

COMPUTER SCIENCE

Reti e TCP/IP

Prof. Lorian Storch
loriano@storchi.org
<https://www.storchi.org/>

STATISTICS

MACHINE LEARNING



COMPUTER SCIENCE

UNITÀ DI MISURA E PRESTAZIONI



STATISTICS

MACHINE LEARNING



Prestazione

La larghezza di banda è la quantità massima di dati (numero di bit) che può essere trasmessa in un canale. Viene **spesso utilizzata come approssimazione della velocità di trasmissione effettiva** (unità di misura, ad esempio **Mbps**).

Throughput (prestazioni): quantità di dati (numero di bit) trasmessi sul canale in un determinato periodo di tempo (unità di misura, ad esempio **Mbps**).

MACHINE LEARNING



Prestazione

- **La latenza o il ritardo è il tempo impiegato da un messaggio per viaggiare da un punto all'altro (l'unità di misura è il tempo, ad esempio ms = millisecondo = $(1/1000)$ s).**
 - I tempi di trasmissione sono dovuti alla **semplice velocità di propagazione del segnale nel mezzo di trasmissione e quindi alla distanza**
 - **Tempi necessari per elaborare, ad esempio, l'intestazione dei pacchetti trasmessi**
- **Il tempo di andata e ritorno (RTT, Round Trip Time) è il tempo impiegato da un messaggio per viaggiare dal punto A al punto B e tornare da B ad A (ping e traceroute).**



Prestazione

```
tracert to www.google.com (216.58.209.36), 30 hops max, 60 byte packets
 1 10.0.0.1 [telecomitalia.it (192.168.1.1)] 0.338 ms 0.203 ms 0.262 ms
 2 * * *
 3 172.18.25.0 (172.18.25.0) 3.892 ms 4.317 ms 172.18.25.4 (172.18.25.4) 3.046 ms
 4 172.18.25.212 (172.18.25.212) 3.731 ms 172.18.25.220 (172.18.25.220) 3.957 ms 172.18.25.212 (172.1
.25.212) 4.022 ms
 5 172.19.184.18 (172.19.184.18) 6.530 ms 172.19.184.22 (172.19.184.22) 7.524 ms 172.19.184.18 (172.1
.184.18) 7.087 ms
 6 172.19.177.72 (172.19.177.72) 10.343 ms 172.19.177.46 (172.19.177.46) 8.037 ms 172.19.177.72 (172.
9.177.72) 9.729 ms
 7 172.19.177.8 (172.19.177.8) 19.077 ms 18.159 ms 172.19.177.2 (172.19.177.2) 16.852 ms
 8 195.22.205.116 (195.22.205.116) 18.488 ms 195.22.192.144 (195.22.192.144) 20.639 ms 195.22.205.98
195.22.205.98) 20.207 ms
 9 72.14.204.72 (72.14.204.72) 22.495 ms 72.14.243.190 (72.14.243.190) 18.924 ms 72.14.219.236 (72.14
219.236) 18.668 ms
10 * 66.249.95.89 (66.249.95.89) 17.472 ms *
11 142.250.211.20 (142.250.211.20) 16.206 ms 142.251.235.174 (142.251.235.174) 14.762 ms 142.251.235.
72 (142.251.235.172) 16.842 ms
12 108.170.232.169 (108.170.232.169) 18.552 ms mil07s12-in-f4.1e100.net (216.58.209.36) 18.694 ms 192
178.104.214 (192.178.104.214) 20.536 ms
```



Prestazione

```
PING 216.58.204.228 (216.58.204.228) 56(84) bytes of data.  
64 bytes from 216.58.204.228: icmp_seq=1 ttl=114 time=16.6 ms  
64 bytes from 216.58.204.228: icmp_seq=2 ttl=114 time=16.8 ms  
64 bytes from 216.58.204.228: icmp_seq=3 ttl=114 time=16.9 ms  
64 bytes from 216.58.204.228: icmp_seq=4 ttl=114 time=16.8 ms  
64 bytes from 216.58.204.228: icmp_seq=5 ttl=114 time=16.9 ms
```

^C

```
--- 216.58.204.228 ping statistics ---
```

```
5 packets transmitted, 5 received, 0% packet loss, time 11ms  
rtt min/avg/max/mdev = 16.630/16.792/16.860/0.085 ms
```

```
root@kali:~# ping 192.168.1.1
```

```
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data.  
64 bytes from 192.168.1.1: icmp_seq=1 ttl=64 time=0.368 ms  
64 bytes from 192.168.1.1: icmp_seq=2 ttl=64 time=0.386 ms  
64 bytes from 192.168.1.1: icmp_seq=3 ttl=64 time=0.302 ms  
64 bytes from 192.168.1.1: icmp_seq=4 ttl=64 time=0.359 ms  
64 bytes from 192.168.1.1: icmp_seq=5 ttl=64 time=0.356 ms
```

IE LEARNING



Prestazione

COMPUTER SCIENCE

Speedtest by Ookla

Server: TIM SpA - Pescara (id = 36728)

ISP: TIM

Latency: 2.85 ms (0.12 ms jitter)

Download: 940.69 Mbps (data used: 682.2 MB)

Upload: 313.48 Mbps (data used: 141.4 MB)

Packet Loss: 0.0%



Prestazione

```
root@raspberrypi:~# iperf -s
```

```
-----  
Server listening on TCP port 5001  
TCP window size: 85.3 KByte (default)  
-----
```

```
[ 4] local 192.168.10.22 port 5001 connected with 192.168.10.218 port 40518  
[ ID] Interval      Transfer    Bandwidth  
[ 4] 0.0-10.1 sec   113 MBytes  94.2 Mbits/sec
```

```
root@buchner: ~
```

```
root@buchner:~# iperf -c rpi.ego.eco
```

```
-----  
Client connecting to rpi.ego.eco, TCP port 5001  
TCP window size: 85.0 KByte (default)  
-----
```

```
[ 3] local 192.168.10.218 port 40518 connected with 192.168.10.22 port 5001  
[ ID] Interval      Transfer    Bandwidth  
[ 3] 0.0-10.0 sec   113 MBytes  94.6 Mbits/sec  
root@buchner:~#
```

MACHINE LEARNING



Unità di misura

- **Velocità di trasmissione dei dati = quantità di informazioni / tempo di trasferimento**
- In generale, questa velocità viene espressa in bit al secondo, ovvero **bit/s** (o **bps**, detto anche **bit rate**). Viene utilizzato anche il **byte al secondo**, **byte/s** (o **Bps**).
- Quindi utilizziamo i prefissi standard **k** (=kilo 10^3), **M** (=mega 10^6), **G** (=giga 10^9), e non le approssimazioni basate sulle potenze di due che vengono utilizzate nell'informatica.
- Convertire da **bps** a **KBps** è semplice, basta dividere per 8. Ad esempio, ADSL **10 Mbps** = $10000/8 = 1250$ **KBps**
- Trasferire un file da **10 MiB** con una linea da **5 Mbps** impiegherà circa **$(10 * 1024 * 1024 * 8) / 5 * 10^6 = 16,8$ s** (ignorando la latenza).



COMPUTER SCIENCE

RETI DI COMPUTER

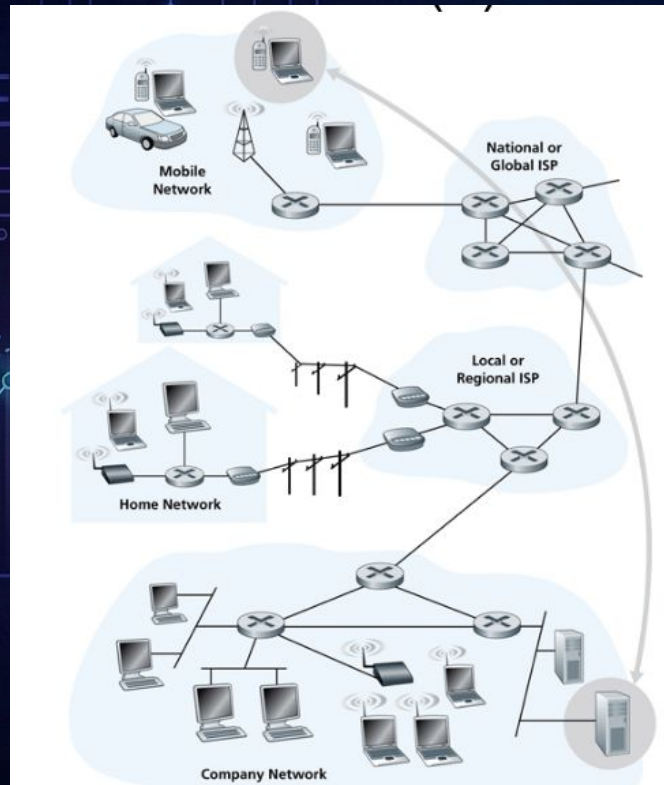
STATISTICS

MACHINE LEARNING



Reti di computer

La rete, costituita da un insieme di computer autonomi interconnessi, è vista come un **sistema che fornisce canali logici** attraverso i quali diverse applicazioni possono comunicare tra loro.



