Laboratorio di Amministratore di Sistema

7. Procedure avanzate

[Cisco ITESS II - Chapter 11]

Università di Venezia – Facoltà di Informatica
feb-mag 2012 - A. Memo

Advanced NOS Administration

11.1 Backups
11.2 Drive Mapping
11.3 Partition and Processes Management
11.4 Monitoring Resources
11.5 Analyzing and Optimizing Network Performance

Overview of Backup Methods

- The backup process involves copying data from one computer to some other reliable storage medium for safekeeping.
- Once the data has been archived, the system administrator can then restore data to the system from any previously recorded backup.
- Considerations that are relevant for storage devices:
  - Cost
  - Size
  - Manageability
  - Reliability

CD-RW Drive
NO631101-1

HP ex2 Hard drive
NO631101-2

HP Superstore Ultrium 215
NO631101-3

Zmanda Cloud Backup for Windows

Zmanda Cloud Backup (ZCB) is a radically simple-to-use and cost-effective backup and disaster recovery solution. ZCB backs up Windows servers, desktops and live applications such as Microsoft Exchange and SQL Server to Amazon's highly dependable online storage.

LTO ULTRIUM 3, capacità di 800 GB, fattore di compressione 2:1. Alta velocità di trasferimento: 40-80 MB / sec compressione 2:1; nativo 20-40MB/sec.

Backup (1)

- Per backup si intende sia la copia dei file che dei dati
- Il backup serve a ripristinare i dati, una volta che siano andati persi
- La perdita dei dati può avvenire per
  - cause legate agli operatori, persone fisiche (80%)
  - cause tecniche (14%)
  - cause ambientali (6%)

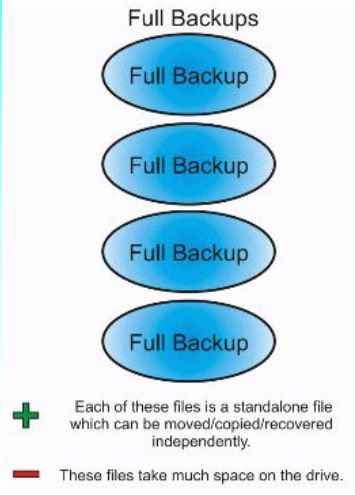
Backup (2)

- Fattori da prendere in esame per decidere la strategia di backup ottimale:
  - quali file devono far parte del backup
  - backup di rete o locale
  - frequenza del backup
  - quando effettuare il backup
  - che metodi di backup attuare
  - che tecnologie adottare

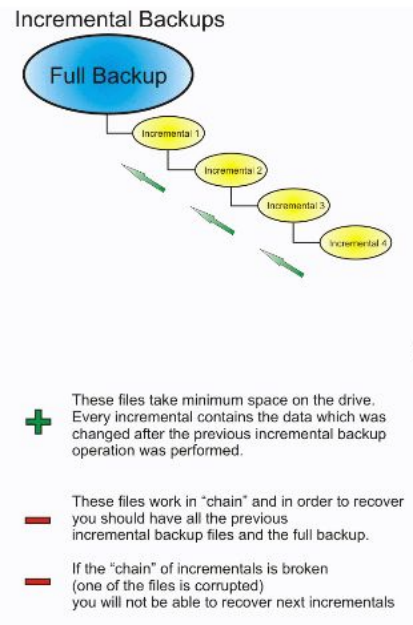
Overview of Backup Methods

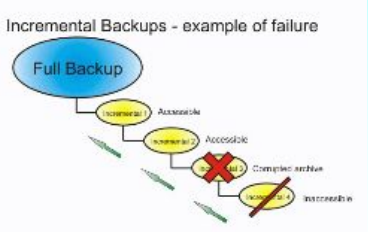
- Commonly used backup devices include tape drives, removable disk drives, recordable compact disc, HDs
- There are four types of backup procedures that define how the backup will take place:
  - **Full** - will backup everything on the hard drive at the scheduled point in the day (daily)
  - **Partial** - backs up selected files (daily)
  - **Incremental** - only the files that have changed since the last backup will be selected for back up
  - **Differential** - backs up files created or changed since the last normal or incremental backup

Backup Normale

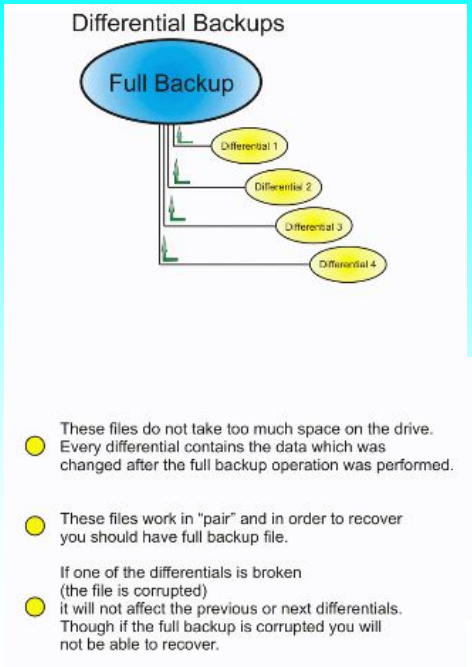


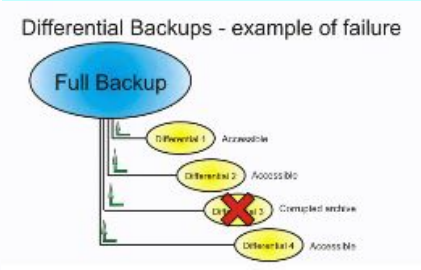
Backup Incrementale



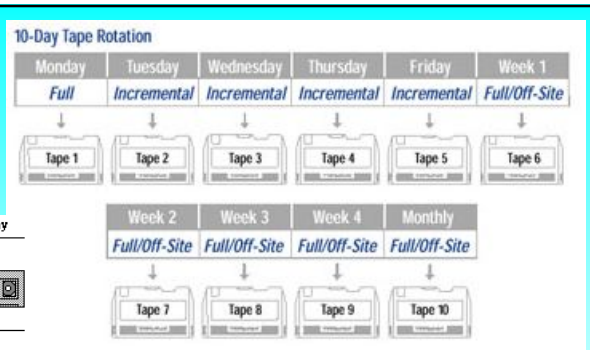


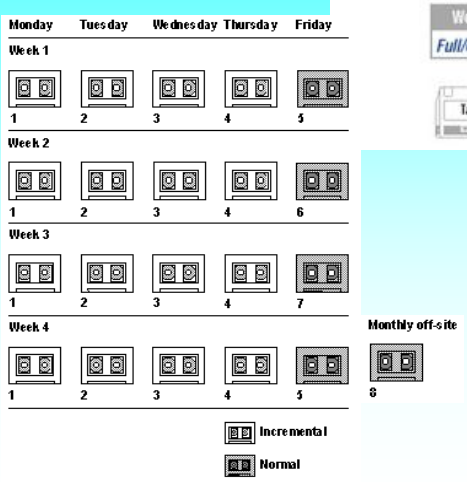
Backup Differenziale





Strategie di backup





Un esempio di backup on the cloud

Ecco come il back up dei dati aiuta le aziende di medie dimensioni a essere più competitive.

