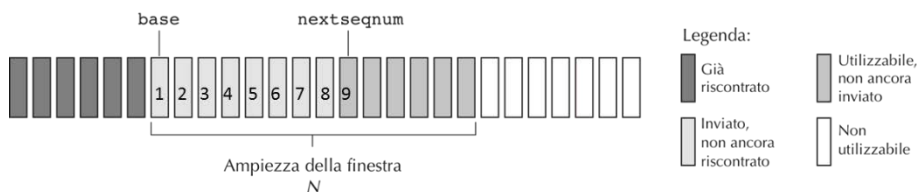


Esame di Reti di Elaboratori – Parte 2
del 5 Settembre 2019

Quesito n. 1 (punti 15)

Descrivere in breve il funzionamento di protocolli di tipo go-back-N e di tipo selective repet evidenziandone similitudini e differenze.

Dato la situazione evidenziata dalla seguente figura

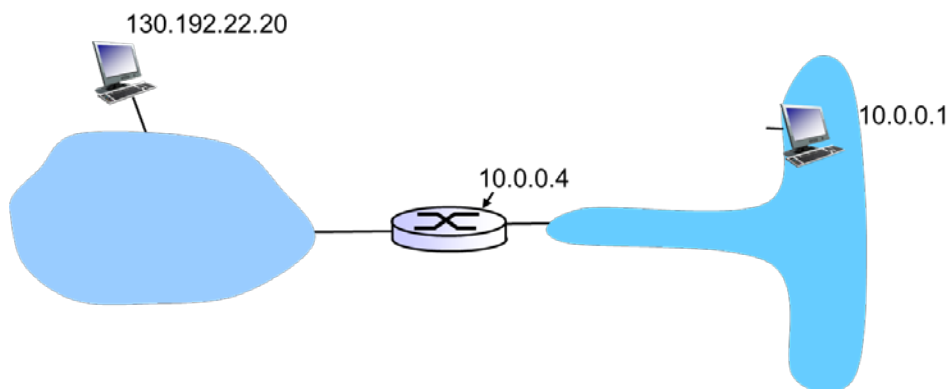


Assumere che il destinatario abbia ricevuto **tutti** i pacchetti dal n. 1 al n. 8 ed abbia inviato **tutti** i riscontri di tali pacchetti ma nel percorso tra destinatario e mittente il riscontro del pacchetto n. 5 sia andato perso (solo quello mentre tutti gli altri dal n. 1 al n. 8 sono stati ricevuti dal mittente).

- Come si comporta in questo caso un protocollo di tipo go-back-N ?
- E uno di tipo selective-repeat ?

Quesito n. 2 (punti 10)

Si consideri lo scenario illustrato nella figura sottostante



e si assuma che l'host con indirizzo 10.0.0.1 abbia le seguenti informazioni:

Indirizzo IP 10.0.0.1

MAC address 0E:20:11:A1:00:9B

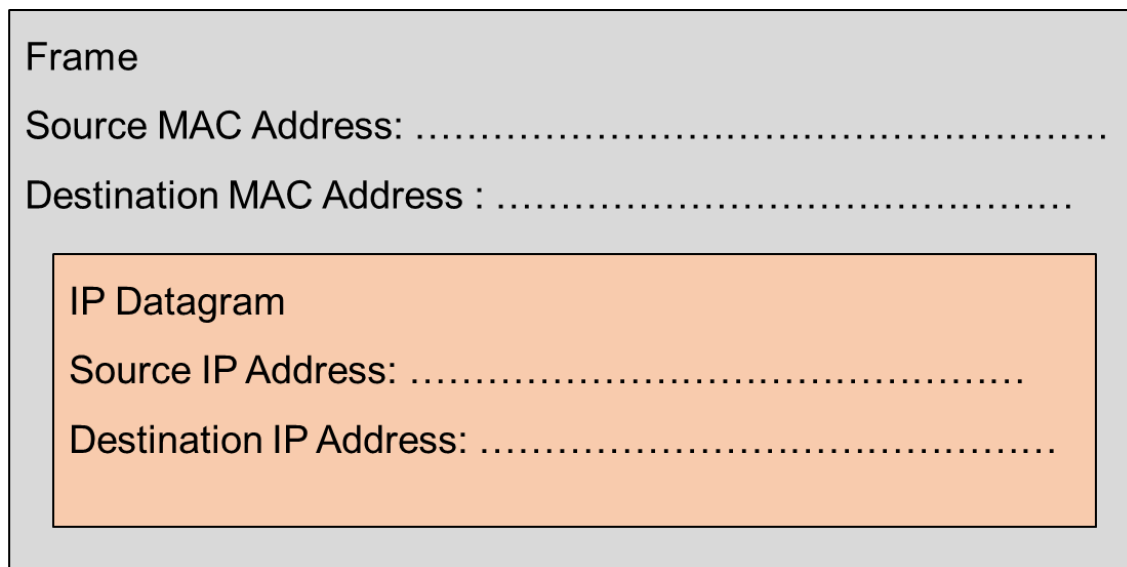
Default gateway: 10.0.0.4

si assuma inoltre che nella tabella ARP di questo host sia presente questa associazione

| IP ADDRESS | Physical ADDRESS |
|------------|-------------------|
| 10.0.0.4 | 01:00:5E:54:1A:0F |

L'host 10.0.0.1 spedisce un datagram all'host 130.192.22.20

determinare gli indirizzi sorgente e destinazione del datagram e gli indirizzi sorgente e destinazione del frame che incapsula tale datagram



Quesito n. 3 (punti 10)

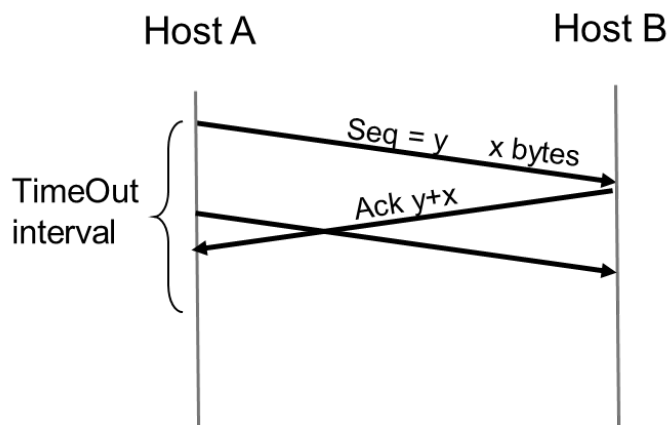
A che cosa serve il comando **tracert** (in ambiente Windows **tracert**)?
Come viene implementato usando ICMP e IP?

Quesito n. 4 (punti 15)

L'host A e B stanno comunicando tramite una connessione TCP e l'host B ha già ricevuto da A tutti i byte fino al byte 126. Supponiamo quindi che l'host A invii, uno dopo l'altro, due segmenti all'host B. Il primo segmento contiene 80 byte di dati mentre il secondo ne contiene 40 byte. Nel primo segmento, il numero di sequenza è 127, il numero di porta di origine è 302 e il numero di porta di destinazione è 80. Assumete che l'host B invia ad A un ACK ogni volta che riceve un segmento. Rispondere ai seguenti quesiti:

