

**MPS - Reti**

# Cosa sono Bit e Byte?

I computer non capiscono le lettere o i numeri, ma comunicano usando un linguaggio molto semplice, fatto di solo due "parole": acceso e spento (on e off).

Queste due parole, che corrispondono ai valori 1 e 0, sono chiamate bit.

**Bit** (binary digit): è l'unità più piccola di informazione in informatica. Un bit è come un interruttore della luce: può essere solo acceso (1) o spento (0). Tutti i dati gestiti dal computer, dalle foto ai video, sono solo una combinazione di bit.

**Byte**: per comodità i bit vengono raccolti a gruppi di 8, un byte è quindi un gruppo di otto bit, è un po' come una lettera del nostro alfabeto.

Combinando otto interruttori (bit), il computer può creare 256 combinazioni diverse ( $2^8 \Rightarrow 256$ ). Questo permette di rappresentare una grande varietà di informazioni, come lettere, numeri e simboli.

# Cosa sono Bit e Byte?

**Un carattere:** la lettera 'A' sul computer è rappresentata da un byte specifico, cioè una sequenza di otto bit. Ad esempio, potrebbe essere 01000001.

**Un file di testo molto semplice,** che contiene solo la parola "CIAO", é composto da quattro byte (uno per ogni lettera). Ogni byte, a sua volta, è una sequenza di otto bit. Quindi, in totale, il file è composto da 32 bit.

**Dimensioni dei file:** quando si esaminano le dimensioni dei file, come 1 KB (kilobyte) o 1 MB (megabyte), si sta misurando quanti byte di informazione contiene quel file.

Kilobyte (KB): 210 byte = 1.024 byte.

Megabyte (MB): 220 byte = 1.048.576 byte.

Gigabyte (GB): 230 byte = 1.073.741.824 byte.

Terabyte (TB): 240 byte = 1.099.511.627.776 byte.

# Rappresentazione esadecimale

**Si usa la rappresentazione esadecimale per rendere i numeri binari più compatti e leggibili per gli esseri umani, i computer lavorano con il sistema binario (0 e 1), che è molto lungo e difficile da gestire.**

**Il Vantaggio Principale é che Il sistema esadecimale (base 16) ha una relazione diretta con il sistema binario (base 2), poiché  $16=2^4$ . Ciò significa che ogni singola cifra esadecimale può rappresentare esattamente un gruppo di quattro bit.**

**Esempio: un byte, composto da 8 bit, può essere rappresentato da due sole cifre esadecimali.**

**Binario: 11010010**

**Esadecimale: D2 (==> 1101 0010)**

**Decimale: 210**

**In questo modo, leggere e scrivere una sequenza di 0 e 1 diventa molto più semplice.**

**<http://www.giulianosrl.com/Con-Esa.htm>**

# Rappresentazione esadecimale

## Applicazioni Pratiche

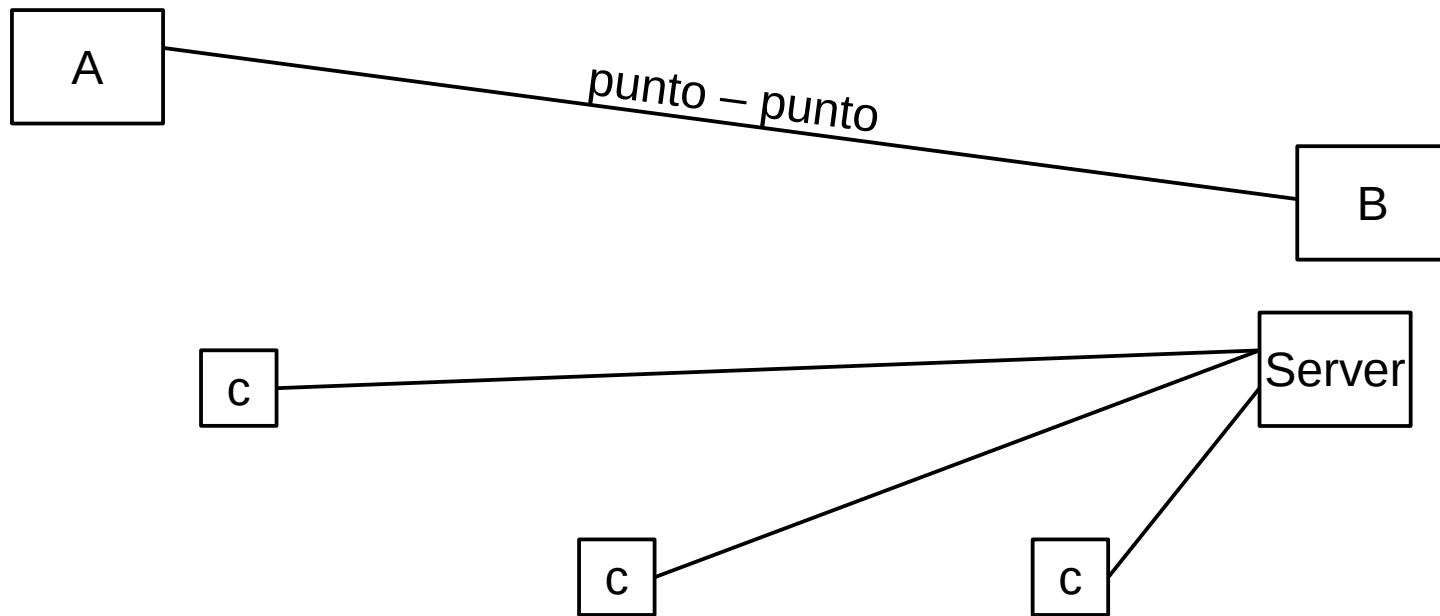
**La notazione esadecimale viene utilizzata in diversi ambiti dell'informatica per migliorare la leggibilità, tra cui:**

**Indirizzi di memoria:** in informatica, le posizioni della memoria di un computer sono spesso rappresentate in esadecimale, è molto più semplice scrivere **A1B4** al posto di **1010000110110100**.

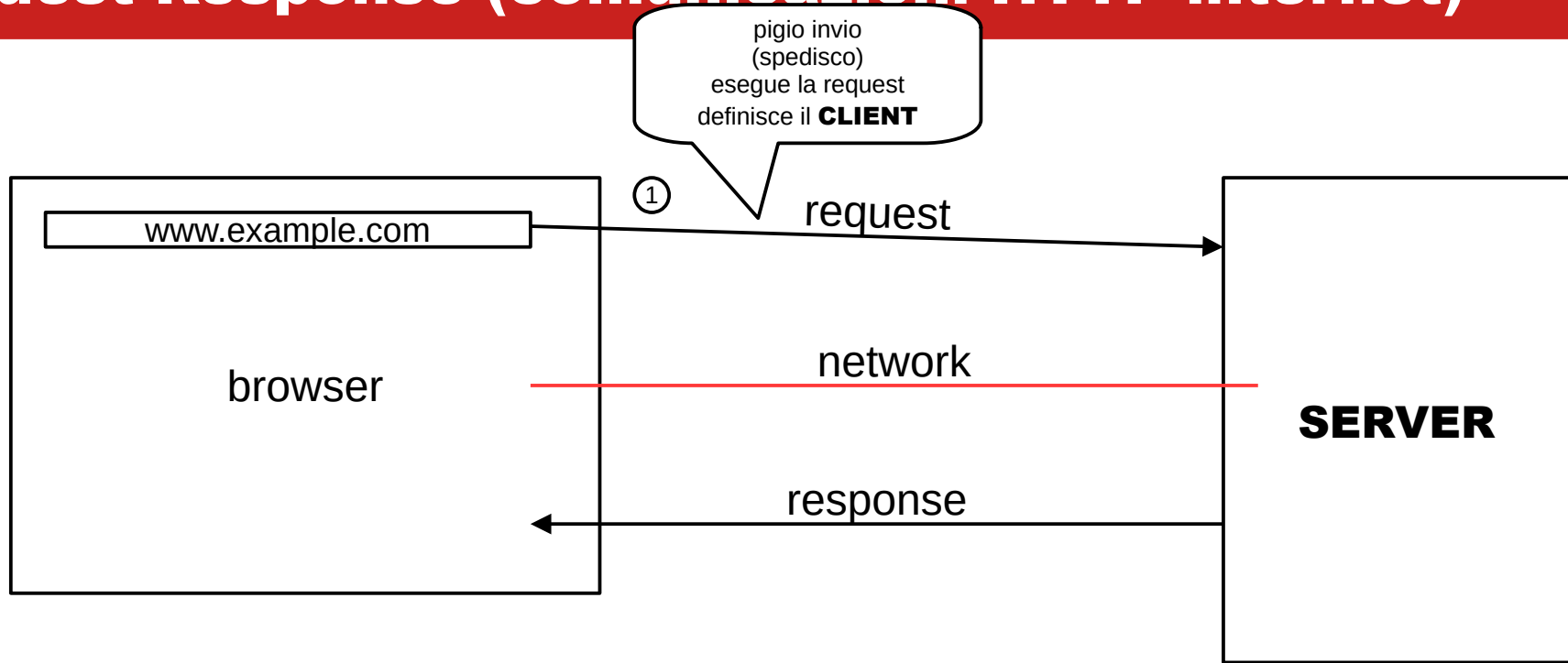
**Codici colore:** nel web design e nella grafica, i colori sono spesso definiti con codici esadecimali (es. **#FF0000** per il rosso puro), dove ogni coppia di cifre rappresenta l'intensità di rosso, verde e blu.

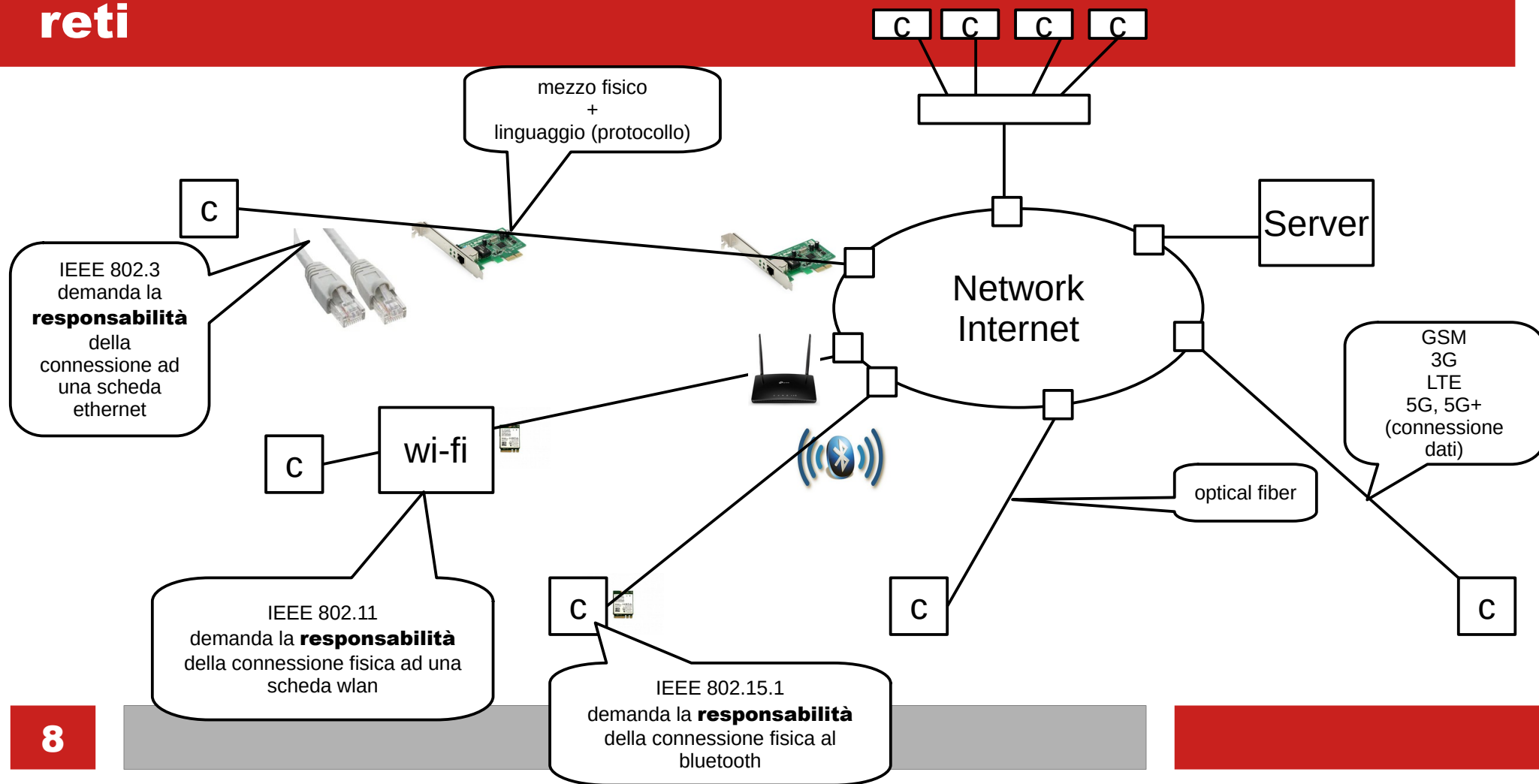
**Codifica dei caratteri:** in standard come l'ASCII o l'Unicode, i caratteri sono sovente rappresentati da valori esadecimali.

