

200 - Permisos bàsics

Curs 2020 - 2021

ASIX M01-ISO UF1-A01-03 Permisos bàsics

Permisos	2
Descripció	2
Permisos Bàsics	2
Permisos en el sistema de fitxers	2
Permisos Significatius	4
Establir permisos	6
Exercicis d'exemple	8

Permisos

Descripció

Conceptes clau:

- ☐ Permisos bàsics.
- ☐ Elements d'una entrada de directori
- ☐ Diferència entre fitxers i directoris.
- ☐ Chmod octal i simbòlic
- ☐ Chown / chgrp
- ☐ Umask
- ☐ Permisos avançats fitxers
- ☐ Permisos avançats directoris

Ordres a treballar:

- ☐ ls -l
- ☐ stat
- ☐ chmod
- ☐ chown
- ☐ chgrp
- ☐ umask

Permisos Bàsics

Permisos en el sistema de fitxers

En GNU/Linux els elements del sistema pertanyen sempre a un usuari i a un grup. Es diu que l'usuari és el propietari (owner) de l'element i el grup és el grup al qual pertany l'element. Això passa no només a fitxers i directoris sinó també a altres elements com per exemple els processos. És a dir, tot element té un usuari i un grup que en són els propietaris.

Per exemple es pot fer un llistat llarg del directori /tmp i observar:

```
$ ls -l /tmp/
total 12
-rw-rw-r--. 1 root  root 1214 Nov 27 08:06 krb5cc_0
-rw-rw-r--. 1 ecanet inf 664 Nov 27 12:23 krb5cc_100366_1E8jVI
-rw-rw-r--. 1 root  root 787 Nov 27 08:06 krb5ccmachine_INFORMATICA.ESCOLADELTREBALL.ORG
drwxr-xr-x. 2 ecanet inf 40 Nov 27 12:02 mozilla_ecanet0
drwxr-xr-x. 2 ecanet inf 60 Nov 27 08:06 ssh-vwibwNdDeuny
drwxr-xr-x. 3 root  root 60 Nov 27 08:05 systemd-private-66c590cbc333473da9a6bd21b611d1be-chrond.service-2bbAEh
```

Del llistat anterior s'ha destacat un fitxer que és propietat de l'usuari *ecanet* i del grup *inf*. El fitxer és propietat de *ecanet* i el fitxer pertany al grup *inf*. També es pot veure llistat els permisos en format simbòlic *rw-----*.

usuari propietari

L'usuari al que pertany l'element. En fer un llistat llarg és la tercera columna.

grup propietari

El grup al qual pertany l'element. En fer un llistat llarg és la quarta columna.

permisos

A tot element del sistema de fitxers se li assignen permisos que determinen el tipus d'accés que es pot fer (lectura, escriptura/modificació, execució i sense permís) segons qui pretén fer l'accés (propietari, grup o altres).

Els *permisos* d'un element es mostren en format simbòlic amb una cadena com per exemple *rwxr-x---*. Podem observar que apareixen indicats permisos de lectura o *read*, d'escriptura o *write*, d'execució o *execute* i denegació de permís o *none*.

- r* el permís *read* indica permís de lectura. Es pot accedir al contingut per llegir-lo.
- w* el permís *write* indica permís d'escriptura. En realitat significa escriure, modificar i esborrar.
- x* el permís *execute* significa o bé el permís d'execució quan es tracta d'un programa executable o bé el permís de drets administratius quan es tracta d'un directori (el permís d'entrar-hi).
- el permís *none* (el guió) significa que no s'ha concedit permís.

Els *permisos* d'un element es mostren en format simbòlic amb una cadena com per exemple *rwxr-x---*. Aquesta cadena indica els permisos d'accés per al *propietari*, per al *grup* o per a *altres* usuaris. De et els podem interpretar en tres blocs de tres caràcters com per exemple així: *rw**x* *r*-*x* *---*

owner group others

- owner* els tres primers caràcters de permisos indiquen els permisos que afecten al *owner*, al propietari del element. En l'exemple serien *rw**x*.
- group* els tres següents caràcters de permisos indiquen els permisos que afecten al *group*, al grup propietari del element. En l'exemple serien *r*-*x*.
- others* els tres últims caràcters de permisos indiquen els permisos que afecten a *others*, a tots els altres usuaris que no són ni el propietari ni usuaris que pertanyen al grup propietari. En l'exemple srien *---*.

Permisos Significatius

Els permisos actuen diferent si es tracta de fitxers o de directoris, de manera que cal analitzar cada cas per separat. Entenem per permisos significatius els permisos més usuals i que tenen més 'sentit'. Es pot establir qualsevol combinació de permisos (per exemple `r---w----x`) però algunes combinacions no tenen ni caps ni peus!.

