

Reti di Calcolatori ed Ingegneria del Web - A.A. 2017/18

Progetto C: Sistema P2P distribuito per la condivisione di file

Lo scopo del progetto è quello di realizzare in linguaggio C usando l'API del socket di Berkeley un sistema P2P per la condivisione di file con directory distribuita. Le caratteristiche salienti del sistema sono:

- Architettura P2P distribuita, in cui sono presenti peer e superpeer con le seguenti caratteristiche:
 - Ogni peer è associato ad un superpeer;
 - I superpeer tengono traccia dei file condivisi dai peer ad esso associati;
 - I superpeer sono collegati tra loro tramite una rete "overlay".
- Comunicazione tra peer e superpeer basata su un opportuno protocollo applicativo da definire. Il protocollo applicativo può usare TCP o UDP come protocollo di trasporto; la scelta deve essere motivata nella relazione, analizzandone vantaggi e svantaggi.
- Comunicazione tra superpeer basata su un opportuno protocollo applicativo da definire. Il protocollo applicativo può usare TCP o UDP come protocollo di trasporto; la scelta deve essere motivata nella relazione, analizzandone vantaggi e svantaggi.
- Un server centralizzato di bootstrap che tiene traccia dei superpeer presenti;

Il software realizzato deve permettere:

- l'ingresso (comando join) e l'uscita (comando leave) di un peer dalla rete P2P;
- la registrazione e l'uscita dei superpeer e la relativa gestione della rete overlay;
- la visualizzazione sul peer della lista dei file locali condivisi;
- il discovery del peer (dei peer) che possiede (possiedono) un determinato file tramite interrogazione del superpeer (comando whohas);
- il download di un file da un peer;

Funzionalità dei peer

Ciascun peer, di tipo concorrente, deve fornire le seguenti funzionalità:

- l'invio di un messaggio di query al nodo di bootstrap per conoscere la lista (non necessariamente completa) dei superpeer presenti;
- l'invio del messaggio join al nodo superpeer prescelto per entrare nella rete P2P; il messaggio contiene la lista dei file condivisi dal peer locale;
- l'invio del messaggio leave al directory server per uscire dalla rete P2P;
- l'invio del messaggio whohas al directory server per ottenere la peerlist dei peer che possiedono un determinato file; il messaggio contiene il nome del file;
- la selezione di un peer dalla peerlist ottenuta in risposta al messaggio whohas (se la peerlist contiene più di un peer). La selezione del peer a cui inviare il messaggio get può avvenire in modo casuale oppure basandosi su un criterio di prossimità (ad esempio ping).
- l'invio del messaggio update al superpeer; il messaggio contiene la lista dei file aggiornati sul peer rispetto a quanto precedentemente comunicato (tramite i comandi join o update). L'aggiornamento può essere effettuato dopo ogni modifica alla lista dei file condivisi dal peer oppure periodicamente;
- l'invio del messaggio get ad un peer remoto per ottenere un file;

- la ricezione del messaggio get inviato da un peer remoto e l'invio del file richiesto o la gestione dell'eventuale errore.

Funzionalità del superpeer

Ciascun nodo superpeer, di tipo concorrente, oltre alle stesse funzionalità previste per un nodo peer, deve fornire:

- la gestione efficiente di una directory contenente i nomi dei file condivisi e gli indirizzi IP dei nodi peer che possiedono tali file;
- l'invio di un messaggio di query al nodo di bootstrap per conoscere la lista (non necessariamente completa) dei superpeer presenti;
- la gestione della rete overlay tra i superpeer;
- la ricezione del messaggio join, contenente informazioni sul peer che intende partecipare alla condivisione dei file e la lista dei file che il peer mette a disposizione. In conseguenza della ricezione di join, il superpeer deve aggiornare la directory ed inviare un ack al peer;
- la ricezione del messaggio leave, contenente un identificativo del peer che intende uscire dalla condivisione dei file. In conseguenza della ricezione di leave, il superpeer deve aggiornare la directory ed inviare un ack al peer;
- la ricezione del messaggio whohas e l'invio al peer della peerlist, ovvero la lista dei peer che possiedono il file richiesto;
- l'invio di un messaggio whohas ai superpeer vicini e la ricezione delle peerlist per la ricerca dei peer che possiedono il file richiesto;
- la ricezione del messaggio update, contenente la lista dei file aggiornati sul peer rispetto a quanto precedentemente comunicato al (tramite i comandi join o update);

I peer ed i superpeer devono essere eseguiti nello spazio utente e senza richiedere privilegi di root; il directory server deve essere in ascolto su una porta di default (configurabile).

Scelta e consegna del progetto

Il progetto può essere realizzato da un gruppo composto al massimo da 4 studenti.

Per poter sostenere l'esame nell'A.A. 2017/18, è necessario prenotarsi per il progetto, comunicando al docente le seguenti informazioni:

- nominativi ed indirizzi di e-mail dei componenti del gruppo;
- progetto scelto.

Per ogni comunicazione via e-mail è necessario specificare [IW16] nel subject della mail.

Eventuali modifiche relative al gruppo devono essere tempestivamente comunicate e concordate con il docente.

Il progetto è valido solo per l'A.A. 2017/18: i progetti scadranno inderogabilmente con la sessione d'esame di febbraio 2019.

La consegna del progetto deve avvenire almeno dieci giorni prima della data stabilita per la discussione del progetto.

La consegna del progetto consiste in:

- un CD-ROM contenente tutti i sorgenti (opportunamente commentati) necessari per il funzionamento e la copia elettronica della relazione (in formato pdf);
- la copia cartacea della relazione.

La relazione contiene:

- la descrizione dettagliata dell'architettura del sistema e delle scelte progettuali effettuate;
- la descrizione dell'implementazione;
- la descrizione delle eventuali limitazioni riscontrate;
- l'indicazione della piattaforma software usata per lo sviluppo ed il testing del sistema ;
- alcuni esempi di funzionamento;
- un manuale per l'installazione, la configurazione e l'esecuzione del sistema.

Valutazione del progetto

I principali criteri di valutazione del progetto saranno:

- rispondenza ai requisiti;
- originalità;
- efficienza;
- scalabilità;
- robustezza ai guasti;
- robustezza al churn;
- leggibilità del codice;
- modularità del codice;
- organizzazione, chiarezza e completezza della relazione;
- semplicità di installazione e configurazione del software realizzato in ambiente Linux.