**In sintesi, la rappresentazione esadecimale non viene usata dai computer, ma è uno strumento per gli utenti teso a semplificare la visualizzazione e la manipolazione di dati binari.**



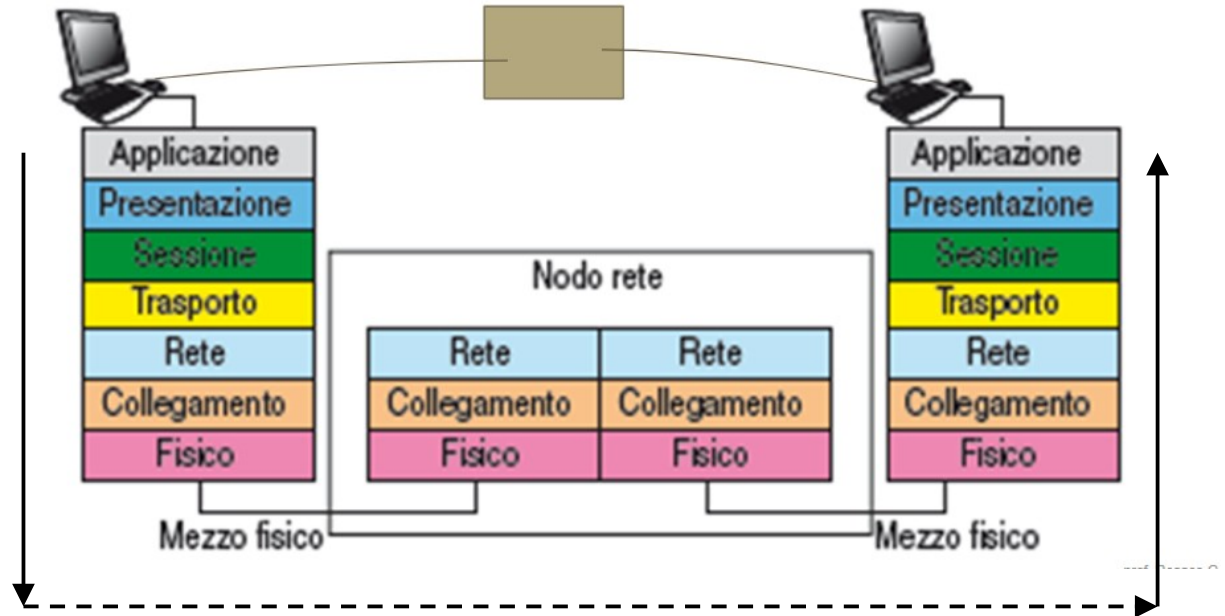
# Request Response (comunicazioni HTTP internet)



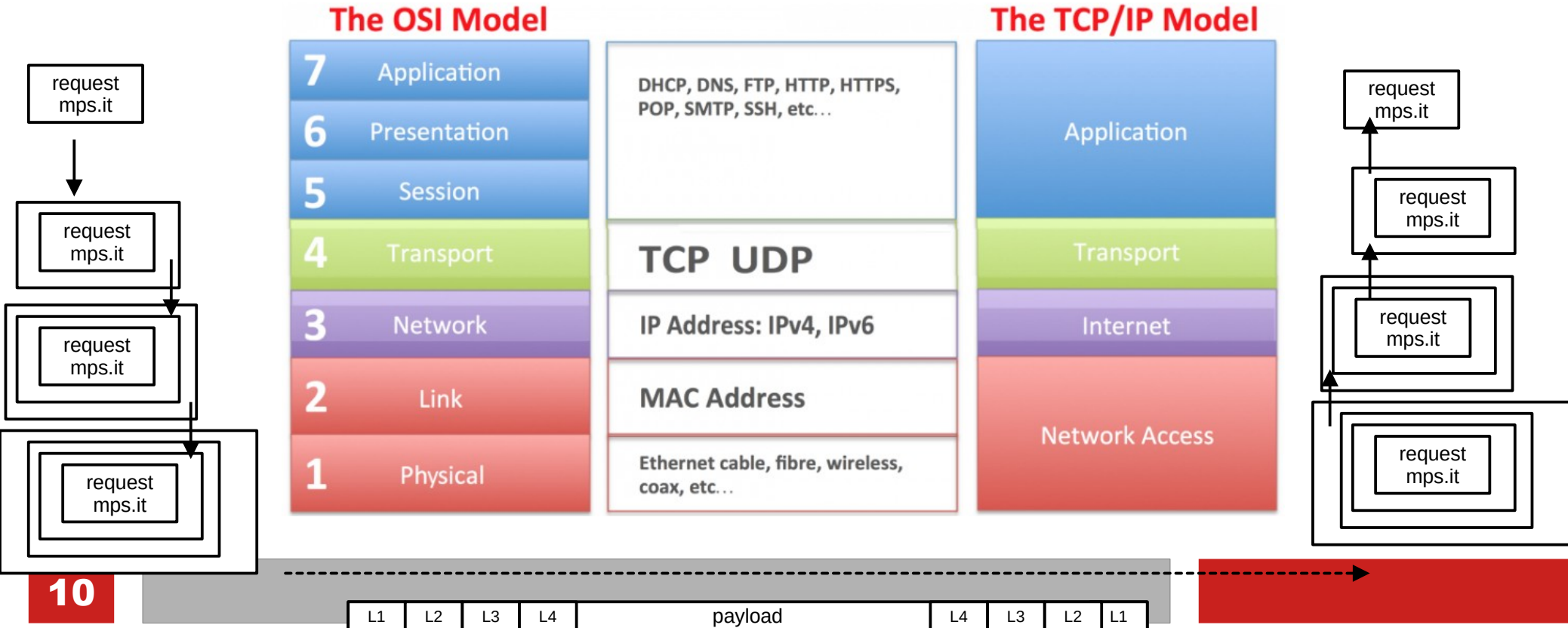




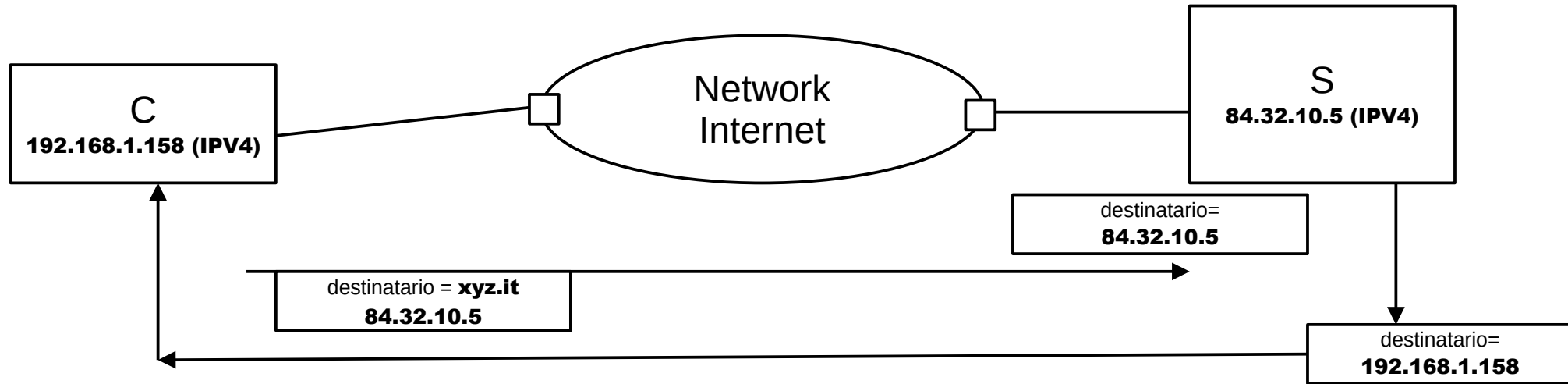
# Modello ISO/OSI



# INTERNET - TCP/IP



# TCP/IP network address



# getmac (mostra i mac address disponibili)

wlan  
card

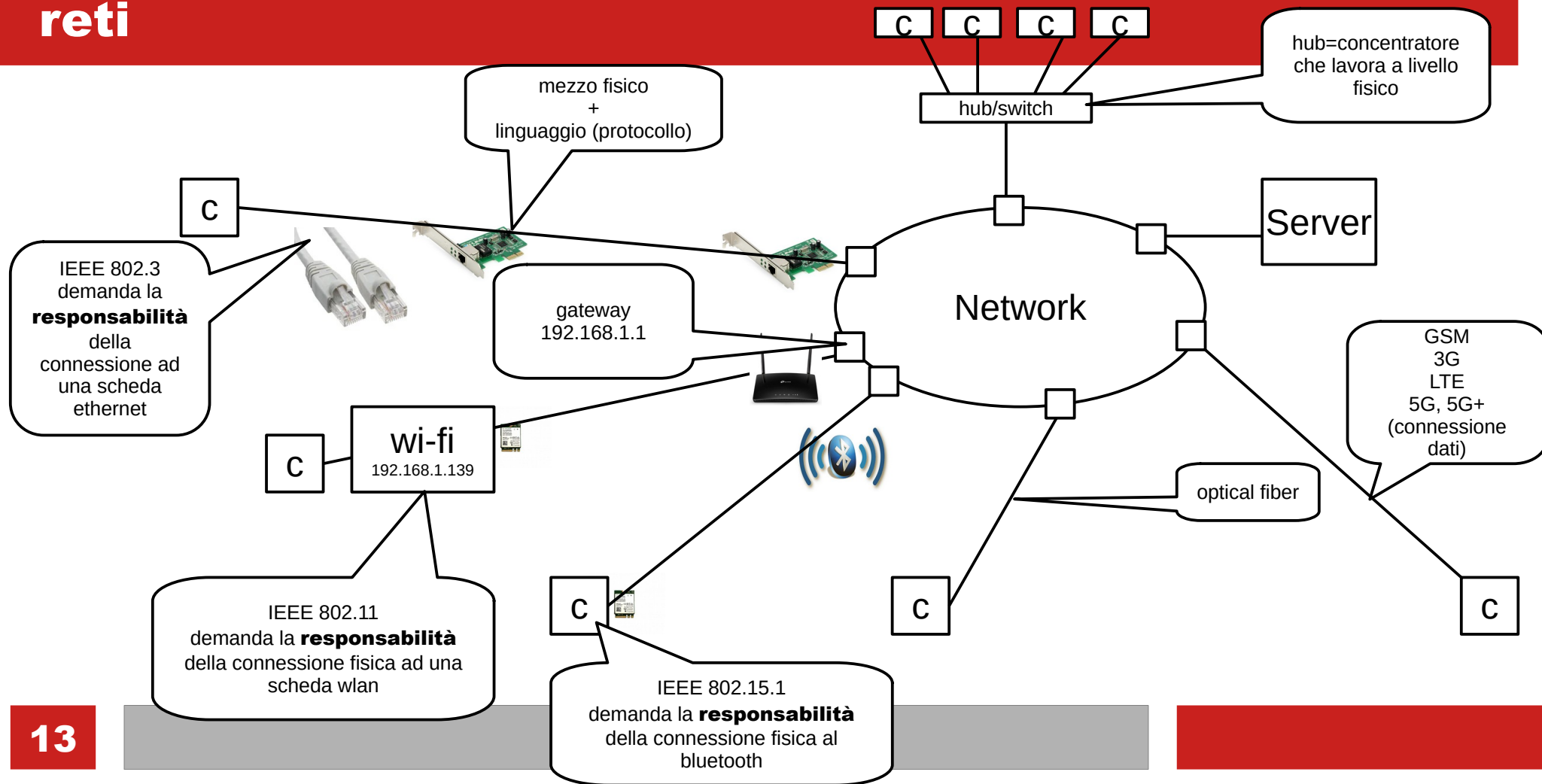
ethernet  
scollegata

```
Command Prompt
Microsoft Windows [Version 10.0.26100.4484]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Administrator>getmac

Physical Address      Transport Name
=====
A8-6D-AA-EB-12-2F     \Device\Tcpip_{32B93C2F-21C1-41D3-A5D8-2209382743F4}
A8-6D-AA-EB-12-33     Media disconnected

C:\Users\Administrator>
```