Oggi le medie aziende devono gestire volumi d'informazioni che crescono e si moltiplicano velocemente. Non possono permettersi di perdere dati preziosi su cui basano tutte le operazioni di business. Con budget sempre più ristretti e minori risorse a disposizione per molte aziende è difficile mantenere all'interno un sistema di back up e recovery affidabile. Ecco perché IBM e i suoi Business Partner stanno utilizzando la potenza del cloud computing per offrire alle medie aziende la stessa protezione dei dati che usano le grandi imprese: un servizio studiato per ridurre i rischi e rispondere alle loro esigenze di budget.

E' il nuovo Servizio di back up basato sul cloud di IBM. Si avvale di un'infrastruttura intelligente che effettua un salvataggio dei dati in uno dei data center IBM. Nel momento in cui i dati vengono salvati, sono protetti e al sicuro. Sempre e ovunque.

*il prezzo base annuale del servizio per un pacchetto "small" da 100GB è a partire da 2844,00 (importo per ogni GB mensile aggiuntivo 2,37). Tutti i prezzi riportati sono indicativi ed IVA esclusa, aggiornati al momento di andare in stampa. IBM si riserva il diritto di modificarli e di modificare anche le specifiche relative ai prodotti. Prodotti, programmi e servizi possono essere ritirati da IBM senza preavviso. IBM, il logo IBM, ibm.com e l'icona del pianeta sono marchi registrati di International Business Machines Corporation in diversi Paesi del mondo. La lista aggiornata dei marchi registrati di IBM è disponibile sul sito www.ibm.com/legal/copytrade.shtml, alla voce "Copyright and trademark information". ©2011 IBM Corp. Tutti i diritti riservati.

1. Riduci i costi complessivi e di gestione fino al 40%.
Il tuo Business Partner IBM può aiutarti a confrontare i costi dei sistemi interni alla tua azienda con quelli di un servizio scalabile basato sul cloud e gestito da IBM evitando di impegnare capitali.

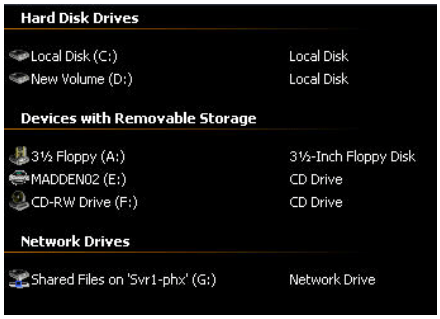
2. Un backup automatizzato e più sicuro.
Il backup viene effettuato automaticamente nella finestra temporale stabilita.

3. Hai quello che ti serve, quando ti serve.
Nel cloud i tuoi dati sono custoditi in più versioni, subito disponibili, così puoi prendere più velocemente decisioni più efficaci.

4. Libera risorse preziose.
Il 95% del risparmio sui costi in azienda deriva dalla riduzione di hardware, software e infrastruttura di backup. E il servizio di backup ti consente di indirizzare il personale IT verso iniziative più strategiche.

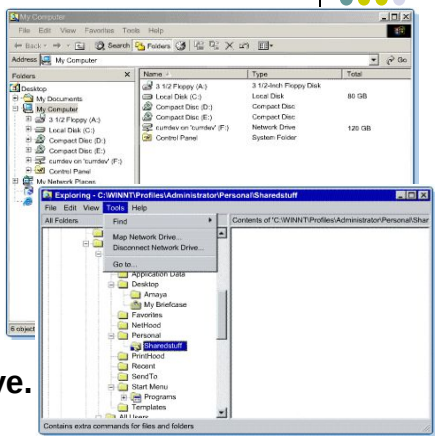
What is Drive Mapping?

- Drive mapping is a useful tool that allows an administrator to share resources that are stored on a server.
- Requires two steps:
  - defining the path to the resource
  - assigning a drive letter
- The client computers that are connected to the network assign a drive letter that will act as a direct path to access those resources stored on a server over the network.
- After a user identifies a network resource to be used locally, the resource can be "mapped" as a drive.

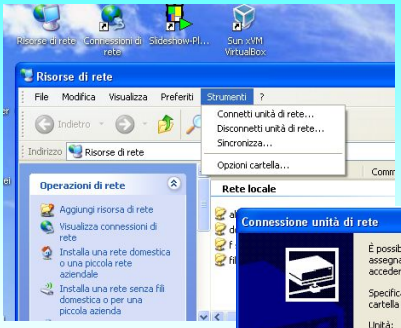


Mapping Drives in Windows Networks

- To map a drive with Windows Explorer, navigate to the folder on the remote system in Windows Explorer by selecting **Network** > **Neighborhood** > **Server name** > **Shared folder name**.
- Another way to do this is to choose the **Tools** menu, and then choose **Map Network Drive.**



Connessione unità di rete



Mapping Drives in Windows Networks

- Instead of mapping drives through Windows Explorer, the **net use** command can be used.

  net use *drive_letter:* *computer_name**share_name* [**/user:***utente password*]

  net use z: \\Server01\Data
  net use z: \\Server01\Data /user:Bob passBob
  net use z: /delete

- **net use** can also be incorporated into a login script that automatically runs when the user logs in to the network.

Mapping Drives in Linux Networks

To map a drive to a Linux server:

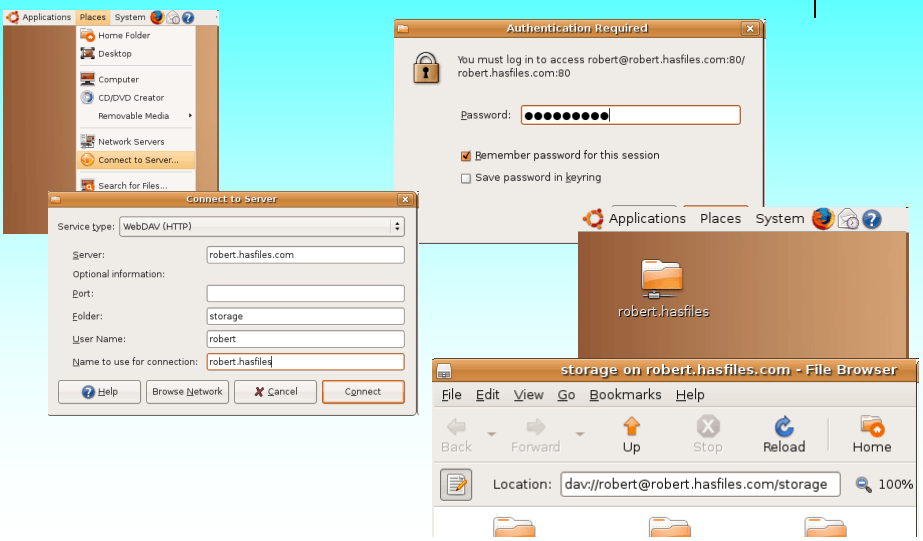
- with a Windows client: need the Samba daemon loaded
- with a Linux client, use the **mount** command to establish a connection to the shared directory on the server.

```
# mount //servername/sharename /localdirectory
```

- The local directory designation that points to the remote share denoted by the first part of the command is called the directory mount point.
- The mount point location must already exist before a share can be mapped to it.

```
# mkdir /localdirectory
```

Accesso remoto via HTTP



Partitions
Using fdisk, mkfs, and fsck

- **fdisk** is a text-based command and requires the use of one-letter commands to manipulate the options.

  fdisk /dev/hda2

  → p → … → w

Option	Description
d	Deletes a partition
n	Creates a new partition
p	Prints or displays the partition layout
q	Ends the session without saving any changes
t	Changes a partitions type code
w	Saves the changes made and quits

- Once the partition changes have been made, a filesystem must be created on the partition.
- This is also referred to as formatting the partition.

