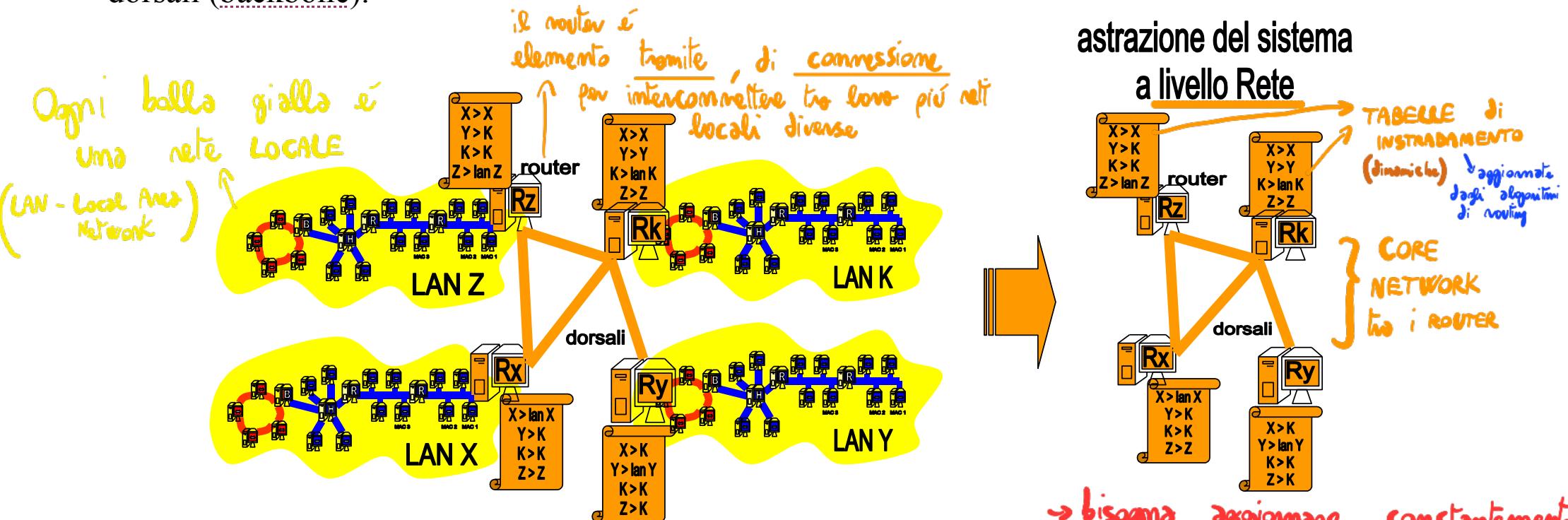


c'è la possibilità di
→ comunicazione di rete tra reti
(INTERNET)

Reti di reti e Internetworking

Le reti locali sono connesse attraverso collegamenti organizzati in modo gerarchico, basati su calcolatori rappresentanti della rete locale (router) a loro volta collegati da linee dati veloci o dorsali (backbone).



Un esempio di rete di reti locali e router collegati da dorsali, e l'astrazione dello stesso sistema al terzo livello (Rete) visto dai router.

La figura mostra a sinistra quattro reti locali, X, Y, Z e K dotate ognuna di un dispositivo router (instradatore) Rx, Ry, Rx e Rk, rispettivamente. Ogni router, oltre ad essere connesso alla propria rete locale, è connesso attraverso linee di comunicazione, dette dorsali, ad altri router. I router e le dorsali sono di colore scuro (arancio), per evidenziare che si tratta di componenti che agiscono al livello 3 (rete) della gerarchia

=> **ROUTER:** - oggetti costosi
- smistano numeri elevatissimi di segmenti di bit

Copyright © Luciano Bononi 2017

IL LIVELLO RETE ha come obiettivo di rendere la rete EFFICIENTE

Reti di Calcolatori - introduzione

ISO/OSI. Una dorsale connette i router Rx e Rz, un'altra i router Rz e Rk, un'altra i router Rx e Rk e infine un'altra i router Rk e Ry. Ogni router è inoltre istruito da una tabella su quale sia il primo router intermedio per comunicare con ogni altro router. Ad esempio, il router Rz può comunicare indirettamente con Ry per mezzo del router intermedio Rk. A Destra viene mostrata l'astrazione del sistema che appare ad ogni router, ovvero al terzo livello (Rete).

Se i milioni di calcolatori oggi connessi a Internet fossero tutti organizzati secondo i protocolli e gli schemi visti finora per le reti locali, la comunicazione tra due calcolatori su Internet richiederebbe di passare per migliaia di calcolatori intermedi, switch, bridge, segmenti di rete, ognuno dei quali aggiungerebbe ritardi di gestione, complessità, rischi di errore. Il problema dell'instradamento dei frame (routing), ovvero il decidere da che parte o su che segmento deve essere inoltrato un frame per raggiungere il destinatario finale, richiederebbe in ogni dispositivo una lista completa (tabella di instradamento) di tutti gli indirizzi MAC dei dispositivi nel mondo, con a fianco l'indicazione della direzione di inoltro. Ovviamente questo limiterebbe in modo critico la scalabilità e la crescita di Internet.

Una soluzione semplice consiste nell'elezione di un rappresentante per ogni rete locale X (il router di X), incaricato di ricevere tutti i pacchetti dati destinati a uno dei calcolatori della rete locale (es. mac1 di X, mac2 di X, ecc.). Ricevuti i pacchetti destinati alla rete locale, il router potrebbe occuparsi di recapitare alla rete locale i pacchetti, come se si trattasse di un frame a livello MAC/LLC destinato all'indirizzo MAC del destinatario. Allo stesso modo, ogni router dovrebbe farsi carico di inoltrare tutti i pacchetti uscenti dalla propria rete locale, verso i router delle reti di destinazione. Per rispettare le direttive dettate dallo standard ISO/OSI, il livello di indirizzamento e la gestione dell'instradamento dei pacchetti tra i router vengono gestiti al terzo livello (rete) della gerarchia dei protocolli di Internet.

Per ciò che riguarda i router, tuttavia, lo scambio diretto tra router di pacchetti destinati alle rispettive reti locali potrebbe ridurre molto la complessità dell'instradamento. I router comunicano quindi attraverso collegamenti dati molto veloci, dette dorsali (backbone). Ogni router deve ricordare in una tabella di instradamento (forwarding table) solo quale sia il primo router intermedio per raggiungere ogni altro router. La visione del sistema al terzo livello (Rete) da parte dei router è quindi simile alla visione che appare a destra nella figura. Si nota come tutti i dettagli delle reti locali siano di fatto nascosti dai router a questo livello.

FORWARDING: è il funzionamento che invia i pacchetti a destinazione

≠

ROUTING: è l'aggiornamento delle tabelle di instradamento

1° problema: tenere le road map aggiornate (percorso del pacchetto)

→ Ogni pacchetto contiene le informazioni sulla sua prossima destinazione dentro ad una tabella: TABELLA di INSTRADAMENTO

PROTOCOLLO di ROUTING → serve a mantenere aggiornate le tabelle di instradamento

nel protocollo IP la rete ha un numero e c'è anche il numero di host

Esempio IP: host z18 rete n7

Inizia a questo punto la trattazione degli aspetti di gestione del **livello rete**. Il concetto di rete non si limita ora alla sola rete locale (LAN) ma si estende alla rete di reti globale (Internet).

- Livello Rete (Network) per Internet:
 - protocollo Internet (Internet Protocol, IP)
 - nuovo tipo di indirizzamento globale e gerarchico (indirizzamento IP)
 - fornisce indirizzi alla rete locale e ai suoi nodi
 - instradamento dei pacchetti dal mittente al destinatario finale (forwarding)
 - Servizio di comunicazione di tipo connectionless
 - nuovi dispositivi amministratori del livello tre: router
 - tabelle di instradamento che illustrano la topologia della rete (livello Rete)
 - nasconde dettagli interni delle LAN al livello Rete
 - protocoli di aggiornamento delle tabelle di instradamento (routing)
 - frammentazione dei dati da spedire in pacchetti → n fragmenti di un pacchetto
 - **busta del pacchetto** di livello rete con gli indirizzi di mittente e destinatario

INDIRIZZO IP HOST NUMBER



Invece il numero dell'host: → sostitutivo del MAC perché è più compatto (il MAC richiede +8 bit)

Esempio: host 47 della rete z (LAN z)

CONVERSAZIONE tra indirizzo MAC e NUMERO DI HOST e viceversa: fatto quando il pacchetto già arrivato → non può più quindi

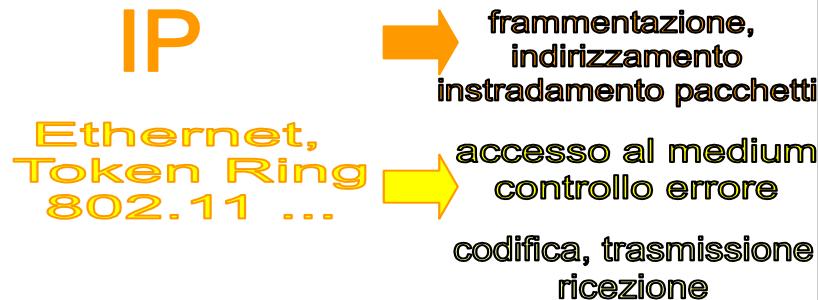
Copyright © Luciano Bononi 2017

ADDRESS RESOLUTION PROTOCOL

IPv4: indirizzo IP ha 32 bit

Reti di Calcolatori - introduzione

IPv6: indirizzo IP di 128 bit
version 6



La collocazione del livello rete, i servizi in esso implementati e il protocollo IP.

La figura mostra l'architettura a livelli dei protocolli di Internet visti finora. Al secondo livello troviamo i protocolli MAC, tra i quali Ethernet, Token ring, 802.11, e protocolli LLC, tra i quali HDLC e PPP. I servizi indicati al livello MAC/LLC sono essenzialmente l'accesso al mezzo trasmittivo e il controllo degli errori di trasmissione. Al terzo livello (rete) troviamo il protocollo IP. Sono indicati i servizi supportati dal protocollo IP a livello rete: frammentazione, indirizzamento e instradamento dei pacchetti dati.

→ La rete INTERNET è scalabile perché se si aumentano le dimensioni delle reti locali, non ha impatto a livello di internet

Reti di Calcolatori - introduzione

Inizia a questo punto la trattazione degli aspetti di gestione del terzo livello dell'architettura dei protocolli di Internet: il livello Rete (Network). Il concetto di rete non si limita ora alla sola rete locale (LAN) ma si estende alla rete di reti globale (Internet). Il livello rete di Internet si basa sul protocollo IP (Internet Protocol). Il protocollo IP definisce un nuovo schema di indirizzamento globale e gerarchico, che permette di identificare univocamente tutti i dispositivi di rete e allo stesso tempo la loro rete locale di appartenenza. Gli indirizzi usati permettono di identificare intere reti locali come un riferimento singolo nella gestione dell'instradamento dei pacchetti. Questo fatto semplifica molto la visione della rete che appare al livello Rete (si veda la figura della diapositiva precedente). Al protocollo IP, si possono associare protocolli di instradamento dei pacchetti dal mittente al destinatario finale (forwarding), originando servizi di trasmissione a pacchetto di tipo connectionless. Il protocollo IP richiede l'adozione di nuovi dispositivi amministratori dell'inoltro dei pacchetti a livello Rete, detti router. I router sono forniti di tabelle di instradamento che illustrano la topologia della rete vista al livello dei router stessi (quindi non è necessario conoscere gli indirizzi dei calcolatori di una LAN al di fuori della LAN stessa. I router devono implementare protocolli di aggiornamento delle tabelle di instradamento (detti protocolli di routing). Ulteriore compito dei router è la gestione della frammentazione dei dati da spedire nei pacchetti, e la creazione della busta di livello rete con gli indirizzi del router mittente e destinatario di ogni pacchetto inoltrato.

Vediamo ora come è definito l'indirizzamento caratteristico del protocollo IP di Internet: IPv4.

Indirizzamento IPv4 - IP versione 4

IPv4

- Il protocollo IP definisce una nuova specie di indirizzi: gli indirizzi IP
- un indirizzo IP viene associato a una e una sola interfaccia di rete (scheda di rete)
 - associazione univoca tra indirizzo MAC e indirizzo IP
 - IP statico (sempre lo stesso) e IP dinamico (può cambiare l'associazione MAC-IP)
- Gli indirizzi IP attualmente usati si riferiscono al protocollo IP versione 4, (**IPv4**)
 - Un indirizzo IPv4: 32 bit (4 Byte) = sequenza di 4 valori decimali separati da punto
 - Ogni valore decimale può essere compreso tra i valori 0 e 255
 - Esempio di indirizzo IP valido: 130.136.25.1
 - Indirizzo IP è sempre composto da due parti:
 - numero della rete IP alla quale appartiene la scheda (**network number**)
 - numero dell'interfaccia di rete (**host number**) all'interno della rete
 - il valore dell'indirizzo IP determina la **classe della rete**: A,B,C

Es. oggi mi compro,
poi mi disastro,
mi ricomprerò
domani e l'
indirizzo IP è
dinamico

Gli indirizzi IPv4 sono divisi
nelle famiglie di classe A, B, C

ogni classe ha il proprio "punto di taglio" tra
numero di rete e numero di host

Indirizzamento IPv4

- Il protocollo IP definisce una nuova specie di indirizzi: gli indirizzi IP
- un indirizzo IP viene associato a una e una sola interfaccia di rete (scheda di rete)
 - associazione univoca tra indirizzo MAC e indirizzo IP
 - IP statico (sempre lo stesso) e IP dinamico (può cambiare l'associazione MAC-IP)
- Gli indirizzi IP attualmente usati si riferiscono al protocollo IP versione 4, (**IPv4**)
 - Un indirizzo IPv4: 32 bit (4 Byte) = sequenza di 4 valori decimali separati da punto
 - Ogni valore decimale può essere compreso tra i valori 0 e 255 → come sono rappresentati
 - Esempio di indirizzo IP valido: 130.136.25.1 → se qua ci fosse 130.136.256.1 >255 non è valido (ma non è un indirizzo IP)
 - Indirizzo IP è sempre composto da due parti:
 - numero della rete IP alla quale appartiene la scheda (**network number**)
 - numero dell'interfaccia di rete (**host number**) all'interno della rete
 - il valore dell'indirizzo IP determina la **classe della rete**: A,B,C

→ Perché esistono queste due modalità?

- se fosse statico, significa che se una scheda di rete si collega ora e poi non lo sa più domani non è disponibile l'uso di quell'indirizzo IP (tanti indirizzi occupati da poche macchine connesse ad internet)

- Lo usiamo macchine che stanno attive su internet 24h/24 → SERVER
(E' necessario l'IP statico) → il SERVER è fisso, è sempre connesso allo stesso router e quindi conviene avere un indirizzo IP statico

- IP dinamico: metodologia che serve a cambiare l'associazione MAC-IP
→ dai telefoni, laptop - si muovono, cambiano spesso rete, quindi non hanno un indirizzo IP statico

Una rete è di classe A se il primo bit più a sinistra è uguale a 0

7 indirizzi di rete disponibili

Reti di Calcolatori - introduzione

Nella classe A, i restanti byte rappresentano il numero di host

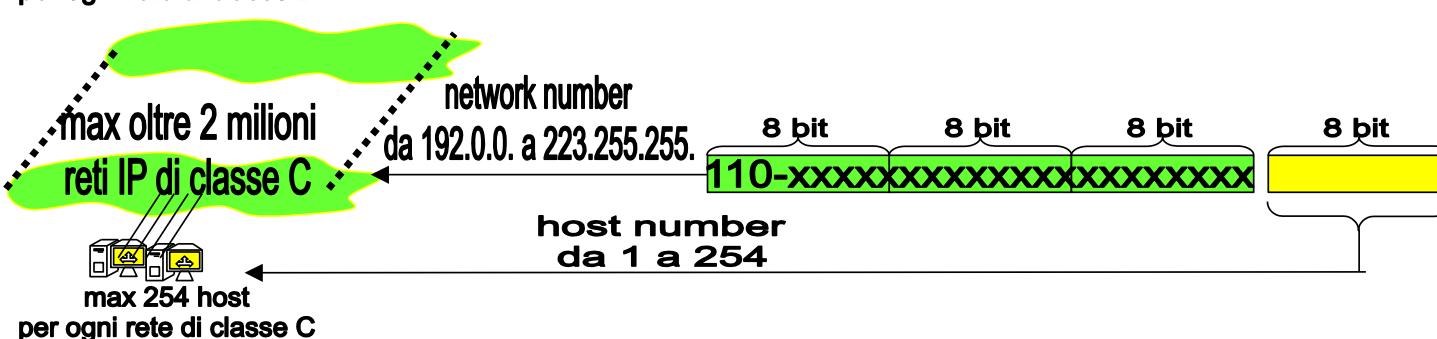
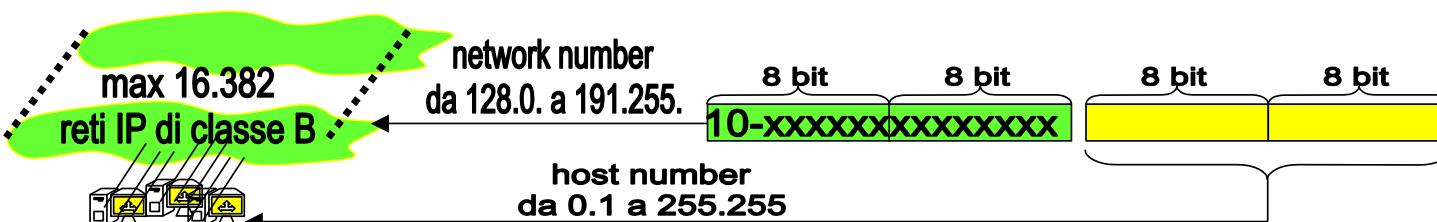
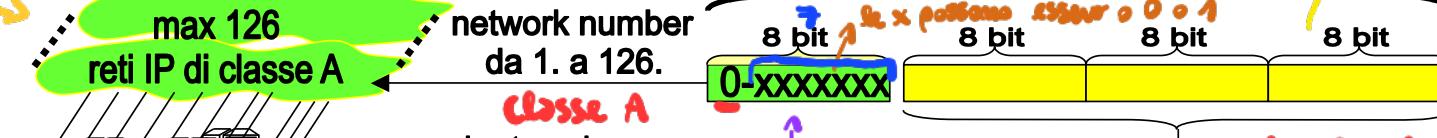
RETE DI CLASSE A
utile per molti host

- TELECOM
PROVIDER

USiamo questo
tipo di rete

3.8

24 host =



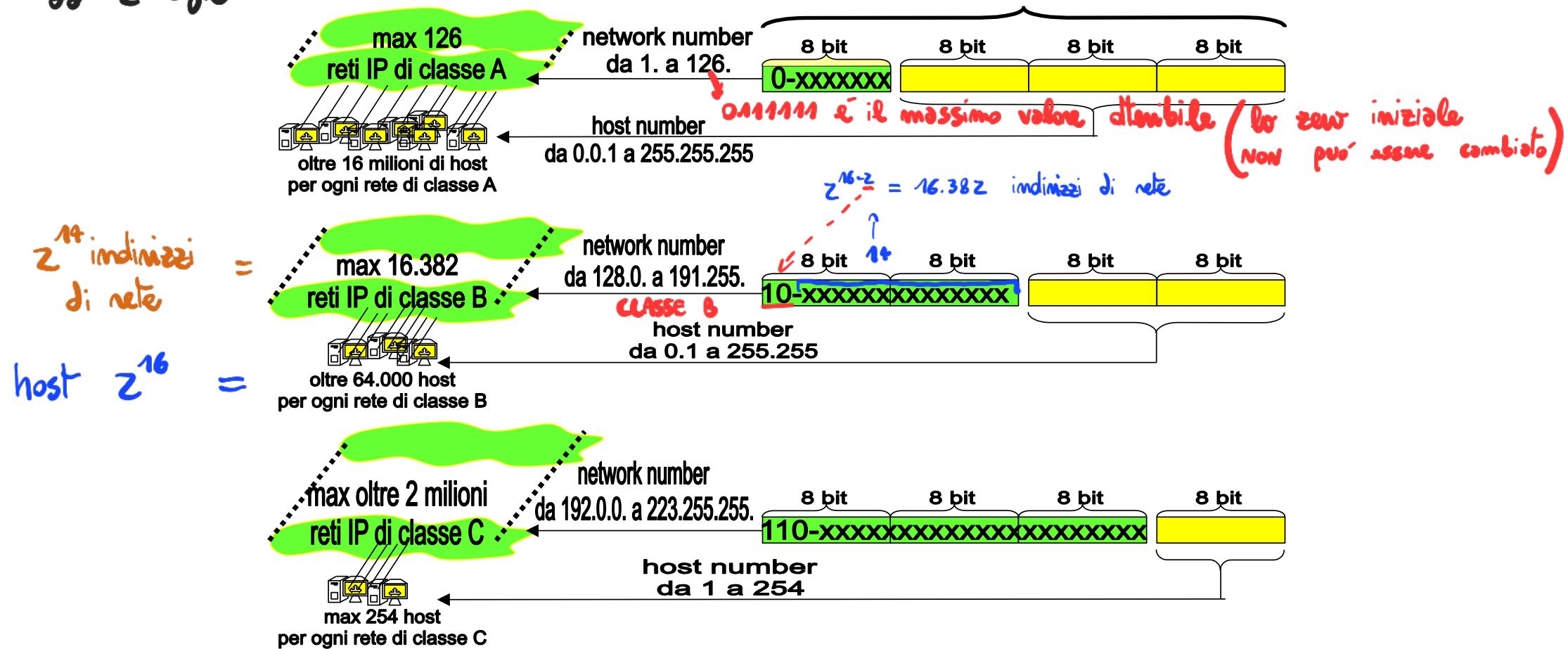
Una classificazione delle tre classi di reti, A, B e C, associate agli indirizzi IPv4.