# ipconfig (mostra indirizzi di rete)

```
C:\Users\Administrator\Documents\MPS Reti>ipconfig

Windows IP Configuration

Wireless LAN adapter Local Area Connection* 1:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

Wireless LAN adapter Local Area Connection* 10:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

Wireless LAN adapter Wi-Fi:

    Connection-specific DNS Suffix  . : lan
    IPv6 Address. . . . . : fd2c:e314:a697::d93
    IPv6 Address. . . . . : fd2c:e314:a697:0:2b29:54ed:1abc:7e52
    Temporary IPv6 Address. . . . . : fd2c:e314:a697:0:f006:6867:bb93:553e
    Link-local IPv6 Address . . . . . : fe80::2ae9:b2a9:8f2c:da4%9
    IPv4 Address. . . . . : 192.168.1.139
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 192.168.1.1

Ethernet adapter Bluetooth Network Connection:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :
```

## arp -a (rivela mac address)

```
C:\Users\Administrator\Documents\MPS Reti>ping 192.168.1.1
```

```
Pinging 192.168.1.1 with 32 bytes of data:
```

```
Reply from 192.168.1.1: bytes=32 time=1ms TTL=64
```

```
Reply from 192.168.1.1: bytes=32 time=2ms TTL=64
```

```
Reply from 192.168.1.1: bytes=32 time=3ms TTL=64
```

```
Reply from 192.168.1.1: bytes=32 time=1ms TTL=64
```

```
Ping statistics for 192.168.1.1:
```

```
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
```

```
Approximate round trip times in milli-seconds:
```

```
    Minimum = 1ms, Maximum = 3ms, Average = 1ms
```

```
C:\Users\Administrator\Documents\MPS Reti>arp -a
```

```
Interface: 192.168.1.139 --- 0x9
```

Internet Address	Physical Address	Type
192.168.1.1	00-1e-42-60-e2-a5	dynamic
192.168.1.255	ff-ff-ff-ff-ff-ff	static
224.0.0.22	01-00-5e-00-00-16	static
224.0.0.251	01-00-5e-00-00-fb	static
224.0.0.252	01-00-5e-00-00-fc	static
239.255.255.250	01-00-5e-7f-ff-fa	static
255.255.255.255	ff-ff-ff-ff-ff-ff	static

# ipconfig /all

Wireless LAN adapter Wi-Fi:

```
Connection-specific DNS Suffix . : lan
Description . . . . . : Intel(R) Wireless-AC 9260 160MHz
Physical Address. . . . . : A8-6D-AA-EB-12-2F
DHCP Enabled. . . . . : Yes
Autoconfiguration Enabled . . . . : Yes
IPv6 Address. . . . . : fd2c:e314:a697::d93(Preferred)
Lease Obtained. . . . . : lunedì 22 settembre 2025 09:44:48
Lease Expires . . . . . : lunedì 22 settembre 2025 21:44:47
IPv6 Address. . . . . : fd2c:e314:a697:0:2b29:54ed:1abc:7e52(Preferred)
Temporary IPv6 Address. . . . . : fd2c:e314:a697:0:f006:6867:bb93:553e(Preferred)
Link-local IPv6 Address . . . . . : fe80::2ae9:b2a9:8f2c:da4%9(Preferred)
IPv4 Address. . . . . : 192.168.1.139(Preferred)
Subnet Mask . . . . . : 255.255.255.0
Lease Obtained. . . . . : lunedì 22 settembre 2025 09:44:54
Lease Expires . . . . . : lunedì 22 settembre 2025 21:44:53
Default Gateway . . . . . : 192.168.1.1
DHCP Server . . . . . : 192.168.1.1
DHCPv6 IAID . . . . . : 94924202
DHCPv6 Client DUID. . . . . : 00-01-00-01-2D-41-61-7C-A8-6D-AA-EB-12-2F
DNS Servers . . . . . : fd2c:e314:a697::1
                        192.168.1.1
                        fd2c:e314:a697::1
NetBIOS over Tcpip. . . . . : Enabled
```



# **L1 Connettori a livello di network**

**Hub = concentratore ethernet, consente di unire piu macchine su un medesimo cavo**

**Repeater = estendono la lunghezza della rete**

**Modem = converte i segnali digitali in analogici (per esempio verso il provider telefonico)**

## L2 Collegamento dati

**Switch:** che ruota i collegamenti in base al mac address, ottimizza

**Bridge:** uniscono 2 segmenti di rete, in genere insieme agli switch

## **L3 Rete**

**Router: importantissimo perché consente di unire reti diverse (con diversi indirizzi) perché lavora tramite indirizzi IP**

**Switch layer 3: con qualche capacità di routing**

## **L4 applicazione**

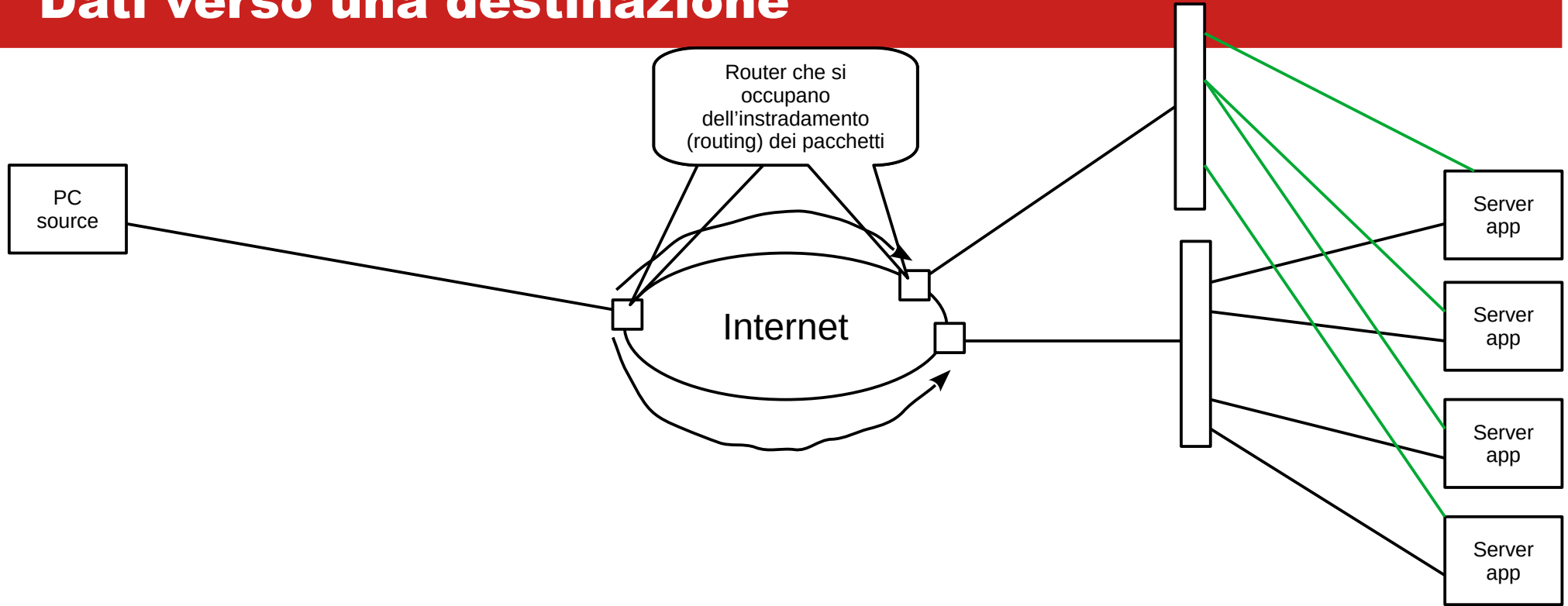
**Firewall: filtrano i pacchetti e costituiscono parte dei meccanismi di sicurezza della rete**

**Proxy, Reverse Proxy: lavorano a stretto contatto delle applicazioni e/o dei protocolli applicativi**

**Cache server: per la memorizzazione intermedia delle risorse**

**Gateway: general purpose**

# Dati verso una destinazione



# netstat -r (tabelle di routing)

```
C:\Users\Administrator\Documents\MPS Reti>netstat -r
=====
Interface List
17...a8 6d aa eb 12 30 .....Microsoft Wi-Fi Direct Virtual Adapter
 4...aa 6d aa eb 12 2f .....Microsoft Wi-Fi Direct Virtual Adapter #2
 9...a8 6d aa eb 12 2f .....Intel(R) Wireless-AC 9260 160MHz
 2...a8 6d aa eb 12 33 .....Bluetooth Device (Personal Area Network)
 1.....Software Loopback Interface 1
=====

IPv4 Route Table
=====
Active Routes:
Network Destination        Netmask          Gateway          Interface        Metric
0.0.0.0                    0.0.0.0          192.168.1.1      192.168.1.139    35
127.0.0.0                  255.0.0.0        On-link          127.0.0.1        331
127.0.0.1                  255.255.255.255  On-link          127.0.0.1        331
127.255.255.255            255.255.255.255  On-link          127.0.0.1        331
192.168.1.0                255.255.255.0    On-link          192.168.1.139    291
192.168.1.139              255.255.255.255  On-link          192.168.1.139    291
192.168.1.255              255.255.255.255  On-link          192.168.1.139    291
224.0.0.0                  240.0.0.0        On-link          127.0.0.1        331
224.0.0.0                  240.0.0.0        On-link          192.168.1.139    291
255.255.255.255            255.255.255.255  On-link          127.0.0.1        331
255.255.255.255            255.255.255.255  On-link          192.168.1.139    291
=====
Persistent Routes:
None
```

## ping <indirizzo | hostName>

```
C:\Users\Administrator\Documents\MPS Reti>ping www.example.com

Pinging a1422.dscr.akamai.net [2.23.231.94] with 32 bytes of data:
Reply from 2.23.231.94: bytes=32 time=28ms TTL=50
Reply from 2.23.231.94: bytes=32 time=138ms TTL=50
Reply from 2.23.231.94: bytes=32 time=89ms TTL=50
Reply from 2.23.231.94: bytes=32 time=25ms TTL=50

Ping statistics for 2.23.231.94:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 25ms, Maximum = 138ms, Average = 70ms
```

Ogni macchina vede sé stessa come 127.0.0.1 oppure localhost (local)

**L'Internet Protocol versione 4 (IPv4) è il protocollo di rete più diffuso, che definisce il sistema di indirizzamento utilizzato per identificare i dispositivi su internet. Funziona al livello di rete del modello OSI, consentendo l'instradamento dei pacchetti di dati dal mittente al destinatario attraverso diverse reti interconnesse.**

## **Struttura dell'Indirizzo**

**Un indirizzo IPv4 è un numero a 32 bit, solitamente rappresentato in notazione decimale puntata, ad esempio 192.168.1.1**

**Ogni indirizzo è suddiviso in due parti principali:**

Indirizzo di **rete**: identifica la rete a cui il dispositivo è collegato.

Indirizzo **host**: identifica il dispositivo specifico all'interno di quella rete.

