

Corso di Reti di Calcolatori
Soluzioni alla prova scritta

Mauro Brunato

Claudio Covelli

David Tacconi

Mercoledì 12 gennaio 2009

Esercizio 1

- 1.1)** Descrivere il metodo di Hamming per il rilevamento degli errori.
1.2) Se un errore di trasmissione colpisce i bit in posizione 4, 33 e 37 di un messaggio invertendoli, che errore verrà riportato dall'algorithm?

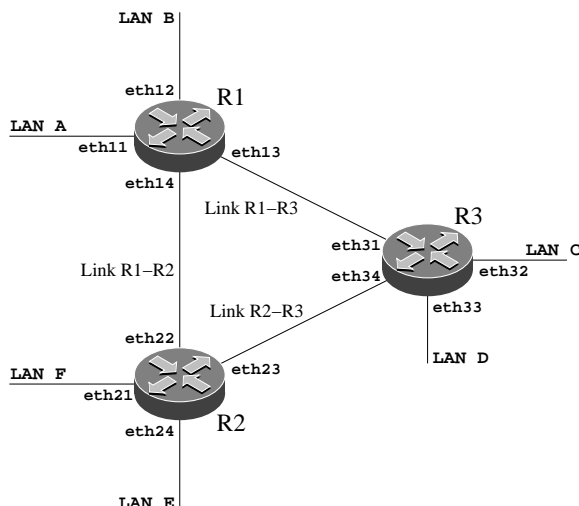
Soluzione —

1.1) Vedere il libro

1.2) I bit di parità coinvolti dalle tre posizioni d'errore sono 1, 4 e 32. Ciascuno di essi è interessato da due degli errori (in altri termini, l'OR esclusivo $4 \oplus 33 \oplus 37$ è nullo): ciascun bit di controllo subisce un numero pari di errori nei bit da lui controllati, quindi le parità risultano corrette e il sistema crede che il messaggio non abbia errori.

Esercizio 2

Si abbiano i 3 router R1, R2, R3 forniti di schede Ethernet e collegati secondo lo schema riportato in figura.



Ad ognuno dei tre router sono collegate, come da figura, due LAN TCP/IP/Ethernet (LAN A – LAN F). Ogni LAN è in grado di contenere fino a un massimo di 30 host . Si ha a disposizione il range di indirizzi IP pubblici da 193.1.7.56 a 193.1.8.40, estremi inclusi.

Si chiede di:

- 2.1)** assegnare, in modo ottimizzato, gli indirizzi IP di rete, di broadcast e la subnet mask alle LAN A–F ed ai link di interconnessione fra i router, spiegando in modo dettagliato le motivazioni alla base delle scelte effettuate
2.2) indicare i range di indirizzi IP rimasti inutilizzati
2.3) indicare la configurazione delle tabelle di routing dei router R1 ed R3 (destination, netmask, gateway, interface)
2.4) spiegare cosa avviene, a livello 3 (IP), quando un host della LAN F cerca di inviare un pacchetto ad un host della LAN A

Soluzione —

2.1) La rappresentazione binaria di 56 è 00111000, che consente la realizzazione di una rete /29 o, più utili ai nostri scopi, di due reti /30 (punto a punto):

Nome	Rete	Broadcast	Host
R1–R2	193.1.7.56/30	193.1.7.59	2
R1–R3	193.1.7.60/30	193.1.7.63	2

A partire dall'indirizzo 64 (rappresentazione binaria 01000000) fino al 255 possiamo allocare 6 reti da 30 host:

Nome	Rete	Broadcast	Host
LAN A	193.1.7.64/27	193.1.7.95	30
LAN B	193.1.7.96/27	193.1.7.127	30
LAN C	193.1.7.128/27	193.1.7.159	30
LAN D	193.1.7.160/27	193.1.7.191	30
LAN E	193.1.7.192/27	193.1.7.223	30
LAN F	193.1.7.224/27	193.1.7.255	30

Resta da allocare una rete punto a punto:

Nome	Rete	Broadcast	Host
R2-R3	193.1.8.0/30	193.1.8.3	2

Alle interfacce di rete possiamo assegnare i seguenti indirizzi (ciascuno all'interno della sottorete di pertinenza:

eth11=193.1.7.65 eth12=193.1.7.97 eth13=193.1.7.61 eth14=193.1.7.57
 eth21=193.1.7.225 eth22=193.1.7.58 eth23=193.1.8.1 eth24=193.1.7.193
 eth31=193.1.7.62 eth32=193.1.7.129 eth33=193.1.7.161 eth34=193.1.8.2

2.2) Rimane libero l'intervallo 193.1.8.4 a 193.1.8.40.