Reti di computer

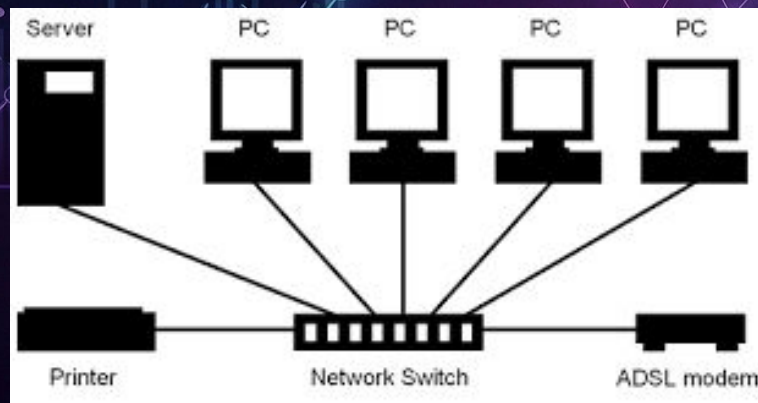
- Un insieme di computer autonomi collegati
 - I nodi (host) possono essere server o PC desktop, oppure dispositivi mobili o altri
 - Le connessioni sono i canali che permettono ai nodi di comunicare. Ovviamente, possono essere trasmesse utilizzando mezzi molto diversi, come il cavo telefonico a doppino intrecciato, la fibra ottica o i canali radio (wireless).
 - Posso avere connessioni dirette o indirette (interruttori).

MACHINE LEARNING



Switch

Uno switch di rete (chiamato anche hub di commutazione o hub di bridging) è un dispositivo di rete che collega tra loro dispositivi informatici utilizzando la **commutazione di pacchetto per ricevere, elaborare e inoltrare i dati al dispositivo di destinazione.**



MACHINE LEARNING



Reti di computer

Posso caratterizzare le reti in base alla loro estensione

- **LAN (Local Area Network)** : rete locale, ovvero una rete che si estende al livello di una singola stanza o al massimo di un edificio.
- **MAN (Metropolitan Area Network)** : può, ad esempio, connettere più LAN.
- **WAN (Wide Area Network)** : Reti che si estendono su aree geografiche. Collegano LAN e MAN (Internet è la WAN per eccellenza).

STATISTICS

MACHINE LEARNING



COMPUTER SCIENCE

INTERNET

STATISTICS

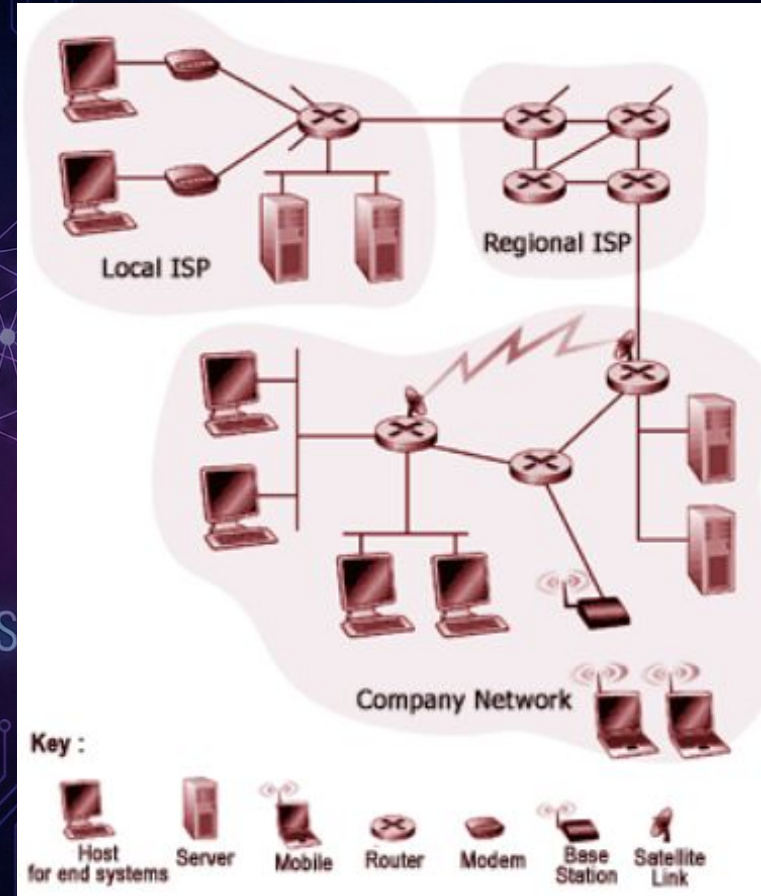
MACHINE LEARNING



Internet

Possiamo descrivere Internet prima di tutto dal punto di vista dei componenti di base .

Internet interconnette milioni di dispositivi in tutto il mondo, inclusi i tradizionali PC desktop, i dispositivi mobili e i server (che memorizzano e trasmettono dati, come pagine HTML, e-mail, ecc.). Questi dispositivi sono chiamati **host o sistemi terminali**.



Internet

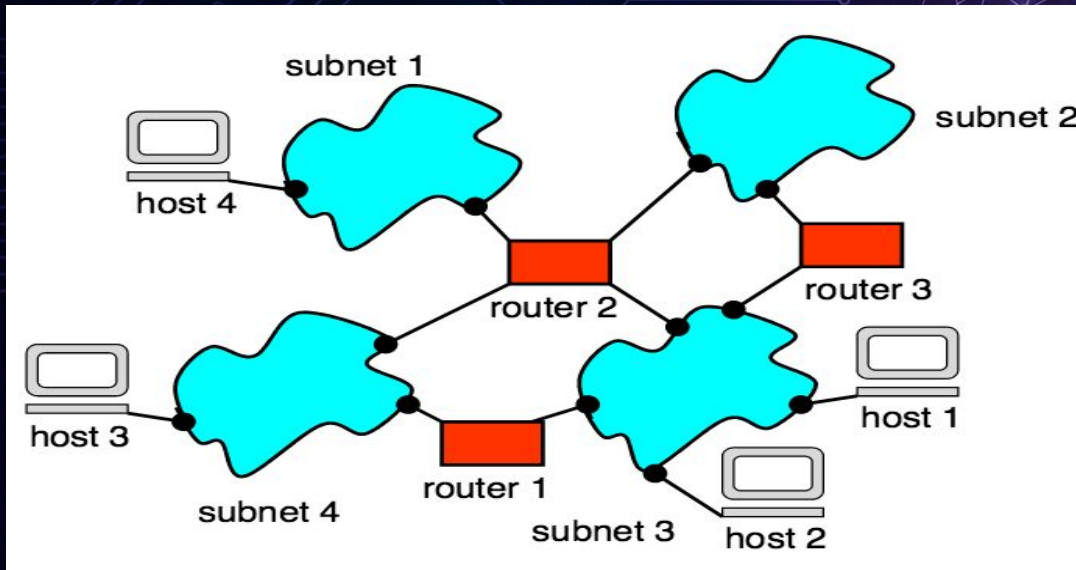
- I vari host sono interconnessi tramite canali di comunicazione. Questi canali possono essere di natura molto diversa, come cavi coassiali, fili di rame, fibre ottiche o collegamenti radio.
- In genere, gli host non sono interconnessi direttamente, ma esistono dispositivi di "commutazione" chiamati router . I router vengono utilizzati per instradare il traffico dati e quindi prendono le informazioni provenienti da un canale e le inviano a un altro canale.
- Internet è un insieme di reti interconnesse. I vari host, così come gli altri dispositivi infrastrutturali, comunicano tra loro utilizzando protocolli comuni (i due protocolli principali sono IP Internet Protocol e TCP Transmission Control Protocol).

MACHINE LEARNING



Internet

- Internet oggi interconnette migliaia di sottoreti



Host : computer connesso a Internet, può essere un client o un server a livello applicativo.

Router : nodo utilizzato per instradare il traffico (un dispositivo gateway di rete di livello 3).

Sottorete : un insieme di host tra i quali esiste una connessione di livello 2 (ad esempio, una LAN).

Ad oggi, circa miliardi di host



Internet: una breve storia

- **Nel 1969 ebbe inizio la progettazione della rete militare ARPANET.** Tra le altre cose, questa rete fu concepita per resistere agli attacchi nucleari. Infatti, era in grado di connettere dispositivi interconnessi, seguendo percorsi alternativi in caso di guasto.
- **Il 1972 ha visto la nascita della posta elettronica (e-mail), del trasferimento di file tramite FTP e dell'accesso remoto.**
- **Vengono introdotti ufficialmente i protocolli IP e TCP.**
- **1979, data di nascita della rete CSNet che interconnette università e centri di ricerca negli Stati Uniti.**



Internet: una breve storia

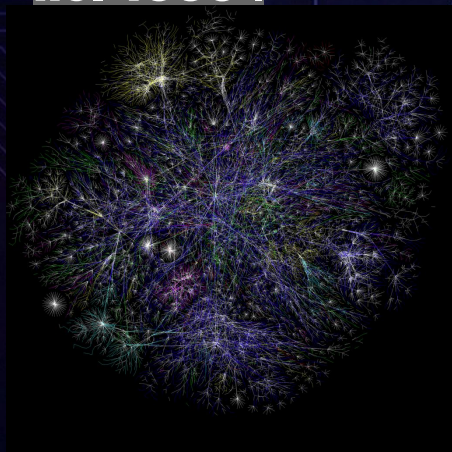
- **Nel 1982, ARPANET e CSNet vengono collegate; questa data è considerata, per certi versi, la data di nascita ufficiale di Internet.**
- **1990: NSFNet, una rete che collega supercomputer, sostituisce Arpanet.** Questo apre la strada agli usi civili e commerciali di Internet.
- **1990** Nello stesso anno **Tim Berners-Lee, che all'epoca lavorava al CERN di Ginevra, inventò l'HTML** (Hyper Text Markup Language), che permette la gestione di informazioni di diversa natura, come testo, immagini ecc. Questo fu il primo passo verso il WWW (World Wide Web).

