



netstudent

# Corso GNU/Linux

Networking – dicembre 2014



*Emmanuel <err@netstudent.polito.it>*

<http://netstudent.polito.it>

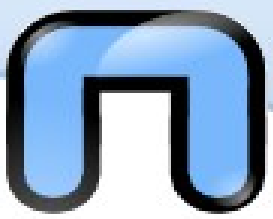


# La rete IP

Comandi

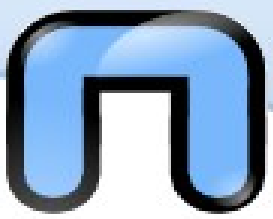
Servizi di rete





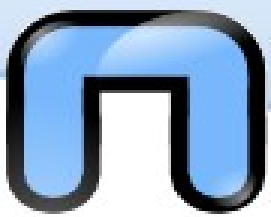
## Comunicazione - 1

- La trasmissione di informazioni in modo automatico tra diversi computer e` da tempo stata una necessità
- Inizialmente spesso si trattava di trasmettere a un computer potente (mainframe) alcune operazioni da parte di client limitati (detti terminali)
- Poi è diventata un comodo strumento per ottimizzare le risorse in ambienti medio-grandi, offrendo i servizi più svariati
- Per esempio stampante di rete, memorizzazione di dati unificata, comunicazione automatizzata



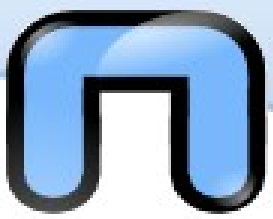
## Comunicazione - 2

- Una “rete” di computer e` un insieme di due o più “host” o “nodi” che possono trasmettersi dati con modalità fra loro comprensibili (“lingua”)
- Per lungo tempo ogni costruttore di computer aveva il suo linguaggio di comunicazione
- Protocollo = lingua + modalità di accesso
- E` una scelta vincente quella di utilizzare protocolli standard (pubblicamente definiti) per la comunicazione



## Classificazione

- La rete può avere una topologia:
  - A stella / Ad anello / A bus / ...
- Una rete può estendersi:
  - PAN Personal Area Network - personale
  - LAN Local Area Network - locale
  - WAN Wide Area Network – geografica
- Tipicamente la comunicazione può essere:
  - Uno a uno / Uno a molti / ...



## Lo stack di rete

- Lo “stack” (pila) di rete è una rappresentazione logico-funzionale del linguaggio di comunicazione
  - Livello da quello fisico a quello applicativo che permettono di semplificare la gestione dal punto di vista applicativo e costruttivo
- Nel tempo si sono diffusi e sono stati standardizzati diversi modelli e stack

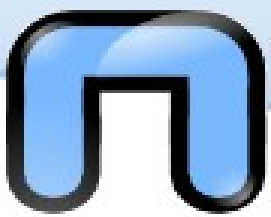


# La rete IP

## I livelli

- Livello 5, 6, 7 – sessione, presentazione, applicazione
- Livello 4 – trasporto TCP, UDP – messaggio, errori
- Livello 3 – indirizzo IP – routing, gateway, etc
- Livello 2 – datalink GENERICO – LLC, MAC
- Livello 1 – fisico GENERICO – 1, 0, tempo

E` un modello molto conveniente e permette di capire chi fa` cosa in modo chiaro

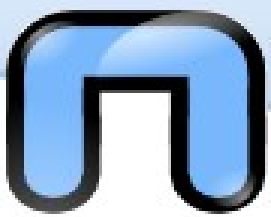


# La rete IP

## Un caso diffuso: Ethernet con TCP/IP

- Stack di rete caratterizzato da genericità del livello fisico
- Diffuso grazie alla sua semplicità utilizzato assieme a vari protocolli tra cui Ethernet
- Topologia a bus o a stella con ripetitore
- Funzionamento “plug & play” di Ethernet

