# 205 - Permisos avançats directoris

Curs 2020 - 2021

ASIX M01-ISO UF1-A01-03 Permisos avançats: SetGID / Stickybit


**Permisos** **1**

Descripció 1

Permisos avançats: diretoris 1

SetGID 1

Stickybit 4

Exercicis d'exemple 5


---

# Permisos

---

## Descripció

**Conceptes clau:**

- ❏ Permisos avançats de directoris
- ❏ SetGID
- ❏ Stickybit

**Ordres a treballar:**

- ❏ chmod

## Permisos avançats: diretoris

Consulteu el document *204 -Permisos Avançats* per saber l'establiment dels permisos avançats en format octal i simbòlic.

### SetGID

Sabem que tot element del sistema de fitxers pertany a un usuari i a un grup. Els permisos sobre l'element s'apliquen en tres blocs o categories: propietari, grup i altres. Imaginem que

uns quants usuaris volen portar a terme una tasca en conjunt i per fer-la han de compartir un directori i altres documents. Quina és la millor forma de fer-ho?

- El més lògic és crear un grup d'usuaris i assignar els usuaris que han de treballar conjuntament a aquest grup com a grup secundari.

- Crear un directori en un lloc d'accés per a tots els usuaris i assignar el directori al grup de treball.

Per exemple anem a crear un projecte anomenat 'projecte', es crearà un grup d'usuaris anomenat 'projecte' i el directori '/var/tmp/projecte'. S'afegiran al grup els usuaris que han de treballar en el projecte, per exemple el vostre usuari, un usuari anomenat user01 i l'usuari guest.

```
$ su -
Password:

[root@a36 ~]# groupadd projecte

[root@a36 ~]# usermod -aG projecte ecanet
[root@a36 ~]# usermod -aG projecte user01
[root@a36 ~]# usermod -aG projecte guest
```

```
$ getent group projecte
projecte:x:1033:ecanet,user01,guest

$ id ecanet
uid=1001(ecanet) gid=1001(ecanet) groups=1001(ecanet),10(wheel),975(docker),971(vagrant),1033(projecte)

$ id user01
uid=1029(user01) gid=1001(ecanet) groups=1001(ecanet),100(users),1032(vclub),1033(projecte)

$ id guest
uid=1000(guest) gid=1000(guest) groups=1000(guest),100(users),1033(projecte)
```

- com a root s'ha creat el grup projecte i s'hi han assignat els tres usuaris.

```
$ mkdir /var/tmp/projecte
$ ls -ld /var/tmp/projecte/
drwxrwxr-x 2 ecanet ecanet 4096 20 des 18:41 /var/tmp/projecte/

**tancar sessió dels usuaris si està oberta per recarregar la pertinença als grups, potser la sessió gràfica i tot!**

$ chgrp projecte /var/tmp/projecte/
$ ls -ld /var/tmp/projecte/
drwxrwxr-x 2 ecanet projecte 4096 20 des 18:41 /var/tmp/projecte/
```

- usant el vostre usuari (ecanet en l'exemple) assignar el directori de treball del projecte al grup projecte.

```
$ cd /var/tmp/projecte/
$ echo "projecte super guai!" > README.md
$ date > date.txt
```

```
[guest@a36 ~]$ cd /var/tmp/projecte/
[guest@a36 projecte]$ echo "autors: ecanet guest user01" > autors.txt
[guest@a36 projecte]$ cal > cal.txt
```

```
[guest@a36 projecte]$ ls -la
drwxrwxr-x  2 ecanet projecte 4096 Dec 20 18:50 .
drwxrwxrwt. 15 root   root    4096 Dec 20 18:50 ..
-rw-rw-r--  1 guest  guest     28 Dec 20 18:50 autors.txt
-rw-rw-r--  1 guest  guest    174 Dec 20 18:50 cal.txt
-rw-rw-r--  1 ecanet ecanet    28 Dec 20 18:49 date.txt
-rw-rw-r--  1 ecanet ecanet    21 Dec 20 18:49 README.md
```

- l'usuari ecanet ha generat dos fitxers dins del directori i l'usuari guest també.
- observeu que els fitxers creats pertanyen al usuari i grup de qui els ha creat.
- això genera el problema de com poden compartir entre els usuaris del grup els fitxers del projecte.

