

# RTFM-Bash

October 31, 2015

```
<h1>RTFM Bash</h1>
<h3>Corsi Linux 2015</h3>

<a class href="https://www.poul.org">
</a>

Filippo Cremonese (filippocremonese@gmail.com)</p>
```

## 0.0.1 Hello World, in Bash

```
In [17]: printf 'Ciao!\n\nBenvenuti alla lezione su bash!' | cowsay
```

```
-----
/ Ciao!                                     \
|                                           |
\ Benvenuti alla lezione su bash! /
-----
      ^__^
      (oo)\_______
      (__)\       )\/\
           ||----w |
           ||     ||
```

Potete trovare queste slide a questo indirizzo <https://github.com/fcremo/slides>

## 1 Perché usare un terminale al giorno d'oggi?

- Svolgere compiti ripetitivi in fretta
- Amministrare e configurare un sistema
- Gestire un computer remoto
- Quando le cose si rompono un terminale è sempre disponibile
- Per essere l33t h4x0rZ

### 1.1 Useremo bash

Sulla maggiorparte dei sistemi potete avviare un emulatore di terminale premendo **CTRL+ALT+T** o cercando **Terminal** o **xterm** tra i software installati.

In alternativa potete accedere ai terminali nativi con **CTRL+ALT+F1** (**CTRL+ALT+F7** per tornare indietro)

## 2 Cos'è possibile fare con bash?

```
In [18]: cat ~/.bash_history | cut -f 2 -d ";" | cut -f 1 -d " " \
        | sort | uniq -c | sort -nr | head -5
```

```
109 echo
    79 cd
    55 ls
    37 cat
    29 cowsay
```

Stampa i cinque comandi più utilizzati contando quante volte sono stati lanciati

## 3 BASH 101: Read The Freaking Manual

Ogni comando ha diverse opzioni, documentate nella manpage relativa.

Comando	Funzione
<code>apropos &lt;what&gt;</code>	cercare comandi rilevanti a <b>what</b>
<code>man &lt;nomecomando&gt;</code>	consultare la documentazione
<code>man man</code>	consultare la documentazione di <code>man</code>
<code>comando -h/-help</code>	stampare l'help di un comando (non sempre disponibile)
TAB	autocompletamento (non sempre disponibile)

## 4 Operare su file e cartelle

Comando	Funzione
<code>pwd</code>	stampare la cartella corrente
<code>ls</code>	elencare i file nella cartella corrente
<code>cd &lt;[nomedir,..]&gt;</code>	entrare nella cartella <code>nomedir</code> /risalire di una cartella
<code>mkdir &lt;nomedir&gt;</code>	creare la cartella <code>nomedir</code>
<code>rm &lt;nomefile&gt;</code>	cancellare il file <code>nomefile</code>
<code>rm -r &lt;nomedir&gt;</code>	cancellare la cartella <code>nomedir</code>
<code>chmod</code>	cambiare permessi a file e cartelle
<code>chown/chgrp</code>	cambiare proprietari a file e cartelle

## 5 Manipolare file di testo

Comando	Funzione
<code>cat &lt;nomefile&gt;</code>	stampare il file <code>nomefile</code> su STDOUT (a schermo)
<code>echo "qualcosa"</code>	stampare <code>qualcosa</code> su STDOUT
<code>nano/vim/emacs</code>	editare il file <code>nomefile</code>
<code>less &lt;nomefile&gt;</code>	visualizzare il file <code>nomefile</code>
<code>head/tail &lt;nomefile&gt;</code>	visualizzare l'inizio/la fine del file <code>nomefile</code>
<code>tee &lt;nomefile&gt;</code>	stampare STDIN su STDOUT e in <code>nomefile</code>

## 6 Cercare tra i file

Comando	Funzione
<code>grep &lt;espressione&gt; &lt;nomefile&gt;</code>	cercare <b>espressione</b> nel file <b>nomefile</b>
<code>find -name "*parola*"</code>	cercare nella cartella corrente (ricorsivamente) i file che contengono “parola” nel nome
<code>locate "parola"</code>	cercare i file che contengono “parola” nel nome in tutto il sistema
<code>sudo updatedb</code>	aggiornare il database di locate

## 7 Altri comandi

Comando	Funzione
<code>./nomefile</code>	eseguire <b>nomefile</b> situato nella cartella corrente
<code>wget "https://poul.org"</code>	scaricare un file dal web (l'homepage del POuL)
<code>file &amp;lt;nomefile&gt;</code>	visualizzare il tipo di file di <b>nomefile</b>
<code>sudo cmd</code>	eseguire il comando <b>cmd</b> come amministratore (root)

## 8 Redirezione

- Con `>` si redireziona STDOUT di un comando su un file
- Con `>` il file viene sovrascritto se esiste, con `>>` si accoda
- Con `<` si collega STDIN di un processo ad un file

```
In [19]: echo "something" > file_di_destinazione;
        cat file_di_destinazione;
```

```
something
```

```
In [20]: echo "another thing" >> file_di_destinazione;
        cat file_di_destinazione
```

```
something
another thing
```

```
In [21]: cat - <file_di_destinazione
```

```
something
another thing
```

## 9 Pipe

- Con `|` si collega STDOUT di un comando allo STDIN di un altro comando

Es: `echo "20" | sudo tee /sys/devices/system/cpu/intel_pstate/max_perf_pct`  
Scrive 20 dentro al file, utilizzando l'utente root

```
In [22]: echo "20" | tee altro_file;
        cat altro_file
```

```
20
20
```

## 10 Jobs

- Per mettere in pausa un comando in esecuzione si utilizza **CTRL+Z**
  - **jobs** elenca i lavori sospesi o in background
  - **bg <num>** se per continuare l'esecuzione del job <num> in background
  - **fg <num>** per continuare l'esecuzione del job <num> in foreground

## 11 Jobs

- Per eseguire in background un comando basta apporre **&** alla fine
  - es: **wget "https://www.poul.org" &** Scarica l'homepage del POuL in background
  - **stdout** ed **stderr** del processo rimangono collegati al terminale Per non mostrarli utilizzare **disown <job n°>**

## 12 Gestione dei software su sistemi Debian e Ubuntu

Comando	Funzione
<code>sudo apt-get update</code>	aggiornare l'elenco dei pacchetti del sistema
<code>sudo apt-get upgrade</code>	aggiornare i pacchetti installati
<code>sudo apt-get install &lt;nomepacchetto&gt;</code>	installare il pacchetto <b>nomepacchetto</b>
<code>sudo apt-get remove &lt;nomepacchetto&gt;</code>	rimuovere il pacchetto <b>nomepacchetto</b>
<code>apt-cache search &lt;nomepacchetto&gt;</code>	cercare <b>nomepacchetto</b> nella lista dei pacchetti disponibili

Grazie per l'attenzione!  
Domande?

<br><span class="