

Domande sulla teoria (4 punti ciascuna)

Lo studente risponda in maniera concisa, ma precisa, alle seguenti domande riguardanti la parte teorica. E' necessario che lo studente ottenga almeno 7 punti (su un totale di 12 punti a disposizione). In caso contrario, gli esercizi non verranno considerati e il voto finale sarà insufficiente.

1. Si spieghi la differenza tra commutazione di circuito e commutazione di pacchetto, sottolineando aspetti positivi e negativi delle due tecniche.
2. In riferimento al livello di rete, si spieghi che cosa succede quando un host si connette ad una rete ed ha bisogno di ricevere un indirizzo IP (non è necessario andare nei dettagli dei protocolli, è sufficiente descrivere a grandi linee i messaggi scambiati). Si specifichi inoltre quali altre informazioni vengono fornite all'host oltre all'indirizzo IP.
3. Si descriva la fase di chiusura della connessione nel TCP, indicando i messaggi scambiati e i principali campi dell'header utilizzati durante tale fase. Si specifichino le diverse sequenze di messaggi in base a chi inizia la chiusura.

1. La commutazione di circuito è l'allocazione di un circuito fisico per la comunicazione tra due host.

Vantaggi:

- Risorse dedicate ad ogni comunicazione
- Ritardo deterministico

Svantaggi:

- Vengono sprecate risorse perchè quando le parti non hanno nulla da trasmettere il canale rimane riservato e inutilizzato.

La commutazione di pacchetto non alloca alcuna risorsa, ma divide il messaggio in pacchetti a cui viene aggiunto un header utile per ricostruire il messaggio e per permettere la consegna del pacchetto. Questi pacchetti poi vengono memorizzati all'interno di un buffer e poi inviati al destinatario.

Vantaggi:

- Le risorse non vengono sprecate perchè vengono utilizzate soltanto quando gli host hanno qualcosa da trasmettere
- Si può utilizzare lo stesso canale per trasmettere più pacchetti di utenti diversi

Svantaggi:

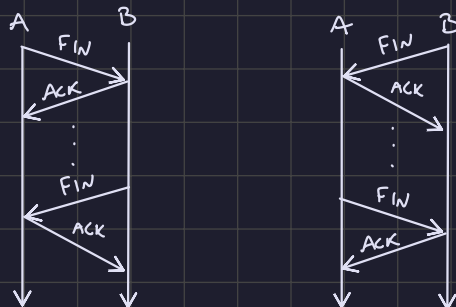
- Più ritardo
- I pacchetti possono essere persi

2. Quando un host si unisce ad una rete richiede un IP al server DHCP (che spesso si trova nel router) tramite una DHCP discover inviata in broadcast che contiene come sorgente l'IP 0.0.0.0. Il server risponderà, sempre in broadcast, con una DHCP offer in cui comunica il proprio IP e l'IP da assegnare all'host che lo ha richiesto. L'host una volta ricevuto questo messaggio, come misura di sicurezza, dovrà inviare un altro messaggio DHCP request in cui inserisce l'IP del server DHCP, e l'IP che gli è stato offerto, questo viene fatto perchè un altro dispositivo potrebbe impersonare il server DHCP. Infine il server DHCP risponderà con un DHCP acknowledge e dopo che l'host riceve questo ultimo messaggio potrà utilizzare l'IP che gli è stato assegnato. Ogni IP assegnato dal DHCP ha un tempo di validità (tempo di lease) che dovrà essere rinnovato. Per distinguere le richieste di più utenti è anche presente un ID (transaction ID) che identifica la singola richiesta. Il server DHCP oltre all'IP da assegnare all'host comunica anche la maschera della rete, l'IP del router gateway e l'IP del server DNS.

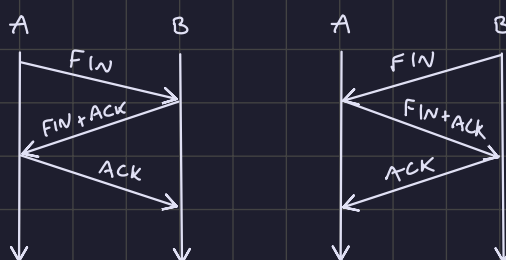
3. Per chiudere una connessione TCP bisogna dichiararlo attraverso uno scambio di messaggi che dovrà essere effettuato da entrambe le direzioni della comunicazione. I principali campi utilizzati sono:
Flag: Per interrompere la connessione viene settato a 1 il flag FIN, mentre il flag ACK viene messo a 1 per inviare il riscontro.

