RETI DI CALCOLATORI – sesto appello – 05.09.2013

Per ottenere una valutazione sufficiente dell'intera prova è necessario ottenere una valutazione sufficiente della prima parte.

Prima parte (10 punti)

Q1. Indicare, giustificando la risposta, se in FTP è possibile o meno avere una connessione di controllo senza una connessione dati.

Q2. Indicare, giustificando la risposta, se è possibile che un router riceva più copie di uno stesso pacchetto se viene utilizzato flooding controllato con RPF anziché flooding controllato con numeri di sequenza.

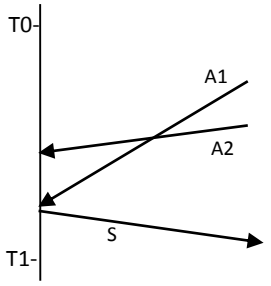
Q3. Un anello Chord utilizza identificatori a 7 bit ed è formato dai nodi con identificatori 3, 6, 33, 58, 61, 73, 81, 93, 100, 115, 120. Indicare, giustificando la risposta, a quali dei suoi finger il nodo 100 inoltra le query relative agli identificatori associati alle chiavi 2, 96 e 119, rispettivamente.

Q4. Indicare, giustificando la risposta, se i protocolli CSMA/CD e CSMA/CA offrono o meno un servizio di trasferimento dati orientato alla connessione.

Seconda parte

E1 (5 punti). Una rete utilizza una variante del protocollo *GoBackN* in cui il receiver non invia un riscontro positivo subito dopo aver ricevuto correttamente il pacchetto atteso, ma invia un riscontro (cumulativo) solamente ogni TR istanti di tempo nel caso in cui abbia ricevuto correttamente almeno un nuovo pacchetto da quando ha inviato l'ultimo riscontro. Se non riceve nessun segmento per TR istanti di tempo, torna ad attendere un tempo uguale a TR. Quando invece riceve un segmento corrotto o con numero di sequenza diverso da quello atteso allora invia subito un riscontro "negativo". Descrivere il comportamento di tale receiver utilizzando un automa a stati finiti.

E2 (5 punti). Supponiamo che al tempo T0 il TCP di un processo applicativo abbia 2MSS di nuovi dati ancora da spedire, 2 MSS di dati in volo, che il valore di *sendBase* sia X, che la dimensione della sua finestra di congestione sia 2MSS e si trovi nello stato di slow start. Indicare – giustificando la risposta – i possibili valori di AckNum e di RcvWin contenuti in A1 e A2.
Per semplicità, supporre che tutti i segmenti contenenti dati contengano 1 MSS di dati, che né A1 né A2 contengano dati e che non scatti nessun timeout nell'intervallo [T0,T1].



E3 (4 punti). Descrivere con uno pseudocodice il modo in cui un router R che utilizza *distance vector* con *poisoned reverse* invia ai suoi vicini le informazioni contenute nel suo vettore D delle distanze. Supporre per semplicità che i nodi della rete siano identificati dai primi N naturali, che R sia identificato da N, che i vicini di R siano identificati dai primi M naturali e che D[x]=<k,y> indichi che la distanza per la destinazione x è k utilizzando come *next hop* y.

E4 (6 punti). Supponiamo che solo due nodi (A e B) di una rete Ethernet debbano trasmettere ciascuno un frame di 64 byte e che al tempo t_0, dopo aver verificato che il canale è inattivo, inizino tutti e due a trasmettere simultaneamente il proprio frame per la prima volta. Supponendo che la frequenza di trasmissione sia di 10 Mbps e che il ritardo di propagazione tra ogni coppia di nodi sia di 0,8 microsecondi, indicare – giustificando la risposta – quale è la probabilità che il primo dei due nodi che riesce a trasmettere con successo un frame riesca a terminare tale trasmissione, subendo il minor numero possibile di collisioni, non prima di t_0+660 microsecondi.

Traccia della soluzione

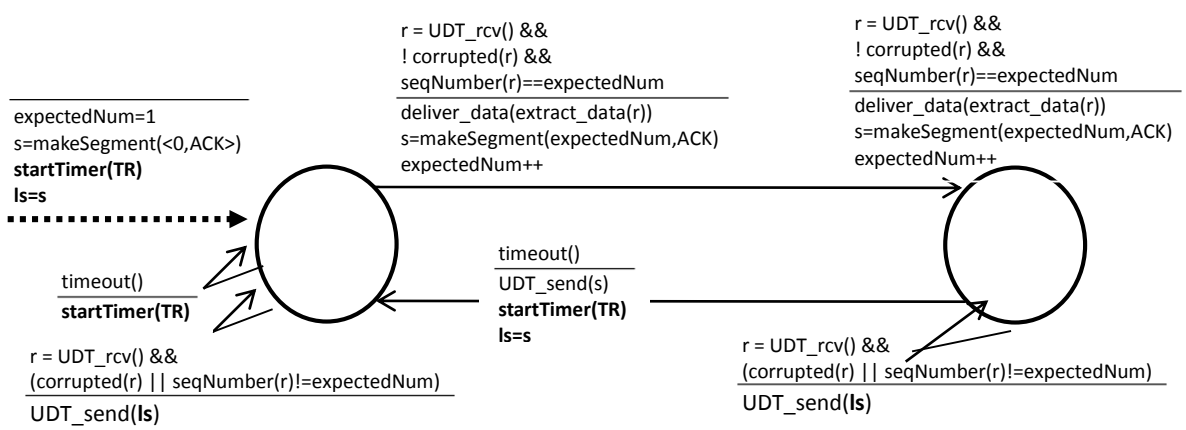
Q1. Si, ciò potrebbe avvenire se sulla connessione di controllo non venisse inviato alcun comando che causa l'apertura di una connessione dati (come i comandi RETR e STOR di richiesta di trasferimento dati).

