

# Reti di calcolatori

VR443470

dicembre 2022

# Indice

<b>1</b>	<b>Introduzione</b>	<b>3</b>
<b>2</b>	<b>ISP, TCP/IP, commutazione dei pacchetti e ritardi</b>	<b>4</b>
2.1	ISP . . . . .	4
2.2	TCP/IP . . . . .	4
2.3	Commutazione dei pacchetti . . . . .	5
2.4	Tipologie di ritardi . . . . .	6
2.5	Sintesi . . . . .	7
<b>3</b>	<b>Tecnica di load balancing, Throughput e collo di bottiglia</b>	<b>9</b>
3.1	Tecnica di load balancing . . . . .	9
3.2	Throughput . . . . .	9
3.3	Collo di bottiglia . . . . .	9
<b>4</b>	<b>Architettura a livelli e incapsulamento</b>	<b>10</b>
4.1	Architettura a livelli . . . . .	10
4.2	Incapsulamento . . . . .	11
<b>5</b>	<b>Indirizzi IP</b>	<b>12</b>
5.1	Indirizzi IP . . . . .	12
5.2	Maschera e blocco CIDR . . . . .	12
5.3	Esercizio di traduzione e numero host . . . . .	13

# 1 Introduzione

**Internet** è una rete di calcolatori che interconnette miliardi di dispositivi di calcolo in tutto il mondo. Gli strumenti in una rete, per esempio cellulari o computer, vengono chiamati **host** (*ospiti*) o **sistemi periferici** (*end system*). Essi sono connessi tra di loro tramite una **rete di collegamenti** (*communication link*) e **commutatori di pacchetti** (*packet switch*). I collegamenti possono essere di vario tipo: cavi coassiali, fili di rame, fibre ottiche e onde elettromagnetiche.

Ogni collegamento detiene una sua **velocità di trasmissione** (*transmission rate*), ovvero la velocità di trasmissione dei dati. L'**unità di misura** è il bit per secondo (bit/secondo, *bps*).

L'insieme delle informazioni, o dati, che vengono inviati o ricevuti prendono il nome di **pacchetto**. L'**obbiettivo** di un commutatore di pacchetti è quello di ricevere un pacchetto che arriva da un collegamento in ingresso e di ritrasmetterlo su un collegamento d'uscita. I due principali commutatori di internet sono: *router* e i commutatori a livello di collegamento (*link-layer switch*). La sequenza di collegamenti e di commutatori di pacchetto attraversata dal singolo pacchetto è nota come **percorso** o **cammino** (*route* o *path*).

Quindi, in sintesi, le definizioni più rilevanti sono:

- ☛ **Internet.** Rete di calcolatori che interconnette i dispositivi di calcolo di tutto il mondo.
- ☛ **Host (o sistemi periferici).** Strumenti in una rete, per esempio computer.
- ☛ **Rete di collegamenti (*communication link*) e commutatori di pacchetto (*packet switch*).** Collega vari *host*, per esempio cavi coassiali o fili di rame.
- ☛ **Velocità di trasmissione (*transmission rate*).** È la velocità di trasmissione dei dati e solitamente la sua **unità di misura** è il bit per secondo, cioè *bps*.
- ☛ **Pacchetto.** Insieme delle informazioni che vengono inviate e ricevute.
- ☛ **Obbiettivo commutatore di pacchetti.** Ricevere un pacchetto proveniente da un collegamento in ingresso e ritrasmetterlo su un collegamento d'uscita. Per esempio i *router*.
- ☛ **Percorso (*route*) o cammino (*path*).** Sequenza di collegamenti e di commutatori di pacchetto attraversata dal singolo pacchetto.

## 2 ISP, TCP/IP, commutazione dei pacchetti e ritardi

### 2.1 ISP

I sistemi periferici accedono ad Internet tramite un servizio chiamato **Internet Service Provider** (ISP). Con **provider** si intende un insieme di commutatori di pacchetto e di collegamenti. Gli **obbiettivi** degli ISP è fornire ai sistemi periferici svariati tipi di accesso alla rete, come quello residenziale a larga banda (e.g. DSL), quello in rete locale ad alta velocità, quello senza fili (*wireless*) e in mobilità.