La connessione si può interrompere in 2 modi:

Da un lato alla volta: La connessione viene chiusa soltanto da una direzione perchè ha finito di trasmettere, ma rimane comunque in ascolto perchè l'altra direzione può comunque continuare a comunicare finchè non interrompe la trasmissione.

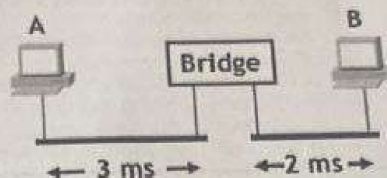


Da entrambi i lati: La connessione può essere direttamente interrotta da entrambi i lati contemporaneamente e questo viene fatto settando entrambi i flag FIN e ACK a 1.



Esercizio 1 (7 punti)

Un Bridge è attestato contemporaneamente su due segmenti distinti di rete; sul segmento 1 c'è una stazione, A, e sul segmento 2 c'è una stazione, B (si veda la figura a fianco). Il Bridge è un particolare tipo di stazione che memorizza ciascuna trama che arriva da un segmento di rete e, una volta ricevuta completamente, la ritrasmette sull'altro segmento di rete (tale comportamento è valido, in modo indipendente l'uno dall'altro, in entrambi i sensi); le trame restano in memoria del Bridge fino a quando la trasmissione sull'altro segmento non è andata a buon fine.



Le stazioni e il Bridge utilizzano un protocollo **CSMA persistent**. Le caratteristiche del sistema sono:

- velocità dei segmenti: 800 Kbit/s;
- lunghezza delle trame generate dalle stazioni: 1200 byte;
- ritardo di propagazione pari ad 3 ms tra la stazione A e il bridge e pari a 2 ms tra la stazione B e il bridge;

Le stazioni generano le seguenti trame:

- stazione A: una trama (A1) all'istante $t_{A1}=645$ msec, e una trama (A2) all'istante $t_{A2}=665$ msec, entrambe dirette a B;
- stazione B: una trama (B1) all'istante $t_{B1}=639$ msec, una trama (B2) all'istante $t_{B2}=659$ msec, e una trama (B3) all'istante $t_{B3}=689$ msec, tutte dirette ad A.

In caso di collisione, si supponga che le stazioni decidono di ritrasmettere Z millisecondi dopo la fine della trasmissione della trama corrotta; il numero Z viene deciso secondo il seguente metodo:

- si attende un tempo pari a $Z = S_c \cdot N + T$, dove
 - S_c = somma delle cifre che compongono l'istante di inizio trasmissione
 - N = numero di collisioni subite da quella trama
 - T tempo di trama

Ad esempio, se l'istante di inizio trasmissione è 418 msec, $Z = (4+1+8) \cdot N + T$

1. graficamente le trasmissioni delle diverse trame, indicando se avviene collisione, in quali istanti e da quali apparati;
2. il periodo di vulnerabilità del sistema preso in considerazione.

