



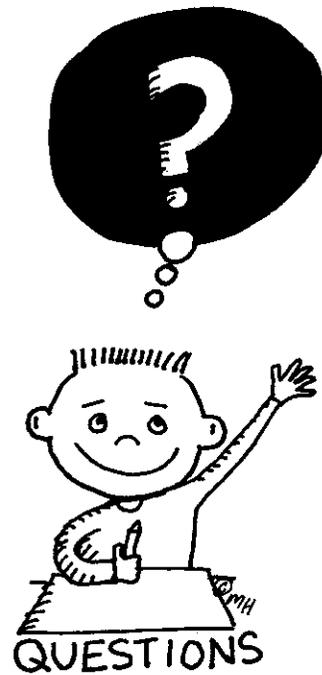
Sistemi Operativi

Corso di Laurea Triennale in Ingegneria Informatica

Esercitazione 2

- Editor
- Utenti
- Permessi

Domande sulle lezioni passate?



Lezione 2

- **Editor**
- **Utenti e gruppi,**
- **Permessi,**
- **Gestione utenti (prima parte).**

Editor di testo

➤ **Esistono vari editor di testo:**

- **vi,**
- **emacs,**
- **vim,**
- **nano**
- **...**

Editor di testo - vi

- **Per creare o aprire un file già esistente:**
 - `vi nomeFile`
- **Esistono due modalità di funzionamento per vi:**
 - 1.modalità comandi:** permette di inserire comandi e scegliere quale azione compiere;
 - 2.modalità editing:** permette di inserire e cancellare testo (come se fosse un editor di testo quale blocco note).

Editor di testi - vi

Elenco di alcuni comandi dell'editor vi

esc	Passa in modalità comandi.
i	Passa in modalità inserimento nella posizione corrente.
o	Apri in inserimento una nuova linea
x	Cancella un carattere.
u	Undo.
r?	Sostituisce con ? il carattere su cui si trova il cursore.
dd	Taglia la riga corrente.
ndd	Taglia n righe.
yy	Copia una riga.
nyy	Copia n righe.

Editor di testi - vi

Elenco di alcuni comandi dell'editor vi

p	Incolla la selezione nella riga sotto il cursore.
/word	Ricerca nel testo la parola word
n	Si posiziona sull'occorrenza successiva (nella ricerca).
N	Si posiziona sull'occorrenza precedente (nella ricerca).
:q	Esce (solo se non si sono fatte modifiche).
:w	Salva.
:wq	Salva ed esce.
:q!	Esce senza salvare.
:help	Richiama l'aiuto in-linea

Editor con interfaccia grafica

- **gvim**
- **kvim**
- **emacs**
- **kate**
- **gedit**
- **...**

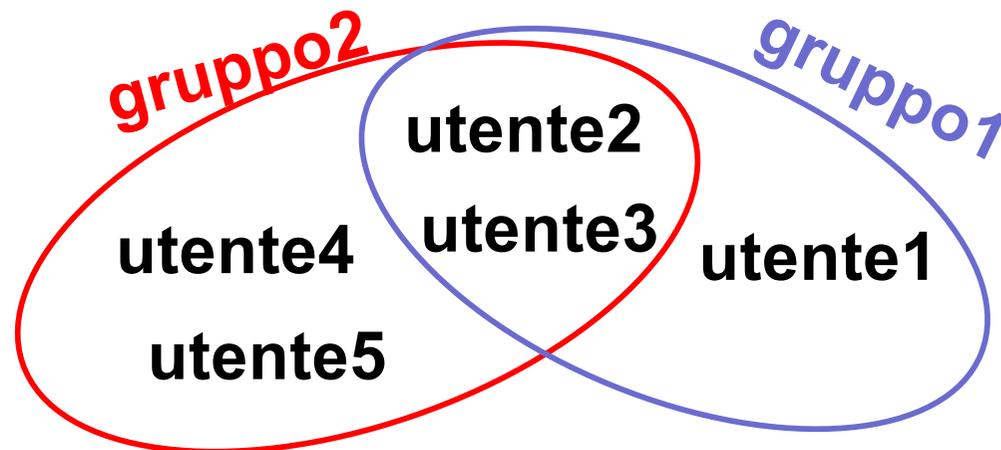
Utenti e gruppi

- Ogni utente è identificato da:
 - un **username**,
 - un **UID** (user ID) numerico.
- Username e UID sono pubblici.
- Ogni utente può appartenere ad uno o più gruppi.

Utenti e gruppi

- **Ogni gruppo è identificato da:**
 - **un groupname,**
 - **un GUID (group ID) numerico.**

Utenti e gruppi



- **utente1** appartiene al gruppo **gruppo1**,
- **utente2** appartiene ai gruppi **gruppo1** e **gruppo2**,
- **utente3** appartiene ai gruppi **gruppo1** e **gruppo2**,
- **utente4** appartiene al gruppo **gruppo2**,
- **utente5** appartiene al gruppo **gruppo2**.