Permisos significatius de fitxers

Aquestes són les combinacions de permisos de fitxers més usuals, amb més 'sentit':

- `rw-` permís total entenent total com a permís de lectura i escriptura. Amb aquest permís el fitxer es pot llegir i modificar. Observeu que diem total però no inclou la `x` (el permís d'execució)
- `r--` significa que el fitxer té permís de lectura però no de modificació. Es pot veure el contingut del fitxer però no modificar-lo.
- `---` aquest és el permís restrictiu del tot, no es permet res. Quan un fitxer té aquest permís assignat no es pot ni observar el seu contingut ni modificar-lo.

Exemples de permisos de fitxers:

(nota: s'ha espaiat els permisos per facilitar-ne la visualització)

`rw- r-- ---`

Un fitxer amb aquests permisos seria de control total pel propietari (que el podrà veure i editar); de visualització per al grup (els usuaris que pertanyen al mateix grup que el fitxer poden veure el contingut del fitxer); i denegació total a altres (els altres usuaris que no són ni el propietari ni pertanyen al grup del fitxer es foten!).

`rw- --- ---`

Amb uns permisos de fitxer com aquests diríem que el propietari és molt 'reservat'. Només ell pot veure i editar el contingut. Per a qualsevol altre usuari el fitxer està xapat.

`rw- rw- r--`

Un fitxer amb aquests permisos és modificable i visualitzable pel propietari i pels usuaris que pertanyen al mateix grup que el fitxer. La resta d'usuaris en poden veure el contingut, però no modificar-lo.

Observeu que els tres exemples permeten veure que tal 'amigable' és el fitxer, usualment el propietari pot fer de tot (`rw-`) i si és 'col·lega' pot deixar al grup també fer de tot (`rw-`) i als altres mirar (`r--`). Un usuari reservat i que vol mantenir la privacitat pot posar (`rw-`) per a ell, (`r--`) per al grup i xapar (`---`) altres. Un usuari molt reservat pot donar-se permís a ell (`rw-`) i xapar la resta (grup `---` i altres `---`).

Atenció: Observeu que quan hem descrit els permisos significatius de fitxers hem OBVIAT la `x`. En general no considerem mai el permís d'*execució* `x` quan parlem de

fitxers. No s'aplica mai per defecte. La **x** en un fitxer significa permís d'**execució** i s'ha d'assignar només a fitxers executables. Es recomana fer-ho sempre manualment, discrecionalment, i mai per defecte. És a dir, si volem que un fitxer que és un programa tingui activat el permís d'**execució x** cal fer-ho manualment (individualment) per exemple amb l'ordre *chmod +x*.

Permisos significatius de directoris

Igual que passa amb els fitxers als directoris també se'ls pot assignar qualsevol combinació de permisos, però no tots tenen massa sentit comú.

Aquestes són les combinacions de permisos de directoris més usuals, amb més 'sentit':

- rwX*** permís total de lectura, escriptura i accés. Amb aquest permís el directori es pot llegir i modificar i s'hi pot entrar dins (la **x** proporciona els drets administratius).
- r-X*** significa que el directori concedeix el permís d'entrar-hi i de llistar el contingut, però no de modificar-lo. Aquest és el permís clàssic per deixar que es pugui accedir al directori però no permetre que es modifiqui. Modificar un directori significa poder afegir i eliminar elements i poder canviar-los el nom.
- aquest és el permís restrictiu del tot, no es permet res. Quan un directori té aquest permís assignat no es pot entrar dins, ni llistar ni modificar.

Exemples de permisos de directoris:

(nota: s'ha espaiat els permisos per facilitar-ne la visualització)

rwX r-X ---

Un directori amb aquests permisos seria de control total pel propietari; d'accés per al grup (els usuaris que pertanyen al mateix grup que el directori poden entrar-hi i poden llistar-ne el contingut); i denegació total a altres (els altres usuaris que no són ni el propietari ni pertanyen al grup del directori es foten!).

rwX --- ---

Amb uns permisos de directori com aquests diríem que el propietari és molt 'reservat'. Només ell té control total del directori. Per a qualsevol altre usuari el directori està xapat.

rwX rwX r-X

Un directori amb aquests permisos és modificable i visualitzable pel propietari i pels usuaris que pertanyen al mateix grup que el directori. La resta d'usuaris poden accedir-hi i llistar-lo però no el poden modificar.

