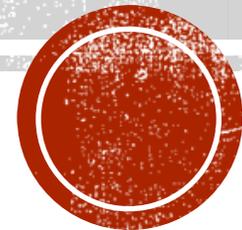


# **GESTIONE DI UTENTI, GRUPPI E FILE**

**SICUREZZA DOCUMENTALE**

**A.A. 2017-2018**



# UTENTI E GRUPPI

- Su Linux, come su molti altri sistemi operativi, è possibile creare una molteplicità di **utenti** in grado di lavorare all'interno del sistema:
  - Linux è un sistema *multiuser*.
- Gli utenti, a loro volta, possono essere raggruppati all'interno di **gruppi** (ciascun utente può partecipare ad uno o più gruppi) caratterizzati da permessi comuni a tutti i membri.





# CREARE, MODIFICARE E CANCELLARE GLI UTENTI

- Di default Linux dispone di un unico utente, cioè l'utente **root** che gode dei pieni poteri.
  - Spetta quindi a questo utente creare nuovi utenti nel sistema.
- Per aggiungere un utente si ricorre al comando **useradd**, questa è la sintassi:
  - **useradd [opzioni] nomeutente**
  - Esempio: creare l'utente "pippo" e assegnarlo al gruppo "visitatori"
  - **useradd -G visitatori pippo**



# MODIFICARE / CANCELLARE UTENTI

- Per modificare il profilo di un utente si utilizza **usermod**, in questo modo:
  - **usermod [opzioni] nomeutente**
  - Si supponga di voler modificare i gruppi di appartenenza di "pippo":
    - **usermod -G utenti pippo**
- Per cancellare un utente si userà, invece, il comando **userdel**;
  - Per cancellare l'utente "pippo" scriviamo:
    - **userdel pippo**



# ASSEGNARE/MODIFICARE PASSWORD

- Ciascun utente può disporre di una password per accedere al sistema.
- Dopo aver creato un nuovo utente con `useradd`, quindi, sarà necessario assegnargli una password mediante il comando **passwd**. Questa è la sintassi:
  - `passwd nomeutente`
- Mediante lo stesso comando è anche possibile modificare una password già assegnata.



# CREARE, MODIFICARE E CANCELLARE GRUPPI

- Per creare un nuovo gruppo utilizziamo il comando **groupadd**; questa è la sintassi:
  - `groupadd [opzioni] nomegruppo`
  - creare un nuovo gruppo chiamato "staff"
  - `groupadd staff`
- Per modificare un gruppo si utilizza **groupmod**
  - Supponiamo di voler rinominare il gruppo "staff" in "colleggi":
  - `groupmod -n colleggi staff`
- Per cancellare un gruppo si utilizzerà il comando **groupdel**.
  - `groupdel colleggi`



# ALCUNI COMANDI IMPORTANTI

- Il comando **pwd** (*present working directory*) ci dice in quale directory ci troviamo.
  - **pwd**
- Il comando **ls**, se digitato senza argomenti, ci fornirà l'elenco completo dei file e delle sotto-directory presenti nella cartella corrente.
  - Per conoscere il contenuto di una directory diversa da quella nella quale ci troviamo, usiamo **ls** accompagnato dal percorso della directory
  - **ls /percorso/della/directory/**
- Per vedere anche alcune informazioni relative ad ogni singolo file e/o directory (proprietario, dimensioni, ecc.) si usa l'opzione "-l" (list)
  - **ls -l /percorso/della/directory/**



# ALCUNI COMANDI IMPORTANTI

- Per vedere anche i file nascosti si usa l'opzione "-a" (all)
  - `ls -al /percorso/della/directory/`
- Per conoscere il contenuto delle sottodirectory si usa l'opzione "-R"
  - `ls -R /percorso/della/directory/`
- Il comando **cd** serve per muoversi tra le directory.
  - `cd /percorso/della/directory/`