$$v = 800 \text{ Kbit/s}$$

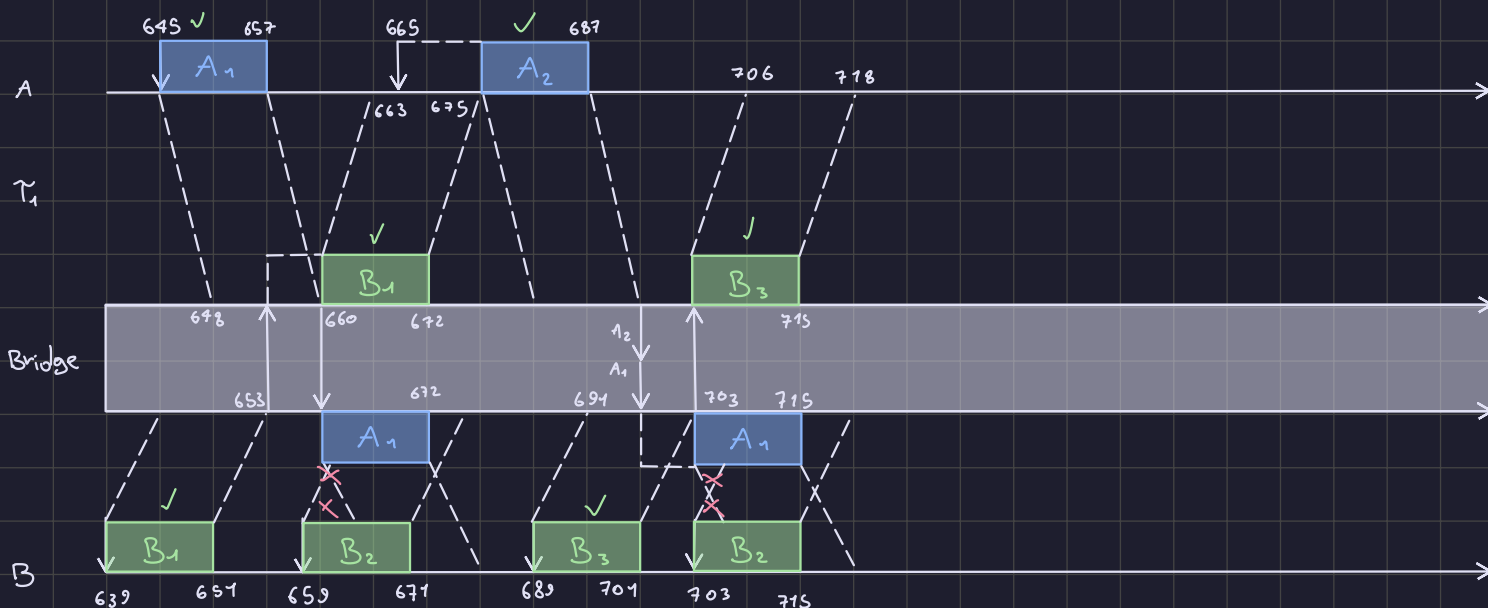
$$L = 1200 \text{ byte}$$

$$\tau_1 = 3 \text{ ms} \quad \tau_2 = 2 \text{ ms}$$

$$A: \begin{cases} t_{A1} = 645 \text{ ms} \rightarrow B \\ t_{A2} = 665 \text{ ms} \rightarrow B \end{cases}$$

$$B: \begin{cases} t_{B1} = 639 \text{ ms} \rightarrow A \\ t_{B2} = 659 \text{ ms} \rightarrow A \\ t_{B3} = 689 \text{ ms} \rightarrow A \end{cases}$$

$$T = \frac{L}{v} = \frac{1200 \cdot 8}{8000} = 12 \text{ ms}$$



Alla collisione tra A1 e B2:

$$Z_{A1} = (6+6+0) \cdot 1 + 12 = 24 \text{ ms} \rightarrow 672 + 24 = 696 \text{ ms} \quad A_1 \text{ ritrasmette per primo}$$

$$Z_{B2} = (6+5+9) \cdot 1 + 12 = 32 \text{ ms} \rightarrow 671 + 32 = 703 \text{ ms}$$

Alla seconda collisione tra A1 e B2:

$$Z_{A1} = (7+0+3) \cdot 2 + 12 = 32 \text{ ms} \rightarrow 715 + 32 = 747 \text{ ms}$$

$$Z_{B2} = (7+0+3) \cdot 2 + 12 = 32 \text{ ms} \rightarrow 715 + 32 = 747 \text{ ms}$$


Bruh va in loop

Esercizio 2 (7 punti)
 Si consideri la rete rappresentata in Figura, collegata ad Internet attraverso il router C (router di default per la rete). Si hanno i seguenti vincoli:

- la LAN 2 contiene un host con indirizzo 130.104.213.12;
- Le LAN 1, 2, e 3 devono poter contenere rispettivamente almeno 520, 300, e 1200 host.

In base ai suddetti vincoli:

1. Si specifichi il blocco CIDR più piccolo da assegnare alla rete;
2. Si assegnino gli indirizzi di rete e di broadcast alle LAN 1, 2, e 3, utilizzando il blocco CIDR individuato nel punto precedente.
3. Si scriva la tabella di routing del router A, considerando come metrica il numero di hop e assumendo che il router C abbia annunciato di poter raggiungere qualsiasi host su Internet in 4 hop.



$$Lan1: 520 \rightarrow 1024 \rightarrow 2^{10} \text{ host}$$

$$Lan2: 300 \rightarrow 512 \rightarrow 2^9 \text{ host}$$

$$Lan3: 1200 \rightarrow 2048 \rightarrow 2^{11} \text{ host}$$

1.

