RETI DI CALCOLATORI – quinto appello – 17.07.2013

Per ottenere una valutazione sufficiente all'ammissione alla prova orale è necessario ottenere una valutazione sufficiente della prima parte.

Prima parte (10 punti)

Q1. Supponiamo che in una rete Ethernet due host A e B abbiano stabilito una connessione TCP tra loro, che la lunghezza della LAN sia 180 metri, che la velocità di propagazione sia di 2 x 10^8 m/s e che la frequenza di trasmissione sia di 10 Mbps. Supponendo che i segmenti TCP contenenti dati siano tutti *full-sized*, ovvero contengano 1 MSS di dati, con 1 MSS=1200 byte, indicare –giustificando la risposta– quale deve essere la durata minima del timer affinché non si verifichino sicuramente timeout prematuri.

Q2. Un router IPv4 riceve un pacchetto IP la cui intestazione non contiene opzioni e il cui campo *length* contiene il valore 4000 e deve inoltrare tale pacchetto su un canale il cui MTU è 2000. Indicare –giustificando la risposta– in che modo il router frammenta tale pacchetto, specificando il valore del campo *length* e di tutti i campi contenenti informazioni rilevanti per la frammentazione.

Q3. Consideriamo una rete di 20 router che utilizzano l'algoritmo di instradamento *link state* e supponiamo che il nodo R di tale rete abbia 5 vicini. Indicare –giustificando la risposta– quale è il numero minimo di iterazioni dell'algoritmo necessarie al nodo R per determinare i percorsi di costo minimo verso tutti gli altri nodi della rete.

Q4. Supponiamo che un nodo RIP R riceva da un suo vicino V un *advertisement* contenente $D_V(X)=n$ e che quindi riceva, sempre da V, un pacchetto P broadcast originato da X. Indicare –giustificando la risposta– che cosa fa R quando riceve P se R utilizza RPF per instradare pacchetti broadcast.

Seconda parte

E1 (5 punti). Consideriamo una variante del protocollo Go-Back-N in cui la prima volta che si verifica un evento che causa la ritrasmissione dei pacchetti presenti nella finestra, questi vengono ritrasmessi tutti. Se l'evento successivo causa ancora la ritrasmissione dei pacchetti, questi non vengono ritrasmessi tutti, ma solo la prima metà. Se di nuovo il primo evento successivo che si verifica causa la ritrasmissione dei pacchetti, vengono ritrasmessi quelli appartenenti al primo quarto dei pacchetti presenti nella finestra. Si procede in questo modo, dimezzando sempre il numero di pacchetti da ritrasmettere (ma ritrasmettendone sempre almeno uno), finché non si riceve un riscontro positivo. In tal caso, ci si riconduce alla situazione iniziale in cui al prossimo evento che causa ritrasmissione si ritrasmette tutta la finestra, per poi (eventualmente) procedere come sopra specificato. Descrivere, <u>utilizzando un automa a stati finiti</u>, il comportamento del sender della variante del protocollo Go-Back-N sopra descritta.

E2 (5 punti). Supponiamo che il TCP di un processo applicativo A al tempo t0, avendo 2 MSS di dati in volo e 1 MSS di nuovi dati da spedire, riceva per la quarta volta consecutiva un riscontro R. Indicare, giustificando la risposta:

(a) che cosa fa il TCP di A quando riceve R se R contiene 0 come valore nel campo RcvWin, e
(b) quali valori dovrebbero avere CW di A in t0 e R.RcvWin affinché il TCP di A possa inviare 1 MSS di nuovi dati dopo avere ricevuto R.

E3 (4 punti). Decrivere, <u>con uno pseudocodice</u>, il comportamento di un nodo che implementa routing broadcast con flooding controllato con numeri di sequenza quando riceve un datagram IP sulla sua interfaccia I. Supporre che le interfacce del nodo siano memorizzate in un vettore INTs e supporre di avere a disposizione, per ogni campo del preambolo IP, un'operazione *nomeCampo* che restituisce il valore contenuto nel campo *nomeCampo* del pacchetto IP.

E4 (6 punti). Supponiamo che solo due nodi (A e B) di una rete Ethernet debbano trasmettere dati: A deve trasmettere per la prima volta due frame di 64 byte mentre B deve trasmettere, anch'esso per la prima volta, un frame di 64 byte. Supponiamo che al tempo t0 entrambi i nodi inizino a trasmettere simultaneamente per la prima volta (A il primo dei suoi due frame, e B il suo frame). Supponiamo inoltre che la frequenza di trasmissione sia 10 Mbps e che il ritardo di propagazione tra ogni coppia di nodi sia di 0,7 microsecondi. Indicare, giustificando la risposta:

(a) il tempo minimo necessario affinché tutti e tre i frame siano trasmessi con successo, e
(b) la probabilità che tutti e tre i frame siano trasmessi con successo entro 210 microsecondi da t0.

