

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 scriva lo pseudo-codice, corredato da commenti, dell'algoritmo CSMA nella variante Collision Detection (CSMA-CD). Si indichi inoltre il motivo che ha portato all'introduzione di tale variante.
2. In riferimento al livello di rete, si spieghi che cos'è il Network Address Translation (NAT), mostrando un esempio con le informazioni rilevanti. Si specifichi inoltre per quale motivo tale funzionalità è stata introdotta.
3. Si descriva la modalità di instaurazione di una connessione TCP, specificando i messaggi scambiati e i campi più significativi dell'header utilizzati durante tale fase (con valori di esempio assegnati a tali campi).

1. L'algoritmo CSMA (Carrier Sense Multiple Access) è un algoritmo che consiste nell'ascoltare il canale prima di trasmettere una trama, in generale se il canale è libero si può trasmettere, altrimenti si aspetta finché non si libera. Questo algoritmo ha più varianti:

- CSMA Persistent
- CSMA P-Persistent
- CSMA Non Persistent
- CSMA-CD (Collision Detection)
- CSMA-CA (Collision Avoidance)

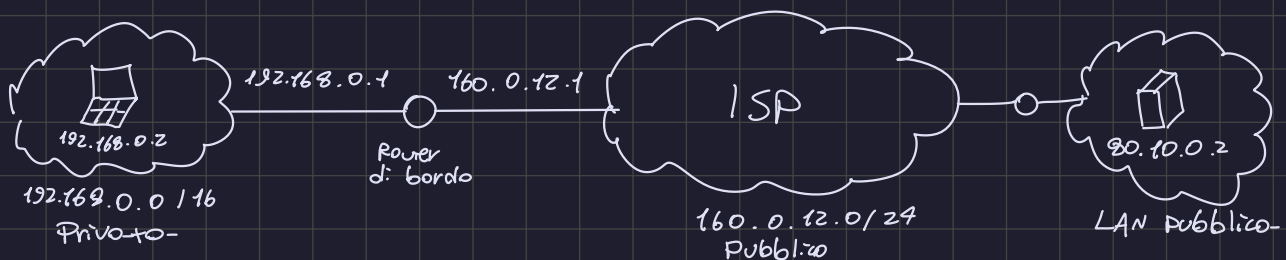
La variante CSMA-CD ha il seguente pseudocodice:

```
while !data.is_empty() // Se ha una trama da trasmettere
  if channel.is_free() // Controlla se il canale è libero
    send(data) // Se è libero trasmette la trama
  else
    disconnect() // Interrompe la connessione
    wait(rand()) // Aspetta un tempo casuale
```

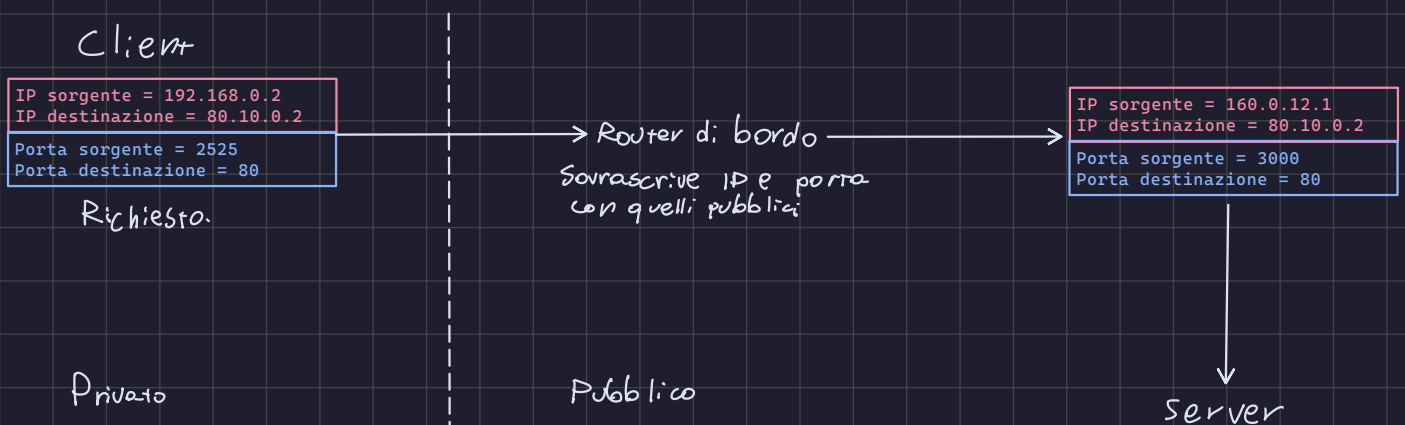
Questa variante è stata introdotta per non mantenere una stazione collegata al mezzo condiviso mentre aspetta che il canale si liberi. Questa variante è utilizzata nelle reti ethernet e può essere implementata anche in combinazione con tutte le altre varianti.

