

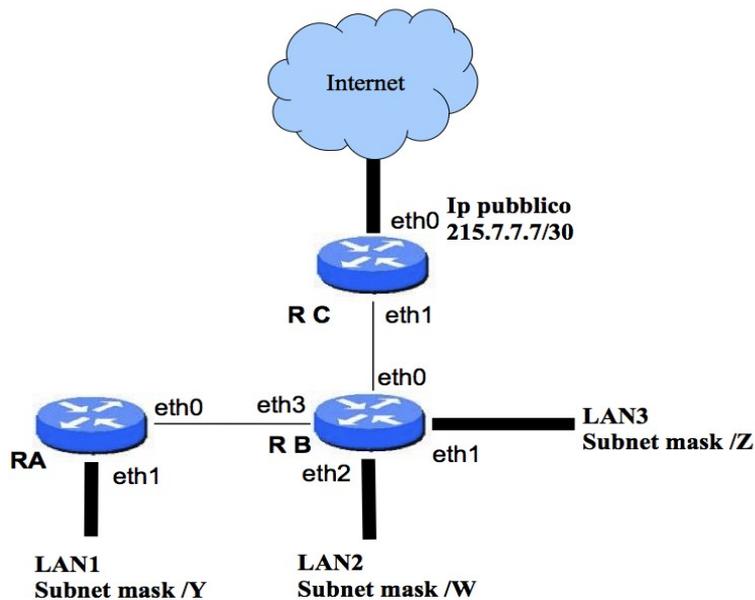


# Reti di calcolatori

"Dry run" del 1 giugno 2012

## Esercizio 1 (11 punti)

I tre router RA, RB, RC sono forniti di schede Ethernet e sono fra loro collegati secondo le modalità rappresentate in figura.



Configurazione della rete per l'esercizio 1

Al router RA è collegata la LAN 1 con subnet mask (rappresentata in *slash notation*) eguale a  $/Y$  ove Y vale 27,26 e 25 a seconda che la lettera iniziale del **nome** dello studente sia rispettivamente compresa fra A-M,N-R,S-Z.

Il router RB ha collegato le reti denominate LAN2 e LAN3. LAN2 ha subnet mask di tipo  $/W$  ove W vale 27,28,29 a seconda che la lettera iniziale del **cognome** dello studente sia rispettivamente compresa nei range A-M,N-R,S-Z. LAN3 ha subnet mask di tipo  $/Z$  ove Z vale rispettivamente 28, 27, 26 a seconda che la lettera iniziale del **cognome** dello studente sia compreso rispettivamente nei range A-M,N-R,S-Z. Infine RC è collegato, tramite il router messo a disposizione da un ISP (Internet Service Provider) e non rappresentato in figura, ad Internet. Alla scheda eth0 di RD è assegnato da ISP l'indirizzo pubblico 215.7.7.7.

Il range di indirizzi IP a disposizione va da **194.6.10.17** a **194.6.10.255**.

Si chiede di:

- assegnare, spiegando il criterio utilizzato, gli indirizzi di rete e broadcast alle LAN 1,2, 3 ed ai link RA-RB, RB-RC;
- indicare la configurazione della tabella di routing del router RB, che dovrà essere in grado di inviare/ricevere pacchetti a/da Internet;
- spiegare, con un esempio, la struttura di un indirizzo IP;
- indicare i trattamenti eseguiti all'interno del router RB, quando una trama Ethernet, contenente un pacchetto IP destinato ad Internet, arriva sull'interfaccia eth2.

## Esercizio 2 (11 punti)

Si consideri il protocollo CSMA (Carrier Sense Multiple Access) e si risponda alle seguenti domande:

1. Come si comporta una stazione quando, dovendo trasmettere un pacchetto, trova il canale libero?
2. Si spieghi la differenza tra la versione 1-persistente e la versione 0-persistente del protocollo, spiegando anche le diverse prestazioni quando il traffico offerto segue un processo degli arrivi di Poisson.
3. In cosa consiste la funzione di Collision Detection (CSMA/CD) ed in che modo modifica le prestazioni del protocollo?
4. Quali sono le condizioni (velocità del canale, dimensione della rete, dimensione dei pacchetti, ...) in cui la funzione CD funziona correttamente e quali invece quelle per cui è inutile?