Fdisk in Windows (old)

```
Microsoft Windows 98
Programma di impostazione del disco rigido
(C)Copyright Microsoft Corp. 1983 - 1998

Opzioni di FDISK

Unità disco rigido corrente: 1

Scegliere una delle seguenti opzioni:

1. Crea partizione o unità logica DOS
2. Imposta partizione attiva
3. Elimina partizione o unità logica DOS
4. Visualizza informazioni sulla partizione
5. Cambia l'unità disco rigido corrente

Digitare il numero della selezione: [1]

Premere Esc per uscire da FDISK
```

```
Visualizza informazioni sulla partizione

Unità disco rigido corrente: 1

Partizione Stato   Tipo    Etichetta    Mbyte   Sistema    Uso
 C: 1        A     PRI DOS   HDD C       8746   FAT32     100%

Lo spazio su disco totale è pari a  8746 MB (1 MB = 1048576 byte)
```

fdisk in Ubuntu

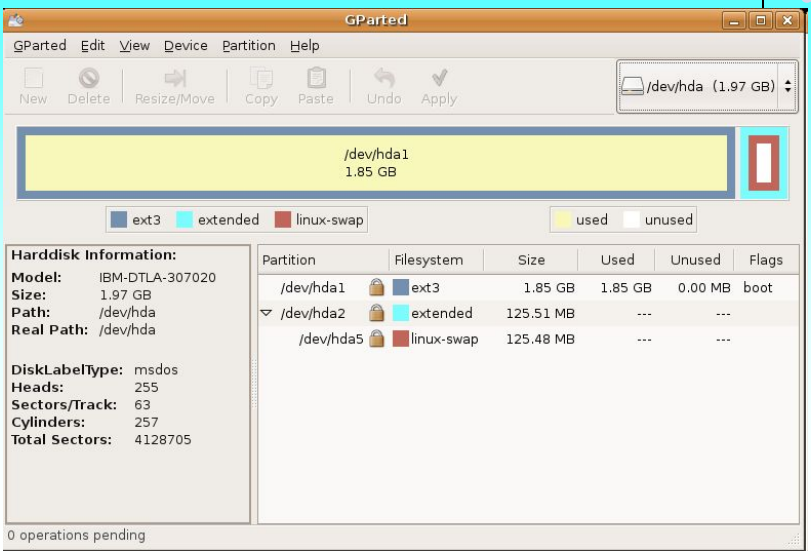
```
Disk /dev/sda: 400.0 GB, 400088457216 bytes
255 heads, 63 sectors/track, 48641 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Disk identifier: 0x43af43af

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1   *           1       48641   390708801    7  HPFS/NTFS

Disk /dev/sdb: 1000.2 GB, 1000204886016 bytes
255 heads, 63 sectors/track, 121601 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Disk identifier: 0xb62f5470

   Device Boot      Start         End      Blocks   Id  System
/dev/sdb1   *           1       95977   770935221    7  HPFS/NTFS
/dev/sdb2           95978      108725   102398310    f  W95 Ext'd (LBA)
/dev/sdb3          108726      121478   102438472+  83  Linux
/dev/sdb4          121479      121601      987997+  82  Linux swap / Solaris
/dev/sdb5           95978      108725   102398278+   7  HPFS/NTFS
```

Gparted di Ubuntu



Partizioni tipiche in Ubuntu

Ubuntu da solo
/dev/hda1 Linux
/dev/hda2 Estesa
/dev/hda5 Linux swap

partizione primaria

partizione estesa

partizione logica

IDE master controller primario

Windows + Ubuntu
/dev/hda1 HPFS/NTFS
/dev/hda2 Estesa
/dev/hda3 Linux
/dev/hda5 Linux swap

Using fdisk, mkfs, and fsck

- Use the **mkfs** utility to create a filesystem in Linux.

  mkfs [-V] [-t fstype] [options] device [blocks]

- Once the partition changes have been made, a filesystem must be created on the partition.
- This is also referred to as formatting the partition.

Option	Description
`-V`	Adding this option to the command will display additional output at the filesystem is created.
`-t fstype`	This option allows the user to specify the filesystem type that will be created. The `fstype` would be replaced with something like `ext3` for an `ext3` filesystem, or `msdos` for a `FAT` filesystem, for example.
`Options`	This parameter is used to specify options specific to the particular filesystem.
`device`	This parameter specifies the device on which the filesystem was created. Usually it will be the same parameter used with the `fdisk` command.
`blocks`	This parameter specifies the size of the filesystems blocks (usually 1024 bytes in size). This value will not always need to be used, because the block size can be determined from the size of the partition.

Using fdisk, mkfs, and fsck

- The **fsck** utility is used to check file systems and repair damaged files.

  fsck [-A] [-V] [-t *fs_type*] [-a] [-l] [-r] [-s] *filesystem*

  #fsck –t ext2 /dev/hda1

- A good practice is to unmount a file system before checking it.
- To check the root file system, you should boot from a recovery/setup floppy.
- Use this utility often to check for file system integrity.
- If fsck makes any changes, reboot your system immediately.

Option	Description
`-A`	This parameter specifies that all files systems marked in `/etc/fstab` will be checked.
`-C`	This parameter will display a text-mode progress indicator while the file system is being checked.
`-V`	This will produce the same output for this command as with the `mkfs` utility.
`-N`	This parameter will display the results of what `fsck` would do, but not actually doing it.
`-fsck- options`	This parameter is used to specify filesystem check options that fsck cannot interpret. Examples are `-a` or `-p`, which perform and automatically check, `-r`, which performs an interactive check, or `-f`, which forces a full system check.
`filesystems`	Specifies the filesystem that is being checked.

Managing System Processes with cron Jobs

- The way to schedule tasks to run at regular intervals on a Linux system is with **Cron Programs**.
- Also known as Cron jobs, they schedule system maintenance tasks that are performed automatically (ex: empties /tmp directory).
- Cron is controlled by the entries in the **/etc/spool/cron** and **/etc/cron.d** directories and **/etc/crontab** file.
- Cron is not a command, but rather it is a daemon that runs once every minute, scans the conf iguration files, and performs the tasks specified.
- There are two types of Cron jobs: **System Cron** jobs and **User Cron** jobs.

