

Progetto Reti Informatiche A.A. 2021/2022

Prof. Giuseppe Anastasi, Ing. Francesco Pistolesi

cod. 545II, 9 CFU

Daniel Deiana, 596571



UNIVERSITÀ DI PISA

Le scelte implementative principali del protocollo di rete sono state quelle di usare IO multiplexing e socket bloccanti con il protocollo TCP , questo perchè vista la natura dell'applicazione si preferisce un approccio più affidabile rispetto ad uno più veloce. Per gestire le disconnessioni improvvise invece si sceglie di mantenere le connessioni attive per tutta l'esecuzione del programma ed usare i socket TCP per rilevarle.

Messaggi e Formato

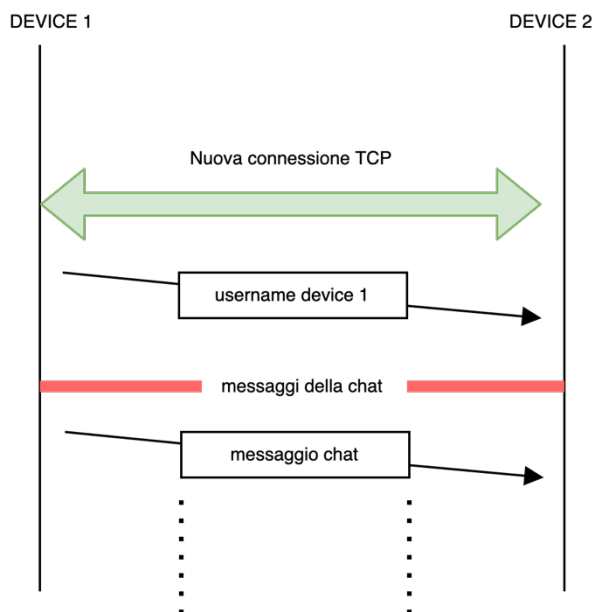
Viene definito un messaggio di servizio per le comunicazioni fra client e server, utilizzato come primo messaggio mandato al server per ogni richiesta del client.

req_type	Options
----------	---------

Il campo req_type ha dimensione 1 byte mentre options ha dimensione variabile. Il primo campo viene usato dal server per discriminare il tipo di richiesta fatta dal client, nel campo options vengono specificate informazioni aggiuntive relative alla richiesta da fare al server. Il mess. di servizio viene anche usato a volte dal server per rispondere al client (utilizzando il campo options) e lasciando il campo req inalterato

