

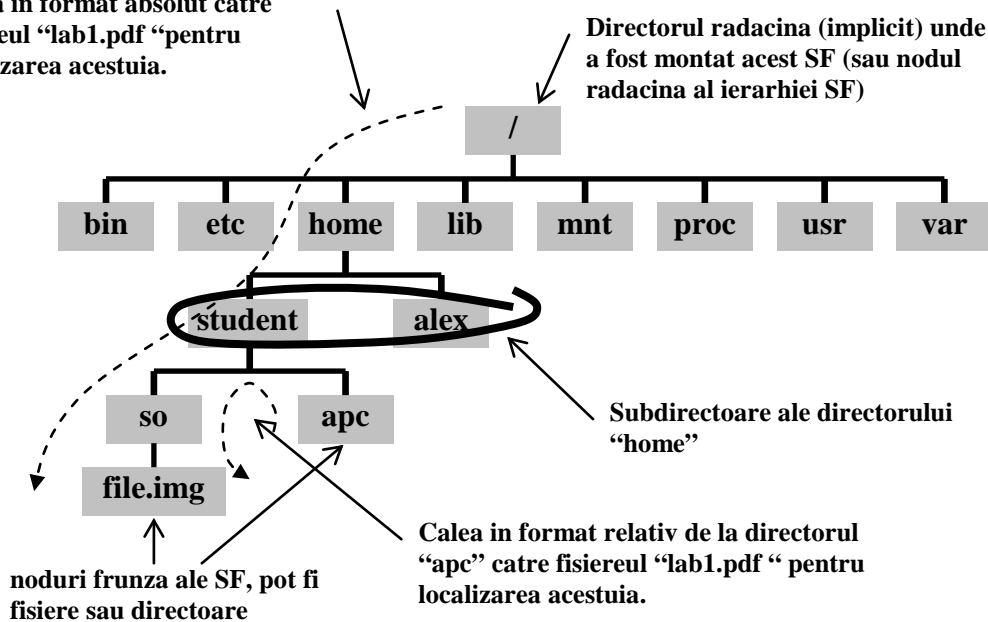
# Managementul sistemului de fisiere (SF) si al utilizatorilor

## Obiectivul lucrarii:

- studiul sistemului de fisiere *ext2*, (organizare, functionare) si managementul acestuia folosind comenzile interpretorului shell (bash);
- intelegerea mecanismelor utilizate pentru gestionarea conturilor utilizatorilor si realizarea de operatii pentru administrarea conturilor (creare, stergere de cont)

## Structura logica a sistemului de fisiere

Calea in format absolut catre fisiereul "lab1.pdf" pentru localizarea acestuia.



Localizarea unui fisier (ex. file.img):

- **prin cale absoluta** (ex. /home/student/so/file.img);
- **prin cale relativa** la un director (ex. ../so/file.img relativ la directorul "apc")

## Formatarea unui sistem de fisiere

Pentru realizarea in siguranta a acestui set de exercitii vom crea un sistem de fisiere cu eticheta "loop0" in interiorul unui fisier din sistemul de fisiere curent. Pentru a crea fisierele se va folosi comanda de copiere de nivel scazut dd, astfel:

```
#dd if=/dev/zero of=file.img bs=1k count=10000
```

Pentru a asigna acestui fisier o interfata tipica unui device de tip bloc se va configura:

```
#losetup /dev/loop0 file.img
```

Acum, noul device poate fi formatat:

```
#mke2fs -c /dev/loop0 10000
```

In urma operatiei de formatare suportul de stocare este structurat logic in 4 blocuri principale:

<b>MBR</b>	<b>Superblock</b>	<b>Inods</b>	<b>Data blocks</b>
------------	-------------------	--------------	--------------------

**Structura SF de tip ext2**

### Montarea unui sistem de fisiere

Sa se determine care sunt sistemele de fisiere montate pe sistemul de lucru.

```
#mount [-t <type> <eticheta_partitie> <directory> [-o <optiuni>]]
```

Sistemele de fisiere deja montate sunt anuntate in fisierul “/etc/mstab”. Vizualizati continutul acestui fisier.

In continuare montati SF creat de dvs. astfel:

```
#mount -t ext2 /dev/loop0 /mnt
```

dupa care accesati si validati noul SF cu comenzile: ls, df.

