Version : **2024.01**

Dernière mise-à-jour : 2024/04/05 07:59

Topic 3 : GNU and Unix Commands

Contenu du Module

- **Topic 3 : GNU and Unix Commands**
  - Contenu du Module
  - Présentation de VI
  - LAB #1 - Créer, ouvrir et fermer des fichiers
    - 1.1 - Commandes
    - 1.2 - Créer un nouveau fichier avec VI
    - 1.3 - Ouvrir un fichier en mode lecture seule avec la commande view
    - 1.4 - Ouvrir un fichier en mode lecture-écriture avec la commande vi
  - LAB #2 - La Commande set
    - 2.1 - Commandes
    - 2.2 - Activer la numérotation des lignes avec la commande set
  - LAB #3 - Se Déplacer dans un Fichier
    - 3.1 - Commandes
  - LAB #4 - Insertion de Texte
    - 4.1 - Commandes
    - 4.2 - Insérer du texte
  - LAB #5 - Recherche de Texte
    - 5.1 - Commandes
    - 5.2 - Rechercher et remplacer du texte
  - LAB #6 - Suppression de Texte
    - 6.1 - Commandes
    - 6.2 - Supprimer des lignes
  - LAB #7 -Copier, Couper et Coller

    - 7.1 - Commandes
    - 7.2 - Copier, couper et coller du texte
  - LAB #8 -Configuration d'une Interface VI Personnelle
  - LAB #9 - 38 Commandes de Base
    - 9.1 - La commande stty
    - 9.2 - La commande date
    - 9.3 - La commande who
    - 9.4 - La commande df
    - 9.5 - La commande free
    - 9.6 - La commande whoami
    - 9.7 - La commande pwd
    - 9.8 - La commande cd
    - 9.9 - La commande ls
    - 9.10 - La commande lsof
    - 9.11 - La commande touch
    - 9.12 - La commande echo
    - 9.13 - La commande cp
    - 9.14 - La commande file
    - 9.15 - La commande cat
    - 9.16 - La commande mv
    - 9.17 - La commande mkdir
    - 9.18 - La commande rmdir
    - 9.19 - La commande rm
    - 9.20 - La commande sort
    - 9.21 - La commande more
    - 9.22 - La commande less
    - 9.23 - La commande find
    - 9.24 - La commande su
    - 9.25 - Les commandes locate et updatedb
    - 9.26 - La commande whereis
    - 9.27 - La commande which
    - 9.28 - La commande uptime
    - 9.29 - La commande w
    - 9.30 - La commande uname

        - 9.31 - La commande du
        - 9.32 - La commande clear
        - 9.33 - La commande exit
        - 9.34 - La commande logout
        - 9.35 - La commande sleep
        - 9.36 - La Commande wall
        - 9.37 - The seq Command
        - 9.38 - La Commande screen
    - LAB #10 - Options et Arguments
    - LAB #11 - Expressions Régulières
        - ERb
        - ERe
    - Outils de Manipulation de Fichiers Texte
        - Présentation des Commandes grep, egrep et fgrep
            - La commande grep
            - La Commande egrep
            - La Commande fgrep
            - LAB #12 - Utiliser grep, egrep et fgrep
        - Présentation de la Commande sed
            - LAB #13 - Utiliser la Commande sed
        - Présentation de La Commande awk
            - Découpage en champs
            - Critères
                - Une expression régulière valide pour la ligne
                - Une expression régulière valide pour un champ
                - Une comparaison
                - Un opérateur logique
                - Une variable interne
            - Scripts awk
            - La Fonction printf
            - Structures de Contrôle
                - if
                - for
                - while

- do-while
- LAB #14 - Utiliser la Commande awk
- LAB #15 -Autres Commandes Utiles
  - 15.1 - La Commande expand
  - 15.2 - La Commande unexpand
  - 15.3 - La Commande cut
  - 15.4 - La Commande uniq
  - 15.5 - La Commande tr
  - 15.6 - La Commande paste
  - 15.7 - La Commande split
  - 15.8 - La Commande diff
  - 15.9 - La Commande cmp
  - 15.10 - La commande patch
  - 15.11 - La commande strings
  - 15.12 - La commande comm
  - 15.13 - La commande head
  - 15.14 - La commande tail
- LAB #16 - Utiliser les commandes ifconfig, grep, tr et cut pour isoler l'adresse IPv4
- LAB #17 - Utiliser les commandes ip, grep, awk et sed pour isoler l'adresse IPv4
- Le Shell
- Le Shell /bin/bash
  - Les Commandes Internes et Externes au shell
  - Les alias
  - Le Prompt
  - Rappeler des Commandes
  - Générer les fins de noms de fichiers
  - Le shell interactif
    - Caractère *
    - Caractère ?
    - Caractères [ ]
  - L'option extglob
    - ?(expression)
    - *(expression)
    - +(expression)