**La separazione tra queste due parti è definita dalla subnet mask (maschera di sottorete, alias netmask), che indica quanti bit dell'indirizzo sono riservati alla rete e quanti all'host.**



## **L'Esaurimento degli Indirizzi IPv4 e il Futuro**

**Il numero limitato di indirizzi IPv4 ( $2^{32} \approx 4,3$  miliardi) è il suo difetto più grande. Con l'enorme crescita di dispositivi connessi a internet (IoT, smartphone, ecc.), gli indirizzi pubblici si sono esauriti.**

**Per risolvere questo problema, è stato introdotto IPv6, che utilizza un indirizzo a 128 bit, offrendo un numero quasi illimitato di indirizzi ( $2^{128}$ ) e nuove funzionalità. Nonostante ciò, IPv4 continua a essere ampiamente utilizzato (molti apparati devono essere sostituiti e molte reti riconfigurate), spesso in combinazione con IPv6.**

## **Indirizzi Pubblici e Privati**

**Non tutti i 4,3 miliardi di indirizzi IPv4 possibili sono utilizzabili su internet. Alcuni intervalli sono riservati per uso speciale, inclusi gli indirizzi privati, che non sono visibili pubblicamente su internet e sono usati all'interno di reti locali.**

**Classe A privata: 10.0.0.0 a 10.255.255.255**

**Classe B privata: 172.16.0.0 a 172.31.255.255**

**Classe C privata: 192.168.0.0 a 192.168.255.255**

**Per permettere a un dispositivo con un indirizzo privato di comunicare con internet, si usa la Network Address Translation (NAT), che traduce l'indirizzo privato in un indirizzo pubblico unico (La NAT è implementata dai router).**

## Indirizzo di Loopback (Classe A)

**L'indirizzo 127.0.0.1 (localhost, local) è il più noto di questa categoria, ma l'intero blocco 127.0.0.0/8 è riservato. Viene usato per il loopback, ovvero per permettere a un dispositivo di inviare pacchetti a sé stesso. Questo è fondamentale per testare i servizi di rete e le applicazioni senza che il traffico lasci il computer.**

**L'indirizzo di loopback è l'indirizzo tramite il quale un computer vede se stesso.**

## Indirizzi di Rete e Broadcast

**Indirizzo di Rete (.0):** il primo indirizzo in un blocco di rete identifica la rete stessa. Non viene assegnato a un singolo dispositivo. Ad esempio, in una rete 192.168.1.0/24, l'indirizzo 192.168.1.0 identifica l'intera rete.

**Indirizzo di Broadcast (.255):** l'ultimo indirizzo in un blocco di rete è l'indirizzo di broadcast. I pacchetti inviati a questo indirizzo vengono ricevuti da tutti gli host presenti in quella sottorete. Ad esempio, 192.168.1.255 invia a tutti i dispositivi nella rete 192.168.1.0/24.

## Indirizzi Link-Local (APIPA)

**L'intervallo 169.254.0.0/16 è usato per l'Automatic Private IP Addressing (APIPA). Un dispositivo si auto-assegna un indirizzo in questo intervallo quando non riesce a ottenere un indirizzo IP da un server DHCP.**

**Questo permette ai dispositivi di una rete locale di comunicare tra loro anche in assenza di un server DHCP, i dispositivi, però, non possono accedere ad Internet.**

## Indirizzi di Multicast e Sperimentali

**Multicast (224.0.0.0/4):** un blocco riservato per il multicast, che consente di inviare dati a un gruppo specifico di destinatari contemporaneamente. È usato, per esempio, per lo streaming video o le videoconferenze.

**Sperimentali (240.0.0.0/4):** questo blocco non è assegnato a scopi specifici ed è riservato per ricerca e sviluppo. I pacchetti con questi indirizzi vengono generalmente bloccati dai router pubblici.

# DHCP

**Il DHCP (Dynamic Host Configuration Protocol) è un protocollo di rete che assegna automaticamente gli indirizzi IP, e altre impostazioni di rete, ai dispositivi (come computer, smartphone e stampanti) che si connettono a una rete.**

**Questo processo elimina la necessità di configurare manualmente ogni dispositivo, semplificando notevolmente la gestione della rete.**

**Il DHCP opera secondo un modello client-server, dove un server DHCP gestisce un pool di indirizzi IP disponibili e un client DHCP (il dispositivo che si connette) richiede un indirizzo a questo pool.**

# DHCP

**Il funzionamento del DHCP si basa su uno scambio di messaggi in quattro fasi, noto come processo DORA (Discover, Offer, Request, Acknowledge):**

**Discover:** Un dispositivo appena connesso (il client DHCP) invia un messaggio broadcast **DHCPDISCOVER** per trovare un server DHCP sulla rete. Il messaggio viene inviato a tutti i dispositivi della rete perché il client non conosce ancora l'indirizzo del server.

**Offer:** Uno o più server DHCP sulla rete rispondono al messaggio con un'offerta **DHCPOFFER**, proponendo un indirizzo IP disponibile dal loro pool.

**Request:** Il client sceglie un'offerta (di solito la prima che riceve) e invia un messaggio **DHCPREQUEST** a tutti i server, confermando la sua scelta e richiedendo l'assegnazione dell'indirizzo offerto. Questo avvisa anche gli altri server che la loro offerta non è stata accettata.

**Acknowledge:** Il server che ha ricevuto la richiesta invia un messaggio **DHCPACK** di conferma. Questo messaggio contiene l'indirizzo IP assegnato, la maschera di sottorete, il gateway predefinito, gli indirizzi dei server DNS e la durata del "lease" (il tempo per cui l'indirizzo è valido).



**Il DHCP lease è il periodo di tempo per cui un client può utilizzare l'indirizzo IP assegnato.**

**Prima della scadenza del lease, il client tenta di rinnovarlo inviando un altro messaggio di richiesta al server.**

**Se il server accetta, il lease viene esteso, altrimenti il client deve avviare di nuovo il processo DORA per ottenere un nuovo indirizzo.**

## **Vantaggi del DHCP**

**Efficienza:** automatizza l'assegnazione degli indirizzi IP, riducendo l'onere amministrativo, specialmente in reti di grandi dimensioni con molti dispositivi.

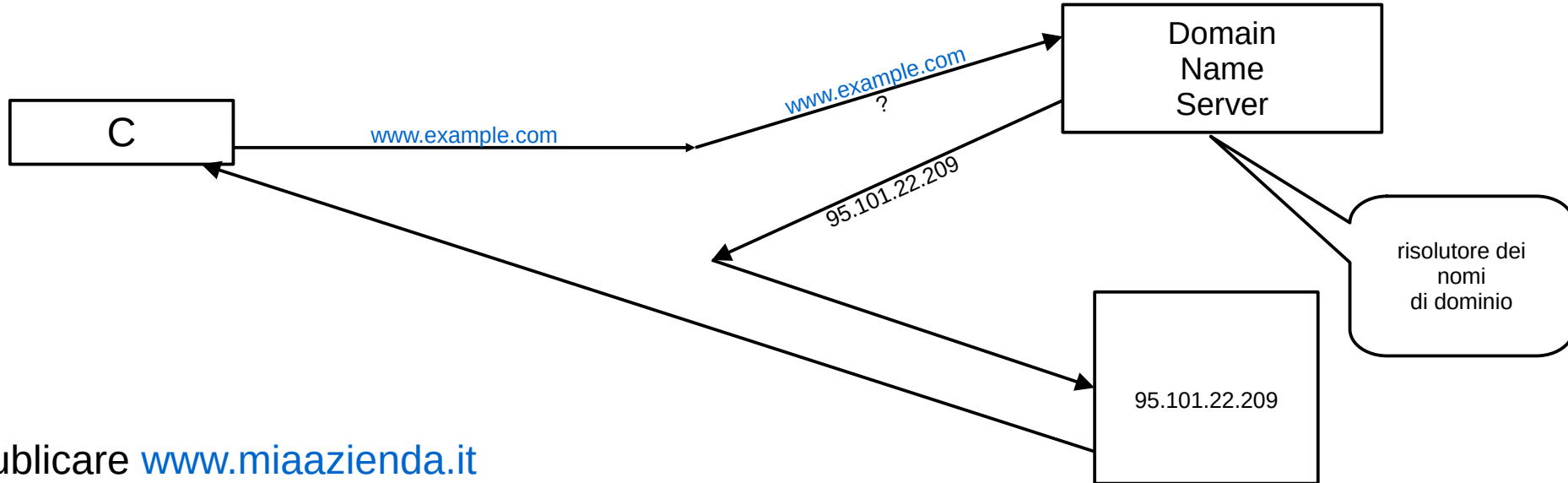
**Riduzione degli errori:** elimina gli errori di configurazione manuale, come l'assegnazione accidentale dello stesso indirizzo a due dispositivi.

**Gestione centralizzata:** permette agli amministratori di controllare le configurazioni di rete da un unico punto.

**Flessibilità:** facilita la gestione di dispositivi mobili e temporanei, come quelli che si connettono a una rete Wi-Fi pubblica, garantendo una connettività fluida.

# DNS è risolutore dei nomi di rete

## traduce da nome simbolico ad indirizzo ip



devo pubblicare [www.miaazienda.it](http://www.miaazienda.it)

**miaazienda.it** è il domain

**www** un server che appartiene al mio dominio quindi [www.miaazienda.it](http://www.miaazienda.it) ==> ip\_address

**mail** server di posta elettronica mail.miaazienda.it ==> ip\_address

# DNS

devo pubblicare [www.miaazienda.it](http://www.miaazienda.it)

**miazienda.it** è il domain

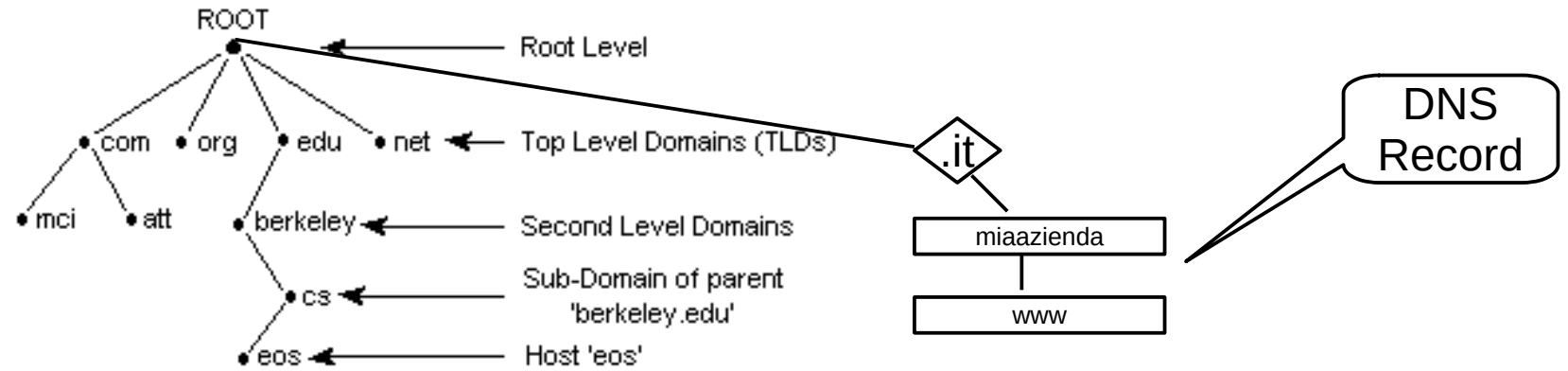
**www** un server che appartiene al mio dominio quindi [www.miazienda.it](http://www.miazienda.it) ==> ip\_address

