

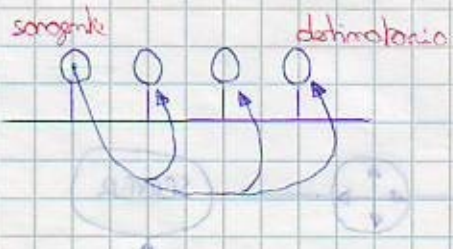
8

Un livello sopra il livello fisico c'è il livello DLG (Data Link Control) che viene anche chiamato livello 2. Questo livello comunica strettamente con l'IP. Sostanzialmente il DLG trasforma i pacchetti provenienti dall'IP in una trama, che altro non è che un pacchetto con un header aggiunto. Quindi:



Si noti che l'header del DLG può essere piccolo o grande a piacere, ma ci deve sempre essere.

Perché viene aggiunto un nuovo header? Consideriamo il caso di ETHERNET in cui i link sono broadcast.



sono obbligati a mettere chi è il destinatario del mio messaggio.

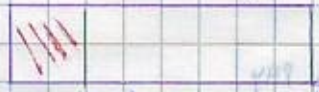
Quindi:



- ↳ c'è l'indirizzo della sorgente
- ↳ c'è l'indirizzo del destinatario

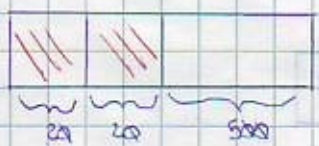
Storicamente il DLG aveva la funzione di ARQ (Automatic Retransmission Request). In quel contesto l'header passava anche un codice di parità che veniva setato a 1 quando nella trama c'erano errori. Oggi però questo controllo non si fa più perché l'errore di bit è diventato raro. Noi quindi assumeremo che se ci sono degli errori di bit, la trama viene completamente usata. Quindi per noi il DLG svolge la funzione di pre-indirizzamento canali. In generale si ha:

↳ 500 byte di PAYLOAD



↳ Header del TCP di 20 byte

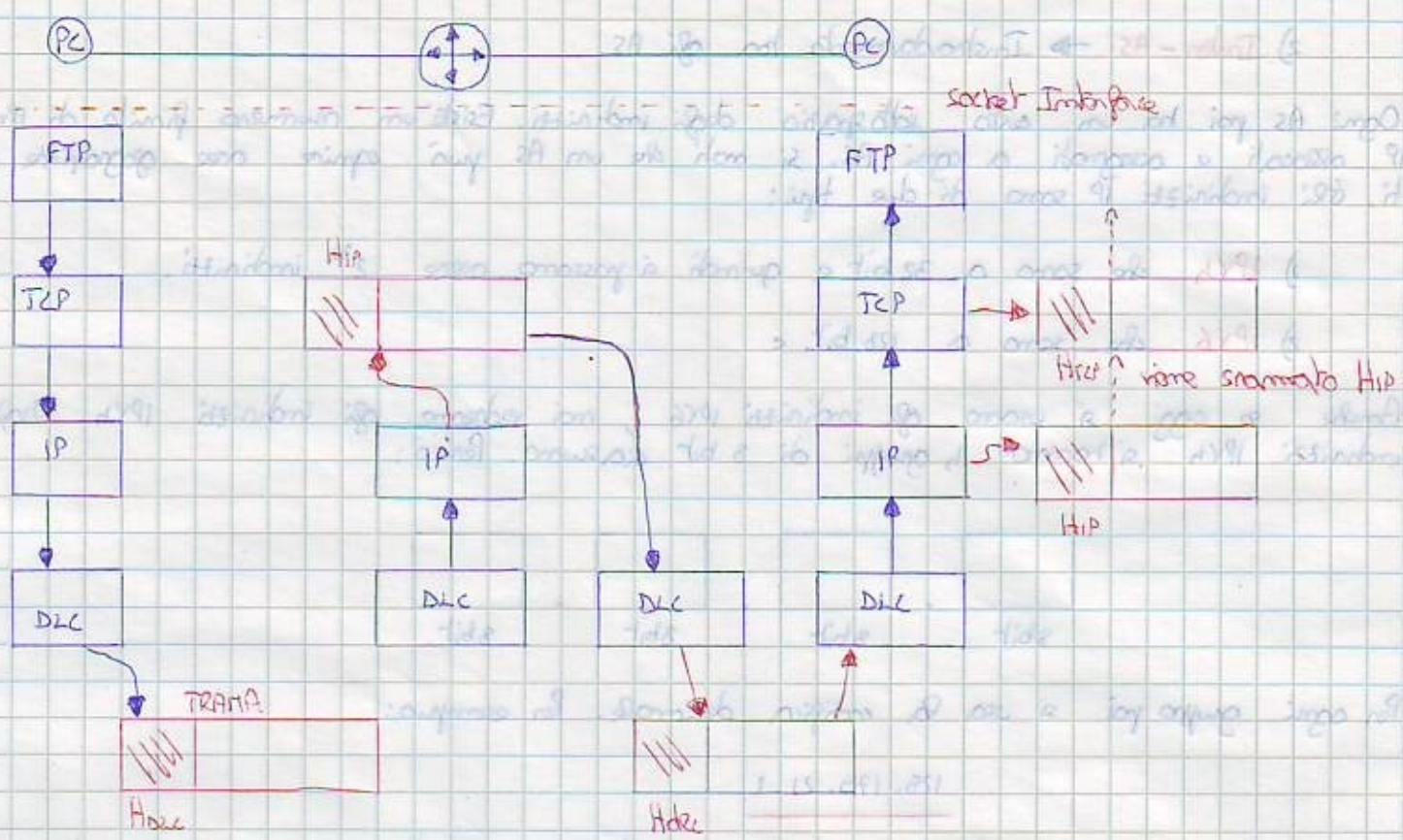
Il livello IP aggiunge il suo header da 20 byte. Quindi il DLG riceve il pacchetto così fatto:



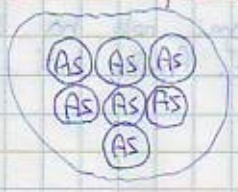
e lo trasforma in TRAMA:



Quindi si ha le seguenti procedure:



Quindi sostanzialmente abbiamo visto le operazioni di **segmentazione** e di **riassemblo** **biggio**. Si ha la segmentazione in trasmissione e il riassemblo in ricezione. Per essere più precisi, nel blocco DLC esiste un'unità chiamata **MTU (Maximum Transfer Unit)** che mi definisce la quantità massima di dati che possono essere trasmessi attraverso una rete fisica, determinata dalle hardware di rete. Quindi se per esempio il pacchetto IP è di 1500 byte, e MTU è di 1000 byte, il pacchetto stesso viene suddiviso in due pacchetti e header. Quindi è MTU determina se un pacchetto deve essere frammentato o meno. Quindi ogni pacchetto potrebbe essere frammentato per adeguarsi all'MTU del DLC. Questo avviene in fase di trasmissione. In fase di ricezione invece, nel TCP si ha il **MSS (Maximum Segment Size)** che mi dice la quantità massima di dati che possono essere trasmessi in un segmento. In pratica e il destinatario lo decide all'arrivo della connessione. Quindi in realtà è MTU e MSS si hanno sia in fase di trasmissione che in fase di ricezione. Si possono aumentare sia l'MTU sia l'MSS fino a un massimo di 1500 byte dopo di che si inizia a perdere. Per default l'MTU e l'MSS hanno delle dimensioni dell'ordine di 500 byte. Analizziamo ora l'IP e in particolare l'**IP ADDRESSING**. Abbiamo detto che l'IP si occupa di trasferire pacchetti. Ma l'IP per svolgere la sua funzione ha bisogno di una **schema di indirizzamento**. Prima di parlare di questo definiremo per noi cosa è Internet. Internet è una collezione di **AS (Autonomous Systems)**. Questa collezione non è strutturata. Quindi:



La cosa è un AS? Un AS è un insieme di router e reti che rientrano in una unità amministrativa e cooperano per scambiarsi informazioni. Quindi un AS è un pezzo di Internet gestito da un'unità amministrativa.

Tra AS vi è uno scambio di pacchetti. L'indirizzamento di pacchetti tra gli AS può essere di due tipi:

1) Inter-AS → Instradamento negli AS.