La figura illustra uno schema di associazione degli indirizzi IPv4 alla classe di rete relativa. Le reti di classe A sono al massimo 126 e ognuna può contenere oltre 16 milioni di host. Per le reti di classe A, il byte di indirizzo più significativo (a sinistra) ha sempre il primo bit uguale a zero, e può assumere i valori da 1 a 126 (network number). I tre byte rimanenti possono assumere oltre 16 milioni di combinazioni, ognuna associabile a un host della rete. Le reti di classe B sono al massimo 16.382 e ognuna può contenere fino a oltre 64.000 host. Per le reti di classe B, i due byte di indirizzo più significativi (a sinistra) hanno sempre i primi due bit uguali alla coppia (uno,zero), e possono assumere i valori da 128.0. a 191.255. (network

Una rete è di classe B se un indirizzo IP dove il primo byte ha i primi due bit a 10 e l'indirizzo di rete è costituito da 2 byte

Reti di Calcolatori - introduzione

indirizzi IPv4



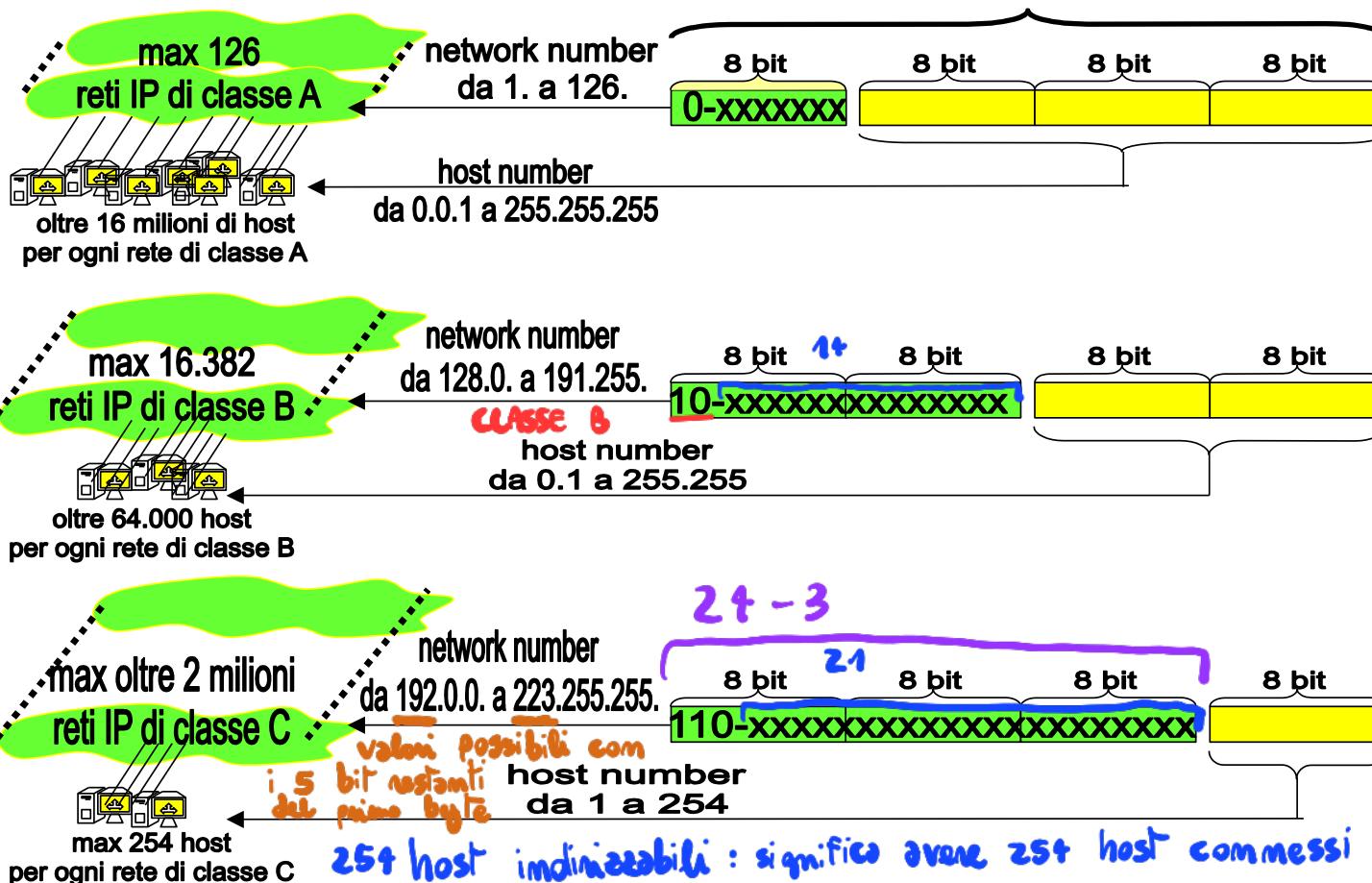
Una classificazione delle tre classi di reti, A, B e C, associate agli indirizzi IPv4.

La figura illustra uno schema di associazione degli indirizzi IPv4 alla classe di rete relativa. Le reti di classe A sono al massimo 126 e ognuna può contenere oltre 16 milioni di host. Per le reti di classe A, il byte di indirizzo più significativo (a sinistra) ha sempre il primo bit uguale a zero, e può assumere i valori da 1 a 126 (network number). I tre byte rimanenti possono assumere oltre 16 milioni di combinazioni, ognuna associabile a un host della rete. Le reti di classe B sono al massimo 16.382 e ognuna può contenere fino a oltre 64.000 host. Per le reti di classe B, i due byte di indirizzo più significativi (a sinistra) hanno sempre i primi due bit uguali alla coppia (uno,zero), e possono assumere i valori da 128.0. a 191.255. (network

Una rete è di classe C se un indirizzo IP dove il primo byte ha i primi tre bit a 110 e l'indirizzo di rete è costituito da 3 byte

Reti di Calcolatori - introduzione

indirizzi IPv4



Una classificazione delle tre classi di reti, A, B e C, associate agli indirizzi IPv4.

La figura illustra uno schema di associazione degli indirizzi IPv4 alla classe di rete relativa. Le reti di classe A sono al massimo 126 e ognuna può contenere oltre 16 milioni di host. Per le reti di classe A, il byte di indirizzo più significativo (a sinistra) ha sempre il primo bit uguale a zero, e può assumere i valori da 1 a 126 (network number). I tre byte rimanenti possono assumere oltre 16 milioni di combinazioni, ognuna associabile a un host della rete. Le reti di classe B sono al massimo 16.382 e ognuna può contenere fino a oltre 64.000 host. Per le reti di classe B, i due byte di indirizzo più significativi (a sinistra) hanno sempre i primi due bit uguali alla coppia (uno,zero), e possono assumere i valori da 128.0. a 191.255. (network number).

ESERCITAZIONE:

- INPUT: Valore decimale \rightarrow OUTPUT: Valore binario

=> Serve l'algoritmo della DIVISIONE:

- Divido INPUT per z (z perché è la base in cui voglio convertire INPUT)

Pseudocodice:

```
void Dividi (valore, base){
```

```
    valoreInt = valore div z // Prende la parte intera del risultato
```

```
    valoreResto[i].add = valore - valoreInt * z // Resto della divisione
```

```
    if (valoreInt > 0){
```

```
        Dividi (valore, base);
```

```
}
```

```
else {
```

```
    for (int i = size (valoreResto[ ]); i-- ; i < 0) { // I resti sono presi
```

```
        print (valoreResto[i]);
```

// al contrario, dall'

```
        print ("\n");
```

// ultimo al primo

```
}
```

```
}
```

Esempio 1:

Dividi (45, 2) i

$$\begin{aligned} 45 \text{ div } 2 &= 22 + 1 \\ 22 \text{ div } 2 &= 11 + 0 \\ 11 \text{ div } 2 &= 5 + 1 \\ 5 \text{ div } 2 &= 2 + 1 \\ 2 \text{ div } 2 &= 1 + 0 \\ 1 \text{ div } 2 &= 0 + 1 \end{aligned}$$

↑
lettura
del
numero

print del risultato: $(101101)_2 = (45)_{10}$

VERIFICA → CONVERSIONE DA BASE Z A BASE 10

⇒ Usa la FUNZIONE VALORE (metodologia inversa alla DIVISIONE)

$$\text{FUNZIONE VALORE } v(\text{base } 10) = \sum_{i=0}^{N-1} (b_i * \text{BASE}^i)$$

N = numero delle cifre della notazione in base BASE

i = posizione i-esima della cifra da DESTRA a SINISTRA

Esempio: verifica

Dati la notazione in BASE Z: 101101_2

$$V = 1 \cdot z^{i=0} + 0 \cdot z^1 + 1 \cdot z^2 + 1 \cdot z^3 + 0 \cdot z^4 + 1 \cdot z^5$$

$$V = 1 \cdot 1 + 0 \cdot 2 + 1 \cdot 4 + 1 \cdot 8 + 0 \cdot 16 + 1 \cdot 32$$

TRAESFORMA IN BINARIO I SEGUENTI VALORI (METODO DELLA DIVISIONE)

$$17_{10} \rightarrow ?$$

$$17 \text{ div } 2 = 8 \text{ resto } 1$$

$$8 \text{ div } 2 = 4 \text{ resto } 0$$

$$4 \text{ div } 2 = 2 \text{ resto } 0$$

$$2 \text{ div } 2 = 1 \text{ resto } 0$$

$$1 \text{ div } 2 = 0 \text{ resto } 1$$

$$\Rightarrow 10001$$

$$67_{10} \rightarrow 1000011_2$$

$$6^6 5^5 4^4 3^3 2^2 1^1 0^0$$

$$1 0 0 0 0 1 1$$



$$z^6 = 64$$

$$67 - 64 = 3 - 2 = 1 - 1 = 0$$

↑
Potenza di 2 più vicina a 67 (e poi a 3, e poi a 1)

$$98_{10} \rightarrow \begin{array}{c} 6^{\circ} & 1^{\circ} \\ \boxed{1100010} \\ 64 \end{array}$$

N.B.
 $2^0 = 1$ è l'UNICO valore (potenza di 2) che rende la cifra dispari

Ora somo a 64 (posiz. 6°)

$$64 + 2^5 = \text{si, si avvicina} \rightarrow 64 + 32 = 96$$

$$\text{Ora somo a } 96 + 2^1 = 98 \quad (\text{posiz. } 1^{\circ})$$

$$111_{10} \longrightarrow 1101111_2$$

$$275_{10} \longrightarrow 100010011_2$$

$$4895_{10} \longrightarrow 1$$

Per fare ciò è necessario imparare a memoria i numeri binari

1
2
4
6
8
16
32
64
128
256
512
1024
2048
4096

Domanda da esame:

• Traduci $465201_7 \rightarrow (?)_{10}$

1^o CHECK: Se tra i valori c'è almeno una cifra NON COMPRESA tra 0 e 6 (che sono i resti ammessi da $\% 7$) il numero in base 7 (in questo caso) non è ammesso.

$$1 \cdot 7^0 + 0 \cdot 7^1 + 2 \cdot 7^2 + 5 \cdot 7^3 + \dots$$

• Quanti bit servono per rappresentare 14917_{10} in binario? $\log_2(14917) = 13, \dots \approx 14$
 (A quale potenza devo elevare la base 2 per poter ottenere 14917_{10})

OTTAUE:

(recapitri)

ESADECIMALE:

(recapitri)

DOMANDA: $534_7 = (?)_3$

Per fare applico la FUNZIONE VALORE a 534_7 in base 10
Poi applico l'algoritmo della divisione

Per passare VELOCEMENTE da base z ad una base z^k
c'è un METODO

Si parte dalla destra del numero binario (cifre meno significative)
raggruppando in k sottogruppi i valori binari e poi convertito da binario
alla base di destino in ogni gruppo

Esempio in base $16 = z^4 \rightarrow$ raggruppamenti da 4 bit
padding \rightarrow bit inutili aggiunti

0010 1001

VALE ANCHE AL CONTRARIO!

Es. $4AFD$ (hex) $\rightarrow z^{16}$ in binario?
 \Rightarrow

4	A	F	D
0100	1010	1111	1101

N.B.: Indirizzo MAC puo' essere espresso

ff:ff:ff:ff:ff:ff (MAC address di broadcast) - 48 bit

Reti di Calcolatori - introduzione

number). I due byte rimanenti possono assumere oltre 64.000 combinazioni, ognuna associabile a un host della rete. Le reti di classe C sono oltre 2 milioni, e ognuna può contenere fino a 254 host. Per le reti di classe C, i tre byte di indirizzo più significativi (a sinistra) hanno sempre i primi tre bit uguali alla terna (uno,uno,zero), e possono assumere i valori da 192.0.0 a 223.255.255 (network number). Il byte rimanente può assumere 254 combinazioni utili, ognuna associabile a un host della rete.

Il protocollo IP definisce una nuova specie di indirizzi: gli indirizzi IP. Gli indirizzi IP attualmente usati si riferiscono al protocollo IP versione 4, (IPv4). Un indirizzo IP viene associato a una e una sola interfaccia di rete (scheda di rete), e più precisamente esiste un rapporto diretto tra un indirizzo MAC dell'interfaccia di rete e un indirizzo IP, dal momento che l'interfaccia è collegata alla rete Internet. Se un calcolatore dispone di più di una scheda di rete, ed è contemporaneamente collegato a Internet mediante entrambe le interfacce, allora il calcolatore potrebbe essere contemporaneamente identificato da più di un indirizzo IP. Se l'associazione univoca tra indirizzo MAC dell'interfaccia di rete e l'indirizzo IP rimane sempre lo stesso, allora si parla di IP statico. In caso contrario, se può cambiare l'associazione MAC-IP a seconda di vari fattori, si parla di IP dinamico. Un indirizzo IPv4 è un valore espresso su 32 bit (4 Byte), e può essere espresso anche come sequenza di 4 valori decimali, separati da punto. Ogni valore decimale può essere compreso tra i valori 0 e 255. Un esempio di indirizzo IP valido è (130.136.250.1). Ogni indirizzo IP è sempre composto da due parti: numero della rete IP alla quale appartiene la scheda (network number), e numero dell'interfaccia di rete (host number) all'interno della rete IP. Sono definite tre classi di reti IP, che si differenziano sulla base del numero massimo di host supportabili. Il valore dell'indirizzo IP determina la classe della rete: A,B,C (vedere figura).

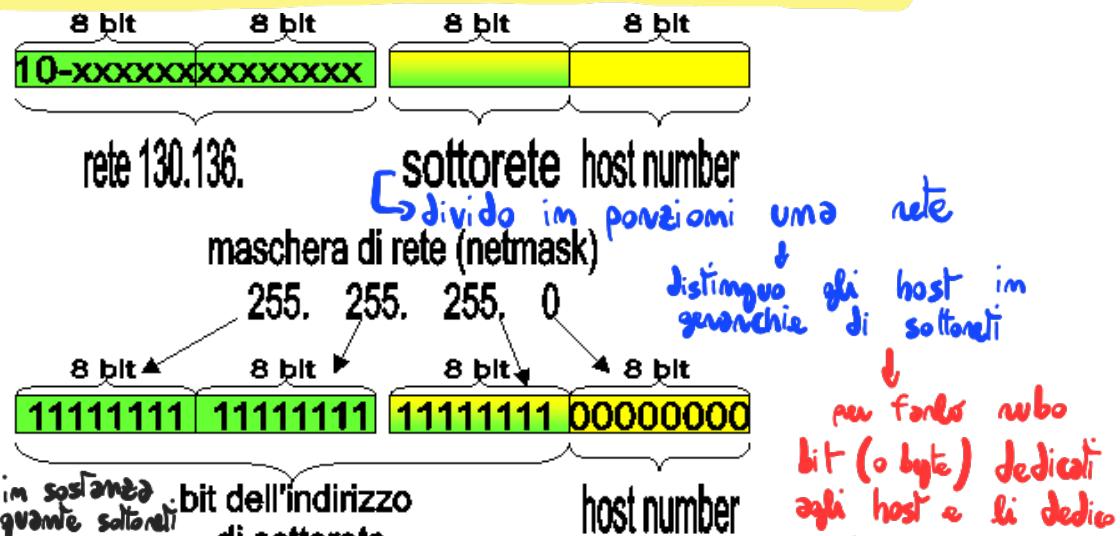
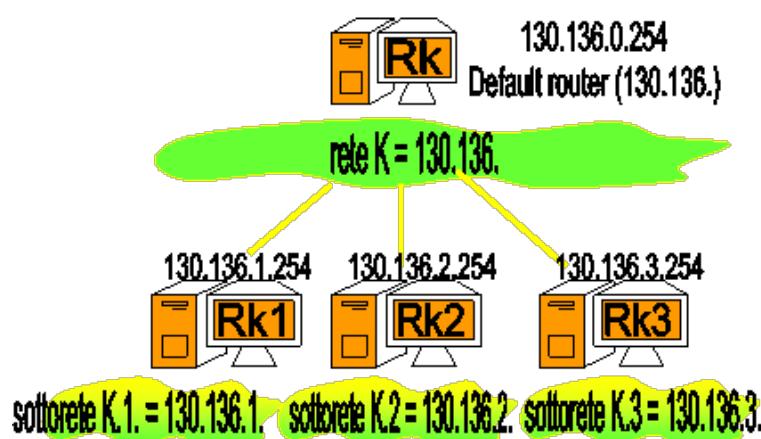
Le reti di classe A sono al massimo 126 e ognuna può contenere fino a oltre 16 milioni di host. Per le reti di classe A, il byte di indirizzo più significativo (a sinistra) ha sempre il primo bit uguale a zero, e può assumere i valori da 1 a 126 (network number) rispetto ai 128 valori possibili. I tre byte rimanenti possono assumere oltre 16 milioni di combinazioni, ognuna associabile a un host della rete. Le reti di classe B sono al massimo 16.382 e ognuna può contenere fino a oltre 64.000 host. Per le reti di classe B, il network number è dato dai due byte di indirizzo più significativi (a sinistra), che hanno sempre i primi due bit uguali alla coppia (uno,zero). I network number di classe B possono assumere i valori da 128.0. a 191.255. I due byte rimanenti (host number) possono assumere oltre 64.000 combinazioni, ognuna associabile a un host della rete. Le reti di classe C sono oltre 2 milioni, e ognuna può contenere fino a 254 host. Per le reti di classe C, i tre byte di indirizzo più significativi (a sinistra) rappresentano il network number, e hanno sempre i primi tre bit uguali alla terna (uno,uno,zero). I network number di classe C possono assumere i valori da 192.0.0 a 223.255.255. Il byte rimanente (host number) può assumere 254 combinazioni utili, su 256 possibili, ognuna associabile a un host della rete.

→ **Sistema scalabile:** un sistema può essere più o meno scalabile se riesce a gestire / o memo
Copyright © Luciano Bononi 2017 un maggiore traffico di dati

Sottoreti (subnetwork)

Reti IP e router sono organizzati secondo una gerarchia, basata sul concetto di rete e sottorete

- indirizzo IP spezzato in due componenti logiche mediante **maschera di rete** (netmask)
 - indirizzo di sottorete (subnetwork): bit uguali a uno nella netmask
 - host number dell'host appartenente alla sottorete: bit uguali a zero nella netmask
- gerarchia di sottoreti, ognuna delle quali è amministrata da un router (il **default router**).
- Esempio: rete di classe B 130.136. con 256 sottoreti: netmask 255.255.255.0
 - es. 130.136.1. è la sottorete 1, 130.136.2 è la sottorete 2...
 - es. 130.136.1.22 è l'host 22 della sottorete 1, 130.136.3.48 è l'host 48 della sottorete 3...
- Indirizzo IP, netmask e default router sono **informazioni di configurazione del livello rete**



Un esempio di rete IP di classe B con tre sottoreti.

MASCHERA
DI
RETE

Copyright © Luciano Bononi 2017
Valore espresso su 4 cifre decimali (valori fra 0 - 255) dove vige
le seguenti regole: il primo bit a sinistra si 1

Reti di Calcolatori - introduzione

La figura mostra a sinistra uno schema gerarchico di strutturazione di una rete di classe B. A partire dall'alto troviamo il router principale (default router) della rete 130.136, il cui IP è 130.136.0.254. Alla rete 130.136 appartengono anche tre router subordinati, con IP 130.136.1.254, 130.136.2.254, 130.136.3.254 rispettivamente amministratori delle sottoreti (130.136.1.), (130.136.2.) e (130.136.3.). Per istruire ogni router subordinato sulla dimensione e sull'interpretazione degli indirizzi IP da amministrare, ogni router subordinato deve essere fornito di una maschera di rete (netmask), evidenziata a destra. La maschera di rete serve solo a definire quali bit degli indirizzi IP vadano interpretati come numero della sottorete (indirizzo della sottorete), cioè i bit uguali a uno, partendo da sinistra. I bit della maschera uguali a zero (a destra) servono invece a descrivere quali bit degli indirizzi IP vadano considerati come il numero dell'host (host number) appartenente alla sottorete. Nell'esempio i 3 Byte a sinistra della maschera sono tutti a uno, quindi i primi 24 bit dell'indirizzo IP rappresentano il numero di sottorete, come risulta dal disegno di sinistra.

Se pongo questo zero nella parte indirizzo di rete
 ↓
 z° sottoreti
 ↓ a seconda che quel bit sia 0/1
 Per ciascuna sottorete posso avere ... host

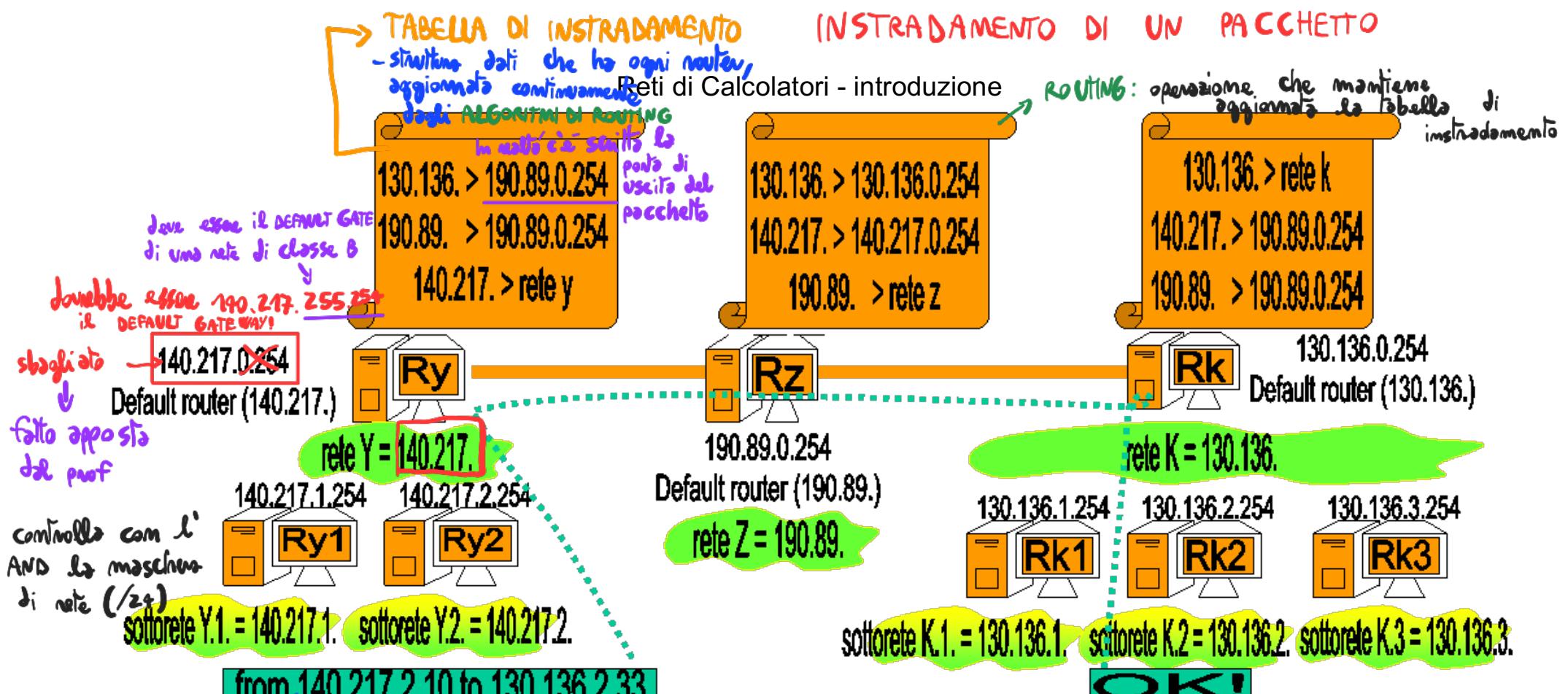
I router e le reti IP sono organizzate secondo una gerarchia, basata sul concetto di rete e sottorete. Ogni indirizzo IP può essere spezzato in due componenti logiche attraverso la maschera di rete (netmask). La prima componente logica è un indirizzo di sottorete (subnetwork), che corrisponde ai bit con la stessa posizione dei bit uguali a uno nella netmask. La seconda componente logica è il campo host number che identifica gli host appartenenti alla sottorete, e corrisponde ai bit corrispondenti ai bit uguali a zero nella netmask. In questo modo è possibile creare una gerarchia di sottoreti, ognuna delle quali è amministrata da un router (il default router). Esempio: data la rete di classe B 130.136, per semplicità decidiamo di considerare possibili 256 sottoreti: netmask 255.255.255.0. Il numero della sottorete è quindi fornito dai primi tre byte dell'indirizzo IP, es. 130.136.1. è la sottorete 1, 130.136.2. è la sottorete 2, mentre ad esempio 130.136.1.22 è l'host 22 della sottorete 1, 130.136.3.48 è l'host 48 della sottorete 3, e così via.