2.3) La tabella di instradamento di R1 è

Destination	Netmask	Gateway	Interface	Spiegazione
193.1.7.64/27	255.255.255.224	—	eth11	LAN A
193.1.7.96/27	255.255.255.224	—	eth12	LAN B
193.1.7.56/30	255.255.255.252	—	eth14	Link R1-R2
193.1.7.60/30	255.255.255.252	—	eth13	Link R1-R3
193.1.7.128/26	255.255.255.192	193.1.7.62	eth13	LAN C e D via R3
0.0.0.0/0	0.0.0.0	193.1.7.58	eth14	Il resto via R2

Si noti che è possibile aggregare alcune reti, e utilizzare la riga di default, per abbreviare la tabella. La tabella di instradamento di R3 è

Destination	Netmask	Gateway	Interface	Spiegazione
193.1.7.128/27	255.255.255.224	—	eth32	LAN C
193.1.7.160/27	255.255.255.224	—	eth33	LAN D
193.1.7.60/30	255.255.255.252	—	eth31	Link R3-R1
193.1.8.0/30	255.255.255.252	—	eth34	Link R3-R2
193.1.7.64/26	255.255.255.192	193.1.7.61	eth13	LAN A e B via R1
0.0.0.0/0	0.0.0.0	193.1.8.1	eth34	Il resto via R2

2.4) L'host della LAN F costruisce il pacchetto IP con l'indirizzo di destinazione. Questo pacchetto non subirà variazioni lungo tutto il percorso (salvo il decremento del Time To Live e le eventuali, improbabili frammentazioni).

Consultando la propria tabella di instradamento, l'host includerà il pacchetto IP in un frame Ethernet destinato all'interfaccia eth21 di R2, il cui indirizzo MAC sarà scoperto tramite interrogazione ARP verso l'IP 193.1.7.225 (gateway per le comunicazioni con tutte le altre reti).

Allo stesso modo, R2 estrae il pacchetto IP, consulta la propria tabella di instradamento e lo include in un frame Ethernet diretto, attraverso il link R1-R2, all'interfaccia eth14 di R1, il cui MAC viene scoperto inviando una query ARP verso l'indirizzo 193.1.7.57 dall'interfaccia eth22.

Il router R1 estrae il pacchetto IP, consulta la propria tabella di instradamento e trova corrispondenza con la prima riga. Include dunque il pacchetto in un frame Ethernet destinato al MAC che otterrà tramite interrogazione ARP verso l'indirizzo di destinazione attraverso la porta eth11.

Finalmente, l'host di destinazione estrae il pacchetto IP dal frame, il suo livello Rete verifica di essere il destinatario finale e inoltra il contenuto ai livelli superiori.

Esercizio 3

Un'applicazione genera dati a una velocità di 500KBps per i primi 10s, e 2MBps per i successivi 10s. Per la spedizione di questi dati è utilizzato un algoritmo Token bucket con i seguenti parametri:

- Un token vale per la spedizione di 1KB.
- L'applicazione può mettere da parte fino a 2000 token.
- L'applicazione riceve 1000 token al secondo.
- Se dei dati non possono essere spediti per mancanza di token, non vengono perduti, perché l'applicazione dispone di un buffer illimitato.
- L'applicazione inizialmente non contiene token.

Descrivere e rappresentare graficamente le grandezze più significative del sistema (ad esempio: velocità dei dati in uscita, occupazione del buffer, disponibilità dei token) dall'istante iniziale al completamento della trasmissione.

Soluzione —

Per semplicità consideriamo $K = 10^3$, $M = 10^6$.

Si veda la figura sottostante. Il primo grafico rappresenta la generazione dei dati da parte dell'applicazione.

Consideriamo (secondo grafico) come evolve il numero di token posseduti dal sistema. Si noti che la velocità di afflusso dei token consente l'invio di 1MBps. Nei primi dieci secondi, dunque, dei 1000 token al secondo che il sistema riceve, 500 vengono usati per la spedizione dei 500KBps. Quindi il numero di token accumulati dal sistema cresce di 500 al secondo. Dopo 4 secondi, l'applicazione possiede 2000 token, e non ne può accumulare altri.

Al decimo secondo, le cose cambiano: l'applicazione comincia a generare 2MBps, quindi ha bisogno di 2000 token al secondo. Visto che la velocità di accumulo dei token è di 1000 al secondo, l'applicazione consuma altri 1000 token al secondo fra quelli accumulati in precedenza per spedire a piena velocità. A 12 secondi, dunque, la riserva di token è esaurita.

A questo punto, l'applicazione può spedire solo 1MBps, quindi comincia a riempire il buffer con i dati restanti, in ragione di 1MBps. Al ventesimo secondo, l'applicazione smette di generare dati, il buffer contiene 8MB, e non possiede token. Quindi riesce a continuare la trasmissione con i token che arrivano, in ragione di 1MBps, finché il buffer non è vuoto, al ventottesimo secondo. A questo punto non v'è nulla da trasmettere e il sechio torna a riempirsi.