En Resum:

- Quan es vol concedir el permís total (usualment al propietari) els permisos per a fitxers són **rw-** i per a directoris **rwX**.
- Quan es vol permetre mirar però no tocar els permisos per a fitxers són **r--** i per a directoris són **r-X**.
- Quan es vol tancar tot i no concedir permisos s'utilitza **---**.
- Observeu que mentre que per a fitxers no assignem mai el permís x (a no ser que sigui un executable), per directoris cal sempre assignar-lo per poder-hi treballar bé.

Establir permisos

L'ordre **chmod** permet establir permisos tant a fitxers com a directoris. Permet usar dues notacions, la notació **octal** que és la més utilitzada i la notació **simbòlica**.

Octal

Amb la notació octal l'ordre chmod assigna un valor en octal (base vuit) per als permisos de owner, un altre per als permisos de group i un altre per als de other. Per tant calen tres dígits octals. S'utilitza un valor en base vuit perquè es representen els permisos amb un valor numèric de la següent manera:

421

rwX Al permís r se li assigna un valor de 4, al permís w un valor de 2 i al permís de 1. D'aquesta manera un valor del 0 al 7 indica una determinada combinació de permisos

- Així si per exemple es vol assignar el permís **rw-** numèricament s'avalua r(4), w(2) i -(0) de manera que el valor octal a usar és **6**.
- Si es vol assignar el permís **r--** en octal equival a r(4), -(0) i -(0), és a dir, al valor octal **4**.
- Si es vol assignar el permís **r-X** en octal equival al valor **5** que surt de r(4), -(0) i x(1).
- El permís més simple és **---** que en octal equival a **0**.

Exemple d'assignació de permisos a fitxers:

chmod 640 file.txt

Significa permisos rw- r-- --. Aquests permisos surten del 6 per al propietari (6 equival a rw-), 4 per al grup (4 equival a r--) i 0 per a altres (0 equival a ---).

chmod 664 file.txt

Significa permisos rw-rw-r--. Aquests permisos surten del 6 per al propietari i el grup (6 equival a rw-) i 4 per a altres (4 equival a r--).

chmod 600 file.txt

Significa permisos rw-----. Aquests permisos surten del 6 per al propietari i el grup (6 equival a rw-).

Exemple de permisos per a directoris:

chmod 750 operatius

Significa permisos rwxr-xr-x. Aquests permisos surten de 7 per al propietari (7 equival a rwx), 5 per al grup (5 equival a r-x) i cap per a altres (o equival a ---).

chmod 755 operatius

Significa permisos rwxr-xr-x. Aquests permisos surten de 7 per al propietari (7 equival a rwx), 5 per al grup i altres (5 equival a r-x).

chmod 700 operatius

Significa permisos rwxr-xr-x. Aquests permisos surten de 7 per al propietari (7 equival a rwx), 0 per al grup i altres (0 equival a ---).

Combinacions significatives:

De fitxers:	De directoris
6 rw-	7 rwx
4 r--	5 r-x
0 ---	0 ---

Exemples d'assignació de permisos a fitxers:

```
[ecanet@a36 ~]$ ls -l
-rw-rw-r-- 1 ecanet ecanet 3539 4 nov 12:36 apunts.txt

[ecanet@a36 ~]$ chmod 640 apunts.txt
[ecanet@a36 ~]$ ls -l apunts.txt
-rw-r----- 1 ecanet ecanet 3539 4 nov 12:36 apunts.txt

[ecanet@a36 ~]$ chmod 600 apunts.txt
[ecanet@a36 ~]$ ls -la apunts.txt
-rw----- 1 ecanet ecanet 3539 4 nov 12:36 apunts.txt
```

- Primerament s'observa que els permisos actuals del fitxer apunts.txt són rw-rw-r-- que equivalen a 664. És a dir, control total de propietari i grup i lectura per a altres.
- Es convien per rw-r----- que equivalen en octal a 640. És a dir, control total pel propietari, lectura per al grup i cap per a altres.
- Finalment es torna encara més restrictiu i assigna els permisos rw----- equivalents a 600. És a dir, control total pel propietari i cap permís per al grup ni per a altres.