La figura mostra a sinistra uno schema gerarchico di strutturazione di una rete di classe B. A partire dall'alto troviamo il router principale (default router) della rete 130.136, il cui indirizzo IP è nell'esempio 130.136.0.254. Alla rete 130.136 appartengono anche tre router subordinati, con IP 130.136.1.254, 130.136.2.254, 130.136.3.254 rispettivamente amministratori delle sottoreti (130.136.1.), (130.136.2.) e (130.136.3.). Per istruire ogni router subordinato sulla dimensione e sull'interpretazione degli indirizzi IP da amministrare, ogni router subordinato deve essere fornito di una maschera di rete (netmask), evidenziata a destra. Per ogni dispositivo di rete o calcolatore connesso a una rete IP (es. Internet), la maschera di rete, l'indirizzo del default router e l'indirizzo IP, costituiscono i tre parametri fondamentali di configurazione del protocollo IP, e devono essere forniti, al momento della connessione, al livello IP.

Instradamento (forwarding) dei pacchetti

- esempio: host 140.217.2.10 spedisce pacchetto IP a host 130.136.2.33
 - pacchetto inviato da host 140.217.2.10 a host 130.136.2.33
 - raccolto dal default router della sottorete Ry2: 140.217.2.254
 - destinatario non appartiene alla sottorete: inoltrato verso il default router di livello superiore: 140.217.0.254
 - raccolto dal default router 140.217.0.254
 - destinatario 130.136.--: tabella di forwarding indica di inviare a 190.89.0.254: inoltrato
 - raccolto da 190.89.0.254
 - dest. 130.136.--: tabella di forwarding indica di inviare a 130.136.0.254: inoltrato
 - raccolto da 130.136.0.254
 - dest. 130.136.--: la tabella di forwarding indica che appartiene a questa rete (k)
 - inoltrato alla rete 130.136.--
 - raccolto dal router 130.136.2.254
 - la tabella indica che dest. appartiene alla sottorete Rk2: 130.136.2. : inoltrato
 - raccolto dall'host destinatario finale 130.136.2.33: OK!

INSTRADAMENTO DI UN PACCHETTO



Un esempio di instradamento su rete IP

La figura mostra un esempio di instradamento su rete IP. Esistono tre router Ry, Rz e Rk rispettivamente amministratori delle reti (140.217.), (190.89.), e (130.136.). Il router Ry è connesso a Rz, e Rz è connesso a Rk. Tale informazione risulta dalle tabelle di instradamento di Ry, Rz, Rk. La rete del router Ry include due sottoreti (140.217.1.) e (140.217.2), amministrate dai rispettivi default router 140.217.1.254 e 140.217.2.254. La rete del router Rk include tre sottoreti (130.136.1.), (130.136.2.) e (130.136.3.), amministrate dai rispettivi default router 130.136.1.254, 130.136.2.254 e 130.136.3.254. Un pacchetto IP inviato dall'host 140.217.2.10 all'host 130.136.2.33 deve compiere il seguente tragitto: passa per il default router di sottorete 140.217.2.254 che lo inoltra al default router di rete 140.217.0.254. Esso controlla la tabella di forwarding e scopre che per raggiungere la destinazione il pacchetto deve essere inoltrato al router intermedio 190.89.0.254. Il router intermedio riceve il pacchetto, verifica la propria tabella di forwarding, scopre che il prossimo destinatario intermedio è il router 130.136.0.254 e vi inoltra il pacchetto.

1) È nella mia sottorete? No! → va al DEFAULT GATE della rete/sottorete
che decide se inoltrarlo fuori dalla rete o meno

Copyright © Luciano Bononi 2017

Reti di Calcolatori - introduzione

Il router 130.136.2.254 riceve il pacchetto e verifica che appartiene alla propria rete, inoltrando quindi il pacchetto internamente. Il router di sottorete 130.136.2.254 riceve il pacchetto e scopre che appartiene alla propria sottorete, inoltrandovi il pacchetto. Finalmente, l'host 130.136.2.33 riceve il pacchetto a lui destinato.

La figura mostra un esempio di instradamento su rete IP. Esistono tre router Ry, Rz e Rk rispettivamente amministratori delle reti (140.217.), (190.89.), e (130.136.). Il router Ry è connesso a Rz, e Rz è connesso a Rk. Tale informazione risulta dalle tabelle di instradamento di Ry, Rz, Rk. La rete del router Ry include due sottoreti (140.217.1.) e (140.217.2), amministrate dai rispettivi default router 140.217.1.254 e 140.217.2.254 . La rete del router Rk include tre sottoreti (130.136.1.), (130.136.2.) e (130.136.3.), amministrate dai rispettivi default router 130.136.1.254, 130.136.2.254 e 130.136.3.254 . Un pacchetto IP spedito dall'host 140.217.2.10 all'host 130.136.2.33 deve compiere il seguente tragitto: passa per il default router di sottorete 140.217.2.254 che, notando che il destinatario non appartiene alla sottorete, lo inoltra al default router del livello superiore di rete: 140.217.0.254. Il default router di rete controlla la propria tabella di forwarding e scopre che per raggiungere la destinazione il pacchetto deve essere inoltrato al router intermedio 190.89.0.254. Il router intermedio riceve il pacchetto, verifica la propria tabella di forwarding, scopre che il prossimo destinatario intermedio è il router 130.136.0.254, al quale inoltra il pacchetto. Il router 130.136.2.254 riceve il pacchetto e verifica che appartiene alla propria rete, inoltrando quindi il pacchetto internamente. Il router di sottorete 130.136.2.254 riceve il pacchetto e scopre che appartiene alla propria sottorete, inoltrando il pacchetto internamente. Finalmente, l'host 130.136.2.33 riceve il pacchetto a lui destinato.

Si possono notare alcuni aspetti importanti: malgrado il numero elevato di host che potrebbero essere parte del sistema considerato, il processo di instradamento permane molto semplice, composto da piccole e semplici operazioni di base. Le tabelle di instradamento sono limitate agli elementi che agiscono allo stesso livello, e quindi l'instradamento è gerarchico.

Routing

azione fatta da protocolli a livello RETE.

Questa azione consiste nell' aggiornare le tabelle di istrundamento/routing

Il problema del routing: aggiornamento delle tabelle di forwarding dei router

- modifiche dei cammini per i dati nella rete
 - possibili soprattutto in reti senza fili a causa della mobilità degli host
 - causate da modifiche agli accordi di servizio tra gestori di sistemi autonomi (AS)
- le modifiche dei cammini rendono sbagliate le tabelle di forwarding dei router
 - i pacchetti possono andare perduti, o seguire strade diverse e arrivare disordinati
- occorrono reazioni da parte dei router per scoprire nuovi cammini
 - ~~andò se que sono usati in modo intercambiabile~~ tutto sotto la stessa gestione amministrativa
 - protocolli (algoritmi) di routing:
 - invio richieste: qualcuno conosce il modo per arrivare al destinatario?
 - Aggiornamento della tabella di forwarding con il cammino migliore
- Esempio di algoritmi di routing su Internet: Routing Information Protocol (RIP), Open Shortest Path First (OSPF), Border Gateway protocol (BGP)

↳ bordo dell' AS

→ Il primo pacchetto da inviare è quello che "apre la strada" e trova il cammino di istrundamento consultando le tabelle (di istrundamento).

Gli altri pacchetti successivi ci metteranno di meno perché in cache hanno le informazioni per arrivare a destinazione, non devono cercarle

(Sarà già grazie
al primo pacchetto,
che le ha già cercate
e trovate)

Reti di Calcolatori - introduzione

•

Il problema del routing può essere definito come il problema di mantenere l'aggiornamento delle tabelle di forwarding in tutti i router della rete.

Questo problema può essere a volte molto complesso, a causa di frequenti modifiche forzate dei cammini per i pacchetti in rete. Le cause di tali modifiche possono essere dovute a molti fattori, ad esempio: la mobilità degli host in reti senza fili, guasti di mezzi trasmissivi, interruzione delle linee, guasti di router, nuove politiche e accordi per lo scambio dei dati tra gestori di dorsali e sistemi autonomi. Un sistema autonomo (AS) è sinonimo di una grossa rete, o una collezione di reti, soggetta a una comune politica di amministrazione. Gli accordi commerciali tra gestori di AS possono modificare i cammini consentiti per lo scambio dei pacchetti di dati. Per realizzare un parallelo intuitivo, gli AS si comportano come nazioni che permettano o meno il passaggio di pacchetti, analoghi a voli aerei, sul loro suolo nazionale.

Ogni volta che si verifica uno dei problemi citati, esistono router che hanno indicazioni errate nelle loro tabelle di forwarding. Tutto ciò può causare la perdita di pacchetti, oppure può determinare un disordine nell'arrivo di pacchetti che hanno seguito strade diverse. Ecco quindi una causa del servizio connectionless ottenuto dal livello rete basato solo sul protocollo IP.

I router hanno bisogno di aggiornare al più presto le loro tabelle, per evitare malfunzionamenti del servizio. I protocolli di routing hanno la funzione di richiedere e scambiare informazioni per trovare cammini alternativi (idealmente il cammino migliore tra le possibili alternative), tra mittenti e destinatari dei pacchetti, e consentire quindi l'aggiornamento delle tabelle di forwarding. A puro titolo informativo, si citano alcune sigle di protocolli di routing adottati in Internet: Routing Information Protocol (RIP), Open Shortest Path First (OSPF), Border Gateway protocol (BGP).

Protocollo ICMP

→ Protocollo usato dagli host per capire come sta andando la rete e usato dai router per aggiornare le tabelle con informazioni mancanti

Internet Control Message Protocol (ICMP); protocollo dei messaggi di controllo su Internet

(tutto avviene a livello RETE)

- Uno standard per definire la comunicazione di informazioni utili alla gestione di Internet
- ICMP è usato da host, router e gateway per scambiare informazioni di livello rete, usando pacchetti definiti con il protocollo IP

di non-funzionamento dei collegamenti

- Rete di destinazione non raggiungibile (possibile interruzione di rete?)
- Rete di destinazione sconosciuta (indirizzo di rete male specificato?)
- Host destinazione non raggiungibile (host spento o scollegato?)
- Host destinazione sconosciuto (indirizzo di host male specificato?)
- Protocollo richiesto non disponibile (servizi non previsti)
- Ricerca di un cammino alternativo per la destinazione (se esiste)

→ posizione di rete ?
→ router scollegato ?
→ La rete specificata è probabilmente

host che conosco,
conosco il MAC ADDRESS

Es. dispositivo disconnesso
dalla rete nel momento in cui
provo a raggiungerlo
(ma magari era connesso)

Reti di Calcolatori - introduzione

Vediamo ora un esempio di protocollo, basato sul protocollo IP, definito per supportare i messaggi di controllo per la gestione della rete al terzo livello dello stack ISO/OSI (Rete). Il protocollo in questione si chiama Internet Control Message Protocol (ICMP), ovvero protocollo dei messaggi di controllo su Internet. ICMP è un protocollo standard, e quindi può essere adottato da parte di tutti i dispositivi per fornire uno strumento generalizzato e compatibile. ICMP viene usato da semplici host, da router e persino da gateway (router speciali con ulteriori funzioni) per scambiare informazioni utili alla gestione del livello Rete. Le informazioni scambiate sono trasferite sotto forma di pacchetti IP.

Alcuni esempi di messaggi ICMP che possono essere scambiati indicano, ad esempio: situazioni di rete di destinazione irraggiungibile (possibile sintomo di problemi di routing, o di rottura di un router), rete di destinazione sconosciuta (possibile sintomo di indirizzo IP di rete male specificato), host di destinazione non raggiungibile (possibile sintomo che l'host sia spento o il cavo di connessione sia male collegato), host di destinazione sconosciuto (possibile sintomo di host number dell'indirizzo IP male specificato, malgrado la rete indicata esista e sia raggiungibile), protocollo richiesto non disponibile (sintomo di un tentativo di dialogo tra dispositivi male configurati, che non forniscono i servizi richiesti), ricerca di cammino alternativo (può essere usato per risolvere i problemi di routing).

Applicazioni basate su ICMP

↳ processi applicativi che possono essere eseguiti su shell

- applicazione **PING**: test di verifica di connessione tra due host
 - esempio: eseguire "ping <indirizzo IP host2>" sull'host1 (sulla shell dell'host1)
 - host1 invia richieste ICMP e host2 risponde (eco), se esiste ed è raggiungibile
 - viene calcolato il tempo di andata e ritorno delle richieste (Round Trip Time, RTT)
- applicazione **Traceroute**: mostra la sequenza di router attraversati da un pacchetto per arrivare all'host destinazione (solo la sequenza dell'andata)
 - esempio: eseguire "traceroute <indirizzo IP host2>" sull'host1 (shell host1)
 - È realizzato con messaggi ICMP spediti in sequenza, verso distanze via via maggiori

serve per verificare se la rete funziona tra due host e inoltre cronometra il tempo di andata e ritorno (RTT - round trip time) del pacchetto + ricezione del messaggio

```

E:\ Prompt dei comandi E: invia un pacchetto dal mio host al
    10.2.2.255.255.255.255, riesce? AND SOONER OR LATER
Esecuzione di Ping 130.136.2.241 con 32 byte di dati:
Risposta da 130.136.2.241: byte=32 durata<1ms TTL=128
Statistiche Ping per 130.136.2.241:
    Pacchetti: Trasmessi = 4, Ricevuti = 4, Persi = 0 (0% persi),
    Tempo approssimativo percorsi andata/ritorno in millisecondi:
        Minimo = 0ms, Massimo = 0ms, Medio = 0ms
C:\>ping 130.136.10.241
Esecuzione di Ping 130.136.10.241 con 32 byte di dati:
Richiesta scaduta.
Richiesta scaduta.
Richiesta scaduta.
Richiesta scaduta.
Statistiche Ping per 130.136.10.241:
    Pacchetti: Trasmessi = 4, Ricevuti = 0, Persi = 4 (100% persi),
C:\>
  
```

(RTT → quando si è a zero → il pacchetto viene varcato)


```

C:\>tracert 193.206.134.21
Rilevazione instradamento verso mi-bo-g.garr.net [193.206.134.21]
su un massimo di 30 punti di passaggio:
  1  <1 ms   <1 ms   <1 ms  csgw-3-0-5.cs.unibo.it [130.136.2.254]
  2  <1 ms   <1 ms   <1 ms  cesia-csgw.cs.unibo.it [130.136.254.254]
  3  <1 ms   <1 ms   <1 ms  alga11.unibo.it [137.204.2.17]
  4  <1 ms   <1 ms   <1 ms  rtg-unibo.bo.garr.net [193.206.128.125]
  5  4 ms    4 ms    4 ms  mi-bo-g.garr.net [193.206.134.21]
  6 * * * → router
Rilevazione completata. nascosto
  
```

(indirizzi IP visti (rete del CESIA))

NAT: indirizzi IP virtuali usate da internet:
10.x.x.x
192.168.x.x

→ NAT SERVER
converte tali indirizzi virtuali a livello trasporto in maniera dipende

Copyright © Luciano Bononi 2017

TTL - Time to leave : è un campo di un pacchetto IPv4 (del livello rete)

quando si è a zero → il pacchetto viene varcato
In questo modo i pacchetti che girano a vuoto vengono

Reti di Calcolatori - introduzione

Alcuni esempi di esecuzione: applicazione PING e traceroute (tracert).

Sono mostrate due finestre di esecuzione delle applicazioni PING e traceroute (tracert). La figura di sinistra mostra l'applicazione PING in esecuzione verso un host di indirizzo IP 130.136.2.241. Il testo mostrato è il seguente: "Esecuzione di Ping 130.136.2.241 con 32 Byte di dati:", seguito da 4 invii di richieste ping ognuna delle quali raggiunge con successo l'host destinazione. I messaggi mostrati infatti dicono: "Risposta da 130.136.2.241: byte=32 durata < 1ms TTL=128". Al termine delle 4 richieste viene mostrato il riepilogo delle statistiche: "Numero pacchetti trasmessi = 4, pacchetti ricevuti = 4, persi = 0 (0%persi). Tempo approssimativo percorsi andata/ritorno in millisecondi: minimo 0ms, massimo 0ms medio 0ms". di seguito, sulla stessa finestra, viene mostrato un esempio di ping alla macchina 130.136.10.241 (non collegata alla rete). Il testo questa volta dice: : "Esecuzione di Ping 130.136.10.241 con 32 Byte di dati:", seguito da 4 invii di richieste ping ognuna delle quali non raggiunge con successo l'host destinazione. I messaggi mostrati infatti dicono: "Richiesta scaduta". Al termine delle 4 richieste scadute viene mostrato il riepilogo delle statistiche: "Numero pacchetti trasmessi = 4, pacchetti ricevuti = 0, persi = 4 (100%persi). La figura di destra mostra il tracciato di indirizzi IP dei router attraversati nel cammino dall'host 130.136.2.241 all'host 193.206.134.21. Il testo dice: "C:\>tracert 193.206.134.21" che è il comando inviato, seguito dalla risposta dell'applicazione tracert: "Rilevazione instradamento verso mi-bo-q.garr.net [193.206.134.21] su un massimo di 30 punti di passaggio:", seguito da 5 righe simili a: "1 <1ms <1ms <1ms csgw-3-0-5.cs.unibo.it [130.136.2.254]" ognuna relativa a un router diverso, fino ad arrivare all'host finale: "1 <1ms <1ms mi-bo-q.garr.net [193.206.134.21]", seguito da "Rilevazione completata".

Reti di Calcolatori - introduzione

Esistono due applicazioni di servizio che sono basate sui messaggi ICMP. Tali applicazioni sono molto utili per la verifica delle cause o del semplice sospetto di problemi di rete. Le applicazioni sono l'applicazione PING e l'applicazione Traceroute.

L'applicazione PING permette di testare la connessione tra due host: eseguendo il comando “ping <indirizzo IP di host2>” da un host1 qualsiasi (mittente) connesso in rete, l'applicazione invia una richiesta ICMP di eco, alla quale l'host2 indicato risponde con una risposta ICMP (eco della richiesta). Dopo l'invio della richiesta, host1 fa partire un timer. In caso di successo, viene calcolato il tempo di andata e ritorno dei pacchetti (durata o Round Trip Time, RTT), mentre in caso di insuccesso viene indicato che il timer per la richiesta inviata è scaduto senza ottenere risposta (secondo esempio della prima figura). Al termine dei tentativi, viene mostrato un elenco di statistiche sul numero di richieste andate a buon fine e i tempi medi stimati di andata e ritorno dei pacchetti.

L'applicazione Traceroute (tracert) permette di verificare la lista di tutti i router attraversati da una richiesta ICMP inviata da host1 a host2. Il comando “tracert <indirizzo IP di host2>” eseguito da host1, causa l'invio di una sequenza di richieste ICMP verso host2, per le quali viene fissato il numero massimo di router da attraversare (numero di passaggi o tempo di vita, TTL), a valori crescenti da 1 in poi. Ogni router a distanza TTL risponde con un messaggio ICMP di errore (tempo di vita scaduto) attraverso il quale è possibile risalire al suo indirizzo IP (ognuno mostrato su righe successive). Un esempio è mostrato nella seconda figura.

protocollo ARP e RARP

Quando un router riceve un pacchetto destinato a un indirizzo IP della propria sottorete, esso deve produrre un frame di livello MAC/LLC che riporti specificato l'indirizzo MAC del destinatario (e non l'indirizzo IP), per poterlo passare al livello MAC/LLC per la trasmissione sulla rete locale.

- Protocollo Address Resolution Protocol (ARP) *→ Risoluzione nel senso che dato un indirizzo IP → ci indica il MAC corrispondente*
 - Usato se il router non conosce l'indirizzo MAC corrispondente all'indirizzo IP
 - Il router genera un frame spedito a tutti i dispositivi della rete locale dove si chiede: "quale è l'indirizzo MAC del dispositivo che ha questo indirizzo IP"?
 - Se tale dispositivo esiste, esso risponde con un frame di livello MAC indirizzato al router, nel quale viene evidenziato l'indirizzo MAC richiesto
- Protocollo Reverse-ARP (RARP)
 - È la versione opposta del protocollo ARP, ma funziona allo stesso modo
 - La domanda è "quale indirizzo IP corrisponde al dispositivo con questo indirizzo MAC"?

