|  |  |
| --- | --- |
|  |  |

Ingegneria di Internet e Web

Progetto A.A. 2019/2020

Trasferimento file su UDP reso affidabile con Go-Back N

0244242

Matteo Conti

**Indice**

[1. Architettura e scelte progettuali 2](#_Toc606296459)

[2. Implementazione](#_Analisi_dei_Requisiti) 3

[3. Limitazioni riscontrate 4](#_Toc2081466291)

[4. Piattaforma utilizzata per sviluppo e testing](#_Toc2147004904) 5

[5. Analisi delle prestazioni](#_Toc518560220) 6

[6. Manuale di installazione e configurazione](#_Toc403811585) 7

# Architettura e scelte progettuali

L’architettura é di tipo Client-Server dove il server è di tipo concorrenziale a processi ed offre tre servizi:

* Get, il client scarica un file dal server.
* Put, il client carica un file sul server.
* List, il server invia al client la lista dei file scaricabili.

Il server ed il client fanno utilizzo di UDP come protocollo di livello 4 e la fruizione del servizio avviene in tre fasi:

* Instaurazione della connessione tramite 3-Way-Handshake.
* Erogazione del servizio da parte del server (Get, Put, List) in modo affidabile tramite protocollo Go-Back N.
* Chiusura della connessione tramite 2-Way-Handshake al termine dell’erogazione del servizio.

Tutte e tre le fasi risultano robuste ad eventuali disconnessioni del client evitando di lasciare risorse del server allocate ma inutilizzate.

Si è scelto di realizzare un server concorrenziale che fa utilizzo di processi per semplicitá di sviluppo e manutenibilitá ed in quanto per gli scopi del progetto difficilmente si incorrerá nella saturazione delle risorse di sistema, tuttavia, come si vedrá piú avanti nel paragrafo riguardante [l’analisi delle prestazioni](#_Analisi_delle_prestazioni), a causa del cambio di contesto abbastanza oneroso rispetto ad altre soluzioni (e.g. thread), questo andrá ad incidere in modo abbastanza evidente sulle prestazioni della Get (lato server) se messa in paragone con la Put (lato client).

L’instaurazione come visto avviene tramite 3-Way-Handhsake, cioé vengono scambiati tre messaggi:

* SYN, inviato dal client al server per richiedere di connettersi.
* SYNACK, inviato dal server al client, riscontra il SYN e manda al client un nuovo numero di porta che sará associato al socket del figlio che si occuperá di servire il client.
* ACKSYNACK, inviato dal client al server, riscontra il SYNACK e serve al server per avere conferma che il client è ancora presente ed ha ricevuto le informazioni necessarie per la fruizione del servizio, in questo modo se ció non è vero il server puó deallocare le risorse dedicate a quello specifico client e renderle di nuovo disponibili per altre eventuali richieste di connessione.

Il protocollo Go-Back N è stato leggermente modificato al fine di permettere la rilevazione di connessioni morte, questo è stato realizzato introducendo un numero massimo di 10 ritrasmissioni consecutive.

La chiusura della connessione è stata realizzata tramite 2-Way-Handshake dato che non è necessario deallocare particolari risorse, in questa fase vengono scambiati due messaggi:

* FIN che indica la fine dell’erogazione del servizio e conseguente chiusura della connessione, nei casi di Get e List esso viene mandato dal server al client, nel caso della Put viene inviato dal client al server.
* FINACK che riscontra il FIN, inviato dal client al server in caso di Get e List e dal server al client in caso di Put.

# Implementazione

# Limitazioni riscontrate

# Piattaforma utilizzata per sviluppo e testing

# Analisi delle prestazioni

# Manuale di installazione e configurazione