# INFORMAZIONI SU FILE/DIRECTORY

- Utilizzando il comando `ls -al` si ottengono informazioni sui file e le directory

```
GPIRRO:db gpirro$ ls -al
total 3235944
drwxr-xr-x 10 gpirro admin    320 Dec  1 07:32 .
drwxr-xr-x  4 gpirro admin   128 Apr 25 2016 ..
-rw-r--r--  1 gpirro admin 16384 Nov 29 14:45 .virtuoso.ini.swp
-rw-r--r--  1 gpirro admin 2097152 Nov 30 21:09 virtuoso-temp.db
-rw-r--r--  1 gpirro admin 1654652928 Dec  1 07:32 virtuoso.db
-rw-r--r--  1 gpirro admin  7861 Oct 17 12:53 virtuoso.ini
-rw-r--r--  1 gpirro admin   13 Nov 30 21:09 virtuoso.lck
-rw-r--r--  1 gpirro admin 12673 Dec  1 07:32 virtuoso.log
-rw-r--r--  1 gpirro admin  0 Oct 17 11:23 virtuoso.pxa
-rw-r--r--  1 gpirro admin  5665 Dec  1 08:02 virtuoso.trx
```

**Permessi**

**Proprietario**

**Gruppo**



# INFORMAZIONI SU FILE/DIRECTORY

```
-rw-r--r-- 1 gpirro admin 16384 Nov 29 14:45 .virtuoso.ini.swp
```

Permessi

- La **colonna dei permessi** contiene 10 lettere (o trattini):
  - Il primo spazio indica la tipologia dell'elemento e può avere i seguenti valori: **d** (directory), **l** (link simbolico), **-** (file);
  - I nove spazi successivi indicano i permessi; più precisamente si tratta di tre distinti gruppi di 3 permessi (**r** = lettura; **w** = scrittura; **x** = esecuzione).
    - Il primo gruppo da tre riguarda il *proprietario*
    - Il secondo riguarda il *gruppo*
    - Il terzo riguarda gli *altri utenti*.



# INFORMAZIONI SU FILE/DIRECTORY

```
-rw-r--r-- 1 gpirro admin 16384 Nov 29 14:45 .virtuoso.ini.swp
```

- Le due colonne **proprietario** e **gruppo** indicano, rispettivamente l'utente proprietario del file ed il gruppo di appartenenza.
- Come modificare queste informazioni?
  - **chmod** è il comando che modifica i permessi (lettura, scrittura, esecuzione).



# IL COMANDO CHMOD

- `chmod` con **sintassi simbolica** consente di assegnare diversi permessi al proprietario, al gruppo ed agli altri utenti
  - `chmod a=rwx nomefile`
  - nel nostro esempio abbiamo assegnato a tutti (**a** = all) tutti i permessi (**rwx**).
  - A sinistra del simbolo uguale (=) abbiamo l'assegnatario del permessi, a destra i permessi assegnati.
  - L'assegnatario viene identificato attraverso una lettera:
    - **a** (tutti)
    - **u** (utente proprietario)
    - **g** (gruppo)
    - **o** (altri utenti)
- I permessi, invece, sono identificati dalle tre lettere **r**, **w** e **x** che abbiamo già visto in precedenza.



# IL COMANDO CHMOD

- `chmod` con **sintassi ottale assegna** i permessi a tutti i livelli simultaneamente. Al posto delle lettere **rwX** si utilizzano 3 numeri:
  - `chmod 777 nomefile`
  - abbiamo assegnato a tutti i massimi permessi (cioè "rwX" come nell'esempio precedente).
- I tre numeri definiscono i permessi dei tre livelli: il primo numero riguarda **l'utente** proprietario, il secondo il **gruppo**, il terzo gli **altri** utenti.
  - **7** corrisponde a **rwX**
  - **6** corrisponde a **rw**
  - **5** corrisponde a **rx**
  - **4** corrisponde a **r**
  - **3** corrisponde a **wX**
  - **2** corrisponde a **w**
  - **1** corrisponde a **X**
  - **0** negato ogni accesso
- Per assegnare al proprietario tutti i permessi, al gruppo solo lettura e scrittura ed agli altri utenti nulla:
  - `chmod 760 nomefile`



# IL COMANDO CHOWN

- `chown` è utilizzato per cambiare l'utente proprietario e/o il gruppo assegnato ad un file o ad una directory
  - `chown nomeproprietario:nomegruppo nomefile`
- Per assegnare il file `immagine.jpg` all'utente `'pippo'` ed al gruppo `'grafici'`:
  - `chown pippo:grafici immagine.jpg`
- **chgrp** si limita a cambiare il gruppo di un file o directory (senza poter intervenire sul proprietario)
  - `chgrp nomegruppo nomefile`
- Tutti e 3 i comandi `chmod`, `chown` e `chgrp` supportano la modalità ricorsiva
- `chmod -R 777 nomecartella`
- `chown -R nomeproprietario:nomegruppo nomecartella`
- `chgrp -R nomegruppo nomecartella`

