

Corso di Laurea in Ingegneria Informatica



**Corso di Reti di Calcolatori
(a.a. 2010/11)**

Roberto Canonico (roberto.canonico@unina.it)

Giorgio Ventre (giorgio.ventre@unina.it)

**Il livello rete in Internet
Il protocollo IP**

3 novembre 2010

**I lucidi presentati al corso sono uno strumento didattico
che NON sostituisce i testi indicati nel programma del corso**

Nota di copyright per le slide COMICS



Nota di Copyright

Questo insieme di trasparenze è stato ideato e realizzato dai ricercatori del Gruppo di Ricerca COMICS del Dipartimento di Informatica e Sistemistica dell'Università di Napoli Federico II. Esse possono essere impiegate liberamente per fini didattici esclusivamente senza fini di lucro, a meno di un esplicito consenso scritto degli Autori. Nell'uso dovranno essere esplicitamente riportati la fonte e gli Autori. Gli Autori non sono responsabili per eventuali imprecisioni contenute in tali trasparenze né per eventuali problemi, danni o malfunzionamenti derivanti dal loro uso o applicazione.

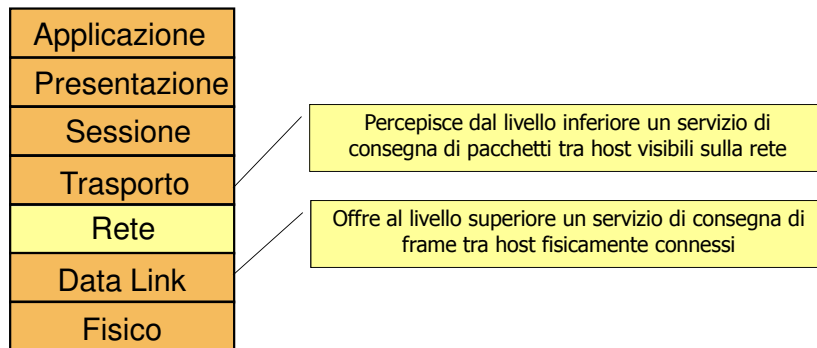
Autori:

Simon Pietro Romano, Antonio Pescapè, Stefano Avallone,
Marcello Esposito, Roberto Canonico, Giorgio Ventre

Il livello rete



- Stack OSI

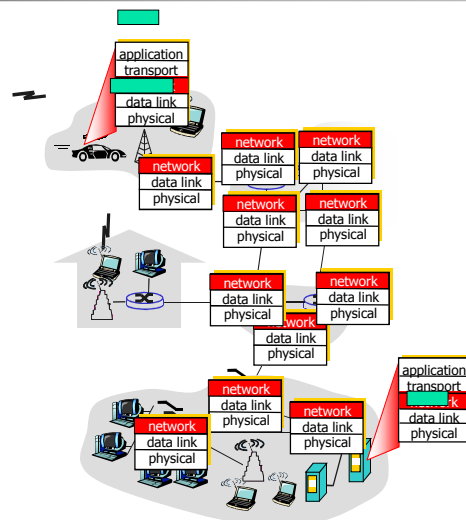


3

Le funzioni del livello rete



- Trasportare i pacchetti dall'host mittente a quello ricevente
- Implementare protocolli di livello rete in *tutti* i router e in *tutti* gli host



