

Configurazione di una rete locale con Linux

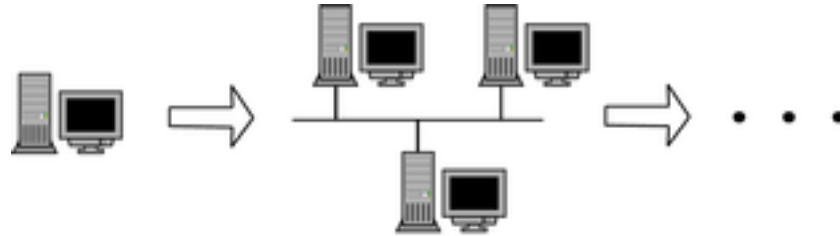


paolo palmerini

paolo@progettoarcobaleno.it

<http://www.progettoarcobaleno.it>

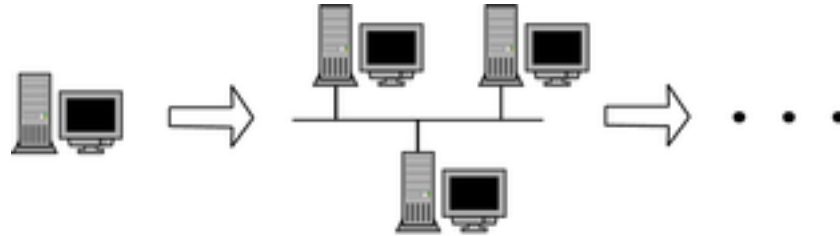
Lavorare in “rete”



- ▶ Che hardware serve per installare una rete?



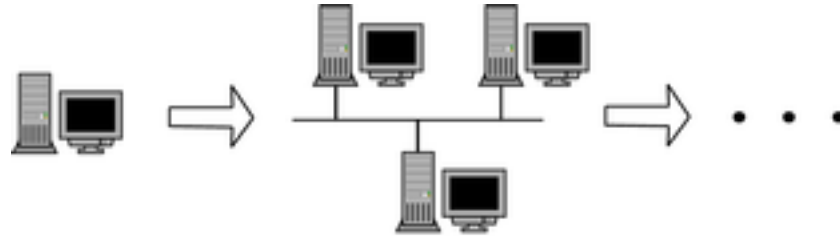
Lavorare in “rete”



- ▷ Che hardware serve per installare una rete?
- ▷ Indirizzi IP, network e routing



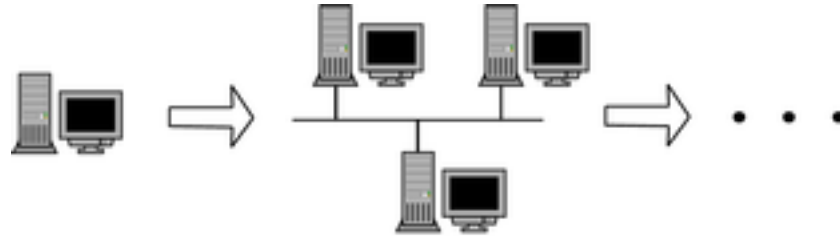
Lavorare in “rete”



- ▷ Che hardware serve per installare una rete?
- ▷ Indirizzi IP, network e routing
- ▷ Configurazione di una semplice rete



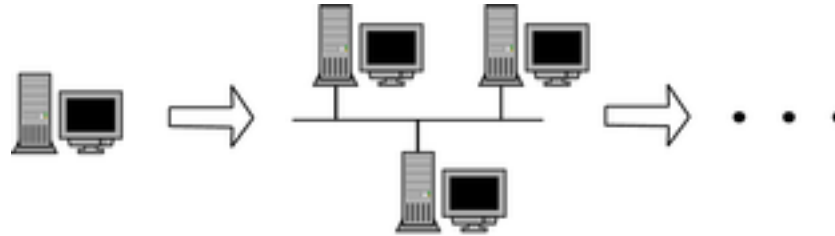
Lavorare in “rete”



