

RELAZIONE PROGETTO RETI INFORMATICHE 2020-2021

STRUTTURA DELLA RETE:

L'applicazione distribuita è stata realizzata con una struttura ad anello implementata tramite l'utilizzo di una lista circolare.

Un peer può entrare a far parte della rete inviando al DS server dei messaggi UDP di boot con la richiesta di registrazione. Il DS server, sempre attraverso messaggi UDP, comunica al peer neo-aggiunto i suoi vicini. In alcuni casi, l'arrivo di un nuovo peer causa la modifica dei vicini di alcuni dei peer già presenti nella rete; per cui il DS server individua questi nuovi vicini e li invia ai peer i cui vicini risultano mutati, ciò avviene sempre mediante messaggi UDP di aggiornamento.

Ogni peer avrà accesso al numero di porta dei suoi vicini, che potrà utilizzare per comunicare con essi ogni qual volta abbia bisogno di scambiare dati, attraverso connessioni TCP.

Una volta concluso lo scambio di dati verranno chiuse le connessioni TCP. Tale scelta è guidata dall'ipotesi che essendo la rete sufficientemente grande è probabile che ogni peer possa cambiare i vicini abbastanza di frequente, quindi l'utilizzo di connessioni TCP persistenti con essi risulterebbe poco vantaggioso.

SCAMBIO DI MESSAGGI

Per lo scambio dei messaggi si utilizza una struttura dati chiamata "Message".

Il messaggio è composto dai campi:

- 'type' il quale specifica la tipologia del messaggio (es. boot, disconnessione, flooding, aggiornamento, etc...);
- 'info' che è utilizzato per contenere informazioni aggiuntive (qualora necessarie) riguardo al messaggio inviato;
- 'port' che viene utilizzato sia durante lo scambio di messaggi di boot, update e disconnect, per la comunicazione della porta, che durante il flooding, per memorizzare il peer che ha fatto la richiesta di flooding;
- 'address' che viene utilizzato nei messaggi di boot e disconnessione;
- 'data' che viene utilizzato per scambiarsi dati numerici (es. entry, dati aggregati, etc...),
- 'payload' che contiene il messaggio di tipo testuale (es. nome di un file aggregato, nome di un register, etc...).

SALVATAGGIO DEI DATI

I dati vengono salvati all'interno di file. Nello specifico sono presenti 4 tipologie di file:

- 1) File di tipologia dd-mm-yyyy.txt, che si trova nel percorso *files/(numero_peer)/register/*. Tale file corrisponde al register e contiene le entry relative al giorno con il quale è denominato. All'interno del file possiamo trovare 4 righe. La prima inizia con 'T: ' e indica il numero totale di entry, relative al numero di tamponi, raccolte da quel peer in quel dato giorno. La seconda che inizia per 'N: ' e assume il medesimo significato della seconda riga ma la misurazione riguarda il numero di casi. La terza che inizia per 'TT: ' indica invece il numero totale di entry, relative ai tamponi, raccolte da tutti i peer in quel dato giorno. Qualora il valore di 'TT: ' risulti 0, vuol dire che non è stato ancora calcolato. La quarta infine, inizia con 'TN: ' e ha il medesimo significato della terza riga, ma rappresenta le entry relative ai nuovi casi.
- 2) File di tipologia tot_dd-mm-yyyy_dd-mm-yyyy.txt, che si trova nel percorso *files/(numero_peer)/register/aggr/(type)/* (type può essere sostituito da 'tamponi' o 'casi' a seconda del tipo di dato su cui si vuole calcolare l'aggregato). In esso possiamo trovare un valore numerico che rappresenta l'aggregato totale, per l'intervallo rappresentato dal nome del file.
- 3) File di tipologia var_dd-mm-yyyy_dd-mm-yyyy.txt, che si trova nel medesimo percorso indicato al punto 2. Esso rappresenta l'aggregato variazione, per l'intervallo rappresentato dal nome del file.

Sono presenti tante righe, quanti sono il numero di giorni dell'intervallo. La struttura della riga è rappresentata dalla forma semplificativa '*date1-date2: d*', dove *date1* è la data più remota e *date2* quella più recente e con *d* il valore della variazione.