Exemples d'assignació de permisos a directoris:

```
[ecanet@a36 ~]$ ls -ld operatius/
drwxrwxr-x 2 ecanet ecanet 4096 18 des 18:05 operatius/

[ecanet@a36 ~]$ chmod 755 operatius/
[ecanet@a36 ~]$ ls -ld operatius/
drwxr-xr-x 2 ecanet ecanet 4096 18 des 18:05 operatius/

[ecanet@a36 ~]$ chmod 750 operatius/
[ecanet@a36 ~]$ ls -ld operatius/
drwxr-x--- 2 ecanet ecanet 4096 18 des 18:05 operatius/

[ecanet@a36 ~]$ chmod 700 operatius/
[ecanet@a36 ~]$ ls -ld operatius/
drwx----- 2 ecanet ecanet 4096 18 des 18:05 operatius/
```

- Primerament es llisten els permisos del directori operatius que són rwxrwxr-x equivalents al valor octal 775. És a dir, control total per a propietari i grup i accés/llistat per a altres.
- Tot seguit es modifiquen els permisos per establir-los a rwx-r-xr-x equivalents al valor octal 755. És a dir, control total pel propietari, i d'accés/lectura per a grup i altres.
- En una nova ordre chmod s'assignen els permisos rwx-r-x--- equivalents al valor octal 750. És a dir, control total per al propietari, accés/lectura per al grup i cap per a altres.
- L'última acció assigna al directori operatius els permisos rwx----- equivalents al valor octal 700. És a dir, control total per al propietari i cap per al grup i altres.

Exercicis d'exemple

Primerament heu de fer l'exercici **00-Creació_de_estructura** per generar els directoris i els fitxers necessaris per fer aquest exercici.

1. Fer actiu el directori [/tmp/mp1/operatius/apunts](#).

Des d'aquest directori realitzarem tots els exercicis, tots. No és permès de canviar de directori amb l'ordre cd. Totes les ordres del sistema es poden realitzar des de qualsevol directori actiu.

Després de fer aquest directori actiu verificar-ho amb l'ordre *pwd*.

```
$ cd /tmp/m01/operatius/apunts

$ pwd
/tmp/m01/operatius/apunts
```

Recordeu quina és l'estructura de directoris i fitxers que estem utilitzant:

```
$ tree /tmp/m01
/tmp/m01
├── operatius
│   └── apunts
```



```

├── carta.txt
├── dades.pdf
├── dossier.odt
├── informe.pdf
├── projecte.odt
├── treball.txt
├── exercicis
└── xarxes

```

```

$ tree /var/tmp/prova/
/var/tmp/prova/

```

2. Llistar el directori actiu i **observar els permisos**

```

$ ls -l
-rw-rw-r-- 1 ecanet ecanet 113 18 des 18:21 carta.txt
-rw-rw-r-- 1 ecanet ecanet 28 18 des 18:21 dades.pdf
-rw-rw-r-- 1 ecanet ecanet 21448 18 des 18:21 dossier.odt
-rw-rw-r-- 1 ecanet ecanet 89 18 des 18:21 informe.pdf
-rw-rw-r-- 1 ecanet ecanet 134 18 des 18:21 projecte.odt
-rw-rw-r-- 1 ecanet ecanet 1159 18 des 18:21 treball.txt

```

3. Com a **root** crear l'usuari **user01** que pertany al mateix grup que el vostre usuari i validar que existeix l'usuari **guest** i que pertany a un altre grup.

```

[ecanet@a36 m01]$ su -
Password:

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

[root@a36 ~]# useradd -g ecanet user01
[root@a36 ~]# passwd user01
Changing password for user user01.
New password:
BAD PASSWORD: The password is shorter than 8 characters
Retype new password:
passwd: all authentication tokens updated successfully.

[root@a36 ~]# id user01
uid=1029(user01) gid=1001(ecanet) groups=1001(ecanet)

[root@a36 ~]# id guest
uid=1000(guest) gid=1000(guest) groups=1000(guest)

```

- primerament inicia sessió com a administrador del sistema amb l'usuari **root** amb l'ordre **su -**.
- mostrar la informació de l'usuari **ecanet**, en especial observar que pertany al grup **1001(ecanet)**.
- crear l'usuari **user01** assignant-li el grup **ecanet** (l'opció **-g**) i assignar-li un password.
- observar que l'usuari pertany al grup **ecanet** igual que l'usuari **ecanet**.
- observar que l'usuari **guest** pertany a un altre grup, al grup **1000(guest)**.