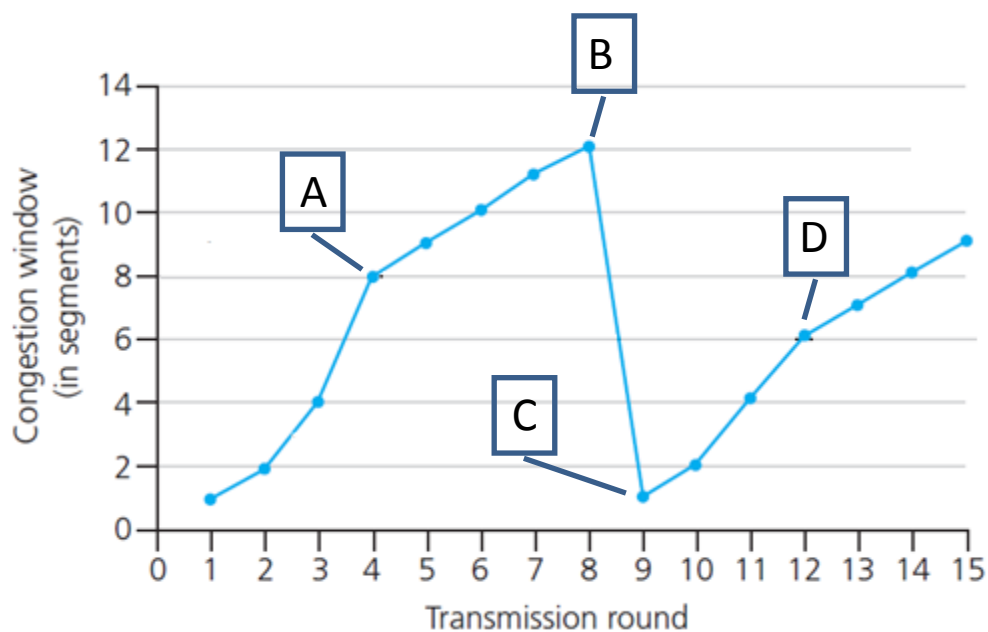
- 1) Nel secondo segmento inviato dall'host da A a B, indicare il valore del sequence number, del numero di porta di sorgente, e del numero di porta di destinazione?
- 2) Se il primo segmento arriva prima del secondo, nell'ACK del primo segmento arrivato, indicare quale è l'acknowledgment number, il numero di porta di sorgente, e il numero di porta di destinazione
- 3) Se il secondo segmento arriva prima del primo (segmento), nel primo ACK inviato da B (nota: si intende il primo ACK inviato da B) quel è l'acknowledgment number contenuto?
- 4) Supponete che i due segmenti inviati da A arrivino all'host B in ordine. Il primo ACK inviato da B viene perso mentre il secondo ACK arriva dopo il primo intervallo di timeout. Disegnate un diagramma temporale (del tipo indicato in figura) mostrando i segmenti e gli ACK inviati. Per ogni segmento fornire il numero di sequenza ed il numero di byte inviati. Per ogni ACK includere nella figura l'acknowledgment number.

Esempio di diagramma temporale



Quesito n. 5 (punti 10)

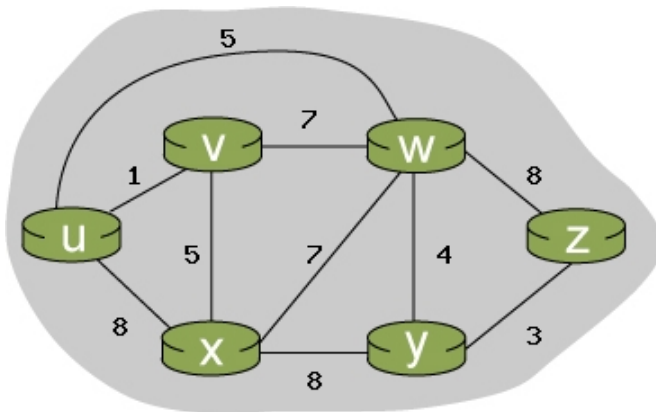
TCP e gestione della congestione: supponete che l'evoluzione della finestra di TCP versione Reno lato mittente sia evidenziata dal seguente diagramma



Spiegare quali sono i fenomeni che si sono verificati in corrispondenza dei round evidenziati dalle etichette A, B, C e D.

Quesito n. 6 (punti 15)

Considerare una rete rappresentata mediante un grafo

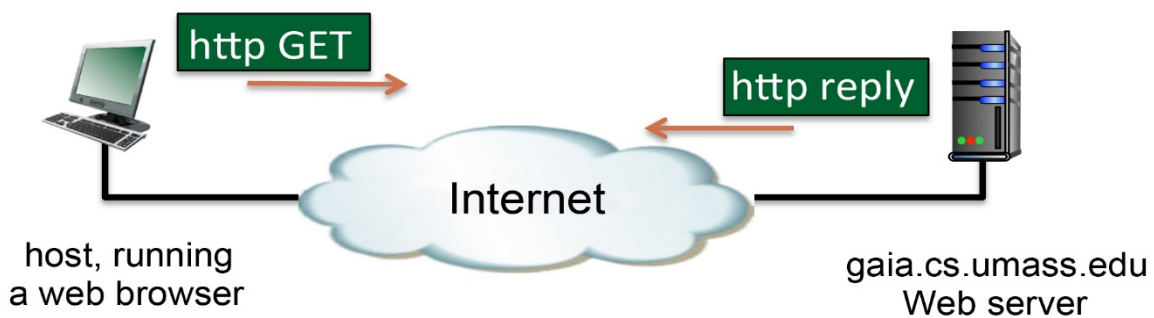


Utilizzando l'algoritmo di Dijkstra

- 1) determinare i percorsi di costo minimo tra il nodo **u** e tutte le altre destinazioni;
- 2) fornire la tabella di forwarding del nodo **u**.

