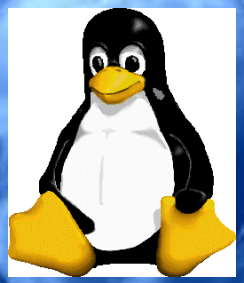
NETLINK s.a.s.



Corso di Linux

Dott. Paolo PAVAN

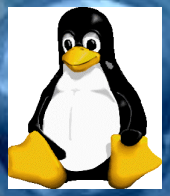
pavan@netlink.it

Origini di Linux

- Nasce a fine anni 1980 ad opera del finlandese Linus Torvalds
- Linux non è gratis e neanche freeware è protetto da licenza GNU:

*"GNU non è di pubblico dominio. **A tutti sarà permesso di modificare e ridistribuire GNU**, ma a nessun distributore sarà concesso di porre restrizioni sulla sua ridistribuzione. Questo vuol dire che non saranno permesse modifiche proprietarie.".*

Richard Stallman Fondatore GNU FSF

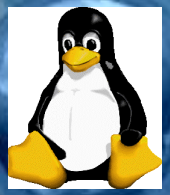


Il Software "Libero"

NETLINK

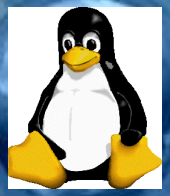
NETLINK .o.a.S

- BSD
- FreeBSD
- GNU (NotUnix)
  - GPL
    - La General Public License è la licenza d'uso che accompagna i software che nascono sotto il marchio GNU
  - Linux: Minix e Kernel Linux
  - Open Source
    - Definizione nata nel 1998 ad identificare i principi secondo cui il software può essere ritenuto libero.



Distribuzioni di Linux

- Slackware
- RedHat
- Debian
- SuSE
- Caldera Open Linux
- Mandrake
- TurboLinux
- Gentoo
- Lindows, Linspire

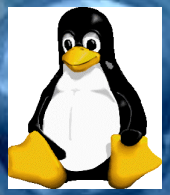


Concetto di Distribuzione

NETLINK

NETLINK s.a.s.

- Una distribuzione è una versione di Linux pronte da installare. E' dotata di:
  - Ambiente Linux completo (kernel, compilatori e programmi base)
  - Tool e programmi per un setup semplice a assistito
  - Programmi aggiuntivi (anche non freeware)
  - Supporto per l'assistenza tecnica.



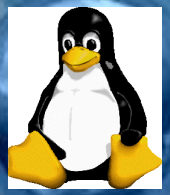
Sono questi ultimi, assieme al tipo di setup (sempre più semplificato) che differenziano una distribuzione da un'altra.

Linux e il Networking

NETLINK

NETLINK s.a.s.

- Linux supporta in modo nativo TCP/IP e molti altri protocolli.
- NFS
- Samba (SMB/CIFS)
- Appletalk
- IPX (Mars_Nwe)

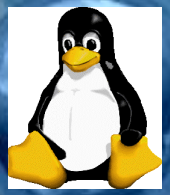


Linux e il Filesystem

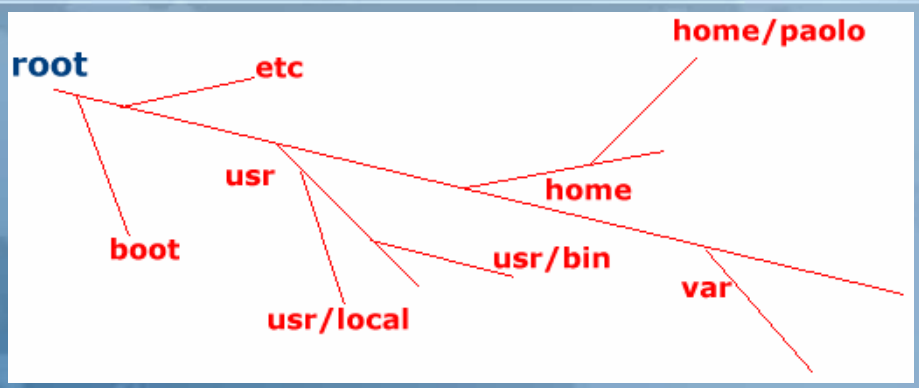
NETLINK

NETLINK.S.a.S.