• ARP → manda un broadcast in una rete locale e cerca tra tutti gli host quello indicato dall'indirizzo IP → tale host può esserci o meno

→ MAC ADDRESS ff:ff:ff:ff → MAC ADDRESS di BROADCAST
 Un host a livello MAC sa se un pacchetto proveniente dal livello rete sia suo sse o c'è un indirizzo MAC di BROADCAST o c'è l'indirizzo MAC specifico di quell'host

→ Tutti gli host hanno la tabella ARP (il protocollo ARP è presente nel sistema operativo)

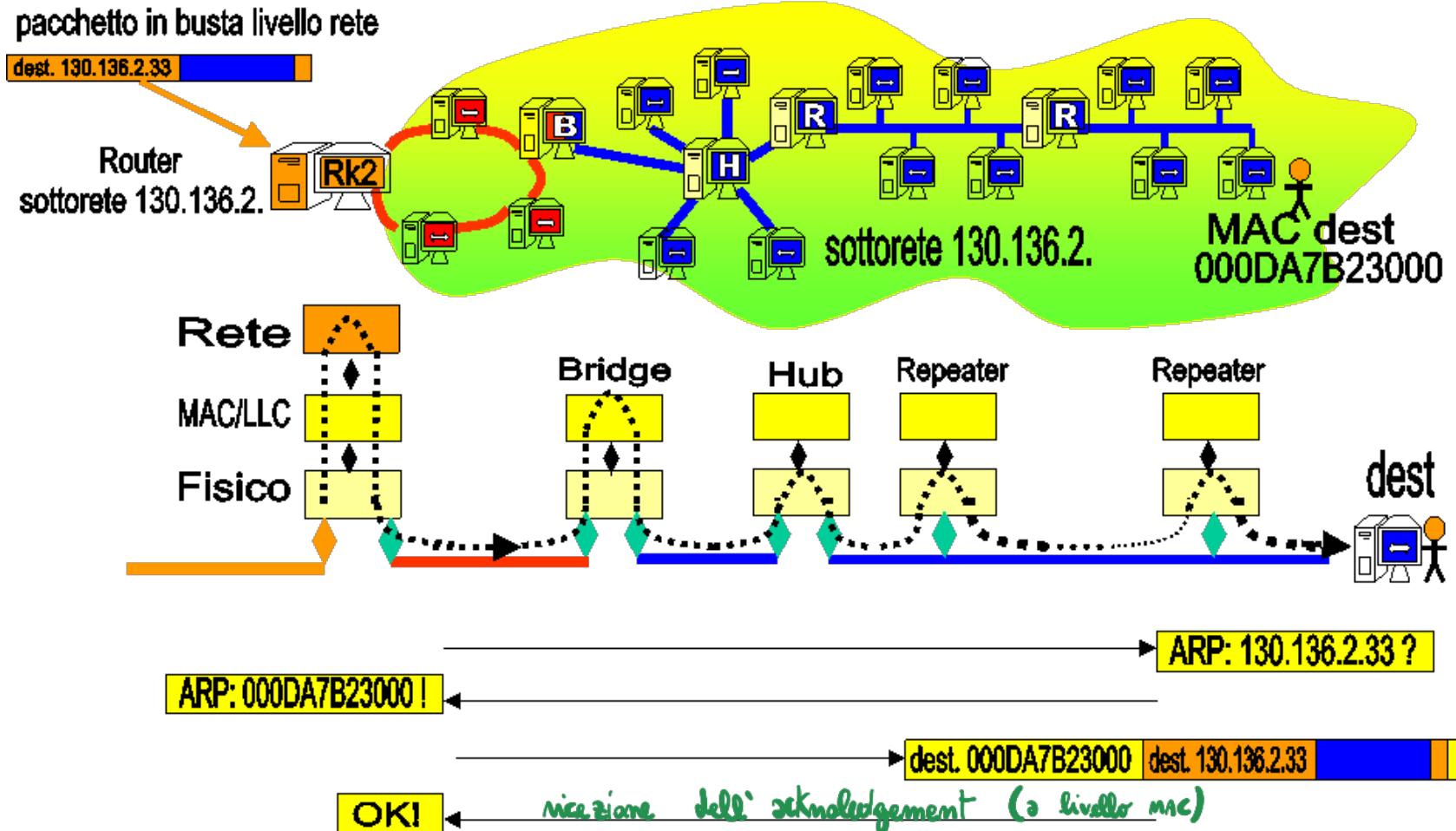
(comando arp -a)

• em0 (dopo il MAC ADDRESS) è un nome logico che identifica un protocollo di rete

• Ho fatto un ping ad un certo indirizzo IP e ho risposto ma non compare nella tabella ARP ma non compare?

→ Perché tale indirizzo IP pingato non fa parte della mia rete locale
(Allora viene inviato al DEFAULT GATEWAY, di cui so già il MAC ADDRESS)
(si occupava poi lui di inviare eventualmente pacchetti al di fuori della rete locale)

Reti di Calcolatori - introduzione



Un'illustrazione della consegna di un pacchetto IP al destinatario finale, usando il protocollo ARP.

La figura mostra la fase di consegna di un pacchetto IP da parte del router all'host destinatario appartenente alla sottorete. Si tratta dell'illustrazione del passo finale descritto nella diapositiva che illustra il forwarding a livello IP. Il router che riceve un pacchetto a livello IP destinato alla sua sottorete (130.136.2.33) spedisce sui segmenti della rete locale un frame in broadcast (cioè ricevuto da tutti i dispositivi) contenente il codice di richiesta ARP, e l'indirizzo IP del destinatario del pacchetto. Il destinatario in questione, se esiste, risponde con un frame indirizzato all'indirizzo MAC del router, con allegato l'indirizzo MAC richiesto. A questo punto il router può quindi preparare la busta di livello MAC/LLC,

Reti di Calcolatori - introduzione

indirizzata al MAC del dispositivo destinatario del pacchetto IP, e contenente il pacchetto IP incapsulato all'interno. Il destinatario riceve il frame e risponde con il frame di conferma per il sottolivello LLC.

Si è visto come i router si occupino di gestire l'inoltro dei pacchetti a livello rete, e si è visto come la gestione basata su indirizzamento IP permetta di nascondere i dettagli interni delle reti locali, al di fuori della rete locale stessa. Si è visto anche che all'interno delle reti locali, la trasmissione su un segmento di rete locale avviene mediante frame di livello MAC/LLC, indirizzati mediante gli indirizzi MAC dei dispositivi, e non attraverso gli indirizzi IP. Un router deve quindi saper gestire l'associazione tra indirizzo IP di un dispositivo a livello rete e il suo indirizzo MAC a livello MAC/LLC. Il protocollo Address Resolution Protocol (ARP) risponde in modo standard a questa esigenza. In altre parole il protocollo ARP è il protocollo che lega l'indirizzamento a livello MAC con l'indirizzamento a livello IP.

Quando un router riceve un pacchetto a livello IP destinato alla sua sottorete (es. 130.136.2.33) esso verifica se a tale IP risulti o meno associato un indirizzo MAC. In caso contrario, il router spedisce sui segmenti della rete locale un frame in broadcast (cioè ricevuto da tutti i dispositivi) contenente il codice di richiesta ARP, e l'indirizzo IP del destinatario del pacchetto. Tale frame equivale quindi al rivolgere a tutti i dispositivi la domanda: "quale indirizzo MAC ha il dispositivo corrispondente al seguente indirizzo IP"? Il dispositivo in questione, se esiste, risponde con un frame indirizzato all'indirizzo MAC del router, contenente il codice di risposta ARP, e con allegato l'indirizzo MAC richiesto.

A questo punto il router può quindi preparare e spedire la busta di livello MAC/LLC, indirizzata al MAC del dispositivo destinatario del pacchetto IP, contenente il pacchetto IP incapsulato all'interno. Il destinatario riceve il frame e risponde con il frame di conferma per il sottolivello LLC.

Esiste anche una versione analoga del protocollo ARP, detta Reverse-ARP, che risponde alla domanda: "quale indirizzo IP corrisponde al dispositivo con questo indirizzo MAC"?

Assegnazione indirizzi IP: DHCP

- Assegnazione dei **numeri di rete** di classe A, B, C
 - vengono assegnati a enti, aziende ed organizzazioni che ne fanno richiesta, da parte di enti internazionali (RIPE, ICANN, ARIN, APNIC).
- Assegnazione dei **numeri di host** ai dispositivi di una rete
 - Assegnazione manuale da parte dell'amministratore di rete
 - associa manualmente indirizzi IP e indirizzi MAC (IP statico)
 - Assegnazione automatica da parte di un **server Dynamic Host Configuration Protocol (DHCP)**: indirizzi IP dinamici
 - Metodo automatico usato in reti **wireless**, reti locali e connessioni domestiche
 - **Server DHCP**: è un **host** che implementa il servizio di assegnazione dell'indirizzo IP agli **host** che ne fanno richiesta
 - Un **DHCP** server dispone di un blocco di **host number** liberi per la sua rete
 - **DHCP** server associa indirizzi IP a indirizzi MAC dei dispositivi che lo richiedono
 - I dispositivi devono scegliere di affidarsi al **server DHCP** (“ottieni automaticamente indirizzo IP”)

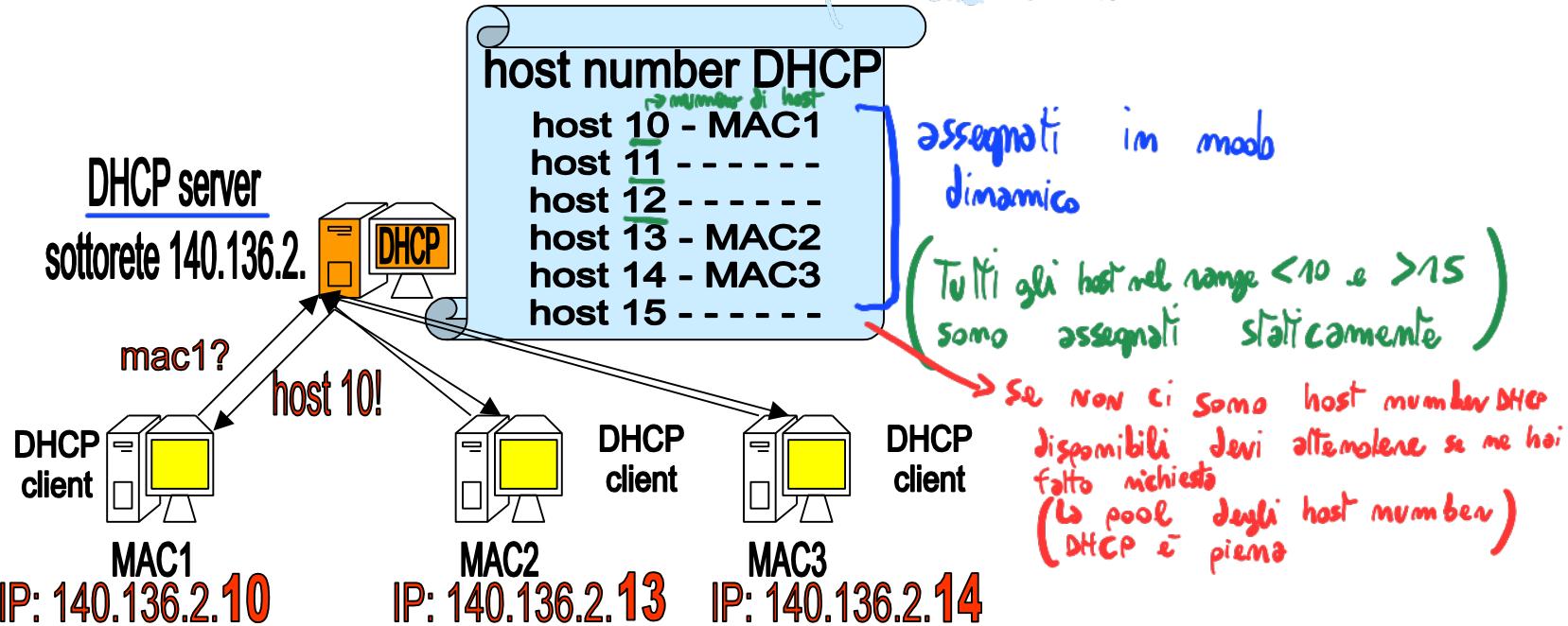
Se io come host ho un indirizzo IP sono connesso a internet e posso comunicare a livello globale

Avere un indirizzo IP vuol dire appartenere ad un dominio di un router, che vuol dire conoscere l'indirizzo del router e la maschera di rete, ecc...

Ma come vengono assegnati gli indirizzi IP?

- due modalità
 - statica
 - dinamica → tramite PROTOCOLLO DHCP

tabella presente nel
DHCP server



segnale di
LIVENESS
↓

messaggio per
capire se un
dispositivo è ancora
lì (è un protocollo)

"intesa" la
comunicazione
allora c'è

LEASING TIME: Un indirizzo DHCP viene dato assieme ad un tempo di LEASING: è il tempo di durata di un indirizzo host DHCP assegnato; quando scade questo tempo → Se io fisicamente cambio la posizione rispetto alla rete, il mio indirizzo IP viene cambiato, il router, tramite DHCP deve indicare al DNS a quale nuovo indirizzo IP devo far "imbattere" le informazioni del vecchio indirizzo IP
• sovraccarico di lavori i router
→ soluzione: mobile IP

Reti di Calcolatori - introduzione

I numeri di rete delle classi A, B e C, ovvero la parte sinistra degli indirizzi IP vengono assegnati da enti internazionali quali RIPE, ICANN, ARIN, APNIC a enti, aziende, consorzi e imprese che ne fanno richiesta motivata.

Un problema molto più pratico riguarda il modo in cui un nuovo dispositivo che venga connesso a una rete esistente, veda associare al proprio indirizzo MAC un indirizzo IP della rete stessa. Il numero di rete o di sottorete viene automaticamente determinato dall'appartenenza alla rete, ovvero alla presenza al di sotto del dominio di gestione di un router.

La prima, ovvia alternativa (molto usata) è quella di avere un amministratore di rete che assegna manualmente uno dei numeri di host disponibili al nuovo indirizzo MAC. In questo modo l'associazione indirizzo MAC e indirizzo IP può essere mantenuta per un tempo indeterminato, e quindi si considera l'indirizzo IP come statico.

La seconda alternativa, molto usata in reti senza fili, in reti locali e nei collegamenti domestici a Internet Service Provider (ISP) via Modem o ADSL consiste nell'utilizzare un server per Dynamic Host Configuration Protocol (DHCP).

Il server DHCP è dotato di una lista di numeri di host liberi per la sottorete amministrata, che provvede ad associare su richiesta agli indirizzi MAC dei dispositivi che lo richiedono. Tale associazione dipende spesso dalla disponibilità degli indirizzi già assegnati in precedenza, quindi allo stesso indirizzo MAC possono essere associati di volta in volta indirizzi IP diversi, e si parla in questo caso di indirizzi IP dinamici.

E' possibile configurare attraverso DHCP anche altri parametri di rete, come la maschera di rete, il default router e il server DNS (che vedremo dopo).

Il servizio DHCP equivale spesso al concetto di rete "plug and play", ovvero rete in cui basta connettere il dispositivo al medium e non c'è bisogno di nessuna configurazione manuale aggiuntiva.

- Il DOMINIO di una RETE/SOTTORETE è un nome logico che viene assegnato al corrispondente indirizzo di rete/sottorete

IPv6 e tunnelling IPv4

Internet è costituito da indirizzi IPv4
 → e un po' di indirizzi IPv6 (in aumento)
 (transizione lenta dai protocolli IPv4 a IPv6)

- Dal 1990 è attiva la definizione e l'implementazione di una nuova versione del protocollo di indirizzamento IP: **IPv6**
- Motivazioni per la definizione di IPv6: Indirizzi IPv4 finiranno negli anni 2008-2018.
- Caratteristiche salienti di IPv6
 - **Indirizzi IPv6:** estesi a 128 bit (16 Byte) anziché i 32 bit (4 Byte) di IPv4
 - circa 15000 indirizzi IPv6 per ogni metro quadrato di tutta la superficie terrestre!
 - Nuova struttura dei campi della busta dei pacchetti di livello rete (IP)
 - Identificazione di parametri per differenziare flussi di dati a priorità diverse
- Integrazione di IPv4 o sostituzione di IPv4?
 - Per ora la sperimentazione IPv6 avviene su reti separate (nuovi router IPv6)
 - Ci sono casi di integrazione tra IPv4 e IPv6 usando tecniche di tunnelling
 - **Tunnelling:** spedire pacchetti IPv6 in buste IPv4.

TOS (4 bit) di IPv4: Type of service → pensato per differenziazione il servizio di rete che si voleva avere

- poteva essere impiegato per dare priorità diversa ai pacchetti, ma non è stato per quello

 → Oggi tutti i router hanno due interfacce di rete, → si imposta una in locale ma in tutto il mondo non si fa!

 In IPv6 la priorità è misurata

una IPv4 e una IPv6
 , caso in cui ho una sola interfaccia di rete, il router si dice **dual stack**
 e al livello rete ha due stack di rete bifunzionali: uno IPv4 e uno IPv6

→ Quando un host (**client**) si connette so se sta usando IPv4 o IPv6 a seconda dell'indirizzo dato dal DHCP server

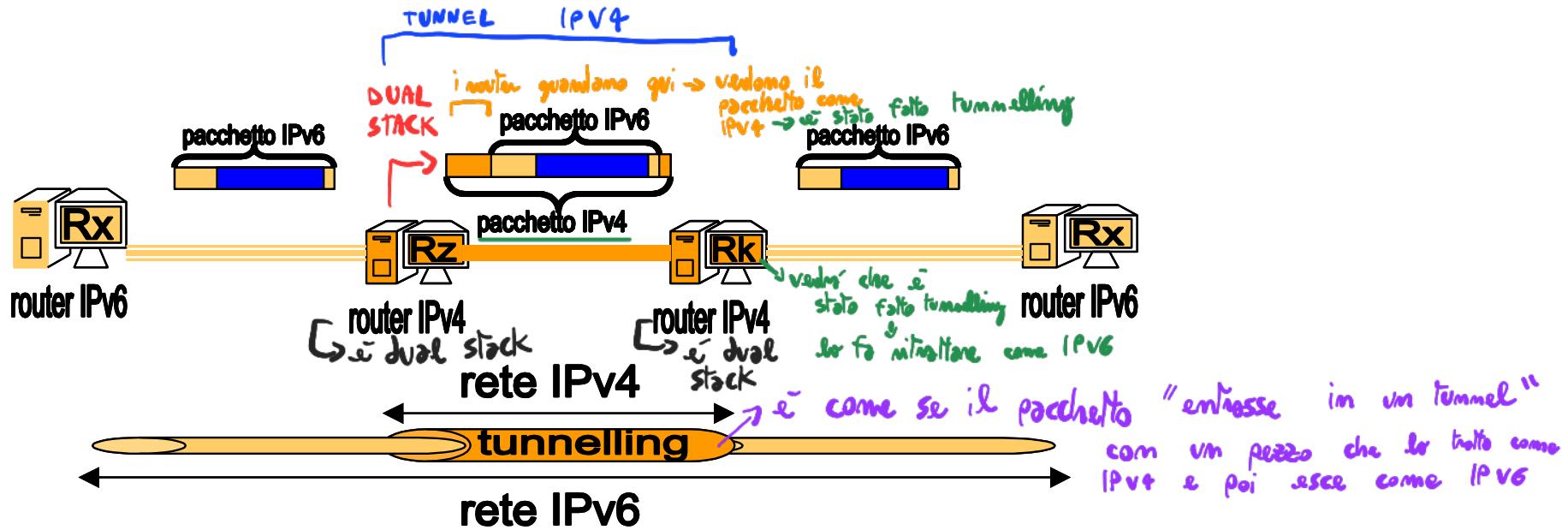
indirizzi IPv4

32 bit (4 Byte)



indirizzi IPv6

128 bit (16 Byte)



Un confronto in scala tra indirizzi IPv4 e IPv6, e un esempio di tunnelling IPv6 in IPv4.

La figura mostra in alto una scala dimensionale degli indirizzi IPv4 di 32 bit rispetto agli indirizzi IPv6 di 128 bit (4 volte più grandi). In basso viene mostrato come sia possibile spedire pacchetti IPv6 tra due router IPv6, passando per cammini che includono router IPv4, attraverso il tunnelling IPv6 in IPv4. Il pacchetto IPv6 viene incapsulato dai router IPv4 in pacchetti IPv4, in modo da poter essere instradato lungo la rete di router IPv4. Uscito dal tunnel IPv4 il pacchetto IPv6 prosegue il suo inoltro fino alla destinazione IPv6 finale.

NOVITA':

Con IPv6 si elimina la FRAMMENTAZIONE a livello rete dei pacchetti
 Copyright © Luciano Bononi 2017

- Meno pacchetti da gestire a livello rete => velocità maggiore
 basta analizzare a livello rete

Reti di Calcolatori - introduzione

Dal 1990 è stato avviato un progetto di definizione e sviluppo di una nuova versione del protocollo IPv4, denominato versione IPv6. In seguito all'esplosione del collegamento di calcolatori in rete, e quindi dell'utilizzo di indirizzi e reti IP, le proiezioni mostrano che al ritmo attuale gli indirizzi IPv4 saranno esauriti nel decennio 2008-2018.

Brevemente, le caratteristiche salienti di IPv6 vanno nella direzione di ovviare a questo problema, oltre a migliorare alcuni aspetti di IPv4.

La caratteristica fondamentale di IPv6 è la definizione di nuovi indirizzi IPv6 composti da 128 bit (16 byte), cioè ben quattro volte la dimensione degli indirizzi IPv4. Questo incredibile numero di indirizzi potrebbe consentire di avere circa 15000 indirizzi IPv6 per dispositivi diversi su ogni metro quadrato di superficie dell'intero pianeta, oceani inclusi.

Sono inoltre stati ridefiniti i campi che costituiscono la busta dei pacchetti di livello IPv4, aggiungendo ad esempio parametri per la gestione di flussi di pacchetti IP con diversi livelli di priorità.

Purtroppo la definizione di IPv6 nella maggioranza dei casi non permette di continuare a usare i vecchi router IPv4, e quindi non è compatibile con l'attuale struttura di Internet. La sperimentazione e lo sviluppo di IPv6 sta procedendo su reti IPv6 separate, che possono in certi casi integrarsi alle reti IPv4 usando la tecnica del tunnelling dei pacchetti IPv6 in IPv4.

Nella figura viene mostrato come sia possibile spedire pacchetti IPv6 tra due router IPv6 passando per cammini che includono router IPv4, attraverso il tunnelling IPv6 in IPv4. Il pacchetto IPv6 viene incapsulato dai router IPv4 in pacchetti IPv4, in modo da poter essere instradato lungo la rete di router IPv4. Uscito dal tunnel IPv4 il pacchetto IPv6 prosegue il suo inoltro fino alla destinazione IPv6 finale.

TUNNELLING: tecnica per spedire pacchetti IPv6 che vengono trattati come pacchetti IPv4

→ Dico che il pacchetto deve essere trattato come IPv4, nonostante il contenuto del pacchetto è IPv6 → Il router vede che il contenuto è IPv6 e il pacchetto viene "buttato" sulle stack di rete IPv6

ESERCITAZIONE: 22/11/2024

Esercitazione su formattazione con requisiti di PERFORMANCE ENGINEERING
(LIVELLO MAC)

Un FRAME di livello MAC ha dimensione programmabile di m bit. (Io posso scegliere io il valore di m)

- La probabilità di errore di un bit sbagliato sul canale fisico è $\underline{P_{BS}} = 10^{-9} \rightarrow$ cioè $1/1.000.000.000 \rightarrow$ è una statistica (\neq probabilità)
↓ indice di prestazione sulla qualità della trasmissione

→ Query

- Q. Che dimensione massima m potrò avere il frame perché la probabilità di un frame ricevuto come sbagliato sia $P_{FS} \leq x$? $[0 < x < 1]$
- Es. $P_{FS} = 1/1000 \rightarrow$ probabilità di avere un frame sbagliato

$$P_{BS} = 10^{-9} \quad (\text{Probabilità che un bit inviato sbagliato})$$

$$P_{BC} = 1 - 10^{-9} \quad (\text{Probabilità che un bit inviato corretto})$$

Assumiamo che gli errori sui bit consecutivi sono indipendenti

$$P_{FS} = 1 - P_{FC}$$

$$\hookrightarrow \text{Probabilità di avere un FRAME CORRETTO } P_{FC} = \text{tutti gli } m \text{ bit corretti} = (P_{BC})^m$$

si può fare perché i bit sono indipend.

$$1 - (P_{bc})^m \leq 1/1000$$

$$-(P_{bc})^m \leq 1/1000 - 1$$

$$(P_{bc})^m \geq 1 - 1/1000$$

Voglio trovare m quindi faccio il logaritmo:

$$\log(P_{bc})^m \geq \log(1 - 1/1000)$$

$$m * \log(P_{bc}) \geq \log(0,999)$$

$$m \leq \log(0,999) / \log(1 - P_{bs})$$

$$m \leq \log(0,999) / \log(1 - 10^{-9})$$

$\Rightarrow m$ e' il valore della dimensione del frame tale che avrei probabilita' di un bit sbagliato ogni $1.000.000.000$ bit (P_{bs})

SECONDA PARTE:

Se usassi una matrice di bit di parita' nella struttura del frame?

