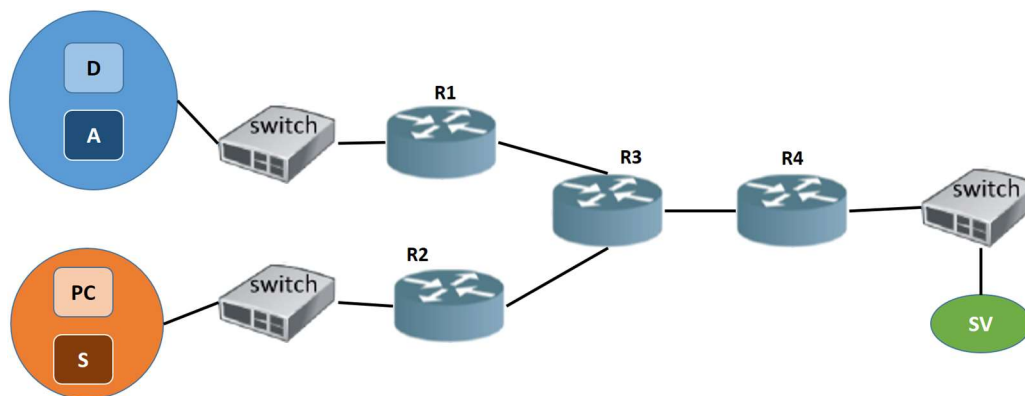


Esame di Laboratorio di Reti di Calcolatori – febbraio 2021

Prova Packet Tracer

La rete di un Dipartimento ha indirizzo base 105.3.188.0/23 ed è organizzata in tre edifici collegati tra loro da switch e router come mostrato dal seguente schema:



La rete blu comprende due VLAN: **D** (docenti) comprendente 72 host, e **A** (amministrazione) comprendente 21 host.

La rete rossa comprende due VLAN: **PC** (aula informatizzata) contenente 49 host, e **S** (aula studenti) che può contenere fino a 96 laptop. I laptop degli studenti in **S** possono collegarsi alla rete assumendo un indirizzo dinamico fornito da un DHCP server incluso fra i 49 host della VLAN **PC**.

La rete verde **SV** è una VLAN contenente 6 server.

I parametri delle varie reti sono forniti nelle tabelle contenute nel file *griglia_F21.pdf*.

Per tale infrastruttura di rete lo studente è richiesto:

- Configurare almeno due host per ogni VLAN come da parametri forniti.
- Dentro **PC** configurare un server DHCP per gli host in **S**. Configurare gli host in **S** in modo che assumano un indirizzo dinamico da tale server.
- Configurare opportunamente switch e router in base ai parametri forniti e alla specifica sopra data, così da consentire la comunicazione sia tra host appartenenti alla medesima VLAN, sia tra host appartenenti a VLAN differenti. *Non aggiungere altri apparati oltre a quelli mostrati in figura.*
- Configurare i router in modo che usino il protocollo di instradamento RIP versione 2. Si faccia in modo che il protocollo RIP non inondi di messaggi le sottoreti non contenenti altri router.
- Configurare un web server in **SV** che sia accessibile da ogni host incluso nella rete.

Modalità di consegna: lo studente deve consegnare la sola activity, nominata *cognome-nome.pkt*, via sito <http://upload.di.unimi.it> autenticandosi con login name e password di Ateneo (cioè @studenti.unimi.it)

Prova Java Socket

Si realizzi un sistema distribuito per il calcolo dei parametri di connessioni TCP. Il comportamento dei processi è definito dalle seguenti specifiche:

CLIENT:

1. comunica con il server utilizzando i servizi di UDP.
2. Alla partenza, chiede l'introduzione da tastiera del proprio nickname, e di due parametri di peso α per l'aggiornamento del valore del RTT di una connessione TCP, e β per l'aggiornamento del valore di

deviazione D di una connessione TCP. Contatta il server presentandosi con il proprio indirizzo e nickname, e con i due pesi.

3. All'immissione da tastiera di un carattere qualsiasi diverso da '.', genera un numero Double tra 1 e 100 che rappresenta la misura M (in ms) del tempo tra il primo invio di un segmento e la ricezione del relativo ack. Mostra tale valore a video, e invia al server un messaggio contenente nickname e valore generato M.
4. Dopo di che si pone in attesa di ricevere dal server i valori aggiornati di RTT, D e RTO, e li stampa a video.
5. Quindi torna a chiedere l'immissione di un carattere da tastiera. In caso il carattere sia differente da '.' riprende dal punto (3), altrimenti invia '.' al server e poi termina.

SERVER:

1. comunica con i client utilizzando i servizi di UDP.
2. Quando riceve un messaggio da un client il cui nickname non è conosciuto, inserisce il nickname e l'indirizzo del client in una propria struttura, insieme ai coefficienti di peso indicati per l'aggiornamento di RTT e D.
3. Quando riceve un messaggio da un nickname conosciuto:
 - a. Se è '.', rimuove nickname e informazioni associate dalla propria struttura
 - b. se è la prima misura M ricevuta da quel nickname, memorizza nella propria struttura interna associata al nickname del client i valori di RTT pari a M, e di D pari a M/2. Calcola il valore di RTO in base a quei valori di RTT e D e invia la terna RTT, D, RTO al client
 - c. se è una misura M successiva alla prima:
 - i. recupera dalla propria struttura i valori attuali di RTT e D
 - ii. calcola i nuovi valori di RTT e D usando M e i pesi inizialmente forniti da quel client
 - iii. calcola il valore di RTO in base a RTT e D aggiornati, e invia la terna RTT, D, RTO al client.
4. Il processo server non termina mai.

La definizione del formato dei messaggi, così come i dettagli implementativi non specificati sopra, sono a discrezione dello studente. Tutti i messaggi ricevuti da ogni processo devono essere mostrati a video.

Modalità di consegna:

Lo studente deve consegnare due file sorgenti cliente.java, server.java; qualunque altro file presente nella consegna non verrà considerato per la correzione.

1. Prima di consegnare, assicurarsi di *salvare il contenuto di tutti i file sorgente* prodotti
2. comprimere i file sorgenti in un *archivio **cognome-nome.zip***; altri formati di compressione non verranno accettati
3. Collegarsi al sito <http://upload.di.unimi.it>
4. Autenticarsi con login name e password di Ateneo (cioè @studenti.unimi.it)
5. Fare upload del *file .zip*
6. Fare logout dal sito.

NB: in caso si procedesse ad effettuare la consegna dell'elaborato più di una volta, concatenare un numero progressivo al nome file, così che sia possibile individuare l'ultima versione. Per la correzione verrà considerata solo l'ultima versione dell'archivio caricata.