

# Identificatori



# Albero dei processi

Ogni processo ha il suo

- Process ID (PID)
- Parent Process ID (PPID) del processo che l'ha generato
- Process Group ID (PGID) del gruppo di processi al quale appartiene

Altri identificatori dei processi:

- REAL USER ID: il UID dell'utente che ha richiesto accesso
- REAL GROUP ID: il GID dell'utente che ha richiesto accesso
- EFFECTIVE USER ID: normalmente uguale al REAL USER ID, usato per la protezione file
- EFFECTIVE GROUP ID: normalmente uguale al REAL GROUP ID, usato per la protezione file

# FILE

Descritti dal blocco INODE:

- Informazioni sul possesso del file (Proprietario, gruppo )
- Tipo del file
- Permessi d'accesso (nel formato rwx rwx rwx)
- Istante dell'ultima modifica
- Istante dell'ultimo accesso
- Numero di link al file
- Dimensione del file
- I blocchi di cui é composto il file

Owner UID
Owner GID
...

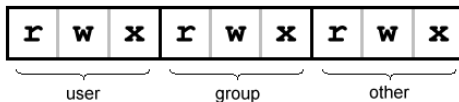
Proprietario del file



## Riassumendo:

Utenti	Processi	File
UID GID	PID PPID PGID Real user ID Effective User ID Real Group ID Effective Group ID	Owner UID Owner GID

# Protezione dei file



Campo FILE TYPE del blocco Inode: Permessi d'accesso al file

- 1 il **proprietario** (indicato con **U**, *user*) cioè l'unico utente che è proprietario (owner) del file, quello il cui User ID coincide con quello del file;
- 2 il **gruppo** (indicato con **G**, *group*) cioè l'insieme di tutti gli utenti che hanno lo stesso Group ID del file;
- 3 gli **altri** (indicato con **O**, *other*) cioè tutti gli utenti.

# Protezione dei file

```

if(effective_UID del processo == Owner_UID del file)
    usa i permessi di Owner;
else
    if(effective_GID del processo == Owner_GID del file)
        usa i permessi di Group;
    else
        usa i permessi di Altri;

```

```
$ ls -l ... -rw-r----- 1 mumolo 12 Oct 2 10:52 dati01 ...
```

proprietario	gruppo	altri
RWX	RWX	RWX
rw-	r--	---

```
shell:      rw-  r--  --- binario:  110  100  000 ottale:      6 4  0
```

il file `—/etc/passwd—` appartenente all'utente `—root—` avrà come file mode `—7 0 0—`

# Bit set-uid e set-gid

**Problema** In certi casi devo eseguire un programma di un altro utente.

Soluzione: se non sono dello stesso gruppo dell'altro utente, il file che contiene il programma potrebbe essere: `r-xr-xr-x`

**Altro problema** Se il programma deve leggere o scrivere un file dell'altro utente, ma il file é `rwX---`, come fare?

⇒ bit set-uid oppure set-gid

Soluzione: se set-UID settato, chi esegue il programma cambia l'EffectiveUserID al OwnerUID del file che contiene il programma.