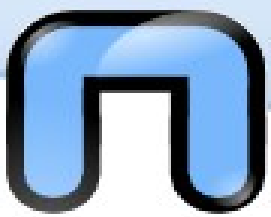




# La rete IP

## TCP/IP

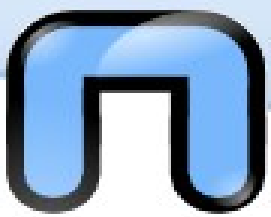
- E` alla base della rete Internet
- Essendo il livello inferiore generico, si adatta bene sia a connessioni cablate che senza fili
- Ogni singola connessione TCP/IP è sempre **identificata da 4 numeri:**
  - Indirizzo IP sorgente "130.192.73.1"
  - Indirizzo IP destinazione "130.192.73.2"
  - Indirizzo TCP/UDP sorgente "32144"
  - Indirizzo TCP/UDP destinazione "80"
- Gli ultimi due vengono chiamate "porte"



# La rete IP

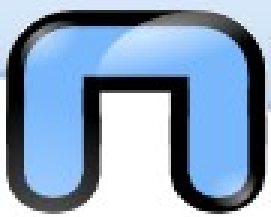
## Indirizzo IP

- Indirizzo livello 3, univoco su una rete
- IPv4: 32 bit divisi in 4 byte, rappresentati in formato decimale separati da punti
- $2^{32}$  possibili indirizzi, organizzati gerarchicamente o in modo modulare
- 11000000 10101000 00000000 00000001 → 192.168.0.1
- IPv6: 128 bit divisi in 8 gruppi di 2 byte, rappresentati in formato esadecimale
- Esempio 2001:0db8:0:0:0:0:1428:57ab
- Retrocompatibile ::c0a8:0001 , ::192.168.0.1



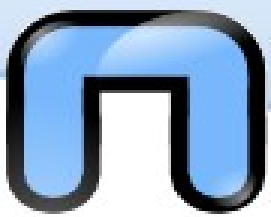
## Assegnazione

- Assegnazione IP da ente IANA
  - Classe A: 1.0.0.0 – 126.255.255.255
  - Classe B: 128.0.0.0 – 191.255.255.255
  - Classe C: 192.0.0.0 – 223.255.255.255
  - Classe D: 224.0.0.0 – 239.255.255.255 (multicast)
  - Classe E: 240.0.0.0 – 254.255.255.255 (riservati)
- 127.0.0.1 host locale (“localhost”)
- 10.0.0.0 – 10.255.255.255 Classe A locale
- 172.16.0.0 – 172.31.255.255 Classe B locale
- 192.168.0.0 – 192.168.255.255 Classe C locale



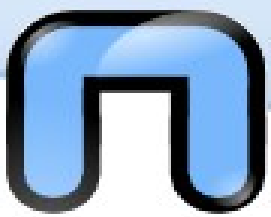
## Porte UDP / TCP

- Indirizzi di trasporto (“porte”) offrono un accesso verso il livello IP di un host con diverse modalità
- Numero da 0 a 65535
- Transmission Control Protocol (TCP)
  - comunicazione affidabile con gestione errori e ritrasmissioni
  - Setup della comunicazione
- User Datagram Protocol (UDP)
  - comunicazione diretta
  - più snello, controllo a livello superiore (5+)



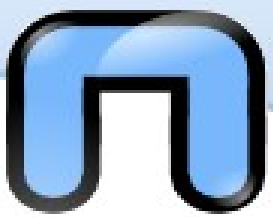
## Client e server

- L'architettura di comunicazione classica si compone di un computer centrale (“server”) che serve le richieste effettuate da diversi “client”
- E` un meccanismo conveniente
- Esistono altri paradigmi di rete, i.e. peer-to-peer
- UNIX, a cui si ispira GNU/Linux, è un sistema operativo progettato appositamente per il funzionamento su rete



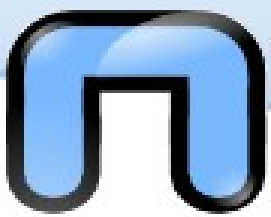
## Connettiamoci! - 1

- Oggi non solo i computer aziendali hanno una qualche interfaccia di rete
- Boom di Internet anni '90
- Tipi di collegamento: modem analogico, digitale, Ethernet, WiFi, ...
- Sono tutti “linguaggi” di livello 1 e 2 standardizzati: ITU-T V.90, IEEE 802.1, SIG Bluetooth, ...