Permessi



Permessi

➤ I permessi possono essere di:

- lettura,
- scrittura,
- esecuzione.

I permessi da applicare vengono differenziati in base alla natura dell'utente che vuole “operare” sul file, directory, ... :

- utente proprietario del file: **owner**,
- gruppo proprietario: **group owner**,
- gli altri utenti: **others**.

Permessi - file

Attributo	Significato
Lettura	Permette di leggere il contenuto del file
Scrittura	Permette di modificare il contenuto del file
Esecuzione	Permette di eseguire un file (binario o script)

- **Il permesso di scrittura**
 - non incide sulla possibilità di cancellare un file,
 - per la cancellazione bisogna considerare i permessi della directory.

Permessi - directory

Attributo	Significato
Lettura	Permette di leggere il contenuto (l'elenco dei file)
Scrittura	Permette di modificare il contenuto
Esecuzione	Permette di attraversare una cartella

- **Negare l'accesso in lettura** impedisce l'esecuzione con successo del comando `ls`.
- **Negare l'accesso in scrittura** impedisce di creare, cancellare e rinominare file.
- **Negare l'accesso in esecuzione** impedisce di fare un `cd` sulla directory.

Permessi – SUID, SGID, Sticky

Attributo	Significato
SUID	Durante l'esecuzione il comando acquisisce i privilegi del proprietario del file stesso
SGID	Durante l'esecuzione il comando acquisisce i privilegi del gruppo a cui il file appartiene
Sticky	Durante l'esecuzione di un comando salva l'immagine nella memoria virtuale

■ Attributo **sticky** su cartelle

- non consente ad un utente di cancellare o rinominare un file a meno che non ne sia il proprietario (anche se ha il permesso in scrittura sulla directory).

Permessi – rappresentazione simbolica

➤ I permessi di un file, directory,... possono essere visualizzati con il comando `ls -l`:

➤ `drwxr-xr-x .. prova`



➤ **t rwx rwx rwx**

- **t** : tipo di file, **d** : directory, **-** : file, ...
- **r**: permesso di lettura,
- **w**: permesso di scrittura,
- **x**: permesso di esecuzione,
- **-** indica l'assenza del permesso corrispondente

Permessi – rappresentazione simbolica

- Il permesso di esecuzione in **owner** può essere anche:
 - **s** : SUID ed esecuzione permessi,
 - **S** : solo SUID.
- Il permesso di esecuzione in **group owner** può essere anche:
 - **s** : SGID ed esecuzione permessi,
 - **S** : solo SGID.
- Il permesso di esecuzione in **others** può essere anche:
 - **t** : sticky ed esecuzione permessi,
 - **T**: solo sticky.

Permessi – rappresentazione ottale

- **La rappresentazione ottale dei permessi consiste in un numero di 4 cifre in base 8.**
 - **lettura: 4,**
 - **scrittura: 2,**
 - **esecuzione: 1,**

 - **SUID: 4,**
 - **SGID: 2,**
 - **sticky: 1.**

Permessi – comando chmod 1/2

- **chmod** permette di modificare i permessi.
- La sintassi generale di **chmod** è la seguente:
- **chmod [who] [how] [which]**
 - **who**: a chi modificare i permessi: owner (u), group (g), others (o), all (a).
 - **how**:
 - '+' aggiunge un permesso,
 - '-' toglie un permesso,
 - '=' assegna un permesso.

Permessi – comando chmod 2/2

➤ Rappresentazione simbolica:

➤ `chmod a+rwx directory1`

➤ Rappresentazione ottale:

➤ `chmod 0741 file`



perm. Speciali SUID, SGID, Sticky

➤ **opzione** -R :

➤ modifica ricorsivamente i permessi delle directory indicate e del loro contenuto.