Quale sarebbe il valore di m che garantisce un frame sbagliato massimo ogni 1000 (frame)?

Sia m il numero totale dei bit del frame

La dimensione della matrice di parita' sono \sqrt{m} quindi agli m bit devo aggiungere $z * \sqrt{m}$ bit di parita'

Se m è la dimensione del frame, allora i dati utili saranno:

$$\Rightarrow m - z^* \sqrt{m}$$

(...)

$$1 - (P_{bc})^m + m * (P_{bc})^{m-1} * (1 - P_{bc}) \leq 1/1000$$

Livello trasporto

cerca di realizzare due forme di comunicazione:
 ↗ servizio trasporto affidabile
 ↗ servizio trasporto non affidabile

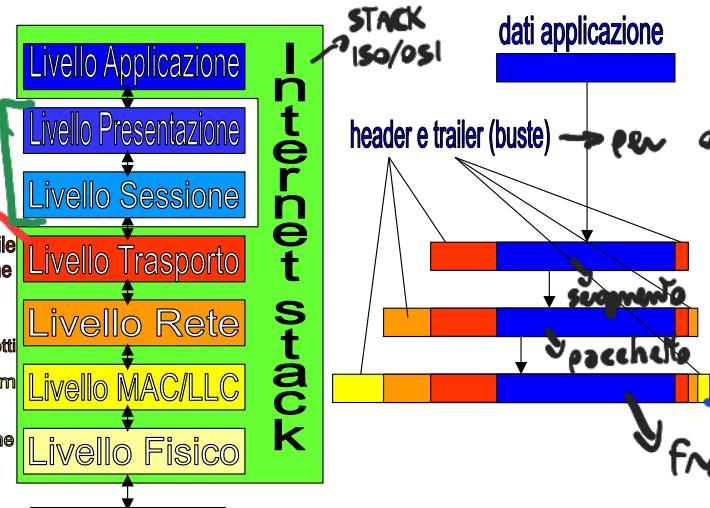
I protocolli di Internet di quarto livello (trasporto) sono essenzialmente il **Transmission Control Protocol (TCP)** e lo **User Data Protocol (UDP)**.

- Livello Trasporto: protocolli TCP e UDP

- Servizio **trasporto affidabile** (protocollo TCP): servizio di tipo connection-oriented
 - configurazione del **numero di porta** (port number) del **socket TCP**
 - numerazione sequenziale dei pacchetti, riordino, eliminazione duplicati
 - Pacchetti di conferma della ricezione (acknowledgment di livello trasporto)
 - Pacchetti non ricevuti (non confermati entro timeout) spediti di nuovo
 - gestione della congestione e controllo di flusso dei pacchetti.
 - Meccanismi a finestra scorrevole (sliding window)
- Servizio **trasporto non affidabile** (protocollo UDP): servizio connectionless

questi due livelli sono implementati a livello applicazione

Nello stack di rete si puo' passare direttamente dal livello trasporto al livello applicazione



per vedere se mancano dei segmenti e recuperare quelli perduti

se dopo un certo tempo il mittente non riceve un segnale di acknowledgement in mittente invia il segmento in

il problema dei duplicati: se il timeout scatta e l'acknowledgement arriva, il mittente invia un duplicato di un certo pacchetto (che è "numerato")

Reti di Calcolatori - introduzione

La collocazione del livello Trasporto, i servizi in esso realizzati e i protocolli TCP e UDP.

La figura mostra l'architettura a livelli dei protocolli di Internet visti finora. Al secondo livello troviamo i protocolli MAC, tra i quali Ethernet, Token ring, 802.11, e protocolli LLC, tra i quali HDLC e PPP. I servizi indicati al livello MAC/LLC sono essenzialmente l'accesso al mezzo trasmissivo e il controllo degli errori di trasmissione. Al terzo livello (rete) troviamo il protocollo IP. Sono indicati i servizi supportati dal protocollo IP a livello rete: frammentazione, indirizzamento e instradamento dei pacchetti dati. Al quarto livello troviamo i protocolli TCP e UDP. TCP in particolare realizza i servizi di connessione orientata alla connessione e controllo della congestione.

, Se si vuole un trasporto di rete NON AFFIDABILE

UDP → definisce le regole di trasporto dei segmenti
TCP → connection - oriented comunicazione da "estremo" a "estremo" del comminio
connessione end to end (cioè ovunque sia il mittente ed il destinatario, il messaggio arriva per intero al destinatario)

- Una architettura di comunicazione che

consente di parlare con tutti in modo

Copyright © Luciano Bononi 2017 affidabile è IP TCP
(E' l'architettura di internet)
→ usa IP TCP come protocolli

livello rete

livello trasporto

Reti di Calcolatori - introduzione

Il quarto livello dei protocolli dell'architettura di Internet è il livello trasporto (*transport*), ed è basato su due protocolli in particolare: il **Transmission Control Protocol (TCP)** e lo **User Data Protocol (UDP)**, che possono essere usati in alternativa tra loro.

Il servizio di trasporto dei pacchetti ottenuto mediante il protocollo TCP è quello di tipo **connection-oriented**: malgrado la rete a livello IP sia non affidabile (cambia l'ordine, duplica, o perde i pacchetti dati spediti), il protocollo TCP si occupa di garantire il ripristino dell'ordinamento dei pacchetti e la ri-trasmissione dei pacchetti perduti. TCP numera sequenzialmente tutti i dati spediti, e in questo modo permette alla sua controparte di procedere al riordino dei pacchetti che dovessero giungere disordinati, ad esempio (pacchetto 10), (pacchetto 9). Inoltre, attraverso il numero d'ordine, la controparte può inviare a sua volta pacchetti di conferma della ricezione (**acknowledgment**, simili, ma da non confondere con gli **acknowledgment** di livello LLC), es. (conferma 10, conferma 9). Grazie al meccanismo di conferma della ricezione dei pacchetti numerati, il mittente può accorgersi di pacchetti non ricevuti e inviarli nuovamente, anche più volte, finché non riceve la conferma attesa.

TCP si basa sul principio di connessione punto a punto tra punti virtuali detti **socket**, che permettono di smistare i pacchetti verso le rispettive applicazioni di livello superiore. Questo punto sarà evidenziato nella diapositiva successiva.

TCP permette infine di controllare la velocità di invio dei flussi dei pacchetti, mediante il meccanismo a finestra scorrevole, in modo da evitare la congestione della rete. Anche questo punto sarà ripreso in una diapositiva successiva.

In alternativa, il protocollo UDP non fa nulla di tutto questo, a parte lo smistamento verso le applicazioni di livello superiore. L'unico aspetto positivo di UDP è la sua estrema semplicità e il fatto che non aggiunge pesanti gestioni all'invio di pacchetti. Il servizio ottenuto, tuttavia, rimane un servizio di tipo **connectionless**, simile a quello fornito dal livello rete con IP.

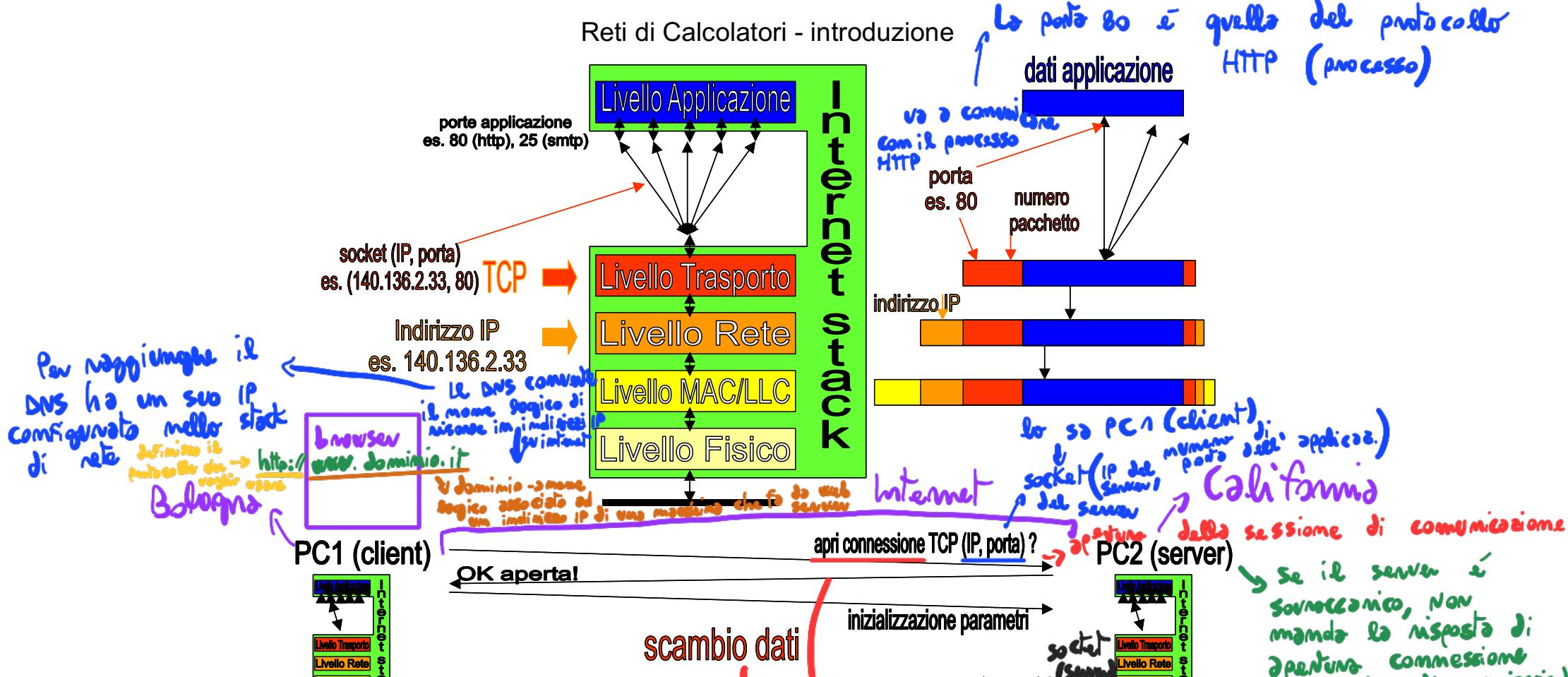
Livello trasporto su Internet: TCP

- TCP è il protocollo di livello trasporto usato di fatto su Internet
- lo standard architetturale "de facto" usato su Internet prevede il **connubio TCP/IP**
- TCP consente lo smistamento dei pacchetti verso le rispettive applicazioni in ascolto su "porte"
→ livello applicazione (protocollo HTTP)
- TCP richiede **attivazione della connessione** punto a punto tra due socket.
 - **Socket**: (indirizzo IP + numero di porta dell'applicazione di livello superiore)
 - In questo modo mette in comunicazione le applicazioni in attesa sui socket.
 - Es. PC1 (client) invia richiesta TCP di connessione sul socket del PC2 (server)
 - Se il socket esiste e non è occupato, TCP di PC2 risponde ok!
 - Se riceve l'ok da PC2, TCP di PC1 può inviare i dati di configurazione
 - Ora avviene lo scambio dati veri e propri a livello TCP
- **Rilascio della connessione** TCP: si liberano le porte usate

• **PORT NUMBER**: Numero intero su 16 bit (2 byte) che si usa per identificare a quale applicazione su internet inviare i dati di livello applicazione (si usa una socket per identificare l'applicazione su internet)

Scritto nelle header/trailer del segmento (a livello applicazione)

⇒ **SOCKET**: coppia INDIRIZZO IP, numero di applicazione destinazione (a livello trasporto)



Il concetto di socket TCP, l'incapsulamento TCP, e le fasi di apertura connessione, scambio dati e chiusura connessione.

La figura mostra i livelli ISO/OSI di Internet tra i quali spicca al livello quarto, il livello trasporto e il suo protocollo TCP. Il livello trasporto mostra al livello applicazione un servizio di spedizione affidabile, mentre il livello applicazione mostra al livello trasporto una porta per ognuna delle applicazioni attive. Il socket viene rappresentato come una linea dal livello trasporto (associato all'indirizzo IP sottostante) a una delle porte delle applicazioni del livello applicazione. A destra dei livelli descritti viene schematizzato l'incapsulamento

Le regole di scambio le decide il protocollo

• **WELL-KNOWN PORT NUMBER** di un protocollo
• **NUMERI DI PORTA** BEN NOTI:

Copyright © Luciano Bononi 2017

NUMERI di PORTA poiché molto usati

com valori da 0 a 1023 (16 bit) che conosciamo tutti

Es. WELL-KNOWN PORT NUMBER DEL PROTOCOLLO HTTP : 80

→ Se chiudo il browser, il client PC1 e il server PC2 hanno un buffer di invio / ricezione e i socket del server e del client sono allontanati

Reti di Calcolatori - introduzione

e quando viene chiuso

la connessione viene liberata la memoria del server ma è il server che dice al client quando

dei dati dell'applicazione. Al livello trasporto, i dati sono encapsulati ponendo in testa al pacchetto il numero di porta e il valore del contatore di pacchetti di TCP. Il pacchetto TCP viene passato al livello inferiore IP che aggiunge l'indirizzo IP del destinatario. Sotto a queste figure viene schematizzato lo scambio di messaggi per l'apertura e la chiusura della connessione TCP. Il client invia la richiesta di apertura del socket, il server risponde con la conferma dell'apertura e il client spedisce i dati di configurazione. In seguito avviene lo scambio dati a livello TCP. Al termine il client invia la richiesta di chiusura della connessione e il server risponde confermando la chiusura.

E' un esempio di sessione TCP

Il protocollo TCP richiede a due dispositivi che intendano comunicare di effettuare preventivamente la configurazione dei parametri del socket TCP, originando in questo modo un canale virtuale di tipo punto a punto tra due socket, ovvero tra due applicazioni di livello superiore alle quali vengono smistati i pacchetti da TCP. Un socket è un punto di arrivo o partenza (virtuale) dei dati a livello trasporto, dal quale è in atto l'invio e la ricezione di pacchetti destinati a un'applicazione, ed equivale a una coppia: (indirizzo IP, numero di porta dell'applicazione). Una volta instaurata la configurazione punto a punto tra due socket, attraverso lo scambio di pacchetti di configurazione, può iniziare lo scambio dei dati a livello trasporto.

La connessione viene instaurata con una richiesta di uno dei due host (il client) nei confronti dell'host server. Deve essere specificato l'indirizzo IP del server e il numero di porta sul quale è in attesa l'applicazione (o il servizio) con la quale si intende dialogare. Il server verifica che il socket esista (verifica il numero di porta) e che non sia già occupato, e in caso affermativo risponde con un pacchetto di conferma. Se il pacchetto di conferma è ricevuto, il client invia un ulteriore pacchetto di conferma contenente i dati di configurazione, dopodiché la comunicazione procede attraverso lo scambio di pacchetti dati tra i livelli TCP, che verranno smistati verso le porte indicate.

Lo scambio dati avviene attraverso pacchetti che saranno ordinati e ritrasmessi in caso di perdita a cura del protocollo TCP.

Al termine del dialogo, la connessione TCP tra due applicazioni può essere rilasciata, liberando la porta occupata.

deallocare il suo buffer, perché se il server non ha ancora ricevuto gli acknowledgement sarebbero impegnate risorse inutilmente !

perché il server, una volta dato lo risposto alla richiesta vuole liberare risorse il prima possibile

• Attacco Dos:
sovraccarico delle richieste di un server
il server occupa risorse su misura

Il client invece vorrebbe tenere ≠ la sessione di comunicazione (scambio dati) aperta il più possibile

Copyright © Luciano Bononi 2017

MULTIPLEXING / DEMULTIPLEXING

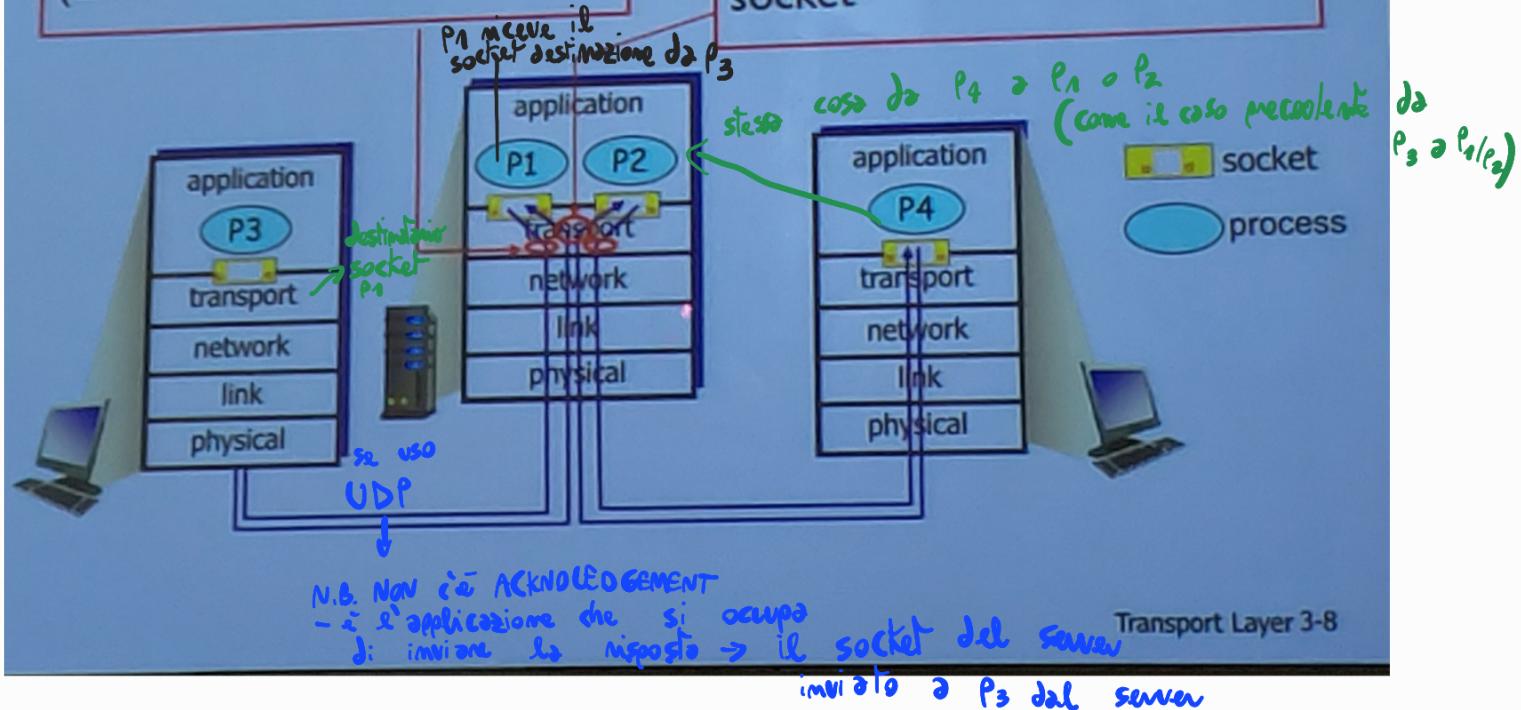
Multiplexing/demultiplexing

multiplexing at sender:

handle data from multiple sockets, add transport header (later used for demultiplexing)

demultiplexing at receiver:

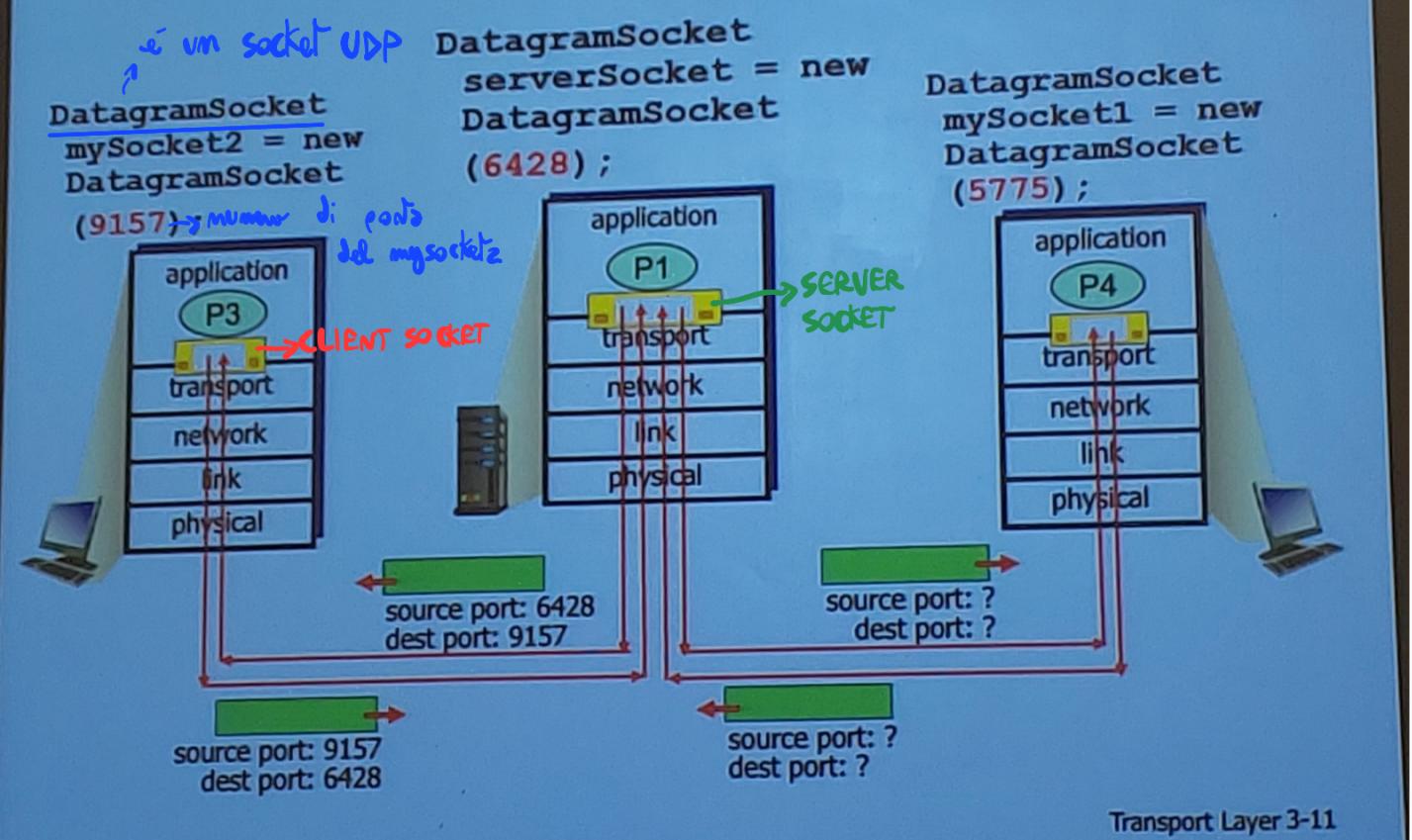
use header info to deliver received segments to correct socket



I PROCESSI SEMPRE ATTIVI: DEMONI (while true)