2) Intra-AS → Instradamento tra gli AS.

Ogni AS poi ha un certo sottospazio degli indirizzi. Esiste un numero finito di indirizzi IP allocati e assegnati a ogni AS. Si può dire che un AS può coprire aree geografiche differenti. Gli indirizzi IP sono di due tipi:

1) IPv4 che sono a 32 bit e quindi si possono avere  $2^{32}$  indirizzi.

2) IPv6 che sono a 128 bit.

Anche se oggi si usano gli indirizzi IPv6, noi vedremo gli indirizzi IPv4. Negli indirizzi IPv4 si hanno 4 gruppi di 8 bit ciascuno. Perciò:



Per ogni gruppo poi si usa la notazione decimale. Per esempio:

125.175.21.1

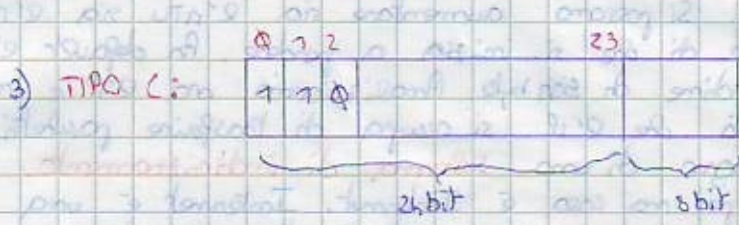
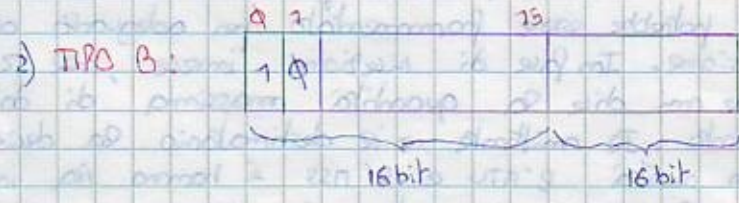
Un primo sistema di indirizzamento che andiamo a vedere è lo sistema di indirizzamento a classi. Le tre prime di indirizzi Internet che andiamo a vedere sono:

→ 24 bit per l'identificatore della rete (host number).



↳ 8 bit per l'identificatore di rete

NB: nello schema classful non c'è questa suddivisione.



→ quando il campo host number è vuoto, si hanno reti puerche.

Quindi per le tipo A si possono avere  $2^{24} = 2^7 = 128$  reti. Ogni rete ha  $2^{24}$  sottindirizzi. Esiste uno speciale indirizzo di host number che è \*. Anche tutti i bit uguali a 0 e a 1 sono speciali. In realtà si hanno:

$2^3 = 2$  reti e  $2^{24} = 2$  indirizzi.

Per le tipo B si hanno  $2^{16} = 2$  reti e  $2^{16} = 2$  indirizzi, mentre per le tipo C si hanno  $2^{24} = 2$  reti e  $2^8 = 2$  indirizzi. Nella notazione decimale gli indirizzi di classe A iniziano

ma con un numero compreso tra 1 e 255. Quindi:

$(1-255) \cdot * \cdot * \cdot * / 8$

↳ indica la lunghezza del network prefix de quozionalmente il modo indirizzamento di tipo classful è inutile.

Per quanto riguarda il tipo B invece si ha:

$(128-191) \cdot * \cdot * \cdot * / 16$

Mentre per il tipo C si ha:

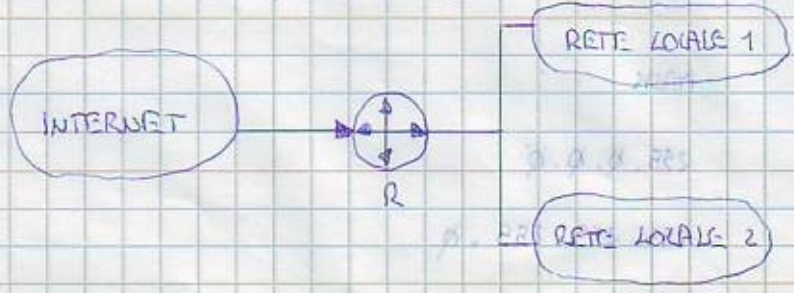
$(192-223) \cdot * \cdot * \cdot * / 24$

Vediamo ora le caratteristiche di subnetting e di indirizzamento classless. Supponiamo di dover gestire la seguente rete:

$128.10 \cdot * \cdot *$  (Classe B).