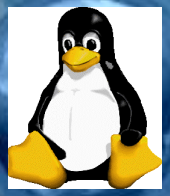
- E' composto da due strutture che si sovrappongono: inode e directory.
  - Superblocco
  - inode
  - blocco di dati
  - directory
- I comandi di amministrazione del filesystem
  - *fdisk, mke2fs, e2fsck*
  - *mount, umount, mkswap*
  - *halt, reboot e shutdown*
  - *cp, mkdir, rm, mv,ls,df e free*



Filesystem

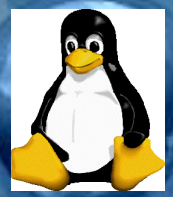


Il **filesystem rappresenta il modo con cui sono organizzati i dati all'interno di un disco** o di una sua partizione. Nei sistemi operativi Unix non esiste la possibilità di distinguere tra un'unità di memorizzazione e un altra, come avviene nel Dos, in cui ogni disco o partizione sono contrassegnati da una lettera dell'alfabeto (A:, B:, C:). Nei sistemi Unix, tutti i filesystem cui si vuole poter accedere **devono essere concatenati assieme**, in modo da formare un unico filesystem globale. Sotto Unix si dice che un file system viene **montato o innestato** su quello principale.



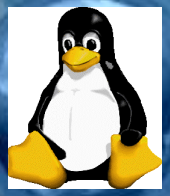
Elementi del Filesystem

- Un filesystem Unix ha due livelli di astrazione logica: inode e directory.
  - Un **inode** è un elemento contenente tutte le informazioni riferite a un file di qualunque tipo (comprese le directory), escluso il nome. In particolare, l'inode contiene i riferimenti necessari a raggiungere i blocchi di dati del file. Gli inode sono raggiungibili tramite il loro numero (numero di inode).
  - Un **blocco di dati** è una zona nel disco utilizzata per contenere dati, corrispondente a un multiplo della dimensione del settore fisico del disco stesso. Il contenuto di un file può essere distribuito su più blocchi di dati (→ cluster).
  - Una **directory** è un file contenente un elenco di nomi di file abbinati al rispettivo inode.
  - Il **superblocco** che contiene informazioni generali sul filesystem.



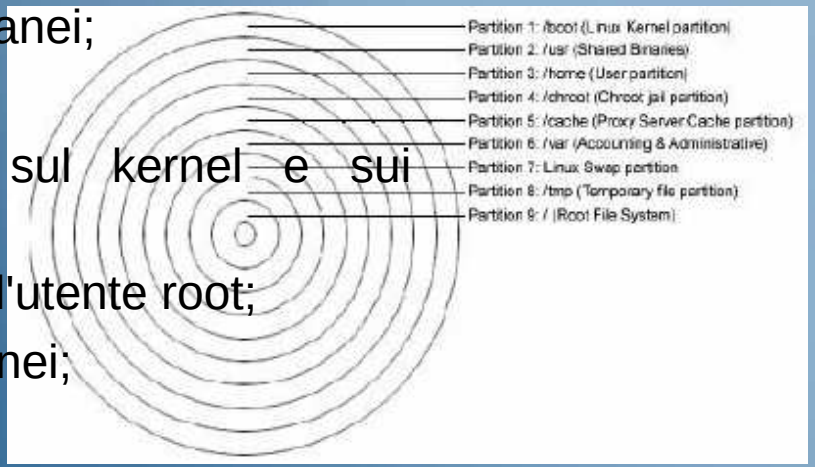
La Radice

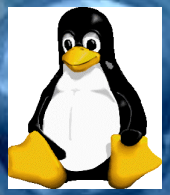
- La directory radice è quella che contiene tutte le altre. Di solito contiene solo directory con l'unica eccezione del file del kernel che può risiedere qui o in /boot/.



La Radice: Struttura

- La radice contiene le seguenti directory:
  - /bin/ -- binari essenziali;
  - /boot/ -- file statici per l'avvio del sistema;
  - /dev/ -- file di dispositivo;
  - /etc/ -- configurazione particolare del sistema;
  - /home/ -- directory personali degli utenti;
  - /lib/ -- librerie essenziali e moduli del kernel;
  - /mnt/ -- punti di innesto temporanei;
  - /opt/ -- applicativi aggiuntivi;
  - /proc/ -- informazioni vitali sul kernel e sui processi;
  - /root/ -- directory personale dell'utente root;
  - /tmp/ -- file e directory temporanei;
  - /usr/ -- gerarchia secondaria;
  - /var/ -- dati variabili.



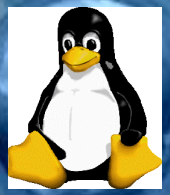


Files

I file sono essenzialmente di tre tipi.

- statici o variabili;
- condivisibili o non condivisibili;
- indispensabili per il boot o meno.

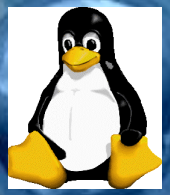
- Quelli statici possono essere resi accessibili in sola lettura (esecuzione compresa), mentre il resto deve necessariamente essere accessibile anche in scrittura.
- Quelli condivisibili possono essere utilizzati da più elaboratori contemporaneamente.