## Connettiamoci! - 2

- In teoria la workstation client non deve avere porte aperte se non quelle dei programmi che si connettono ai server
- Solo il server dovrebbe avere delle porte sempre aperte in attesa di eventuali client:
- Corrispondono a dei protocolli detti “servizi” gestiti da programmi
- Nell'installazione di un servizio, soprattutto se il pc è connesso ad Internet, è sempre consigliabile verificare che questo sia protetto
- Chiunque può accedervi!

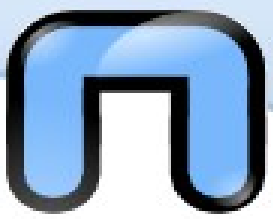


# La rete IP

## Un pò oltre - 1

- Siccome non è possibile nè è desiderabile collegare direttamente assieme tutti i computer, esistono apparati (router, gateway, firewall, ...) che prendono in carico i pacchetti a livello IP e li ritrasmettono o meno, eventualmente modificando gli indirizzi sorgente e destinazione
- Oltre all'indirizzo IP, è possibile impostare una “maschera”, che assegna il gruppo di indirizzi IP (detta “classe”) con i quali un computer può comunicare direttamente

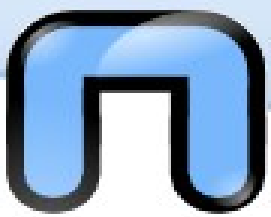




# La rete IP

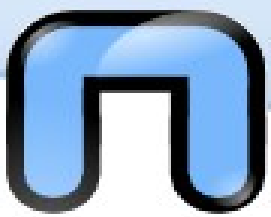
## Un pò oltre - 2

- Esempio: il computer con indirizzo 192.168.0.123 e con maschera 255.255.255.0 riceve i messaggi provenienti da 192.168.0.1 fino a 192.168.0.254
- La maggior parte delle classi sono assegnate ad enti, aziende, gruppi, etc per l'utilizzo diretto su Internet
- Alcune classi possono essere impiegati per utilizzi privati o per utilizzi particolari:
  - Rete locale privata: es. 192.168.0.0
  - Rete da uno a molti (multicast): es. 225.1.1.1



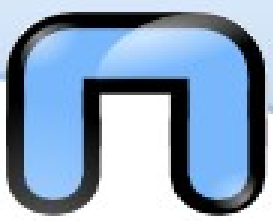
## Un pò oltre - 3

- Per poter uscire dalla propria rete locale definita dalla classe, si accede ad un host particolare, il “default router” o “gateway”
- Questo ha sicuramente più di una interfaccia
- I dati che vanno dalla sorgente a qualsiasi destinazione non locale passano sempre attraverso questo
- I router sono host che fanno solo “instradamenti” e hanno diverse interfacce solitamente con livelli fisico/datalink ottimizzati allo scopo