Esistono 3 **tipi** di livelli di ISP:

**Livello 1.** *Internazionale* (Telecom, TIM, ...);

**Livello 2.** *Nazionale* (Fastweb);

**Livello 3.** *Locale* (solitamente per professionisti).

Più è basso il livello, più gli ISP sono costituiti da *router* ad alta velocità interconnessi tipicamente tramite fibra ottica.

### 2.2 TCP/IP

I sistemi periferici, i commutatori di pacchetto e altre parti di Internet fanno uso di **protocolli** che controllano l'invio e la ricezione di informazioni all'interno della rete. Esistono **due principali protocolli** Internet: ***Transmission Control Protocol*** (TCP) e ***Internet Protocol*** (IP). In particolare, l'IP specifica il formato dei pacchetti scambiati tra router e sistemi periferici. Generalmente ci si riferisce a questi due protocolli tramite il nome collettivo TCP/IP.

## 2.3 Commutazione dei pacchetti

Esistono due diversi approcci per spostare quantità di dati all'interno di una rete: la **commutazione di circuito** e la **commutazione di pacchetto**.

### Commutazione di circuito

Nella **commutazione di circuito** le risorse richieste lungo un percorso (buffer e velocità di trasmissione sui collegamenti) sono **riservate** per l'intera durata della sessione di comunicazione.

#### Vantaggi:

- ✓ **Velocità costante** durante il collegamento poiché le risorse sono riservate e non condivise. Questo si traduce in un **ritardo contenuto**.

#### Svantaggi:

- ✗ **Spreco di risorse** poiché i circuiti sono inattivi durante i periodi di silenzio, ovvero nei periodi in cui non c'è comunicazione;
- ✗ **Complicazioni** nello stabilire circuiti e nel riservare larghezza di banda *end-to-end*.

In questo contesto, i ritardi possono essere causati solamente per tre motivi: (1) a causa dell'instaurazione del circuito, (2) a causa della distanza tra sorgente e destinazione, (3) a causa della trasmissione vera e propria.

### Commutazione di pacchetto

Nella **commutazione di pacchetto** la sorgente divide i messaggi in parti più piccole, ovvero in **pacchetti** assegnando a ciascuno un'intestazione. I pacchetti viaggiano attraverso collegamenti e commutatori di pacchetto dalla sorgente alla destinazione.

#### Vantaggi:

- ✓ **Ottimizzazione** delle risorse poiché c'è una condivisione di esse nei momenti di inattività.

#### Svantaggi:

- ✗ **Possibile perdita** di pacchetti nel caso in cui un buffer di un nodo sia saturo. Questo comporta un buffer overflow e una conseguente perdita;
- ✗ **Ritardo dovuto a *store and forward* e numero di nodi intermedi**. A causa dello *store and forward*, ogni nodo deve attendere di ricevere l'intero pacchetto prima di ritrasmetterlo. Inoltre, con l'aumentare dei nodi intermedi, il ritardo aumenta.  
(approfondimento *store and forward*)

## 2.4 Tipologie di ritardi

Esistono diverse tipologie di ritardo perché quando un pacchetto parte da un *host* (sorgente), passa attraverso una serie di *router* e conclude il viaggio in un altro *host* (destinazione). Questo comporta un ritardo in ciascun nodo (*host* o *router*). I principali ritardi sono: **ritardo di elaborazione**, **ritardo di accodamento**, **ritardo di trasmissione** e **ritardo di propagazione**. L'insieme di questi ritardi è chiamato **ritardo totale di nodo** (*nodal delay*).

### Ritardo di elaborazione

Il tempo richiesto per esaminare l'intestazione del pacchetto e per determinare dove dirigerlo fa parte del **ritardo di elaborazione** (*processing delay*). Per dirigere si intende il tempo che impiega il *router* a determinare la sua parte di uscita.

### Ritardo di accodamento

Una volta in coda, il pacchetto subisce un **ritardo di accodamento** (*queuing delay*) mentre attende la trasmissione sul collegamento. La lunghezza di tale ritardo dipenderà dal numero di pacchetto precedentemente arrivati, accodati e in attesa di trasmissione sullo stesso collegamento. In altre parole, è il tempo speso nel *buffer* prima che il pacchetto venga ritrasmesso.