Utenti e Gruppi del sistema

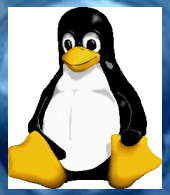
Il sistema è controllato da:

- users
- Appartengono a specifici gruppi
- root è il super user
  - a poteri totali su sistema
  - Ha utente e gruppo 0
  - Va assegnato a utenti esperti
- file passwd
  - root:x:0:0:paolo,,,,:/root:/bin/bash
- File shadow
  - root:$1$1au2xYB1$C8925wTLfJYctPBCp.Azh0:11954:0:99999:7:::
- File group
  - root::0:root



Owner e Permessi

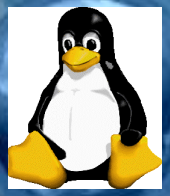
- Identificati da un numero (UID e GID)
- Comandi di gestione
  - Adduser, deluser e finger
- Gestione dei permessi:
  - Notazione testuale
  - Notazione Ottale



Sistema testuale

- R (Read = Lettura)
- W (Write = Scrittura)
- X (Execute =Esecuzione)

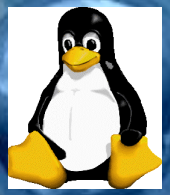
- Esempi
  - chmod u+x script.sh
  - chmod –r index.html



Sistema Ottale (assoluto)

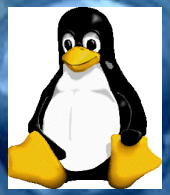
User			Group			Others		
R	W	X	R	W	X	R	W	X
4	2	1	4	2	1	4	2	1

- chmod 755 public_html
  - Assegna rwx allo user proprietario e la lettura e l'esecuzione al gruppo e agli altri
- chmod 644 index.html
  - Assegna lettura e scrittura allo user proprietario e la lettura al gruppo e agli altri



Kernel

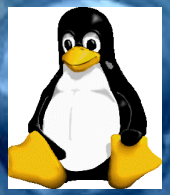
- Il kernel, come suggerisce il nome, è il **nocciolo del sistema operativo**. I programmi utilizzano il kernel per le loro attività, e in questa maniera sono sollevati dall'agire direttamente con la CPU. Di solito, è costituito da un file unico, il cui nome potrebbe essere *vmlinuz.*



Kernel: particolarità

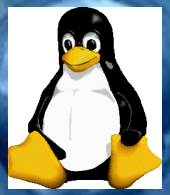
Il Kernel può anche essere visto come il principale software del Sistema Operativo che potrebbe anche includere la gestione grafica:

- sotto Linux (come nella maggior parte dei sistemi Unix-like), l'ambiente XWindow non fa parte del Kernel di Linux in quanto gestisce soltanto operazioni grafiche (grazie ad istruzioni I/O eseguite in User Mode e accesso diretto alla scheda video).
- negli ambienti Windows (Win9x, WinME, WinNT, Win2K, WinXP) sono un mix tra ambiente grafico e kernel.



Kernel: Struttura

- Statico:
  - Tutti gli elementi supportati vanno a costituire un kernel monolitico.
  - E' più pesante ma più sicuro
- Modulare:
  - può comprendere anche moduli aggiuntivi, per la gestione di componenti hardware specifici che devono poter essere attivati e disattivati durante il funzionamento del sistema.
  - Snello e versatile ma consente l'attacco degli LKM.

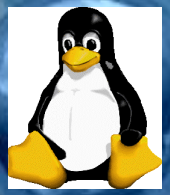


Kernel: Avvio del Sistema

NETLINK

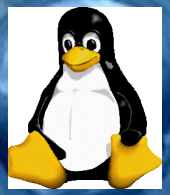
NETLINK S.a.S.

- Quando il kernel viene avviato (attraverso il sistema di avvio), esegue una serie di controlli diagnostici in base ai tipi di dispositivi (componenti hardware) per il quale è stato predisposto, quindi monta **(mount)** il filesystem principale **(root)**, e infine avvia la procedura di inizializzazione del sistema **(Init)**.



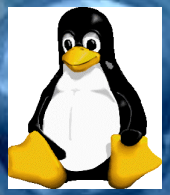
Kernel: Versioni

- Si suddivide in versioni stabili e versioni sperimentali.
- La dicitura standard è 2.x.y, dove se x è pari significa che la versione è stabile se invece è dispari allora è sperimentale.
  - X: major number
  - y: minor number