**Problema!**

Observeu que el directori on es desenvolupa el projecte entre varis usuaris pertany al grup projecte i com que tots els usuaris hi pertanyen i els permisos són rwxrwxr-x tenen permís de lectura/escriptura. Ara bé, cada usuari quan hi genera un fitxer ho fa amb el seu usuari i grup, de manera que els uns no poden accedir/modificar el contingut dels altres. Bé, podran o no en funció dels permisos other de cada fitxer (en l'exemple tenen r però no w).

Una solució incorrecta seria posar els permisos de others a rw però en general aquesta no és la solució buscada. Imaginem que és un projecte super secret i en realitat el que es vol és que others del directori i dels fitxers sigui ---.

### SetGID

La solució consisteix en activar el setGID al directori. Aquest permís en directoris provoca que tot el contingut que es crea de nou dins del directori pertany al mateix grup que el directori (i no pas al grup del propietari que crea l'element). Aquest permís s'aplica al bloc de permisos del grup i és necessari que el permís x estigui concedit.

És a dir, si el directori és del grup projecte i té el setGID activat, tots els fitxers i directoris que es creen de nou dins del directori passen a ser del grup projecte. Observeu que afecta als elements nous, no als que ja existeixen.

```
$ chmod 2770 /var/tmp/projecte/
$ ls -ld /var/tmp/projecte/
drwxrws--- 2 ecanet projecte 4096 20 des 18:50 /var/tmp/projecte/
```

```
[guest@a36 projecte]$ ls -la
total 24
drwxrws---  2 ecanet projecte 4096 Dec 20 18:50 .
drwxrwxrwt. 15 root   root    4096 Dec 20 18:50 ..
-rw-rw-r--  1 guest  guest     28 Dec 20 18:50 autors.txt
-rw-rw-r--  1 guest  guest    174 Dec 20 18:50 cal.txt
-rw-rw-r--  1 ecanet ecanet    28 Dec 20 18:49 date.txt
-rw-rw-r--  1 ecanet ecanet    21 Dec 20 18:49 README.md
```

- el propietari del directori (l'usuari ecanet en l'exemple) assigna els permisos 2770 que activen el setGID  i tanquen a altres l'accés.
- observeu que apareix la s en el lloc del permís x al bloc de permisos del grup.
- observei també que l'usuari guest continua tenint control total (igual que ecanet i user01) sobre el directori perquè és membre del grup projecte.

```
$ echo "nou fitxer de ecanet en el projecte" > estudi.pdf
$ mkdir exemples
```

```
[guest@a36 projecte]$ echo "nou document de guest" > informe.odt
[guest@a36 projecte]$ mkdir treballs
```

```
[guest@a36 projecte]$ ls -la
drwxrws---   4 ecanet projecte 4096 Dec 20 19:09 .
drwxrwxrwt. 15 root   root     4096 Dec 20 18:50 ..
-rw-rw-r--   1 guest  guest     28 Dec 20 18:50 autors.txt
-rw-rw-r--   1 guest  guest    174 Dec 20 18:50 cal.txt
-rw-rw-r--   1 ecanet ecanet    28 Dec 20 18:49 date.txt
-rw-rw-r--   1 ecanet projecte  36 Dec 20 19:08 estudi.pdf
drwxrwsr-x   2 ecanet projecte 4096 Dec 20 19:08 exemples
-rw-rw-r--   1 guest  projecte  22 Dec 20 19:09 informe.odt
-rw-rw-r--   1 ecanet ecanet    21 Dec 20 18:49 README.md
drwxrwsr-x   2 guest  projecte 4096 Dec 20 19:09 treballs
```

- l'usuari ecanet ha generat un fitxer i un directori i l'usuari guest un altre fitxer i un altre directori.
- els fitxers que ja existien continuen pertanyent al mateix grup, no s'han modificat.
- els fitxers nous que s'han creat (estudi.pdf i informe.odt) tenen com a propietari l'usuari que els ha creat però tenen com a grup el grup *projecte*.
- el mateix passa amb els dos directoris creats de nou, que tenen com a grup assignat el grup projecte.
- A més a més tots dos directoris nous hereten el permís setGID de manera que el seu contingut també pertany al grup *projecte*.