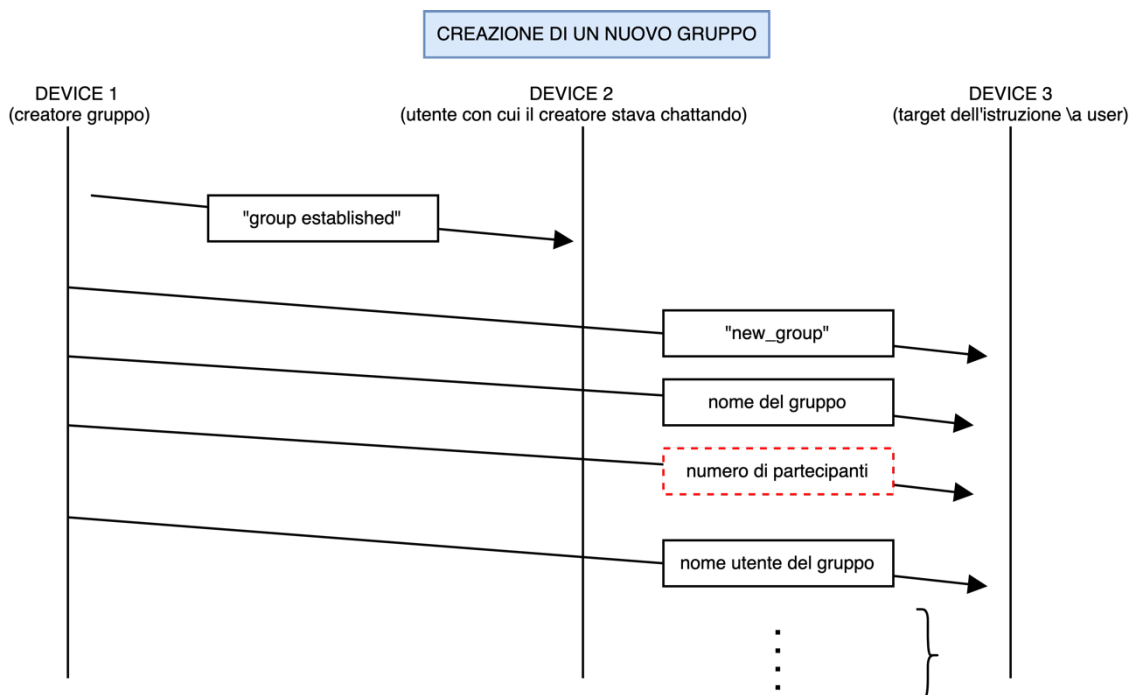
Chat singole e di gruppo

Per iniziare una chat viene stabilita una connessione TCP con il device target, se da una chat singola voglio passare ad una chat di gruppo il device della chat singola con cui la inizio deve essere necessariamente online. All'inizio di una connessione con un altro device viene sempre inviato il proprio nome.



Creazione di un gruppo

Per creare un gruppo bisogna essere già in chat con un altro utente (per poter accedere al comando \a user). Viene creato un gruppo nuovo la prima volta che si esegue un comando \a, le volte successive (sempre rispetto alla stessa chat di gruppo) aggiungono un nuovo utente al gruppo già esistente.