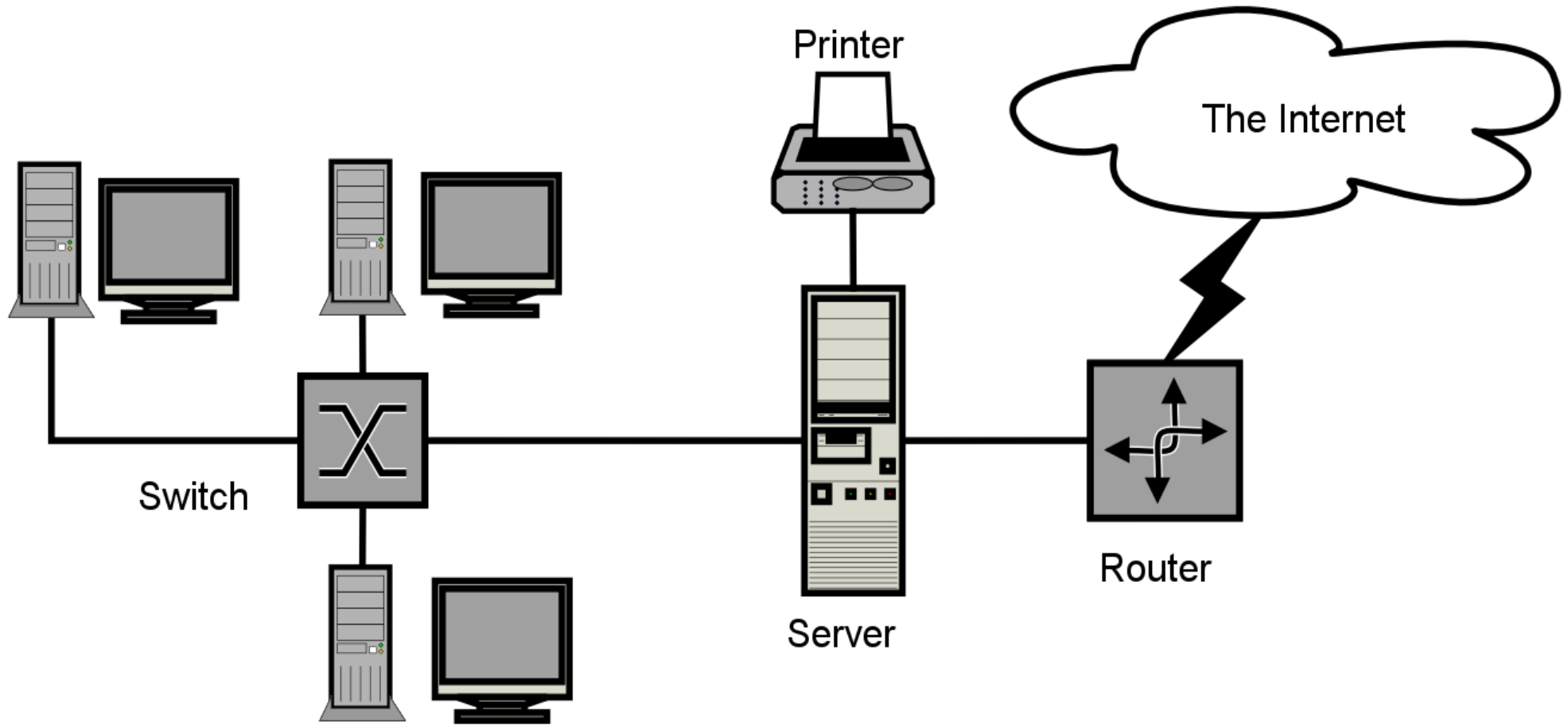
## Un pò oltre - 4

- Siccome ricordare a memoria la coppia indirizzo IP e porta destinazione non è intuitivo, è nato un indice testuale gestito in modo gerarchico, detto “nomi di dominio” (DNS)
  - `www.polito.it`, `ftp.polito.it`
- Da questo deriva anche il sistema di indirizzi email
  - `err@netstudent.polito.it`
- Esistono dei meccanismi comodi per assegnare in modo automatico indirizzi IP, router di default, nome del client, etc...
  - DHCP



# La rete IP

## Schema LAN





La rete IP

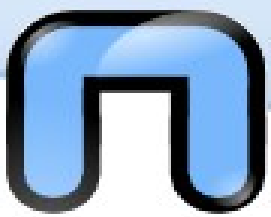
Comandi

Servizi di rete

```
root@debian:/home/fra# ifconfig
eth1  Link encap:Ethernet  HWaddr 00:1b:24:b0:73:56
      inet addr:192.168.1.2  Bcast:192.168.1.255  Mask:255.255.255.0
      inet6 addr: fe80::21b:24ff:feb0:7356/64  Scope:Link
      UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
      RX packets:290370 errors:0 dropped:0 overruns:0 frame:0
      TX packets:251195 errors:0 dropped:0 overruns:0 carrier:0
      collisions:0 txqueuelen:1000
      RX bytes:286641821 (273.3 MiB)  TX bytes:31730509 (30.2 MiB)
      Interrupt:17

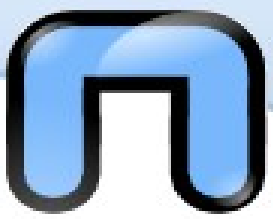
lo    Link encap:Local Loopback
      inet addr:127.0.0.1  Mask:255.0.0.0
      inet6 addr: ::1/128  Scope:Host
      UP LOOPBACK RUNNING  MTU:16436  Metric:1
      RX packets:315 errors:0 dropped:0 overruns:0 frame:0
      TX packets:315 errors:0 dropped:0 overruns:0 carrier:0
      collisions:0 txqueuelen:0
      RX bytes:21308 (20.8 KiB)  TX bytes:21308 (20.8 KiB)

root@debian:/home/fra# █
```