L'utlitat del permís ***setGID*** en directoris és fer que el contingut nou que s'afegeix al diretori pertany automàticament al mateix grup del directori. Això permet, per exemple, crear directoris de treball compartits entre els usuaris que són membres del grup. Però també es pot usar per generar directoris on els usuaris (alumnes per exemple) poden deixar-hi documents que passen a pertànyer a un grup (profes per exemple) del que no en formen part.

## Stickybit

Un altre dels permisos avançats aplicables a directoris és l'anomenat ***Stickybit*** que protegeix els fitxers del borrat accidental per part d'altres usuaris. Amb l'Sticky bit només el propietari de l'element pot eliminar-lo. Aquest permís s'aplica al bloc others i és necessari que el permís x estigui concedit (de fet el més sensat és tenir almenys la w i la x).

Imaginem un directori compartit amb permisos totals per a altres on tots els usuaris hi poden deixar documents (per exemple el lliurament de treballs dels alumnes), un alumne 'maldestre' pot esborrar accidentalment els documents dels altres alumnes. Per evitar-ho s'assigna al directori el permís especial de Stickybit que impedeix a un usuari esborrar elements d'altres usuaris encara que en tingui el permís w del directori.

Els directoris temporals del sistema tenen el permís Sticky bit activat per defecte: /var/tmp i /tmp

```
$ ls -ld /var/tmp/ /tmp/
drwxrwxrwt  18 root root  380 20 des 19:55 /tmp/
drwxrwxrwt. 15 root root 4096 20 des 19:45 /var/tmp/
```

```
$ date > /tmp/date.txt
```

```
[guest@a36 dades]$ ls -ld /tmp/ /tmp/date.txt
drwxrwxrwt 18 root   root   400 Dec 20 19:57 /tmp/
-rw-rw-r--  1 ecanet ecanet  28 Dec 20 19:57 /tmp/date.txt

[guest@a36 dades]$ rm /tmp/date.txt
rm: remove write-protected regular file '/tmp/date.txt'? y
rm: cannot remove '/tmp/date.txt': Operation not permitted
```

- l'usuari ecanet crea un fitxer anomenat date.txt dins del directori /tmp.
- aquest directori té permisos rwt per a altres, significa que té rwx i també el permís Sticky bit activat.
- l'usuari guest no pot esborrar el fitxer date.txt tot i tenir control total sobre el directori /tmp.

```
[guest@a36 dades]$ cal > /tmp/cal.txt
```

```
$ ls -ld /tmp/ /tmp/cal.txt
drwxrwxrwt 18 root  root  420 20 des 20:01 /tmp/
-rw-rw-r--  1 guest guest 174 20 des 20:00 /tmp/cal.txt

$ rm /tmp/cal.txt
rm: remove write-protected regular file '/tmp/cal.txt'? y
rm: cannot remove '/tmp/cal.txt': Operation not permitted
```

- l'usuari guest crea un fitxer dins del directori /tmp anomenat cal.txt
- l'usuari ecanet tot i tenir permisos de control total del directori /tmp no pot eliminar el fitxer perquè no és seu, l'Sticky bit no ho permet.

# Exercicis d'exemple

1. Crea un grup del sistema anomenat *sintesi* i assigna al grup el teu usuari, user01 i guest.

```
$ su -
Password:
```

```
[root@a36 ~]# groupadd sintesi

[root@a36 ~]# usermod -aG sintesi ecanet
[root@a36 ~]# usermod -aG sintesi user01
[root@a36 ~]# usermod -aG sintesi guest
```

```
[root@a36 ~]# getent group sintesi
sintesi:x:1034:ecanet,user01,guest

[root@a36 ~]# id ecanet
uid=1001(ecanet) gid=1001(ecanet)
groups=1001(ecanet),10(wheel),975(docker),971(vagrant),1033(projecte),1034(sintesi)

[root@a36 ~]# id user01
uid=1029(user01) gid=1001(ecanet) groups=1001(ecanet),100(users),1032(vclub),1033(projecte),1034(sintesi)

[root@a36 ~]# id guest
uid=1000(guest) gid=1000(guest) groups=1000(guest),100(users),1033(projecte),1034(sintesi)
```