Questa rete è collegata ad Internet mediante alcuni router. Per prima cosa bisogna assegnare gli indirizzi della rete alla interfaccia del PC. L'assegnazione può essere manuale o automatica (mediante server DHCP).

In questo caso abbiamo un numero elevato di indirizzi disponibili ed è opportuno assegnare tali indirizzi con qualche criterio. Possa creare delle sottoreti cioè una gerarchia nella rete basata sulle zone geografiche per ritrovare più facilmente i PC stessi nella rete. Notiamo subito che nello schema originale di indirizzamento IP a ciascuna rete viene assegnato un indirizzo di rete univoco e ogni host ha l'indirizzo di rete come prefisso del proprio indirizzo specifico, detto indirizzo di host. Si noti che nasce l'esigenza di indirizzare un singolo indirizzo di rete ha più reti fisiche. Per esempio:



Supponiamo di voler inviare del traffico ad un generico host presente nella rete 1. Tutto il traffico viaggia lungo Internet. Tutti i router seguono l'instradamento tramite R. Una volta che il pacchetto raggiunge R, esso deve essere inviato sulla rete fisica corretta. Supponiamo per esempio che:



NB: ottetto = gruppo di 8 bit.

In breve si associa all'ottetto e indirizzo per distinguere le due reti, mentre l'ottetto serve per identificare il host. Quindi un indirizzo IP a 32 bit comprende una parte

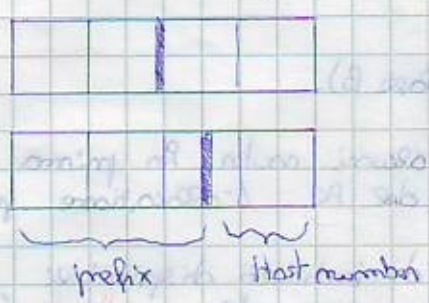
che identifica un sito con più reti fisiche, e una parte base che identifica una rete fisica e le relative host interessate. Ora introduciamo il concetto di **network mask** (**subnet mask**). La Network Mask è una particolare configurazione a 32 bit tale che i bit nella maschera impostati a 1 identificano la rete, mentre i bit impostati a zero identificano il host. Per esempio:

11111111 11111111 11111111 00000000

I primi tre ottetti mi identificano la rete, mentre il quarto identifica il host. Se ho per esempio:

11111111 11111111 00011000 01000000

I primi due ottetti i due bit del terzo ottetto e un bit del quarto sono assegnati alla rete. Prendiamo ora un indirizzo IP e aggiungiamo la network prefix qual'è la **longitudinale** degli address, cioè togliendoci due host numbers. Graficamente:



Quarta e una parte della ottava della Network Mask.

Se per esempio abbiamo:

RETE 130.5.\*.\*  
255.255.0.0  
-----  
130.5.0.0

Quindi la network mask è una stringa che messa in AND bit a bit con l'indirizzo IP mi fornisce il network prefix.

Le mask di default sono:

| CLASSE | MASK          |
|--------|---------------|
| A      | 255.0.0.0     |
| B      | 255.255.0.0   |
| C      | 255.255.255.0 |

Quindi prendo subnetting con la rete 130.5.\*.\* /16 si ottiene:

130.5.\*.\* /21 ← siamo passati ora a questa rete.