4. Assignar al fitxer **carta.txt** els permisos de **control total** per al propietari, grup i altres.

```

$ chmod 666 carta.txt
$ ls -l carta.txt

```

```
-rw-rw-rw- 1 ecanet ecanet 113 18 des 18:21 carta.txt
```

```
$ cat carta.txt
```

```
bin  
boot  
...
```

```
[user01@a36 ~]$ cd /tmp/m01/operatius/apunts/
```

```
[user01@a36 apunts]$ echo "hola sóc user01 i puc modificar" >> carta.txt
```

```
[user01@a36 apunts]$ cat carta.txt
```

```
bin  
boot  
...  
hola sóc user01 i puc modificar
```

```
[guest@a36 ~]$ cd /tmp/m01/operatius/apunts/
```

```
[guest@a36 apunts]$ echo "hola soc guest i puc modificar carta" >> carta.txt
```

```
[guest@a36 apunts]$ cat carta.txt
```

```
bin  
boot  
...  
hola sóc user01 i puc modificar  
hola soc guest i puc modificar carta
```

- primerament com a propietari de l'element es modifiquen els permisos de carta.txt donant permís a tothom de tot.
- l'usuari user01 pot modificar i veure el contingut de carta.txt.
- l'usuari guest també pot modificar i veure el contingut de carta.txt.

5. Assignar al fitxer carta.txt els permisos de control total per al propietari, de lectura per al grup i de cap per a altres.

```
$ chmod 640 carta.txt
```

```
$ ls -l carta.txt
```

```
-rw-r----- 1 ecanet ecanet 183 18 des 18:41 carta.txt
```

```
$ cat carta.txt
```

```
bin  
boot  
...
```

```
[user01@a36 apunts]$ echo "hola sóc user01 i NO puc modificar carta" >> carta.txt
```

```
-bash: carta.txt: Permission denied
```

```
[user01@a36 apunts]$ cat carta.txt
```

```
bin  
boot  
...
```

```
[user01@a36 apunts]$ cp carta.txt newcarta.txt
```

```
[user01@a36 apunts]$ ls -l newcarta.txt
```

```
-rw-r----- 1 user01 ecanet 183 Dec 18 18:50 newcarta.txt
```

```
[guest@a36 apunts]$ echo "hola soc guest i puc ni mirar ni modificar carta" >> carta.txt
```

```
-bash: carta.txt: Permission denied
```

```
[guest@a36 apunts]$ cat carta.txt
```

```
cat: carta.txt: Permission denied
```

- primerament es modifiquen els permisos de carta.txt i es mostra el seu contingut.
- l'usuari *user01* no és el propietari però si pertany al mateix *grup* que el fitxer, per tant se li apliquen els permisos de grup *r--*. Amb aquests permisos no pot modificar el fitxer però si veure el seu contingut o copiar-lo.
- l'usuari *guest* no és ni el propietari ni pertany al mateix grup que el fitxer, per tant se li apliquen els permisos de *other*. En aquest cas els permisos *---* de manera que no pot ni modificar ni accedir al contingut del fitxer.

6. Assignar al fitxer carta.txt els permisos de lectura per al propietari i de cap per al grup i altres.

```
$ chmod 400 carta.txt
$ ls -l carta.txt
-r----- 1 ecanet ecanet 183 18 des 18:41 carta.txt

$ echo "hola sóc el propietari" >> carta.txt
bash: carta.txt: Permission denied

$ cat carta.txt
bin
boot
...
```

```
[user01@a36 apunts]$ echo "hola sóc user01 i NO puc accedir a carta" >> carta.txt
-bash: carta.txt: Permission denied

[user01@a36 apunts]$ cat carta.txt
cat: carta.txt: Permission denied
```

```
[guest@a36 apunts]$ echo "hola soc guest i puc ni mirar ni modificar carta" >> carta.txt
-bash: carta.txt: Permission denied

[guest@a36 apunts]$ cat carta.txt
cat: carta.txt: Permission denied
```

7. Com usuari *user01* crear un fitxer anomenat *newcarta.txt* amb el contingut d'executar l'ordre "*uname -a*". Observa els permisos amb què es crea el fitxer. Pot el vostre usuari (ecanet en l'exemple) veure el contingut i modificar-lo? Pot guest?

```
[user01@a36 apunts]$ uname -a > newcarta.txt

[user01@a36 apunts]$ ls -l newcarta.txt
-rw-r----- 1 user01 ecanet 134 Dec 18 19:03 newcarta.txt
```

```
$ echo "hola radiola" >> newcarta.txt
bash: newcarta.txt: Permission denied

$ cat newcarta.txt
Linux a36.informatica.escoladeltreball.org 4.18.19-100.fc27.x86_64 #1 SMP Wed Nov 14 22:04:34 UTC 2018
x86_64 x86_64 x86_64 GNU/Linux
```

```
[guest@a36 apunts]$ echo "adeu andreu" >> newcarta.txt
-bash: newcarta.txt: Permission denied

[guest@a36 apunts]$ cat newcarta.txt
cat: newcarta.txt: Permission denied
```

- primerament l'usuari **user01** crea el fitxer newcarta.txt. En aquest exemple el fitxer s'ha creat amb els permisos per defecte rw-r----- però això pot variar segons el valor de la umask (s'estudia més endavant).
- amb aquests permisos l'usuari ecanet només pot llegir el contingut de newcarta.txt però no modificar-lo, ja que pertany al mateix grup que el fitxer.
- en canvi l'usuari guest que no és el propietari ni pertany al mateix grup que el fitxer no pot fer res, se li apliquen els permisos de others.