MACHINE LEARNING



Internet: una breve storia

- **Nel 1993 è stato creato Mosaic, il primo browser.**
- **Nel 1994 nacque Yahoo!, il primo motore di ricerca. Nella seconda metà degli anni '90 ne vennero creati molti altri, culminando nella nascita di Google nel 1998.**



Internet oggi contiene circa **10 miliardi di computer**, ciascuno dei quali (dispositivo) contiene in media un paio di miliardi di transistor. Pertanto, Internet interconnette **10^{19} transistors** : in conclusione, ci sono **10.000 volte più transistor che sinapsi nel cervello umano.**

Oggi Internet viene utilizzato per la vendita di beni e servizi, la trasmissione audio e video, la collaborazione online, il lavoro, i giochi online e l'interazione sociale

MACHINE LEARNING



COMPUTER SCIENCE

INTERNET

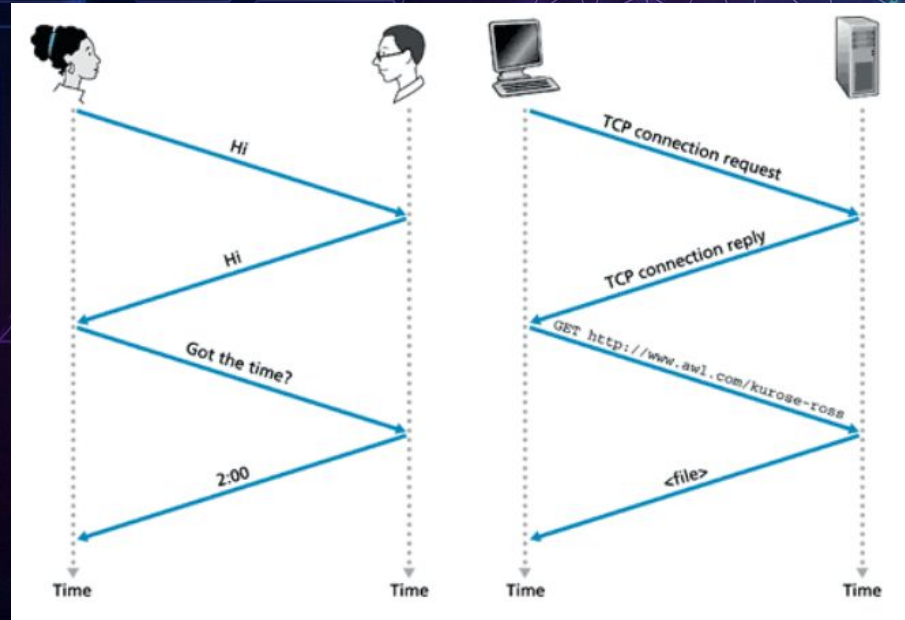
STATISTICS

MACHINE LEARNING



Protocollo di comunicazione

- Un insieme di regole (descritte formalmente) che definiscono i metodi di comunicazione tra due o più entità.

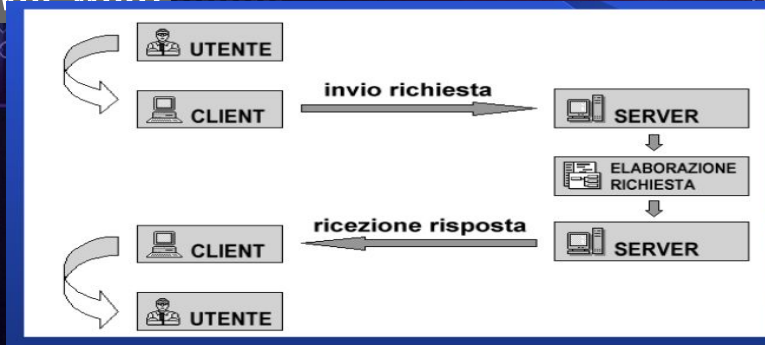


MACHINE LEARNING



Modello di rete: Client-Server

- La maggior parte dei servizi telematici offerti da Internet si basa sulla **modalità di interazione client/server (diversa dal P2P)**.
- Il client è dotato di un software specifico in grado di inviare richieste di servizio a un server specifico. Il client formatta le richieste in un modo appropriato e comprensibile per il server, utilizzando un protocollo specifico (ad esempio, un browser web o un server web).

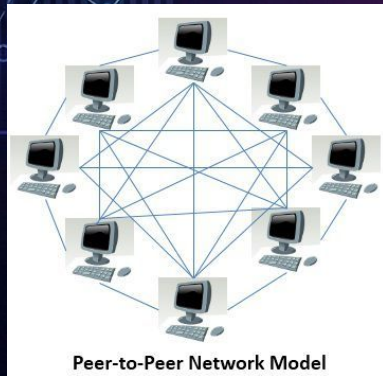


MACHINE LEARNING



Modello di rete: P2P Peer-to-Peer

- In questo caso non vi è alcuna distinzione tra client e server; ciascun nodo può infatti agire sia da client che da server a seconda che stia richiedendo o fornendo dati.
- Per entrare a far parte della rete P2P, dopo l'adesione il nodo deve iniziare a fornire servizi e sarà in grado di richiederli ad altri nodi (ad esempio BitTorrent).



MACHINE LEARNING



COMPUTER SCIENCE

TCP/IP

STATISTICS

MACHINE LEARNING



TCP/IP: Funzionamento di base

- Un protocollo è un insieme di regole che il mittente e il destinatario devono seguire.
- Un esempio banale di un "protocollo" (non formale) che due persone seguono quando si incontrano e una chiede l'ora:
 - Bob saluta Alice
 - Alice saluta Bob
 - Bob chiede l'ora
 - Alice dice l'ora
 - Bob ringrazia e saluta
 - Alice ti saluta

STATISTICS

MACHINE LEARNING



TCP/IP: Funzionamento di base

COMPUTER SCIENCE

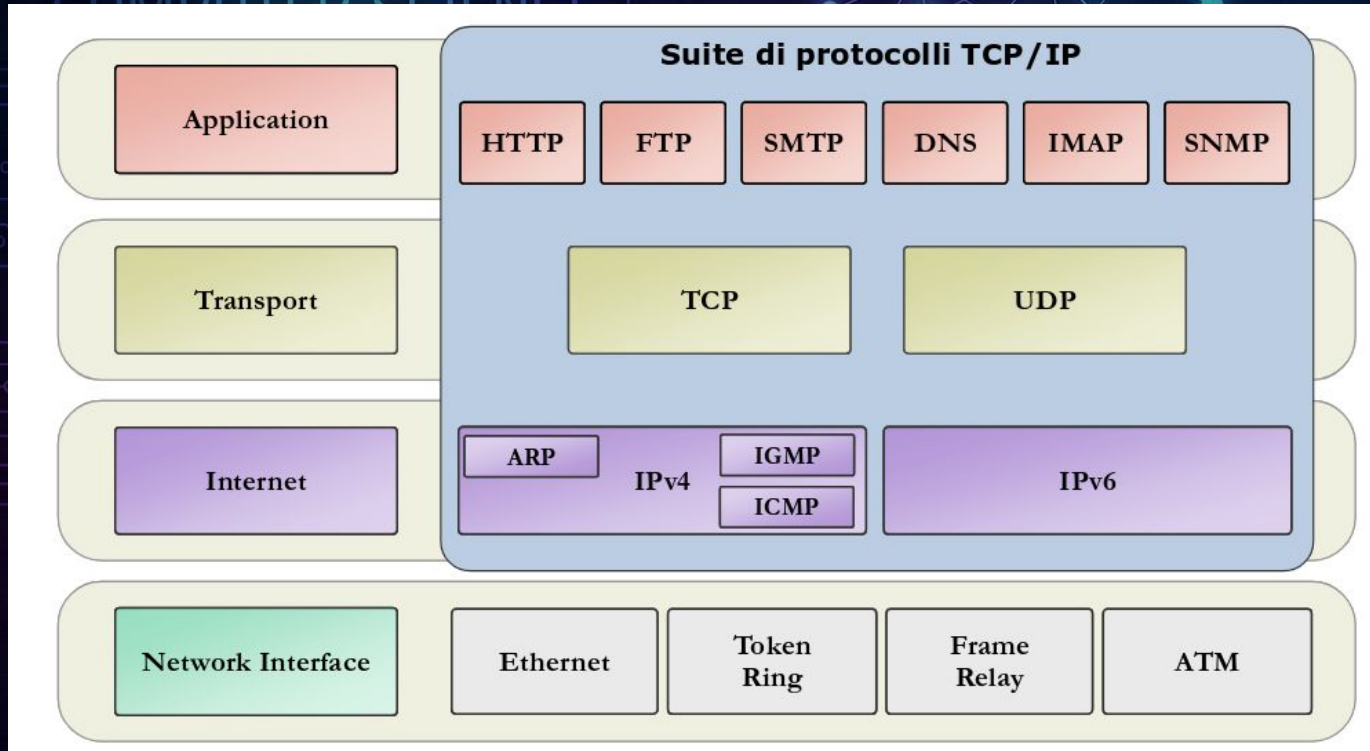


E infine una ricevuta (ACK)