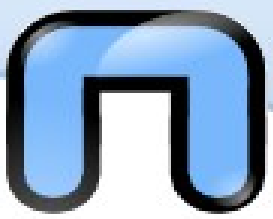
## Dettaglio connessione

- Sempre una singola connessione per combinazione  
IP locale : porta locale ↔ IP destinazione : porta dest
- Un solo programma apre una combinazione di  
indirizzo locale / porta sorgente (“socket”)
- Per sapere le combinazioni dei 4 indirizzi TCP/IP  
attualmente utilizzati:
  - `netstat -n -Ainet`
- Per conoscere programmi e porte associate:
  - `lsof -Pani`



## net-tools vs iproute2

- Al pacchetto base di configurazione della rete (a volte chiamato “net-tools”), è stato affiancato o sostituito il pacchetto “iproute2”
- Comandi “net-tools” sono storici (“netstat”, “ifconfig”, etc)
- Comandi “iproute2” più moderni
- Equivalenti ma con in più la gestione delle source-route
- `netstat -n -Ainet` oppure `ss -n`



# Comandi

## Comandi utili

- Visualizzare le interfacce di rete:

```
ifconfig -a      (ip addr)
```

- Modificare l'indirizzo IP di una interfaccia:

```
ifconfig eth0 inet 192.168.1.1 netmask 255.255.255.0
```

- Visualizzare il default router:

```
netstat -rn      (ip route)
```

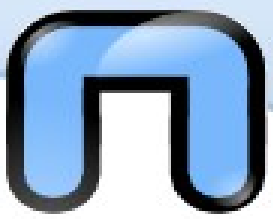
- Impostare il default router:

```
route add default gw 192.168.1.254
```

- Impostare tutti i precedenti con DHCP:

```
dhclient eth0     (pump -i eth0)
```

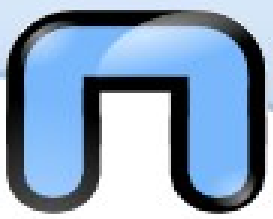




# Comandi

## Altri comandi

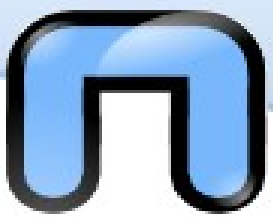
- `tcpdump`
- `ping`
- `tracert`
- `nc`



# Comandi

## File utili - 1

- Elenco porte e nome servizi:  
`/etc/services`
- Elenco server DNS:  
`/etc/resolv.conf`
- Configurazione statica di DNS:  
`/etc/hosts`
- Configurazione all'avvio degli indirizzi:  
`/etc/network/interfaces`
- Quest'ultimo varia di distro in distro!



# Comandi

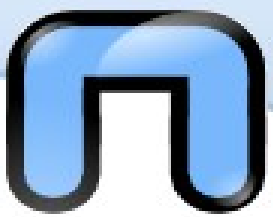
## Debian/Ubuntu

```
# cat /etc/network/interfaces
auto lo
iface lo inet loopback

iface eth0 inet static
    address 192.168.1.10
    network 192.168.1.0
    netmask 255.255.255.0
    broadcast 192.168.1.255
    gateway 192.168.1.254

auto eth0
auto eth1

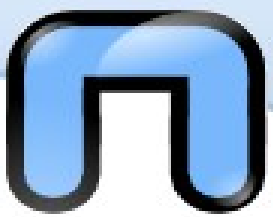
iface eth1 inet dhcp
```



# Comandi

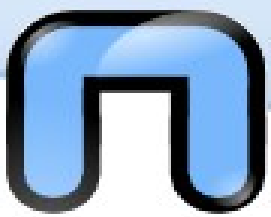
## Livello datalink e fisico?

