

Buono a sapersi

mobytrick

16 Settembre 2015

Per la serie che l'educazione deve essere continua, ho scoperto un metodo per creare e gestire file a manetta in ambiente Unix/Linux.

La gestione di massa dei file è possibile facendo uso delle *Regular Expressions*. Se ad esempio voglio selezionare i file che iniziano con una determinata stringa, sia questa ad esempio `temp`, allora il comando è il seguente:

```
ls temp*
```

Nell'esempio proposto, il carattere `*` (asterisco) va inteso come *da zero ad ∞ caratteri*. Il concetto di infinito però non si addice all'informatica, ove non ha proprio senso. Il simbolo ∞ va più propriamente interpretato come *qualsivoglia*. Ad ogni modo si tenga presente che il comando `ls`, ed altri simili tipo `cp`, `mv`, etc implementano un *RE engine* (Regular Expression engine) piuttosto rudimentale.

Lo stesso risultato si poteva ottenere, ma in maniera meno efficiente, con la seguente *pipe*:

```
ls | grep temp
```

Si tenga presente che l'uso delle Regular Expressions è ammesso coi comandi `ls`, `cp`, `mv`, `rm` ed altri che hanno la caratteristica di lavorare con nomi di file **che esistono già**. Rimane scoperto il comando `touch`, che è spesso usato per creare file nuovi. Il comando:

```
touch temp*
```

crea un unico file il cui nome è `temp*`. E se avessi la necessità di creare un ingente quantità di file? Si sa che in Unix, e di riflesso in Linux, la dimensione delle directory cresce sempre. Popolandone una, la dimensione viene via via aumentata per poter gestire il numero dei file. Se poi questi vengono tolti la

dimensione non viene riadeguata al ribasso. Per rendersene conto basta creare una gran quantità in una directory. Quindi farsi dare la sua dimensione; poi eliminare tutti i file ed osservare di nuovo la dimensione. Che dopo l'aumento rimarrà al livello raggiunto. Il problema è quello di creare in maniera veloce un gran numero di file. Poiché in Unix le vie per raggiungere un determinato risultato sono molteplici, me la cavavo con il seguente script:

```
#!/bin/bash
for idx in `seq 1 200`
do
    touch temp${idx}
done
exit 0
```

che crea 200 (numero *arbitrario*) file il cui nome è composto da una parte iniziale fissa `temp` seguita da un numero compreso nell'intervallo `[1, 200]`. Gli estremi dell'intervallo possono essere definiti anche parametricamente e nulla vieta di rendere tale pure la parte iniziale.

Proprio in questi giorni mi sono imbattuto casualmente in una sintassi per me sconosciuta. Col comando:

```
touch temp{1..25}
```

vengono creati 25 (pure questo, numero *arbitrario*) file il cui nome è composto dalla stringa fissa `temp` seguita da una oppure due cifre. Un bel passo avanti rispetto allo script. La *man page* del comando `touch` non riporta tale possibilità, che può essere utilizzata anche con altri comandi, tipo `ls`. L'intervallo può essere costituito anche da lettere e può trovarsi anche in posizioni diverse dalla fine.