- @(expression)
- !(expression)
- Caractères d'Échappement
- Codes Retour
- Redirections
- Pipes
- Substitutions de Commandes
- Chaînage de Commandes
- Affichage des variables du shell
  - Les variables principales
  - Les Variables de Régionalisation et d'Internationalisation
  - Les variables spéciales
- La Commande env
- Options du Shell Bash
  - Exemples
    - noclobber
    - noglob
    - nounset
- Les Scripts Shell
  - Exécution
  - La commande read
    - Code de retour
    - La variable IFS
  - La commande test
    - Tests de Fichiers
    - Tests de chaînes de caractère
    - Tests sur des nombres
    - Les opérateurs
    - Tests d'environnement utilisateur
  - La commande [[expression]]
  - Opérateurs du shell
  - L'arithmétique
    - La commande expr
      - Opérateurs Arithmétiques

            - Opérateurs de Comparaison
            - Opérateurs Logiques
        - La commande let
            - Opérateurs Arithmétiques
            - Opérateurs de comparaison
            - Opérateurs Logiques
            - Opérateurs travaillant sur les bits
        - Structures de contrôle
            - If
            - case
                - Exemple
        - Boucles
            - for
            - while
            - Exemple
    - Scripts de Démarrage
        - LAB #1- Scripts de Démarrage
            - ~/.bash_profile
            - ~/.bashrc
- Les Types de Processus
- Les Commandes relatives aux Processus
    - La Commande ps
    - La Commande pgrep
    - La Commande pstree
    - La Commande top
    - Les Commandes fg, bg et jobs
    - La Commande wait
    - La Commande nice
    - La Commande renice
    - La Commande nohup
    - La Commande kill
    - La Commande pkill

Présentation de VI

VI est un éditeur de texte puissant qui dispose de trois modes :

- *Commande*
- *Ex-mode*
- *Insertion*

En mode commande, il est possible de saisir des commandes telles **i** pour insérer du texte.

En mode Ex-Mode, la plupart des commandes VI sont précédées par la touche :, par exemple **:q** pour quitter.

En mode insertion, il est possible :

- d'éditer le texte,
- ajouter du texte,
- rechercher du texte,
- copier du texte,
- coller du texte,
- couper du texte,
- remplacer du texte.

LAB #1 - Créer, ouvrir et fermer des fichiers

1.1 - Commandes

Commande	**Description**
vi *nomFichier*	Ouverture ou création d'un fichier
vi	Ouverture d'un nouveau fichier
vi -r *nomFichier*	Récupération d'un fichier après une panne
view *nomFichier*	Ouverture d'un fichier en mode lecture seule
:wq	Quitter en sauvegardant et en modifiant la date de modification

Commande	Description
ZZ	La même chose que :wq
:x	Quitter en sauvegardant sans modifier la date de modification en cas de non-modification du fichier
:q	Quitter en cas de non-modification du fichier ou suite à l'utilisation de la commande **view**
u	Annulation de la dernière commande
U	Annulation des modifications apportées à la ligne en cours
:q!	Quitter sans sauvegarder

1.2 - Créer un nouveau fichier avec VI

Copiez les 25 lignes ci-dessous :

```
This is line 1
This is line 2
This is line 3
This is line 4
This is line 5
This is line 6
This is line 7
This is line 8
This is line 9
This is line 10
This is line 11
This is line 12
This is line 13
This is line 14
This is line 15
This is line 16
This is line 17
This is line 18
This is line 19
This is line 20
This is line 21
```

```
This is line 22
This is line 23
This is line 24
This is line 25
```

Créez un nouveau fichier appelé **vitext** en utilisant la commande vi:

```
[trainee@centos7 ~]$ vi vitext
```

A faire - Appuyez sur la touche **i** de votre clavier pour basculer en mode *insertion*. Cliquez le bouton central (la roue) de votre souris afin de coller dans le fichier les lignes 1 à 25. Utilisez ensuite la touche Echap pour basculer en mode *Commande*. Appuyez sur la touche : suivi par la touche X pour sauvegarder et quitter VI.