**mail** server di posta elettronica mail.miaazienda.it ==> ip\_address



# DNS

DNS Hierarchy

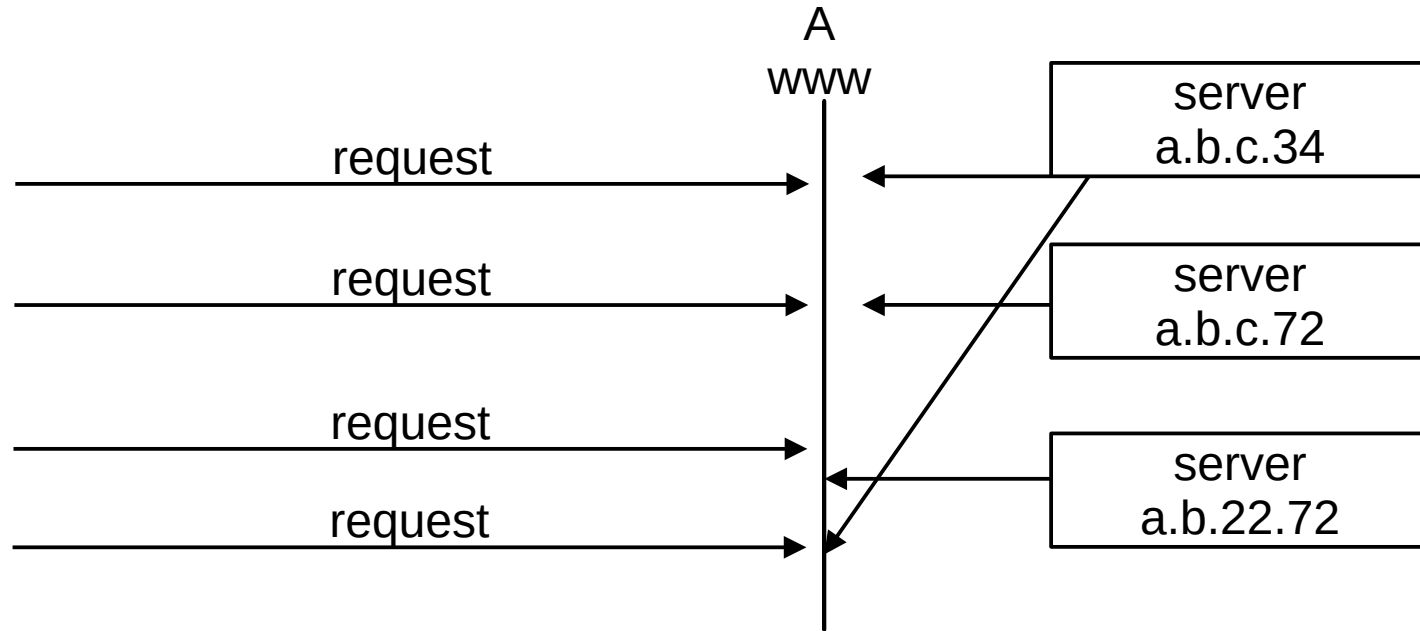


# nslookup

```
Server: [1.1.1.1]
Address: 1.1.1.1

Non-authoritative answer:
Name:      a1422.dscr.akamai.net
Addresses: 2a02:26f0:8d00:3c::5f64:b584
           2a02:26f0:8d00:3c::5f64:b58b
           23.55.104.151
           23.55.104.155
Aliases:   www.example.com
           www.example.com-v4.edgesuite.net
```

# DNS – Round Robin (load balancer)



# DNS

**Il DNS (Domain Name System) funziona come la rubrica telefonica per Internet.**

**Invece di dover memorizzare indirizzi IP numerici complessi, il DNS traduce i nomi di dominio facili da ricordare (come [www.google.com](http://www.google.com)) negli indirizzi IP numerici che i computer utilizzano per localizzare i siti web.**



**Il processo per tradurre un nome di dominio in un indirizzo IP, chiamato "risoluzione del nome" (Name Resolution), coinvolge diversi tipi di server DNS in un processo gerarchico e iterativo.**

**Tutto parte quando l'utente avvia la query: ad esempio digitando un nome di dominio (es. `www.example.com`) nel browser, il computer (il DNS client) invia una richiesta a un server DNS ricorsivo. Questo server è solitamente fornito dall' ISP (Internet Service Provider) o da un servizio pubblico (es: Cloudflare, Google, ...)**

# DNS

**Il server ricorsivo interroga la gerarchia DNS: se il server ricorsivo non ha l'indirizzo IP nella sua cache (non authoritative answer), inizia un processo di interrogazione a cascata:**

**Root Server (.):** Il server ricorsivo interroga un Root Server. Questi sono i 13 server principali che si trovano in cima alla gerarchia DNS. Il Root Server non conosce l'indirizzo IP del sito, ma sa qual è il server TLD (Top-Level Domain) responsabile per i domini .com.

**TLD (Top-Level Domain) Server:** Il server ricorsivo interroga il server TLD per .com. Questo server sa quali server sono responsabili per i domini che terminano in .com. In questo caso, indicherà al server ricorsivo qual è il server autoritativo per example.com.

**Server Autoritativo:** Il server ricorsivo interroga il server autoritativo per example.com. Questo è il server che contiene i record DNS definitivi per quel dominio specifico. A questo punto, il server autoritativo risponde con l'indirizzo IP richiesto.

# DNS

**Il DNS, infine, risponde al browser: il server DNS ricorsivo invia l'indirizzo IP al computer. che lo memorizza nella sua cache per un certo periodo (definito dal Time to Live - TTL del record DNS). L'indirizzo IP ottenuto viene quindi utilizzato per connettersi direttamente (come numero) al server web del sito. L'intero processo avviene in una frazione di secondo (se non vi sono problemi), permettendo all'utente di accedere al sito web senza percepire ritardi.**

**Ecco i più utilizzati primary e secondary DNS servers:**

**Cloudflare DNS (one.one.one.one)**

**IPv4: 1.1.1.1 e 1.0.0.1**

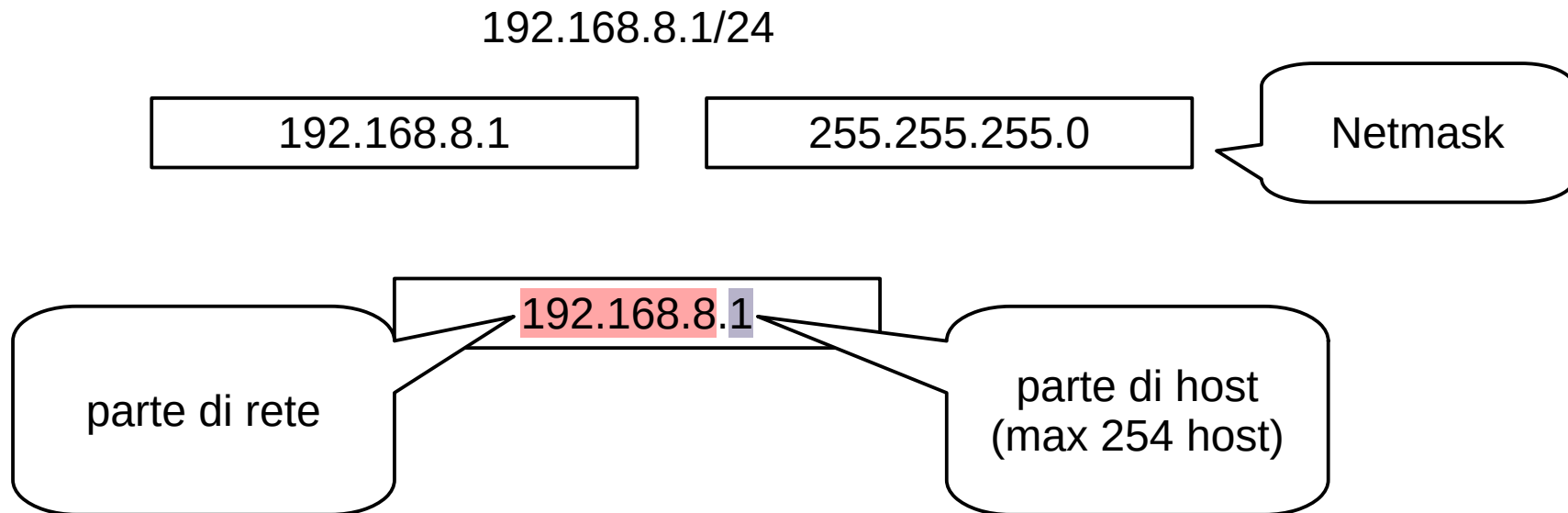
**IPv6: 2606:4700:4700::1111 e 2606:4700:4700::1001**