2. Il NAT è un meccanismo che "traduce" gli indirizzi IP privati in indirizzi IP pubblici per poter comunicare con reti esterne a quella locale. La traduzione viene effettuata dal router di bordo della rete sfruttando il fatto che ogni router deve avere al minimo 2 interfacce che si affacciano su reti diverse. Quando un host della rete privata vuole comunicare con un host della rete pubblica il router di bordo sostituisce il suo IP privato con quello pubblico dell'interfaccia del router che dà sull'ISP e il router salva questi valori all'interno di una tabella chiamata tabella NAT. Se più utenti della stessa rete privata vogliono comunicare con lo stesso host della rete pubblica possono nascere conflitti perché non si riuscirebbe più a distinguere a che host bisogna inviare i pacchetti ricevuti, quindi per risolvere questo problema si salva nella tabella anche il valore della porta sorgente e della porta destinazione associati alla nuova porta da sostituire per distinguere a che host vanno inviati i pacchetti ricevuti. Quando si parla di NAT però spesso si fa riferimento al NAT (Network Address & Port Translation) che utilizza anche le porte. Tutto questo meccanismo è trasparente dal punto di vista dell'utente. Un esempio è il seguente:

Un client di una rete privata vuole comunicare con un server di una rete pubblica:



Il NAT funziona nel seguente modo:



IP sorgente = 80.10.0.2
IP destinazione = 192.168.0.2
Porta sorgente = 80
Porta destinazione = 2525

Risposto.

Router di bordo
Ripristino IP e
porta originali:

IP sorgente = 80.10.0.2
IP destinazione = 160.0.12.2
Porta sorgente = 80
Porta destinazione = 3000

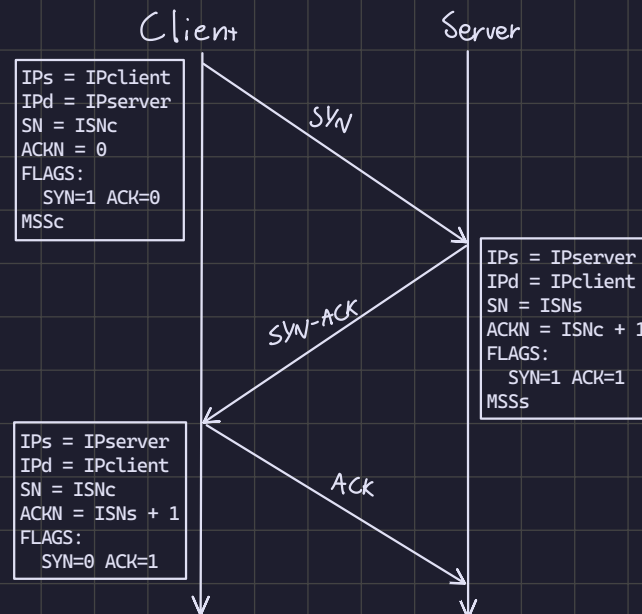
Genera una
risposta.