MACHINE LEARNING



Architettura TCP/IP



01010101010101010
0110101001010101
01010111010110101
0111111001011100
0110101011010101
0101010101010101
0110101000100010
0101011101010101
01010111010100101
0111111001010101
0110101011011101

EARNING



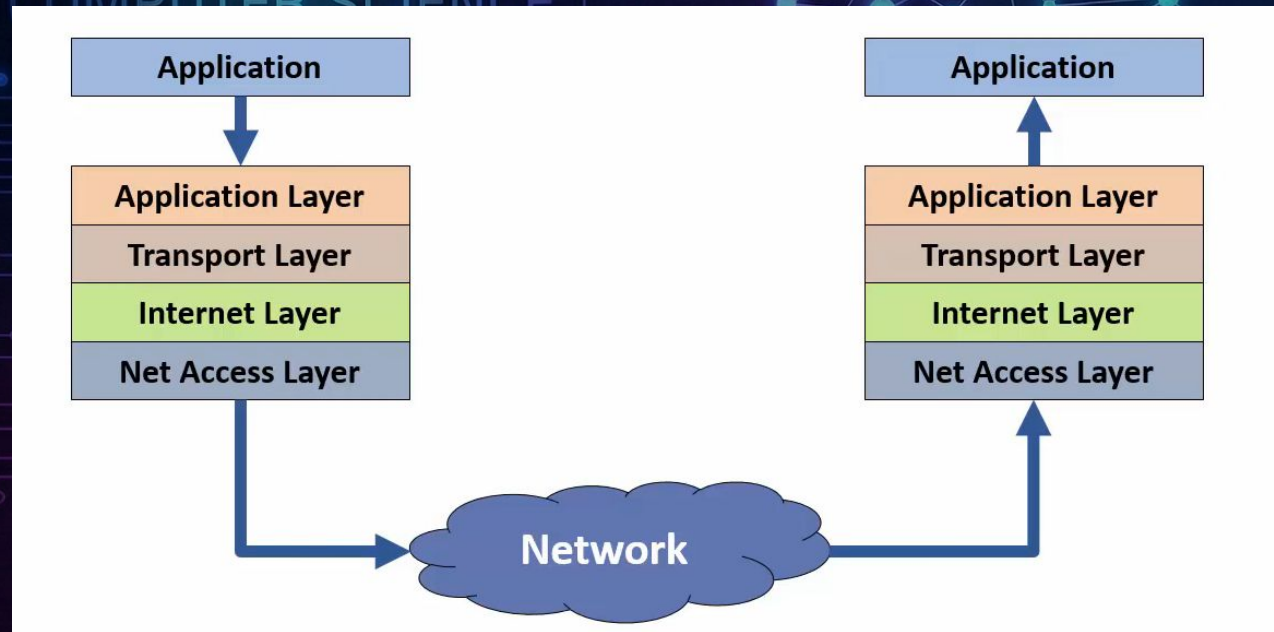
TCP/IP: Funzionamento di base

- L'idea della **stratificazione** . Ad esempio, il **protocollo HTTP è costruito sopra TCP** . Un browser non deve preoccuparsi di come TCP è implementato, deve solo sapere che funziona.
- I dati che il livello di trasporto riceve dal livello applicativo vengono frammentati in pacchetti. I pacchetti di dati vengono riassemblati a destinazione (ogni pacchetto può seguire percorsi diversi).
- Il protocollo TCP aggiunge informazioni aggiuntive a ciascun pacchetto , come ad esempio il numero di sequenza di cui fa parte.
- Il pacchetto viene quindi passato al livello di rete, dove il **protocollo IP instrada i pacchetti verso l'host di destinazione** nel modo più appropriato.

MACHINE LEARNING



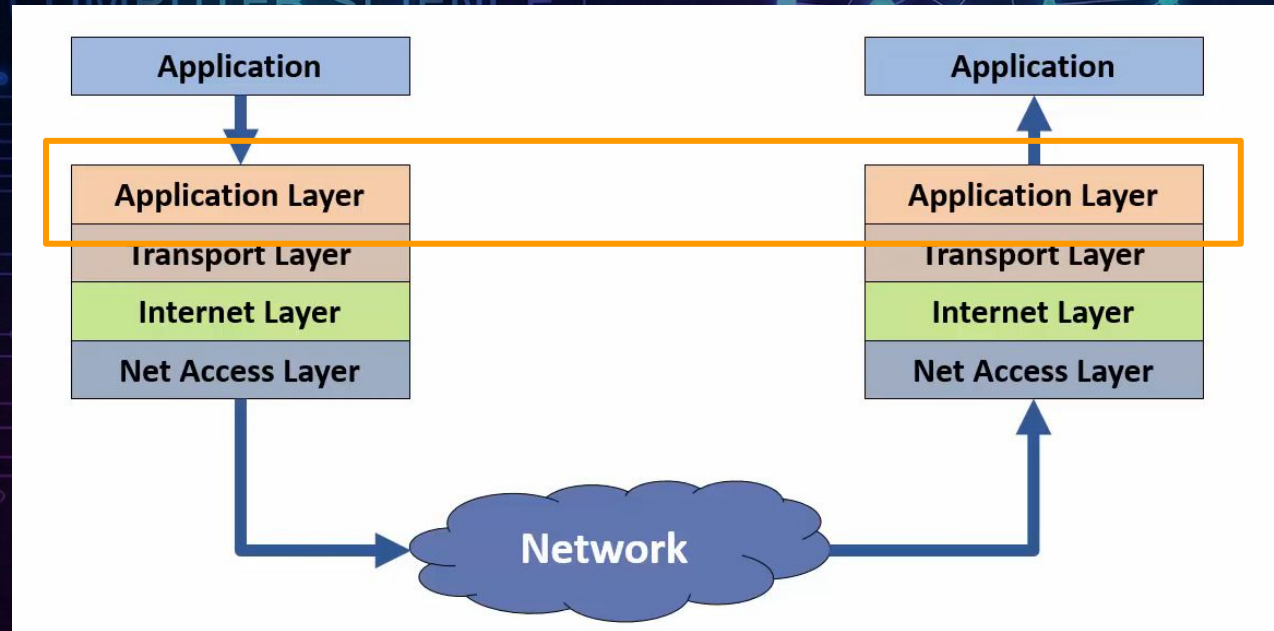
TCP/IP



LEARNING



TCP/IP



LEARNING

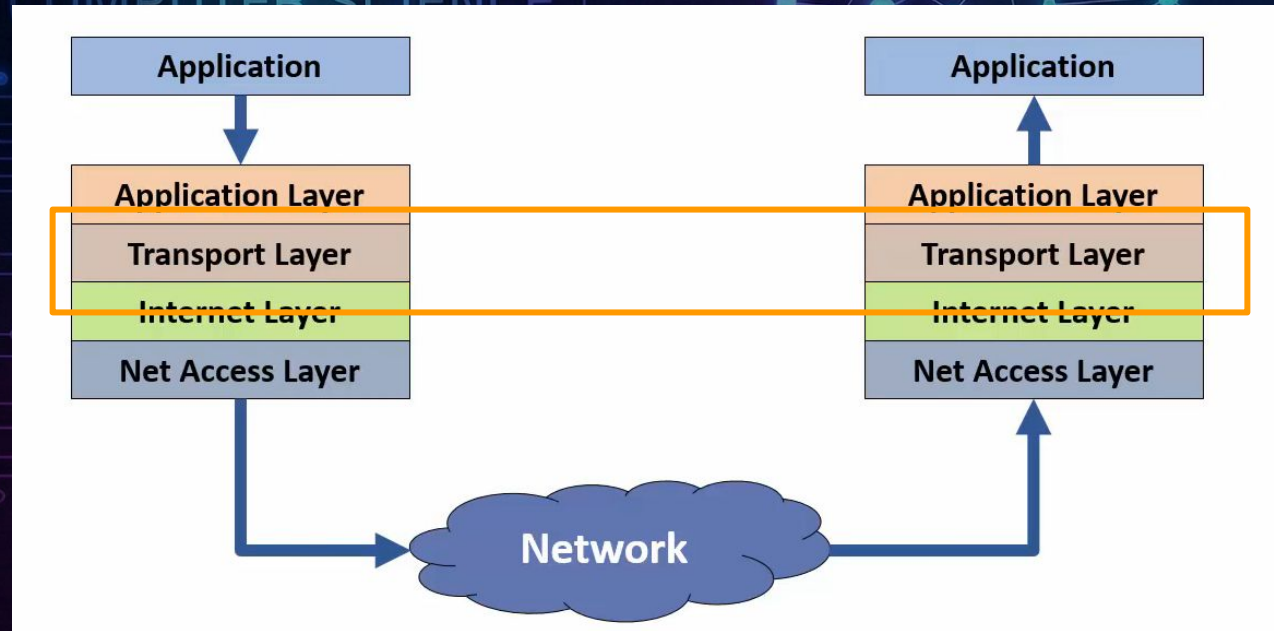


Lo stack TCP/IP: livello applicazione

- **Livello applicazione** : troviamo numerosi protocolli come HTTP, FTP, DNS, SMTP, ecc. A questo livello due applicazioni si scambiano **messaggi** senza preoccuparsi di come questi verranno recapitati.
- Questa è l'interfaccia con l'utente; ad esempio, quando consultiamo una pagina web, il protocollo gestisce la sessione di interazione tra il nostro browser (client) e il server web.



