Progetto Programmazione di reti

ANNO ACCADEMICO 2021-2022

Paolucci Matteo

Matricola: 0000881589

Mail: matteo.paolucci4@studio.unibo.it

Introduzione

La traccia svolta è la numero 2, cioè una coppia di programmi (client e server) che si scambiano dati tramite il protocollo UDP.

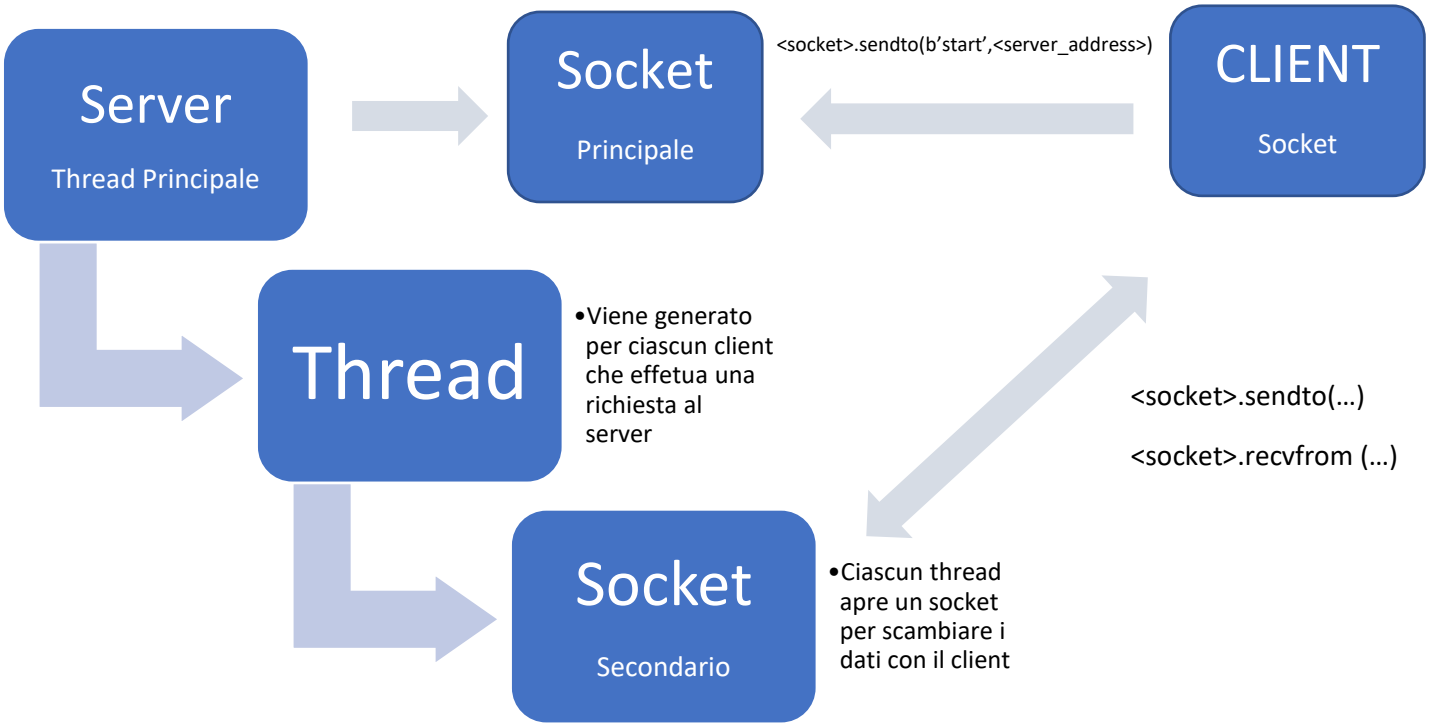
Il client permette sia l'upload di file dal client al server, sia il download di file dal server al client. Inoltre sul client è possibile visualizzare i file locali che possono essere caricati sul server e i file remoti che possono essere scaricati dal server.

Il server può gestire più utenti contemporaneamente mediante l'utilizzo di più socket UDP che vengono eseguiti in parallelo. Il server comunica con il client attraverso dei messaggi che indicano come stanno procedendo le operazioni richieste dal client e se durante quest'ultime generano degli errori.

Architettura generale

Il server si compone di più thread che gestiscono più client contemporaneamente. Il thread principale è quello che rimane sempre attivo e in ascolto, in questo caso, sulla porta 10000. Ogni volta che un client effettua una richiesta sulla porta 10000 all'indirizzo IP del server, viene generato un nuovo thread che gestisce il client e le sue richieste. I thread a loro volta aprono un nuovo socket ciascuno, che permette di scambiare dati con il client che ha effettuato la richiesta. Quando un client non risponde per più di 60 secondi il thread viene automaticamente chiuso e con lui il socket che era stato creato per scambiare dati con il client.

Schema di funzionamento



Dettagli implementativi

Sia il download che l'upload prevedono una ritrasmissione dei dati, da parte del server nel primo caso e da parte del client nel secondo, nel caso in cui venga perso un pacchetto. Per fare ciò si sono utilizzati dei sequence number abbinati a dei pacchetti di ACK.

In caso di perdità di un pacchetto inoltre verrano trasmessi la metà dei dati dall'iterazione successiva, i dati inviati aumenteranno progressivamente se la connessione non genera ulteriori perdite.

Guida utente

Prima di eseguire i programmi è necessario posizionarsi nella root del progetto e scrivere da terminale:

```
pip install -r requirements.txt
```

Per iniziare va avviato il server e in seguito uno o più client.

All'inizio di ciascun programma verrà chiesto di digitare l'inidirizzo IP (valido) del server prima di poter procedere.

Nel client si possono digitare 4 comandi:

- "**list**": stampa i file che sono disponibili sul server
- "**llist**": stampa i file che sono disponibili sul client
- "**get**": permette di scaricare un file dal server
    1. Il server chiederà che file si vuole scaricare
    2. Se il file verrà trovato inizierà il download
    3. Una barra di caricamento mostrerà il progresso del download
    4. Al termine del download ci sarà una conferma lato server
- "**put**": permette di caricare un file sul server
    1. Il server chiederà che file si vuole scaricare
    2. Se il file verrà trovato inizierà l'upload
    3. Una barra di caricamento mostrerà il progresso dell'upload
    4. Al termine dell'upload ci sarà una conferma lato server

Per uscire dai programmi si può utilizzare un Keyboard Interrupt (**CTRL-C**).