- ▷ Che hardware serve per installare una rete?
- ▷ Indirizzi IP, network e routing
- ▷ Configurazione di una semplice rete
- ▷ Condivisione accesso ad Internet



Lavorare in “rete”



- ▷ Che hardware serve per installare una rete?
- ▷ Indirizzi IP, network e routing
- ▷ Configurazione di una semplice rete
- ▷ Condivisione accesso ad Internet
- ▷ Condivisione risorse



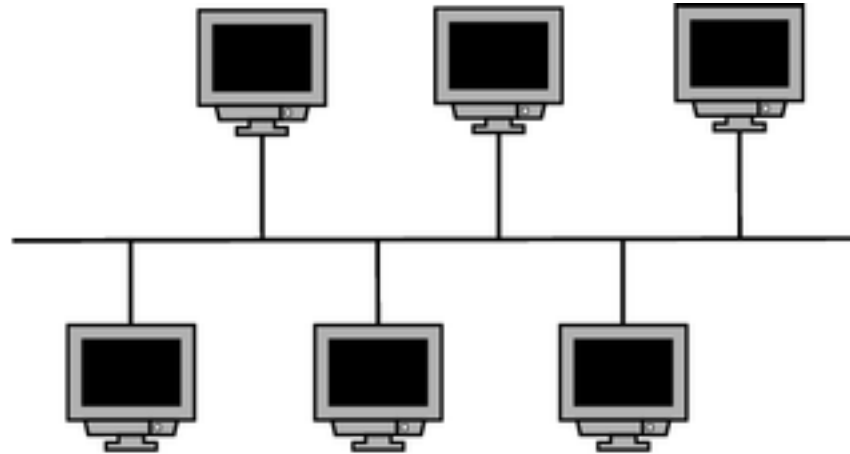
Hardware

Scelta dei componenti base:

- ▶ Interfaccia di rete (scheda): ISA, PCI, ...
- ▶ Connessioni e cavi: RJ-45 (coassiale) o UTP-5 (doppino)?
- ▶ Tipo di scheda: 10 Mbps, 100Mbps, 1Gbps



Ethernet su coassiale (o BNC)



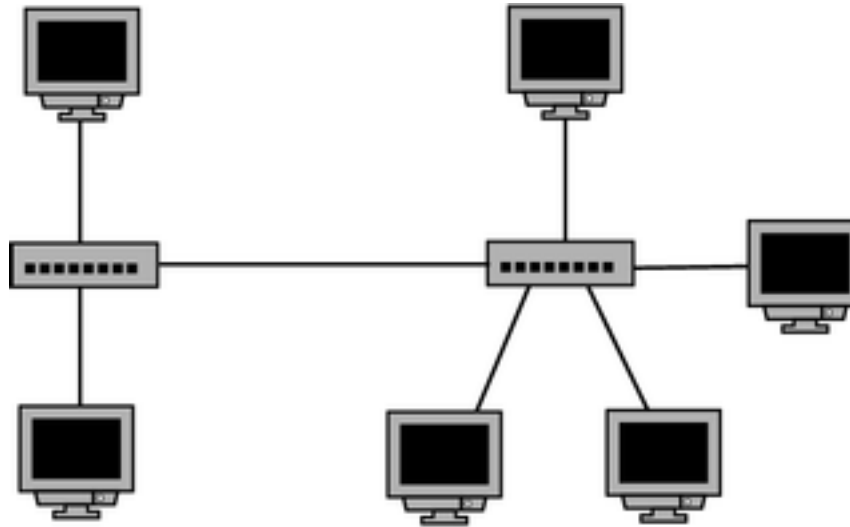
- ▶ Economica: hardware di recupero o a bassissimo prezzo
- ▶ Struttura semplice (lineare)

Controindicazioni:

Un singolo punto di rottura compromette tutta la rete!