4

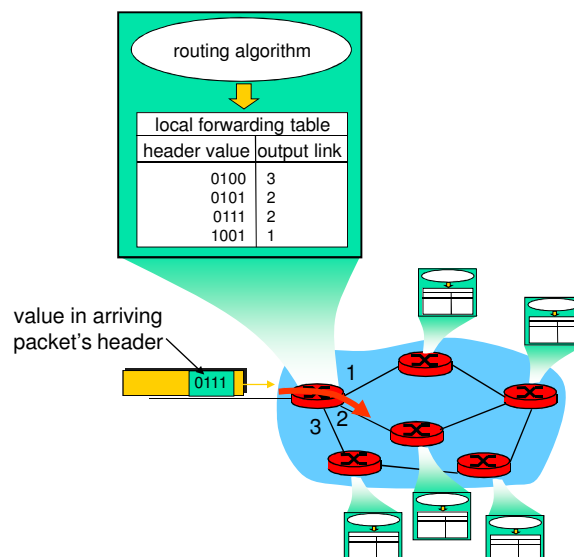
Le funzioni di forwarding e routing



- **forwarding**: spostare i pacchetti dalla coda/interfaccia di ingresso a quella di uscita
 - **routing**: determinare la strada che un pacchetto deve seguire da una sorgente ad una destinazione:
 - *routing algorithms*
- analogia:**
- **routing**: processo di costruzione di un viaggio dalla partenza all'arrivo
 - A mano
 - Usando un software
 - ...
 - **forwarding**: processo di movimentazione ad ogni singola rotonda...

5

La relazione tra forwarding e routing



6

Le funzioni del livello rete



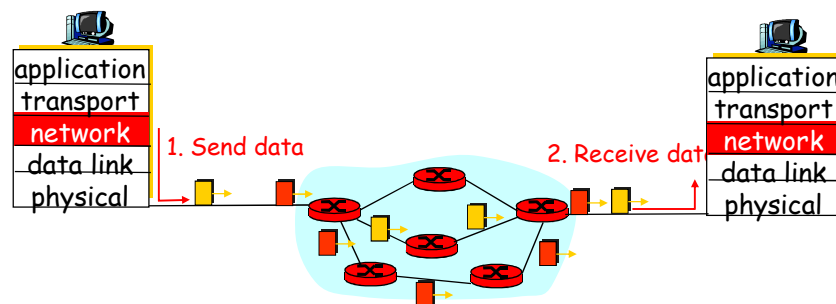
- Le reti possono essere classificate a seconda del metodo utilizzato per trasportare i pacchetti dalla sorgente alla destinazione:
 - **Reti a datagrammi**
ogni pacchetto è instradato indipendentemente dagli altri pacchetti dello stesso flusso
 - **Reti a circuiti virtuali**
viene precalcolato un percorso e tutti i pacchetti del flusso seguono questo percorso
- **NB: parliamo comunque di reti a commutazione di pacchetto!**

7

Reti a datagrammi



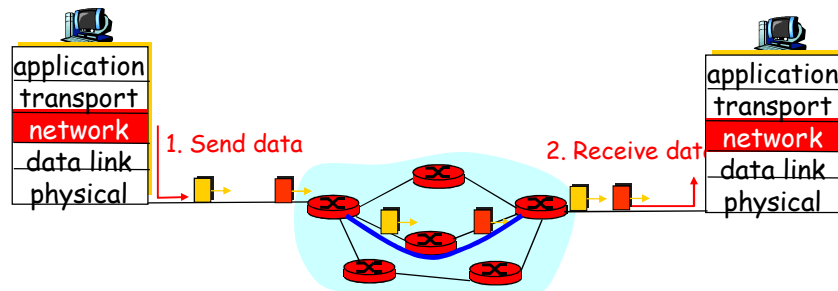
- Ogni router che riceve un pacchetto decide indipendentemente a chi mandarlo sulla base dell'indirizzo destinazione contenuto nel pacchetto
- Pacchetti tra la stessa coppia sorgente-destinazione possono seguire percorsi differenti



8

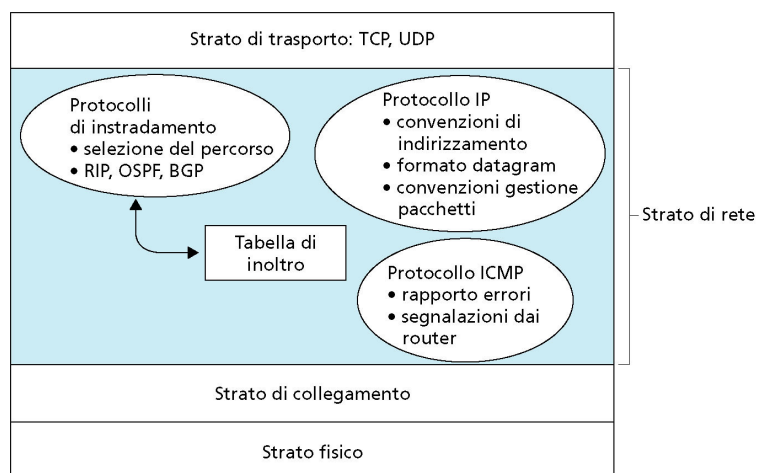
Reti a circuiti virtuali

- Ogni pacchetto contiene il numero del circuito virtuale
- Il circuito virtuale è stabilito prima della trasmissione dei dati
- I nodi devono conservare informazioni su tutti i circuiti virtuali che li attraversano