Vizualizati din nou continutul fisierului /etc/mstab.

Determinati o comanda pentru a demonta SF-ul, apoi:

### Verificarea SF

```
#e2fsck -p -C0 <eticheta_partitie>
```

### Operatii asupra directoarelor

In Linux directoarele sunt interpretate ca niste fisiere cu o structura stricta. Directoarele contin lista de nume de fisiere/subdirectoare si inodurile asociate, pe care le includ. O astfel de pereche (nume de fisier, inod) se numeste legatura (en. link). Atributele fisierelor sunt memorate in structuri inod care sunt asociate unic la un fisier.

**NOTA:** orice director contine fisierele “.” si “..”; acestea precizeaza inodul directorului curent, respectiv al directorului parinte.

Sa se listeze valorile inod pentru fisierele din directorul curent:

```
$ls -il [<calea_spre_director>]
```

Pentru a naviga in sistemul de fisiere :

```
$cd <calea_spre_director>
```

Listati continut unui director folosind comanda de mai jos:

```
$ls [<calea_spre_director>]
```

Comanda : ls [<nume\_director> | <nume\_fisier>] [optiuni]

### Crearea si stergerea unui director

```
$mkdir [<calea_spre_director>/]<nume_director>
```

Vizualizati atributele directorului nou creat. Cum au fost stabilite aceste atribute ?

### Stergerea unui director

```
$rm -rf <calea_spre_director>
```

sau,

```
$rmdir <calea_spre_director>
```

### Copierea unui director

```
$cp -r <calea_spre_directorul_sursa>  
          <calea_spre_directorul_destinatie>[/<nume_director>]
```

### Mutarea unui director

```
$mv -r <calea_spre_directorul_sursa>  
          <calea_spre_directorul_destinatie>[/<nume_director>]
```

## Operatii asupra fisierelor

### Crearea unui fisier (fara continut)

```
$touch [<calea_spre_director>/]<nume_fisier>
```

### Crearea unui link catre un fisier

Se va executa comanda urmatoare, mai intai fara argumentul -s pentru a crea un link hardware:

```
$ln [-s] <calea_spre_fisierul_tinta> [<calea_spre_director>/]<nume_link>
```

Apoi se va verifica ca atat fisierul link creat, cat si fisierul tinta au acelasi inod.

In continuare, se repeta aceasta comanda dar cu argumentul -s pentru a crea un link simbolic.

Se modifica atributele de acces la fisierul tinta prin crearea unui link ? Justificati raspunsul.

### Stergerea unui fisier sau a unui link

```
$rm <nume_fisier_sau_link>
```

Prin stergerea unui link este sters si fisierul tinta ?

## Copierea unui fisier

```
$cp <calea_spre_fisierul_sursa>  
    <calea_spre_directorul_destinatie>[/<nume_fisier>]
```

## Mutarea unui fisier

```
$mv <calea_spre_fisierul_sursa>  
    <calea_spre_directorul_destinatie>[/<nume_fisier>]
```

## Determinarea tipului unui fisier

```
$file [<calea_spre_director>]/<nume_fisier>
```

## Cautarea unui fisier sau a unui director

```
$find [<nume_director>] [<expresie>]
```

Unde expresie: -name <nume\_fisier\_sau\_director>

-user <username> sau -uid <numar>

-group <groupname> sau -gid <numar>

-type {d, f, s, l}

-size [{+,-}]<numar>

-inum <number>

-perm <attribute\_de\_acces>

-exec <command> {} +

-fprint <filename>

**Alte comenzi bash:** df, du, fdisk, grep, sort, cat, less, more, head, tail, sync, dd.

## Controlul accesului la un fisier/director

Pentru implementarea politicilor de securitate pentru controlul accesului utilizatorilor/proceselor la fisiere, majoritatea sistemelor de operare utilizeaza tehnica DAC<sup>1</sup>. Tehnica DAC presupune ca proprietarul unui fisier/director are control complet asupra drepturilor de acces la acesta (DAC nu utilizeaza conceptele de rol pentru subiecti si de niveluri de securitate).