Rete su doppino (UTP)



- ▶ Meno soggetta a malfunzionamenti
- ▶ Struttura “a stella” (necessita di ripetitori o *hub*)
- ▶ Supporta velocità maggiori



Indirizzi IP

- ▷ Ad ogni PC in rete viene assegnato un indirizzo IP
 - ★ è un numero fra 0.0.0.0 e 255.255.255.255
 - ★ es: 146.48.82.89
- ▷ Anche le reti hanno un indirizzo: serve per definire l'insieme degli IP che possono essere assegnati alle macchine appartenenti alla rete.
 - ★ es: nella rete 192.168.10.0 posso avere indirizzi da 192.168.10.1 a 192.168.10.254
- ▷ Esistono degli indirizzi speciali per le reti “private”.
 - ★ es: tutti gli indirizzi compresi fra 192.168.0.0 e 192.168.255.255

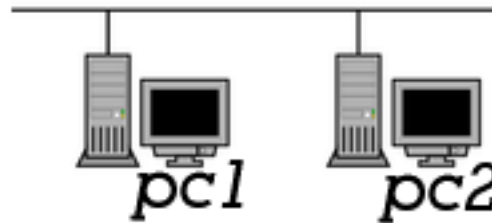


Indirizzi e nomi

- ▷ Gli indirizzi numerici sono “difficili” da ricordare: meglio dare dei **nomi**!
- ▷ Agli indirizzi IP vengono associati nomi:
 - ★ es: 148.46.82.89 → `miles.cnuce.cnr.it`
- ▷ Esistono dei server che si occupano di *risolvere* (tradurre) gli indirizzi in nomi e viceversa: i **DNS**



Esempio: due PC in rete



- ▷ Rete: 192.168.10.0
- ▷ pc1
 - ★ Indirizzo: 192.168.10.1
- ▷ pc2
 - ★ Indirizzo: 192.168.10.2



Configurazione base del PC

1. Installare la scheda di rete dentro il PC
2. Installare il driver della scheda di rete

```
[root@pc1 ~]# modprobe ne2k-pci  
[root@pc1 ~]#
```

3. Impostare l'indirizzo IP

```
[root@pc1 ~]# ifconfig eth0 192.168.10.2  
[root@pc1 ~]#
```

4. Si ripete l'operazione con un pc2



Routing e Nomi

- ▶ E' necessario specificare quale strada (routing) devono fare le comunicazioni per andare ad un PC ad un altro

- ★ Su entrambi i PC: `route add -net 192.168.10.0 eth0`

- ▶ Per poter chiamare i PC “per nome”

- ★ Su entrambi i PC:

- ```
[root@pc1 ~]# cat /etc/hosts
127.0.0.1 localhost localhost.localdomain
192.168.10.1 pc1
192.168.10.2 pc2
```



# Verifica della configurazioni

- ▶ Su ogni PC: `ifconfig`

```
[root@pc1 ~]# ifconfig
eth0 Link encap:Ethernet HWaddr 00:A0:0C:14:84:0B
 inet addr:192.168.10.1 Bcast:192.168.10.255 Mask:255.255.255.0
 UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
 RX packets:0 errors:0 dropped:0 overruns:0 frame:0
 TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
 collisions:0 txqueuelen:100
 Interrupt:3 Base address:0x300
```

- ▶ Routing

```
[root@pc1 ~]# route -n
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
192.168.10.0 0.0.0.0 255.255.255.0 U 0 0 0 eth0
127.0.0.0 0.0.0.0 255.0.0.0 U 0 0 0 lo
```



# Verifica della configurazioni

- ▶ I PC “si vedono”?

```
[root@pc1 ~]# ping 192.168.10.2
PING 192.168.10.2 (192.168.10.2) from 192.168.10.1 : 56(84) bytes of data.
64 bytes from 192.168.10.2: icmp_seq=0 ttl=255 time=720 usec
64 bytes from 192.168.10.2: icmp_seq=1 ttl=255 time=730 usec
64 bytes from 192.168.10.2: icmp_seq=2 ttl=255 time=750 usec
^C
--- 192.168.10.2 ping statistics ---
2 packets transmitted, 2 packets received, 0% packet loss
round-trip min/avg/max/mdev = 0.720/0.735/0.750/0.015 ms
```