** potser cal tornar a iniciar sessió?**

2. Crear un directori anomenat */tmp/sintesi*, assignar-lo al grup *sintesi* i modificar els permisos per no permetre cap mena d'accés a altres i activar el setGID.
   Amb el vostre usuari crear un *fitxer* i un *directori* dins.
   Amb l'usuari guest crear *dins* del directori creat per el vostre usuari un *fitxer* i un *directori*.

```
$ mkdir /tmp/sintesi

$ chgrp sintesi /tmp/sintesi

$ chmod 2770 /tmp/sintesi/

$ ls -ld /tmp/sintesi/
drwxrws--- 2 ecanet sintesi 40 20 des 19:30 /tmp/sintesi/
```

```
$ echo "creant un fitxer" > /tmp/sintesi/hola.txt

$ mkdir /tmp/sintesi/dades

$ ls -la /tmp/sintesi/
drwxrws---  3 ecanet sintesi  80 20 des 19:34 .
drwxrwxrwt 17 root   root    360 20 des 19:33 ..
drwxrwsr-x  2 ecanet sintesi  40 20 des 19:34 dades
-rw-rw-r--  1 ecanet sintesi  17 20 des 19:34 hola.txt
```

```
[guest@a36 dades]$ echo "guest creant un fitxer" > fitxer.txt

[guest@a36 dades]$ mkdir exemples

[guest@a36 dades]$ ls -la
drwxrwsr-x 3 ecanet sintesi 80 Dec 20 19:36 .
drwxrws--- 3 ecanet sintesi 80 Dec 20 19:34 ..
drwxrwsr-x 2 guest  sintesi 40 Dec 20 19:36 exemples
-rw-rw-r-- 1 guest  sintesi 23 Dec 20 19:36 fitxer.txt
```

- es crea al directori, es canvia de grup assignant-li el grip sintesi i es modifiquen els permisos activant el setGID.
- l'usuari ecanet crea un fitxer i un directori i tots dos passen a ser del grup sintesi.

- l'usuari guest fa actiu el directori que ha creat abans l'usuari ecanet. Dins seu crea un fitxer i un directori. Tot el que e screa pertany al grup sintesi perquè el permís s de setGID s'ha propagat als directoris de dins de sintesi.

3. Amb el teu usuari crea el directori /tmp/lliuraments on altres usuaris del sistema hi deixaran pràctiques (lliurament de treballs, pràctiques, apunts, etc). Aquest directori ha de tenir permís total per a tothom i tenir el setGID activat.
   (el directori pertany al grup per defecte de l'usuari).

```
$ mkdir /tmp/lliuraments

$ chmod 2777 /tmp/lliuraments/

$ ls -ld /tmp/lliuraments/
drwxrwsrwx 2 ecanet ecanet 40 20 des 19:41 /tmp/lliuraments/
```

4. Fes que el teu usuari deixi un document dins del directori i que també li deixin l'usuari guest i l'usuari root. Observar aquests documents a quin grup pertanyen.

```
$ echo "enunciat de l'examen de permisos" > /tmp/lliuraments/enunciat.pdf

[guest@a36 dades]$ echo "lliurament treball de permisos" > /tmp/lliuraments/guest-permisos.odt

[root@a36 ~]# echo "sóc root i estic a abu dabi" > /tmp/lliuraments/blackisblack.pdf

$ ls -la /tmp/lliuraments/
drwxrwsrwx  2 ecanet ecanet 100 20 des 19:45 .
drwxrwxrwt 18 root   root   380 20 des 19:45 ..
-rw-r--r--  1 root   ecanet  29 20 des 19:45 blackisblack.pdf
-rw-rw-r--  1 ecanet ecanet  33 20 des 19:43 enunciat.pdf
-rw-rw-r--  1 guest  ecanet  31 20 des 19:44 guest-permisos.odt
```

- els tres usuaris han deixat documents dins del directori de lliuraments i els tres documents pertanyen al grup ecanet perquè el directori pertany a aquest grup i té el setGID activat.
- observeu que NI guest NI root pertanyen al grup ecanet. A diferència de l'exercici anterior on el setGID s'utilitzava per fer un espai de treball comú, en quest exemple s'utilitza per 'apropiar-se' dels elements que s'hi lliuren.
- així l'usuari ecanet que no podia accedir als documents amb els permisos de altres (només hi ha r---) pot accedir al de guest perquè els permisos de grup li pertanyen.