1.3 - Ouvrir un fichier en mode lecture seule avec la commande view

Ouvrez maintenant le fichier **/home/trainee/vitext** en lecture seule :

```
[trainee@centos7 ~]$ view vitext
```

Vous obtiendrez un résultat similaire à celui-ci :

```
This is line 1
This is line 2
This is line 3
This is line 4
This is line 5
This is line 6
This is line 7
This is line 8
```

```
This is line 9
This is line 10
This is line 11
This is line 12
This is line 13
This is line 14
This is line 15
This is line 16
This is line 17
This is line 18
This is line 19
This is line 20
This is line 21
This is line 22
This is line 23
This is line 24
This is line 25
~
"vitext" [readonly] 25L, 391C
```

Attention : Assurez-vous que vous ne voyez que les 25 premières lignes de ce fichier.

Important : Notez la présence de la dernière ligne avec la mention [readonly].

1.4 - Ouvrir un fichier en mode lecture-écriture avec la commande vi

Quittez view avec la commande **:q** et ouvrez le fichier **/home/trainee/vitext** en mode lecture-écriture :

```
[trainee@centos7 ~]$ vi vitext
```

Vous obtiendrez un résultat similaire à celui-ci :

```
This is line 1
This is line 2
This is line 3
This is line 4
This is line 5
This is line 6
This is line 7
This is line 8
This is line 9
This is line 10
This is line 11
This is line 12
This is line 13
This is line 14
This is line 15
This is line 16
This is line 17
This is line 18
This is line 19
This is line 20
This is line 21
This is line 22
This is line 23
This is line 24
This is line 25
"vitext" 25L, 391C                                              25,1          All
```

Important : Notez que vi est lancé en mode Commande.

LAB #2 - La Commande set

2.1 - Commandes

Commande	Description
:set nu	Visualisation de la numérotation des lignes
:set number	Visualisation de la numérotation des lignes
:set nonu	Suppression de la numérotation des lignes
:set nonumber	Suppression de la numérotation des lignes
:set ic	Rechercher sans tenir compte de la casse
:set noic	Rechercher en tenant compte de la casse

2.2 - Activer la numérotation des lignes avec la commande set

Activer la numérotation des lignes avec la commande **:set nu** ou la commande **:set number**. Vous obtiendrez un résultat similaire à celui-ci :

```
  1 This is line 1
  2 This is line 2
  3 This is line 3
  4 This is line 4
  5 This is line 5
  6 This is line 6
  7 This is line 7
  8 This is line 8
  9 This is line 9
 10 This is line 10
 11 This is line 11
 12 This is line 12
 13 This is line 13
 14 This is line 14
 15 This is line 15
 16 This is line 16
```

```
 17 This is line 17
 18 This is line 18
 19 This is line 19
 20 This is line 20
 21 This is line 21
 22 This is line 22
 23 This is line 23
 24 This is line 24
 25 This is line 25
:set nu                                                    25,1          All
```

```
  1 This is line 1
  2 This is line 2
  3 This is line 3
  4 This is line 4
  5 This is line 5
  6 This is line 6
  7 This is line 7
  8 This is line 8
  9 This is line 9
 10 This is line 10
 11 This is line 11
 12 This is line 12
 13 This is line 13
 14 This is line 14
 15 This is line 15
 16 This is line 16
 17 This is line 17
 18 This is line 18
 19 This is line 19
 20 This is line 20
 21 This is line 21
 22 This is line 22
 23 This is line 23
```

```
 24 This is line 24
 25 This is line 25
:set number                                                    25,1          All
```

LAB #3 - Se Déplacer dans un Fichier

3.1 - Commandes

Commande	Description
h ou ← ou Backspace	Déplacement du curseur un caractère vers la gauche
j ou ↓ ou ↵ Entrée	Déplacement du curseur une ligne vers le bas
k ou ↑	Déplacement du curseur une ligne vers le haut
l ou → ou Barre d'espacement	Déplacement du curseur un caractère vers la droite
b	Déplacement du curseur un mot vers la gauche
w	Déplacement du curseur un mot vers la droite
e	Déplacement du curseur vers la fin du mot en cours
H	Déplacement du curseur en haut de l'écran
M	Déplacement du curseur au milieu de l'écran
L	Déplacement du curseur en bas de l'écran
G ou **:$**	Déplacement du curseur vers la dernière ligne du fichier
1G ou **:0**	Déplacement du curseur vers la première ligne du fichier
27G	Déplacement du curseur vers la ligne 27
Ctrl+f	Défilement d'une page-écran vers l'avant
Ctrl+d	Défilement d'une demi-écran vers l'avant
Ctrl+b	Défilement d'une page-écran vers l'arrière
Ctrl+u	Défilement d'une demi-écran vers l'arrière

A faire : Testez chaque commande afin de visualiser les résultats. Revenez ensuite à la première page-écran et positionnez votre curseur au début de la ligne 13.