## Esercizio 3 (11 punti)

Si deve realizzare un semplice protocollo di livello applicativo per supportare l'interrogazione delle basi dati di funzionamento dispositivi collegati in rete, ad esempio per un sistema di automazione domestica. Il protocollo, di tipo client/server, presuppone che i dispositivi in rete siano sempre in ascolto e possano essere interrogati tramite l'apposita applicazione, in qualsiasi momento.

1. Dove si trovano i server di questo protocollo e dove si trovano i client ?

L'interrogazione delle basi dati avviene con query semplici (si potrebbero, a titolo di esempio, usare delle basi dati SQL per implementare l'applicazione, ma questo è indipendente dal protocollo di supporto), e le risposte sono altrettanto semplici, e non prevedono mai l'invio di grandi moli di dati. Il protocollo è di tipo proprietario e non è quindi necessario ottenere l'assegnazione da parte dell'IETF di "port" dedicate a questo protocollo. Si definisca:

1. se usare a livello trasporto il protocollo UDP oppure TCP e si giustifichi la risposta;
2. i "port" da assegnare ai server ed ai client;
3. il formato di un possibile header per il protocollo da realizzare, anche alla luce della risposta data al punto 2;
4. un esempio di comandi e risposte per questo protocollo, spiegando perché è significativo.

# Soluzioni

## Esercizio 1

Il mio cognome inizia per L, quindi /Y = /27; /W=/27 ; /Z=/28

a) La network mask indica il numero di '1' a sinistra in una word di 32 che, messa in AND logico con un indirizzo IP identifica la parte di rete dell'indirizzo. Quindi /27 indica che vi sono solo 5 bit a 0 e quindi 32 indirizzi IP a disposizione nella subnet, mentre /28 indica che ci sono 4 bit come host-id e quindi 16 indirizzi a disposizione (inclusi indirizzo di rete e broadcast).

Ai link, che sono collegamenti punto-punto e necessitano quindi solo di due indirizzi di host possiamo assegnare subnet con network mask /30 (indirizzo di rete, indirizzo broadcast e due indirizzi host)

Abbiamo a disposizione  $255-17=238$  indirizzi e ne servono  $32+32+16+4+4=88$ , quindi l'assegnazione è fattibile. Assegnamo a LAN1 e LAN2 due blocchi contigui a partire da 194.6.10.128 e a seguire il blocco per LAN3. Gli indirizzi per i due link li possiamo invece assegnare nel range basso degli indirizzi a disposizione, ad esempio .32 e . 36.

Una possibile assegnazione è quindi:

**LAN1** → **194.6.10.128/27**

**LAN2** → **194.6.10.160/27**

LAN3 → 194.6.10.176/28  
RA-RB → 194.6.10.32/30  
RB-RC → 194.6.10.36/30

b)

194.6.10.176/28	eth1
194.6.10.160/27	eth2
194.6.10.32/30; 194.6.10.128/27	eth3
default	eth0

c) Un indirizzo IP è formato da due parti net-id e host-id, che sono una divisione arbitraria in due parti dei 32 bit che compongono l'indirizzo. La separazione è indicata dalla network mask come spiegato all'inizio del punto a). Prendendo come esempio un indirizzo arbitrario di LAN1 abbiamo

network-id 194.1.10.128 | host-id (128)+11  
11000010.00000110.000001010.100 | 01011

ovvero la parte di host-id è quanto si deve "aggiungere" al net-id per formare l'indirizzo IP dell'interfaccia.

d) L'entità Ethernet di input controlla la validità della trama tramite il checksum. Se la trama è valida estrae il pacchetto IP e lo passa all'entità di livello rete (IP). L'entità IP consulta la tabella di routing e, poiché presumibilmente l'indirizzo di destinazione non è noto decide che va inoltrato sull'interfaccia di default, che è quella a cui è collegato il router RC. Inoltre poiché non è una consegna diretta, l'entità IP decide che il next-hop deve essere il router RC. Consulta le tabelle ARP dell'interfaccia IP di uscita e, se l'indirizzo IP di RC non è presente lo risolve tramite ARP. A questo punto il pacchetto IP viene passato all'entità Ethernet di uscita associato all'indirizzo MAC dell'interfaccia eth1 di RC, il pacchetto viene incapsulato e spedito.