- 4) File di tipologia *last_register_saved.txt*, che si trova in *files/(numero_peer)/*. In questo file è contenuto l'ultimo register, le cui entry sono state inviate ai vicini, in seguito ad una disconnessione del peer con il comando stop.

SCAMBIO DEI DATI

Nel caso in cui il peer debba calcolare dei dati aggregati, ma non abbia a disposizione tutte le entry per poter effettuare il calcolo, per prima cosa richiede ai vicini se abbiano già pronto quel dato aggregato, attraverso dei messaggi TCP. In tali messaggi il campo '*type*' è uguale o a T, nel caso del totale, oppure a V, nel caso della variazione. Se i vicini non possiedono tali dati, il peer deve ricavarsi tutte le entry complessive della tipologia richiesta nel comando '*aggr*', per ogni giorno appartenente all'intervallo. Quindi per ogni giorno il peer legge il register corrispondente e legge il campo TT/TN del file; se possiede quel dato ha terminato per quel giorno, altrimenti deve richiedere quel dato ai suoi vicini attraverso un messaggio TCP con '*type*' uguale a P. Qualora nemmeno loro possedessero quel file, allora il peer fa partire un flooding, attraverso un messaggio TCP con '*type*' uguale ad F per calcolare tale dato. Il flooding prevede di inviare il dato T/N al suo vicino, il quale lo sommerà a quello in suo possesso, ed a sua volta invierà tale somma al suo vicino e così via. Il flooding termina in due circostanze: (1) tutti i peer sono stati raggiunti, (2) se uno di questi peer possiede il dato di tipo TT/TN, in tal caso lo comunicherà direttamente al peer che ha fatto partire il flooding.

Una volta calcolato questo dato il peer lo memorizzerà nel proprio campo TT/TN e passerà al giorno successivo dell'intervallo. Utilizzando questa politica, durante il primo flooding per quel giorno dovranno essere attraversati tutti i nodi della rete cioè *n*, per cui ci porremo nel caso peggiore in analisi.

Ma già a partire dal secondo flooding il numero massimo di nodi che dovrà essere attraversato corrisponde a *n-1*, fino ad arrivare all'*n*-esimo flooding in cui nel caso peggiore deve fare solo uno step.

CALCOLO DEI DATI AGGREGATI

Quando il peer ha ottenuto il valore totale delle entry, per ogni data all'interno dell'intervallo, si procede con il calcolo del dato aggregato.

SESSIONE E SALVATAGGIO

Quando un peer si disconnette attraverso il comando stop, il peer legge la data presente nel file *last_register_saved.txt* per controllare l'ultimo register del quale aveva comunicato le entry ai vicini. Per tutti i register successivi a quella data si calcola il valore TT/TN attraverso il flooding e lo si invia ai suoi vicini. Nel caso in cui un peer debba comunicare le entry relative al giorno corrente ma si disconnetta prima delle 18, allora il peer comunica ad un vicino il valore T/N e solo successivamente verrà azzerato. Il vicino, ricevuto il valore, lo somma al valore T/N contenuto nel suo register.

I file non vengono cancellati dai percorsi in cui risiedono, in quanto ad un successivo ricollegamento, non verranno persi i valori complessivi ed aggregati che sono stati calcolati durante la connessione alla rete. Una volta copiati tutte le entry, per ogni register superiore alla data in *last_register_saved.txt*, questo file viene aggiornato con la data in cui è avvenuta la disconnessione del peer.

SVANTAGGI

Lo svantaggio principale, risiede nella stessa struttura ad anello utilizzata per implementare la rete. Infatti l'operazione di flooding, nel caso peggiore, comporta l'invio di un pacchetto che deve attraversare tutti i nodi presenti nella rete. Se i nodi sono numerosi, i tempi di attesa per tale operazione aumenteranno in base all'aumentare del numero dei nodi, per questo motivo vengono introdotti nei register i valori TT/TN, con lo scopo di diminuire questi tempi di attesa.