9

Lo strato di rete di Internet



10

IP (Internet Protocol)



- IP è un protocollo di livello rete usato per lo scambio di dati tra reti di calcolatori
- I dati sono trasportati con la tecnica dei datagrammi
- Offre un servizio di comunicazione *connection-less*
- Gestisce indirizzamento, frammentazione, riassemblaggio e multiplexing dei protocolli
- Costituisce la base sulla quale si basano tutti gli altri protocolli, collettivamente noti come *TCP/IP suite*
 - TCP, UDP, ICMP, ARP
- È responsabile dell'instradamento dei pacchetti

11

Il datagramma IP



- Un pacchetto IP è anche chiamato *datagramma*
- È costituito da un *header* e un'area dati
- I datagrammi possono avere dimensioni diverse
- La dimensione dell'header è solitamente fissata (**20 byte**) a meno che non siano presenti opzioni
- Un datagramma può contenere fino a un massimo di 65535 byte ($2^{16} - 1$)

12

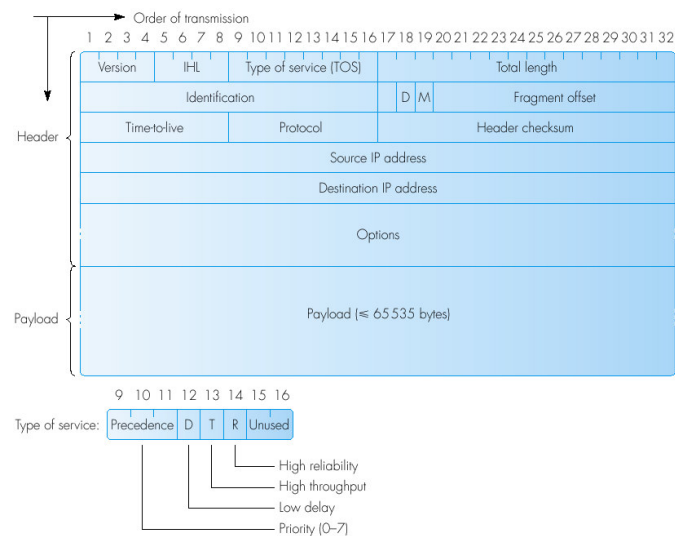
L'header IP



- L'header contiene tutte le informazioni necessarie per la consegna del datagramma alla destinazione
 - Indirizzo destinazione
 - Indirizzo sorgente
 - Identificativo
 - Ed altro ancora...
- I router esaminano l'header di ogni datagramma e inoltrano il pacchetto lungo il percorso verso la destinazione
 - Usano tabelle di routing per calcolare il *next hop*
 - Aggiornano tali tabelle usando protocolli di routing dinamici

13

Formato del pacchetto IP



IHL = intermediate header length
D = don't fragment

14

Formato del pacchetto IP



- **Version**
 - 4 bit, versione del protocollo IP cui il pacchetto è conforme
- **IP header length (IHL)**
 - 4 bit, lunghezza dell'header, in multipli di 32 bit (max 60 byte)
- **Type-of-Service (ToS)**
 - 8 bit, specifica come un protocollo di livello superiore vorrebbe che il pacchetto fosse trattato
- **Total length**
 - 16 bit, specifica la lunghezza in byte dell'intero pacchetto (header + dati)
 - ...max 64kB, cioè 65535 byte ($2^{16} - 1$)

15

Formato del pacchetto IP



- **Time-to-live (TTL)**
 - 8 bit, contatore che viene gradualmente decrementato fino a zero, punto in cui il pacchetto viene scartato. Serve ad evitare che un pacchetto resti perennemente in circolo
- **Protocol**
 - 8 bit, indica il protocollo di livello superiore che riceve il pacchetto dopo che l'elaborazione IP è terminata
 - Analogo al numero di porto di livello trasporto: 6 indica TCP, 17 indica UDP
- **Header checksum**
 - 16 bit, aiuta a garantire l'integrità dell'header IP
- **Source Address**
 - 32 bit, specifica il nodo mittente