**Google DNS**

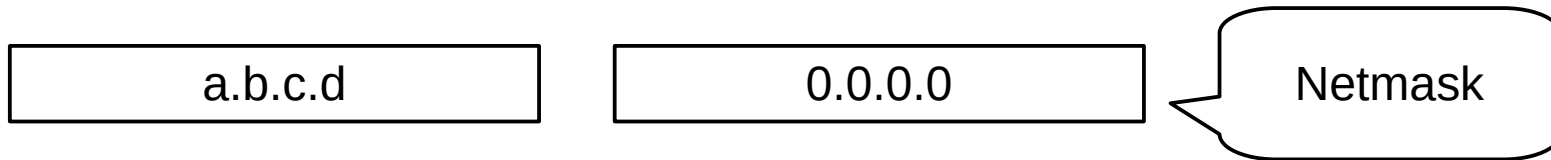
**IPv4: 8.8.8.8 e 8.8.4.4**

**IPv6: 2001:4860:4860::8888 e 2001:4860:4860::8844**

# subnet



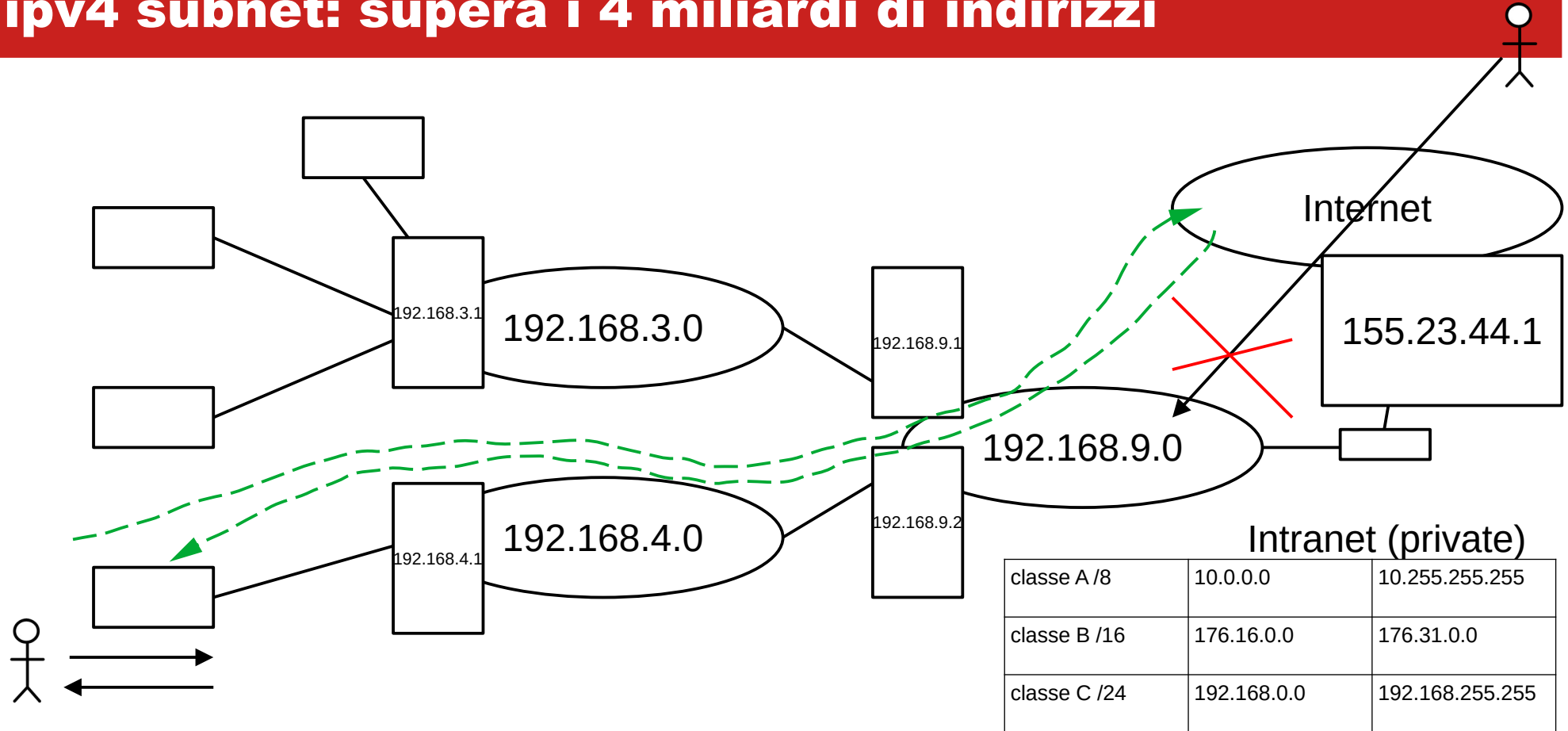
## subnet IPV4



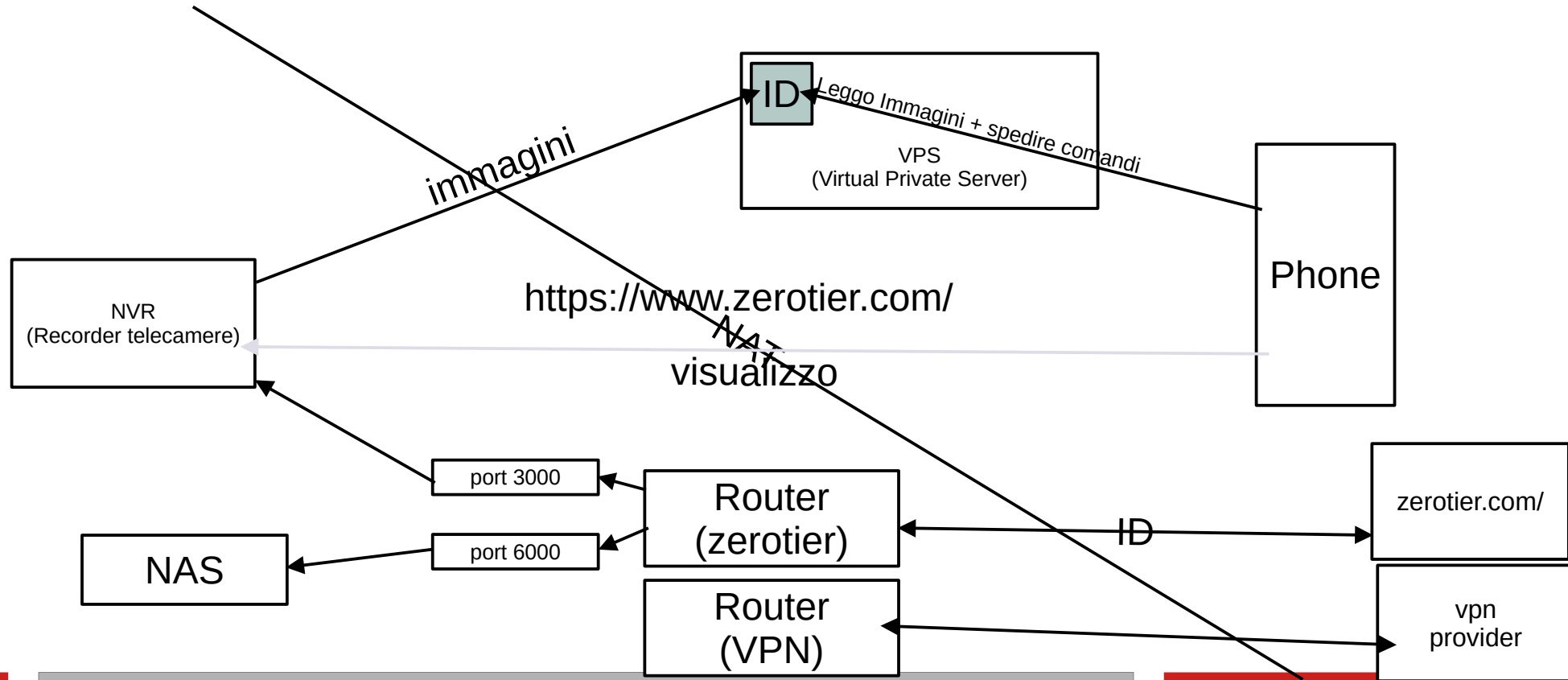
La parte di rete non esiste, tutti sono hosts  
in totale 4.294.967.294

da IPV4 a IPV6 passano 340 trilioni di trilioni di trilioni di indirizzi

# ipv4 subnet: supera i 4 miliardi di indirizzi

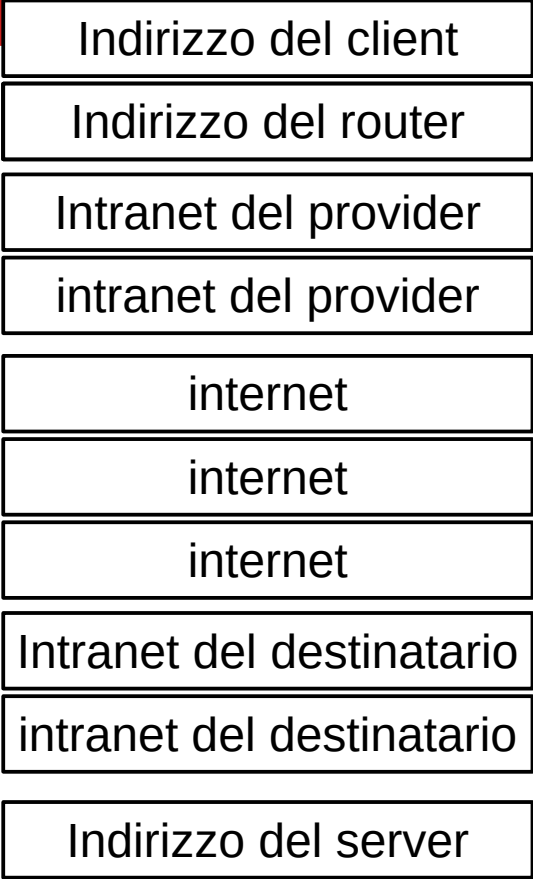


# Telecamere/Allarmi/Caldaie/... da casa





# traceroute



classe A	10.0.0.0	10.255.255.255
classe B	176.16.0.0	176.31.0.0
classe C	192.168.0.0	192.168.255.255

**IPv6 (Internet Protocol version 6) è la versione più recente del protocollo Internet, creata per affrontare l'esaurimento degli indirizzi IPv4 e introdurre miglioramenti nella gestione della rete.**

**Un indirizzo IPv6 è un numero di 128 bit, ciò si traduce in un numero di indirizzi disponibili pari a  $2^{128}$  (circa 340 trilioni di trilioni di trilioni).**

**l'indirizzo ipv6 viene scritto tramite la notazione esadecimale, suddiviso in otto blocchi da 16 bit, separati da due punti.**

**Esempio:           2001:0db8:85a3:0000:0000:8a2f:0370:7334**

**semplificazione 1 ==> 2001:db8:85a3:0000:0000:8a2f:370:7334**

**semplificazione 2 ==> 2001:db8:85a3:::8a2f:370:7334**

**Per rendere la lettura più semplice, ci sono regole di abbreviazione:**

Omettere gli zero iniziali: 0db8 diventa db8

Sostituire blocchi di zeri contigui con :: 2001:db8:85a3::8a2e:370:7334 (questa abbreviazione può essere usata solo una volta per indirizzo).

**L'IPv6 non è solo una soluzione all'esaurimento degli indirizzi, ma introduce anche diversi miglioramenti:**

**Eliminazione del NAT:** Poiché ogni dispositivo può avere un indirizzo pubblico unico, il Network Address Translation (NAT) non è più necessario, semplificando la connettività end-to-end.

**Configurazione automatica degli indirizzi (SLAAC):** I dispositivi possono generare autonomamente il proprio indirizzo IP senza la necessità di un server DHCP, basandosi sul prefisso della rete e sull'ID dell'interfaccia.

**Header semplificato:** L'header del pacchetto IPv6 è più semplice e ha una dimensione fissa (40 byte). Ciò consente ai router di processare i pacchetti più velocemente, migliorando le prestazioni della rete.

**Sicurezza integrata:** IPv6 include di serie il protocollo IPsec, che fornisce funzionalità di crittografia e autenticazione a livello di rete, rendendo le comunicazioni più sicure.

**Nessun broadcast:** IPv6 sostituisce il broadcast con il multicast, che invia pacchetti solo a un gruppo specifico di destinatari, riducendo il traffico di rete inutile.

**In IPv6, gli indirizzi speciali sono intervalli di indirizzi con scopi specifici, analoghi a quelli di IPv4 ma con una struttura e una nomenclatura diverse. Non sono destinati all'uso generico per i dispositivi su Internet.**



## **Indirizzi di Loopback**

**L'indirizzo di loopback in IPv6 è ::1 (equivalente a 127.0.0.1 in IPv4). Viene utilizzato da un host per inviare pacchetti a sé stesso, utile per testare la connettività di rete e i servizi senza che il traffico lasci il dispositivo.**

## Indirizzi Link-Local (FE80::/10)

**Gli indirizzi link-local sono utilizzati solo per la comunicazione all'interno di un singolo segmento di rete (link), non sono instradabili al di fuori di esso. Iniziano sempre con fe80:: e sono generati automaticamente da un'interfaccia di rete senza bisogno di un server DHCP.**

**Sono usati per scopi come la scoperta di router o l'autoconfigurazione degli indirizzi.**

## Indirizzi Unicast Unici Locali (FC00::/7)

**Gli Unique Local Addresses (ULA) sono simili agli indirizzi privati di IPv4 (come 192.168.1.0). Vengono utilizzati all'interno di reti private e non sono instradabili su Internet. Hanno un prefisso che inizia con fc00::/7 e sono progettati per essere unici all'interno di una rete privata.**

## Indirizzi di Multicast (FF00::/8)

**Gli indirizzi multicast sono usati per inviare un pacchetto a un gruppo di destinatari contemporaneamente. In IPv6, il broadcast è stato eliminato e sostituito interamente dal multicast. Ogni indirizzo multicast inizia con ff00::/8. Esistono indirizzi multicast specifici, come ff02::1 che rappresenta tutti i nodi di un link, e ff02::2 che rappresenta tutti i router di un link.**

## Indirizzi Anycast

**Gli indirizzi anycast sono una novità di IPv6. Un indirizzo anycast viene assegnato a più interfacce, tipicamente appartenenti a nodi diversi. Quando un pacchetto viene inviato a un indirizzo anycast, viene consegnato all'interfaccia più vicina (in termini di routing). Questa tecnica è usata principalmente per la distribuzione del carico (load balancing) e per migliorare l'affidabilità di servizi critici come il DNS.**

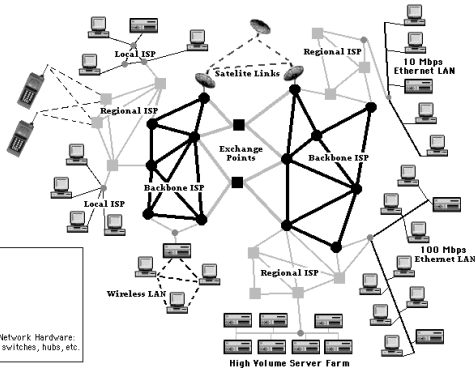
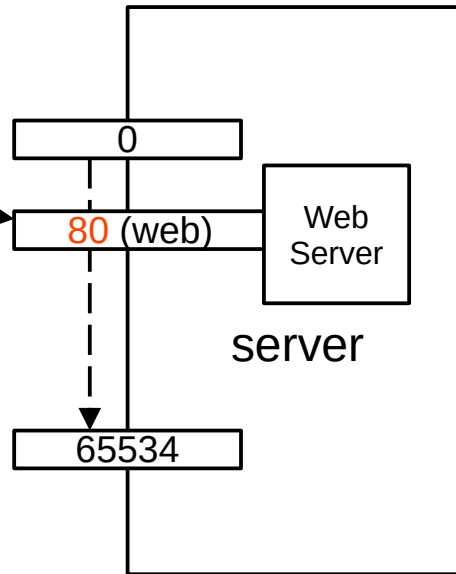


# Servizi IP ( TCP / UDP )

il TCP orientato alla connessione e i pacchetti vengono ricevuti nello stesso ordine in cui sono stati trasmessi

l'UDP non è orientato alla connessione e i pacchetti vengono ricevuti in ordine casuale (non è garantito l'ordine di ricezione)

*client http request*



## Network Protocol Stack

Focus of this article

**Application Layer**  
HTTP, FTP, SMTP, DNS

What users interact with

**Transport Layer**  
TCP, UDP

How data is packaged

**Internet Layer**  
IP, ICMP, ARP

How data is routed

**Link Layer**  
Ethernet, Wi-Fi, PPP

Physical connections

# TCP

## TCP (Transmission Control Protocol)

**Il TCP è un protocollo orientato alla connessione, il che significa che stabilisce un "canale di comunicazione" affidabile tra due dispositivi prima di inviare i dati (come fare una telefonata: prima si compone il numero e si aspetta che l'altra persona risponda, solo allora si inizia a parlare).**

# TCP caratteristiche principali

**Affidabilità:** Garantisce che i dati arrivino a destinazione in modo completo e nell'ordine corretto. Se un pacchetto di dati si perde, viene richiesto e inviato nuovamente.