Il blocco CIDR totale è composto dalla potenza di 2 che contiene la somma di tutti gli host di tutte le lan in base 2:

$$1024 + 512 + 2048 = 3584 \rightarrow 4096 = 2^{12} \text{ host}$$

Ci viene dato un indirizzo appartenente alla lan 2 da cui possiamo derivare l'indirizzo di rete:

130 . 104 . 213 . 12

↓

130 . 104 . 1 1 0 1 0 1 0 1 0 0 0 0 1 1 0 0

Sappiamo che il blocco CIDR totale ha bisogno di 12 bit di suffisso, quindi 20 bit di prefisso. Un indirizzo di rete ha tutti i bit di suffisso a 0, mentre un indirizzo di broadcast li ha tutti a 1.

Ret: 130 . 104 . 1 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0

Prefisso

↓

130 . 104 . 208 . 0/20

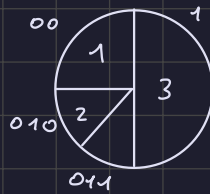
Broadcast: 130 . 104 . 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1

Prefisso

↓

130 . 104 . 223 . 255/20

2.



Lan 2:

Ci viene dato un indirizzo della lan 2 da cui possiamo derivare la sottorete a cui appartiene. La lan 2 ha bisogno di 9 bit di suffisso, quindi 23 di prefisso, di cui 3 saranno dedicati alla sottorete.

Rete: 130 . 104 . 1 1 0 1 0 1 0 0 0 0 0 0 0 0 0
 Prefisso Sottorete



130 . 104 . 212 . 0 / 23

Broadcast: 130 . 104 . 1 1 0 1 0 1 0 1 1 1 1 1 1 1 1
 Prefisso Sottorete



130 . 104 . 213 . 255 / 23

Lan 1:

La lan 1 ha bisogno di 10 bit di suffisso, quindi 22 di prefisso, di cui 2 saranno dedicati alla sottorete.

Rete: 130 . 104 . 1 1 0 1 0 0 0 0 0 0 0 0 0 0
 Prefisso Sottorete



130 . 104 . 208 . 0 / 22

Broadcast: 130 . 104 . 1 1 0 1 0 0 1 1 1 1 1 1 1 1
 Prefisso Sottorete



130 . 104 . 211 . 255 / 22

Lan 3:

La lan 3 ha bisogno di 11 bit di suffisso, quindi 21 di prefisso, di cui 1 sarà dedicato alla sottorete.

Rete: 130 . 104 . 1 1 0 1 1 0 0 0 0 0 0 0 0
 Prefisso Sottorete



130 . 104 . 216 . 0 / 21

Broadcast: 130 . 104 . 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1

Preciso

Sottrarre



130 . 104 . 223 . 255 / 21

3.

A:	dst	next hop	cost
	LAN 1	A	1
	LAN 2	B	1
	LAN 3	A	1
	Internet	C	5
	B	B	1
	C	C	1
	A	A	1

Il costo per arrivare a C è 1, quindi quando C comunica di arrivare ad ogni host su internet in 4 hop A aggiorna la tabella tenendo in considerazione che per arrivare a internet deve passare per C, quindi il costo diventa $4+1=5$

Esercizio 3 (7 punti)

Un'applicazione A deve trasferire 96800 byte all'applicazione B utilizzando il protocollo TCP. Si supponga che la connessione tra A e B sia già stata instaurata. La trasmissione dei segmenti inizia al tempo $t=0$. Sono noti i seguenti parametri:

- MSS concordata pari a 1100 byte;
- RCVWND annunciata da B ad A pari a 17600 byte; a partire dal tempo $t_a > 5.0$ la destinazione annuncia una RCVWND pari a 3300 byte; a partire dal tempo $t_b > 10.0$ la destinazione annuncia una RCVWND pari a 17600 byte;
- SSTHRESH iniziale = RCVWND;
- CWND = 1 segmento a $t=0$;
- RTT pari a 1.0 secondo, costante per tutto il tempo di trasferimento;
- RTO base = $2 \cdot \text{RTT}$; nel caso di perdite consecutive dello stesso segmento, i timeout seguenti raddoppiano fino ad un massimo di 4 volte il RTO base, dopodiché la connessione viene abbattuta;
- il tempo di trasmissione dei segmenti è trascurabile rispetto RTT;
- il ricevitore riscontra immediatamente i segmenti.

Inoltre si supponga che la rete vada fuori servizio nei seguenti intervalli di tempo:

- da $t_1 = 5.5s$ a $t_2 = 6.5s$;
- da $t_3 = 16.5s$ a $t_4 = 18.0s$.