- Finora abbiamo analizzato programmi per configurare livello IP e TCP-UDP
- *Ifconfig* e *ip* visualizza e imposta le interfacce su cui può essere utilizzato il livello IP
- Per la maggior parte dei livelli datalink e fisico indica le caratteristiche salienti (i.e. indirizzo MAC)
- Permette, nel caso di Ethernet, anche la configurazione parziale del livello datalink e fisico



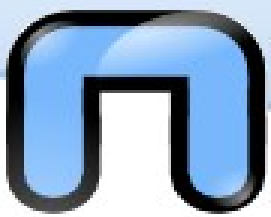
## Livello datalink e fisico - 2

- Comando “arp”
- Per accedere ai livelli sotto IP diversi da Ethernet, è possibile utilizzare altri programmi, tipicamente ad hoc per ogni tecnologia
- i.e. per il WiFi 802.11: iwconfig, iwlist, ...
- i.e. per il modem analogico: wvdial, pppconfig, ...
- i.e. per il modem digitale: pppoeconf, ...



## Esempio: WiFi

- Trovare l'AP:  
`iwlist <if> scan`
- Collegarsi all'AP:  
`iwconfig <if> essid "<essid>"`
- Se richiesta chiave WEP:  
`iwconfig <if> mode managed key "<key>"`
- Se il servizio è offerto, prendere le impostazioni di rete da DHCP:  
`dhclient <if>`
- Altrimenti, configurare l'IP con `ifconfig`, il gateway con `route`, il dns editando `/etc/resolv.conf`



# Comandi

## GUI vs shell

- Tutto ciò che è stato illustrato viene eseguito da linea di comando (shell)
- Per l'impostazione dei parametri di rete si può in generale utilizzare l'interfaccia grafica (GUI)
- i.e. NetworkManager su Ubuntu
- Di solito, i comandi sono più “ostici” da utilizzare ma più dettagliati (i.e. Netstat)
- Il debugging dei problemi è molto più immediato



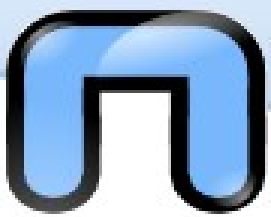
La rete IP

Comandi

**Servizi di rete**



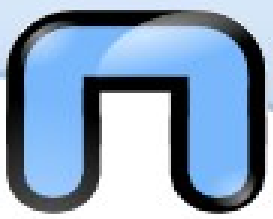




# Servizi di rete

## Server

- Con la parola “server” il programma che offre un servizio particolare
- Per estensione si indica anche il computer che offre i servizi
- Tipicamente i programmi server, siccome non interagiscono con l'utente fisico, vengono eseguiti all'avvio del computer in background come demoni
- Quindi vengono di solito eseguiti da `/etc/init.d/`



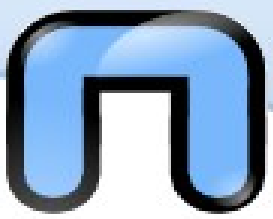
## inetd vs xinetd

- Quelli più semplici sono raccolti da un demone “super partes”, inetd oppure xinetd
- inetd: Un solo file per tutta la configurazione:
  - `/etc/inetd.conf`
- xinetd: Un file per ogni servizio:
  - `/etc/xinetd/`
- Anche loro quindi sono eseguiti da `/etc/init.d/inetd` o `/etc/init.d/xinetd`



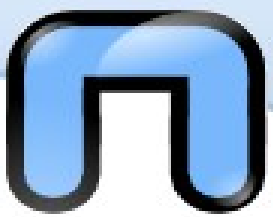
## Il servizio DHCP

- Vi sono servizi ibridi, che non lavorano solo su IP ma anche ai livelli inferiori
- E` il caso di DHCP, che configura il livello IP e applicativo (DNS) in modo automatico
- DHCP lavora anche a livello 2, infatti e` legato al mondo Ethernet (protocollo di livello datalink)