Se più client diversi vogliamo connettersi allo stesso server usano il **WELCOMING SOCKET**, che ricevuti dal server, il suo (del server) sistema operativo assegna a ciascun client un **CLIENT SOCKET** (numeri di porto con valori alti)

Connectionless demux: example



NUMERI di PORTA:

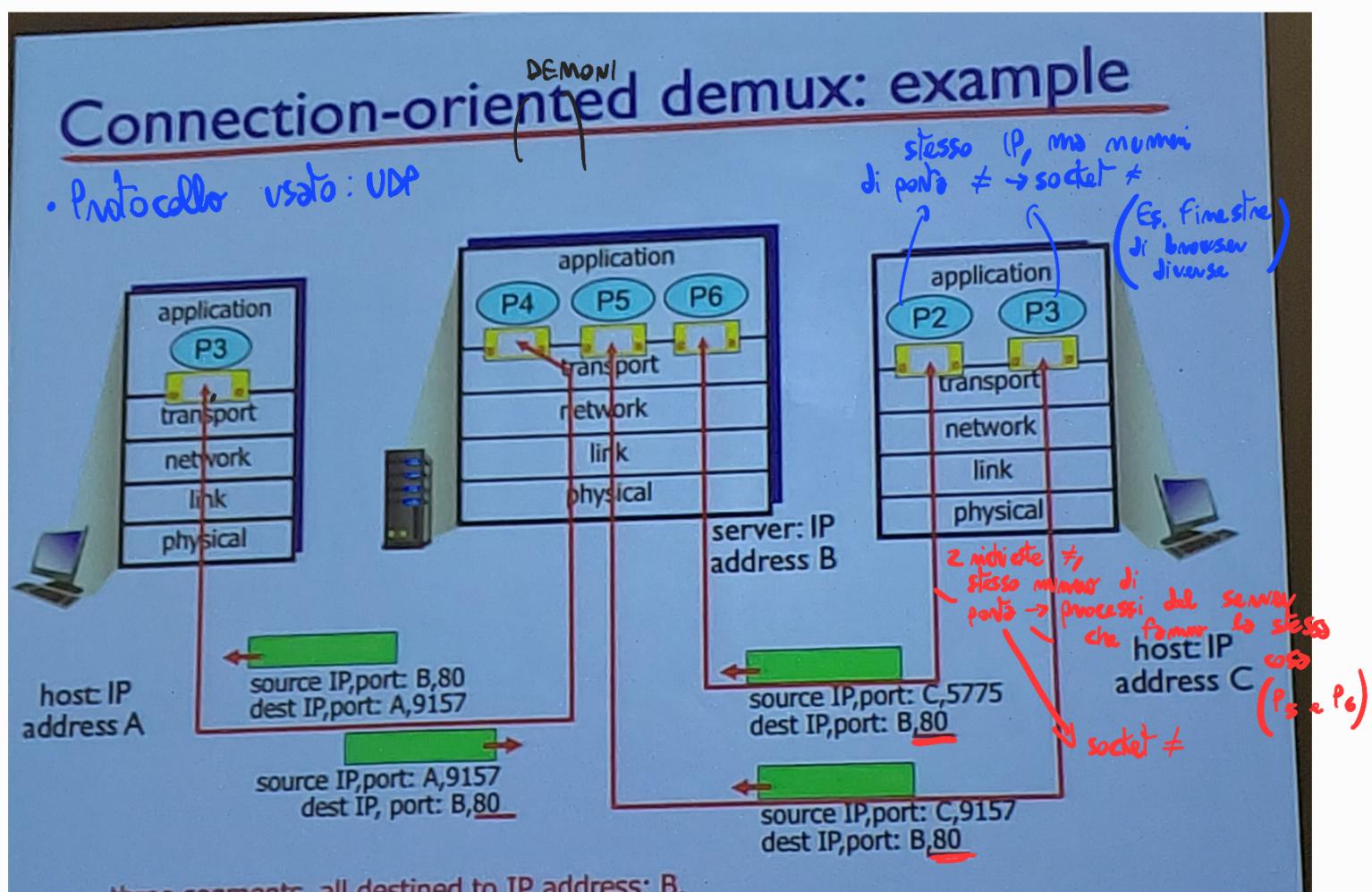
Se uso numeri di porta da 1024 a 16 000 sono dette porte alte, alcuni sono liberi altre sono usate da 30 000 in su, sono spesso porte riservate al sistema operativo

Quelle da 0 a 1024 sono le WELL-KNOWN PORT NUMBER

IN UNA CONNESSIONE DI TIPO TCP: es. numero di porta: 80

In realtà ogni volta che (mando una richiesta al) un server ho un WELCOMING SOCKET, che una volta arrivata al server verrà sostituito con un numero di porta più alto → CLIENT SOCKET relativo al client che ha fatto la richiesta (sul server)

Se ho tante richieste da gestire, il server clona un processo (P_4, P_5, P_6) per ogni richiesta dei client, così che ogni client socket manda un segmento al socket di uno dei processi clonati



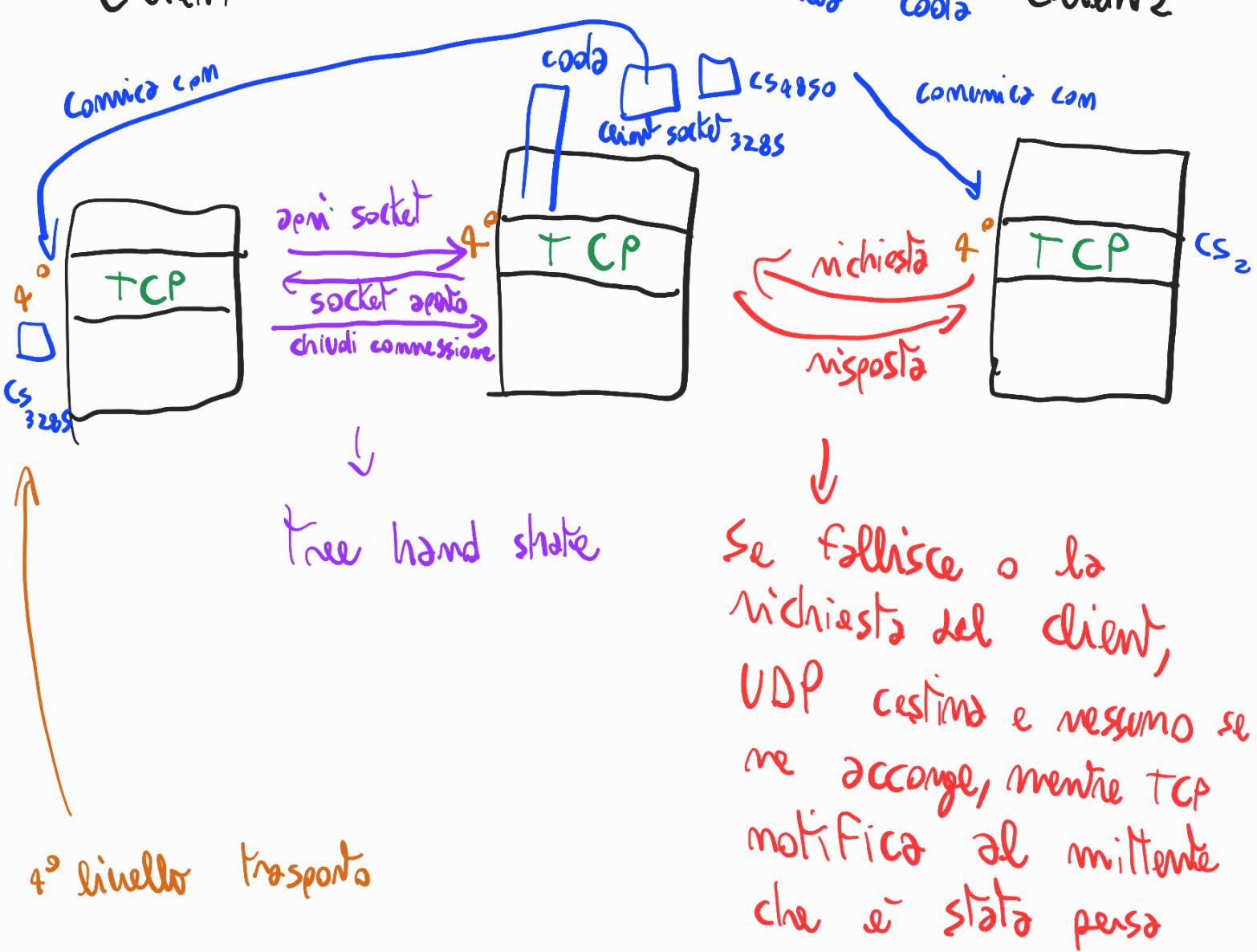
Il socket mittente (client) invia i dati al socket del destinatario (server) → Il socket del server gestisce una richiesta per volta

MODALITÀ DI COMUNICAZIONE: UDP e TCP

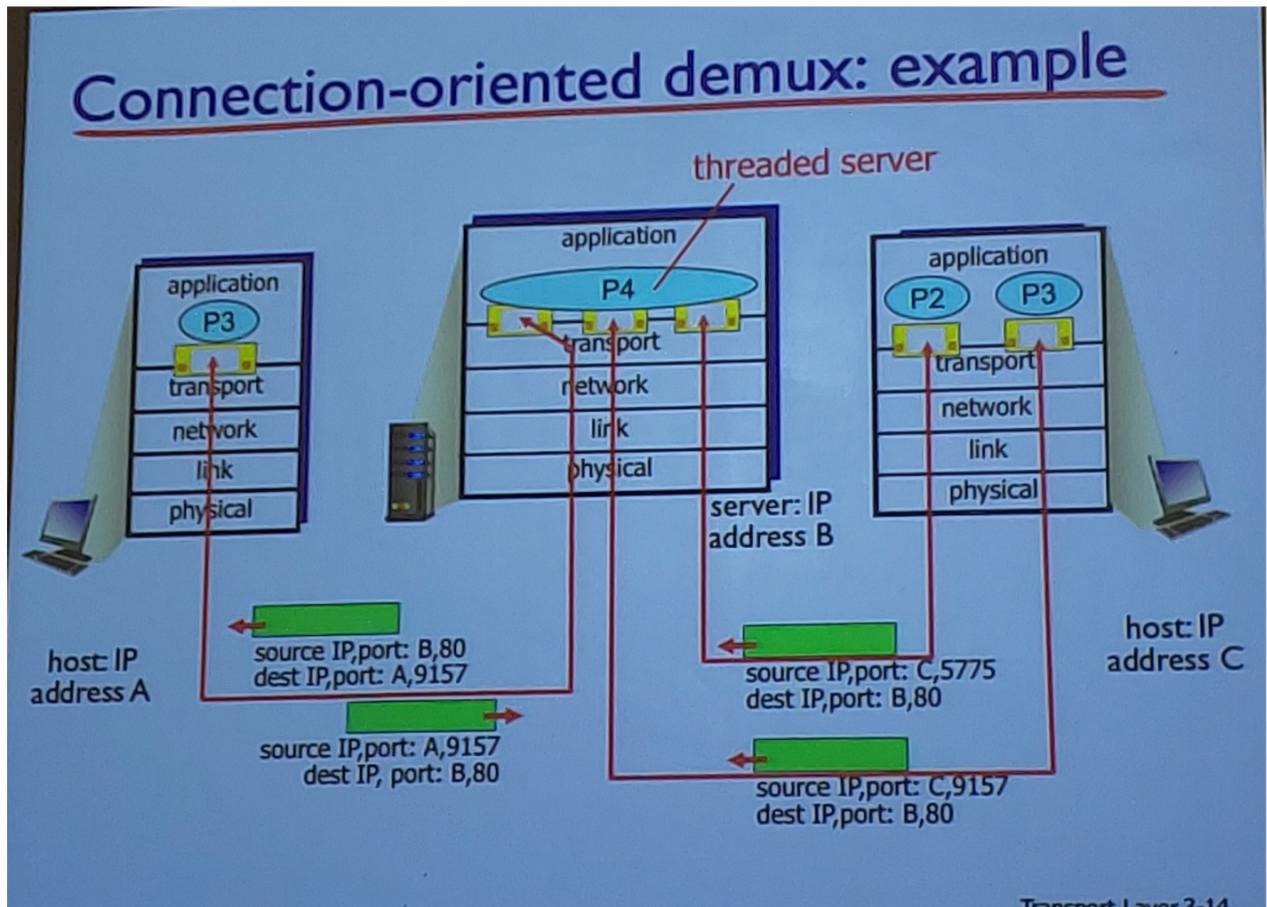
- SOCKET UDP in py: implementato con una coda FIFO per le richieste dei client in attesa che si liberino il socket del server
- Se si riempie la coda, la richiesta viene **accusata** (l'utente non se ne accorge)

• SOCKET TCP in py: al livello trasporto
 TCP ha una coda di attesa sul welcoming socket
 devo avere la gerarchia dell'apertura e chiusura della connessione TCP

- 1) Client richiede di creare una connessione TCP al WELCOMING socket
- 2) Il server crea un socket dedicato per quel client (client socket con numero di porta alta)
- 3) Se il server è multithread, gestisce il numero massimo messi in attesa sullo stesso socket, gli altri processi gli altri client



Connection-oriented demux: example



- PROCESSO MULTITHREAD: È un processo UNICO, che gestisce ogni richiesta in un singolo thread (non sono thread diversi)

Se il socket del server non è stato creato, non invia nulla ma:

→ UDP: appena viene creato il socket, non invia niente, la richiesta è stata restituita

→ TCP: la richiesta continua ad essere inviata al server fino a quando socket del server viene creato, allora la richiesta invia

Usare le reti per far comunicare processi:

- IPC - Inter process communication

PROBLEMA:

Se si interrompe la sessione di comunicazione e un server non aveva finito di scambiare dati, e immediatamente viene ristabilita la connessione con lo stesso client precedente (che si è ricomesso con un welcoming socket)

avrà un numero di porta diverso assegnato e deve finire di scambiare i dati mancati con il server.

Come fa il server a sapere che tu sei il client con cui era stata interrotta la connessione ora che il numero di porta è diverso?

=> A livello sessione c'è presente un ID che identifica una determinata sessione

Controllo di flusso e congestione di rete

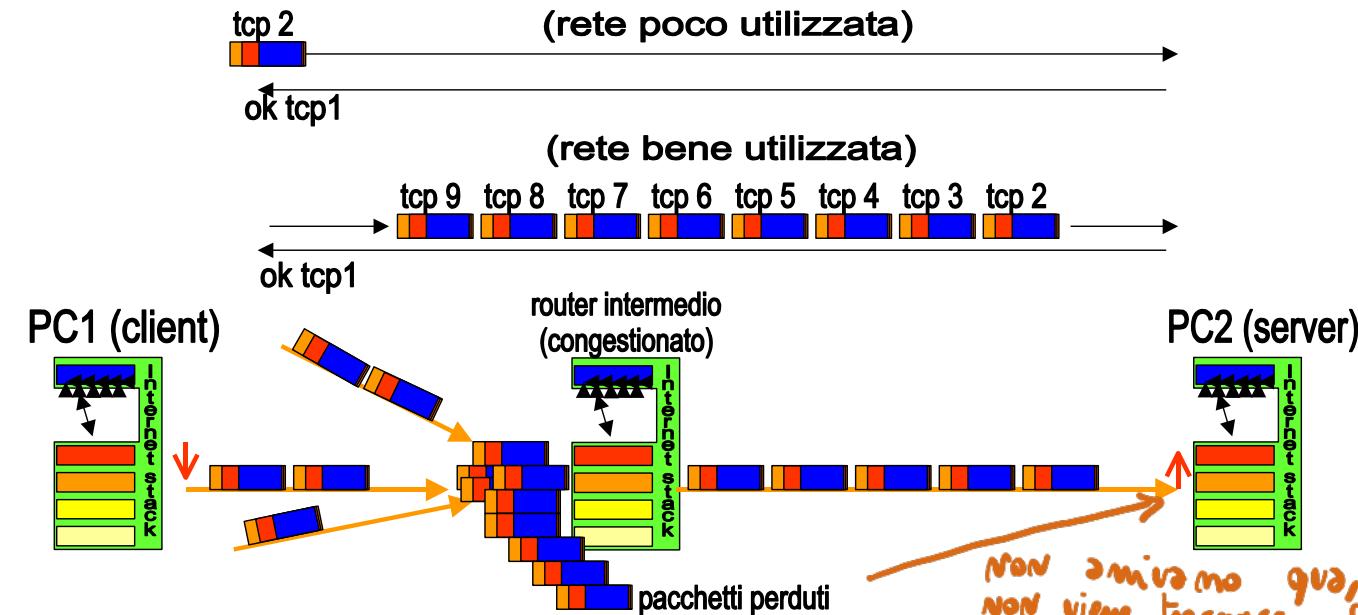
TCP funziona tra due dispositivi di una rete, anche molto distanti tra loro, ed implementa tecniche di controllo di flusso e controllo della congestione di rete.

- Problema: la latenza di rete (tempo di andata e ritorno dei pacchetti) è alta
 - Invio pacchetto e attendo conferma prima di inviare il successivo?
 - Solo un pacchetto in tutto il cammino: troppo lento!
 - Invio tutti i pacchetti uno di seguito all'altro in un colpo solo?
 - Possibile congestione nei router intermedi, che perdono pacchetti: troppo veloce!
 - Possibile saturazione del destinatario, che riceve pacchetti troppo velocemente
 - Scopo del **controllo di flusso**: inviare pacchetti al massimo ritmo sostenibile dal destinatario finale
→ ECP → explicit control protocol → protocollo che modifica il controllo di congestione
 - Scopo del **controllo di congestione**: inviare pacchetti al massimo ritmo sostenibile dal router più lento della rete dal mittente al destinatario finale
- Esistono vari metodi: meccanismo a **finestra scorrevole (sliding window, SW)**.
c'è un pacchetto con maggiore priorità

Il processo di trasmissione è controllato da un numero intero: sliding window
→ Può gestire i router congestionali

SLIDING WINDOW:

Reti di Calcolatori - introduzione



Il problema del controllo di flusso e della congestione a livello trasporto.

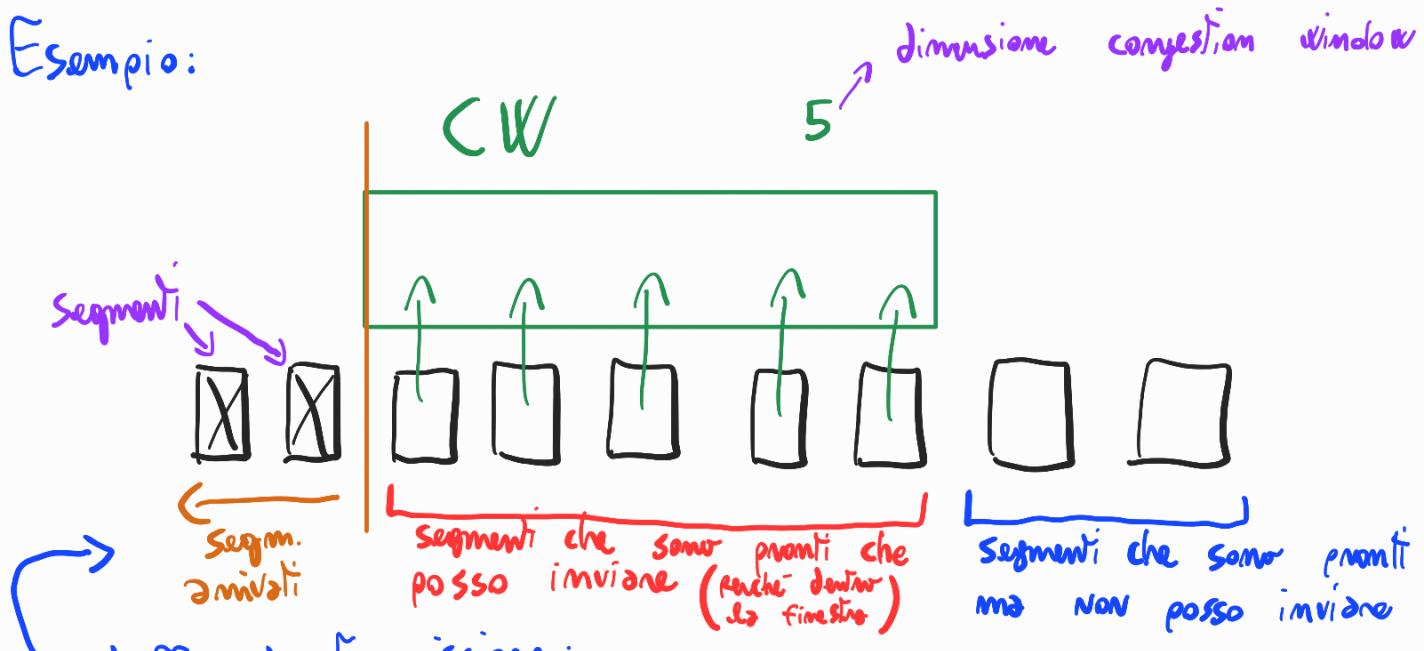
La figura mostra due dispositivi, client e server, che scambiano un flusso di pacchetti a livello trasporto con il protocollo TCP. Tra il client e il server esiste un router impegnato dai pacchetti IP di molte connessioni TCP (da parte di altri client e server). Il router non è in grado di sostenere un ritmo troppo veloce per inoltrare i pacchetti, che si accatastano e possono finire perduti. Questo esempio rappresenta un caso di congestione di un router intermedio.