8. Fer actiu el directori /var/tmp/operatius en les sessions de cada un dels tres usuaris amb els que s'està treballant.

<pre>\$ pwd /tmp/m01/operatius/apunts \$ cd .. \$ pwd /tmp/m01/operatius</pre>
<pre>[user01@a36 apunts]\$ pwd /tmp/m01/operatius/apunts [user01@a36 apunts]\$ cd .. [user01@a36 operatius]\$ pwd /tmp/m01/operatius</pre>
<pre>[guest@a36 apunts]\$ pwd /tmp/m01/operatius/apunts [guest@a36 apunts]\$ cd .. [guest@a36 operatius]\$ pwd /tmp/m01/operatius</pre>

9. Usant el vostre usuari (ecanet en l'exemple) assignar al **directori apunts** el permís total al propietari i al grup, i d'accés/lectura a altres.

<pre>\$ chmod 755 apunts/ \$ ls -ld apunts/ drwxr-xr-x 2 ecanet ecanet 180 18 des 18:50 apunts/ \$ ls -la apunts/ drwxr-xr-x 2 ecanet ecanet 180 18 des 18:50 . drwxrwxr-x 4 ecanet ecanet 80 18 des 18:21 .. -r----- 1 ecanet ecanet 183 18 des 18:41 carta.txt -rw-rw-r-- 1 ecanet ecanet 28 18 des 18:21 dades.pdf -rw-rw-r-- 1 ecanet ecanet 21448 18 des 18:21 dossier.odt -rw-rw-r-- 1 ecanet ecanet 89 18 des 18:21 informe.pdf -rw-r----- 1 user01 ecanet 134 18 des 19:03 newcarta.txt -rw-rw-r-- 1 ecanet ecanet 134 18 des 18:21 projecte.odt -rw-rw-r-- 1 ecanet ecanet 1159 18 des 18:21 treball.txt</pre>
<pre>[user01@a36 operatius]\$ ls -la apunts/ drwxr-xr-x 2 ecanet ecanet 180 Dec 18 18:50 . drwxrwxr-x 4 ecanet ecanet 80 Dec 18 18:21 .. -r----- 1 ecanet ecanet 183 Dec 18 18:41 carta.txt -rw-rw-r-- 1 ecanet ecanet 28 Dec 18 18:21 dades.pdf -rw-rw-r-- 1 ecanet ecanet 21448 Dec 18 18:21 dossier.odt -rw-rw-r-- 1 ecanet ecanet 89 Dec 18 18:21 informe.pdf -rw-r----- 1 user01 ecanet 134 Dec 18 19:03 newcarta.txt -rw-rw-r-- 1 ecanet ecanet 134 Dec 18 18:21 projecte.odt -rw-rw-r-- 1 ecanet ecanet 1159 Dec 18 18:21 treball.txt [user01@a36 operatius]\$ echo "hola" > apunts/fileuser01.txt -bash: apunts/fileuser01.txt: Permission denied [user01@a36 operatius]\$ rm apunts/dades.pdf rm: cannot remove 'apunts/dades.pdf': Permission denied</pre>

```
[user01@a36 operatius]$ mv apunts/newcarta.txt apunts/newname.txt
mv: cannot move 'apunts/newcarta.txt' to 'apunts/newname.txt': Permission denied
```

```
[user01@a36 operatius]$ cd apunts/
```

```
[user01@a36 apunts]$ pwd
/tmp/m01/operatius/apunts
```

```
[user01@a36 apunts]$ cd ..
```

```
[guest@a36 operatius]$ ls -la apunts/
drwxr-xr-x 2 ecanet ecanet 180 Dec 18 18:50 .
drwxrwxr-x 4 ecanet ecanet 80 Dec 18 18:21 ..
-r----- 1 ecanet ecanet 183 Dec 18 18:41 carta.txt
-rw-rw-r-- 1 ecanet ecanet 28 Dec 18 18:21 dades.pdf
-rw-rw-r-- 1 ecanet ecanet 21448 Dec 18 18:21 dossier.odt
-rw-rw-r-- 1 ecanet ecanet 89 Dec 18 18:21 informe.pdf
-rw-r----- 1 user01 ecanet 134 Dec 18 19:03 newcarta.txt
-rw-rw-r-- 1 ecanet ecanet 134 Dec 18 18:21 projecte.odt
-rw-rw-r-- 1 ecanet ecanet 1159 Dec 18 18:21 treball.txt
```

```
[guest@a36 operatius]$ echo "soc guest" >> apunts/hola.txt
-bash: apunts/hola.txt: Permission denied
```

```
[guest@a36 operatius]$ rm apunts/dades.pdf
rm: remove write-protected regular file 'apunts/dades.pdf'? y
rm: cannot remove 'apunts/dades.pdf': Permission denied
```

```
[guest@a36 operatius]$ cd apunts/
```

```
[guest@a36 apunts]$ pwd
/tmp/m01/operatius/apunts
```

```
[guest@a36 apunts]$ cd ..
```