- ▶ La rete é configurata correttamente!!
- ▶ Per aggiungere altre macchine si ripetono le stesse operazioni





# Usiamo la rete: FTP (*File Transfer Protocol*)

- ▶ “Navigazione” nelle directories del sistema remoto
- ▶ Trasferimento di files (in entrambe le direzioni)

```
PC1> ftp PC2
Connected to PC2.
220 PC2 FTP server (Version 6.2/OpenBSD/Linux-0.10) ready.
Name (PC2:utente): utente1
331 Password required for utente1.
Password:
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```



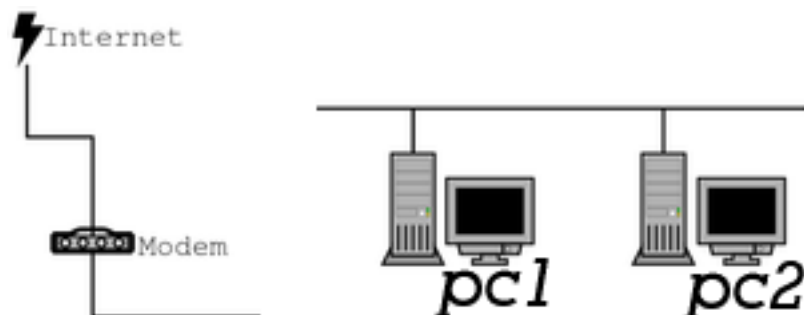
# Telnet

Aprire un “terminale” sul PC remoto, dal quale è possibile impartire tutti i comandi a disposizione

```
PC1> telnet PC2
Trying 192.168.10.2...
Connected to PC2.
Escape character is '^]'.
Debian GNU/Linux 2.2 PC2
PC2 login: gb
Password:
Last login: Wed Nov 28 13:48:26 2001 from :0 on 4
Linux PC2 2.4.14 #2 Sun Nov 18 17:15:16 GMT 2001 i686 unknown
No mail.
[gb@pc2 ~] $
```



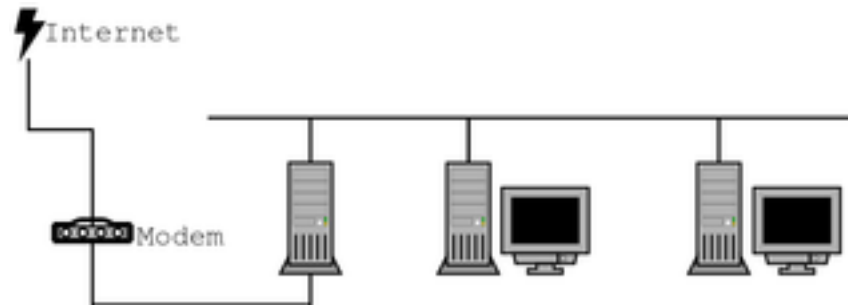
# Esempio 2: Condivisione accesso Internet



- ▶ Vogliamo dare accesso ad internet a tutti i computer della rete...
- ▶ ... ma abbiamo un unico punto di accesso!



# Esempio 2 (continua)



- ▶ (quasi) nessuna modifica ai due PC già presenti
- ▶ aggiunta di un terzo elemento... il **router**



# Modifica alla configurazione dei PC

## 1. Configurare pc1 e pc2

