CS 377 Final Project

SHARED CLOUD FILE SYSTEM

- Anurag Gumidelli

Key Features - from OS

- Thread Safety
- Cloud Managed File System
- Concurrency - (multi threaded)
- Resource locking
- Consistency

Usage

Makefile is already prepared

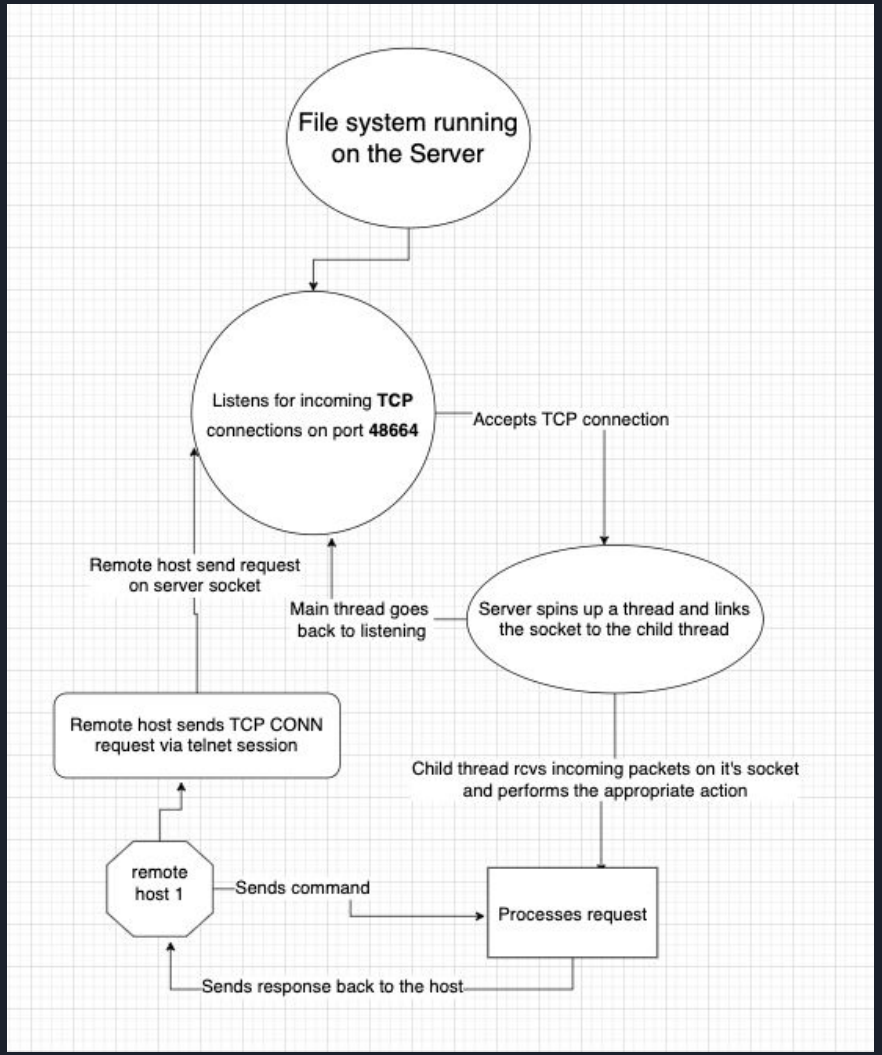
- User@PC: $ make → generates the binary files

  ./cloud_fs <diskPath>

  disk obj can be created by ` ./create_fs <diskPath> `

Showing flow for single remote host connection

SYSTEM DESIGN



Worker threads

```
48    struct argPointers {
49      void * socket_fd;
50      void * file_pointer;
51    };
52
53    void* worker(void *used_args);
```

TCP Server Socket

```
// keep listening to incoming connections and spin a new thread after each thread is created and give them a file object
while(true){
  if ((new_socket
    = accept(server_fd, (struct sockaddr*)&address,
        (socklen_t*)&addrlen))
    < 0) {
    perror("accept");
    exit(EXIT_FAILURE);
  }

  cout << "User " << new_socket << " has connected to the file system!\n";

  pthread_t new_conn;

  argPointers *worker_args = (argPointers *)malloc(sizeof(argPointers));

  (*worker_args).socket_fd = (void *)(long long int)new_socket;
  (*worker_args).file_pointer = (void *)f;

  rc = pthread_create(&new_conn, NULL, worker, (void *)worker_args);
  assert(rc == 0);

  connections.push_back(new_conn);
}
```

Connect to the file system via telnet

```
[elnux3 cs-377-final-project) > telnet localhost 48664
 Trying 127.0.0.1...
 Connected to localhost.
 Escape character is '^]'.
[C file3 2
 Successfully created file file3.
```

Accepted commands

Create - ‘C <fileName> <numBlocks>’

Write - ‘W <fileName> <blockNum>’

Read - ‘R <fileName> <blockNum>’

List - ‘L’

Quit - ‘Q’

DEMO

THE END