**Controllo del flusso:** Regola la velocità di trasmissione per evitare di sovraccaricare il ricevitore.

**Conferma di ricezione:** Per ogni pacchetto inviato, il destinatario manda una conferma di ricezione (ACK).

**Ambiti di utilizzo:** Viene usato per applicazioni dove l'integrità dei dati è cruciale, come la navigazione web (HTTP/HTTPS), il trasferimento di file (FTP) e l'invio di email (SMTP).

# UDP

## UDP (User Datagram Protocol)

**L'UDP è un protocollo senza connessione e molto più semplice e veloce del TCP. È come spedire una cartolina: la si scrive e la si invia, ma non si ha la certezza che arrivi a destinazione né una conferma di ricezione. La sua priorità è la velocità, a discapito dell'affidabilità.**

# UDP caratteristiche principali

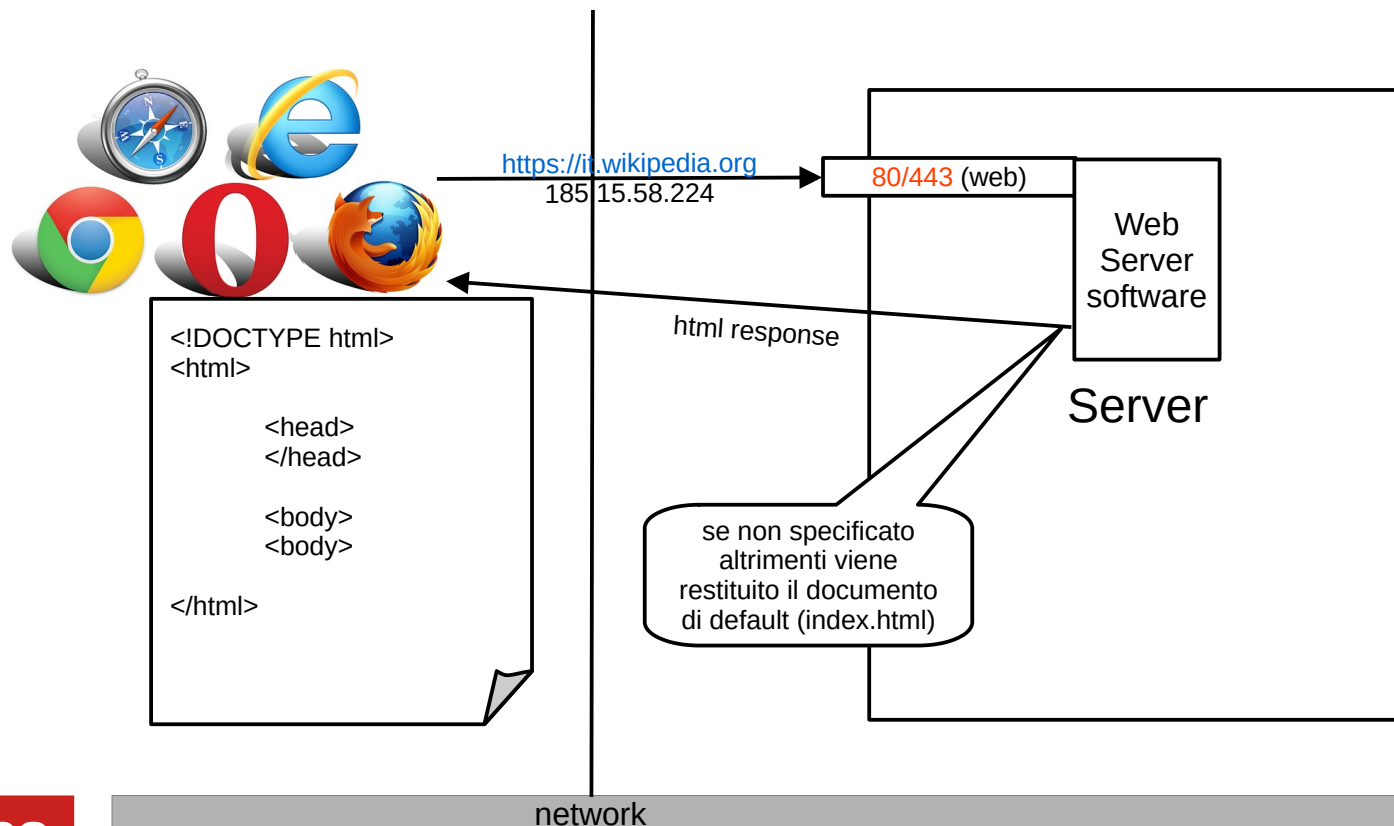
**Velocità:** Non essendoci il "dialogo" iniziale e le conferme di ricezione, la trasmissione è molto più rapida.

**Nessuna garanzia:** Non assicura che i dati arrivino a destinazione o che lo facciano nell'ordine corretto, i pacchetti possono perdersi senza che il mittente ne venga a conoscenza.

**Overhead ridotto:** Essendo un protocollo più "leggero", consuma meno risorse.

**Ambiti di utilizzo:** È ideale per applicazioni dove la velocità è più importante dell'affidabilità, come il gaming online, lo streaming video e le chiamate VoIP, dove la perdita di qualche pacchetto non compromette l'esperienza utente in modo significativo.

# HTTP



Tutti i **codici di stato** della risposta HTTP sono separati in cinque classi o categorie. La prima cifra del codice di stato definisce la classe di risposta, mentre le ultime due cifre non hanno alcun ruolo di classificazione o categorizzazione.

1xx informational response – the request was received, continuing process

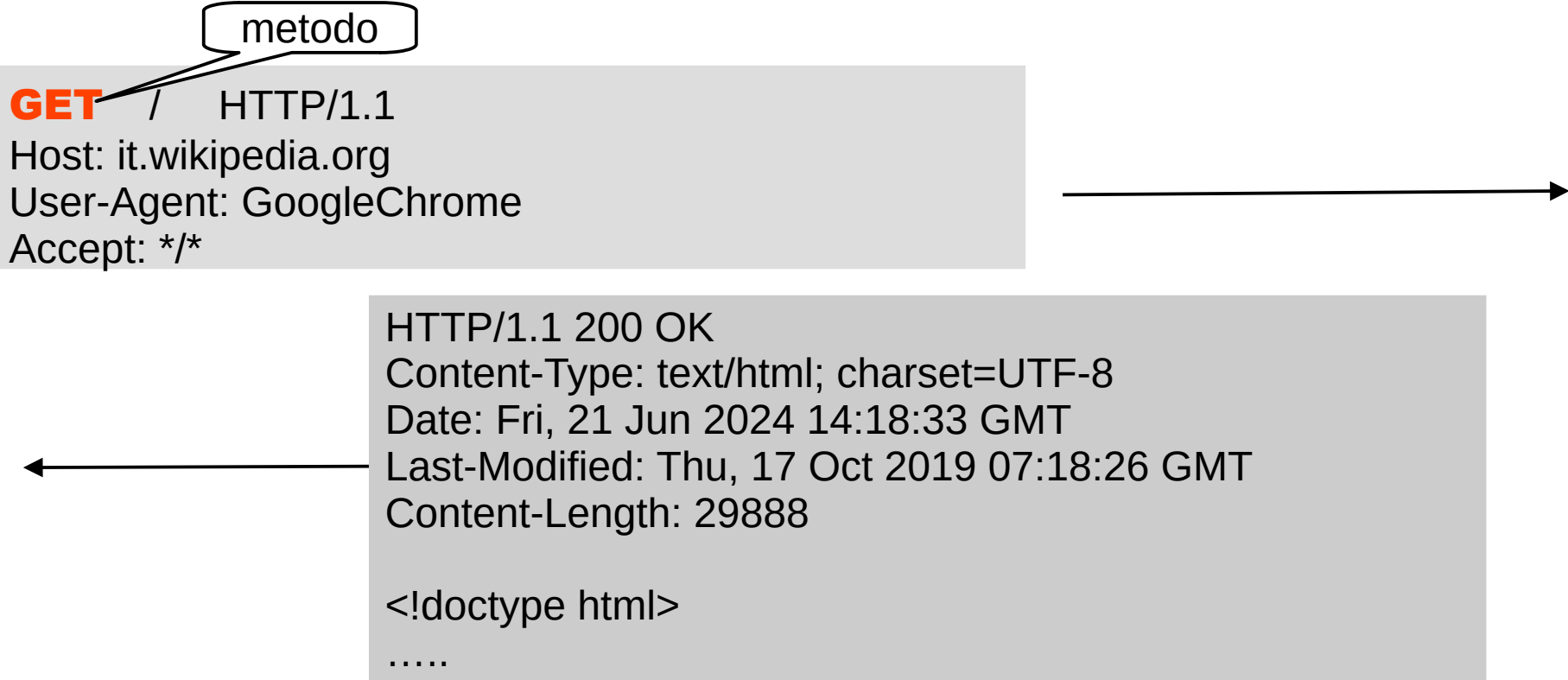
2xx successful – the request was successfully received, understood, and accepted

3xx redirection – further action needs to be taken in order to complete the request

# HTTP Request

metodo

```
GET / HTTP/1.1
Host: it.wikipedia.org
User-Agent: GoogleChrome
Accept: */*
```



```
HTTP/1.1 200 OK
Content-Type: text/html; charset=UTF-8
Date: Fri, 21 Jun 2024 14:18:33 GMT
Last-Modified: Thu, 17 Oct 2019 07:18:26 GMT
Content-Length: 29888
```

```
<!doctype html>
```