Managing System Processes with cron Jobs

The /etc/crontab File

```
SHELL=/bin/bash
PATH=/sbin:/bin:/usr/sbin:/usr/bin
MAILTO=root
HOME=/

#run-parts
01 * * * * root run-parts /etc/cron.hourly
02 * * * * root run-parts /etc/cron.daily
22 * * * * root run-parts /etc/cron.weekly
42 * * * * root run-parts /etc/cron.monthly

0-59/5 * * * * root /usr/bin/mrtg /etc/mrtg/mrtg.cfg
```

- To create a System Cron Job, you have to modify the **/etc/crontab** file.
- The file begins with set of environmental variables. These set some parameters for the Cron jobs such as the PATH and MAILTO
- The other lines in this file specify the minute, hour, day, month, and day of the week the job will run (24h format).
- [*] = all, [x-y] = from x to y, [/x] = every x min, [x,y] = at x and y min.s

```
0   8   *    *    *   root echo "Good morning!!!"
/5  *   15   *    *   root /bin/ls /var/log>/temp/ls.out
```

1. minute 0-59
2. hour 0-23
3. day of month 1-31
4. month 1-12
5. day of week 0-7
6. owner
7. the command to be run

** may be used as everyone, range (0-4,8-12) and list (1,2,5,9) are allowed*

Altra possibilità di System Cron Job

Altra possibilità è la creazione di uno script (sh, bash, perl o altro) per l'esecuzione dell'operazione desiderata. Tale script andrà inserito in una delle directory seguenti

/etc/cron. hourly
/etc/cron. daily
/etc/cron.monthly
/etc/cron.weekly

per poter essere eseguito ad intervalli ben definiti (se non specificato, alle 4:00)

Managing System Processes with cron Jobs

- To create a User Cron Job, you have to use the **crontab** utility.

crontab [-u user] [-l|-e|-r] [*file*]

- If a user is not specified, the User Cron job will be created for the current user.
- *file* is the file containing the crontab commands with the same syntax used for a System Cron job.

crontab -u jsmith myTest.cron

```
# /home/jsmith/myTest.cron
# User Cron Job Test
MAILTO=jsmith@localhost
* * * * * jsmith echo "How are you?"
```

Managing System Processes with cron Jobs

- The **at** command is similar to using cron, at the time and/or date specified by the command.

```
at 1:23
lp /home/rtalbot/thesis/*
echo "Your files were printed" | mail –s"Complete" boss
```

Return

Ctrl-d

after termination

```
Job 55842688.a at Thurs June 17 01:23:00 2004
```

to cancel the job

```
at –d 55842688.a
```

- For several commands, it's best to put them in a file

```
at 8:00 –f scheduledJobs
```

Managing System Processes with at command

Managing System Processes with Cron Jobs

Format	Action
`at hh:mm`	Schedules job at the hour (hh) and minute (mm) specified,using a 24-hour clock.
`at hh:mm month day year`	Schedules job at the hour (hh) minute (mm), month, day,and year specified.
`at -l`	Lists scheduled jobs; an alias for the atq command.
`at now +count time-units`	Schedules the job right now plus count number of time-units; time units can be minutes, hours, days, or weeks.
`at -d job_id`	Cancels the job with the job number matching job_id; analias for the atrm command.

Core Dumps

- **Core Dump** is a recording of the memory that a program was using at the time it crashed.
- The purpose of Core Dumps is to allow programmers to study the file to figure out exactly what caused the program to crash.
- To locate the Core files (not only) on a Linux system:

  # find / -name core

- The main properties of the Core file are:
  - the Owner of the file states who executed the program
  - the Creation Date of the core file is the date at which the crash occurred and when the Core Dump was created
  - the Creating Program properties of core file will tell which program crashed and generated the Core Dump file (use gdb)

arpum - PuTTY

```
I A  hello.c
#include <stdio.h>

int main()
{
        int* pointer = NULL;
        puts("Hello world");
        *pointer = 42;
        return 0;
}
File hello.c saved
```

arpum - PuTTY

```
berkes@arpum:~$ ulimit -c unlimited
berkes@arpum:~$ gcc -o hello -g -Wall hello.c
berkes@arpum:~$ ./hello
Hello world
Segmentation fault (core dumped)
berkes@arpum:~$ ls -l core*
-rw-------    1 berkes   users       65536 Jan 25 17:38 core
berkes@arpum:~$
```

arpum - PuTTY

```
Segmentation fault (core dumped)
berkes@arpum:~$ ls -l core*
-rw-------    1 berkes   users       65536 Jan 25 18:21 core
berkes@arpum:~$ gdb hello core
Core was generated by `./hello'.
Program terminated with signal 11, Segmentation fault.
Reading symbols from /lib/libc.so.6...done.
Loaded symbols for /lib/libc.so.6
Reading symbols from /lib/ld-linux.so.2...done.
Loaded symbols for /lib/ld-linux.so.2
#0  0x0804834e in main () at hello.c:7
7               *pointer = 42;
(gdb) bt
#0  0x0804834e in main () at hello.c:7
#1  0x4002ebb4 in __libc_start_main () from /lib/libc.so.6
(gdb) quit
berkes@arpum:~$
```

core.3737

Overview | [threaddump] 02:54:31 PM | [heapdump] 02:54:39 PM

core.3737

Thread Dump

```
Thread 3802 "Timer-5": (state = BLOCKED)
        at java.lang.Object.wait(Native Method)
        - waiting on <0xaa1266c8> (a java.util.TaskQueue)
        at java.util.TimerThread.mainLoop(Timer.java:509)
        - locked <0xaa1266c8> (a java.util.TaskQueue)
        at java.util.TimerThread.run(Timer.java:462)

Thread 3740 "DestroyJavaVM": (state = BLOCKED)

Thread 3801 "Timer-4": (state = BLOCKED)
        at java.lang.Object.wait(Native Method)
        - waiting on <0xaa1216b0> (a java.util.TaskQueue)
        at java.util.TimerThread.mainLoop(Timer.java:509)
        - locked <0xaa1216b0> (a java.util.TaskQueue)
        at java.util.TimerThread.run(Timer.java:462)

Thread 3800 "pool-1-thread-3": (state = BLOCKED)
        at sun.misc.Unsafe.park(Native Method)
        at java.util.concurrent.locks.LockSupport.parkNanos(LockSupport.java:198)
        at java.util.concurrent.SynchronousQueue$TransferStack.awaitFulfill(SynchronousQueue.java:424)
        at java.util.concurrent.SynchronousQueue$TransferStack.transfer(SynchronousQueue.java:323)
        at java.util.concurrent.SynchronousQueue.poll(SynchronousQueue.java:874)
        at java.util.concurrent.ThreadPoolExecutor.getTask(ThreadPoolExecutor.java:944)
        at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:906)
        at java.lang.Thread.run(Thread.java:619)

Thread 3798 "ContainerBackgroundProcessor[StandardEngine[com.sun.appserv]]": (state = BLOCKED)
        at java.lang.Thread.sleep(Native Method)
        at org.apache.catalina.core.ContainerBase$ContainerBackgroundProcessor.run(ContainerBase.java:1800)
        at java.lang.Thread.run(Thread.java:619)

Thread 3797 "ContainerBackgroundProcessor[StandardEngine[com.sun.appserv].StandardHost[server].StandardContext[/__JWSappclien
        at java.lang.Thread.sleep(Native Method)
        at org.apache.catalina.core.ContainerBase$ContainerBackgroundProcessor.run(ContainerBase.java:1800)
        at java.lang.Thread.run(Thread.java:619)

Thread 3796 "ContainerBackgroundProcessor[StandardEngine[com.sun.appserv].StandardHost[server].StandardContext[/__wstx-servic
        at java.lang.Thread.sleep(Native Method)
        at org.apache.catalina.core.ContainerBase$ContainerBackgroundProcessor.run(ContainerBase.java:1800)
        at java.lang.Thread.run(Thread.java:619)

Thread 3794 "ContainerBackgroundProcessor[StandardEngine[com.sun.appserv].StandardHost[server].StandardContext[]]": (state =
        at java.lang.Thread.sleep(Native Method)
        at org.apache.catalina.core.ContainerBase$ContainerBackgroundProcessor.run(ContainerBase.java:1800)
```