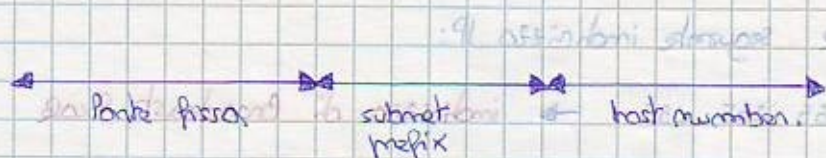
In binario è:

100000100 . 00000101 . 00000000 . 00000000

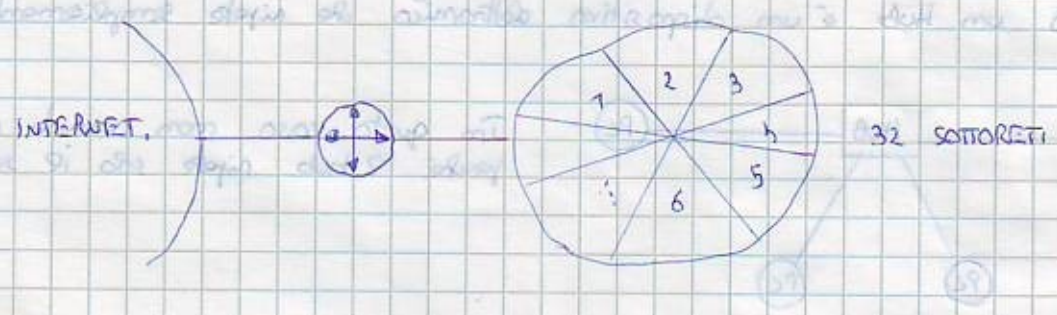
La MASK è:

11111111 . 11111111 . 11110000 . 00000000 → 21.

Quindi graficamente c'è:

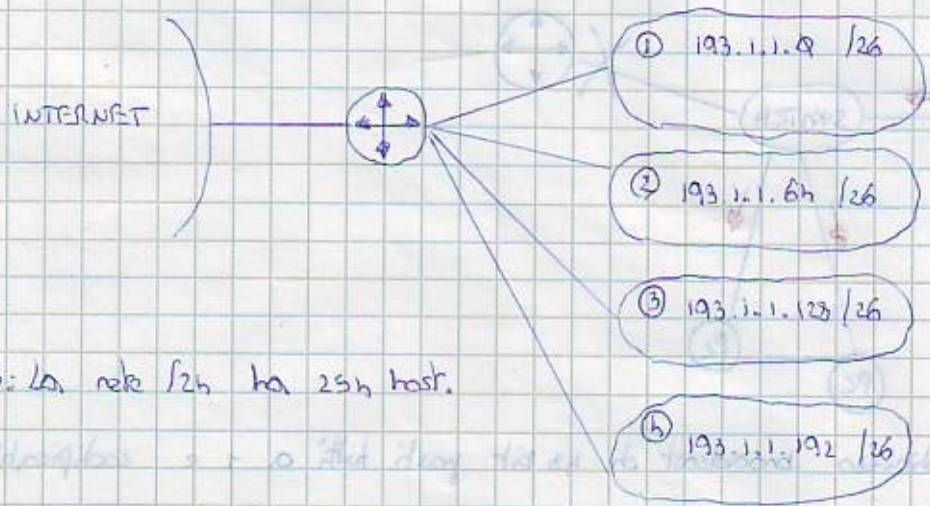


Così prendo lo quindi suddiviso lo spazio degli indirizzi in 32 sottoreti.



Consideriamo un altro esempio:

193.1.1.0 /24 → rete CLASSE C.



Abbiamo 4 sottoreti con 62 host ciascuna.

NB: La rete /24 ha 254 host.

- 1°: 193.1.1. 00000000
- 2°: 193.1.1. 01000000
- 3°: 193.1.1. 10000000
- 4°: 193.1.1. 11000000

Suddividiamo ora di avere 64, seguente sottoreti:

192.1.1.0 /24

e a rete gestire 6 laboratori assegnando 6 subnet, una per ogni laboratorio.