Modelul utilizat pentru specificarea politicilor de securitate pentru controlul accesului la fisiere este modelul matrice. Conform acestui model, attributele de acces ale unor obiecte (fisier/director) corespunzatoare mai multor categorii de subiecti pot fi specificate astfel:

Subject \ Object	File_1	File_2	Program_file
Joe	read	read/write	execute
Gamers	read/write	read	none
Jane	read/write	read	execute

<sup>1</sup> en. Discretionary Access Control

### In cazul Linux:

Categorii de subiecti:

**user/owner (u), grup (g), others (o), all (a);**

Atribute de acces:

**read (r), write (w), execute (x), SUID/SGID (s), sticky bit (t);**

### Atributele de acces: SUID/SGID

De regula, atributele set-UID si set-GID sunt setate fisierelor program care trebuie sa realizeze operatii de sistem privilegiate, pentru a putea fi executate si de catre un utilizator normal (neprivilegiat). De exemplu, programul `passwd`.

In acest scop, sistemul kernel permite modificarea valorilor identificatorilor UID si GID ai unui utilizator (proces) normal pe durata executarii fisierelor cu atributele SUID/SGID setate. Pentru a suporta aceasta facilitate sistemul utilizeaza doua perechi de identificatori: (UID-real, GID-real) care sunt definiti in fisierul `/etc/passwd`, respectiv (UID-efectiv, GID-efectiv) care au valoarea egala cu UID-ul/GID-ul proprietarului fisierului program executat cu atributele SUID/SGID setate. Toti acesti identificatori sunt mosteniti de toate procesele fiu, create de programul respectiv.

### Atributul de acces: sticky bit

Acest atribut este asigant directoarelor. Atunci cand este setat, utilizatorii care pot sa creeze/stearga propriile fisiere in acest director, nu pot insa sa opereze cu fisierele celorlalti utilizatori.

### Vizualizarea atributelor de acces ale unui fisier/director

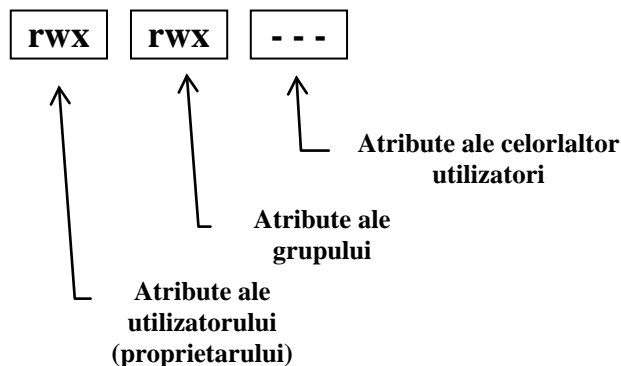
Se poate utiliza comanda "`ls -l`" al carei rezultat este afisat in formatul urmatoare:

```
<type><attribute> <links> <owner> <group> <size> <time> <name>
```

### Semnificatia campurilor:

Campul **<type>**: poate avea urmatoarele valori: directory (**d**), normal file (**-**), character device (**c**), block device (**b**), pipe FIFO (**=**), socket (**s**), link (**l**).

Structura campului **<attribute>**:



Campul **<links>**: reprezinta numarul de legaturi catre acest fisier/director; sau

Campul **<owner>**: numarul de nume alternative cu care este referit;  
precizeaza numele utilizatorului sau UID-ul real care este proprietar al fisierului/directorului;  
Campul **<group>**: precizeaza numele grupului sau GID-ul real care ar putea avea acces la fisier;  
Campul **<size>**: precizeaza dimensiunea acestui fisier/director;  
Campul **<time>**: precizeaza data ultimei modificari a fisierului/directorului;  
Campul **<name>**: numele fisierului/directorului

### Modificarea atributelor de acces:

Pentru a modifica attributele de acces ale unui fisier/director se va utiliza urmatoarea comanda cu argument simbolic:

```
$chmod [ugoa] [+-=] [rwxXstugo] <nume_fisier_sau_director>
```

sau cu argument numeric,

```
$chmod [[[[{0,1,2,4}]<0-7>]<0-7>]<0-7>] <nume_fisier_sau_director>
```

### Exercitiu

Sa se modifice attributele de acces la mai multe fisiere.

Sa se utilizeze comanda de mai jos pentru a face ca un fisier sa devina imuabil.

```
$chattr +i <nume_fisier_sau_director>
```

Incercati acum sa stergeti fisierul/directorul cu acest atribut.