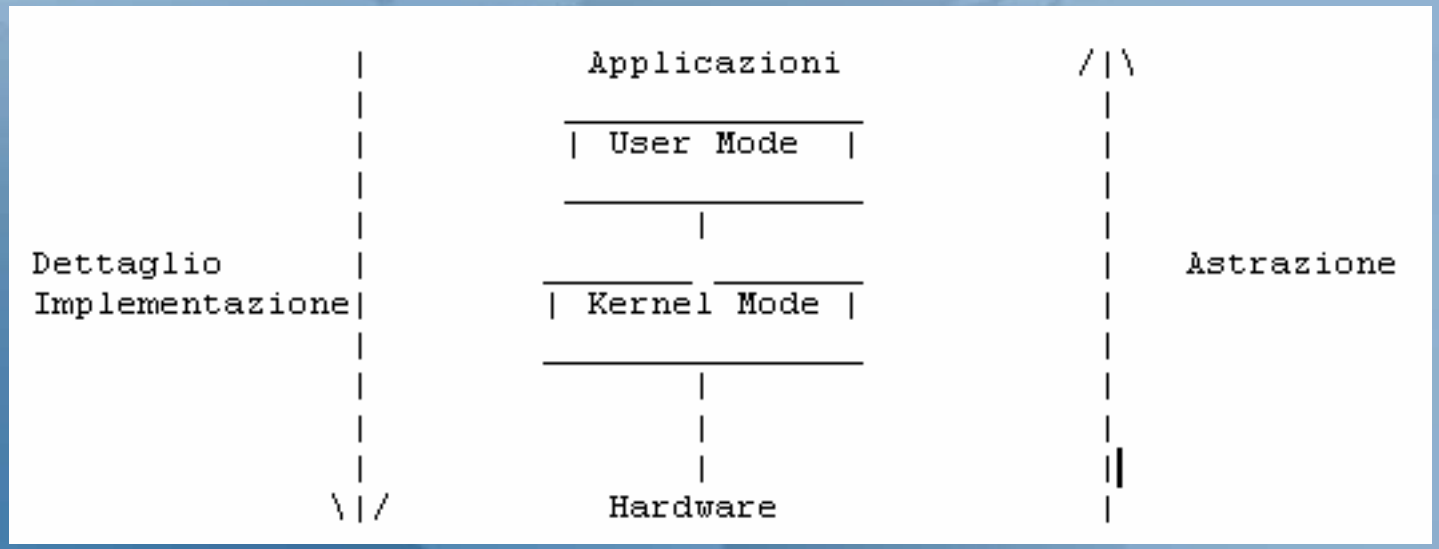


Perché User Mode e Kernel Mode

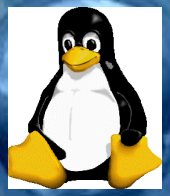
- Una volta sui sistemi operativi c'era il rischio di lanciare un programma e mandare in crash tutto il sistema, in modo repentino ed irreparabile.
- Introducendo due livelli di esecuzione (Kernel mode e User mode) si è risolto il problema
  - **Kernel Mode**: in cui la macchina opera con risorse critiche, come l'hardware (IN/OUT o memory mapped), accesso diretto alla memoria, IRQ, DMA)
  - **User Mode**: in cui gli utenti possono far girare le loro applicazioni senza preoccuparsi di bloccare il sistema.



User e Kernel Mode: Schema

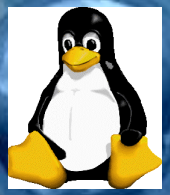


Kernel Mode impedisce alle applicazioni User Mode di danneggiare il sistema o le sue caratteristiche.
I moderni processori implementano in hardware almeno 2 stati differenti. Intel per esempio possiede 4 stati che determinano il PL (Privilege Level) permettendo quindi di avere gli stati 0,1,2,3 , con 0 usato in modo Kernel.
I sistemi Unix-like richiedono soltanto 2 livelli di privilegio.



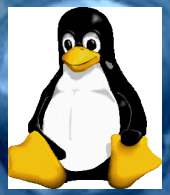
Passaggio tra User Mode a Kernel Mode

- quando avviene il "salto" tra uno e l'altro?
  - Quando viene chiamata una System Call il task volontariamente inizia ad eseguire del codice nello stato Kernel.
  - Quando arriva un IRQ (o eccezione) viene eseguito un gestore IRQ (o gestore eccezione), qundi il controllo ritorna al task interrotto come se non fosse successo niente.



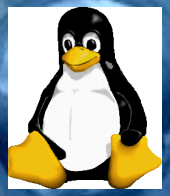
System Call

- Le System Calls sono come delle normali funzioni, soltanto che operano in Kernel Mode per eseguire operazioni sull'OS (in effetti le System Calls sono parte integrante dell'OS).
- Una System Call può essere chiamata quando:
  - si deve accedere ad un device di I/O device o ad un file (come le SC read e write)
  - è richiesto un elevato livello di privilegio per accedere ad alcune informazioni riservate (come il pid, o per cambiare la politica di scheduling e così via)
  - è richiesto un cambiamento di contesto esecutivo (come eseguire la "fork" o eseguire un'altra applicazione con la SC "exec").
  - si deve eseguire un particolare tipo di comando (come "chdir", "kill", "brk", o "signal")



La Console

- Console Virtuali
- Completamento automatico dei path
- Redirezionamenti input e output

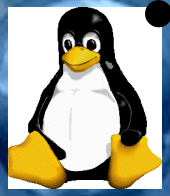


Caratteristiche Principali

NETLINK

NETLINK S.p.A.