Il NAT è stato introdotto per separare gli indirizzi IP in due categorie: indirizzi IP privati e indirizzi IP pubblici. Questa distinzione serve per sopperire alla mancanza di indirizzi, infatti il numero di indirizzi è limitato e non basta per gestire tutti i dispositivi di tutto il mondo, di conseguenza per evitare di cambiare tutti i router è stata introdotta questa distinzione per permettere di fare cambiamenti soltanto alle reti locali. L'idea principale è quella di riservare gli indirizzi pubblici soltanto ai router di bordo e assegnare gli indirizzi privati più volte in reti diverse.

3. Il protocollo TCP è un protocollo connection oriented, cioè prima di scambiare dei dati bisogna instaurare una connessione e per farlo si inviano dei messaggi che utilizzano principalmente i seguenti campi dell'header:

- Flag:
 - SYN: Il flag syn viene settato a 1 per indicare di voler instaurare una connessione
 - ACK: Il flag ack viene settato a 1 per inviare il riscontro di un messaggio ricevuto
- IP destinazione/sorgente
- Sequence number
- Acknowledge number
- MSS

Lo scambio di messaggi è il seguente:

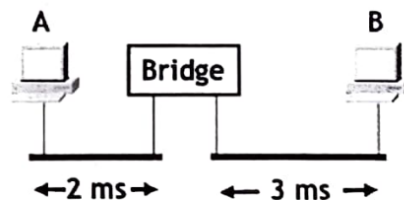


Il client invia un messaggio con il flag syn a 1 al server indicandogli che vuole instaurare una connessione, il server la riceve e risponde con un riscontro, per non inviare più messaggi diversi il server mette nello stesso messaggio anche l'intenzione di avviare la connessione anche nella direzione server-client quindi setta sia ack che syn a 1. Quando il client riceve il messaggio del server che gli comunica che anche lui vuole instaurare una connessione il client gli manda il riscontro.

I primi messaggi scambiati contengono anche la maximum segment size, cioè la grandezza massima di un segmento che un host può ricevere, e si utilizzerà il minimo tra i due valori comunicati.

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 2 ms tra la stazione A e il bridge e pari a 3 ms tra la stazione B e il bridge;

Le stazioni generano le seguenti trame:

- stazione A: una trama (A1) all'istante $t_{A1}=241$ msec, una trama (A2) all'istante $t_{A2}=259$ msec, e una trama (A3) all'istante $t_{A3}=289$ msec, tutte dirette ad B.
- stazione B: una trama (B1) all'istante $t_{B1}=245$ msec, e una trama (B2) all'istante $t_{B2}=270$ msec, entrambe dirette a 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, $Z = (4+1+8) \cdot N + T$

Determinare:

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

$$V = 800 \text{ kbit/s}$$

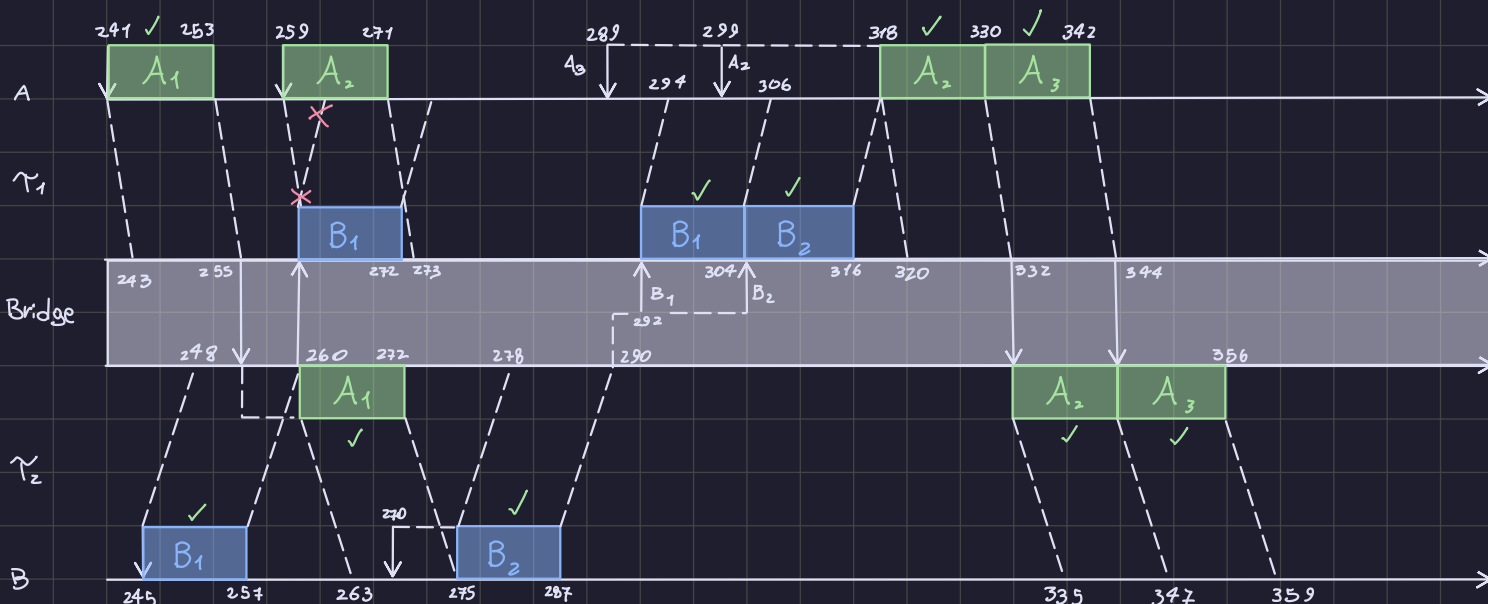
$$L = 1200 \text{ byte}$$

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

$$A: \begin{cases} t_{A1} = 241 \text{ ms} \rightarrow B \\ t_{A2} = 259 \text{ ms} \rightarrow B \\ t_{A3} = 289 \text{ ms} \rightarrow B \end{cases}$$

$$B: \begin{cases} t_{B1} = 245 \text{ ms} \rightarrow A \\ t_{B2} = 270 \text{ ms} \rightarrow A \end{cases}$$

$$T = \frac{L}{V} = \frac{1200 \cdot 8}{800 \cdot 10^3} = 12 \text{ ms}$$



Alla collisione tra B1 e A2:

$$Z_{B1} = (2 + 6 + 0) \cdot 1 + 12 = 20 \text{ ms} \rightarrow 272 + 20 = 292 \text{ ms}$$

B1 ritrasmette per primo

$$Z_{A2} = (2 + 5 + 9) \cdot 1 + 12 = 28 \text{ ms} \rightarrow 271 + 28 = 299 \text{ ms}$$

Il periodo di vulnerabilità, cioè il periodo in cui possono verificarsi collisioni, di questo sistema è 2τ perché nel csma persistent ci possono essere collisioni soltanto se una stazione trasmette mentre un'altra trama è già stata trasmessa ma non è ancora arrivata alla stazione. In questo caso per il segmento a sinistra $\tau = 2 \text{ ms}$, quindi il periodo di vulnerabilità è 4 ms , per il segmento di destra $\tau = 3 \text{ ms}$ e il periodo di vulnerabilità è di 6 ms .