Si tracci l'andamento della CWND nel tempo e si determini in particolare:

- il valore finale di CWND (sia graficamente, sia esplicitandolo);
- i valori assunti dalla SSTHRESH durante il trasferimento (graficamente);
- il tempo necessario per il trasferimento dei dati (sia graficamente, sia esplicitandolo);
- il numero di segmenti trasmessi ad ogni intervallo, specificando se ne vengono ricevuti i riscontri o meno (sia graficamente, sia esplicitando i valori).

$$L = 96800 \text{ byte} = \frac{96800}{1100} = 88 \text{ segmenti}$$

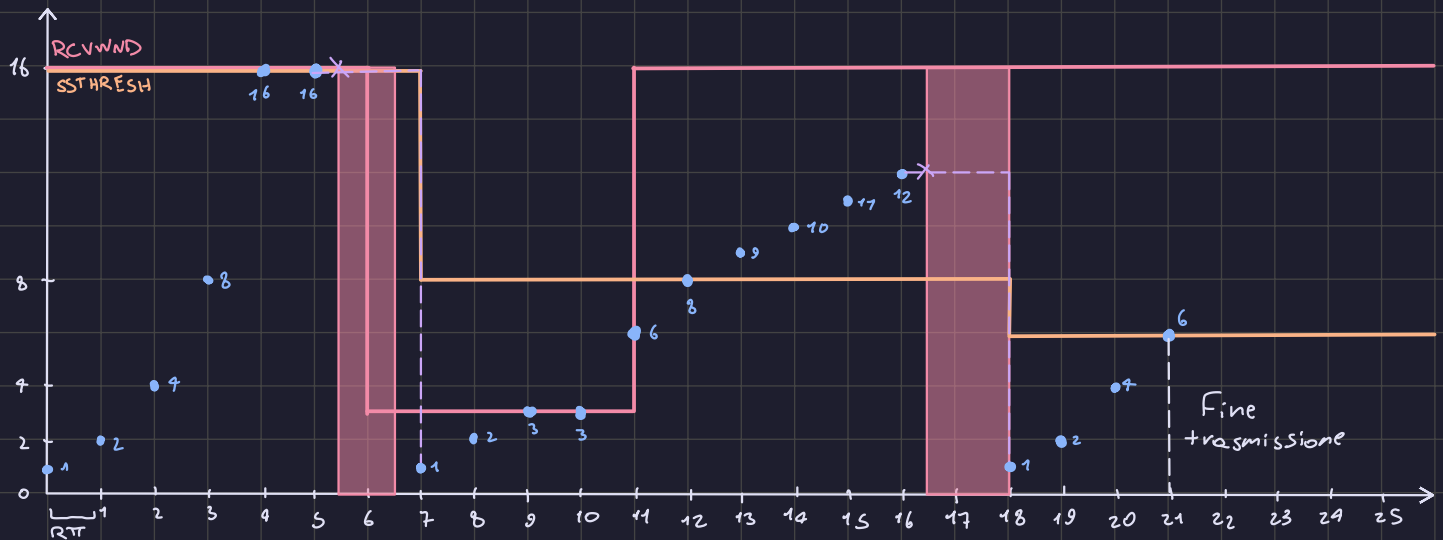
$$\text{MSS} = 1100 \text{ byte}$$

$$\text{RCVWND} = \begin{cases} 17600 \text{ byte} & t \leq 5 & = 16 \text{ segmenti} \\ 3300 \text{ byte} & t_a > 5 & = 3 \text{ segmenti} \\ 17600 \text{ byte} & t_b > 10 & = 16 \text{ segmenti} \end{cases}$$

$$\text{SSTHRESH} = \text{RCVWND}$$

$$\text{RTT} = 1s \text{ costante}$$

$$\text{Down} = \begin{cases} (5.5, 6.5) \\ (16.5, 18) \end{cases}$$



$$\# \text{segmenti trasmessi} = 1 + 2 + 4 + 8 + 16 + 16 + 1 + 2 + 3 + 3 + 6 + 8 + 9 + 10 + 11 + 12 + 1 + 2 + 4 = 88$$

Il tempo necessario a trasmettere tutti i segmenti è 21 secondi e la cwnd finale vale 6.

A $t=11$ cwnd diventa 6 perchè $cwnd = \min(sssthresh, cwnd_old + \#ack) = \min(8, 3 + 3) = 6$

A $t=21$ cwnd diventa 6 perchè $cwnd = \min(sssthresh, cwnd_old + \#ack) = \min(6, 4 + 4) = 6$