## Esercizio 2

1. Ascolta il canale per un tempo pre-definito  $T_a$  e se il canale è libero trasmette.
2. Nel caso 0-persistente su canale occupato la trasmissione viene rimandata per un tempo casuale  $T_r \gg T_t$ , dove  $T_t$  è il tempo di trasmissione di una trama. Nel caso 1-persistente, invece, si tenta la trasmissione appena la funzione di Carrier Sensing indica che il canale è libero. Il throughput ottenuto da 1p è più elevato a carichi molto bassi, ma va a zero per carico che tende ad infinito. il throughpt di 0p è invece stabile e tende ad 1 per carico che tende all'infinito. Il ritardi di accesso per 0p a carichi bassi è però più elevato che nel caso di 1p.

(segue disegno alla lavagna dell'andamento del throughput)

3. La collision detection permette, in reti cablate, ad un trasmettitore di rilevare la presenza di altre trame sul canale oltre a quella che si sta trasmettendo, e quindi di interrompere immediatamente la trasmissione, riducendo lo spreco di risorse in presenza di collisioni. La presenza di CD di fatto consente di rendere stabili anche le versioni 1p del protocollo, consentendo un atteggiamento "aggressivo" delle sorgenti per ridurre il ritardo di accesso delle trame.

4. La funzione CD funziona tanto meglio quanto più la rete ha dimensioni ridotte (basso ritardo di propagazione end-to-end), le trame sono grandi e la velocità di trasmissione è bassa. Invece per reti di grandi dimensioni, con trame piccole e velocità di trasmissione elevata la funzione CD diventa praticamente inutile, perché una collisione al trasmettitore non implica una collisione al ricevitore e viceversa.

## Esercizio 3

1a. Il server è la parte di protocollo che deve sempre essere in ascolto per ricevere richieste di connessione/comunicazione. Quindi i server si troveranno su tutti i dispositivi di automazione domestica. Il client è invece l'entità protocollare che inizia la comunicazione e in questo caso si può trovare su un normale PC oppure su una centralina di controllo del sistema di automazione domestica.

1b. Poiché l'applicazione è di tipo "transazionale" e non orientato alla connessione e poiché la quantità di dati per ogni transazione è piccola e presumibilmente può stare in un solo datagramma, è più semplice e naturale usare UDP e lasciare all'applicazione l'onere di controllare la correttezza della comunicazione ed eventuali richieste di ritrasmissione.

2. I server possono usare una qualsiasi porta superiore a 1024, ad esempio 11111, purché sia nota ai client. I client usano porte effimere (cioè una qualsiasi porta > 1024 e non nota a priori).

3. L'header deve contenere i campi necessari all'applicazione, quindi ad esempio un codice di comando, una versione del protocollo e così via. Poiché siamo a livello applicativo si può scegliere se avere un header con tag testuali "stile http" oppure un header binario con campi di lunghezza fissa.

Segue discussione su possibili diverse soluzioni a questo livello.

4. Un set minimo è composto dai due comandi <set><parameter><value> e <get><parameter> per scrivere e leggere i valore dei parametri dei dispositivi. Ai due comandi corrisponderanno opportuni codici di risposta, ad esempio, per usare le convenzioni di Internet:

200 OK - <parameter><value>

per confermare l'assegnazione di un valore a un parametro con il comando <set> oppure

401 global error - <parameter> does not exist

e così via.

Segue discussione.