Ubuntu core dumps

Core Dumps

- In order to effectively manage system processes on a Linux system, it is important to be able to determine what processes are running on a system and which process are critical and non-critical.
- The processes that are currently running on a Linux system can be viewed by using the **ps** command.
- Example:

  ps -A –forest

Option	Description
`-A, -e`	If the `ps` command is issued by itself, it will only display the processes that are currently running in the terminal, which doesn't provide much information. These options will cause the `ps` command to display information about all the processes that are currently running on the system. The `-A` and `-e` options will display all the process that are currently running on the system. The `-x` option will display all the processes that are being used by the user who enters the command.
`-u user`	This option will let the user display all the processes being used by a specific user. The user's username or user ID can be entered here.
`-H, -f, -forest`	These options will group processes together in a hierarchy to show the parent-to-child relationship between processes.
`-w`	By default the `ps` command shortens its output, so that it all can be displayed on the terminal screen. This option will tell the `ps` command not to do this. This is helpful when redirecting the output to a text file, which accepts wide output and can be read. To redirect the output to a text file, type use

Core Dumps

- Example: **ps -A –forest**

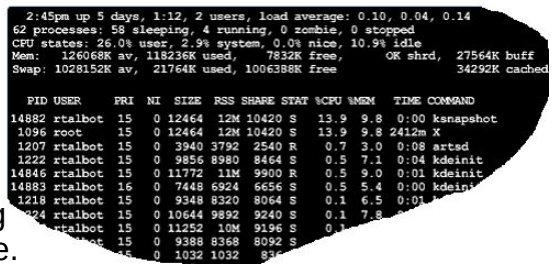
```
[rtalbot@cisco-test1 etc]$ ps -A --forest
  PID TTY          TIME CMD
    1 ?        00:00:04 init
    2 ?        00:00:00 keventd
    3 ?        00:00:00 kapmd
    4 ?        00:00:00 ksoftirqd_CPU0
    5 ?        00:00:00 kswapd
    6 ?        00:00:00 bdflush
 1086 tty5     00:00:00 mingetty
 1087 tty6     00:00:00 mingetty
 1088 ?        00:00:00 kdm
 1096 ?        1-16:12:44 \_ X
 1097 ?        00:00:00 \_ kdm
 1109 ?        00:00:00     \_ startkde
 1219 ?        00:00:00         \_ kwrapper
 1190 ?        00:00:00 kdeinit
 1207 ?        00:00:06 \_ artsd
 1222 ?        00:00:01 \_ kdeinit
 1228 ?        00:00:18 \_ autorun
11433 ?        00:00:04 \_ kdeinit
11435 pts/1    00:00:00     \_ bash
11834 pts/1    00:00:00         \_ ps
 1193 ?        00:00:00 kdeinit
```

```
# ps uxf
USER PID %CPU %MEM VSZ RSS  TTY  STAT START  TIME COMMAND
root  1343  0.0   0.5  2360 1024 pts/3  S    16:29   0:00  su –
root  1347  0.0   0.6  2448 1300 pts/3  S    16:29   0:00  - bash
```

Important Information in the `ps -A -forest`

Value	Description
`Username`	This value is not displayed in this example. However if the -u user option had been added, then the corresponding username would precede the entries.
`Process ID (PID)`	This is the process number that is used to identify the process. It is important, because it is what is used to terminate or kill a process, which will be described later.
`Parent Process ID (PPID)`	This is the same as the PID. However, it corresponds to the parent process, and the PID refers to the child process.
`TTY`	This identifies a terminal and is referred to as the Teletype. For example not all processes will have TTY-like daemons and X programs. Text-mode programs do have these numbers, and they refer to a console or remote login session.
`CPU Time`	There are two items that are of concern here, the TIME and %CPU headings. The TIME heading indicates the total amount of CPU time consumed, and the %CPU heading represents the percentage of CPU time that is currently being used when the ps command is executed. This heading can help determine what processes might be consuming too much CPU time and need to be terminated. Terminating (killing) processes will be covered in the next section.
`CPU Priority`	It is possible to give certain processes priority over other processes by restricting CPU usage. The priority of a process is given by its priority code. The default value is zero. Positive numbers represent decreased priority, and negative numbers represent increased priority.
`Memory Use`	There are a few headings that represent the process memory use. This can help identify certain processes that might be causing a system performance to decrease. For example, the Resident Set Size (RSS) heading represents the memory used by the program, and %MEM shows what percentage of memory the process is using.
`Command`	The last column represents the command that was used to launch the process.

Core Dumps



- The **top** command functions much like the Windows 2000 Performance tool by providing detailed information regarding CPU and RAM usage.
- The **kill** command can be used to terminate the process.

  # kill –s *signal pid*
- The **signal** option represents the specified signal that is sent to the process.
- There are 63 different parameters that can be entered for the **signal** that is sent to the process. Each will terminate the process in a different manner.

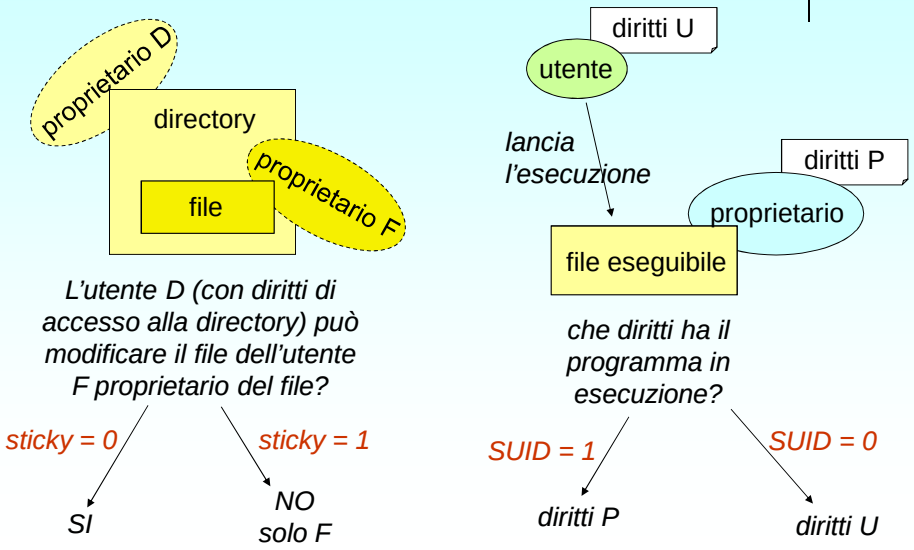
Assigning Permissions for Processes

- Typically, programs have the same types of permission and can read the same files as the user who runs that program.
- Regular users cannot execute the **su** command, because it requires root account privileges.
- Programs such as these are run using the **SUID** (Set User ID) or **SGID** (Set Group ID) bit, which allows to run the program with the permission of whoever owns the file, instead of the user who executes the program.