16

Formato del pacchetto IP



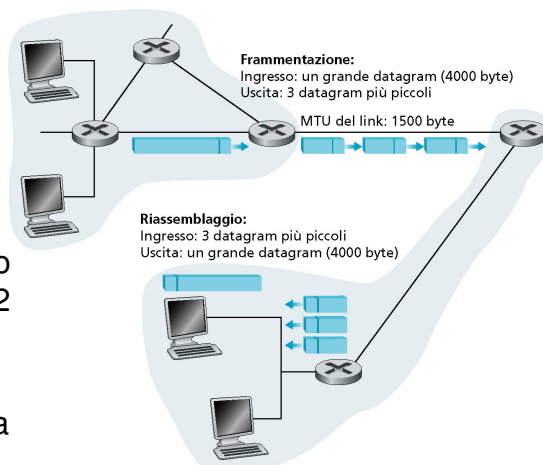
- **Destination Address**
 - 32 bit, specifica il nodo ricevente
- **Identification**
 - I pacchetti possono essere frammentati lungo il percorso
 - Questo campo (16 bit) è un identificativo del datagramma
- **Flags**
 - Il bit D indica se il pacchetto può essere frammentato
 - Il bit M indica se il pacchetto è l'ultimo frammento
- **Fragment offset**
 - 13 bit, identifica la posizione del frammento all'interno del pacchetto

17

Frammentazione e riassetblaggio IP



- Tutti i frammenti tranne l'ultimo devono essere multipli di 8 byte
- 8 byte è la dimensione del frammento elementare
- Avendo 13 bit a disposizione, ci possono essere al massimo 8192 frammenti per ogni datagramma
- La dimensione massima di un datagramma è 65535 byte



18

Opzioni



- È il modo per estendere IP con un numero variabile di opzioni
 - Security
 - Source routing
 - Route recording
 - Stream identification
 - Timestamping
- A causa delle opzioni, l'header può essere di lunghezza variabile
 - Questo è il motivo della presenza del campo IHL
 - Se l'opzione non occupa 4 byte (o un suo multiplo), vengono inseriti dei bit di riempimento (tutti zero)
 - La presenza opzionale di questi campi rende difficile la gestione in implementazioni hw-based

19

IP è consegna Best effort



- IP *non* garantisce di prevenire:
 - Datagrammi duplicati
 - Consegna ritardata o fuori ordine
 - Corruzione di dati
 - Perdita di pacchetti (e di frammenti)
- La consegna affidabile dei pacchetti può avvenire grazie a meccanismi di controllo da parte di protocolli di livello superiore

20

Sistema di numerazione decimale (cenni)



- La base è 10. Questo vuol dire che:
 - Si utilizzano 10 simboli (le cifre da 0 a 9)
 - Ogni cifra è pesata secondo una potenza di 10 con esponente crescente da destra verso sinistra

- es.

$$(2536)_{10} = 2 \cdot 10^3 + 5 \cdot 10^2 + 3 \cdot 10^1 + 6 \cdot 10^0$$

21

Sistema di numerazione binaria (cenni)



- La base è 2. Questo vuol dire che:
 - Si utilizzano 2 simboli (le cifre 0 e 1)
 - Ogni cifra è pesata secondo una potenza di 2 con esponente crescente da destra verso sinistra

- es.

$$(11010)_2 = 1 \cdot 2^4 + 1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 = (26)_{10}$$

- Abbiamo così visto anche il modo per convertire un numero da base 2 a base 10

22

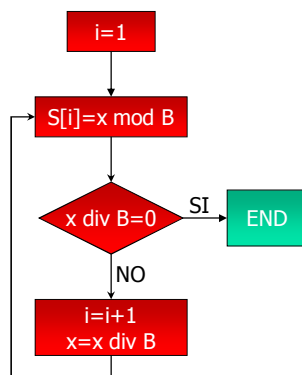
Sistema di numerazione esadecimale (cenni)