### Ritardo di trasmissione

Data  $L$  la lunghezza del pacchetto, in bit, e  $R$  *bps* la velocità di trasmissione del collegamento dal *router A* al *router B*, il **ritardo di trasmissione** (*transmission delay*) sarà  $L \div R$ . Questo è il tempo richiesto per trasmettere tutti i bit del pacchetto sul collegamento.

Più semplicemente, dipende dalla velocità di trasmissione e dalla dimensione del pacchetto ed è possibile sintetizzarlo con la formula:

$$t_{\text{trasm}} = \frac{\text{dim\_pacchetto}}{\text{velocità\_trasmissione}}$$

### Ritardo di propagazione

Una volta immesso sul collegamento, un bit deve propagarsi fino al *router B*. Il tempo impiegato è il **ritardo di propagazione** (*propagation delay*). In altre parole è il tempo impiegato per percorrere la distanza verso il *router* successivo.

## Strumenti di misurazione

Esistono diversi **strumenti per misurare il ritardo**:

- **PING**. Dato un indirizzo di destinazione, il calcolatore manda una serie di messaggi e misura il tempo che intercorre tra l'invio e la ricezione della risposta, chiamato anche *Round Trip Time* (RTT).
- **TRACEROUTE**. Misura il *Round Trip Time* tra la sorgente e **tutti** gli apparati di rete intermedi.

## 2.5 Sintesi

- **Internet Service Provider (ISP).** Strumento utilizzato dai sistemi periferici per accedere ad Internet.
- **Provider.** Insieme di commutatori di pacchetto e di collegamenti, solitamente è un'azienda che fornisce servizi.
- **Obbiettivi ISP.** Fornire vari tipi di accesso alla rete ai dispositivi che si collegano (e.g. DSL, *wireless*, ecc.).
- **Tipi di ISP:**
  - **Livello 1.** *Internazionale* (Telecom, TIM, ...);
  - **Livello 2.** *Nazionale* (Fastweb);
  - **Livello 3.** *Locale* (solitamente per professionisti).
- **Definizione TCP/IP.** Protocolli più famosi utilizzati dai sistemi periferici, i commutatori di pacchetto e altre parti di Internet. N.B. il protocollo IP specifica il formato dei pacchetti scambiati tra *router* e sistemi periferici.
- **Definizione commutazione di circuito.** Le risorse sono riservate per l'intera comunicazione.
  - ☞ **Vantaggio commutazione di circuito.** Velocità costante grazie ad un canale dedicato e quindi ritardo contenuto.
  - ☞ **Svantaggio commutazione di circuito.** Spreco di risorse in caso di silenzi durante la comunicazione.
  - ☞ **Causa dei ritardi nella commutazione di circuito.** I motivi possono essere tre:
    - I Instaurazione del circuito;
    - II Distanza tra sorgente e destinazione;
    - III Trasmissione vera e propria della comunicazione.
- **Definizione commutazione di pacchetto.** La sorgente divide i messaggi in parti più piccole chiamate **pacchetti**.
  - ☞ **Vantaggio commutazione di pacchetto.** Ottimizzazione delle risorse poiché c'è una condivisione durante l'inattività.
  - ☞ **Svantaggio commutazione di circuito.** Eventuale perdita di pacchetti nel caso in cui un nodo intermedio abbia il *buffer* saturo (generazione di *buffer overflow*); ritardo causato da *store and forward* poiché ogni pacchetto per essere inoltrato deve essere completamente trasmesso; all'aumentare dei nodi intermedi, il ritardo aumenta.
- **Ritardo di elaborazione (*processing delay*).** Tempo impiegato dal *router* per esaminare l'intestazione del pacchetto e determinare l'uscita.
- **Ritardo di accodamento (*queuing delay*).** Tempo impiegato dal pacchetto all'interno della coda del buffer del *router*.