There are a few security risks involved when using the SUID or SGID bit to allow programs to run with the permission of the other users:

- Applying the SUID root permissions for the fdisk command could allow a user to completely erase the hard drive of the server.
- Another security risk is if there are bugs in any of the SUID or SGID programs. If these programs contain problems or bugs and they are executed by users who should not have the permission to do so, those programs could potentially cause more damage to the system than if they were executed with the normal privileges.

Sticky bit, SUID, SGID



Other command for managing processes

bg places the current job or specified job in the background

fg places the current job or specified job in the foreground.

nice run a program with modified scheduling priority; the priority range on a Linux system is -20 (most favorable scheduling) to 19 (least favorable).
[-n xxx] add xxx to the priority (default 10).
Non-root users may only alter their nice values between 0 and 20

renice change the priority of a running program
[-u user] all the user's processes will change their nice

Comandi &, jobs, bg e fg

- Per lanciare un processo e restituire il controllo alla shell, aggiungere $ alla fine del comando (background):

```
$ telnet &
```

```
[1] 6694
```

```
$ ftp &
```

```
[2] 6851
```

```
$ ps
```

```
 PID TTY        TIME CMD
6192 tty1   00:00:00 bash
6694 tty1   00:00:00 telnet
6851 tty1   00:00:00 ftp
6860 tty1   00:00:00 ps
```

```
$ jobs
```

```
[1]-  Stopped         telnet
[2]+  Stopped         ftp
```

Comandi &, jobs, bg e fg

- Per portare in foreground un processo:

```
$ fg %1        (e poi CTRL Z per sospendere il processo)
```

```
telnet
telnet>
[1]+  Stopped        telnet
```

- Per riportare in background un processo:

```
$ bg %1
```

```
telnet
telnet>
[1]+  Stopped        telnet
```

- CTRL Z per sospendere (stop, ma non quit)
- CTRL C per terminare (o altro comando del processo)

Priorità

- Eseguo un programma in background, per potergli modificare le priorità:

```
$ telnet &
```

```
[1] 8287
```

```
$ ps aux | grep telnet
```

```
1000 8287 0.0 0.0 3169 856 tty1 T  10:37 0:00 telnet
1000 8298 0.0 0.0 3004 752 tty1 R+ 10:37 0:00 grep telnet
```

```
$ renice 0 8287
```

```
8287: old priority 0, new priority 0
```

```
$ renice -1 8287
```

```
renice 8287: setpriority: Permission denied
```

Priorità

- Solo con diritti di root posso migliorare la priorità:

```
$ sudo renice -1 8287
```

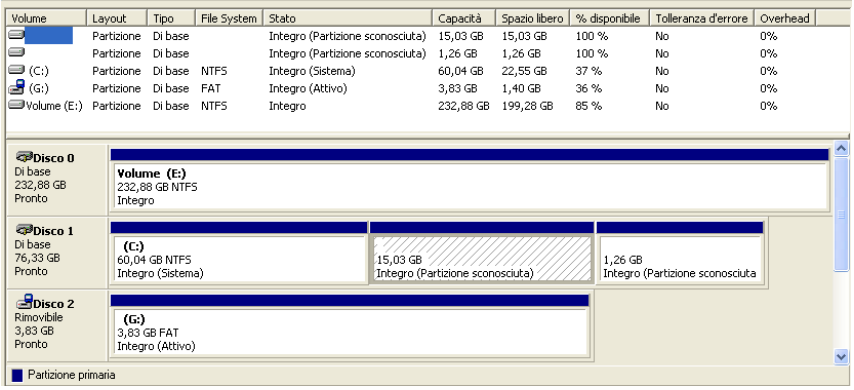
```
8287: old priority 0, new priority -1
```

```
$ ps aux | grep telnet
```

```
1000 8287 0.0 0.0 3160 856 tty1 T< 10:37 0:00 telnet
1000 8720 0.0 0.0 3004 768 tty1 S+ 10:48 0:00 grep telnet
```

Disk Management

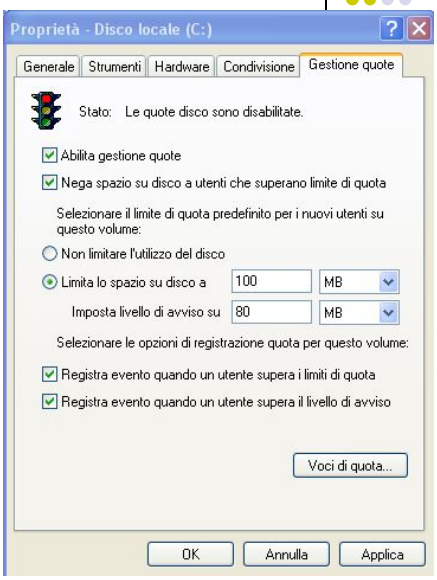
- By regularly using error-checking and defragmentation programs and continually managing free disk space, the system administrator can maintain a healthy hard drives.



Volume	Layout	Tipo	File System	Stato	Capacità	Spazio libero	% disponibile	Tolleranza d'errore	Overhead
	Partizione	Di base		Integro (Partizione sconosciuta)	15,03 GB	15,03 GB	100 %	No	0%
	Partizione	Di base		Integro (Partizione sconosciuta)	1,26 GB	1,26 GB	100 %	No	0%
(C:)	Partizione	Di base	NTFS	Integro (Sistema)	60,04 GB	22,55 GB	37 %	No	0%
(G:)	Partizione	Di base	FAT	Integro (Attivo)	3,83 GB	1,40 GB	36 %	No	0%
Volume (E:)	Partizione	Di base	NTFS	Integro	232,88 GB	199,28 GB	85 %	No	0%

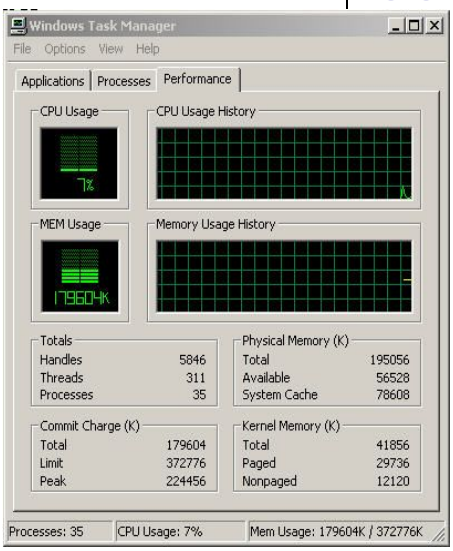
Disk Management

- One preventive disk management tool available to system administrators is the use of "quotas" for user accounts.
- A quota acts as a storage ceiling that limits the amount of data each user can store on the network.