- primerament l'usuari ecanet estableix els permisos del directori apunts a 755.
- es poden observar els permisos del directori amb l'opció -d per indicar directori.
- si es fa un llistat del directori amb els fitxers ocults usant l'opció -a es pot observar l'entrada corresponent a l'element punt ".". Aquest element és el propi directori.
- l'usuari user01 té assignats els permisos r-x perquè no és el propietari i pertany al mateix grup que el fitxer. Pot llistar el contingut del directori i fer-lo directori actiu, però no el pot modificar.
- no poder modificar vol dir que no es poden afegir, eliminar ni modificar els nom dels elements.
- l'usuari guest també té permisos r-x corresponents a others. També pot llistar el directori i fer-lo actiu, però fer-hi modificacions.

10. Assignar al directori apunts permís d'accés/lectura per al propietari i cap permís per al grup i altres.

```
$ chmod 500 apunts/
```

```
$ ls -ld apunts/
dr-x----- 2 ecanet ecanet 180 18 des 18:50 apunts/
```

```
$ echo "hola soc el propietari" >> apunts/hola.txt
bash: apunts/hola.txt: Permission denied
```

```
$ cd apunts/
```

```
$ pwd
/tmp/m01/operatiu/apunts
```

```
$ cd ..
```

```
[user01@a36 operatiu]$ ls apunts/
ls: cannot open directory 'apunts/': Permission denied
```

```
[user01@a36 operatiu]$ cd apunts/
-bash: cd: apunts/: Permission denied
```

```
[guest@a36 operatiu]$ ls -la apunts/
ls: cannot open directory 'apunts/': Permission denied
```

```
[guest@a36 operatiu]$ cd apunts/
-bash: cd: apunts/: Permission denied
```

- primerament el propietari assigna els permisos 500 al directori apunts.
- el propietari no pot crear nous fitxers dins del directori perquè està protegit contra escriptura, no té el permís w.
- si pot llistar el directori i entrar-hi dins. La x proporciona el permís per accedir-hi amb cd i la combinació rx proporciona el permís per llistar.
- tant l'usuari user01 com guest no poden accedir al directori ni modificar-lo ni llistar-lo perquè no tenen cap permís.

11. El propietari del directori apunts concedeix permís total sobre el directori al propietari i al grup i permís **només r** a altres (atenció, no r-x sinó r--). Què pot fer guest?

```
$ chmod 774 apunts/
```

```
$ ls -ld apunts/
drwxrwxr-- 2 ecanet ecanet 180 18 des 18:50 apunts/
```

```
[guest@a36 operatiu]$ ls -l apunts/
ls: cannot access 'apunts/newcarta.txt': Permission denied
ls: cannot access 'apunts/dades.pdf': Permission denied
ls: cannot access 'apunts/dossier.odt': Permission denied
ls: cannot access 'apunts/projecte.odt': Permission denied
ls: cannot access 'apunts/informe.pdf': Permission denied
ls: cannot access 'apunts/treball.txt': Permission denied
ls: cannot access 'apunts/carta.txt': Permission denied
total 0
```

```
-????????? ? ? ? ?      ? carta.txt
-????????? ? ? ? ?      ? dades.pdf
-????????? ? ? ? ?      ? dossier.odt
-????????? ? ? ? ?      ? informe.pdf
-????????? ? ? ? ?      ? newcarta.txt
-????????? ? ? ? ?      ? projecte.odt
-????????? ? ? ? ?      ? treball.txt
```

```
[guest@a36 operatiu]$ cd apunts
-bash: cd: apunts: Permission denied
```

- primerament el propietari assigna al directori apunts els permisos 774.
- l'usuari guest pot accedir al directori amb els permisos de others que són r--.
- aquests permisos permeten llistar però NO entrar al directori, ja que falta la x. Per això no es permet el cd.
- el permís de llistar r si va sol sense la x tampoc és útil perquè no pot accedir als atributs de cada fitxer. Observeu que mostra el nom dels elements però no pot mostrar els seus atributs.

nota per tenir sentit el permís r en els directoris ha de ser acompanyat de la x.