```
PC1:~# route add default gw 192.168.10.47 eth0
PC1:~#
```

## 2. Si aggiunge un nuovo PC

★ pc-gw: 192.168.10.47 (*gateway*)

Il *gateway* è il nostro punto di accesso a Internet: in questo caso quindi il nostro router (192.168.10.47 è appunto il suo indirizzo IP)



# Un router economico



PC di recupero (un 486 è sufficiente)

Hardware per la connessione a internet (es.: modem)

Scheda di rete per la connessione alla rete locale

1 dischetto...



# Router linux su 1 dischetto:

Un sistema linux completo su di un dischetto floppy: una volta configurati pochi parametri (es.: numero di telefono, nome utente, password...) è pronto

Linux Router Project (LRP)

<http://www.linuxrouter.org>

Freesco

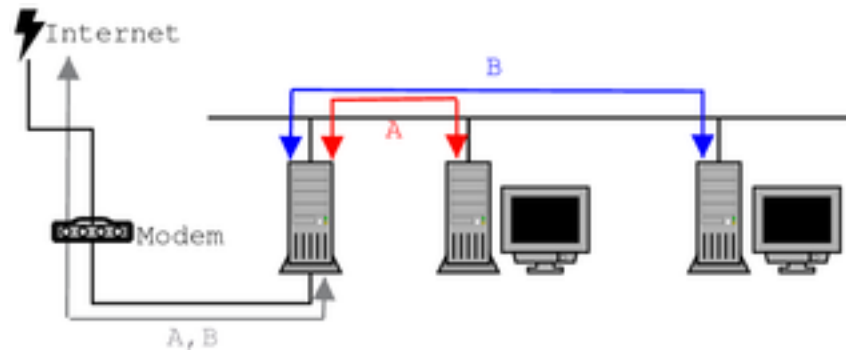
<http://www.freesco.com>

Floppyfw

<http://www.zelow.no/floppyfw>



# Il trucco: IP masquerading...



Tutte le richieste effettuate dai PC nella rete appaiono provenire dal router

Le informazioni in risposta sono instradate dal router al legittimo destinatario





# Riepilogo

- ▶ Abbiamo creato una rete locale con 2 o piu' PC



# Riepilogo

- ▷ Abbiamo creato una rete locale con 2 o piu' PC
- ▷ Abbiamo fornito accesso ad Internet a tutti i PC in rete...



# Riepilogo

- ▷ Abbiamo creato una rete locale con 2 o piu' PC
- ▷ Abbiamo fornito accesso ad Internet a tutti i PC in rete...
- ▷ ... tramite l'installazione di un router



# Riepilogo

- ▷ Abbiamo creato una rete locale con 2 o piu' PC
- ▷ Abbiamo fornito accesso ad Internet a tutti i PC in rete...
- ▷ ... tramite l'installazione di un router
- ▷ (non abbiamo mai dovuto riavviare il PC!!)

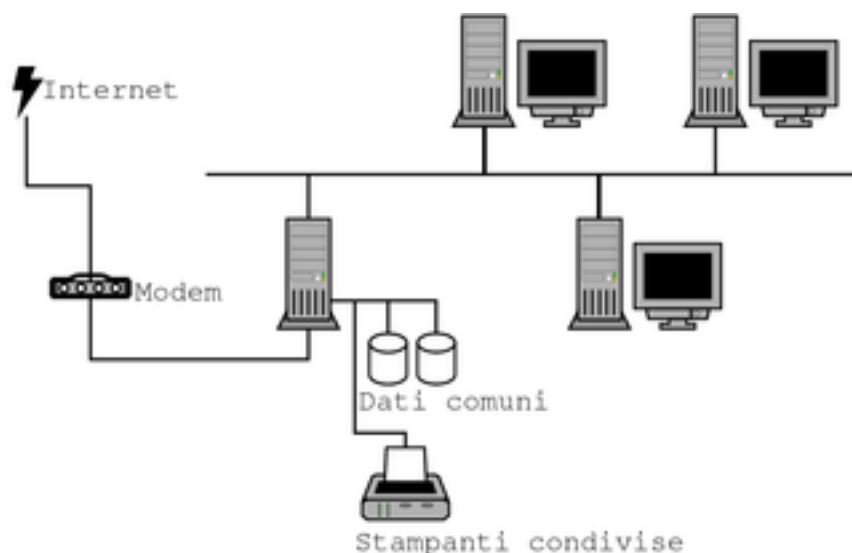