^f
"congestion
window"

→ Tutti i pacchetti hanno l'esigenza di trasmettere velocemente
Il meccanismo di controllo di congestione è una variabile intez:
Tale numero rappresenta il numero massimo di segmenti che posso trasmettere prima
che i bit nella finestra di congestione ricevono l'ACK

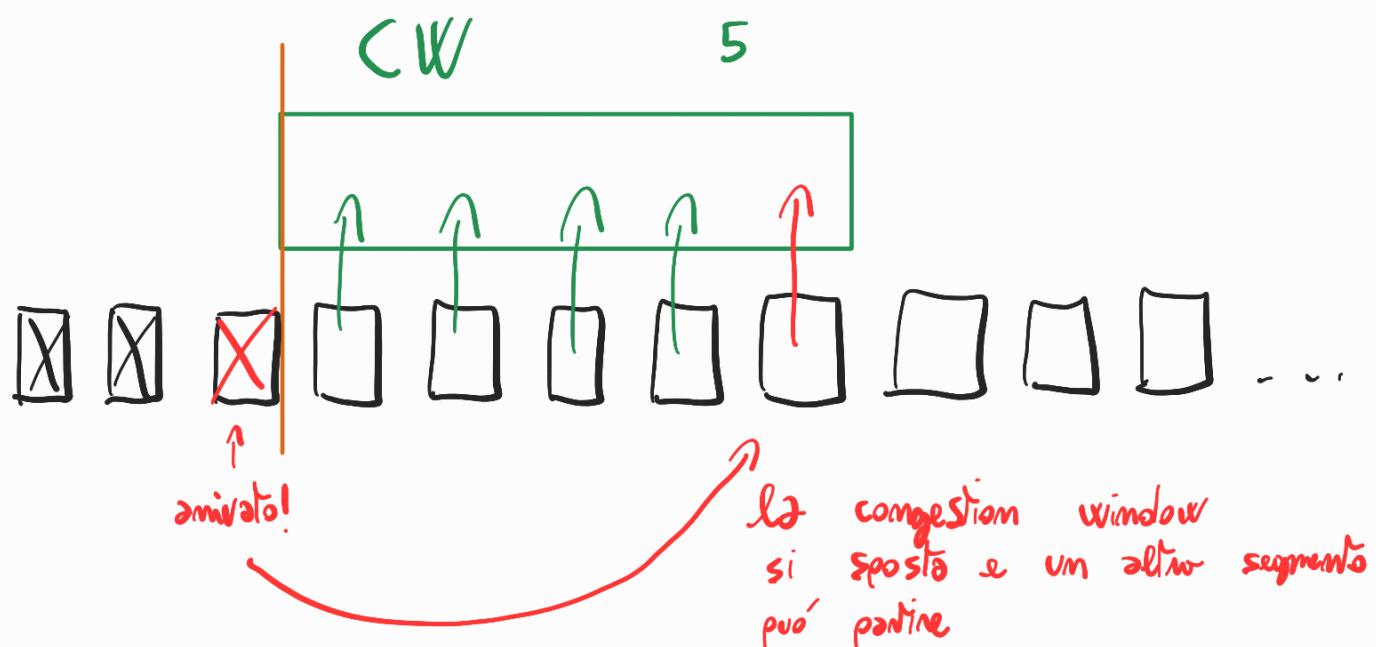
- STOP AND WAIT: trasmetti il pacchetto e aspetti l'ACK (acknowledgment)

Esempio:



buffer di trasmissione:

(della mia applicaz.
o livello trasporto)



Se la rete funziona bene perché ricevo l'ACK la dimensione dello finestra viene raddoppiata e si vede come va.
Si continua a raddoppiare finché ricevo gli ACK. Se comincio a non ricevere gli ACK (o livello trasporto) vuol dire che il pacchetto è stato perduto o rallentato, allora comincia un timeout e se non ritengo l'ACK dei segmenti c'è una congestione → viene ripristinata la dimensione a 1 della CW (congestion window) → Si dice c'è la CHIUSURA (o 1) della congestion window

→ Questi algoritmi che si occupano del controllo di congestione si occupano di fare una "diagnosi" della rete → il protocollo TCP si occupa (anche) la rete

Come si è detto in precedenza, TCP richiede una conferma per ogni pacchetto inviato. La distanza tra due dispositivi che scambiano pacchetti a livello trasporto può essere molto significativa. Il tempo per inviare un pacchetto e ottenere la conferma dell'avvenuta ricezione può quindi diventare dell'ordine dei secondi.

Il problema del controllo di flusso dei pacchetti nel protocollo TCP si basa su due scopi apparentemente in contraddizione tra loro. Il primo scopo è quello di saturare il più possibile la rete di pacchetti, inviandoli a un ritmo elevato. Questo favorisce l'utilizzo delle risorse e le prestazioni della rete (si spediscono e si ricevono tanti bit al secondo). Se si decidesse di inviare un pacchetto e aspettare l'arrivo della conferma, la rete sarebbe usata solo in minima percentuale, e si riuscirebbero a spedire solo pochi bit al secondo. Quindi la rete, pur essendo veloce nell'invio dei bit, verrebbe sfruttata al minimo delle potenzialità. E' quindi evidente quanto sia opportuno spedire i pacchetti a un ritmo il più veloce possibile.

D'altra parte, occorre evitare che un ritmo di invio troppo elevato possa causare il sorgere della congestione nei router intermedi del cammino dei pacchetti, dal mittente TCP (client) al destinatario TCP (server). Se un router si trova a dover inoltrare troppi pacchetti, provenienti da flussi TCP diversi, i pacchetti si accumulano fino ad andare perduti e la rete va in crisi. In tal caso si deve ricorrere a una tecnica di controllo della congestione.

Una forma di congestione può comparire anche sul destinatario finale, nel caso in cui esso non sia in grado di ricevere i pacchetti inviati troppo velocemente. In tal caso si deve ricorrere a una tecnica di controllo di flusso.

TCP usa un meccanismo per il controllo di flusso, detto a finestra scorrevole (sliding window), e un meccanismo per il controllo della congestione, basato sul dimensionamento della finestra scorrevole. Tutto ciò per cercare il massimo ritmo di spedizione che possa garantire l'inoltro dei pacchetti da parte del router più lento del cammino, e prevenire la saturazione del destinatario finale.

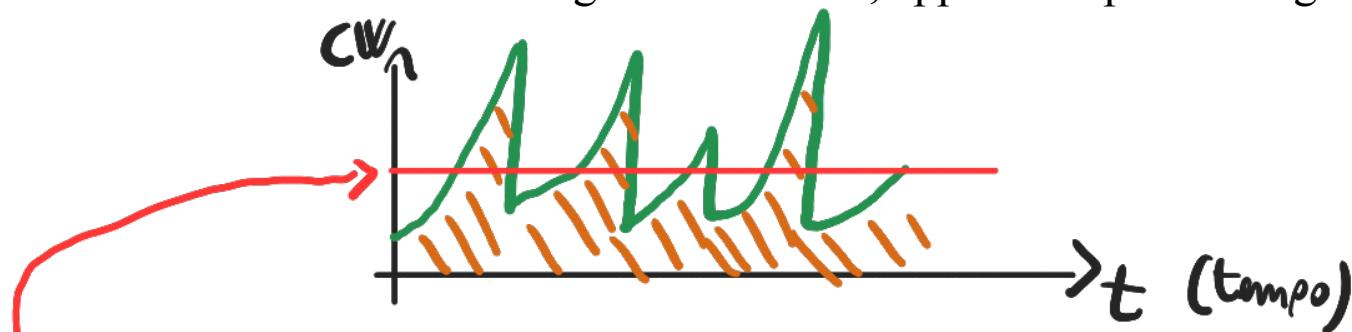
finestra scorrevole di TCP

Si trova anche come CW, Congestion window

medi negativo

Una finestra scorrevole (sliding window, SW) è un valore intero, es. 1 e rappresenta il numero massimo di pacchetti che un mittente può spedire di seguito, in attesa di ricevere la loro conferma.

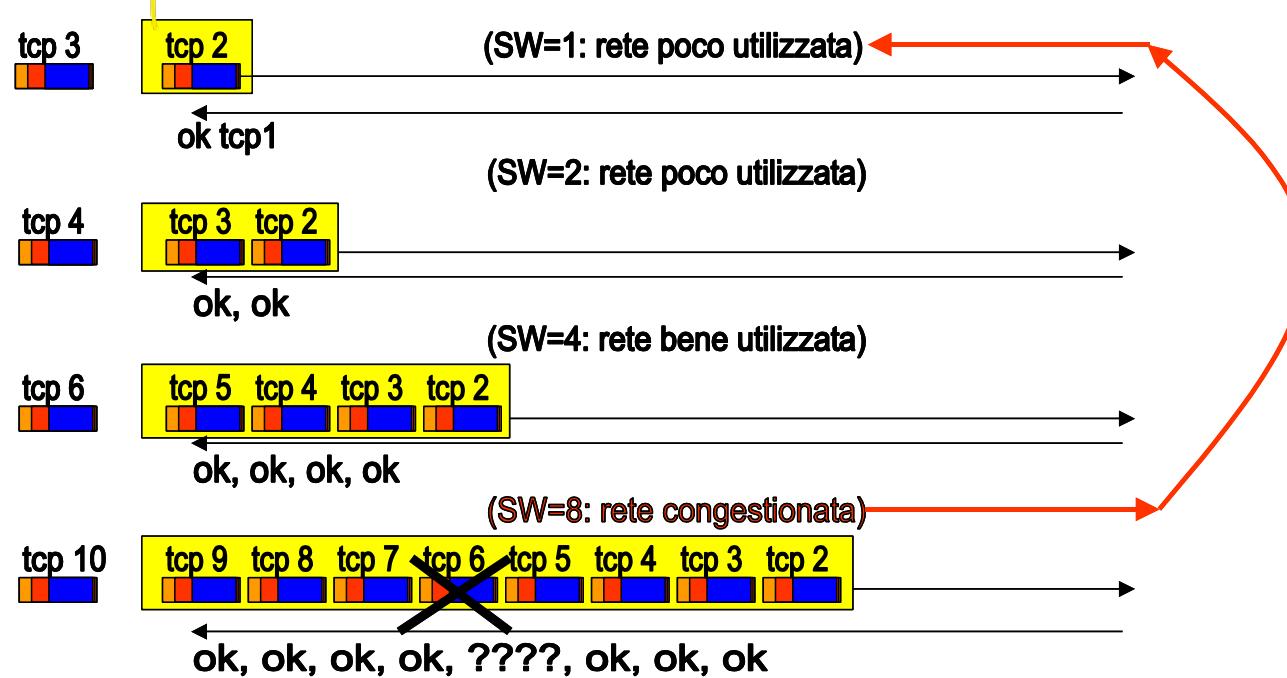
- Ogni mittente TCP spedisce al massimo ritmo possibile un gruppo di SW pacchetti.
- **Controllo di flusso:** non spedisce più di SW pacchetti oltre l'ultimo non confermato
 - Se i primi SW pacchetti vengono confermati, spedisce i successivi SW pacchetti
 - Se un pacchetto non viene confermato lo rispedisce, prima di spedire i successivi SW
- **Controllo della congestione:** spedisce un numero SW variabile di pacchetti
 - Se SW pacchetti spediti sono tutti confermati, aumenta SW
 - Es. spedisce al massimo ritmo prima SW=2, poi 4, poi 8... pacchetti.
 - Appena un pacchetto degli SW inviati non viene confermato (scade il timeout)
 - Assume che ciò sia dovuto a congestione su un router
 - Riparte da SW minimo
 - Cerca di accelerare il ritmo gradualmente e, appena scopre la congestione, rallenta.



valore medio dei dati inviati dall'utente

SLIDING WINDOW

Reti di Calcolatori - introduzione



il meccanismo di controllo di flusso di TCP: finestra scorrevole (Sliding Window, SW)

La figura mostra una sequenza di pacchetti spediti su una rete a livello TCP, da sinistra a destra, usando il meccanismo di controllo di flusso della finestra scorrevole composto con un meccanismo di controllo della congestione basato su finestra di dimensione variabile. Il meccanismo parte con finestra uguale a uno, il che significa che solo un pacchetto può essere spedito prima di ricevere la conferma della ricezione. In questo caso la rete è poco utilizzata. Se la conferma è ricevuta, la finestra viene raddoppiata, spedendo due pacchetti al massimo ritmo di invio. Se entrambi i pacchetti vengono confermati, si passa alla finestra di dimensione quattro, inviando quattro pacchetti al massimo ritmo di invio. Se i pacchetti sono confermati si passa a finestra di otto pacchetti. A questo punto, almeno uno degli otto pacchetti non viene confermato.

Si suppone che questo fatto sia dovuto a un router congestionato e quindi si rallenta il ritmo di invio ripartendo dalla finestra minima (pari a uno). Il massimo grado sostenibile di invio per la rete in esame è stato quindi ottenuto con finestra pari a quattro.

Reti di Calcolatori - introduzione

La finestra scorrevole è un valore intero, che parte da un valore minimo (ad esempio il valore uno). L'idea alla base del controllo di flusso a finestra scorrevole è quello di spedire non più di SW pacchetti consecutivi, a partire dall'ultimo pacchetto non confermato, e quindi attendere la ricezione di una conferma. Un valore di SW uguale a 1 significa che solo un pacchetto può essere spedito, poi occorre aspettare di ricevere la conferma della ricezione. In questo caso la rete è poco utilizzata. Ogni volta che alcuni pacchetti spediti sono confermati, allora è possibile spedire i pacchetti successivi mantenendosi entro il limite massimo di SW pacchetti dall'ultimo pacchetto non ancora confermato. Eventuali pacchetti non confermati sono rispediti fino al ricevimento della conferma. Il senso di questo meccanismo è quello di lasciare in sospeso non più di SW pacchetti, per evitare di saturare il mittente. Questo meccanismo, molto semplificato, realizza il controllo di flusso di TCP.

Se i pacchetti vengono confermati, si può adottare un meccanismo dinamico per accelerare gradualmente il ritmo di invio dei pacchetti, ovvero la dimensione della finestra SW, fino a che non si nota la perdita di almeno un pacchetto tra quelli inviati.

Se i pacchetti vanno perduti, TCP assume anche che la causa di ciò sia la presenza di un router intermedio congestionato, e quindi rallenta il ritmo di invio dei pacchetti per dare modo al router congestionato di smaltire i pacchetti accumulati. Tale meccanismo, sommariamente descritto, è il meccanismo di controllo della congestione di rete di TCP.

Osservando l'esempio, partendo con SW uguale a 1, se la conferma è ricevuta, la finestra viene raddoppiata, spedendo due pacchetti al massimo ritmo di invio. Se entrambi i pacchetti vengono confermati, si passa alla finestra di dimensione quattro, inviando quattro pacchetti al massimo ritmo di invio. Se i pacchetti sono confermati si passa a finestra di otto pacchetti. A questo punto, nell'esempio, almeno uno degli otto pacchetti non viene confermato. Si suppone che questo fatto sia dovuto a un router congestionato e quindi si rallenta il ritmo di invio ripartendo dalla finestra minima (pari a uno). Il massimo grado sostenibile di invio per la rete in esame nell'esempio è stato quindi ottenuto con finestra pari a quattro.

Nomi di Dominio e servizio DNS

- Gli utenti preferiscono usare **nomi per le risorse in rete**, anziché indirizzi IP
- Nomi di dominio per le reti: sono mnemonici e hanno una struttura gerarchica
- Anche i nomi di dominio sono assegnati in modo univoco come i numeri di rete
 - I nomi delle risorse sono arbitrari ma non duplicabili entro il dominio stesso.
- Problema: I protocolli di rete e i router pretendono di usare indirizzi IP
- Soluzione: Il servizio **Domain Name System (DNS)**
 - È basato su una catena di **server DNS** organizzati gerarchicamente
 - Ogni host in rete deve conoscere almeno un **DNS server** → *che si occupa di tradurre indirizzo IP in nomi (logici) di dominio e viceversa*
 - Ogni server DNS conosce almeno un DNS server superiore
 - I **server** ricevono richieste (protocollo DNS) e forniscono indirizzi IP
 - Se un **server** non conosce la risposta inoltra la richiesta DNS a un **server superiore**
 - I **server DNS radice (DNS root server)** conoscono tutti i domini e i loro IP! (ma sono pochi e costosi)

→ Di fatto, il DNS serve a noi utenti per navigare più facilmente su internet, e quest'ultimo funzionerebbe tranquillamente senza DNS

• DNS
E' unico per ogni host

• Nella rete e sottorete che hanno lo stesso nome di dominio



Es. www.cs.unibo.it

Reti di Calcolatori - introduzione

sotto il ROOT LEVEL c'è

TOP LEVEL DOMAIN

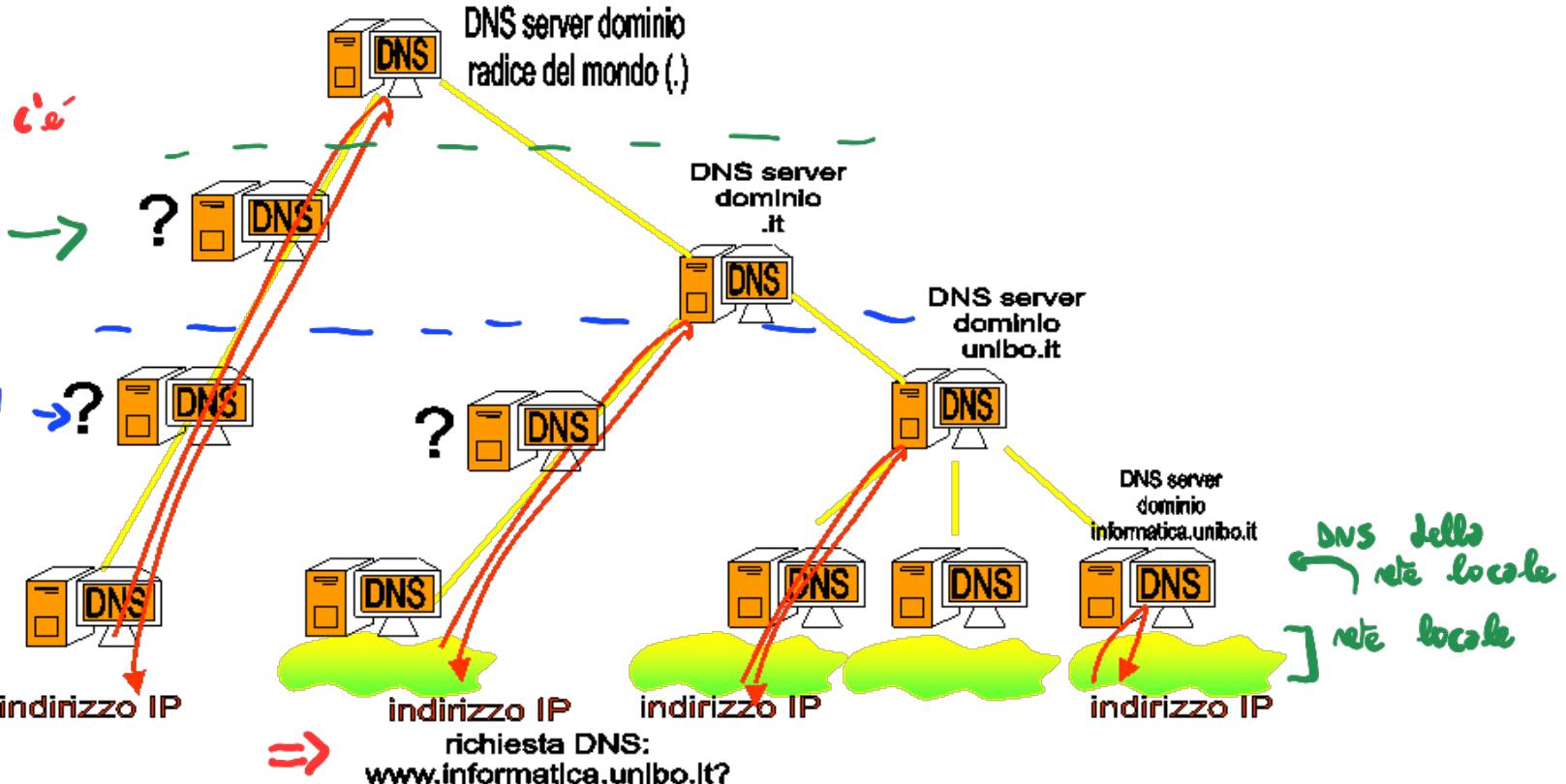
sotto il top level

AUTHORITATIVE DOMAIN → ?

• DNS viaggia su
UDP

Se non arriva
la risposta,
il livello applica
rimanda la
richiesta

(Essendo UDP,
si può perdere
il pacchetto,
ma non ha
affidabilità)



un esempio di richieste DNS per associare indirizzi IP (sconosciuti) a risorse e nomi di dominio.

La figura mostra una possibile gerarchia di server DNS e un esempio di richieste DNS che risalgono la gerarchia dei server fino ad essere soddisfatte. Una richiesta DNS consiste nella richiesta dell'indirizzo IP associato a una risorsa, tipicamente a un host di un dominio. La richiesta della risorsa www.informatica.unibo.it, che equivale all'indirizzo IP del server web del dominio informatica.unibo.it può essere richiesta dal dominio stesso, e in tal caso viene risolta dal DNS locale. Da un altro dominio in Italia viene risolta da un server DNS del dominio .it, mentre dal resto del mondo viene risolta da un server di dominio radice del mondo.

→ Di solito l'indirizzo IP del DNS server coincide con l'indirizzo IP di router

(A volte no, per una questione di sicurezza)

Reti di Calcolatori - introduzione

Gli utenti di Internet preferiscono usare nomi mnemonici per identificare le risorse in rete, ad esempio nomi di host appartenenti a una certa rete, oppure indirizzi di e-mail di utenti di una certa rete. Anche le reti, risultano spesso facilmente identificabili attraverso i nomi di dominio della rete. I nomi di dominio hanno quindi lo stesso senso degli indirizzi IP, e infatti vengono assegnati da enti internazionali, come gli indirizzi IP, per evitare confusione e nomi duplicati. Le risorse appartenenti a un dominio possono avere nomi scelti arbitrariamente (ad esempio nomi di host, indirizzi di e-mail) purchè non siano duplicati all'interno del dominio stesso. Nomi di risorse duplicati sono ammessi in domini diversi, (ad esempio, pippo@topolinia.it e pippo@paperopoli.com). I nomi di dominio hanno una struttura gerarchica del tipo:

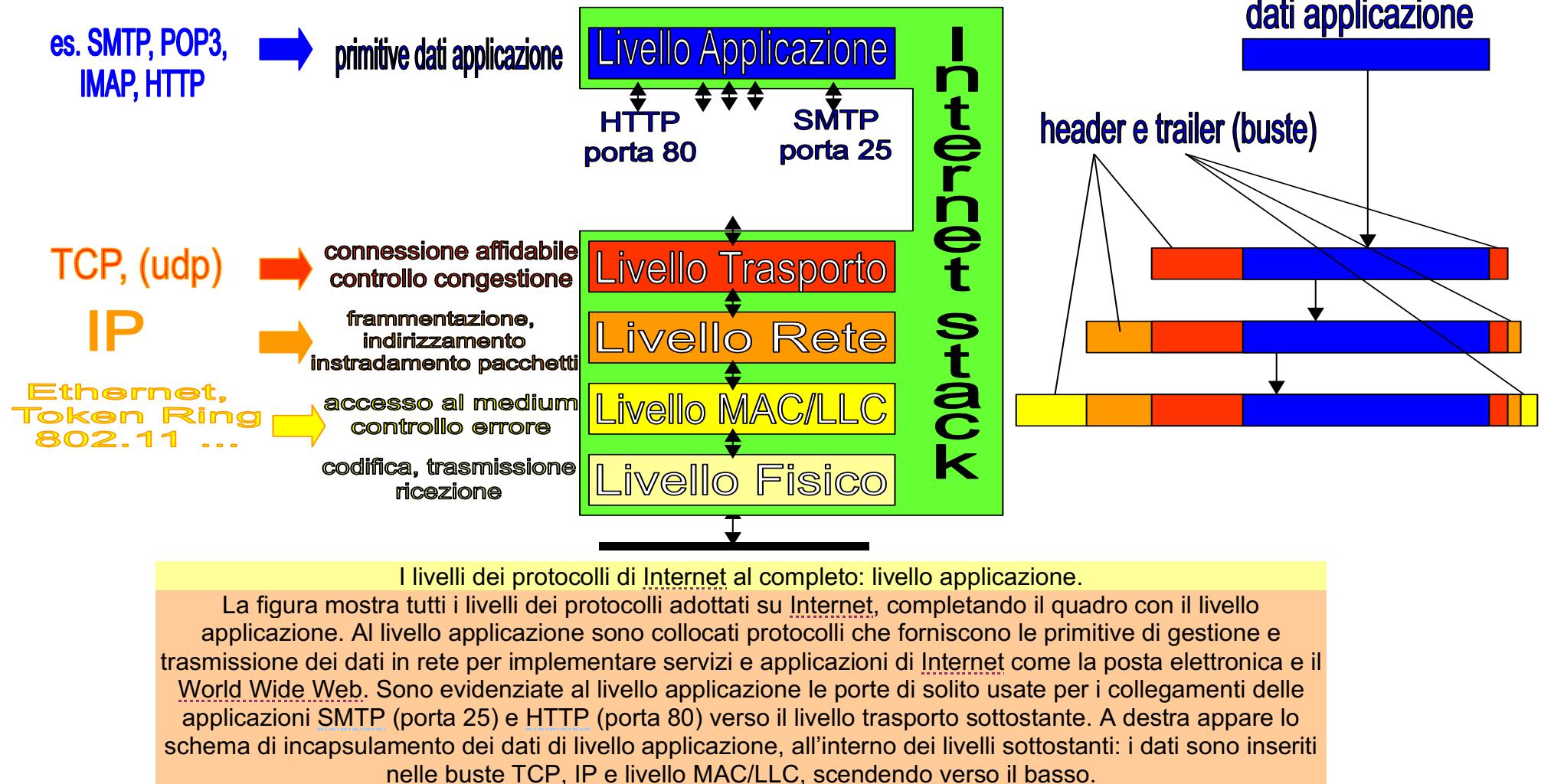
(nomerisorsa.sottodominio.sottodominio.dominioradice). Ad esempio www.informatica.unibo.it è il nome dell'host che agisce da web server per il sottodominio informatica, del sottodominio Università di Bologna, del sottodominio di livello massimo .it (Italia). In realtà il dominio radice del mondo, che esiste implicitamente, non si scrive mai.