- La base è 16. Questo vuol dire che:
 - Si utilizzano 16 simboli (le cifre da 0 a 9 e le lettere da A a F)
 - Ogni cifra è pesata secondo una potenza di 16 con esponente crescente da destra verso sinistra
- es.
 $(1E5)_{16} = 1 \cdot 16^2 + 14 \cdot 16^1 + 5 \cdot 16^0 = (485)_{10}$
- Abbiamo così visto anche il modo per convertire un numero da base 16 a base 10

23

Conversione da decimale (cenni)



i: contatore
S: stringa risultato
x: numero da convertire
B: base

Esempio

Vogliamo convertire 25 in binario:

25 mod 2 = 1 $\Rightarrow$ S=**1**
25 div 2 = 12
12 mod 2 = 0 $\Rightarrow$ S=**01**
12 div 2 = 6
6 mod 2 = 0 $\Rightarrow$ S=**001**
6 div 2 = 3
3 mod 2 = 1 $\Rightarrow$ S=**1001**
3 div 2 = 1
1 mod 2 = 1 $\Rightarrow$ S=**11001**
1 div 2 = 0
END

$(25)_{10} = (11001)_2$

24

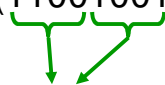
Relazione tra numeri binari ed esadecimali



- Una stringa di 4 bit può assumere $2^4=16$ diversi valori
- Se consideriamo ogni stringa come numero binario, essa rappresenta un numero compreso tra 0 e 15
- In base 16 abbiamo 16 cifre che assumono un valore compreso tra 0 e 15



- 1 cifra esadecimale “rappresenta” 4 cifre decimali

- es. $(201)_{10}$  $(11001001)_2$
 $(C9)_{16}$

25

Indirizzi IP



- Ad ogni host è assegnato un indirizzo IP o indirizzo Internet
 - È un numero di 32 bit = 4 byte
 - Unico in tutta Internet
- Ogni indirizzo IP è diviso in un prefisso e un suffisso
 - Il prefisso indica la rete alla quale l'host è collegato
 - Due reti differenti hanno numero di rete differente
 - Il suffisso identifica l'host all'interno della rete
 - Due host sulla stessa rete non possono avere lo stesso suffisso, ma host su reti diverse possono avere lo stesso suffisso



26

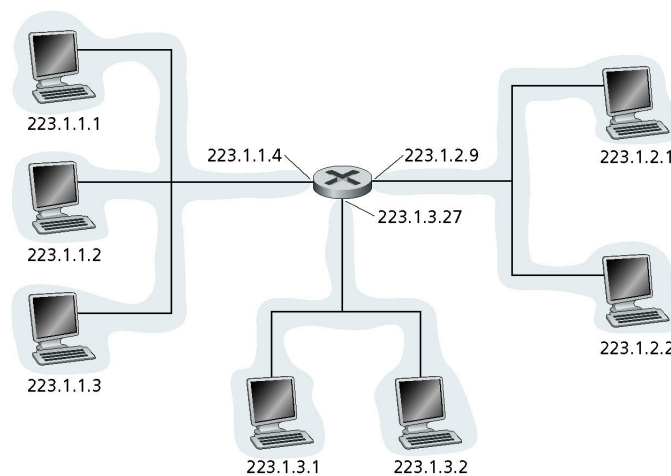
Chi assegna gli indirizzi IP?



- **ICANN:**
 - Internet **C**orporation for **A**ssigned **N**ames and **N**umbers
- Assegna gli indirizzi
- Gestisce il DNS
- Assegna i nomi dei domini
- Risolve eventuali dispute (conflitti di nomi e/o indirizzi)

