

# LINUX SERVER ADMINISTRATION

## DOCUMENTATIE CURS

DOCUMENTATIE

INTREABA PROFESORUL

CURSURILE MELE

### 1. Shell Scripts

#### 1.1 Introducere Bash

#### 1.2 Variabile, constante si functii

#### 1.3 Parametri pozitionali

#### 1.4 Alte facilitati Bash

#### 1.5 Flow Control

#### 1.6 Substituirea comenzilor

### 2. Linux Kernel

### 3. Serverul DHCP

### 4. Serverul FTP

### 5. NFS - Network File System

### 6. Serverul DNS

### 7. Serverul Apache

### 8. Serverul MySql

### 9. NETFILTER

### 10. Sistemul de e-Mail

### 11. Serverul Postfix

### 12. Serverul POP/IMAP

### 13. Managementul Logurilor

### 14. Exemple practice (Ubuntu 14.04 LTS)

### 15. Webmin

1 Shell Scripts » 1.1 Introducere Bash

## Introducere Bash

Shellul este interpretorul de comenzi. Acesta parseaza comenzile introduse de utilizator, le verifica din punct de vedere al sintaxei si le trimite kernelului spre executie.

Exista mai multe shelluri (Bourne, Korn, C Shell etc), dar default pe Linux este shellul numit **Bash** (Bourne Again Shell).

Fizic Bash-ul este reprezentat de fisierul `/bin/bash`

Pe langa interpretarea si executia comenzilor, Bash-ul ofera si alte facilitati printre care: editarea liniei de comanda, history, alias-uri, dar si un limbaj de programare structurat propriu.

Un script de shell este un fisier text (ASCII) care contine comenzi de shell organizate intr-un mod logic. Pe langa comenzi, script-ul mai contine si variabile (locale sau globale), parametri pozitionali precum si structuri de programare specifice (if..then..else, while, do, until, for, case, select etc).

Un script este folosit pentru a imbrina comenzi si diverse structuri de control pentru a oferi o noua functionalitate sau a automatiza task-uri.

Etapele crearii unui script sunt:

- a) crearea unui fisier text folosind editorul preferat ( **Exemplu:** vim)
- b) editarea fisierului pentru a contine functionalitatea dorita
- c) testarea scriptului adica executia/rularea acestuia

Moduri de executie ale unui script numit `script1`:

1. Din directorul care contine scriptul se ruleaza : `./script1`

Fisierul `script1` trebuie sa aiba dreptul de executie. Fiindca directorul curent (.) nu se afla in \$PATH (altfel grave probleme de securitate), scriptul trebuie sa fie executat cu calea absoluta sau cu `./nume_script`

In acest caz se porneste un shell nou ca sub proces al shellului curent in care se executa comenzile din script.

2. Din directorul care contine fisierul `script1` se ruleaza: `. script1` sau `source script1`.

Se executa comenzile din `script1` in shell-ul curent. Fisierul `script1` nu trebuie sa aiba dreptul de executie.

Una dintre diferentele dintre prima si a doua varianta este ca variabilele exportate sau globale sunt vizibile in shell-ul curent in varianta a doua, dar nu si in prima varianta.

### Nota



Varianta cea mai folosita de executie a unui script este prima: `./script`

Un script trebuie sa contina pe prima linie o referire la programul cu care se executa (Bash, Perl, PHP, Python etc). Astfel prima linie a unui script Bash este invariabil: `#!/bin/bash`

In interiorul scriptului `#` (diez) reprezinta un comentariu. Exceptie face prima linie.

`echo` se foloseste pentru a afisa un string (sir de caractere) sau o variabila la consola. **Exemplu:** `echo "Linux"`

### Exemplu

1. Script care afiseaza mesajul: Acesta este primul exemplu !

```
#!/bin/bash
echo "Acesta este primul exemplu !"
```



2. Script care creaza un fisier numit `continut_etc.txt` in directorul curent ce contine fisierele si directoarele din directorul `/etc`, iar apoi afiseaza spatiul total, ocupat si liber de pe fiecare partiie de pe Hard Disk.

```
#!/bin/bash
ls -l /etc > continut_etc.txt
echo "Fisierul continut_etc.txt a fost creat in directorul curent!"
echo "Informatii partitii:"
df -h
```

### Important

O cunoastere acceptabila a modului de creare de scripturi este esentiala pentru orice Administrator de sistem. In cazul persoanelor incepatoare care nu au experienta in programare (orice limbaj) procesul de scripting pare la prima vedere extrem de complicat. Este nevoie de multe exercitii si exemple pentru o intelegere aprofundata. Intotdeauna se incepe cu exercitii simple si anume scripturi formate din cateva randuri.

### Resurse:

- [Bash - wikipedia](#)



Durata: 1.30 min

Marime: 2.1 MB

Nota: Se recomanda folosirea [VLC](#) pentru vizualizare tutorial video.