Tutto ciò è comodo ma viola le esigenze del livello rete e dei router che pretendono solo indirizzi IP.

Per risolvere il problema, è nato il servizio Domain Name System (DNS) che attraverso una gerarchia di server e un protocollo standard per le richieste permette di risolvere l'associazione tra nome della risorsa e indirizzo IP. Ogni host in rete deve conoscere un server DNS al quale inviare le richieste e ogni server DNS deve conoscere almeno un server DNS di livello superiore. I server di livello superiore conoscono un numero sempre maggiore di nomi e relativi indirizzi IP, ma sono sempre meno per motivi di costo. L'esempio mostra come viene soddisfatta una richiesta DNS a seconda del punto della rete di server DNS dalla quale parte. Se un server DNS non conosce la risposta passa la richiesta al livello superiore, finché qualcuno non conosce l'indirizzo IP.

Livello applicazione

- Il **livello applicazione**: primitive e protocolli per spedire e ricevere dati delle applicazioni
- I livelli Sessione e Presentazione sono raramente implementati sugli host di Internet
- Il livello applicazione si appoggia sul livello trasporto (in particolare sul protocollo TCP)
- Esempi di famose applicazioni di rete e servizi del livello applicazione
 - **Posta elettronica (E-mail)**: basati su protocolli di livello applicazione SMTP, POP3, IMAP
 - Simple Mail Transfer Protocol (SMTP): per la spedizione e trasporto dei messaggi
 - Post Office Protocol 3 (POP3): per la consegna dei messaggi all'utente
 - Internet Mail Access Protocol (IMAP): alternativa a POP3
 - **World Wide Web (WWW)**: basato su applicazione e protocollo HTTP
 - Hyper Text Transfer Protocol (HTTP): protocollo per trasferire pagine web
 - **DNS**: Domain Name Service (protocollo DNS)



Reti di Calcolatori - introduzione

Il livello applicazione dei protocolli di Internet contiene l'implementazione delle funzioni e dei servizi che permettono alle applicazioni di rete in esecuzione sull'host di spedire e ricevere i dati. I protocolli sottostanti di Presentazione e Sessione, previsti dallo Standard ISO/OSI, non sono quasi mai considerati nell'architettura dei protocolli di Internet.

Il livello Applicazione si appoggia direttamente sul livello trasporto e, in particolare, molte applicazioni che richiedono servizi connection-oriented si basano sul protocollo TCP, attraverso numeri di porta che nel tempo sono diventati standard “de facto”.

Ad esempio, la spedizione e il trasferimento dei messaggi di posta elettronica, basati sul protocollo di livello applicazione Simple Mail Transfer Protocol (SMTP) è comunemente associata alla porta di livello applicazione 25. La porta 80 è destinata al protocollo di trasferimento di ipertesti HyperText Transfer Protocol (HTTP) alla base del trasferimento delle pagine di siti del World Wide Web.

Altri esempi di protocolli e servizi che si collocano al livello applicazione sono il protocollo e servizio di Domain Name Service (DNS) e i protocolli IMAP e POP3 per la consegna della posta elettronica.

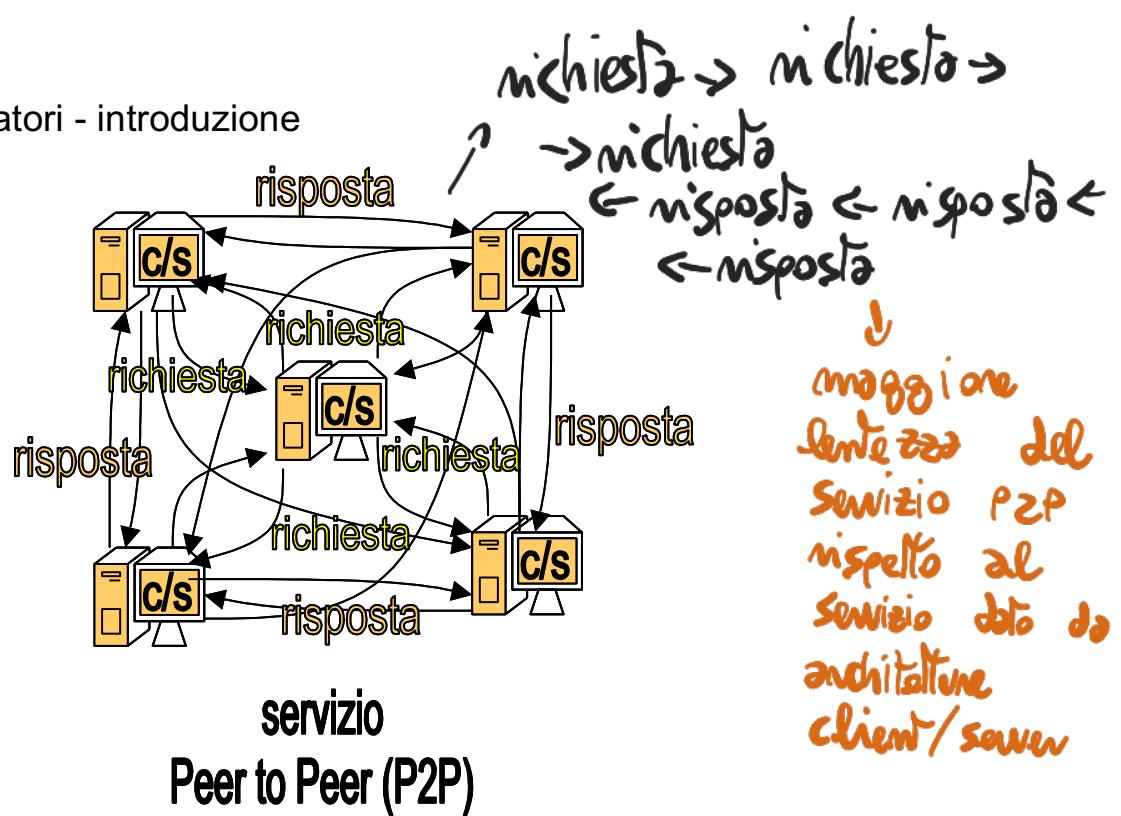
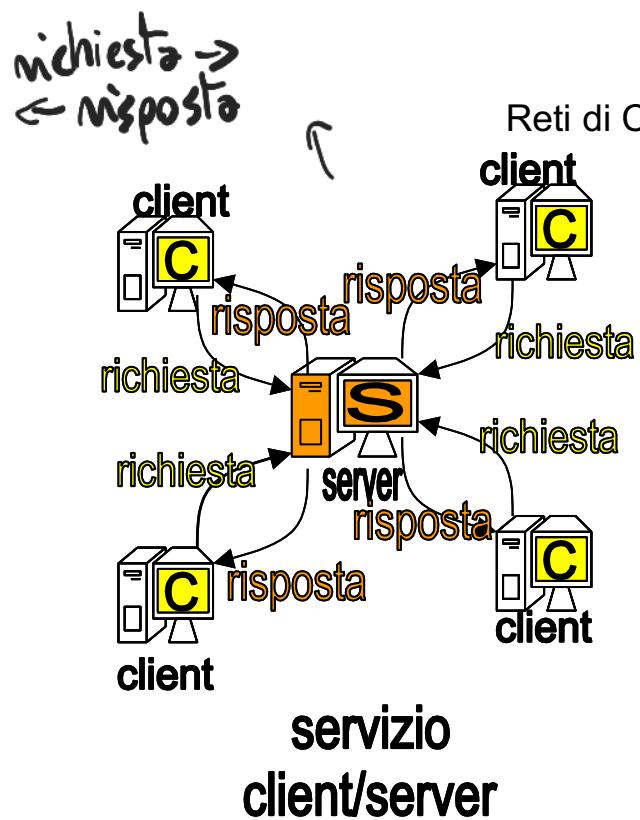
Una dettagliata illustrazione sul mondo dei servizi e protocolli applicativi di Internet sarà oggetto di un modulo apposito.

Servizi Client/Server e Peer to Peer

Le applicazioni e i servizi su Internet possono essere realizzati secondo due modalità architetturali

- Architettura Client/Server → Il client manda richieste al server
 - I Client sono host che spediscono richieste di servizio
 - I Server sono host sui quali sono in esecuzione i servizi che soddisfano le richieste
 - Esempio: servizio DNS, servizio World Wide Web, servizio posta elettronica
- Architettura Peer to Peer (P2P)
 - Tutti gli host sono contemporaneamente sia client che server
 - Ogni host agisce da Server cercando di soddisfare le richieste ricevute, se possibile
 - Ogni host agisce da Client quando spedisce ad altri host le richieste, o per se stesso, o per soddisfare richieste di terzi
 - Esempio: servizi di condivisione dati (file-sharing): Freenet, Gnutella, Kazaa
- Servizi ibridi:
 - Esistono server che aiutano solo a trovare in fretta i giusti host Peer to Peer
 - Esempio: Napster (file-sharing)

→ Inizialmente (Napster) un server diceva quali host avevano certi file e apriva una connessione tra loro per scambiarsi il file → poi si evolve in P2P dove i server controlli spariscono, sono gli stessi client a farlo



Due esempi di applicazione o servizio basati su architettura Client/Server e Peer to Peer (P2P).

La figura a sinistra mostra un'applicazione o un servizio basato sull'architettura Client/Server. Nell'esempio esiste un host server S al quale quattro host Client C inviano le loro richieste. Il server S è l'unico destinatario possibile per le richieste da parte dei client. Ad ogni richiesta, il server S restituisce una risposta del servizio direttamente al client corrispondente. Nel paradigma architretturale Peer to Peer, invece, esistono cinque host che agiscono sia da Client che da Server. Un host può sia generare richieste verso altri host (i suoi peer host), sia fornire risposte, se è in grado di completare il servizio richiesto.

Reti di Calcolatori - introduzione

Le applicazioni e i servizi su Internet possono essere realizzati secondo almeno due modalità architettonali distinte: Architettura Client/Server e architettura Peer to Peer (P2P).

Nell'architettura Client/Server, i Client sono host che spediscono richieste di servizio ai Server. I Server sono i soli host sui quali sono in esecuzione i servizi che permettono di soddisfare le richieste. Un esempio di servizi di tipo Client/Server sono: il servizio DNS, dove ogni host può agire da client spedendo richieste degli indirizzi IP ai DNS server, oppure il servizio World Wide Web, e il servizio di posta elettronica, entrambi basati su client che chiedono pagine web o spediscono e-mail, e server che mantengono le informazioni o memorizzano le e-mail spedite.

Nell'architettura Peer to Peer (P2P), invece, tutti gli host sono contemporaneamente sia client che server. Ogni host agisce da Server cercando di soddisfare, se possibile, le richieste ricevute da altri host. Ogni host agisce da Client quando spedisce ad altri host le sue richieste, o per conto personale, o per cercare di soddisfare richieste di terzi. Un esempio di servizi P2P sono: i servizi di condivisione dati (file-sharing) basati su protocolli Freenet, Gnutella, Kazaa.

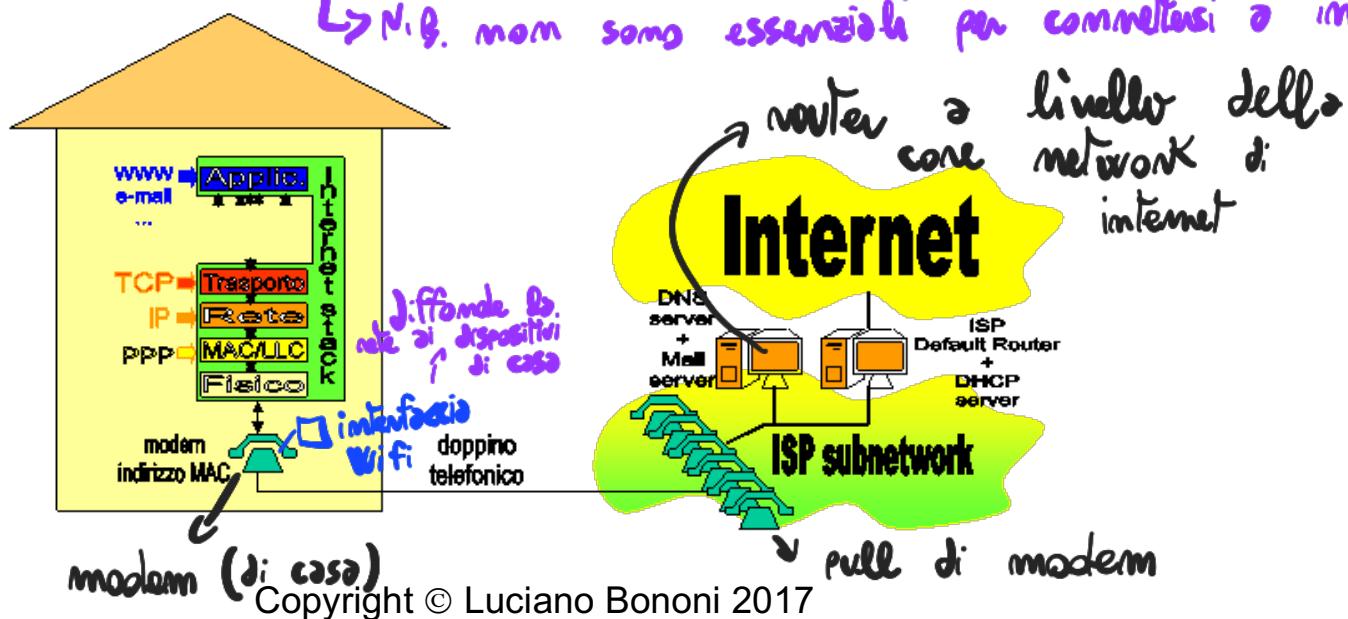
Esistono anche servizi ibridi, nei quali esistono server che aiutano solo a trovare più rapidamente gli host P2P migliori per comunicare e implementare servizi P2P (esempio: file-sharing con Napster).

Configurazione TCP/IP

→ configurazione un sistema TCP/IP - vuol dire connetterlo a INTERNET

- Riassunto: cosa serve per configurare un host per la connessione a Internet?
 - Esempio: computer domestico e Internet Service Provider (ISP) via modem.
- Installare il dispositivo o scheda di rete
 - Creare istanza dello stack TCP/IP e PPP per il dispositivo (modem)
 - Installare firewall per impedire accessi esterni all'host?
- Informazioni da fornire (manualmente, o attraverso DHCP server dell'ISP)
 - Indirizzo IP
 - Default Router e maschera di rete (netmask)
 - Indirizzo IP del Server DNS
 - Indirizzo IP dei Server DHCP e SMTP, POP3, IMAP?

→ protocolli a livello applicazione
↳ N.B. non sono essenziali per connettersi a internet, potrai essere anche senza



Reti di Calcolatori - introduzione

Schema di collegamento del computer domestico a Internet.

La figura mostra un computer domestico, rappresentato dalla pila di protocolli adottati, per il collegamento domestico via modem a Internet. I protocolli utilizzati sono: al livello applicazione tutti i protocolli utili, ad esempio SMTP per la posta elettronica e HTTP il WWW. A livello trasporto e rete viene configurata la coppia TCP/IP. Per il livello LLC/MAC, assumendo di usare un modem, viene adottato il protocollo PPP. Sul lato dell'Internet Service Provider (ISP) si trova: un insieme di modem ai quali collegarsi (telefonando), e quindi una rete locale sulla quale si trovano tutti i server necessari: server DNS, server DHCP, server Mail, e soprattutto il Default Router che inoltra il traffico da e verso il resto di Internet.

Si fornisce ora un breve esempio riassuntivo che illustra quali informazioni e quale configurazione sia necessaria per il collegamento di un host a Internet. Si considera il caso di un host domestico, che viene connesso attraverso il servizio di connessione fornito da un Internet Service Provider (ISP), via modem telefonico.

Un modem è un dispositivo di rete che trasmette i bit a un altro modem attraverso la linea telefonica.

Per il collegamento dell'host occorre quindi: installare il modem e applicare al modem i protocolli PPP per il livello LLC, e TCP/IP per i livelli trasporto e rete. Tali protocolli sono di solito forniti con il sistema operativo in uso.

A questo punto, all'atto della connessione telefonica, occorre configurare (manualmente o meglio automaticamente, attraverso DHCP) le seguenti informazioni: indirizzo IP dell'host (relativo alla sottorete dell'ISP), maschera di rete, Indirizzo IP del default router, e indirizzo IP del DNS server. Possono poi essere inseriti gli indirizzi IP di servizi applicativi come posta elettronica (server SMTP, POP3 o IMAP).

Al rilascio della comunicazione, automaticamente l'host viene cancellato dalla sottorete dell'ISP e l'indirizzo IP eventualmente riciclato per altri dispositivi.

Cenni sulla sicurezza in rete

- Prevenire e contrastare la diffusione di programmi dannosi (computer virus)
 - Possono essere ricevuti via e-mail, via web e se eseguiti causano perdita dei dati
- Prevenire e contrastare l'accesso a sistemi di rete privati connessi a Internet
 - Bloccare l'accesso ai dati e alle applicazioni degli intrusi

↗ emulano il comportamento di un virus
 • si riproducono espontaneamente quando la situazione è favorevole
 → quando si ha un contatto, il virus si trasmette
 (E' internet nella nostra analogia)

- Filtrare i pacchetti a livello rete → viene dei pacchetti → firewall
- Firewall: è il primo router visto dall'esterno della rete, che filtra i pacchetti
- Consentire l'accesso ai dati richiesti da applicazioni e dal personale autorizzato
 - Lista del personale autorizzato (registration) e autenticazione (login e password)
 - Application gateway: è un server che verifica che chi usa certe applicazioni rischiuse sia autorizzato a farlo
- Segretezza (Privacy) dei dati trasmessi
 - Uso di tecniche di crittografia e cifratura dei dati

Reti di Calcolatori - introduzione

○

Introduciamo brevemente alcuni aspetti legati alla sicurezza dei sistemi e dei dati della comunicazione in rete.

Un primo aspetto della sicurezza consiste nel dotarsi degli strumenti per prevenire e contrastare la diffusione di programmi dannosi (computer virus), che attraverso le reti possono diffondersi all'interno di posta elettronica, documenti scaricati, e se eseguiti possono causare la perdita di dati importanti per l'utente e per il funzionamento del sistema stesso.

Un secondo aspetto riguarda la prevenzione e il contrasto dell'accesso indiscriminato ai sistemi di rete privati.

In particolare, occorre bloccare l'accesso ai dati e alle applicazioni di intrusi: per fare questo è possibile filtrare i pacchetti di dati a livello rete, attraverso dei router speciali detti firewall. I firewall sono posti di solito come il primo router che i pacchetti incontrano dall'esterno entrando nella rete privata.

D'altra parte, potrebbe essere opportuno consentire l'accesso dall'esterno della rete ai dati e alle applicazioni di utenti autorizzati. Le tecniche usate in tal senso si basano su liste di persone autorizzate e registrate (registration) e sulla verifica dell'identità basata su autenticazione (login e password privata). Un application gateway è un server che verifica tutte le applicazioni rischiose, consentendone l'uso solo da parte delle persone autorizzate.

L'ultimo aspetto da considerare fa parte in maniera indiretta delle questioni di sicurezza.

Si tratta della segretezza (privacy) dei dati trasmessi in rete (che tutti potrebbero intercettare).

Le soluzioni in uso si basano su tecniche di crittografia e cifratura dei dati, la cui presentazione, tuttavia, esula dai contenuti di questo modulo.

SERVIZI DIFFERENZIATI: insieme di comportamenti dei router che specificano un tempo limite di invio o una certa quantità di bit che sicuramente verrà rispettato

Servizi differenziati e Internet2

Un aspetto critico della comunicazione su Internet è la mancanza di garanzie sui tempi di consegna dei dati (qualità del servizio). → **QoS** → quality of service

- posso imporre che i pacchetti arrivino tutti (servizio connection-oriented di TCP)
- non posso imporre che i pacchetti arrivino entro tempi fissati su Internet
 - manca una progettazione opportuna dei router
 - Problema: i router fanno semplicemente del loro meglio
 - Ma smistano tutti i pacchetti come se avessero tutti la stessa urgenza
- Internet2 e i Servizi Differenziati → *Non esistono delle piattaforme infrastrutture di rete private* → *nuovi router con nuovi protocolli*
 - Una nuova infrastruttura per Internet2: nuove linee dorsali e nuovi router
 - I router spediscono prima i pacchetti urgenti, e poi quelli non urgenti.
 - riservare le risorse lungo il cammino per comunicare i dati
 - Garantisce la qualità del servizio su tutto il collegamento mittente-destinatario

Uno degli aspetti critici della comunicazione su Internet al giorno d'oggi è la mancanza di garanzie sui tempi di consegna dei dati (qualità del servizio). Attraverso il protocollo TCP si possono garantire servizi connection-oriented, senza perdita di dati, malgrado il fatto che i router possano perdere o disordinare i pacchetti. Non è però possibile garantire che i pacchetti arrivino entro un tempo fissato. Questo fatto dipende da una carenza progettuale e architettonica di Internet, alla quale i protocolli possono difficilmente porre rimedio. I router di Internet, infatti, smistano tutti i pacchetti con la stessa urgenza, quindi in situazioni di congestione i vincoli di tempo di consegna possono cadere.

La nuova struttura di Internet2 e dei Servizi Differenziati permette di garantire i requisiti di qualità del servizio di comunicazione che Internet non può supportare. Tutto si basa essenzialmente su nuovi router, che sono in grado di spedire prima i pacchetti urgenti e poi tutti gli altri, se avanza tempo. Ciò va di pari passo allo sviluppo di reti sulle quali sia possibile riservare in anticipo le risorse, lungo tutto il cammino tra mittente e destinatario dei dati.

In questo modo è possibile supportare comunicazione con garanzie di qualità del servizio su scala globale.