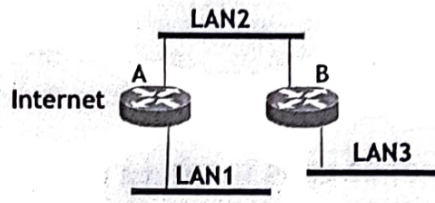
Esercizio 2 (7 punti)

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 85.171.134.68;
- Le LAN 1, 2 e 3 devono poter contenere rispettivamente almeno 1200, 400 e 300 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, cercando di massimizzare la dimensione del blocco (o dei blocchi) di indirizzi lasciati liberi.
3. Si scriva la tabella di routing del router B, considerando come metrica il numero di hop e assumendo che il router A abbia annunciato di poter raggiungere qualsiasi host su Internet in 3 hop.



$$\text{LAN1: } 1200 \rightarrow 2048 = 2^{11} \text{ host}$$

$$\text{LAN2: } 400 \rightarrow 512 = 2^9 \text{ host}$$

$$\text{LAN3: } 300 \rightarrow 512 = 2^9 \text{ host}$$

1. Il blocco CIDR più piccolo è dato dal blocco di host in base 2 più piccolo che riesce a contenere la somma di tutti i blocchi in base 2 delle singole lan:

$$\text{Blocco CIDR} = 2048 + 512 \cdot 2 = 3072 \rightarrow 4096 = 2^{12} \text{ host}$$

Sappiamo che il blocco CIDR ha bisogno di 12 bit di suffisso, quindi di 20 bit di prefisso. Possiamo ricavare l'indirizzo di rete partendo dall'indirizzo che ci è stato fornito:

85 . 171 . 134 . 68



85 . 171 . 1 0 0 0 0 1 1 0 . 0 1 0 0 0 1 0 0

Un indirizzo di rete è caratterizzato da tutti i bit di suffisso posti a 0, mentre un indirizzo di broadcast è caratterizzato da tutti i bit di suffisso posti a 1:

Rete: 85 . 171 . 1 0 0 0 0 0 0 0 . 0 0 0 0 0 0 0 0

Prefixo



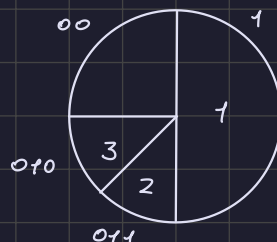
85 . 171 . 128 . 0 /20

Broadcast: 85 . 171 . 1 0 0 0 1 1 1 1 . 1 1 1 1 1 1 1 1

Prefixo



85 . 171 . 143 . 255 /20



2. La lan 2 ha bisogno di 9 bit di suffisso, quindi 23 di prefisso di cui 3 saranno dedicati alla sottorete. Possiamo ricavare la sottorete utilizzando l'indirizzo appartenente alla lan2 che ci è stato fornito

Rete: 85 . 171 . 1 0 0 0 0 1 1 0 . 0 0 0 0 0 0 0 0

Prefisso Sottorete

↓

85 . 171 . 134 . 0 / 23

Broadcast: 85 . 171 . 1 0 0 0 0 1 1 1 . 1 1 1 1 1 1 1 1

Prefisso Sottorete

↓

85 . 171 . 135 . 255 / 23

La lan 3 ha bisogno di 9 bit di suffisso, quindi 23 di prefisso di cui 3 saranno dedicati alla sottorete:

Rete: 85 . 171 . 1 0 0 0 0 1 0 0 . 0 0 0 0 0 0 0 0

Prefisso Sottorete

↓

85 . 171 . 132 . 0 / 23

Broadcast: 85 . 171 . 1 0 0 0 0 1 0 1 . 1 1 1 1 1 1 1 1

Prefisso Sottorete

↓

85 . 171 . 133 . 255 / 23

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

Rete: 85 . 171 . 1 0 0 0 1 0 0 0 . 0 0 0 0 0 0 0 0

Prefisso Sottorete

↓

85 . 171 . 136 . 0 / 23

Broadcast: 85 . 171 . 1 0 0 0 1 1 1 1 . 1 1 1 1 1 1 1 1

Prefisso Sottorete

↓

85 . 171 . 143 . 255 / 23

3. La tabella di routing del router B è la seguente:

dst	next hop	cost
LAN1	A	2
LAN2	B	1
LAN3	B	1
Internet	A	4
B	B	0
A	A	1

Esercizio 3 (7 punti)

Un'applicazione A deve trasferire 95200 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 1400 byte;
- RCVWND annunciata da B ad A pari a 11200 byte; a partire dal tempo $t_a > 3.0$ la destinazione annuncia una RCVWND pari a 19600 byte; a partire dal tempo $t_b > 8.0$ la destinazione annuncia una RCVWND pari a 14000 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 $t_1=8.0\text{s}$ a $t_2=10.0\text{s}$;
- da $t_3=13.5\text{s}$ a $t_4=14.5\text{s}$.

