

CAPITOLUL 3: Gestionarea Fișierelor și Directoarelor

Durată estimată studiu: 120 -150 minute

3.1 Manipulare fișiere

Comenzi esențiale pentru lucrul cu fișiere și directoare:

mkdir - Creare directoare

```
mkdir NumeDir                # Crează un singur director
mkdir -p Dir1/Dir2/Dir3      # Crează directoare înlănțuite
mkdir Dir{1..10}             # Crează Dir1, Dir2, ... Dir10
```

cp - Copiere fișiere/directoare

```
cp fisier.txt /destinatie/    # Copiază fișier
cp -r director/ /destinatie/  # Copiază director recursiv
```

mv - Mutare/Redenumire

```
mv fisier.txt /alta_locatie/  # Mută fișier
mv nume_vechi.txt nume_nou.txt # Redenumeste fișier
```

rm - Ștergere

```
rm fisier.txt                # Șterge fișier
rm -r director/              # Șterge director recursiv
rm -rf director/             # Șterge forțat fără confirmare
```

Exemple practice

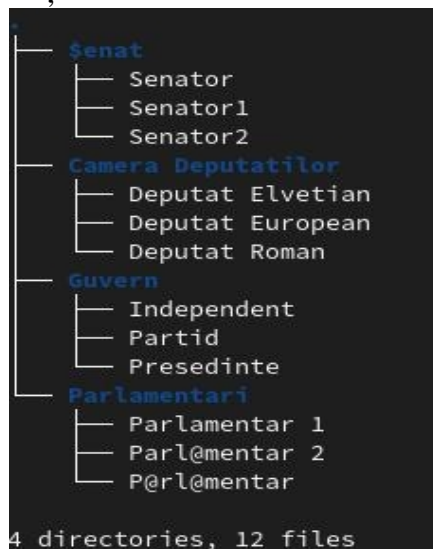
```
$ mkdir -p Proiect/Documente/PDF # Creează ierarhie de directoare
$ touch Proiect/raport.txt        # Creează fișier gol
$ cp raport.txt raport_backup.txt # Face copie
$ mv raport_backup.txt Documente/ # Mută în subdirector
```

Caracterele speciale în nume: Folosiți ghilimele sau backslash

```
$ touch "nume cu spatii.txt"
$ mv Pr3$3d!nt3 "Director Nou"
```

Cerințe:

1. Creați următorul arbore de fișiere într-un director numit "ExercitiuPATH" :



2. Mutați fișierul Senator în directorul Parlamentari.
3. Copiați fișierul Deputat Elvetian în directorul Senat.
4. Schimbați numele fișierului Independent în Asociat.
5. Creați un director numit Serial. În acest director creați folosind o singură comandă 10 directoare cu câte 5 directoare fiecare.
6. Copiați Partid în Camera Deputatilor.
7. Mutați Parlamentar 1 în Guvern.
8. Stergeți Deputat Roman.
9. Mutați Senator2 în Guvern
10. Stergeți Senator1.
11. Redenumiți Presedinte în Pr3\$3d!nt3.
12. Copiați directorul Serial în Senat.
13. Redenumiți Senat în Senat.
14. Stergeți directorul Serial.
15. Redenumiți Deputat European în D3put@t 3ur*opean.
16. Mutați Parl@mentar2 în Senat.
17. **Stergeți P@rl@mentar.**

3.2 Permisuni de bază, proprietari, grupuri

Structura permisiunilor: rwx rwx rwx (User, Group, Others)

r (read) = 4, w (write) = 2, x (execute) = 1

Comenzi de bază

```
ls -l                # Vizualizare permisiuni
chmod [permisiuni] fis # Modificare permisiuni
chown utilizator fis  # Schimbare proprietar
chgrp grup fis        # Schimbare grup
```

Metoda OCTALĂ (numerică)

```
chmod 755 script.sh    # rwxr-xr-x (7=rwx, 5=r-x, 5=r-x)
chmod 644 document.txt # rw-r--r-- (6=rw-, 4=r--, 4=r--)
chmod 000 secret.txt    # ----- (fără permisiuni)
```

Metoda SIMBOLICĂ

```
chmod u+x script.sh    # Adaugă execute pentru user
chmod g-w fisier.txt    # Elimină write pentru group
chmod o+r document.txt # Adaugă read pentru others
chmod a+x program       # Adaugă execute pentru all (a=u+g+o)
```

Exemple concrete

```
$ chmod 540 file.txt    # r-xr----- (user: r+x=5, group: r=4, others: 0)
$ chmod 507 file.txt    # r-x---rwx (user: 5, group: 0, others: 7)
$ chmod u=rx,g=,o=rwx   # Același rezultat cu simboluri
```