# Riepilogo

- ▷ Abbiamo creato una rete locale con 2 o piu' PC
- ▷ Abbiamo fornito accesso ad Internet a tutti i PC in rete...
- ▷ ... tramite l'installazione di un router
- ▷ (non abbiamo mai dovuto riavviare il PC!!)
- ▷ Cosa possiamo fare ancora?



# Condivisione dati e stampanti



Fra sistemi Linux:

- ▶ NFS  
(Network File System)
- ▶ cups, lpd  
(Printing Daemon)
- ▶ Per condividere risorse con altri sistemi:  
...SAMBA



# Configurazioni

- NFS (condivisione dischi)

- ▷ Server: file /etc/exports

- /home 192.168.10.0(rw)
    - /usr/local 192.168.10.0(ro)

- ▷ Client: file /etc/fstab

- 192.168.10.47:/home /home1 nfs defaults 0,0
    - 192.168.10.47:/usr/local /opt1 nfs defaults 0,0

- CUPS (condivisione stampanti)

- ▷ Server: tool grafico di configurazione

- ▷ Client: “niente”!

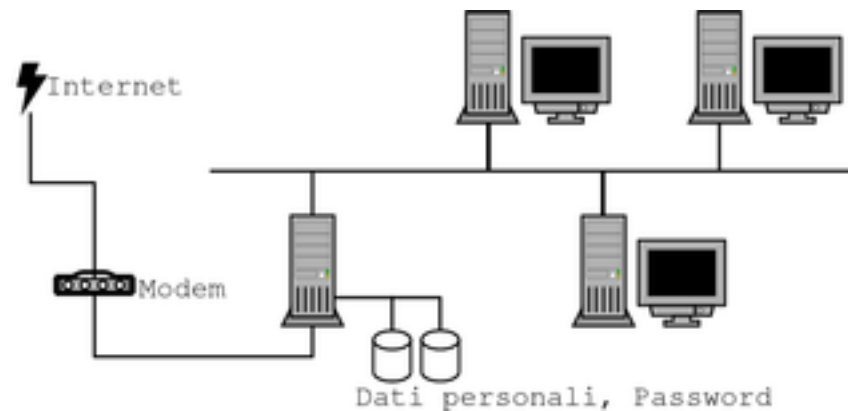
- SAMBA (condivisione con windows)

- ▷ Server: file /etc/smb.conf

- ▷ Client: “Connetti unita’ di rete”



# Condivisione utenti e password

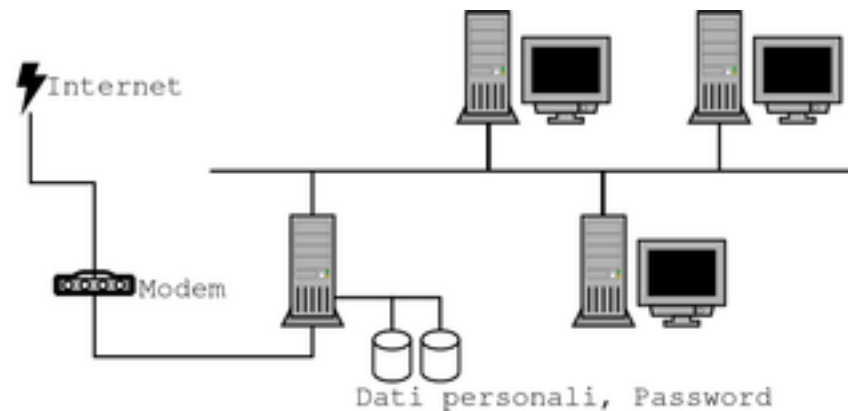


- ▶ Linux e' un sistema multiutente: per accedere ad un PC bisogna fornire **username** e **password**





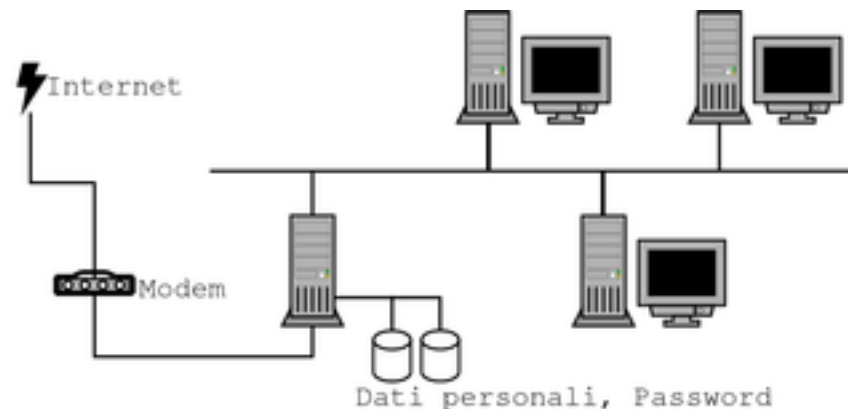
# Condivisione utenti e password



- ▶ Linux e' un sistema multiutente: per accedere ad un PC bisogna fornire **username** e **password**
  - ★ Su ciascun PC!



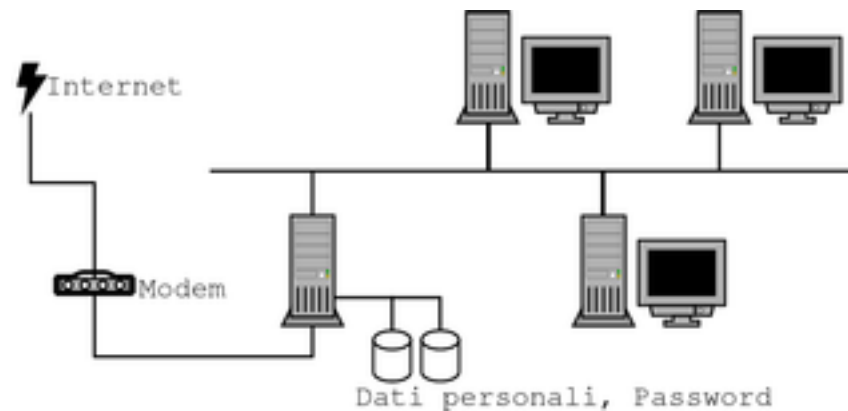
# Condivisione utenti e password



- ▶ Linux e' un sistema multiutente: per accedere ad un PC bisogna fornire **username** e **password**
  - ★ Su ciascun PC!
- ▶ La nostra rete comincia a crescere...



# Condivisione utenti e password



- ▶ Linux e' un sistema multiutente: per accedere ad un PC bisogna fornire **username** e **password**