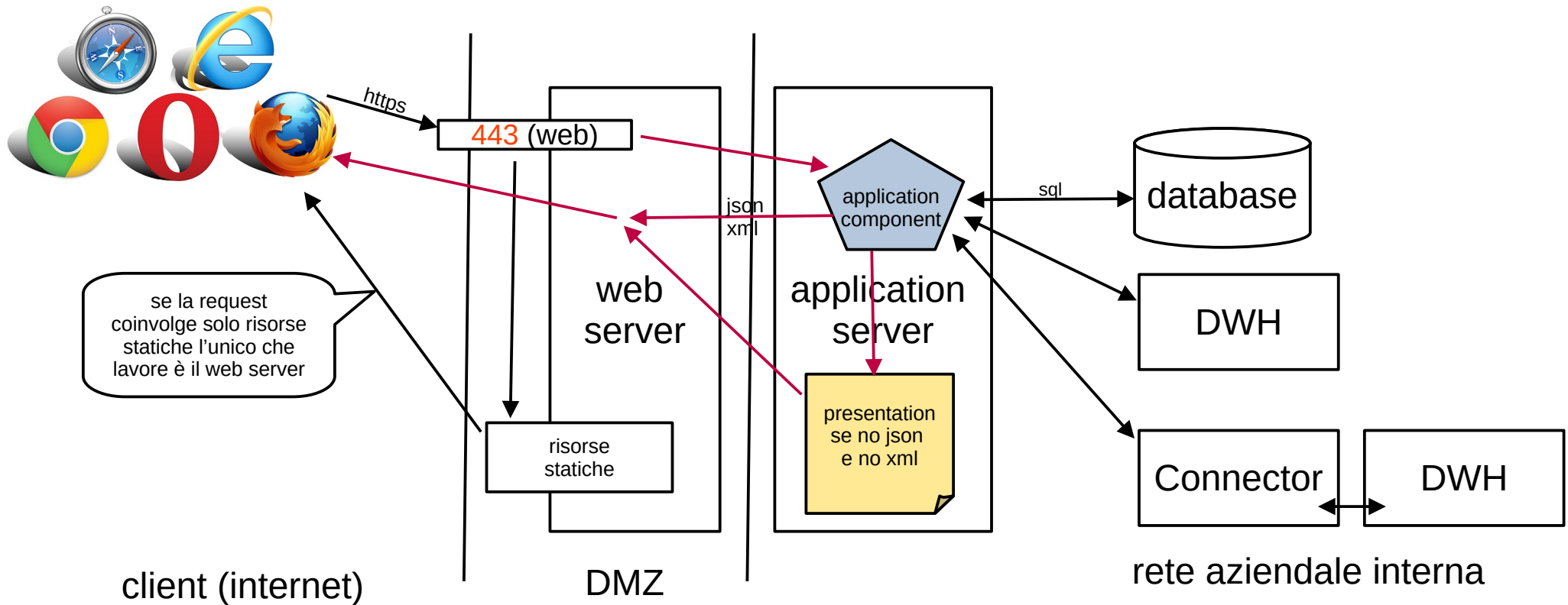
```
.....
```

# HTTP methods

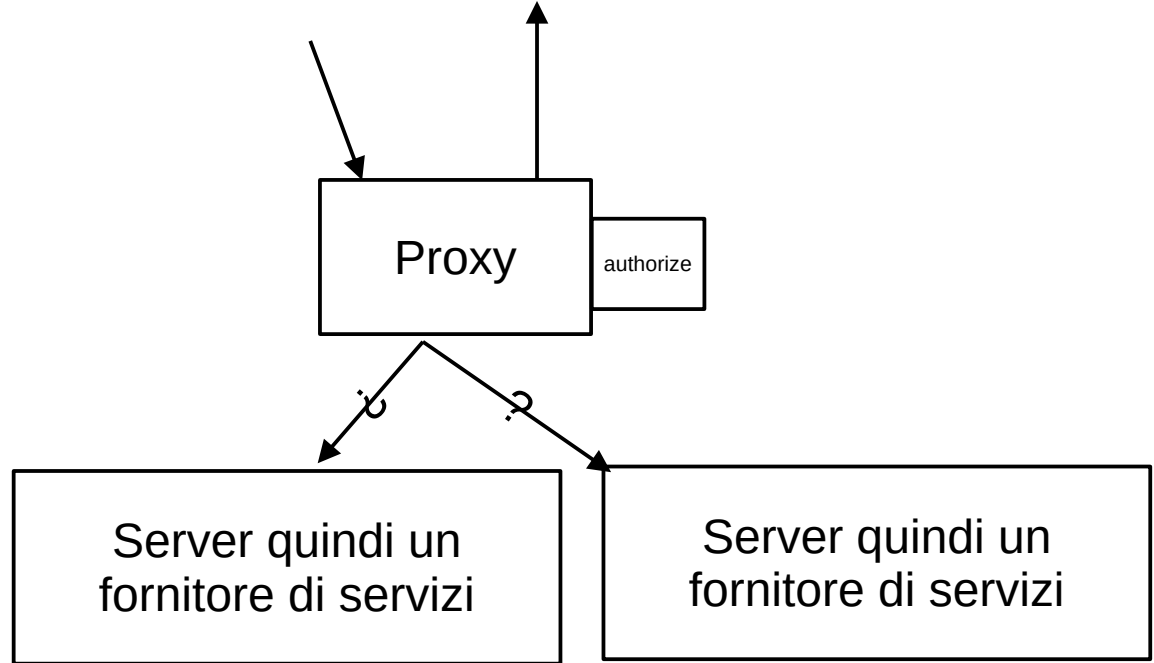
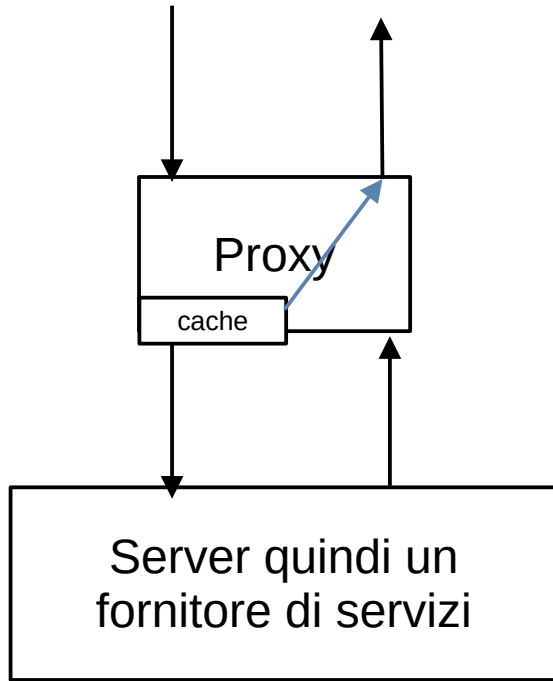
Metodo	Scopo	SQL	CRUD
GET	Leggi risorse	SELECT	Read
POST	Creo risorse	INSERT	Create
PUT (PATCH)	Modifica	UPDATE	Update
DELETE	Cancella	DELETE	Delete



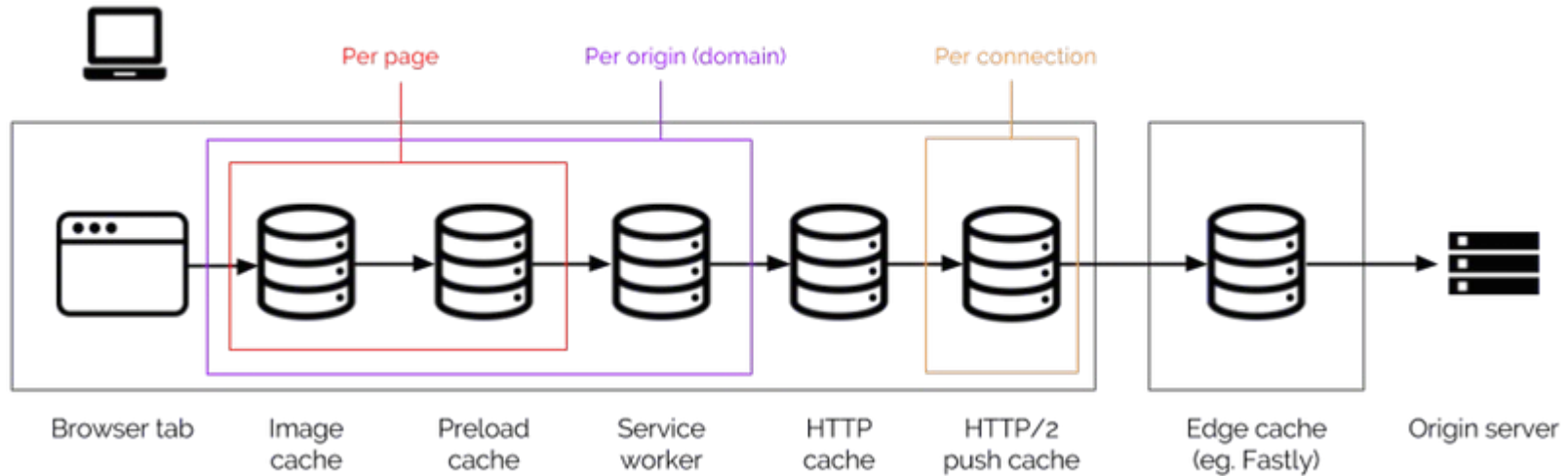
# Web Application – three tier architecture



# Proxy e Reverse Proxy



# Cache



# HTTP

**L'HTTP (Hypertext Transfer Protocol) è il protocollo alla base del World Wide Web. Definisce le regole per lo scambio di file (come pagine web, immagini e video) tra un client (il tuo browser) e un server (il computer che ospita il sito web).**

**È un protocollo stateless, ovvero ogni richiesta client viene gestita dal server come una transazione indipendente, senza ricordare le richieste precedenti (per renderlo stateful vengono usati i cookie e session)**

# HTTP

**Una transazione HTTP segue un modello di request-response e si articola in tre fasi principali:**

**1) Connessione: Il client stabilisce una connessione TCP (Transmission Control Protocol) con il server sulla porta 80 (o 443 per HTTPS).**

**2) Request HTTP: Il client invia una richiesta al server che include:**

Metodo HTTP: L'azione che il client vuole eseguire (GET, POST, PUT, DELETE).

URL: L'indirizzo della risorsa richiesta.

Header: Informazioni aggiuntive (es. tipo di browser, cookie, ecc.).

Body (opzionale): Dati da inviare al server (es. un modulo compilato).

## 3) Response HTTP: Il server elabora la richiesta e invia una risposta al client. Questa risposta include:

Codice di stato ([Status Code](#)): Un codice numerico che indica l'esito della richiesta (es. 200 OK, 404 Not Found, 500 Internal Server Error).

Header: Informazioni aggiuntive sulla risposta (es. tipo di contenuto, data).

Body: I dati richiesti (es. il codice HTML della pagina, un'immagine).

# HTTP vs. HTTPS

**HTTPS (Hypertext Transfer Protocol Secure) è la versione sicura di HTTP. Aggiunge un livello di crittografia (tramite TLS/SSL) per proteggere i dati scambiati tra client e server, rendendoli illeggibili a terzi. È fondamentale per la sicurezza di transazioni sensibili come pagamenti online o login.**

**Il protocollo SSL (Secure Sockets Layer), e il suo successore TLS (Transport Layer Security), è un protocollo crittografico che assicura una comunicazione sicura tra un server e un client. Il suo obiettivo è proteggere la riservatezza e l'integrità dei dati scambiati, garantendo che non possano essere intercettati, letti o modificati da terze parti.**

# Come funziona una transazione SSL/TLS

**Il processo di negoziazione e stabilimento di una connessione sicura è chiamato handshake SSL/TLS. Avviene in più fasi:**

- 1) Client Hello:** Il browser (client) invia un messaggio "Client Hello" al server. Questo messaggio contiene la versione di TLS, un numero casuale, e l'elenco degli algoritmi di cifratura che supporta.
- 2) Server Hello:** Il server risponde con un messaggio "Server Hello", selezionando il protocollo e l'algoritmo di cifratura più forte che entrambi supportano. Invia anche un numero casuale ed il proprio certificato SSL.
- 3) Verifica del Certificato:** Il browser verifica il certificato del server per garantirne l'autenticità. Il certificato contiene la chiave pubblica del server e informazioni sulla sua identità. Se la verifica fallisce (es. il certificato è scaduto o non è emesso da un'autorità di certificazione riconosciuta), il browser avvisa l'utente dell'anomalia.



## Come funziona una transazione SSL/TLS

**4) Scambio di Chiavi:** Il browser utilizza la chiave pubblica del server (presente nel certificato) per crittografare un'altra chiave casuale, chiamata chiave di sessione, e la invia al server. Solo il server, con la sua chiave privata, può decifrare questa chiave di sessione.

**5) Comunicazione Crittografata:** A questo punto, sia il client che il server hanno la stessa chiave di sessione segreta. Tutte le comunicazioni successive saranno crittografate e decifrate usando questa chiave, garantendo che i dati siano protetti per tutta la durata della sessione



# POP, IMAP e SMTP

**I protocolli POP, IMAP e SMTP sono i protocolli standard usati dai client di posta elettronica (come Outlook, Thunderbird, l'app Mail dello smartphone) per inviare e ricevere email da un server di posta.**

# POP, IMAP e SMTP

## POP (Post Office Protocol)

**Il protocollo POP è come una cassetta della posta: scarica tutte le email dal server sul dispositivo e, per impostazione predefinita, le elimina dal server.**

Vantaggi: Le email sono disponibili offline e non occupano spazio sul server.

Svantaggi: Le email sono legate ad un solo dispositivo. Se l'email è sul computer, non sarà accessibile dal telefono, a meno che non si configuri il client per non eliminare le email dal server (opzione "Lascia una copia sul server") che però invalida il vantaggio di cui sopra.

# POP, IMAP e SMTP

## IMAP (Internet Message Access Protocol)

**Il protocollo IMAP è come una libreria: mantiene le email sul server e le sincronizza su tutti i tuoi dispositivi.**

Vantaggi: accesso alla stessa casella di posta, con gli stessi messaggi, le stesse cartelle e lo stesso stato di lettura, da qualsiasi dispositivo.

Svantaggi: Richiede una connessione a Internet costante per accedere alle email e occupa spazio sul server.

# POP, IMAP e SMTP

## **SMTP (Simple Mail Transfer Protocol)**

**Il protocollo SMTP è l'unico dei tre a gestire l'invio delle email, è il protocollo che il client di posta usa per comunicare con il server di posta in uscita per spedire un messaggio.**

**SMTP si occupa di trasferire l'email dal client verso il server del destinatario, fungendo da "corriere" della posta elettronica.**

**SMTP serve solo per l'invio, non gestisce la ricezione delle email.**