- **Ritardo di trasmissione (*transmission delay*).** Tempo che dipende dal rapporto tra la dimensione del pacchetto e la velocità di trasmissione.
- **Ritardo di propagazione (*propagation delay*).** Tempo impiegato per percorrere la distanza verso il *router* successivo.
- **Strumenti per la misurazione del ritardo.** I due strumenti sono “PING” e “TRACEROUTE”. La differenza è che PING misura il RTT tra sorgente e destinazione, mentre il TRACEROUTE misura il RTT tra sorgente e ogni nodo intermedio.



## 3 Tecnica di load balancing, Throughput e collo di bottiglia

### 3.1 Tecnica di load balancing

Nel momento in cui il **mittente** (sorgente) calcola il **percorso migliore** per inviare i suoi dati al destinatario, può accadere che **trovi due o più strade identiche**. Con quest'ultimo termine si intende che i percorsi con il costo minimo, e quindi i più efficienti, siano due o più. In questo caso, viene applicata la tecnica di load balancing.

La tecnica di **load balancing** prevede di suddividere il carico dei pacchetti in tutti i percorsi migliori trovati. In questo modo, la comunicazione non avrà un unico percorso sovraccaricato, ma il carico sarà diviso tra più percorsi.

### 3.2 Throughput

Un'altra misura che influisce sulle prestazioni in una rete di calcolatori è il throughput *end-to-end*. Esistono **due tipi di throughput**:

- **Throughput istantaneo**, in ogni istante di tempo  $p$ , è la velocità (misurata in bit per secondo, *bps*) alla quale il destinatario  $B$  sta ricevendo il file.
- **Throughput medio** è dato da una formula specifica. Se l'oggetto da inviare è formato da  $F$  bit e il trasferimento richiede  $T$  secondi affinché il destinatario  $B$  riceva tutti gli  $F$  bit, allora il throughput medio del trasferimento dell'oggetto da inviare è di

$$\frac{F}{T} \text{ bit per secondo}$$

### 3.3 Collo di bottiglia

Quindi, la connessione *end-to-end* presenta criticità nel momento in cui più dispositivi dividono la strada tra sorgente e destinazione. Si parla, infatti, di **collo di bottiglia** (*bottleneck link*), nel momento in cui la velocità di trasferimento viene diminuita a causa di un canale più piccolo o a causa di un dispositivo con banda minore.

## 4 Architettura a livelli e incapsulamento

### 4.1 Architettura a livelli

Un'**architettura a livelli** consente di manipolare una parte specifica e ben definita di un sistema articolato e complesso.

Questa struttura è data dal fatto che fin quando ciascun **livello** (*layer*, o strato) fornisce lo stesso servizio allo strato superiore e utilizza gli stessi servizi dello strato inferiore, la parte rimanente del sistema rimane invariata al variare dell'implementazione a quel livello.

I **servizi** vengono offerti da un determinato livello a quello superiore, ovvero si tratta del **modello di servizio** (*service model*) di un livello. Più in generale, **ogni livello fornisce il suo servizio** effettuando determinate azioni all'interno del livello stesso e utilizzando i servizi del livello immediatamente inferiore.

Nel caso di sistemi grandi e complessi, che vengono costantemente aggiornati, la capacità di cambiare l'implementazione di un servizio senza coinvolgere altre componenti del sistema costituisce un ulteriore importante vantaggio legato alla stratificazione. Quindi, i pro e i contro di questa architettura sono:

- **Vantaggio**
  - Il sistema è strutturato e dunque permette il trattamento dei componenti senza stravolgere l'intera architettura o struttura.
- **Svantaggi**
  - Possibilità di duplicazione delle funzionalità tra due o più livelli, ovvero che un livello cloni le caratteristiche del livello inferiore;
  - Possibilità che la funzionalità presente ad un livello possa richiedere informazioni presenti solo ad un altro livello.