  - ★ Su ciascun PC!
- ▶ La nostra rete comincia a crescere...
- ▶ Vorremmo centralizzare la procedura di autenticazione



# NIS (Network Information Service)

- ▶ Tramite NIS l'autorizzazione all'accesso è effettuata dal server (che contiene tutte le password ecc.)
- ▶ Tramite NFS (già visto...) i dati personali degli utenti stanno sul server e sono condivisi con tutte le postazioni.
- ▶ La rete inizia ad avere un notevole carico... occorre un collegamento stabile e veloce!



# Hardware

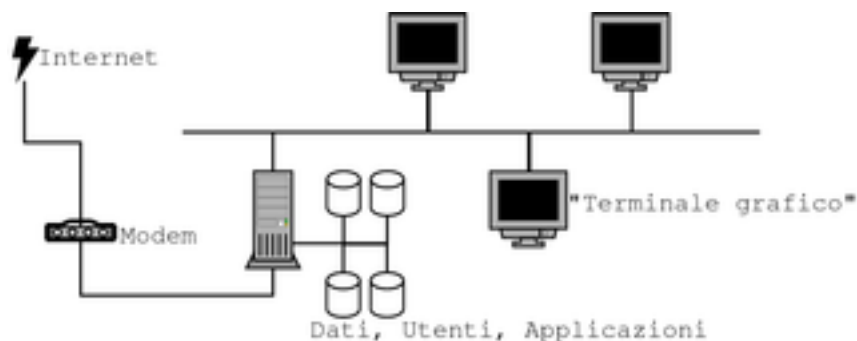
Con l'ultimo passaggio, anche i requisiti del server sono maggiori:

- ▶ Potenza di calcolo
- ▶ Spazio e velocità sistema disco

Con un passo successivo si può cominciare a ridurre le esigenze dal punto di vista delle postazioni di lavoro...



# Sistema a “terminali”



- ▶ Anche il carico di far “girare” le applicazioni è spostato sul server... la postazione di lavoro è ridotta ad un’installazione minimale
- ▶ Questo puo’ essere particolarmente utile per “riciclare” vecchi PC che non hanno abbastanza risorse per essere autonomi (disco, CPU, ecc..)



# Proteggere il server

- ▶ Ora che tutto e' concentrato sul server, abbiamo bisogno di essere sicuri che il server non si guasti e perda tutti i dati



# Proteggere il server

- ▶ Ora che tutto e' concentrato sul server, abbiamo bisogno di essere sicuri che il server non si guasti e perda tutti i dati
- ▶ Utilizzo di un gruppo di continuita'





# Proteggere il server

- ▶ Ora che tutto e' concentrato sul server, abbiamo bisogno di essere sicuri che il server non si guasti e perda tutti i dati
- ▶ Utilizzo di un gruppo di continuita'
- ▶ Back-up periodico