## Alcuni servizi - 1

- WWW - navigazione web HTTP:
- utilizza porta destinazione 80 su TCP
- modalità SSL utilizza porta destinazione 443 TCP
- FTP – trasferimento file binari:
- utilizza porte destinazione 20 e 21 su TCP
- Telnet – terminale remoto:
- utilizza porta destinazione 23 su TCP
- SSH – terminale remoto e trasferimento dati sicuro:
- porta 22 su TCP o UDP



## Alcuni servizi - 2

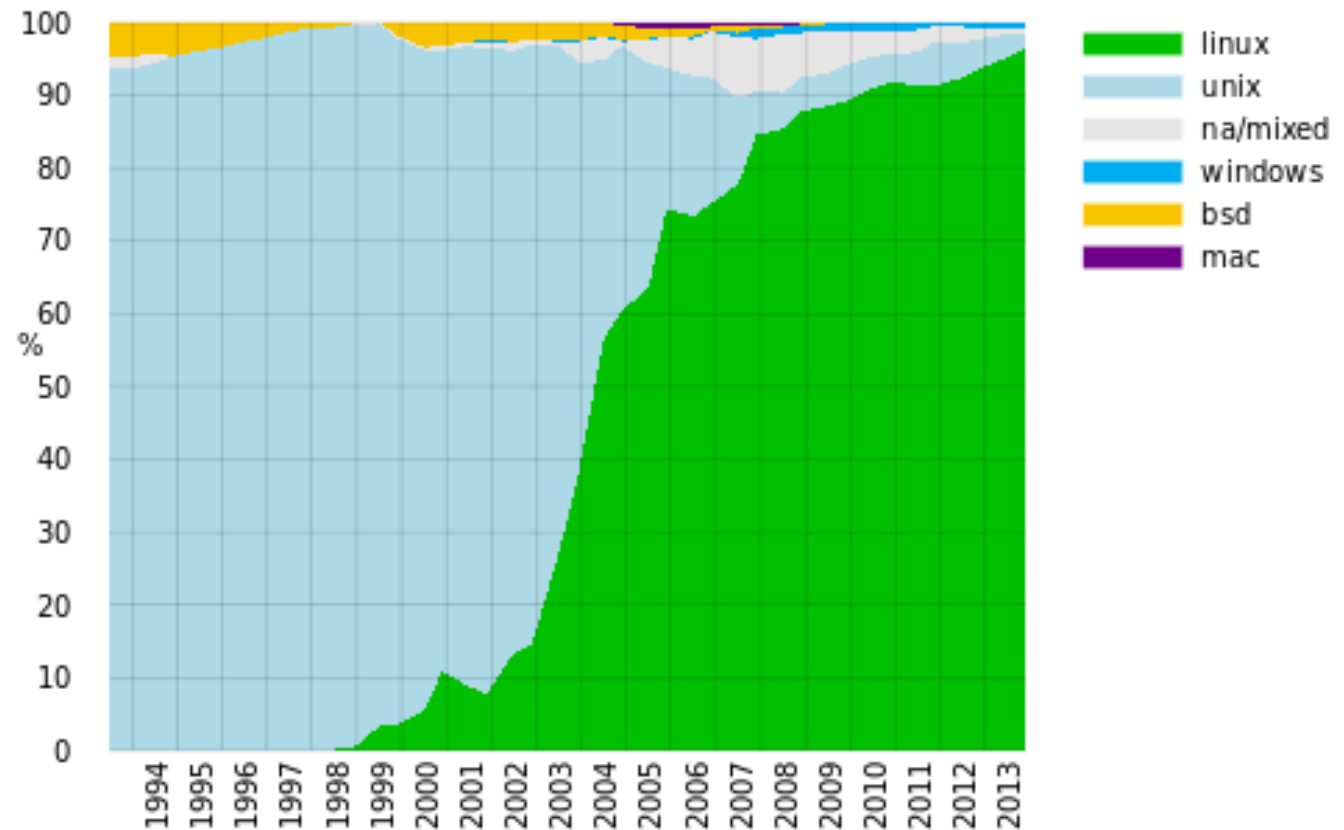
- WWW (HTTP o HTTPS):
- client Firefox, dillo, ... ; server Apache, httpd, ...
- FTP:
- client `bsd ftp`, `gftp`, ... ; server `ftpd`, `Proftpd`, ...
- SSH:
- client `openssh ssh`, `putty`, ... ; server `openssh sshd`, ..
- Database:
- Client `postgresql`, `mysql`, `odbc`; ...

