



**Mezzora
d'amicizia**



**INTRODUZIONE
ALLA**

BASH

(Bourne Again SHell)

by **Wolf**

24/06/2009

<http://www.linuxvar.it>

Luigi Bassetti – Gigi aka *Wolf*

email:

luigi dot **bassetti** at **email** dot **it**

Introduzione

Cos'è la BASH?

BASH è una “shell testuale”, ovvero un'interfaccia testuale per interagire col sistema.

**Perchè usare BASH nell'epoca delle
interfacce grafiche sempre più evolute?**

Interfacce grafiche

VS

BASH

Interfacce grafiche

Vantaggi:

Migliore facilità di utilizzo (più user friendly).

Curva di apprendimento meno ripida.

Non necessita di ricordarsi i nomi di comandi come per le interfacce testuali.

Svantaggi:

Minore interoperabilità tra programmi.

Interoperabilità più rigida.

Maggior impiego di risorse di sistema.

BASH

Vantaggi:

Alta interoperabilità tra i vari comandi e programmi.

Flessibilità, velocità e potenza non raggiungibili dalle interfacce grafiche.

Possibilità di svolgere compiti complessi in minor tempo.

Minor consumo di risorse di sistema.

Possibilità di usare un vero e proprio ambiente di programmazione che interagisce con comandi e programmi già presenti.

Svantaggi:

Necessità di ricordarsi molti nomi di comandi e di programmi testuali.

Curva d'apprendimento più ripida.

Ci sono operazioni che sono più facili da eseguire graficamente e altre testualmente. Ci sono però degli insiemi di operazioni che si possono fare testualmente ma non graficamente.

La BASH dà un'elasticità che le interfacce grafiche non permettono, senza contare la possibilità offerta dallo scripting attraverso degli script scritti in dei file (per le operazioni che si dovranno ripetere più volte). Questa caratteristica è però anche utile a riga di comando per eseguire compiti ripetitivi che richiederebbero molto tempo.

BASH in pratica...

Programmi e comandi testuali in Bash, e loro opzioni

Un esempio:

ls

ls: un comando per vedere il contenuto di una directory

Alcune opzioni (le opzioni hanno solitamente uno o due trattini davanti):

ls -l (per vedere le caratteristiche dei file)

ls -a (per vedere i file e le directory nascosti)

ls -color (per vedere file e directory colorati in modo da distinguerli meglio)

ls -lah -color (mischiamo le opzioni)

ls sta per “list”. Per facilitare la memorizzazione dei comandi ricordarsi che spesso sono una contrazione del significato del comando.

Come trovare i manuali dei comandi?

man

“man man”

man -a

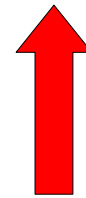
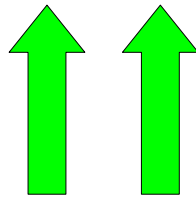
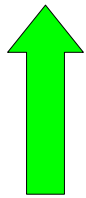
Cercare nei man con “/” e “?”

apropos & whatis

(come esercizio cercate la differenza tra i due)

Una digressione su: i permessi

drwxr-xr-x	2 gigi users	72	2006-02-23	15:26	Racconti/
-rwxr-xr-x	1 gigi users	1,6K	2005-02-06	19:57	rc.local*
-rw-r--r--	1 gigi users	235	2004-10-26	18:14	Recapiti.txt



chmod – il comando per cambiare i permessi dei file e delle directory

file – determina il tipo di file

...altri comandi tra i più usati...

**su e sudo – diventare superuser,
ovvero root**

exit – per ritornare l'utente precedente o per uscire da un terminale

cd - Change working directory

**. e .. ovvero la directory
corrente e la directory superiore
(ricordarsi che . precede anche i nomi di
file e directory nascosti)**

**cd senza opzioni per ritrovarsi
nella propria home directory**

**cd – per ritornare nella directory
precedente**

tree – l'albero delle directory
(e dei file)

**ATTENZIONE: BASH è case sensitive,
ovvero fa distinzione tra maiuscole e
minuscole.**

**Il tasto “tab”
e
l'autocompletamento,
dei comandi e dei
percorsi**

**I tasti su e giù (freccia),
andata e ritorno verso i
comandi precedentemente
digitati.**

pwd - print name of current/working
directory

...non dimentichiamoci **Is! ;-)**

cp - copy files and directories

mv - move (rename) files

rm - remove files or directories

rmdir – rimuove directory vuote

I caratteri jolly *, ? e [], senza dimenticare \, un modo più potente per usare i comandi appena descritti.