TCP/IP



LEARNING



STACK TCP/IP: Livello di trasporto

Il livello di trasporto è responsabile della consegna dei dati da un processo all'altro. Preleva i dati dal livello applicativo, li elabora e si assicura che raggiungano l'applicazione corretta sull'host di destinazione.

- **Segmenti: L'unità di dati scambiata a questo livello è chiamata segmento (o datagramma in UDP).**
- **Fornitore di servizi: offre servizi di trasporto direttamente al livello applicativo superiore (HTTP, FTP, ecc.).**
- **Due protocolli principali: TCP (Transmission Control Protocol) e UDP (User Datagram Protocol).**



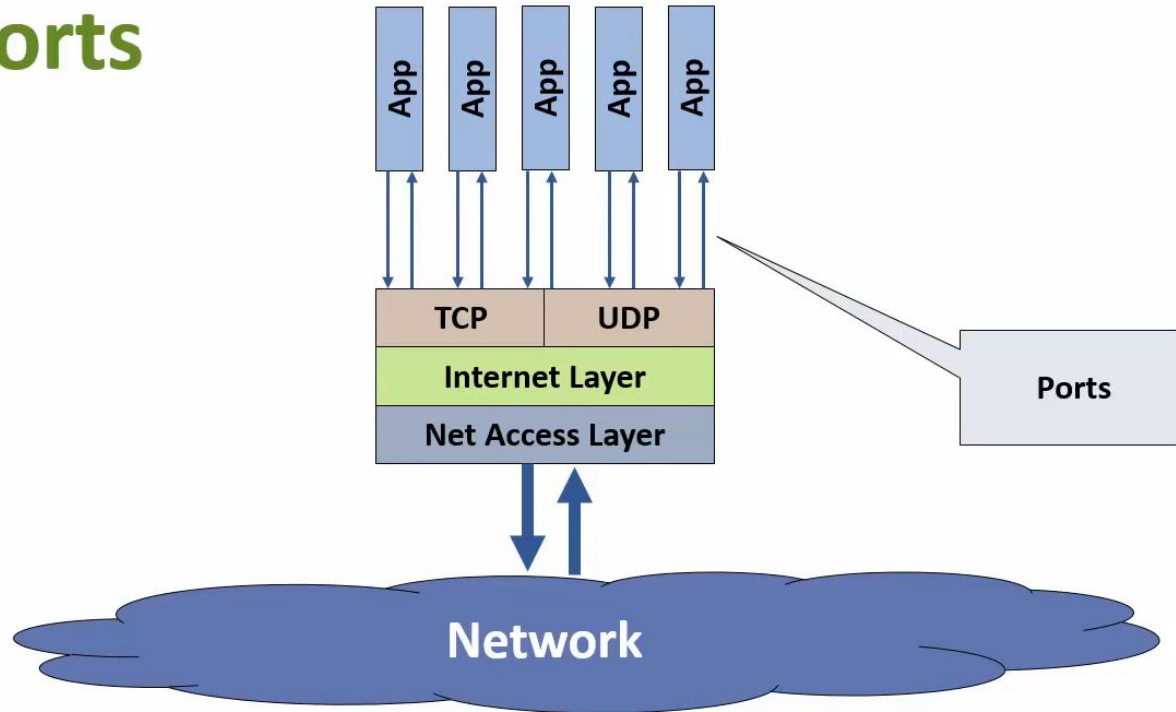
STACK TCP/IP: Livello di trasporto

- **Livello di trasporto** : a questo livello troviamo i due protocolli di base **TCP** e **UDP**; a questo livello due host si scambiano segmenti.
- I protocolli a questo livello **offrono il servizio di trasporto** al livello applicativo. Ad esempio, per gestire **più sessioni attive contemporaneamente, TCP e UDP utilizzano numeri di porta** (porte logiche) diversi.
- **TCP**
 - Per ogni finestra di pacchetti inviati, TCP avvia un contatore temporale
 - Il destinatario invia un **ACK** se ha ricevuto il pacchetto.
 - Se il mittente non riceve un **ACK** prima della scadenza del tempo (o...). Il mittente, ad esempio, si occupa di reinviare i dati



STACK TCP/IP: Livello di trasporto

Ports



101010101010101010
010110101001010101
001010111010110101
010111111001011100
110110101011010101
101010101010101010
110110101000100010
001010111010101010
001010111010100101
010111111001010101
110110101011011101

LEARNING



STACK TCP/IP: Livello di trasporto

MULTIPLEXING

Host IP: 192.168.1.5

Port 80 **Web (HTTP)**

Port 443 **Secure (HTTPS)**

Port 25 **Email (SMTP)**

How does a computer run a web browser, an email client, and a multiplayer game all at the same time on a single network connection?

The Apartment Analogy

The **IP Address** is like the street address of an apartment building (the Host).

The **Port Number** (Logical Port) is like the specific apartment number inside that building (the Application Session).



STACK TCP/IP: Livello di trasporto

Il TCP (Transmission Control Protocol) è progettato per garantire la massima affidabilità. Assicura che ogni singolo segmento arrivi correttamente e nell'ordine giusto.

- **Windows: TCP invia i pacchetti in gruppi chiamati "finestre".**
- **Contatore del tempo: per ogni finestra/pacchetto inviato, TCP avvia un timer per il conto alla rovescia.**
- **ACK (Conferma di ricezione): Il destinatario DEVE inviare una conferma di ricezione (ACK) se riceve correttamente il pacchetto.**
- **Ritrasmissione: se il mittente non riceve una conferma (ACK) prima dello scadere del tempo, presume che il pacchetto sia andato perso e ritrasmette i dati.**



STACK TCP/IP: Livello di trasporto

Il protocollo UDP (User Datagram Protocol) privilegia la velocità rispetto all'affidabilità. Non stabilisce una connessione, non utilizza timer e non richiede conferme di ricezione (ACK).

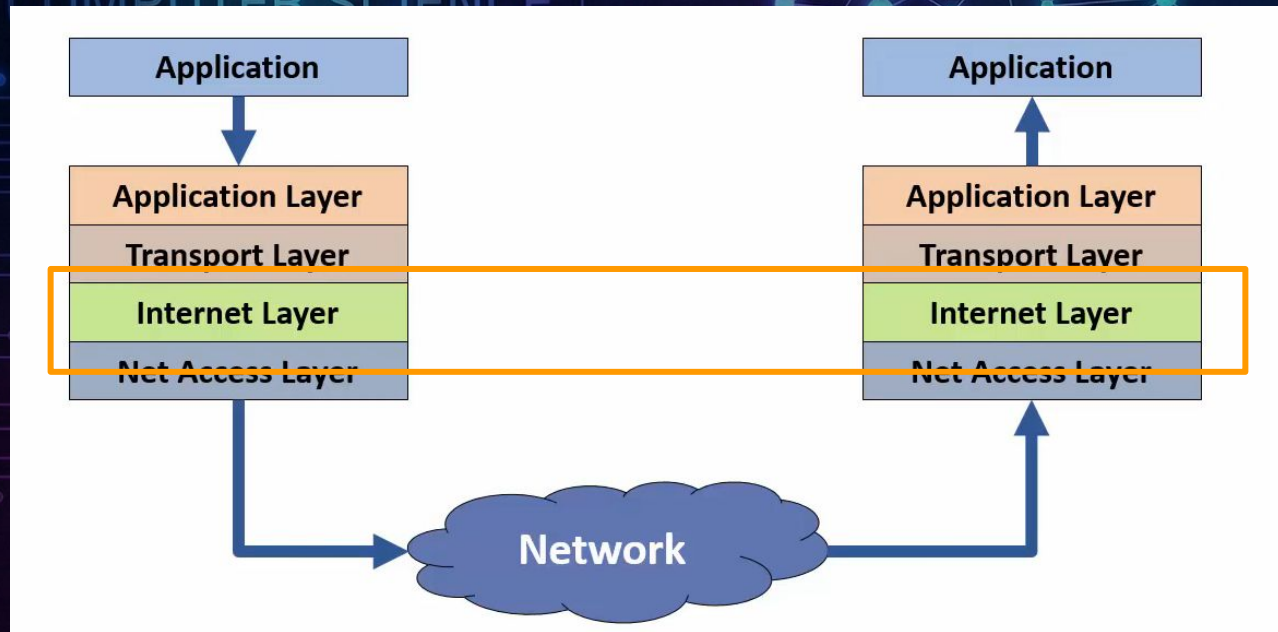
"Spara e dimentica"

Il protocollo UDP si limita a inviare i segmenti di dati al destinatario il più velocemente possibile. Se un pacchetto viene perso lungo il percorso, rimane perso. Il mittente non lo reinvia mai.