Prendo:

|         |     |        |     |
|---------|-----|--------|-----|
| 192.1.1 | 000 | 000000 | /25 |
| 192.1.1 | 001 | 000000 | /27 |
| 192.1.1 | 010 |        | /27 |
|         | 011 |        |     |
|         | 100 |        |     |
|         | 101 |        |     |
|         | 110 |        |     |
|         | 111 |        |     |

Quindi se ho: Ho creato 3 sottoreti.  
 11111111.11111111.11111111.00000000 (111)000000  
 /19 → cioè ho 19 bit per la 1

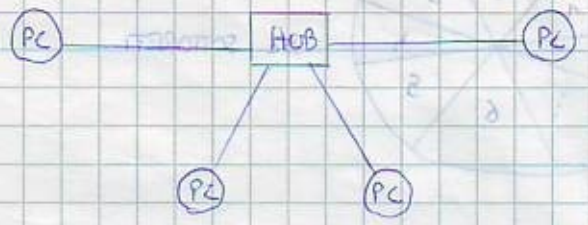
NB: I bit della subnet mask vanno sempre messi in modo contiguo.

13

Consideriamo ora le seguenti indirizzi IP:

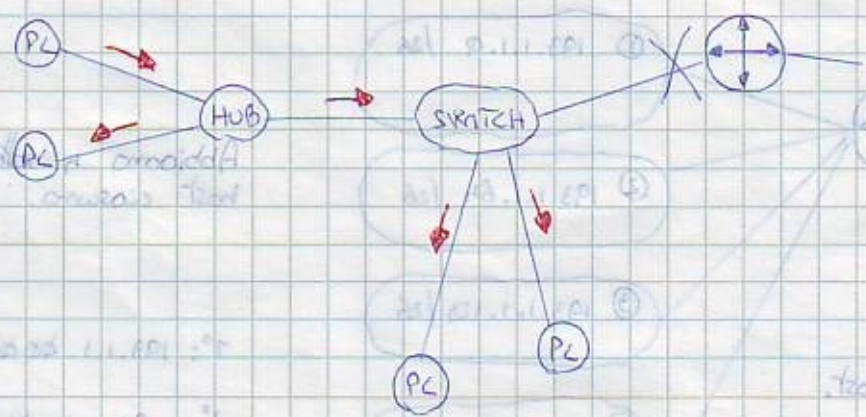
255.255.255.255 → indirizzo di Broadcast locale

Così facendo si spedisce il pacchetto a tutta la rete. Questo metodo però è comodo solo in alcuni casi. Vediamo ora brevemente cosa accade se si usano i broadcast usando Hub e SWITCH. Innanzitutto un hub è un dispositivo elettronico che ripete semplicemente il segnale.



In questo caso non accade nulla perché l'hub ripete solo il segnale.

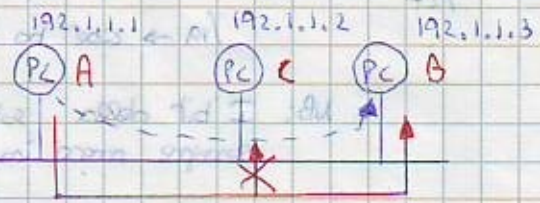
Se uso uno switch che è un dispositivo meno complesso del router e più complesso dell'hub, e che ha il compito di smistare la trama scegliendo per una determinata trama che la trama giunge ad ogni PC connesso nella rete locale in questione. Per esempio:



Anche Ethernet ha il suo indirizzo broadcast di 32 bit tutti a 1 e codificati in decimale:

FF.FF.FF.FF.FF.FF

La trama viene spedita su rete locale (per esempio ETHERNET). Si ricordi infatti che lo switch funziona a livello MAC e non possiede la nozione di pacchetto IP, ma possiede la nozione di trama. Quindi lo switch "vede" e manda due indirizzi Ethernet e "capisce" che è un messaggio BROADCAST e quindi lo "spara" dappertutto. Il router però lo blocca. Consideriamo per esempio A e B che si collegano alla rete fisica moderna. Assegniamo a queste macchine gli indirizzi IP  $I_A$  e  $I_B$ , e gli indirizzi fisici  $P_A$  e  $P_B$ . Si ipotizzi che A voglia inviare un pacchetto a B.



La trama viene spedita a tutte le macchine.

Supponiamo che A conosca solo  $I_B$  di B. Come fa A a tradurre  $I_B$  in  $P_B$ ? Si usa in merito il protocollo ARP (Address Resolution Protocol). In tutta funzione, così: quando A vuole risolvere l'indirizzo IP di B, cioè vuole trovare l'indirizzo fisico di B, trasmette un