- Ambiente multipiattaforma
- Elevata stabilità di funzionamento
- Elevata capacità di supporto del carico di lavoro (Applicazioni Server)
- Elevate prestazioni
- Completo supporto per il networking
- Evoluto supporto per la programmazione
- Ambiente grafico avanzato
- Tool e applicativi nativi per internet

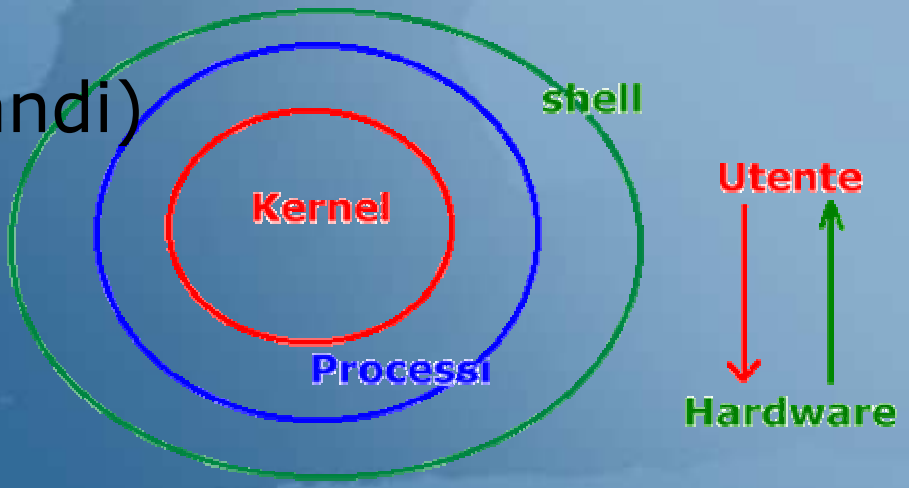


Struttura di Linux

NETLINK

LINUX S.p.A.

- Linux è un sistema operativo multitasking a 32 bit molto completo e flessibile.
- La sua struttura è nucleare
  - Kernel
  - Processi (comandi)
  - Shell



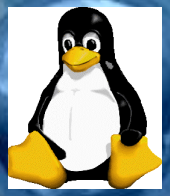
Linux e i Processi

NETLINK

NETLINK S.p.A.

Processo

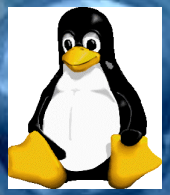
Un singolo programma, nel momento in cui viene eseguito, è un processo. La nascita di un processo, cioè l'avvio di un programma, può avvenire solo tramite una richiesta da parte di un altro processo già esistente. Si forma quindi una sorta di gerarchia dei processi organizzata ad albero. Il processo principale (root) che genera tutti gli altri, è quello del **programma init** che a sua volta è attivato direttamente dal kernel.



La Tabella dei Processi

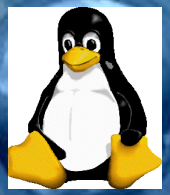
Il kernel gestisce una tabella dei processi che serve a tenere traccia del loro stato. In particolare sono registrati i valori seguenti:

- il nome dell'eseguibile in funzione
- gli eventuali argomenti passati all'eseguibile al momento dell'avvio attraverso la riga di comando
- il numero di identificazione del processo
- il numero di identificazione del processo che ha generato quello a cui si fa riferimento
- il nome del dispositivo di comunicazione se il processo è controllato da un terminale
- il numero di identificazione dell'utente
- il numero di identificazione del gruppo.



Nascita e morte di un processo

- L’avvio di un programma con la conseguente nascita di un processo avviene solo tramite la richiesta di un processo già esistente, utilizzando la chiamata di sistema **fork()**
- La morte del processo avviene invece con la chiamata di sistema **exit (),** che rende un processo *Zombie* le cui tracce saranno eliminate dal programma che lo ha generato.

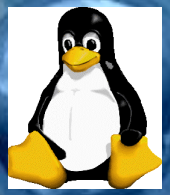


I Segnali

NETLINK

NETLINK S.p.A.

I segnali sono dei messaggi elementari che **possono essere inviati a un processo**, permettendo a questo di essere informato di una condizione particolare che si è manifestata e di potersi uniformare.