12. Modificar el permís del directori apunts concedint a altres només el permís x (sense concedir el permís r). Què pot fer l'usuari guest?

```
$ chmod 771 apunts/

$ ls -ld apunts/
drwxrwx--x 2 ecanet ecanet 180 18 des 18:50 apunts/

[guest@a36 operatius]$ cd apunts/

[guest@a36 apunts]$ ls
ls: cannot open directory '.': Permission denied

[guest@a36 apunts]$ ls -la
ls: cannot open directory '.': Permission denied

[guest@a36 apunts]$ cd ..
```

- primerament el propietari assigna al directori apunts els permisos 771.
- l'usuari guest pot fer actiu el directori apunts entrant-hi amb l'ordre cd ja que té concedit el permís x. Però no pot llistar el contingut, gens ni mica, perquè no té el permís r.

13. Assignar el directori apunts el permís r-x per a others i observar què pot fer guest.

```
$ chmod 775 apunts

$ ls -ld apunts/
drwxrwxr-x 2 ecanet ecanet 180 18 des 18:50 apunts/

[guest@a36 operatius]$ ls apunts/
carta.txt dades.pdf dossier.odt informe.pdf newcarta.txt projecte.odt treball.txt

[guest@a36 operatius]$ cd apunts

[guest@a36 apunts]$ pwd
/tmp/m01/operatius/apunts

[guest@a36 apunts]$ cd ..

[guest@a36 operatius]$ echo "hola que tal" > apunts/hola.txt
-bash: apunts/hola.txt: Permission denied
```

- Ara finalment l'usuari guest pot llistar i fer actiu el directori apunts en tenir permisos r-x. El que no pot fer és modificar el directori.

14. Assignar al directori apunts permisos de control total per a others. Observar què pot fer guest.

```
$ chmod 777 apunts

$ ls -ld apunts/
drwxrwxrwx 2 ecanet ecanet 180 18 des 18:50 apunts/

[guest@a36 operatius]$ cd apunts/
```

```

[guest@a36 apunts]$ ls
carta.txt dades.pdf dossier.odt informe.pdf newcarta.txt projecte.odt treball.txt

[guest@a36 apunts]$ echo "hola que tal" > hola.txt
[guest@a36 apunts]$ ls
carta.txt dades.pdf dossier.odt hola.txt informe.pdf newcarta.txt projecte.odt treball.txt

[guest@a36 apunts]$ mv hola.txt adeu.txt
[guest@a36 apunts]$ ls
adeu.txt carta.txt dades.pdf dossier.odt informe.pdf newcarta.txt projecte.odt treball.txt

[guest@a36 apunts]$ rm adeu.txt

[guest@a36 apunts]$ rm informe.pdf
rm: remove write-protected regular file 'informe.pdf'? y

[guest@a36 apunts]$ ls
carta.txt dades.pdf dossier.odt newcarta.txt projecte.odt treball.txt

```

- ara que els permisos per a others són rwx l'usuari guest pot també modificar el contingut del directori. És a dir, esborra, crear i modificar el directori.
- es pot fer actiu i llistar el directori.
- es crea un fitxer hola.txt i posteriorment s'esborra.
- també s'esborra un altre dels fitxers tot i no ser propietat de guest.

15. Des del directori actiu operatius i com a usuari guest canviar el nom del directori apunts per anotacions (els permisos del directori apunts són 777).

```

[guest@a36 operatius]$ pwd
/tmp/m01/operatius

[guest@a36 operatius]$ ls -la
drwxrwxr-x 4 ecanet ecanet 80 Dec 18 18:21 .
drwxrwxr-x 4 ecanet ecanet 80 Dec 18 18:21 ..
drwxrwxrwx 2 ecanet ecanet 160 Dec 18 19:53 apunts
drwxrwxr-x 2 ecanet ecanet 40 Dec 18 18:21 exercicis

[guest@a36 operatius]$ mv apunts anotacions
mv: cannot move 'apunts' to 'anotacions': Permission denied

```

- tot i que el directori apunts té permisos oberts a tothom 777 l'usuari guest no li pot canviar el nom. Perquè? perquè canviar-li el nom NO és una modificació del directori apunts sinó del directori operatius (el que el conté). I guest NO té permisos de modificació (la w) del directori operatius.