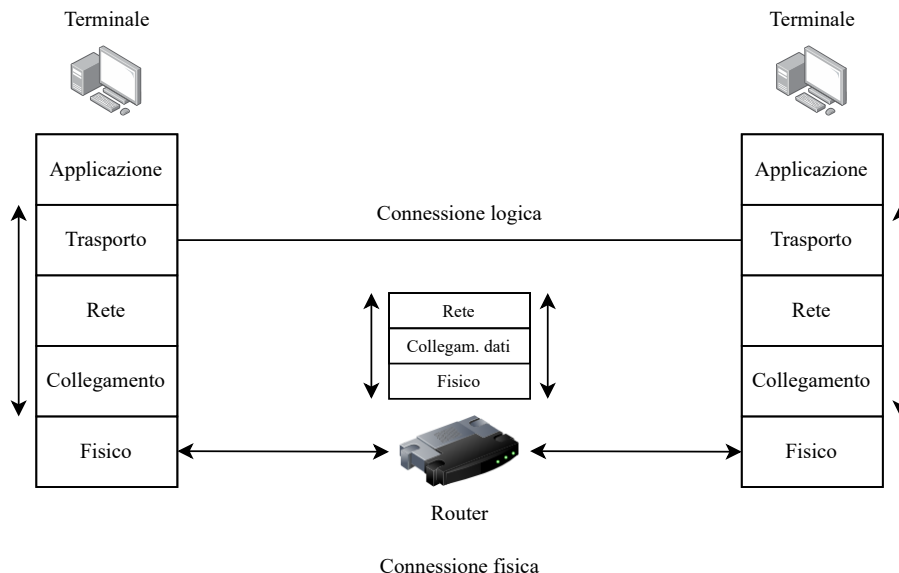
Ogni livello ha un **protocollo** e l'insieme dei protocolli vengono definiti **pila di protocolli** (*protocol stack*). La pila di protocolli di Internet consiste di cinque livelli:

1. Fisico
2. Collegamento
3. Rete
4. Trasporto
5. Applicazione

Un **protocollo** definisce il formato e l'ordine dei messaggi scambiati tra due o più entità in comunicazione, così come le azioni intraprese in fase di trasmissione e/o ricezione di un messaggio o di un altro evento.

## 4.2 Incapsulamento

L'**incapsulamento** (o imbustamento) è un modus operandi applicato nel momento in cui si deve inviare un messaggio.



La comunicazione avviene nel seguente modo:

1. Parte nel **livello di applicazione** del host mittente il quale crea un **messaggio a livello di applicazione** (*application-layer message*) concatenando informazioni aggiuntive, o meglio le informazioni di intestazione. Alla fine del processo di creazione, il messaggio viene passato al livello inferiore, quello di trasporto;
2. A **livello di trasporto** vengono aggiunte altre informazioni di intestazione. Le intestazioni di applicazione e trasporto formano il **segmento a livello di trasporto** (*transport-layer segment*) che incapsula il messaggio a livello di applicazione. Infine, il livello di trasporto passa il messaggio al livello di rete;
3. A **livello di rete** vengono aggiunte informazioni come gli indirizzi dei sistemi periferici di sorgente e di destinazione. Facendo così viene creato un **datagramma a livello di rete** (*network-layer datagram*). Infine, il messaggio viene passato al livello collegamento (*link*);
4. A **livello di collegamento** le informazioni aggiuntive creano un **frame a livello di collegamento** (*link-layer frame*);
5. A **livello fisico** vengono inviati i dati al router e qui termina l'incapsulamento.

Per cui ad ogni livello, il pacchetto ha due tipi di campi: l'intestazione e **payload** (il carico utile trasportato). Il payload è tipicamente un pacchetto proveniente dal livello superiore.

## 5 Indirizzi IP

### 5.1 Indirizzi IP

Un indirizzo IP consente di rendere **identificativo** e **univoco** un host all'interno della rete. Gli indirizzi IP vengono **rappresentati** con 32 bit e utilizzando una **notazione decimale puntata**. Prendendo in considerazione l'architettura di rete spiegata nel capitolo precedente, l'IP si posiziona al livello di rete, nel quale viene aggiunto al messaggio da inviare.

La **notazione decimale puntata** è una rappresentazione degli indirizzi IP che facilita la lettura. Il modus operandi per ottenere tale notazione è il seguente:

1. Dividere i bit in 4 gruppi, ovvero 8 bit per ciascun gruppo;
2. Traduzione di ogni gruppo da binario a decimale;
3. Divisione di ogni gruppo da un punto.

