm on each 8 length block of the original message.
  - On the client detect if there is any error or not