MACHINE LEARNING



TCP/IP



LEARNING



Lo stack TCP/IP: livello di rete

- **A questo livello troviamo il protocollo IP (Internet Protocol). Mentre il livello di trasporto gestisce la sessione tra le applicazioni, il livello di rete si occupa dell'effettiva consegna dei dati da una macchina fisica all'altra attraverso più reti.**
- **Due responsabilità principali:**
 - **Indirizzamento: Assegnazione di un'identità univoca a ciascun nodo.**
 - **Pianificazione del percorso: trovare il percorso migliore attraverso il labirinto di internet.**



Lo stack TCP/IP: livello di rete

Proprio come il servizio postale necessita di un indirizzo stradale valido per recapitare una lettera, Internet necessita di indirizzi IP.

- **Univoco:** ogni nodo (computer, telefono, server, frigorifero intelligente) connesso a una rete è identificato in modo univoco da un indirizzo IP.
- **Strutturato:** l'indirizzo non è casuale. Contiene informazioni sulla sottorete specifica a cui appartiene il dispositivo, consentendo ai router di sapere in quale direzione generale inviare i dati.

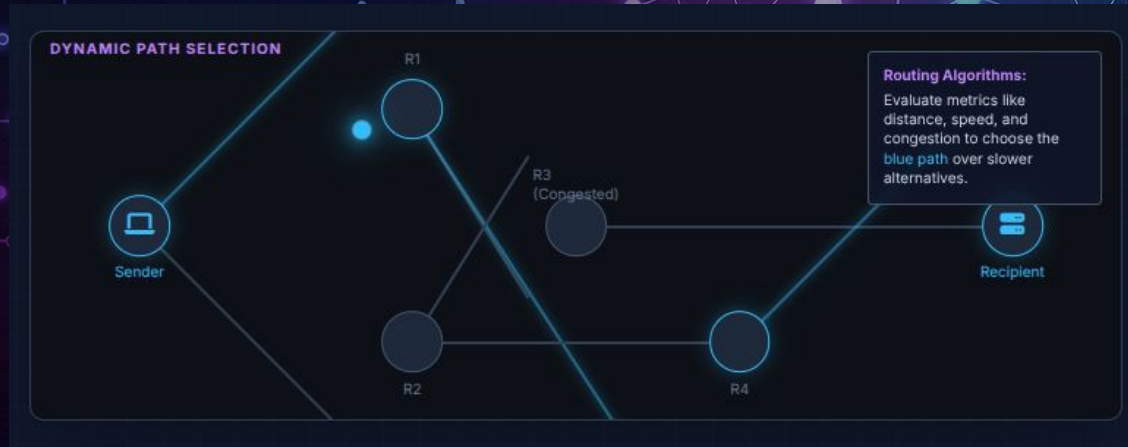


MACHINE LEARNING



Lo stack TCP/IP: livello di rete

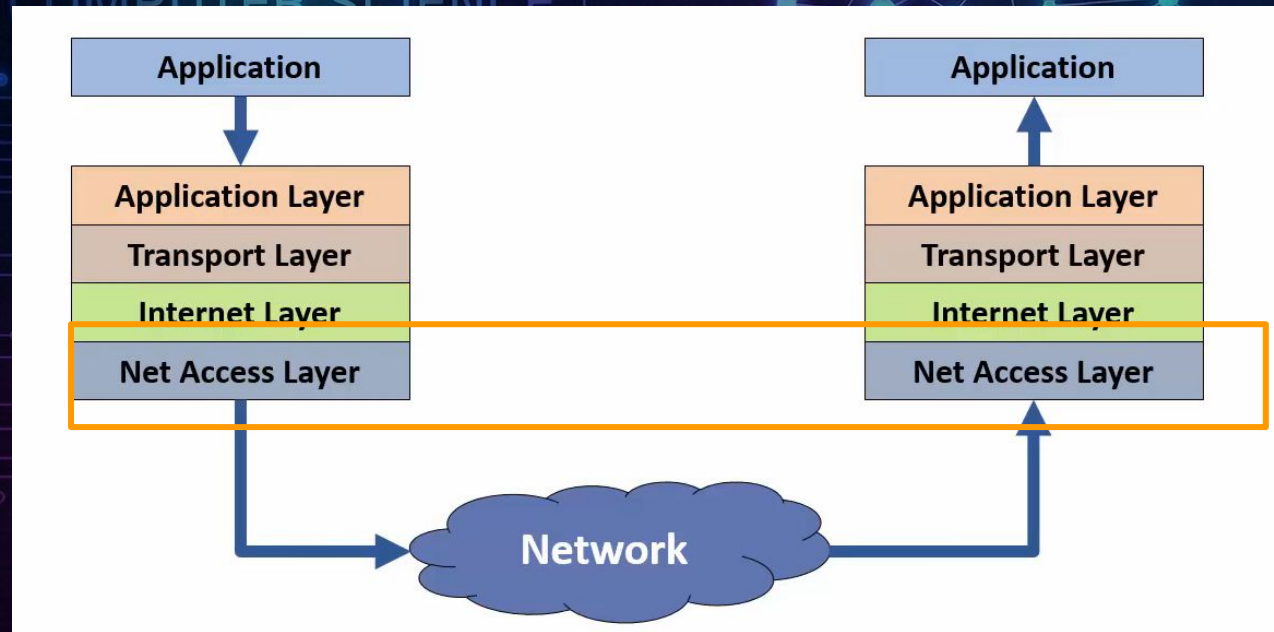
Il routing è la funzionalità che consente alla rete di selezionare il percorso migliore da seguire per far arrivare i pacchetti di dati dal mittente al destinatario attraverso più router interconnessi.



NE LEARNING



TCP/IP



LEARNING



Lo stack TCP/IP: livello di accesso alla rete

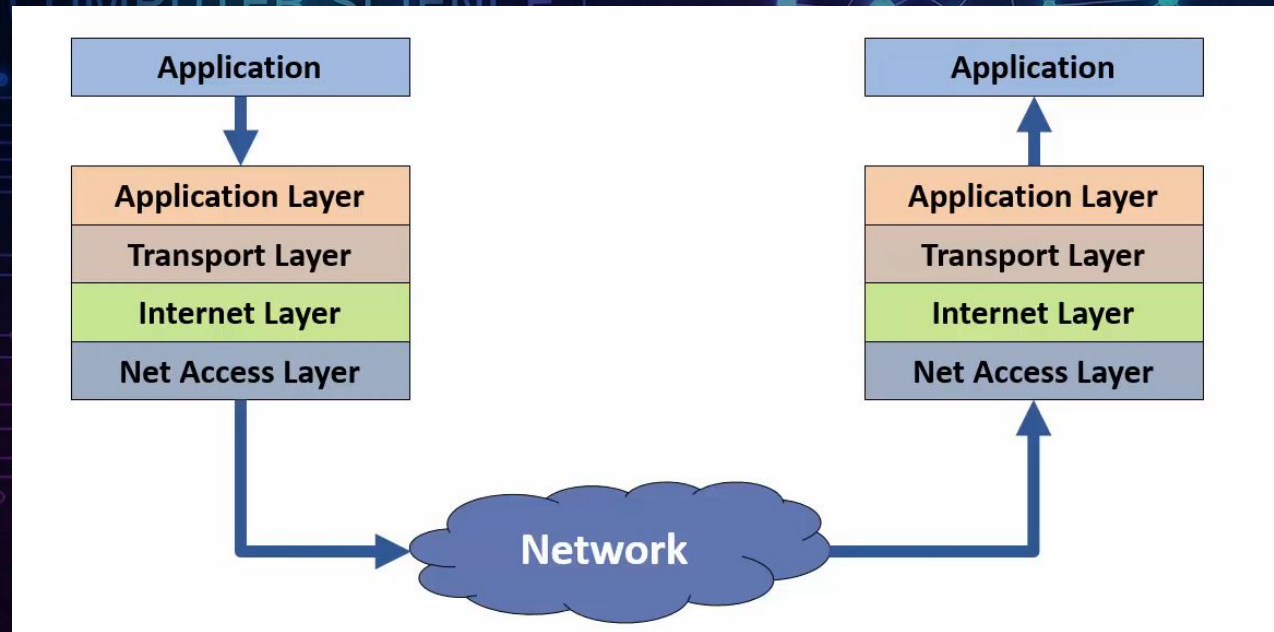
- Il modello TCP/IP specifica solo che al di sotto del livello IP deve esserci un livello di accesso alla rete che si occupi effettivamente dell'invio dei pacchetti.
- A livello di collegamento, i protocolli decidono come il messaggio deve essere trasferito per ogni fase del percorso. Ad esempio, come passare dal primo host al primo router e così via (**indirizzi MAC ed Ethernet**).
- A livello fisico, i dati vengono quindi convertiti in **segnali elettrici o elettromagnetici oppure...**