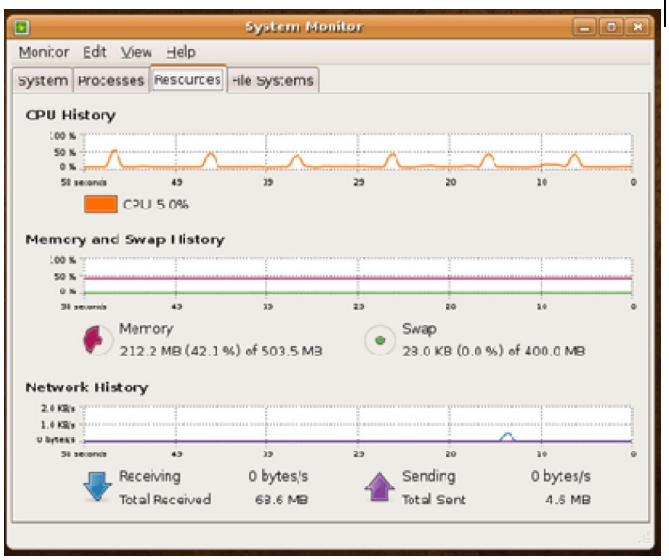


Memory Usage

- Memory diagnostic tools that allow RAM intensive applications to be discovered, and stopped if necessary, are typically built into most NOS platforms.
- System administrators can compensate for the lack of memory through the use of "virtual memory".
- Virtual memory allocates space on the hard drive and treats it as an extension of the system RAM.

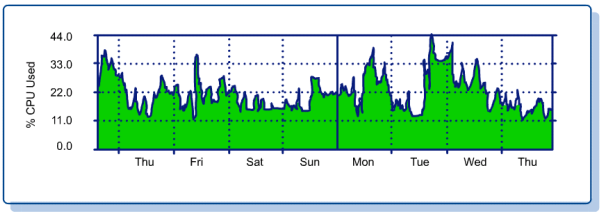


Ubuntu System Monitor



CPU Usage

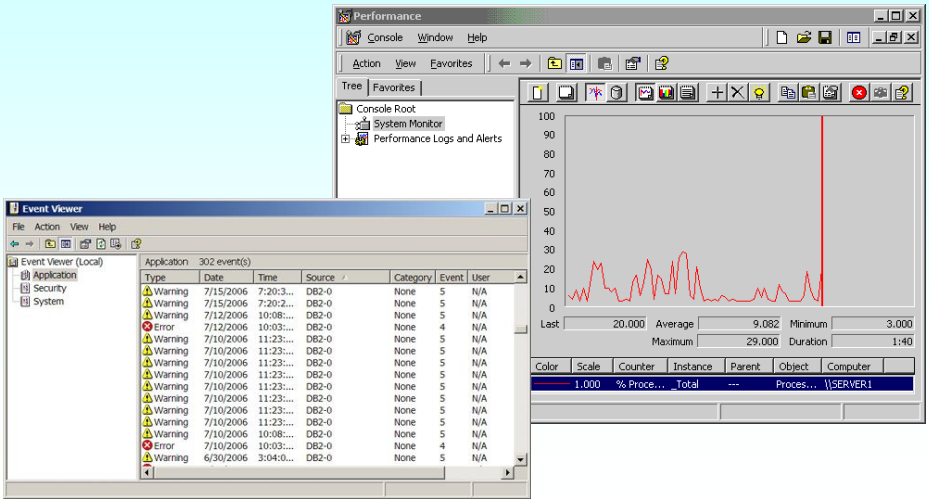
- All information used by the NOS, including the NOS itself, is processed millions of times per second by the CPU to display this information to the user.
- Built-in tools are commonly provided to allow system administrators to monitor the current level of CPU activity.
- This feedback is often presented in terms of the percentage of the CPU currently being used and is refreshed at frequent intervals.



Reviewing Daily Logs

- Most computer programs, servers, login processes, as well as the system kernel, record summaries of their activities in log files.
- These summaries can be used and reviewed for various things, including software that might be malfunctioning or attempts to break into the system.
- In Windows 2000, the Computer Management tool allows users to browse the logged events generated by the NOS.
- Two categories beneath the System Tools heading store logged information. They are "Event Viewer" and "Performance Logs and Alerts"

Reviewing Daily Logs



Reviewing Daily Logs

- Linux uses log daemons to control the events that are entered in the system log.
- Most of the Linux systems log files are located in the **/var/log** directory.
- These log files are maintained by the system log daemon (**Syslogd**) and the kernel log daemon (**klogd**).
- These two daemons are configured using the **syslog.conf** file.

The syslog.conf File

```
# Log all kernel messages to the console.
# Logging much else clutters up the screen.
#kern.*                                         /dev/console

# Log anything (except mail) of level info or higher.
# Don't log private authentication messages!
*.info;mail.none;authpriv.none;cron.none  /var/log/messages

# The authpriv file has restricted access.
authpriv.*                                      /var/log/secure

# Log all the mail messages in one place.
mail.*                                          /var/log/maillog


# Log cron stuff
cron.*                                          /var/log/cron

# Everybody gets emergency messages
*.emerg                                         *

# Save news errors of level crit and higher in a special file.
uucp,news.crit                                  /var/log/spooler

# Save boot messages also to boot.log
local7.*                                        /var/log/boot.log
~
"/etc/syslog.conf" [readonly] 26L, 693C      1,1      All
```

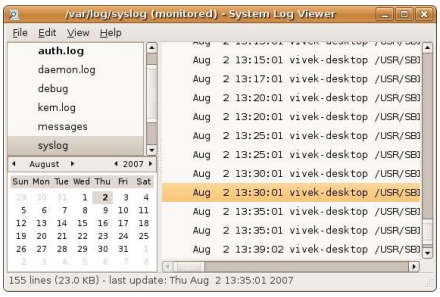
Reviewing Daily Logs

Linux Log files and usage:

- **/var/log/messages** : General log messages
- **/var/log/boot** : System boot log
- **/var/log/auth.log** : User login and authentication logs
- **/var/log/daemon.log** : Running services such as squid, ntpd and others log message to this file
- **/var/log/faillog** : User failed login log file

Text view:

tail -f /var/log/auth.log
more /var/log/daemon.log
cat /var/log/mysql.err
less /var/log/messages
grep -i fail /var/log/boot



Ubuntu /var/log

- Vediamo un file di log:

```
$ ls -al /var/log/auth*
```

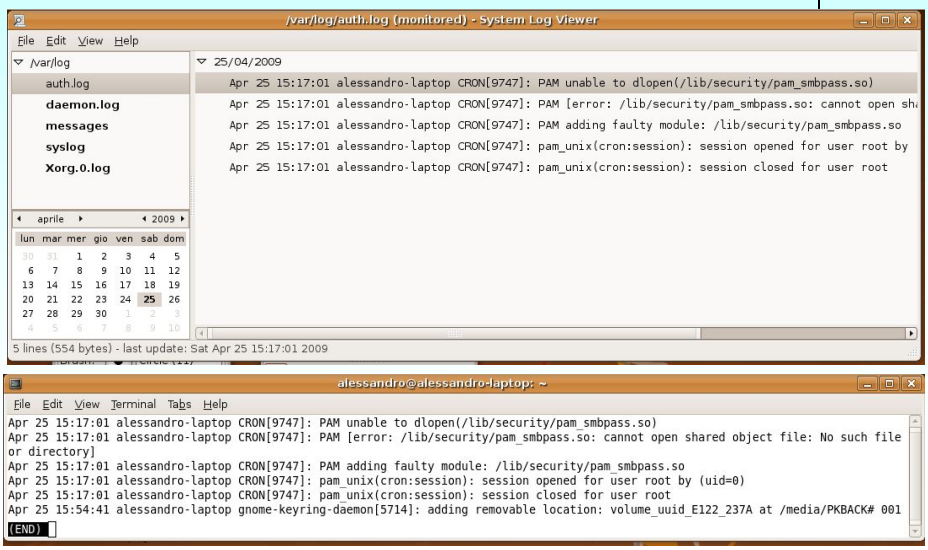
```
-rw-r----- 1 syslog adm     554 2009-04-20 15:17 auth.log
-rw-r----- 1 syslog adm 149.078 2009-04-20 15:11 auth.log.0
-rw-r----- 1 syslog adm   9.467 2009-04-18 10:42 auth.log.1.gz
-rw-r----- 1 syslog adm     876 2009-04-16 12:30 auth.log.2.gz
```

```
$ cat /var/log/auth.log | grep “Apr 21”
```

```
Apr 21 14:11:04 amemo CRON[9747]: PAM unable to dlopen(/lib/sec
Apr 21 14:11:04 amemo CRON[9747]: PAM [error: /lib/security/pam
Apr 21 14:11:04 amemo CRON[9747]: PAM adding faulty module: /li
Apr 21 14:11:04 amemo CRON[9747]: pam_unix(cron:session): sessi
```