Q2. Si, ciò può avvenire se per esempio il router aggiorna il valore di nextHop(s) da v1 a v2 dopo avere ricevuto da v1 un pacchetto p generato da s e prima di ricevere p da v2.

Q3. Dato che i finger del nodo 100 sono i nodi 115, 120, 6 e 58, il nodo 100 inoltra le query relative agli identificatori associati alle chiavi 2, 96 e 119 ai finger 120, 58 e 115, rispettivamente.

Q4. Mentre CSMA/CD non è orientato alla connessione, l'utilizzo dei frame di controllo RTS e CTS introduce un meccanismo di hanshake che permette a un trasmittente e a un ricevente di prenotare il canale per la loro connessione, in modo da evitare collisioni durante il trasferimento dei dati.

E1.



E2. Osserviamo che A1.AckNum≤A2.AckNum dato che A1 è stato spedito prima di A2. Osserviamo inoltre che A1.AckNum non può essere minore di A2.AckNum dato che, se così fosse, il valore di A1.RcvWin verrebbe ignorato e il controllo del flusso non permetterebbe di spedire nuovi dati all'arrivo di A1. Pertanto A1.AckNum=A2.AckNum.

- Se A1.AckNum=A2.AckNum=X e A1 è il terzo riscontro duplicato consecutivo ricevuto, allora S viene spedito per il meccasnismo di fast retrnsmit, indipendentemente dal valore di A1.RcvWin e di A2.RcvWin.
- Il caso Se A1.AckNum=A2.AckNum=X+1Mss non può invece verificarsi. A2.RcvWin dovrebbe infatti essere ≤1MSS (altrimenti verrebbero spediti nuovi dati a seguto della ricezione di A2) mentra A1.RcvWin dovrebbe essere ≥2MSS, ma ciò non può avvenire dato che il pari non potrebbe riscontrare nuovi dati con A2.
- Il caso Se A1.AckNum=A2.AckNum=X+2Mss non può verificarsi per lo stesso motivo.

E3.

```
// utilizziamo un vettore E di N naturali per spedire le distanze ai vicini
for (i=0; i<M; i++) do {
      for (j=0; j<N; j++) do
            if D[j]==<k,i>
                  then E[j]= maxInt
                  else E[j]= k;
      send(i,E)
  }
```

E4. Osserviamo che se dopo ciascuna delle prime tre collisioni entrambi i nodi attendono il massimo possibile, e dopo la quarta[1] collisione un nodo spedisce mentre l'altro attende, allora il primo nodo può terminare con successo la trasmissione del frame dopo (0,8 + 4,8 + 1x51,2 + 9,6 + 0,8 + 4,8 + 3x51,2 + 9,6 + 0,8 + 4,8 + 7x51,2 + 9,6 + 0,8 + 4,8 + Kx51,2 + 9,6 + 51,2) μs = (624 + Kx51,2) μs, con K∈[0,14]. La probabilità che ciò avvenga è: $\left(\frac{1}{2}\times\frac{1}{2}\right)\times\left(\frac{1}{2^2}\times\frac{1}{2^2}\right)\times\left(\frac{1}{2^3}\times\frac{1}{2^3}\right)\times\left(1-\frac{16}{2^4\times 2^4}\right)=\frac{1}{2^2}\times\frac{1}{2^4}\times\frac{1}{2^6}\times\frac{15}{2^4}=\frac{15}{2^{16}}$.

[1] Nello scenario in questione il numero minimo di collisioni è 4. Se infatti dopo ciascuna delle prime due collisioni entrambi i nodi attendono il massimo possibile, e dopo la terza collisione un nodo spedisce dopo avere atteso 7x51,2 μs (mentre l'altro attende 8x51,2 μs), allora il primo nodo riesce a terminare con succcesso la trasmissione del frame già dopo (0,8 + 4,8 + 1x51,2 + 9,6 + 0,8 + 4,8 + 3x51,2 + 9,6 + 0,8 + 4,8 + 6x51,2 + 9,6 + 51,2) μs = 608,8 μs.