

Programmazione in ambienti distribuiti I

(01FQT, 01FEL, 06EKE, 02HDF)

AA 2009-2010, Esercitazione di laboratorio n. 2

Esercizio 2.1 (client-server UDP base)

Scrivere un client che invii ad un server UDP (all'indirizzo e porta specificati come primo e secondo parametro sulla riga di comando) un datagramma contenente il nome (massimo 31 caratteri) passato come terzo parametro sulla riga di comando, attenda quindi un qualunque datagramma di risposta dal server. Il client termina mostrando il contenuto (testo ASCII) del datagramma ricevuto oppure segnalando che non ha ricevuto risposta dal server.

Sviluppare un server UDP (in ascolto sulla porta specificata come primo parametro sulla riga di comando) che risponda ad ogni datagramma ricevuto inviando un datagramma di risposta contenente la stringa composta da "Ciao, " seguita dal nome contenuto nel datagramma ricevuto e quindi dal carattere ".".

Provare ad effettuare degli scambi di datagrammi tra client e server con le seguenti configurazioni:

- client invia datagramma ad una porta su cui il server è in ascolto
- client invia datagramma ad una porta su cui il server non è in ascolto
- client invia datagramma ad un indirizzo non raggiungibile (es. 10.0.0.1)

Provare a collegare il proprio client col server sviluppato da un altro gruppo e viceversa.

Esercizio 2.2 (client UDP perseverante)

Modificare il client dell'esercizio 2.1 in modo che – nel caso non riceva risposta dal server entro 3 secondi – ritrasmetta la richiesta (fino ad un massimo di 5 volte) e quindi termini indicando se ha ricevuto risposta o meno.

Effettuare le stesse prove dell'esercizio 2.1.

Esercizio 2.3 (server UDP limitante)

Modificare il server dell'esercizio 2.1 in modo che invii risposta ad un client solo se questi non ha effettuato più di tre richieste (dall'attivazione del server). Il server deve essere in grado di riconoscere gli ultimi 10 client che hanno fatto richiesta.

Provare quindi ad attivare verso questo server quattro volte il client dell'esercizio 2.2.

Provare infine ad attivare verso questo server due client posti su due nodi di rete diversi, in alternanza tra loro, quattro volte per ciascun client.

Esercizio 2.4 (server TCP iterativo)

Sviluppare un server TCP (in ascolto sulla porta specificata come primo parametro sulla riga di comando) che accetti richieste di trasferimento file da client ed invii il file richiesto.

Sviluppare un client che si colleghi ad un server TCP (all'indirizzo e porta specificati come primo e secondo parametro sulla riga di comando) per richiedere dei file e memorizzarli localmente. I nomi dei file da richiedere vengono forniti su standard input, uno per riga. Ogni file richiesto deve essere salvato localmente e deve essere fornito su standard output un messaggio circa l'avvenuto trasferimento, con nome e dimensione del file.

Per richiedere un file il client invia al server i tre caratteri ASCII "GET" seguiti dal nome del file (max 64 caratteri) e da CR LF (senza alcun spazio):

GET *nomefile* CR LF

Il server risponde inviando:

+OK CR LF

seguito dal numero di byte (intero su 32 bit in formato di rete) e quindi dai byte del file in oggetto.

Il client può richiedere più file inviando più comandi GET. Quando intende terminare la comunicazione invia:

QUIT CR LF

e chiude il canale.

In caso di errore (es. comando illegale, file inesistente) il server risponde sempre con

-ERR CR LF

e quindi chiude il canale col client.

Provare a collegare il proprio client col server sviluppato da un altro gruppo e viceversa.

Mentre è in corso un collegamento provare ad attivare un secondo client verso il medesimo server.

Provare ad attivare sul medesimo nodo una seconda istanza del server sulla medesima porta.

Provare a collegare il client ad un indirizzo non raggiungibile.

Provare a collegare il client ad un indirizzo esistente ma ad una porta su cui il server non è in ascolto.

Provare a disattivare il server (battendo ^C nella sua finestra) mentre un client è collegato.