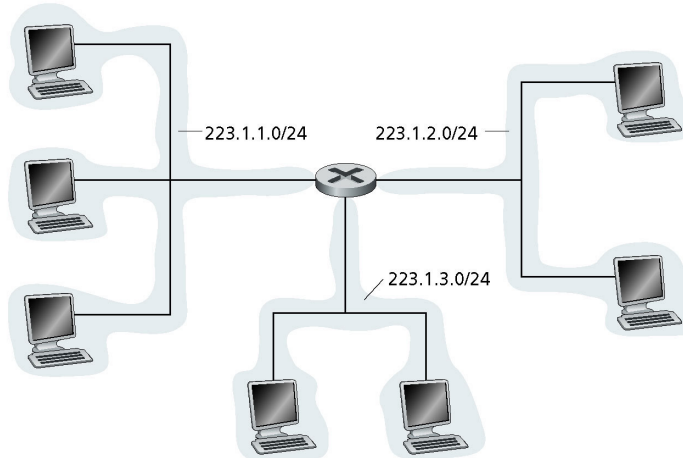
27

Indirizzi delle interfacce...



28

...indirizzi delle reti

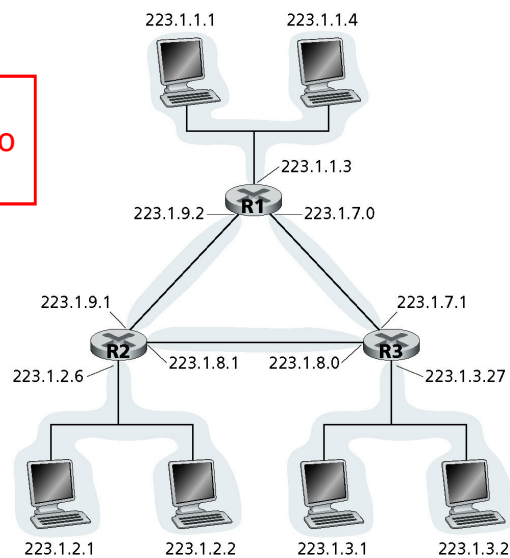


29

Tre router che interconnettono sei host



Domanda:
Quante sono
le reti?



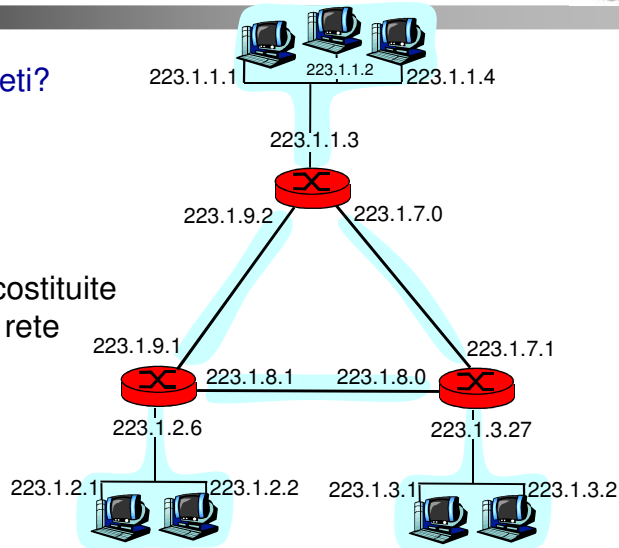
30

Soluzione...

Come trovare le reti?

- 'Staccare' ogni interfaccia dal corrispondente router/host
- Creare "isole" costituite da segmenti di rete disgiunti

Nell'esempio:
sistema
interconnesso
costituito da sei
reti



31

Notazione *dotted decimal*

- La notazione dotted decimal rappresenta gli indirizzi IP come 4 numeri decimali separati da punto
- Ogni numero decimale, poiché rappresenta un byte, è compreso tra 0 e 255

32-bit Binary Number	Equivalent Dotted Decimal
10000001 00110100 00000110 00000000	129 . 52 . 6 . 0
11000000 00000101 00110000 00000011	192 . 5 . 48 . 3
00001010 00000010 00000000 00100101	10 . 2 . 0 . 37
10000000 00001010 00000010 00000011	128 . 10 . 2 . 3
10000000 10000000 11111111 00000000	128 . 128 . 255 . 0

32

Classi di indirizzi



- La parte di indirizzo che specifica la rete e quella che specifica l'host non hanno lunghezza fissa, ma variano a seconda della *classe* a cui appartiene l'indirizzo
- Sono state definite 5 classi:
 - 3 (A, B, C) sono usate per gli indirizzi degli host e si differenziano per la lunghezza della parte rete/host
 - 1 (D) è usata per il *multicast*
 - 1 (E) è riservata per usi futuri

