Comp 112: Networks
Tufts University, Spring 2020

Prof. Dogar
Trung Truong

Final Project Proposal

1. Proposed idea
   For the final project, I'm planning to implement a "Napster-like" peer-to-peer (p2p) file transferring protocol. The final deliverable will include a working index server that logs and keeps track of seeders and peers, and client nodes that can deliver content to one another in a p2p manner.

2. Invariances

   (a) The index server must know every registered file segment and machine addresses for such segment.

   (b) The index server must be able to handle if a peer is down.

   (c) The peer nodes must be in charged of file transferring.

3. Architecture
   The project will include the following components:

   - Peer nodes:

     (a) To initiate file transferring, a "seeder" peer will register a file name to be downloaded, the SHA-256 hash of each file segment, and the IP addresses of the machines containing such segment to the index server.

     (b) To request a file, a "leecher" peer will request the file name to the index server, and the index server will respond with the segment IDs and the IP address of the machines containing the requested file segments. The "leecher" peer can proceed to communicate with the "seeder" peers to request each file segment. Once the "leecher" received the correct file segment (which can be compared with the original hash), it will be qualified to be a "seeder" for such segment in other downloads in the future.

   - Index server:

     (a) The index server has a table of file names. For each file, it contains a list of segment IDs, which is the SHA-256 hash of the file segment, and a list of IP addresses of machine containing such segment. Once requested, the server will respond to a peer node with a message containing the list of segment ID's and a randomized machine for each segment, and the peer can proceed requesting each segment from respective machine.

(b) The index server also has a cache layer for faster response time. A logging module will also be implemented to record requests.

4. Tools

   This projected will be implemented in C++. Additional library will be included to perform SHA-256.

5. Timeline

   - 3/27 - Checkpoint 1: Working prototype of index server
   - 4/13 - Checkpoint 2: Working prototype of peer node
   - 4/27 - Final Project Due: Finished write-up
   - 4/29 - Final Project Demos/Presentations