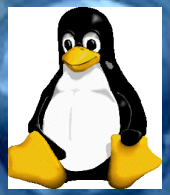
Tipologie di Segnali

NETLINK

NETLINK S.p.A.

Segue un breve elenco dei segnali più importanti:

- SIGINT - Segnale di Interruzione
- SIGTERM - Segnale di Conclusione
- SIGKILL - Segnale di “Morte” o di Fine
- SIGHUP - Segnale di Aggancio



Status dei Processi

- ps -lxw

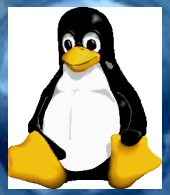
```
F   UID   PID  PPID PRI  NI    VSZ   RSS WCHAN  STAT TTY        TIME COMMAND
100   0     1     0   8   0    416   212 136174 S    ?          0:04 init [3]
040   0     2     1   9   0      0     0 11b665 SW   ?          0:00 [keventd]
040   0     3     0  19  19      0     0 114e52 SWN  ?          0:00 [ksoftirqd_CPU0]
040   0     4     0   9   0      0     0 124a76 SW   ?          0:00 [kswapd]
040   0     6     0   9   0      0     0 12d82a SW   ?          0:00 [kupdated]
040   0    74     1   9   0      0     0 836cee SW   ?          0:00 [eth0]
040   0    86     1   9   0   1792   796 136174 S    ?          0:01 /usr/sbin/syslogd
```

- ps -aux

```
USER       PID %CPU %MEM   VSZ  RSS TTY      STAT START   TIME COMMAND
root         1  0.0  0.2   416  212 ?        S    11:15   0:04 init [3]
bin         80  0.0  0.7  1388  672 ?        S    11:15   0:00 /sbin/rpc.portmap
root        86  0.0  0.8  1792  796 ?        S    11:15   0:01 /usr/sbin/syslogd
root        89  0.0  0.6  1304  584 ?        S    11:15   0:00 /usr/sbin/klogd -
daemon     104  0.0  0.6  1360  652 ?        S    11:15   0:00 /usr/sbin/atd -b
root       111  0.0  1.4  2900 1408 ?        S    11:15   0:00 sendmail: accepti
root       119  0.0  0.5  1264  504 ttyS0    S    11:15   0:00 gpm -m /dev/mouse
nobody    7821  0.0  3.4  5968 3228 ?        S    14:01   0:00 /usr/local/apache
nobody    7822  0.0  5.1  7236 4868 ?        S    14:01   0:00 /usr/local/apache
nobody    7823  0.1  5.0  7072 4736 ?        S    14:01   0:00 /usr/local/apache
nobody    7824  0.0  3.2  5836 3092 ?        S    14:01   0:00 /usr/local/apache
```

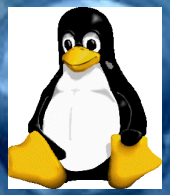
Intestazioni dello Status

- **UID Il numero di UID e quindi l'utente proprietario del processo.**
- **PID Il numero del processo, cioè il PID.**
- **PPID Il PID del processo genitore (quello da cui ha avuto origine).**
- **USER Il nome dell'utente proprietario del processo.**
- **PRI La priorità del processo.**
- **NI Il valore *nice*.**
- **SIZE La dimensione dell'immagine del processo in memoria (virtuale).**
- **RSS La dimensione della memoria**
- **RAM effettivamente utilizzata. SWAP La dimensione della memoria di *swap* utilizzata.**
- **SHARE La quantità di memoria condivisa utilizzata dal processo.**
- **WCHAN L'evento per cui il processo è in attesa.**
- **STAT Lo stato del processo.**
- **TT Il terminale, se il processo ne utilizza uno.**
- **TIME Il tempo totale di utilizzo della CPU.**
- **CTIME Il tempo di CPU sommando anche l'utilizzo da parte dei processi figli.**
- **COMMAND Il comando utilizzato per avviare il processo.**



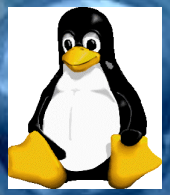
Interpretazione dello stato

- La colonna STAT indica lo stato del processo:
  - R: processo in funzione residente in memoria
  - S: processo in pausa o dormiente
  - D: processo in pausa non interrompibile
  - T: processo sospeso
  - Z: processo zombie
  - W: processo che non utilizza memoria
  - N: processo rallentato con nice positivo



Scheduling e Priorità

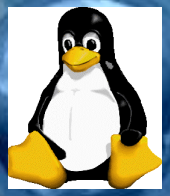
- La gestione simultanea dei processi è ottenuta normalmente attraverso la suddivisione del tempo di CPU, in maniera tale che a turno ogni processo abbia a disposizione un breve intervallo di tempo di elaborazione. Il modo con cui vengono regolati questi turni è lo ***scheduling***, ovvero la pianificazione di questi processi.
- La maggiore o minore percentuale di tempo di CPU che può avere un processo è regolata dalla priorità espressa da un numero. Il numero che rappresenta una priorità deve essere visto al contrario di come si è abituati di solito: **un valore elevato rappresenta una bassa priorità**, cioè meno tempo a disposizione, **mentre un valore basso (o negativo) rappresenta una priorità elevata**, cioè più tempo a disposizione.



