

Prontuario dei comandi piú comuni

Elisabetta RONCHIERI

6 novembre 2014

Data: 7 Novembre 2014

1 Introduzione

Questa non é una guida ma solo un prontuario ai comandi Linux piú utili durante l'utilizzo di Git Bash che permette di usare i comandi git come se si lavorasse in ambiente LINUX e UNIX.

2 Comandi Unix

Si fa notare che per avere i dettagli dei comandi basta far seguire il comando da spazio più help preceduto da due volte -.

2.1 Comandi di sistema

date [opzioni]
riporta data e ora corrente.

echo [stringa di testo]
riporta la stringa di testo specificata in standard output.

exit chiude la shell.

help < *comando* >
visualizza una breve schermata d'aiuto, per le funzioni built-in.

hostname visualizza il nome della macchina attuale.

pwd visualizza la directory di lavoro corrente alla UNIX, considerando l'unità di lavoro come una cartella.

script < *file* >
memorizza ogni cosa che appare sullo schermo in un file fino a quando si esegue exit.

uname -a
ottiene il nome del sistema operativo, la versione del kernel e altro.

which < *comando* >
riporta il percorso del comando o l'alias di shell in uso.

2.2 Comandi di pipeline e simili

comando > filename ridireziona l'output di comando verso il filename. Il contenuto del file viene cancellato e sostituito dal nuovo testo.

comando >> filename ridireziona l'output di comando verso il filename. Il testo viene appeso in coda al file.

comando < filename usa il filename come input di comando.

comando < filename-in > filename-out combinazione delle precedenti.

comando < filename-in >> filename-out combinazione delle precedenti.

2.3 Gestione file

cd [directory]
cambia directory.

cp [opzioni] file1 file2
copia file1 in file2. Questo comando crea o sovrascrive file2.

cp -r dir1 dir2
se si vuole copiare una directory e tutto il suo contenuto in un'altra directory usare l'opzione *-r*.

ls [opzioni] [directory o file]
elenaca il contenuto della directory specificata o i permessi del file specificato (*-l* visualizza i dettagli, *-r* le sottodirectory).

mkdir [opzioni]
directory crea una directory.

mv [opzioni] file1 file2
muove file1 in file2. Il file viene rinominato.

pwd stampa la directory di lavoro corrente.

rm [opzioni] file
rimuove un file (*-i* chiede conferma prima di eliminare i file, *-f* non chiede mai nulla).

rm -r dir
rimuove una directory e il suo contenuto (*-i* chiede conferma prima di eliminare i file, *-f* non chiede mai nulla).

rmdir [opzioni] directory
rimuove una directory vuota.

touch [opzioni] [data/ora] file
aggiorna l'ora di ultima modifica. Se il file non esiste, di default lo crea.

2.4 File testuali

cat [opzioni] file
concatena (ossia scrive su schermo) un file.

grep [opzioni] 'stringa di ricerca' argomento
ricerca nell'argomento tutte le occorrenze della stringa di ricerca specificata e visualizza le righe in cui compare.

grep -ir 'stringa di ricerca' .
come sopra: la opzione *-r* permette di effettuare la ricerca in un'intera cartella di files; la opzione *-i* permette di non distinguere le maiuscole e minuscole.

head [-numero] file
visualizza le prime linee (di un numero specificato) di un file.

tail [-numero] file
visualizza le ultime linee (di un numero specificato) di un file.