Quesito n. 7 (punti 10)

Considerare la figura seguente, in cui un client http invia un messaggio GET HTTP a un server Web all'indirizzo `gaia.cs.umass.edu` ed il server invia una http REPLY



Supponete che il client ottenga il seguente messaggio (http reply):

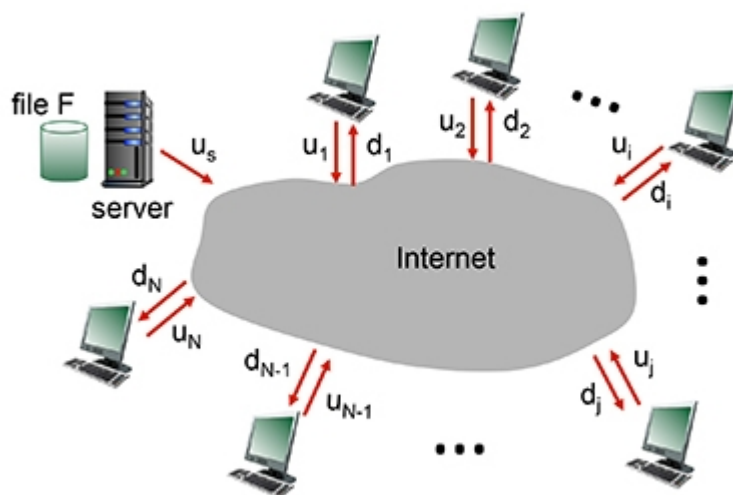
```
HTTP/1.1 200 OK
Date: Mon, 02 Sep 2019 20:05:30 +0000
Server: Apache/2.2.3 (CentOS)
Last-Modified: Mon, 02 Sep 2019 20:30:30 +0000
ETag:17dc6-a5c-bf716880.
Content-Length: 4467
Keep-Alive: timeout=50, max=81
Connection: Keep-alive
Content-type: text/html
```

Rispondere alle seguenti domande e per ogni risposta spiegare in quale riga si trova l'informazione corrispondente:

- 1) Il messaggio di risposta utilizza HTTP 1.0 o HTTP 1.1?
 - 2) Il server è stato in grado di inviare correttamente il documento?
 - 3) In quale data e ora è stata inviata questa risposta?
 - 4) Quando è stato modificato il file l'ultima volta sul server?
 - 5) Quanti byte ci sono nel documento restituito dal server?
 - 6) Qual è la modalità di connessione predefinita per il protocollo HTTP? La connessione nella risposta è persistente o non persistente?
 - 7) Qual è il tipo di file inviato dal server in risposta?
 - 8) Il messaggio di risposta utilizza messaggi keep-alive separati?
 - 9) Qual è il nome del server e la sua versione?
 - 10) Qual è il valore di timeout per il messaggio di risposta?
-
-

Quesito n. 8 (punti 15)

Distribuzione di un file che inizialmente si trova su un server: confronto tra un'architettura di tipo client-server ed una di tipo peer-to-peer



Il problema è distribuire un file di dimensioni pari a 5 Gbit (denotata con F) a 6 differenti peer. Supponete che il server abbia una velocità di upload di $u = 56$ Mbps e che i 6 peer abbiano velocità di upload $u_i = 11$ Mbps velocità di download $d_i = 20$ Mbps, per $i = 1, 2, \dots, 6$ (le velocità di upload e download dei peer sono uguali). Rispondere alle seguenti domande:

- 1) Qual è il tempo minimo necessario per distribuire questo file dal server centrale ai 6 peer usando il modello client-server?
- 2) Per la domanda 1, qual è il motivo di questo tempo minimo specifico? La velocità di upload del server o la velocità di download dei client specifico?

- 3) Qual è il tempo minimo necessario per distribuire questo file usando il download peer-to-peer?
- 4) In riferimento alla domanda 3, qual è il motivo di questo tempo minimo specifico: la velocità di upload del server, la velocità di download dei client (essendo tutte uguali)? Oppure la somma delle velocità di upload del server e dei peer?