Linux e l'ambiente grafico

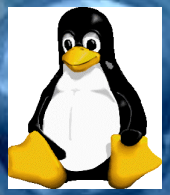
- Linux funziona in grafica utilizzando la logica del Client-Server
  - X Server
  - Client: fwm, olwm, fmw98
  - Nuovi: Gnome, Kde

- Riferimenti du http://www.xfree.org



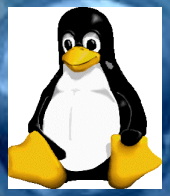
Utilizzo di Linux

- Workstation per programmatori e workstation grafica
- Server Web, Posta e DNS
- Host Internet
- Client LAN manager
- File e Print Server
- Application server



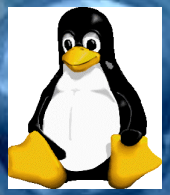
Documentazione

- HOW-TO
  - *http://metalab.unc.edu/pub/Linux/docs/HOWTO*
- Lista degli HOWTO e dei mini HOWTO tradotti in italiano
  - *http://www.pluto.linux.it/ildp/HOWTO/HOWTO-INDEX-3.html*



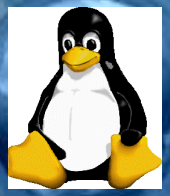
Installazione

- Linux ed i Dischi di partenza:
  - Boot Disk
  - Root Disk
  - Rescue Disk
- Setup
  - partizionamento
  - installazione pacchetti
  - configurazione



Avvio del sistema Linux

- Il boot è il modo con cui un sistema operativo può essere avviato quando l'elaboratore viene acceso:
  - LiLO
  - Loadlin
  - Grub



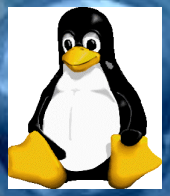
Il LiLO (Linux Loader)

NETLINK

NETLINK S.p.A.

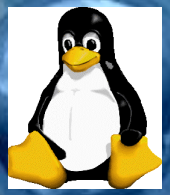
Consente l'avvio di un sistema operativo GNU/linux o anche di altri sistemi (dual boot):

- */etc/lilo.conf*
- *lilo*
- *liloconfig*
- directory /boot/ contiene i file utilizzati per effettuare l'avvio del sistema



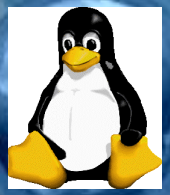
I Comandi

- Comandi Utente
- Comandi di Amministrazione del sistema
- Comandi di Amministrazione del filesystem
- Comandi di Rete



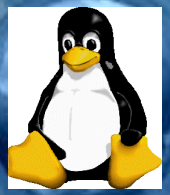
I Servizi

- Procolli di rete
  - tpc, udp e icmp
- Servizi di rete
  - ftp
  - telnet
  - sendmail
  - samba
  - ssh



La Rete

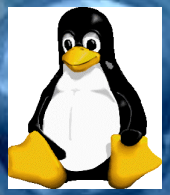
- Livelli ISO OSI
  - Lo Standard ISO-OSI descrive un'architettura di rete con i seguenti livelli:
    - 1. Livello Fisico (esempi: PPP ed Ethernet)
    - 2. Livello Data-link (esempi: PPP ed Ethernet)
    - 3. Livello di Rete (esempi: IP, e X.25)
    - 4. Livello di Transporto (esempi: TCP, UDP)
    - 5. Livello di Sessione (esempio: SSL)
    - 6. Livello di Presentazione (esempio: codifica binary-ascii sul protocollo FTP)
    - 7. Livello Applicazione (esempio: Netscape)
- I primi 2 livelli sono di solito implementati in hardware mentre i livelli successivi in software (o in firmware per i routers).
- Un OS è capace di gestire molti protocolli: uno di questi e' il TCP/IP (il più importante sui livelli 3-4).