Alcuni esempi:

ls *.txt (elenca tutti i file che hanno l'estensione .txt)

cp -r mese? directory-mesi/ (copia tutti i file e le directory il cui nome inizia con “mese” e finisce con un carattere qualsiasi nella directory “directory-mesi”)

rm giann[ia].txt (cancella i file “gianni.txt” e “gianna.txt”)

rm -r mese[1-6] (cancella tutte le directory (o i file) che iniziano con “mese” e hanno i numeri da 1 a 6, perciò “mese1”, “mese2”, “mese3”, “mese4”, “mese5” e “mese6”)

mv impari\? insegnamento/ (sposta il file “impari?” nella directory “insegnamento”)

rm -fr /

**ATTENZIONE AI COMANDI
DISTRUTTIVI!!!**

**alias, unalias e memorizzazione
delle opzioni di default**

```
# Define some "alias" for all users and root:  
alias cp='cp -i'  
alias mv='mv -i'  
alias rm='rm -i'  
alias less='less -r'  
alias la='ls -a'  
alias ll='ls -l'  
alias lla='ls -la'  
alias grep='grep --color=auto'
```


cat – concatenare file e visualizzarne il contenuto

less – mostrare un lungo output facendolo scorrere avanti e indietro

grep – ricerche nel testo dei file o nell'output

grep e le **espressioni regolari**, simili ai caratteri jolly ma più complesse e più potenti

permettono ricerche complesse nel testo dei file

se volete potete approfondire questo argomento per conto vostro, trattarlo potrebbe allungare sensibilmente il tempo di questa mezzora, possiamo comunque riprenderlo in una prossima lezione...

find – cercare qualcosa

**Concatenare i comandi passando
l'output all'input di un altro
programma:**

|
(pipe)

Redirigere l'output di un comando in un file:

>

(scrive l'output in un file, se non c'è lo crea, se c'è lo sovrascrive)

>>

(scrive l'output in un file appendendolo al contenuto già presente. Se non esiste lo crea)

tee

(permette con le sue opzioni di eseguire le operazioni di **>** o **>>** ma visualizzando contemporaneamente l'output a schermo)

Alcuni esempi riassuntivi:

ls *.txt | grep 'prova' (fa una lista dei file con estensione “.txt” e in quella lista visualizza i nomi che contengono “prova”)

ls *.txt | grep 'prova' > elenco-file-prova.txt (come sopra, ma anziché visualizzare a schermo l'output lo scrive nel file “elenco-file-prova.txt”)

ls *.txt | grep 'prova' | tee -a elenco-file-prova.txt (come sopra, in più visualizza l'output anche a schermo. Con l'opzione -a (append) di tee se “elenco-file-prova.txt” non esiste lo crea, se esiste aggiunge l'output in coda senza sovrascriverlo)

find . -iname '*.jpg' -exec cp '{}' directory-immagini ';' (cerca tutti i file che finiscono con “.jpg” nella directory corrente e in tutte le sottodirectory senza far differenza tra minuscole e maiuscole e li copia nella directory “directory-immagini”)

find . -iname '*.jpg' | less (cerca tutti i file che finiscono con “.jpg” nella directory corrente e in tutte le sottodirectory senza far differenza tra minuscole e maiuscole e li mostra, con less fa scorrere l'output in su e in giù)

Domande?

Abbiamo visto i seguenti comandi:

file determina il tipo di file

ls mostra il contenuto delle directory

cp copia file e directory

mv muove e rinomina file e directory

rm rimuove file o directory

rmdir rimuove directory vuote

caratteri jolly caratteri speciali per cercare nomi di file

man mostra le pagine di manuale

su cambia utente o diventa superuser (root)

sudo esegue un comando come un altro utente

exit ritorna l'utente precedente o esce da un terminale

alias crea una sostituzione di comando

unalias annulla la sostituzione di comando creata con **alias**

cat concatena file e/o ne stampa il contenuto

less fa scorrere in su e in giù schermate di output lunghe

grep esegue ricerche nei testi dei file o nell'output

espressioni regolari caratteri speciali per cercare stringhe

find programma molto potente per cercare file e directory

| canalizza l'output di un comando verso l'input di un'altro programma

> reindirige l'output in un file sovrascrivendolo se non c'è

>> reindirige l'output in un file appendendo il contenuto se c'è

tee come **>** e **>>**, ma mostra anche l'output

Sul nostro sito potete trovare, oltre a questa presentazione, una serie di esercizi con i quali potete verificare quanto avete appreso. In questo caso però gli esercizi migliori consistono nell'approfondimento tramite i man dei comandi citati nella mezzora e dallo sperimentare opzioni e interazioni degli stessi.



**Mezzora
d'amicizia**



**Grazie dell'attenzione, buona... Bash a tutti e... alla
prossima puntata!**