Traccia della soluzione

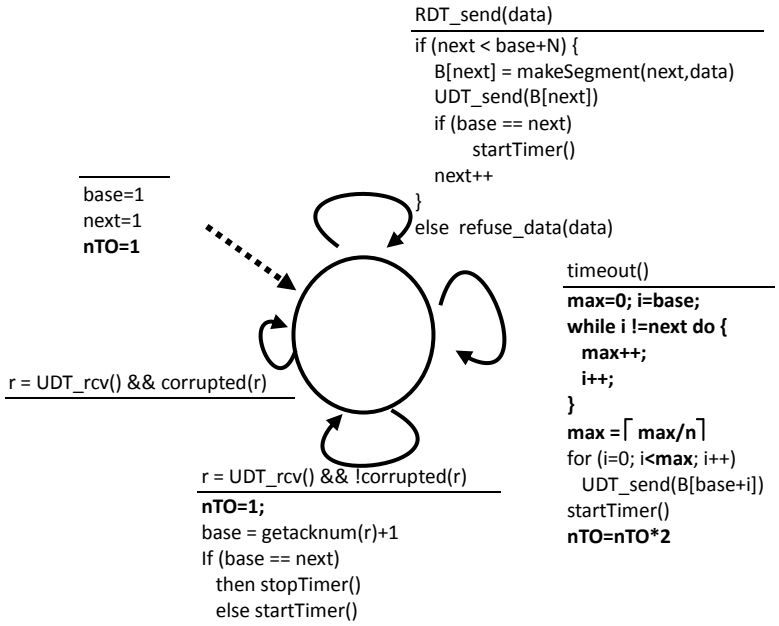
Q1. La lunghezza del timer deve essere maggiore del tempo necessario affinché un host possa ricevere il riscontro di un segmento. Un segmento TCP *full-sized* verrà trasportato in un frame di 1200+20+20+26 byte, dove 20 è la dimensione minima degli header TCP e IP e 26 la dimensione minima di header+trailer Ethernet. Un riscontro TCP che non contiene dati verrà invece trasportato in un frame di 20+20+6+26 byte[1]. La lunghezza del timer dovrà quindi essere almeno maggiore di $\frac{1266\times 8}{10^7}+\frac{72\times 8}{10^7}+2\times\frac{180}{2\times 10^8}$ ovvero di 1072,2μs.

Q2. Dato che il payload del pacchetto IP (con numero di identificazione X) contiene 3980 (=4000-20) byte, il router dovrà creare 3 frammenti (con numero di identificazione X): il primo con *flag*=1, *offset*=0 e *length*=1996[2], il secondo con *flag*=1, *offset*=247 e *length*=1996 e il terzo con *flag*=0, *offset*=494 e *length*=48.

Q3. 19, perché a ogni iterazione viene aggiunto un solo nodo all'insieme dei nodi per i quali è stato determinato il percorso di costo minimo da R.

Q4. Quando R riceve P: se $nextHop_R(X)$=V allora R inoltra P a tutti i suoi vicini, V escluso, altrimenti R scarta P.

E1. Rappresentiamo il buffer di spedizione con un vettore B di dimensione D[3].



E3. Supponiamo che il nodo abbia ricevuto un datagram *p* broadcast sull'Interfaccia *I* e supponiamo che il nodo utilizzi una tabella hash *lastFrom* dove *lastFrom(m)=x* indica che *x* è il numero di identificazione di pacchetto più alto tra tutti i pacchetti generati da *m* che il nodo ha ricevuto.

```
m = p.sourceAddress();
id = m.identification();
if lastFrom(m)==NULL || lastFrom(m)<id {
    for j=0 to INTs.length() {
            "aggiorna TTL di p";
            if INTS[j]!= I
                    send(INTs[j],p);
    }
    lastFrom(m)=id;

}
```

E2. (a) Esegue una ritrasmissione veloce del segmento più vecchio ancora in volo e dimezza il valore di CW. (b) Poiché dopo la ricezione di R il valore di CW viene dimezzato e poiché vi sono ancora 2 MSS di dati in volo, R.RcvWin deve essere almeno 3 MSS e CW in t0 deve essere almeno 6 MSS.

E4. (a) A e B rilevano la collisione dopo 0,7μs e inviano quindi entrambi il jam (4,8μs). Affinché tutti e tre i frame siano trasmessi con successo nel minimo tempo possibile, B riprova subito a spedire e A no. In quel caso infatti B termina la sua spedizione con successo dopo altri (0,7+9,6+51,2)μs e A termina dopo altri (0,7+9,6+51,2+9,6+51,2)μs, ovvero al tempo t0+189,3μs.
(b) Affinché tutti e tre i frame siano trasmessi con successo entro 210μs, non può essere "sprecato" alcuno slot dato che ogni trasmissione richiede almeno 60,8μs per sensing (9,6μs) e trasmissione (51,2μs) e quindi tre trasmissioni richiedono almeno 182,4μs. Inoltre, dato che la prima collisione dopo t0 richiede (0,7+4,8+0,7)μs e ogni successiva collisione richiede (9,60,7+4,8+0,7)μs, possono avvenire al più due collisioni dopo t0 dato che 182,4+6,2+15,8<210<182,4+6,2+2x15,8. I casi possibili sono quindi quattro:

- dopo la prima collisione B rispedisce subito mentre A attende 512 bit time,
- dopo la prima collisione sia B che A cercano di rispedire subito, e dopo la successiva collisione B rispedisce subito mentre A attende 512 bit time,
- dopo la prima collisione A rispedisce subito mentre B attende 512 bit time, e dopo la successiva collisione uno dei due rispedisce subito mentre l'altro attende 512 bit time.

La probabilità che ciò avvenga è: $\frac{1}{2\times 2}+\frac{1}{2\times 2}\times\frac{1}{4\times 4}+\frac{1}{2\times 2}\times\frac{2}{2\times 4}=\frac{21}{64}$.

[1] Il campo dati di un frame Ethernet deve contenere almeno 46 byte.
[2] Dato che max {i|i x 8 ≤1980}=247 e (247x8)+20=1996.
[3] Assumiamo che tutte le operazioni aritmetiche che coinvolgono numeri di sequenza sia effettuate modulo D.