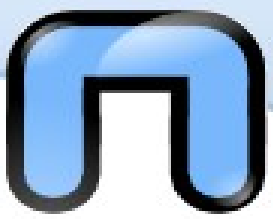


# Servizi di rete

## Considerazioni

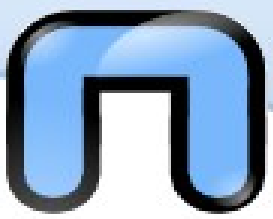
- Il caso Apache
- Servizi di rete e paradigma UNIX
- Server su sistemi operativi derivati da UNIX sono più “idonei”





# Filesystem remoto

- Fra tutti i servizi esistenti, uno molto comodo è il filesystem remoto
- In pratica un server mette a disposizione uno spazio su disco raggiungibile in remoto da diversi client, in modo tale che ai client sembra che sia a loro locale
- Questo è possibile grazie al modello a livelli
- Esistono molte implementazioni di filesystem remoto: NFS, CIFS, SMB, GoogleFS, Freenet, ...



# Filesystem remoto

- Un filesystem remoto molto diffuso in ambiti medio-piccoli nasce con la piattaforma Microsoft Windows
- La prima implementazione si chiamava SMB, quella moderna CIFS
- Un gruppo di sviluppatori ha creato (con duro lavoro) sia un client che un server compatibile con le implementazioni Microsoft, e ha arricchito il pacchetto con feature aggiuntive
- Implementazione di nome Samba Suite







# Copyleft



Quest'opera, per volontà degli autori, è rilasciata sotto la disciplina della seguente licenza

## **Creative Commons Public License**



### **Attribuzione-Condividi allo stesso modo 2.5 Italia**



Tu sei libero:

-  di riprodurre, distribuire, comunicare al pubblico, esporre in pubblico, rappresentare, eseguire e recitare quest'opera
-  di modificare quest'opera

Alle seguenti condizioni:

-  **Attribuzione.** Devi attribuire la paternità dell'opera nei modi indicati dall'autore o da chi ti ha dato l'opera in licenza e in modo tale da non suggerire che essi avallino te o il modo in cui tu usi l'opera.
-  **Condividi allo stesso modo.** Se alteri o trasformi quest'opera, o se la usi per crearne un'altra, puoi distribuire l'opera risultante solo con una licenza identica o equivalente a questa.

Ogni volta che usi o distribuisce quest'opera, devi farlo secondo i termini di questa licenza, che va comunicata con chiarezza. In ogni caso, puoi concordare col titolare dei diritti utilizzi di quest'opera non consentiti da questa licenza. Questa licenza lascia impregiudicati i diritti morali. Le utilizzazioni consentite dalla legge sul diritto d'autore e gli altri diritti non sono in alcun modo limitati da quanto sopra.

Questo è un riassunto in linguaggio accessibile a tutti del codice legale (la licenza integrale) che è disponibile alla pagina web:

<http://creativecommons.org/licenses/by-sa/2.5/it/legalcode>



# Copyleft

Quest'opera, è stata realizzata grazie al contributo di molte persone. Le slide sono distribuite con licenza Creative Commons. Esse sono state modificate dai molti docenti che hanno prestato il loro servizio gratuito nelle lezioni dei corsi Netstudent. In ordine sparso (e sperando di non dimenticare nessuno): Giovanni Berton Giachetti, Daniele Lussana, Alessandro Ugo, Emmanuel Richiardone, Andrea Garzena, Stefano Cotta Ramusino, Roberto Preziosi, Marco Papa Manzillo, Puria Nafisi Azizi, Luca Necchi, Luca Barbato, David Putzer, Alberto Grimaldi, Nicola Tuveri, Stefano Colazzo, Laura De Martini, Silvio Colloca ecc...