Bit-uri speciale

```
chmod 4--- file        # SUID (S la user)
chmod 2--- file        # SGID
chmod 1--- file        # Sticky bit
```

Cerințe:

1. Realizați următorul arbore de fișier:

```

Exercitiul5
-- dir1
|  -- a
|    |-- file1.txt
|    |-- file2.txt
|  -- b
|    |-- file3.txt
|  -- c
|    |-- file4.txt
-- dir2
|  -- a
|    |-- file5.txt
|    |-- file6.txt
|  -- b
|    |-- file7.txt
|    |-- file8.txt
|  -- c
|    |-- file10.txt
|    |-- file9.txt
  
```

2. Vizualizați permisiunile fișierului *file1.txt*.
3. Schimbați owner-ul fișierului *file1.txt* din cel obișnuit în *root*.
4. Schimbați grupul fișierului *file2.txt* din cel obișnuit în *root*.
5. Setați permisiuni doar de *read* pentru *file3.txt* pentru user, group și others.
6. Setați permisiuni cu valoarea octală *540* pentru *file4.txt*. Ce înseamnă această valoare?
7. Faceți ca permisiunile pentru *file5.txt* să arate astfel: *-r-x---rwx*, folosind comanda care conține valoarea octală.
8. Faceți ca permisiunile pentru *file6.txt* să arate astfel: *---S-----*, folosind valoarea octală.
9. Preveniți *file6.txt* să poată fi șters de către *others*.
10. Faceți ca permisiunile pentru *file7.txt* să arate astfel: *-rwxrwxrwx*, folosind simboluri.
11. Faceți ca permisiunile pentru *file8.txt* să arate astfel: *-----*, folosind simboluri.
12. Faceți ca permisiunile pentru *file9.txt* să arate astfel: *-----*, folosind valoarea octală.

3.3 Scripting

Structura de bază a unui script Bash

```
#!/bin/bash                                # Shebang (întotdeauna prima linie)
# Comentarii încep cu #
```

Variable

```
MESAJ="Hello World"                       # Atribuire (fără spații!)
echo $MESAJ                               # Afișare cu $
HOSTNAME=$(hostname)                      # Stochează output comandă
```

Input utilizator

```
read -p "Nume: " NUME                     # Citește de la tastatură
echo "Salut, $NUME"
```

Condiții (if)

```
if [ -f "fisier.txt" ]; then               # Testează dacă există fișier
    echo "Fișier găsit"
elif [ -d "director" ]; then              # Testează dacă există director
    echo "Director găsit"
else
    echo "Nimic găsit"
fi
```

Teste utile

```
-f fisier      # True dacă fișier există
-d director    # True dacă director există
-z "$VAR"      # True dacă variabilă goală
$? -eq 0       # True dacă comanda anterioară a reușit
```

Exemple practice

Script simplu

```
echo "Shell scripting este distractiv"
```

Cu variabilă

```
MSG="Shell scripting este distractiv"
echo $MSG
```

Captură hostname

```
HOST=$(hostname)
echo "Acest script rulează pe $HOST"
```

Verificare fișier cu locate

```
read -p "Calea fișierului: " CALE
if locate "$CALE" > /dev/null 2>&1; then
    echo "Fișierul există în baza de date"
else
    echo "Eroare: Fișierul nu a fost găsit"
fi
```

Identificare IP

```
IP=$(hostname -I | awk '{print $1}')
echo "Adresa IP: $IP"
```

Cerinte:

1. Scrie-ti un script care afiseaze "Shell scripting este distractiv";
2. Modificați scriptul de mai sus ca să includă o variabila. Variabila va contine mesajul "Shell scripting este distractiv", afișați variabila.
3. Creați un script care sa stocheze iesirea comenzii hostname intr-o variabila. Afișați mesajul "Acest script ruleaza pe [x]", unde [x] este outputul comenzii hostname.
4. Realizați un script care sa citeasca date de la tastatura, afisându-le ulterior.
5. Creați un script care să foloseasca comanda locate, daca fișierul exista sa afișeze un mesaj, iar în caz contrar sa afișeze un mesaj de eroare.
6. Scrieți un script care sa afiseze pe ecran urmatoarele : Linux, sheel, bash, script, terminal. Incercați sa faceți asta în cel mai scurt mod .
7. Scrieți un script care sa identifice daca calea oferita de la tastatura este fisier sau director, afisand acest rezultat.
8. Scrieți un script care să identifice adresa IP a stației de lucru.
9. Scrieti un script care sa verifice daca sunt updateuri de facut. Daca sunt de facut sa afiseze un mesaj de genul "updateuri disponibile", daca nu sunt updateuri de facut sa afiseze mesajul "Nu sunt updateuri disponibile" și în cazul unei erori sa afiseze mesajul "Eroare va rog verificati conexiunea la internet".