Il Networking del sistema

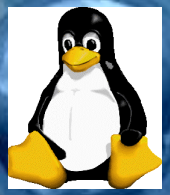
NETLINK

- Ifconfig e route
  - Permettono la configurazione del device di rete
- netstat
  - Permette di visualizzare le connessioni di rete, le tabelle di routing e le statistiche relative alle interfacce di rete del sistema.
- fuser
  - Permette di identificare quale processo è in ascolto su una determinata porta
    - root@glock:~# fuser -n tcp 113
    - 113/tcp: 414
    - ps ax |grep 414
    - 414 ? S 0:00 /usr/sbin/inetd
- lsof
  - Mostra informazioni relative a file o directory utilizzati da determinati processi
    - lsof /dev/log
    - COMMAND PID USER FD TYPE DEVICE SIZE NODE NAME
    - syslogd 60 root 0u unix 0xcf6fd560 48 /dev/log
    - lsof -i@192.168.17.150
    - COMMAND PID USER FD TYPE DEVICE SIZE NODE NAME
    - Xvnc 1801 root 5u IPv4 6677 TCP slack:5901->client01:4021 (ESTABLISHED)
    - sshd 2594 root 4u IPv4 25320 TCP slack:ssh->client01:4264 (ESTABLISHED)
- Ping e Traceroute
  - Strumenti standard per testare il transito dei pacchetti in rete



Kernel e rete: RX Level

- In RX l'OS:
  - Gestisce il dialogo a basso livello con i devices (come schede ethernet o modem) ricevendo i pacchetti dall'hardware,
  - Costruisce "pacchetti" TCP/IP partendo da "frames" (come Ethernet o PPP),
  - Converte i "pacchetti in "sockets" passandoli al giusto applicativo (grazie al numero di porta) oppure
  - Instrada i "pacchetti" nella giusta coda
    - frames pacchetti socketsNIC ---------> Kernel ----------> Applicazione | pacchetti ----------> Instradamento - RX -

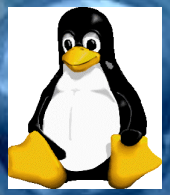


Kernel e rete: TX Level

NETLINK

NETLINK.S.p.A.

- Nello stadio di TX l'OS:
  - Converte i "sockets" oppure
  - I dati accodati in "pacchetti" TCP/IP
  - Espande i "pacchetti" in "frames" (come Ethernet o PPP)
  - Manda i "frames" utilizzando i devices Hardware
    - sockets packets framesApplication ---------> Kernel ----------> NIC packets /|\ Forward ---------- ------ - TX -

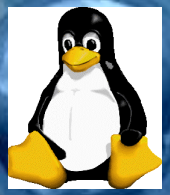


La Sicurezza: Hardware

NETLINK

NETLINK.S.a.S.

- **Difese Attive o Preventive** : sono quelle che hanno come obiettivo quello di rendere il sistema inarrestabile:
  - Usare, se possibile, hardware certificato in ogni sua parte
  - Utilizzare i driver più aggiornati (usare internet per ricerca)
  - Utilizzare dischi ad architettura SCSI se possibile implementare architetture RAID Hardware se possibile utilizzare (in casi eccezionali) strutture tipo cluster
  - Utilizzare un gruppo di continuità

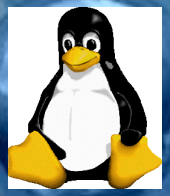


La Sicurezza: Hardware

NETLINK

NETLINK s.a.s.

- **Difese Passive o post crash** : sono quelle che hanno come obiettivo quello di premettere un rapido e totale ripristino del sistema in caso di crash:
  - installare un unità di backup (DAT) su ogni sistema se possibile.



La Sicurezza: Software

NETLINK

NETLINK S.r.l.

- Scelta Distribuzione
- Partizionamento dischi
- Installazione Pacchetti
- Configurazione
  - Controllo Check list
  - Non installare software inutili
  - Disabilitare i servizi inutili
  - Eseguire portscan e test per la sicurezza
  - Controllo account e permission
  - Aggiornare software e kernel alle ultime release
  - Wrappare i servizi

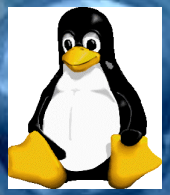
» continua……

La Sicurezza: Software

NETLINK

NETLINK S.p.A.

- Restrizione dell'accesso
- Livelli di Sicurezza: Linux C2
- Auditing del Sistema
- Crittografia (Dati e Filesystem)
- Firewalling
  - ipchains
  - iptables e netfilter
- Analisi debolozze del sistema
- Contromisure per possibili attacchi



La Sicurezza: Filesystem

NETLINK

NETLINK S.a.S.

- Eliminazione dei bit SUID e SGID negli script di shell (anche se questo non dovrebbe causare problemi con Linux).
- Verifica di tutti i programmi che hanno il bit SUID o SGID attivato, a meno di quelli che notoriamente devono avere questo privilegio.
- Verifica della presenza del bit sticky nelle directory che sono accessibili in scrittura da tutti gli utenti.
- Verifica del valore di umask dell'account root.
- Verifica dei permessi dei file di dispositivo.