5. Partint de l'exemple anterior amb un directori /tmp/lliuraments del vostre usuari feu que els usuaris ecanet, root i guest afegeixin un fitxer nou dins del directori (n'hi ha d'haver dos de cada usuari).
   A continuació amb l'usuari guest esborra un fitxer de root i un del vostre usuari. pot?

```
$ echo "segon fitxer de ecanet" > /tmp/lliuraments/efile.txt

[root@a36 ~]# echo "segon fitxer de root" > /tmp/lliuraments/rfile.txt

[guest@a36 dades]$ echo "un nou fitxer de guest" > /tmp/lliuraments/gfile.txt
```

```
[guest@a36 dades]$ ls -la /tmp/lliuraments/
drwxrwsrwx  2 ecanet ecanet 160 Dec 20 20:10 .
drwxrwxrwt 18 root   root   420 Dec 20 20:11 ..
-rw-r--r--  1 root   ecanet  29 Dec 20 19:45 blackisblack.pdf
-rw-rw-r--  1 ecanet ecanet  22 Dec 20 20:09 efile.txt
-rw-rw-r--  1 ecanet ecanet  33 Dec 20 19:43 enunciat.pdf
-rw-rw-r--  1 guest  ecanet  23 Dec 20 20:10 gfile.txt
-rw-rw-r--  1 guest  ecanet  31 Dec 20 19:44 guest-permisos.odt
-rw-r--r--  1 root   ecanet  21 Dec 20 20:10 rfile.txt
```

```
[guest@a36 dades]$ rm /tmp/lliuraments/efile.txt
rm: remove write-protected regular file '/tmp/lliuraments/efile.txt'? y

[guest@a36 dades]$ rm /tmp/lliuraments/rfile.txt
rm: remove write-protected regular file '/tmp/lliuraments/rfile.txt'? y

[guest@a36 dades]$ ls -la /tmp/lliuraments/
drwxrwsrwx  2 ecanet ecanet 120 Dec 20 20:11 .
drwxrwxrwt 18 root   root   420 Dec 20 20:11 ..
-rw-r--r--  1 root   ecanet  29 Dec 20 19:45 blackisblack.pdf
-rw-rw-r--  1 ecanet ecanet  33 Dec 20 19:43 enunciat.pdf
-rw-rw-r--  1 guest  ecanet  23 Dec 20 20:10 gfile.txt
-rw-rw-r--  1 guest  ecanet  31 Dec 20 19:44 guest-permisos.odt
```

- l'usuari guest si ha pogut esborrar els altres dos fitxers perquè el directori té permisos rwx per a altres i són els permisos que se li apliquen, per tant pot afegir, esborrar i canviar el nom dels elements del directori lliuraments.

6. Amb el vostre usuari protegir el directori lliuraments perquè els usuaris no puguin esborrar els elements dels altres usuaris. Continueu respectant el setGID del directori al mateix temps que afegiu l'Sticy bit.
   Verifiqueu que ara guest no pot esborrar els fitxers de root ni del vostre usuari, però sí els seus fitxers.

```
$ chmod +t /tmp/lliuraments/

$ ls -ld /tmp/lliuraments/
drwxrwsrwt 2 ecanet ecanet 120 20 des 20:11 /tmp/lliuraments/
```

```
[guest@a36 dades]$ rm /tmp/lliuraments/blackisblack.pdf
rm: remove write-protected regular file '/tmp/lliuraments/blackisblack.pdf'? y
rm: cannot remove '/tmp/lliuraments/blackisblack.pdf': Operation not permitted

[guest@a36 dades]$ rm /tmp/lliuraments/enunciat.pdf
rm: remove write-protected regular file '/tmp/lliuraments/enunciat.pdf'? y
rm: cannot remove '/tmp/lliuraments/enunciat.pdf': Operation not permitted

[guest@a36 dades]$ rm /tmp/lliuraments/guest-permisos.odt
```