Kandidatnummer: 15344

TCP_kode

Connection.c

Klassen «connection.c» implementerer operasjoner på TCP-socketer som blir brukt av hele programmet. De funksjonene håndterer: opprettelse, kobling(connection), lesing, skriving og lukking av TCP-socketer.

Funksjoner:

Tcp_connect(char *hostname, int port): kobles til en remote tcp-server og returnerer socket sin file descriptor.

Tcp_read(int sock, char *buffer, int n): leser data fra tcp-socket og returnerer antall bytes som er lest.

Tcp_write(int sock, char *buffer, int bytes): skriver data til tcp_socket og returnerer antall bytes som er skrevet.

Tcp_write_loop(int sock, char *buffer, int bytes): skriver data til tcp_socket inni en loop til at alle bytes er skrevet og returnerer det totale antall bytes som er skrevet.

Tcp_close(int sock): lukker tcp-socket

Tcp_create_and_listen(int port): oppretter en tcp-server-socket, binder den til en spesifikk port og starter listening fra innkommende connections, så returnerer den server-socket sin file descriptor.

Tcp_accept(int server_sock): aksepterer en connection fra innkommende klient på den server-socketen og returnerer file descriptor til klinet-socketen.

Tcp_wait(fd_set *wating_set, int wait_end): venter for aktivitet på en set av socketer ved å bruke select(), så returnerer antall socketer med aktivitet.

Tcp_wait_timeout(fd_set *wating_set, int wait_end, int seconds): venter for aktivitet på en set av socketer ved å bruke select() med en timeout i sekunder, så returnerer antall socketer med aktivitet.

Disse funksjonene forbereder en API for oppretting, kobling(connection), lesing, skriving og lukking på TCP-socketer samt administrerer server-socketer og ventet på aktivitet på socketer.

Proxy.c:

Event loop i main:

Med FD_ZERO skal først initialisere file descritor set, så ved å bruke FD_SET legger til file descriptoren i en file descriptor set, deretter itererer gjennom clients_lista, hvis klienten i indeks i (clients[i]) ikke er null, så putter clinets[i]->sock i file descriptor set, så sjekker max_fd og eventuelle error, hvis det skjer error i select(), så kommer ut av loopen med break.

I neste steg skal sjekke om returverdien for file descriptor(server_sock) in file descriptor set(&set), hvis det ikke er 0 så sender file descriptor(server_sock) til handleNewClinet(server_sock) for å oppretter en ny klient og putter den i klientlista(clients). Deretter itererer gjennom alle klientene i clients og sjekker at hvis clients[i] ikke er null og clinets[i]-> sock finnes i &set , så kalle på handleClinet(cilent[i]) for å håndterer klienten.

Conversion between XML and binary format:

Når klienten sendes til handleClient(Client *client), sjekker client-> mode, hvis det er XMLSender, så konverterer det til record ved å kalle på xmlToRecode() og deretter kaller på forwardMessage(recode). Hvis client->mode er binSender, så konverterer det til record ved å kalle på BinaryToRecord(), og deretter kaller på forwardmessage(record). I forwardMessage() itererer gjennom alle klienter og sjekker client->destination med record->destination, hvis begge to hadde samme destination, så sjekker client->mode, hvis det var XMLReceiver, så sender videre recorden til recordtoXML. Og hvis det var BinaryReceiver, så sender videre til recordToBinary(), deretter rydder opp recorden ved å kalle på deleteRecord().

Explain what works:

Har testet koden med både valgrind og uten valgrind. Koden passerte alle testene fra test1 til og med test10 i samsvar med expected filene på både min lokal maskin og ifi sin maskin. Sjekket resultatet også ved å bruke diff og har ikke noe memory leak i alle 10 testene.

Parts of the task have not been solved successfully:

Har Error i proxy.c på grunn av grad og course i klassen record.c som ikke er initialisert i precoden og det var ikke lov å endre klassen record.c i precoden.

Om test 11 prøvde å løse den ved å legge til en hjelpefunksjon (splitBufferIntoBinaryRecords()) som kalles i handleClient når client->mode er BinarySender, men dessverre fikk ikke til siden stringLength ikke ble lest riktig, så måtte kommentere den ut.