# Proteggere il server

- ▶ Ora che tutto e' concentrato sul server, abbiamo bisogno di essere sicuri che il server non si guasti e perda tutti i dati
- ▶ Utilizzo di un gruppo di continuita'
- ▶ Back-up periodico
- ▶ Utilizzo di dischi "ridondanti"



# Un sistema di backup automatico

- ▶ Un PC (anche il solito 486 che ora come router non va più bene e avanza...)
- ▶ Un bel po' di spazio disco
- ▶ Usando la temporizzazione delle operazioni effettuata da linux (`cron`), e utilizzando il già noto NFS, si può, nottetempo, far copiare tutti i dati importanti sul 486...



# RAID: meglio abbondare!



- ▶ Si utilizzano due dischi identici ma se ne “vede” solo uno
- ▶ L’informazione è ridondante... una rottura non solo non causa perdita di dati, ma nemmeno l’arresto del sistema!
- ▶ In alcune distribuzioni i dischi RAID vengono configurati durante l’installazione di Linux



# Riepilogo finale

Con hardware poco costoso e software libero, anche una piccola realta' puo' dotarsi di una sofisticata infrastruttura informatica

- ▷ Scelta dell'hardware:
  - ★ BNC economico e poco robusto
  - ★ UTP un po' piu' caro ma piu' affidabile
  
- ▷ Per collegare due (o piu') PC:
  - ★ Assegnare gli indirizzi (nella stessa rete!)
  - ★ Definire il routing
  
- ▷ E' possibile condividere l'accesso ad internet tramite un unico punto di accesso
  - ★ E' necessario configurare un router (freesco)



# Riepilogo (cont.)

- ▷ Condivisione risorse interne
  - ★ Stampanti (cups, lpd)
  - ★ Dischi (NFS, samba)
  - ★ Applicazioni
  - ★ Autenticazione utenti (NIS)
  
- ▷ Per avere un server robusto
  - ★ Dischi in mirroring (RAID 1)
  - ★ Back-up periodico (tramite un altro PC)