### Modificarea atributului proprietar si/sau grup

Pentru a modifica attributele proprietar ale unui fisier/director se va utiliza comanda:

```
$chown [<owner>][.<group>] <nume_fisier_sau_director>
```

### Exercitiu

Sa se modifice proprietarul si grupul la mai multe fisiere.

### Restrictionarea atributelor implicite de acces

Atributele de acces care vor fi setate implicit unui fisier la crearea acestuia, pot fi mascate prin utilizarea urmatoarei comenzi:

```
$umask [[<0-7>]<0-7>]<0-7>]
```

Argumentul comenzii are o structura similara argumentului numeric al comenzii `chmod`. Valoarea argumentului va fi determinata astfel: pentru fiecare atribut care se va anula, bitul de rang corespunzator din argument va avea valoarea "1".

### Exercitiu

Sa se modifice attributele de acces implicite, apoi creati niste fisiere. Se va inspecta valorile acestor attribute.

## Managementul utilizatorilor

Informatiile despre utilizatori sunt memorate in fisierele “passwd” si “shadow”.  
Structura unei linii din fisierul **/etc/passwd**:

```
username:passwd:UID:GID:fullname:directory:shell
```

### Semnificatia campurilor:

**passwd:** contine parola criptata (ex. DES sau MD5) si codata in Base64;  
**UID:** identificatorul utilizatorului;  
**GID:** identificatorul de grup de care apartine implicit acest utilizator;  
**fullname:** este un comentariu;  
**directory:** directorul “casa” al acestui utilizator;  
**shell:** este calea catre programul shell asignat implicit utilizatorului la conectare (login)

### Exercitiu.

Incercati sa recunoasteti si sa interpretati valorile acestor campuri pe sistemul dvs.

Structura unei linii din fisierul **/etc/shadow**:

```
username:passwd:last:may:must:warn:expire:disable:reserved
```

### Semnificatia campurilor:

**passwd:** contine parola criptata (ex. DES sau MD5) si codata in Base64;  
**last:** data ultimei modificari a parolei (exprimata in zile de la 1.01.1970);  
**may:** numar de zile inainte ca parola sa poata fi schimbata;  
**must:** numar de zile dupa care parola trebuie schimbata;  
**warn:** numar de zile inainte care parola sa expire, cand utilizatorul va fi notificat;  
**expire:** numar de zile dupa expirarea parolei, cand contul ca vi dezactivat;  
**disable:** numar de zile pana la dezactivarea contului (exprimat in zile de la 1.01.1970);  
**reserved:** camp rezervat pentru utilizari ulterioare

### Exercitiu.

Incercati sa recunoasteti si sa interpretati valorile acestor campuri pe sistemul dvs.

## Adaugarea si stergerea unui cont utilizator

In acest scop sa se utilizeze comenzile `useradd/userdel` pentru a crea/sterge un cont utilizator. Dupa crearea unui cont utilizator sa se vizualizeze fisierul `/etc/passwd`.

## Modificarea parolei de conectare a unui utilizator

```
$passwd [<username>]
```

## Crearea/stergere si modificarea unui grup

In acest scop sa se utilizeze comenzile `groupadd/groupdel` pentru a crea/sterge uni grup. Dupa crearea unui grup sa se vizualizeze fisierul `/etc/group`. Componenta unui grup se poate modifica prin simpla editare a fisierului `/etc/group` sau, prin comanda: `gpasswd [-a][[-d]]`.

#### Anexa A:

##### Notatii:

Notatie	Semnificatie
[ arg ]	Argumentul <i>arg</i> este optional. Valoarea argumentului <i>arg</i> este data chiar de valoarea <i>arg</i> dintre paranteze
<arg>	Argumentul <i>arg</i> este obligatoriu. Valoarea argumentului se deduce din <i>arg</i> , care in acest caz este un text explicativ.
Arg1   arg2	Optiune intre argumentele <i>arg1</i> si <i>arg2</i> .
{arg1,arg2,arg3}	Argumentul este obligatoriu. Va avea una dintre valorile enumerate intre paranteze.

#### Anexa B:

Editoare de fisiere recomandate pentru acest laborator: **vi**, **nano**, **pico**, **mcedit**<sup>2</sup>.

---

<sup>2</sup> Este instalat odata cu managerul de fisiere: mc