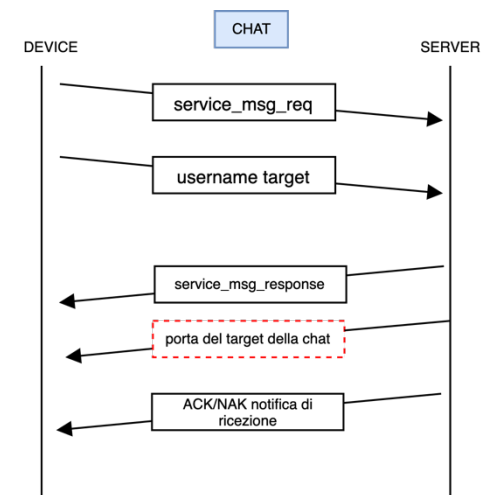
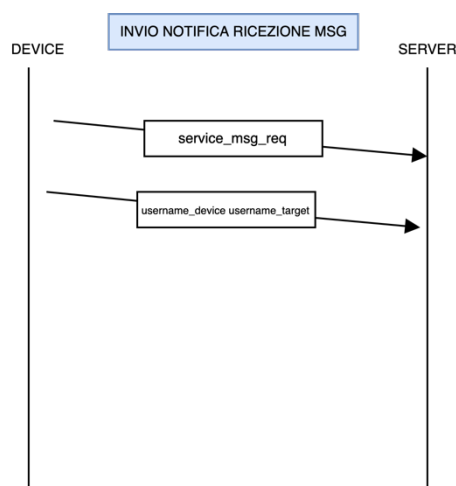
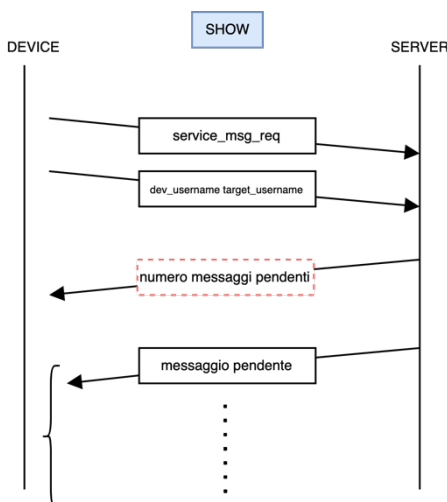
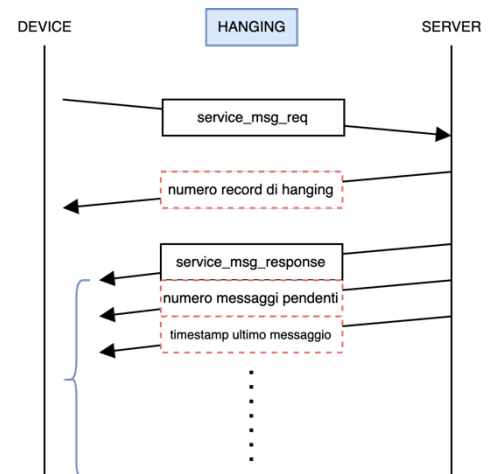
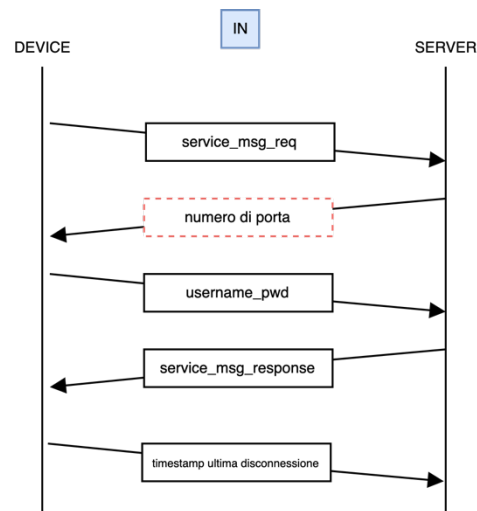
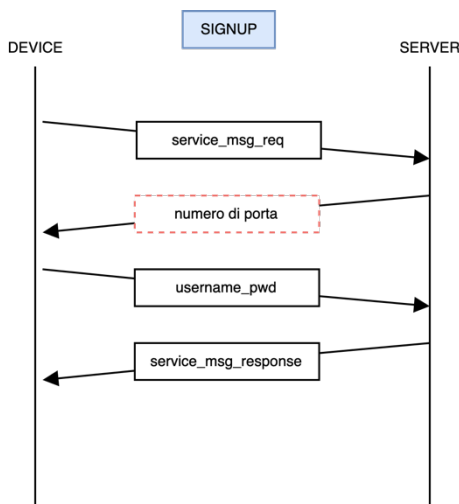
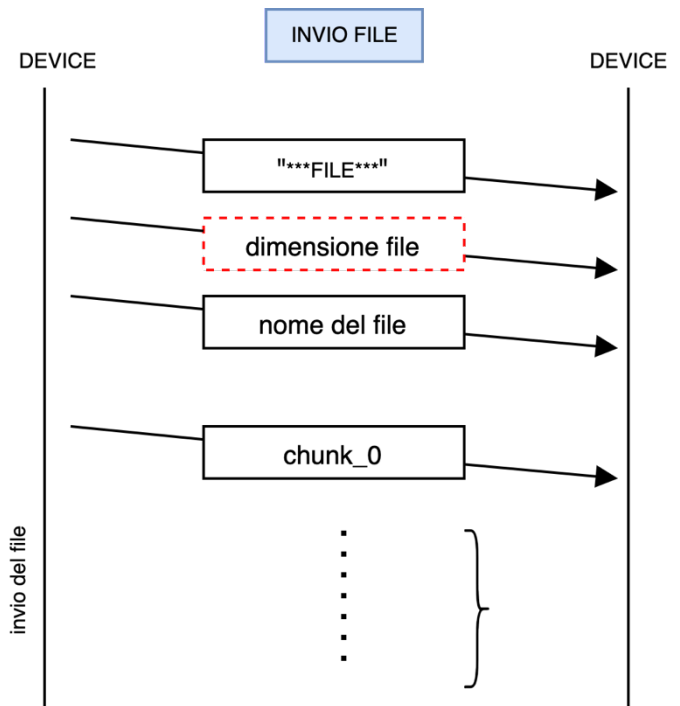
Nell'immagine precedente device 3 riceve i nomi di tutti i partecipanti e, se non già presenti, istanzia con loro delle connessioni TCP, gli altri utenti che si vedono arrivare la richiesta (come device 2 che è stato avisato del nuovo gruppo) sono già in un gruppo e sanno che dovranno aggiungere il nuovo arrivato alla loro lista di utenti. Gli stessi messaggi inviati ad user3 da user1 vengono inviati ogni qualvolta si voglia aggiungere un nuovo utente alla sessione.

Invio di File

Per l'invio dei file il protocollo di trasporto è TCP dato che anche in questo caso preferiamo l'integrità del contenuto. Nel caso di chat di gruppo la soluzione adottata è la stessa. Per condividere un file è necessario che esso si trovi nella cartella personale del device.

Nota: Nei diagrammi i blocchi tratteggiati simboleggiano dei dati per cui precedentemente non viene inviata la dimensione.

Nella parte restante della documentazione seguono i diagrammi di tutte le interazioni fra device e server, per avere dettagli maggiori guardare i commenti nel codice.



debolezze delle scelte progettuali

1. Tenere le connessioni aperte fa sprecare risorse (buffer, etc) soprattutto al server al crescere del numero di utenti online.
2. Il protocollo non prevede una completa bit transparency sui messaggi delle chat.