STATISTICS

MACHINE LEARNING



TCP/IP

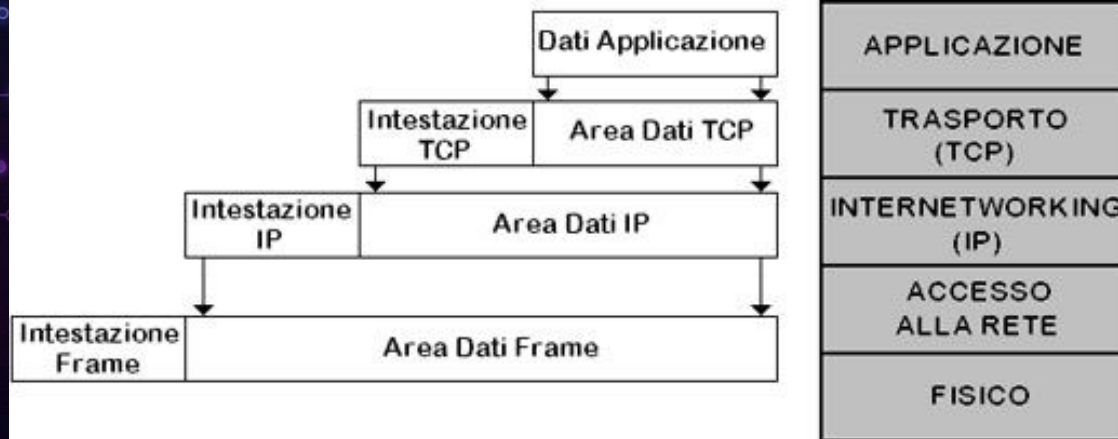


LEARNING



Lo stack TCP/IP

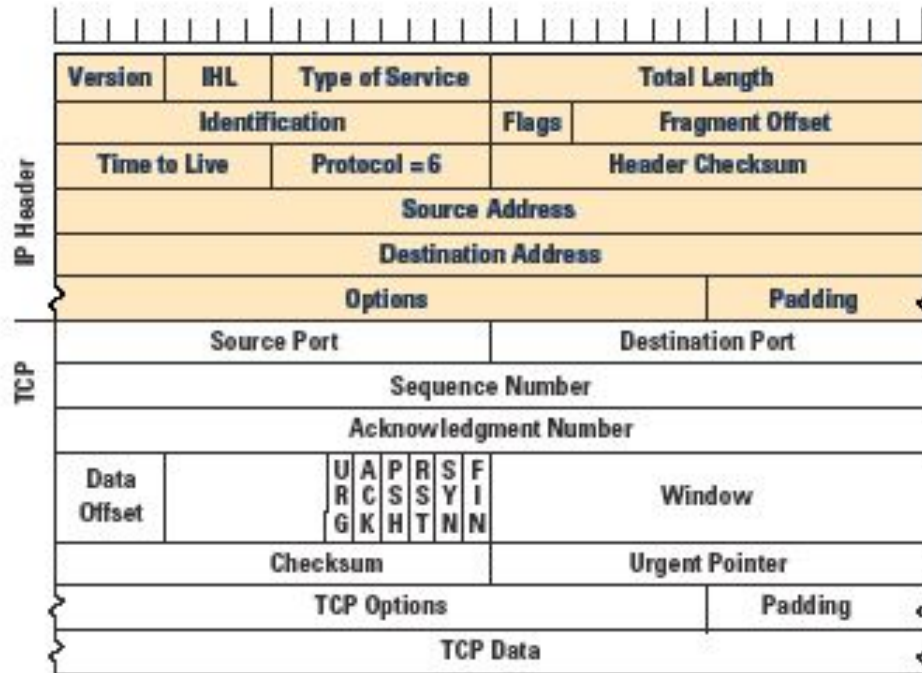
- Quando un'applicazione deve inviare dati, questi vengono trasmessi attraverso i vari livelli, uno alla volta, fino a raggiungere la rete fisica sottostante. Lungo il percorso, ogni livello aggiunge informazioni ai dati, creando una "struttura di rete" (incapsulamento):



INE LEARNING

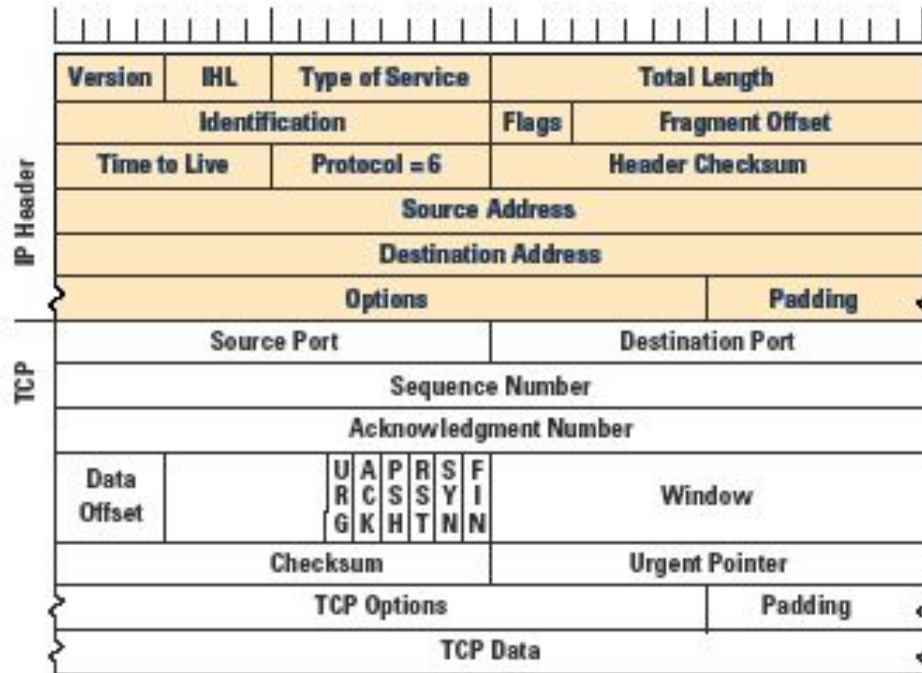


TCP/IP: intestazioni



L'intestazione TCP (la logistica): questo è il livello interno. Gestisce la comunicazione specifica tra le applicazioni. Utilizza i numeri di porta per garantire che i dati vengano inviati al programma corretto (come Chrome rispetto a Spotify) e i numeri di sequenza per rimettere i pacchetti nell'ordine corretto se arrivano incompleti.

TCP/IP: intestazioni



Intestazione IP (l'indirizzo): questo è il livello più esterno. Gestisce l'instradamento attraverso la rete. Contiene gli indirizzi IP di origine e di destinazione per garantire che il pacchetto trovi la macchina fisica corretta nella vastità di Internet.

COMPUTER SCIENCE

INDIRIZZI IP, DNS, DHCP E...

STATISTICS

MACHINE LEARNING



indirizzi IP

- Ogni computer connesso a Internet è identificato dal suo indirizzo IP , che è composto da quattro gruppi di un byte ciascuno (32 bit in totale). Ogni numero può assumere valori da 0 a 255.
- Ad esempio: **192.167.12.66** (indirizzi IP statici o dinamici, IP privati, ecc.)
- L'ultimo numero in genere identifica un host, mentre i numeri precedenti indicano la sottorete a cui appartiene tale host.
- Il numero massimo di indirizzi IPv4 è quindi $256 \times 256 \times 256 \times 256$
- IPv6 : 128 bit e quindi 2^{128} circa $3,4 \times 10^{38}$ indirizzi (IoT: Internet delle cose)



ADDRESSING: IPV4 VS. IPV6

IPv4 (Legacy)

- **Structure:** 32-bit address (4 bytes).
- **Notation:** Dotted decimal (e.g., 192.168.1.1).
- **Capacity:** 2^{32} addresses (~4.3 Billion).
- **Limit:** **Exhausted** (Allocations depleted in 2011).

IPv6 (Future Standard)

- **Structure:** 128-bit address (16 bytes).
- **Notation:** Hexadecimal (e.g., 2001:0db8::).
- **Capacity:** 2^{128} addresses.
- **Scale:** 3.4×10^{38} (Undecillion).

Why the shift?

The explosion of IoT (Internet of Things) devices requires a virtually infinite address space that IPv4 cannot provide.

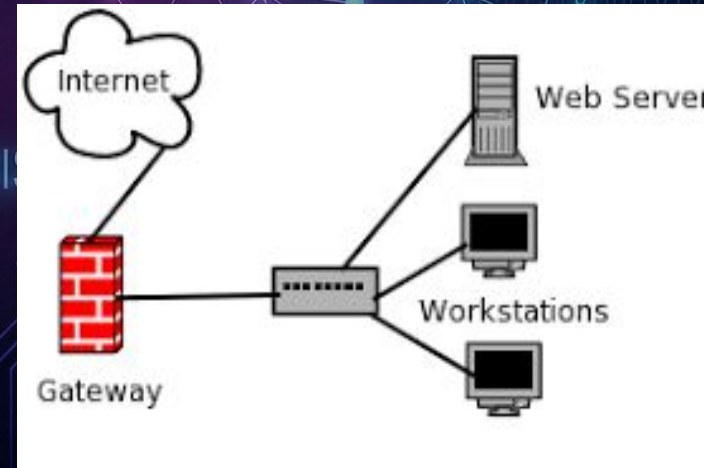
0101010101010
0101001010101
0111010110101
1111001011100
0101011010101
0101010101010
0101000100010
0111010101010
0111010100101
1111001010101
0101011011101

RNING



Indirizzi IP: Gateway

- Il gateway predefinito viene utilizzato per instradare i pacchetti verso altre destinazioni.
- Il DHCP viene quasi sempre utilizzato per fornire automaticamente al client l'indirizzo IP del gateway predefinito.