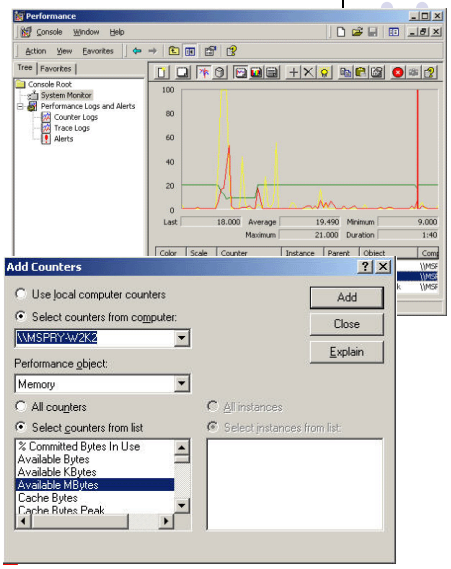
PAM (Pluggable authentication module) è un modulo aggiuntivo ad OpenLDAP e forza i client all’utilizzo di password normalizzate. Utilizza la libreria libpam-cracklib ed un dizionario di password non accettabili.

Ubuntu /var/log



Checking Resource Usage on Windows 2000 and Windows XP

- System resources are monitored in Windows 2000 and Windows XP with the Performance tool.
- This application is found under the **Start** menu > **Programs** > **System Administration** > **Performance** menu option.
- Users can then right-click on the graph and select **Add Counters** to specify which system resources to monitor in the graph.

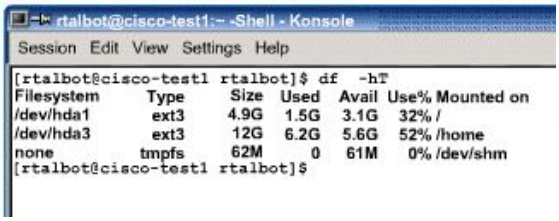


Checking Resource Usage on Linux

- The **df** command is used to display the amount of disk space currently available to the various filesystems on the machine.
- When a directory name is specified, the **du** command returns the disk usage for both the contents of the directory and the contents of any subdirectories beneath it.
- The **top** command functions much like the Windows 2000 Performance tool by providing detailed information regarding CPU and RAM usage.

Il comando df

- Il comando **df** (*disk space of the file system*) visualizza l'ammontare di spazio libero e occupato su tutti i dischi attualmente montati.
  - [-h] Aggiunge a ciascuna dimensione un suffisso, come M per megabyte binario («mebibyte»)
  - [-i] Dà informazioni sull'uso degli inode, invece che dei blocchi
  - [-l] Limita il risultato ai soli filesystem locali
  - [-T] Stampa il tipo di ciascun filesystem



Il comando du

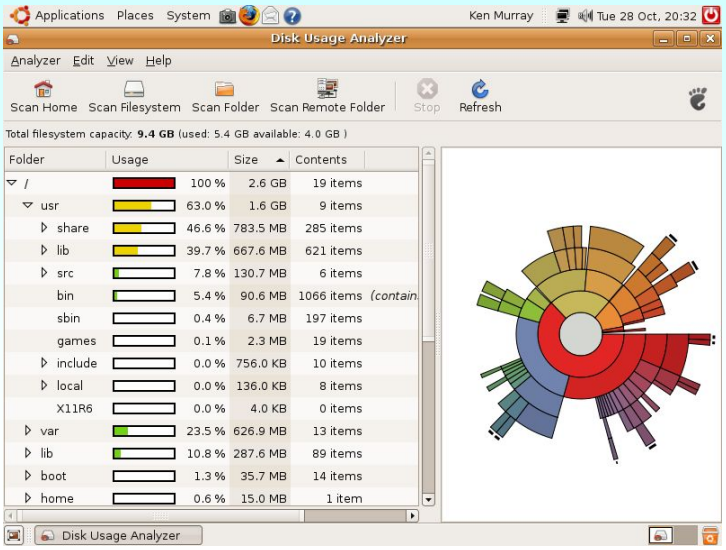
- Il comando **du** (*disk usage statistics*) visualizza la quantità usata di spazio su disco da una certa directory

The du Command

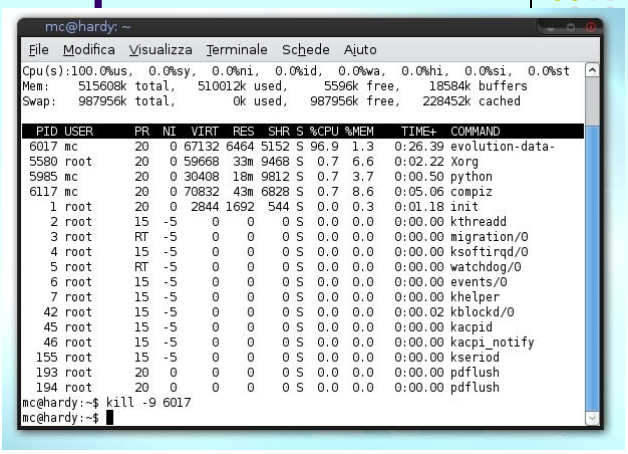
```
[rtalbot@cisco-test1 rtalbot]$ su root
Password:
[root@cisco-test1 rtalbot]# du /usr -h --max-depth=1
142M    /usr/bin
483M    /usr/lib
3.0M    /usr/libexec
12M     /usr/sbin
640M    /usr/share
88M     /usr/x11R6
4.0k    /usr/dict
4.0k    /usr/etc
3.9M    /usr/games
4.3M    /usr/include
92k     /usr/local
4.0k    /usr/src
1.4M    /usr/kerberos
8.8M    /usr/i386-glibc21-linux
1.4G    /usr/
[root@cisco-test1 rtalbot]#
```

[-c] produce un totale finale
[-h] risultati in formato comprensibile
[-S] esclude le sottodirectory
[-s] solo il totale per ogni argomento
[-x] salta le dir. degli altri file system
[--max-depth=N] fino ad una profondità di N livelli

Disk Usage grafica (baobab)



Il comando top



- Il comando **top** mostra i processi che usano più CPU, e fornisce in tempo reale istantanee dell'attività del processore.
- Mostra una lista dei task del sistema che fanno un uso più intenso della CPU, e può mettere a disposizione un'interfaccia interattiva per manipolare i processi.