33

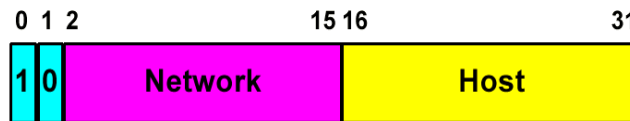
Indirizzi di classe A



- Campo rete
 - 7 bit
 - Massimo 128 reti
 - Il primo byte è compreso tra 0 e 127
- Campo host
 - 24 bit
 - Massimo $2^{24} \approx 16\text{M}$ host

34

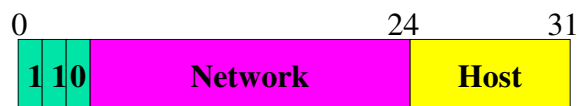
Indirizzi di classe B



- Campo rete
 - 14 bit
 - Massimo 16k reti
 - Il primo byte è compreso tra 128 e 191
- Campo host
 - 16 bit
 - Massimo $2^{16} \approx 64k$ host

35

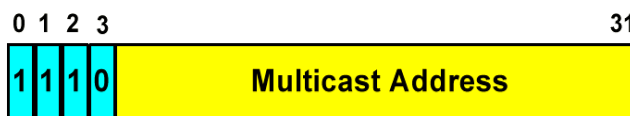
Indirizzi di classe C



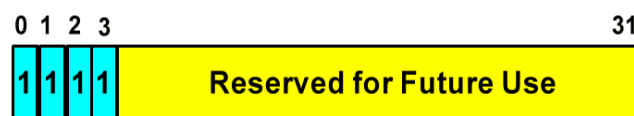
- Campo rete
 - 21 bit
 - Massimo 2M reti
 - Il primo byte è compreso tra 192 e 223
- Campo host
 - 8 bit
 - Massimo 256 host

36

Indirizzi di classe D e E



Classe D



Classe E

37

Indirizzi IP “speciali”



- **Network address**
 - La rete stessa ha un indirizzo, il cui suffisso è costituito da tutti '0'
 - Nessun host può quindi avere tutti '0' nel suffisso
- **Directed broadcast address**
 - Per mandare un messaggio in broadcast ad una rete il suffisso è costituito da tutti '1': il pacchetto è inviato a tutti gli host di una specifica rete
- **Limited broadcast address**
 - L'intero indirizzo (sia la parte rete che la parte host) è costituito da tutti '1': 255.255.255.255
 - Broadcast sulla LAN locale

38

Indirizzi IP “speciali”



- **This computer address**
 - L'intero indirizzo è costituito da tutti '0'
 - Per ottenere un indirizzo automaticamente all'avvio, si potrebbe usare IP per comunicare...
 - ...ma non abbiamo ancora un indirizzo:
 - vedremo che per l'assegnazione di tale indirizzo si utilizzano protocolli quali DHCP (Dynamic Host Configuration Protocol) e BOOTP (Boot Protocol)
- **Loopback address**
 - Ogni indirizzo che comincia con 127 indica il computer locale
 - 127.0.0.1 è il più comune, ma va bene anche 127.0.44.53
 - Usato per test, nessun pacchetto esce sulla rete
 - Utile quando il computer non ha schede di rete

39

Indirizzi IP



- La suddivisione degli indirizzi IP in classi non è efficiente perché comporta lo spreco di indirizzi
- 32 bit $\Rightarrow 2^{32} \approx 4$ miliardi di indirizzi diversi, ma non tutti vengono usati
- Rimedi:
 - Indirizzi privati
 - Sottoreti

40

Indirizzi IP privati



- L'RFC 1597 riserva i seguenti blocchi di indirizzi per uso privato:
 - 10.0.0.0 - 10.255.255.255 Classe A
 - 172.16.0.0 - 172.31.255.255 Classe B
 - 192.168.0.0 - 192.168.255.255 Classe C
- I router di Internet non inoltrano pacchetti aventi indirizzo sorgente o destinazione compreso in uno di questi blocchi
- Vedremo dopo il modo per tradurre tali indirizzi in indirizzi Internet pubblici (NAT)