Due domini di indirizzamento



Public IP

The global identity of a network. It is completely unique across the entire Internet and allows external devices to find and communicate with your network.

Assigned by your **ISP** (Internet Service Provider).

Routable on the global internet.

Think of it as your home's **Mailing Address**.

Example Output:

203.0.113.42



Private IP

The internal identity within a Local Area Network (LAN). It is only unique inside your specific home or office network, allowing local devices to talk to each other.

Assigned by your local **Router** (via DHCP).

Hidden from the outside internet.

Think of it as your specific **Room Number**.

Common Range:

192.168.1.105

101010101010
101001010101
111010110101
111001011100
101011010101
101010101010
101000100010
111010101010
111010100101
111001010101
101011011101

NING



Il Bridge: Traduzione degli indirizzi di rete (NAT)

COMPUTER SCIENCE

If Private IPs are hidden from the internet, how does your phone load a webpage? The answer is **NAT**.

 **The Translator:** Your router acts as a gateway. It takes requests from your internal **Private IP** and translates them to its single external **Public IP**.

 **Inherent Security:** Because external internet traffic only sees the router's Public IP, your individual devices remain shielded behind the NAT firewall.

 **IPv4 Conservation:** NAT allows dozens of devices in a home to share a single Public IP, drastically slowing the depletion of global IPv4 addresses.



La maschera di sottorete (il divisore)

How does a computer know exactly where the Network part ends and the Host part begins? It uses a secondary number called the **Subnet Mask**.

▼ **The Filter:** The mask acts like a digital stencil placed over the IP address.

The 255s: Wherever the subnet mask has a **255**, it tells the computer: *"This section belongs to the Network."*

The 0s: Wherever the subnet mask has a **0**, it tells the computer: *"This section belongs to the Host."*

Standard Home Setup:

IP: 192.168.1.105

Mask: 255.255.255.0



Esempio: una tipica rete domestica

Observe how the **Network Part** remains identical across all devices on the same Wi-Fi, while the **Host Part** uniquely identifies each piece of hardware.

Device	Full IP Address	 Network Part	 Host Part
Wi-Fi Router	192.168.1.1	192.168.1	.1
 Your Phone	192.168.1.105	192.168.1	.105
 Your Laptop	192.168.1.106	192.168.1	.106
 Smart TV	192.168.1.110	192.168.1	.110

Esempio 2: Una rete aziendale

SUBNET MASK: 255.255.0.0

10.50.120.5

NETWORK PART

HOST PART

← The Shifted Boundary

By changing the mask to `255.255.0.0`, the dividing line moves to the left. Now, only the first two numbers define the network.

🏢 Massive Capacity

This leaves two full numbers for the Host Part. Instead of just 254 devices, this large corporate network can hold over **65,000** unique devices!

EARNING



COMPUTER SCIENCE

DNS

STATISTICS

MACHINE LEARNING



DNS

- Per gli esseri umani è difficile memorizzare i numeri, mentre è molto più facile memorizzare i nomi. Ecco perché esistono i servizi DNS (**Domain Name System**). Si tratta di sistemi utili per tradurre nomi e indirizzi in entrambe le direzioni.

```
redo@eeegw:~$ host www.storchi.org
www.storchi.org has address 82.221.102.244
redo@eeegw:~$ host gw-thch.unich.it
gw-thch.unich.it has address 192.167.12.66
redo@eeegw:~$ host 192.167.12.66
66.12.167.192.in-addr.arpa domain name pointer gw-thch.unich.it.
redo@eeegw:~$
```

INE LEARNING



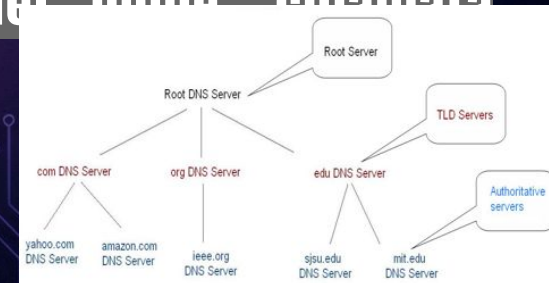
DNS

- Ciascun host viene quindi identificato dall'utente tramite un nome simbolico:
 - gw-thch.unich.it
- I nomi vengono assegnati in modo univoco e gestiti amministrativamente secondo una gerarchia.
- I nomi identificano in modo univoco un host all'interno di un dominio:
 - è il dominio
 - unich è il sottodominio all'interno di it
- I domini principali sono:
 - .gov .edu .com essenzialmente negli Stati Uniti associato al tipo di organizzazione
 - I vari paesi invece hanno domini del tipo: .it, .uk, .fr, .de



DNS

- Prima dell'introduzione del sistema DNS, la corrispondenza tra indirizzi IP e nomi era gestita dalla scheda di rete SRI-NIC, che essenzialmente manteneva un elenco in un file hosts.txt.
- In pratica, **il DNS è un database distribuito**. Le informazioni sono distribuite su molti computer, server DNS, ognuno dei quali è responsabile di una determinata porzione del nome, chiamato **dominio**.



La struttura ad albero gerarchica

Servers in the DNS system are not just a random flat list. They are organized in a strict **hierarchical tree structure**, read from right to left.

- **Root Level (.):** The top of the tree. They know where the TLD servers are.
- **TLD Level:** Top-Level Domains like `.org`, `.com`, `.it`.
- **Authoritative Level:** The specific domain (e.g., `storchi`). These hold the actual IP records.
- **Subdomains:** The final specific host (e.g., `www`).



DNS

I server sono organizzati in una **struttura ad albero gerarchica**. Ecco il flusso completo, passo dopo passo, di ciò che accade quando il tuo browser richiede un sito come www.storchi.org:

1. **Richiesta e verifica della cache** : il dispositivo richiede l'indirizzo IP al resolver DNS della rete locale (generalmente fornito dal provider di servizi Internet). Il resolver verifica innanzitutto nella sua cache locale se l'indirizzo è già presente in una richiesta recente.
2. **Server radice** : se l'indirizzo IP non è presente nella cache, il resolver interroga un server DNS radice. Il server radice non conosce l'indirizzo IP esatto, ma sa quali server gestiscono l'estensione .org.
3. **Il server TLD**: il server radice indirizza il resolver al server del dominio di primo livello (TLD) per .org.
4. **Il server autorevole** : il server TLD .org non conosce l'indirizzo IP per "www". Invece, indirizza il resolver al server dei nomi autorevole specificamente registrato e responsabile per storchi.org.
5. **Risposta definitiva**: Il server autorevole cerca il record www specifico, trova l'indirizzo IP esatto e lo invia al tuo resolver locale. Il tuo resolver lo memorizza nella cache per la volta successiva e lo trasmette al tuo browser.



Protocolli di alto livello

- Vengono utilizzati diversi tipi di protocolli, ognuno per un servizio specifico:
 - **HTTP (HyperText Transfer Protocol)** Accesso alle pagine ipertestuali (WEB) all'interno del WWW (https crittografato)
 - Trasferimento e copia di file tramite **FTP (File Transfer Protocol)**.
 - **SMTP (Simple Mail Transfer Protocol)** per l'invio di messaggi di posta elettronica, **POP3** per il download di messaggi di posta elettronica sul computer, **IMAP** utile per controllare i messaggi da più dispositivi.

Una risorsa sulla rete viene quindi “identificata” tramite l'URL: **LEARNING**

<http://hostname.it/index.html>



COMPUTER SCIENCE

Firewall, DHCP, ...

STATISTICS

MACHINE LEARNING



IP ADDRESSES: FIREWALL



The firewall is a network security system that controls all incoming and outgoing traffic according to well-defined rules.

IP Addresses: DMZ

A DMZ (demilitarized zone) is a physical or logical subnetwork that contains and exposes an external organization's services to an unprotected network, usually a larger network such as the Internet.



DHCP: l'esempio del router domestico

The Problem

Every device on a network needs an IP address. Manually typing an IP address into your phone, smart TV, and laptop every time you connect to Wi-Fi would be a nightmare.

🔧 The DHCP Solution

- The **ADSL/Fiber Router you have at home** has a built-in DHCP Server.
- It maintains a "pool" (database) of available Private IP addresses (e.g., 192.168.1.2 to 192.168.1.254).
- When a new device connects, the router automatically **leases** an IP address to it so it can access the network instantly.

Your Home Router

Doing multiple jobs at once:

📶 Access Point

🛡️ NAT / Firewall

🖨️ DHCP Server

Assigns IPs