Negli indirizzi IP è importante dividere il prefisso dal suffisso poiché ogni parte ha un significato diverso:

- **Prefisso**, identifica una rete all'interno di Internet;
- **Suffisso**, identifica un host all'interno della rete.

Non esiste un numero specifico di indirizzi IP per il prefisso e per il suffisso. Questo perché dipendono entrambi dalla grandezza della rete; più è grande la rete e meno bit ha di prefisso.

Un **esempio**: 157.27.12.63/16, dove 157.27 identifica il prefisso e 12.63 il suffisso.

### 5.2 Maschera e blocco CIDR

Per identificare il numero di bit presenti nel prefisso, il calcolatore utilizza una sequenza di 32 bit in cui i bit del prefisso sono posti tutti a uno e i restanti a zero. Questo metodo si chiama maschera e un esempio di **maschera** 16 (notazione: /16):

11111111111111111100000000000000

Che rappresenta l'indirizzo: 255.255.0.0

Per cui si può affermare che la maschera **identifica** la grandezza della rete. Pensandoci, più è grande la maschera e più piccola è la rete visto che c'è una stretta relazione con il prefisso di un indirizzo IP.

Un **blocco CIDR** (*Classless Inter-Domain Routing*) è un intervallo di indirizzi IP che sono disponibili nella propria rete.

### 5.3 Esercizio di traduzione e numero host

Dato il seguente indirizzo in notazione binaria:

11100111 11011011 10001011 01101111

Si rappresenta in notazione decimale puntata.

Per **prima cosa** si esegue la traduzione di ogni gruppo da binario a decimale:

- $11100111 \rightarrow 2^7 \cdot 1 + 2^6 \cdot 1 + 2^5 \cdot 1 + 2^4 \cdot 0 + 2^3 \cdot 0 + 2^2 \cdot 1 + 2^1 \cdot 1 + 2^0 \cdot 1 = 231$
- $11011011 \rightarrow 2^7 \cdot 1 + 2^6 \cdot 1 + 2^5 \cdot 0 + 2^4 \cdot 1 + 2^3 \cdot 1 + 2^2 \cdot 0 + 2^1 \cdot 1 + 2^0 \cdot 1 = 219$
- $10001011 \rightarrow 2^7 \cdot 1 + 2^6 \cdot 0 + 2^5 \cdot 0 + 2^4 \cdot 0 + 2^3 \cdot 1 + 2^2 \cdot 0 + 2^1 \cdot 1 + 2^0 \cdot 1 = 139$
- $01101111 \rightarrow 2^7 \cdot 0 + 2^6 \cdot 1 + 2^5 \cdot 1 + 2^4 \cdot 0 + 2^3 \cdot 1 + 2^2 \cdot 1 + 2^1 \cdot 1 + 2^0 \cdot 1 = 111$

E infine si riscrive la notazione in notazione decimale puntata:

231.219.139.111

Adesso viene eseguita la conversione binaria della notazione decimale puntata del seguente indirizzo:

221.34.255.82

Per **prima cosa** si esegue la traduzione di ogni gruppo. La traduzione non è banale, difatti si prenderà ciascun gruppo, si dividerà per due e nella colonna di destra verranno scritti i riporti. Infine, il numero binario sarà scritto dal numero di riporto del numero più basso, fino al numero più alto:

$221 \div 2$	1
$110 \div 2$	0
$55 \div 2$	1
$27 \div 2$	1
$13 \div 2$	1
$6 \div 2$	0
$3 \div 2$	1
$1 \div 2$	1

E così via. Unica accortezza da **ricordare** che se il numero di bit fosse meno di 8, si aggiungono zeri nella parte più significativa.

Dopo alcuni calcoli l'indirizzo IP in binario è:

11011101 00100010 11111111 01010010

Una rete con un suffisso di /20, quanti host contiene? Dato un indirizzo IP da 32 bit, se il suffisso ha 20 bit, allora:  $2^{32} \div 2^{20} = 2^{12}$ . Una rete con un suffisso di 20 bit, avrà un prefisso di 12 bit, ovvero  $2^{12} \rightarrow 4096$  indirizzi.