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).

$$L = 95200 \text{ byte} = \frac{95200}{1400} = 68 \text{ segmenti}$$

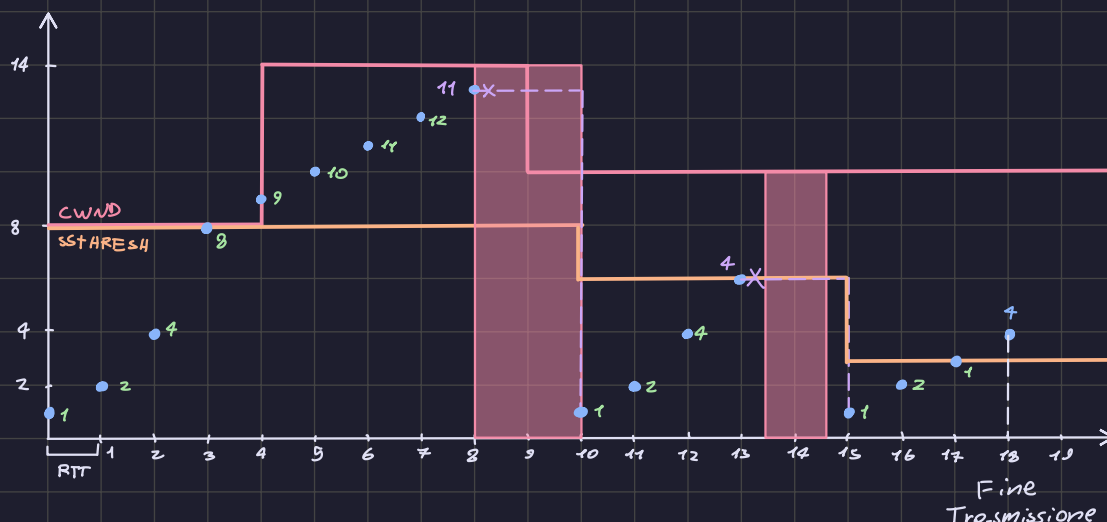
$$\text{MSS} = 1400 \text{ byte}$$

$$\text{RCVWND} = \begin{cases} 11200 \text{ byte} & t \leq 3 = 8 \text{ segmenti} \\ 19600 \text{ byte} & t > 3 = 14 \text{ segmenti} \\ 14000 \text{ byte} & t > 8 = 10 \text{ segmenti} \end{cases}$$

$$\text{SSTHRESH} = \text{CWND}$$

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

$$\text{Down} = \begin{cases} (8, 10) \\ (13.5, 14.5) \end{cases}$$



$$\# \text{segmenti trasmessi} = \begin{matrix} & 14 & 9 & 1 \\ & \text{11} & \text{6} & \text{3} \\ & \text{12} & \text{6} & \text{3} \\ & \text{11} & \text{6} & \text{3} \\ & \text{10} & \text{6} & \text{3} \\ & \text{9} & \text{6} & \text{3} \\ & \text{8} & \text{6} & \text{3} \\ & \text{7} & \text{6} & \text{3} \\ & \text{6} & \text{6} & \text{3} \\ & \text{5} & \text{6} & \text{3} \\ & \text{4} & \text{6} & \text{3} \\ & \text{3} & \text{6} & \text{3} \\ & \text{2} & \text{6} & \text{3} \\ & \text{1} & \text{6} & \text{3} \end{matrix} = 68$$

Il tempo necessario per trasmettere tutti i segmenti è di 18 secondi e la cwnd finale vale 4.