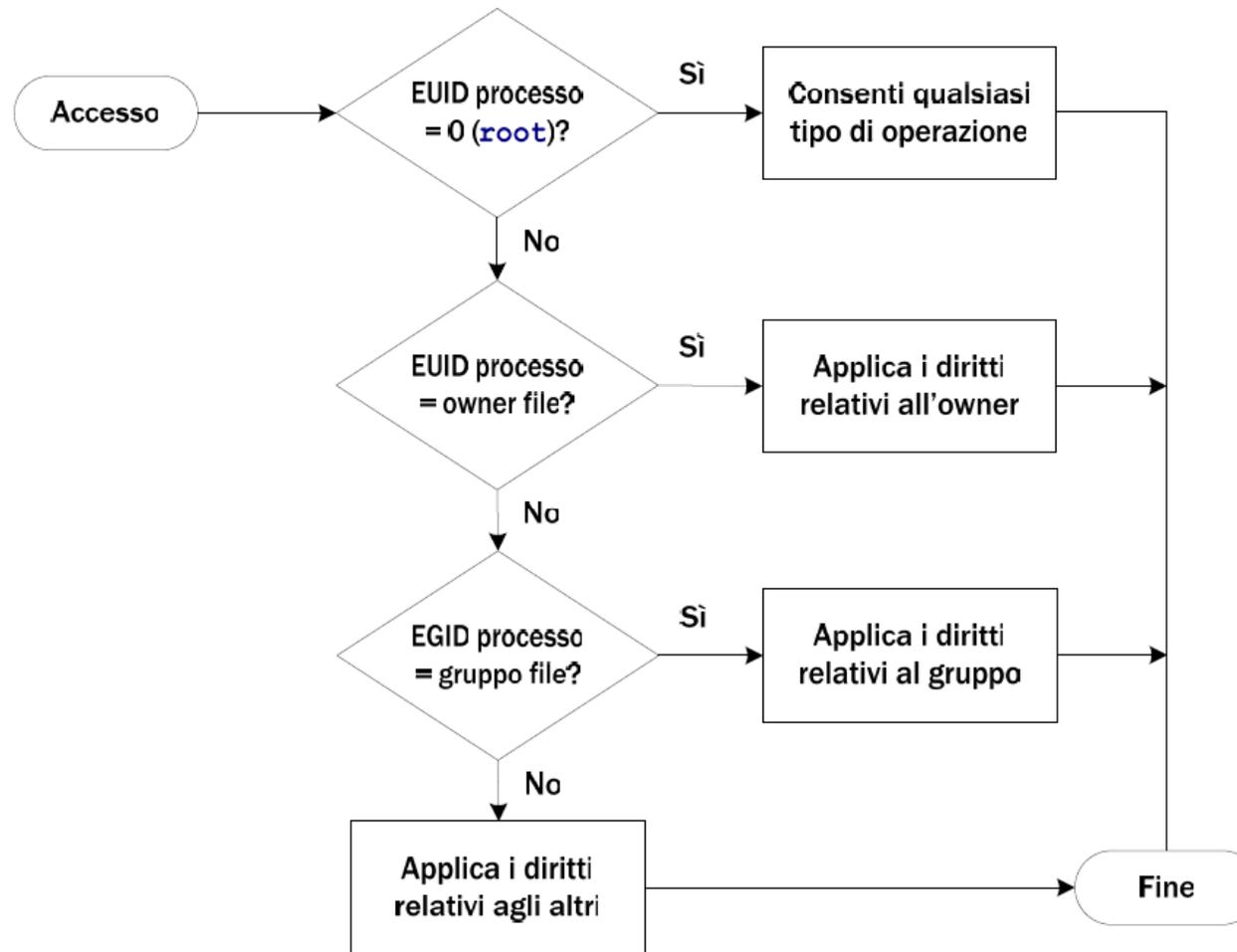
Permessi

- `chmod 0755 miofile`
è equivalente a:
- `chmod 755 miofile`
- Tutti gli 0 in testa alla notazione ottale possono essere omessi, ma non quelli seguenti.
- `chmod 0076 miofile`
è equivalente a:
- `chmod 76 miofile`
- Ma se lo 0 è riferito ai permessi di “others”, e gli altri non sono 0:
- `chmod 750 miofile # Lo 0 finale deve essere indicato`

Real e Effective User ID

Suffisso	Gruppo	Descrizione
UID	Real	Utente che ha mandato in esecuzione il programma
GID	Real	Gruppo principale dell'utente che ha mandato in esecuzione il programma
EUID	Effective	Utente usato nel controllo degli accessi (privilegi effettivi utente)
EGID	Effective	Gruppo usato nel controllo degli accessi (privilegi effettivi gruppo)

Permessi - Applicazione



Permessi – proprietari dei file

- Quando viene creato un **file/directory/...**,
 - **l'owner è rappresentato dall'EUID,**
 - **il group owner è rappresentato dall'EGUID.**

Utenti – passwd, whoami, id, groups

- **passwd** :
permette di cambiare la password
- **whoami** :
visualizza l'EUID
- **id** :
visualizza UID e *GID dell'utente selezionato o di quello corrente*
- **groups** :
visualizza i gruppi ai quali l'utente appartiene

Utenti – creazione e rimozione

- **adduser** :
crea un nuovo utente “nomeUtente”
 - `adduser nomeUtente`
- **deluser** :
rimuove l'utente “nomeUtente”
 - `deluser nomeUtente`

Utenti – comando su

- Il comando `su` (switch user o substitute user) viene tipicamente usato da:
 - **utenti non privilegiati** per avviare una **shell come root**
 - ⇒ per poterlo eseguire l'utente deve appartenere anche al gruppo **wheel**,
 - **root** per avviare una **shell come utente ordinario**.
- **Richiede autenticazione**
 - **password dell'utente che si vuole diventare.**

Utenti – comandi `chown`, `chgrp`

- Il comando `chown` permette di modificare il proprietario e/o il gruppo assegnato ad un file o directory.
- Il comando `chgrp` permette di modificare il gruppo assegnato ad un file o directory.