Important : Pour savoir pourquoi les touches H, J, K et L sont utilisés comme un pavé directionnel, consultez cette page.

LAB #4 - Insertion de Texte

4.1 - Commandes

Touche(s)	Description
i	Insertion du texte avant le curseur
I	Insertion du texte en début de ligne
a	Insertion du texte après le curseur
A	Insertion du texte en fin de ligne
o	Insertion d'une ligne après la ligne courante
O	Insertion d'une ligne avant la ligne courante
R	Remplacement du texte existant
Echap	Passage du mode *Insertion* en mode *Commande*

4.2 - Inserer du texte

Insérez une ligne en dessous de la ligne courante en utilisant le commande **o**. Notez que vous êtes maintenant en mode *Insertion*. Tapez ensuite **Linux is super**. Vous obtiendrez un résultat similaire à celui-ci :

```
1 This is line 1
2 This is line 2
3 This is line 3
4 This is line 4
5 This is line 5
6 This is line 6
7 This is line 7
```

```
  8 This is line 8
  9 This is line 9
 10 This is line 10
 11 This is line 11
 12 This is line 12
 13 This is line 13
 14 Linux is super
 15 This is line 14
 16 This is line 15
 17 This is line 16
 18 This is line 17
 19 This is line 18
 20 This is line 19
 21 This is line 20
 22 This is line 21
 23 This is line 22
 24 This is line 23
 25 This is line 24
-- INSERT --                                                14,15         Top
```

Attention - Ne changez pas la taille de votre terminal. Vous ne devez visualiser QUE les premières 25 lignes.

Passez maintenant en mode *Commande* en appuyant sur la touche Echap puis placez-vous sur la dernière ligne de l'écran en utilisant le commande **L**. Positionnez-vous à la fin de la ligne en mode *Insertion* en utilisant la commande **A** et saisissez de nouveau la phrase **Linux is super**. Vous obtiendrez un résultat similaire à celui-ci :

```
  1 This is line 1
  2 This is line 2
  3 This is line 3
  4 This is line 4
  5 This is line 5
```

```
  6 This is line 6
  7 This is line 7
  8 This is line 8
  9 This is line 9
 10 This is line 10
 11 This is line 11
 12 This is line 12
 13 This is line 13
 14 Linux is super
 15 This is line 14
 16 This is line 15
 17 This is line 16
 18 This is line 17
 19 This is line 18
 20 This is line 19
 21 This is line 20
 22 This is line 21
 23 This is line 22
 24 This is line 23
 25 This is line 24Linux is super
-- INSERT --                                                  25,30         Top
```

Passez maintenant en mode *Commande* en appuyant sur la touche Echap puis placez-vous au début de la première ligne de l'écran en utilisant le commande **H**. Positionnez-vous au quatrième mot en utilisant la commande **w** trois fois. Passez en mode *Insertion* en utilisant la commande **i** puis saisissez de nouveau la phrase **Linux is super**. Vous obtiendrez un résultat similaire à celui-ci :

```
  1 This is line Linux is super1
  2 This is line 2
  3 This is line 3
  4 This is line 4
  5 This is line 5
  6 This is line 6
  7 This is line 7
  8 This is line 8
```

```
  9 This is line 9
 10 This is line 10
 11 This is line 11
 12 This is line 12
 13 This is line 13
 14 Linux is super
 15 This is line 14
 16 This is line 15
 17 This is line 16
 18 This is line 17
 19 This is line 18
 20 This is line 19
 21 This is line 20
 22 This is line 21
 23 This is line 22
 24 This is line 23
 25 This is line 24Linux is super
-- INSERT --                                                    1,28          Top
```

Passez maintenant en mode *Commande* en appuyant sur la touche Echap puis placez-vous au début de la première ligne de l'écran en utilisant le commande **H**.

LAB #5 - Recherche de Texte

5.1 - Commandes

Touche(s)	Description
/ chaîne	Recherche *chaîne* vers le bas
//	Recherche l'occurence suivante vers le bas de la dernière recherche
? *chaîne*	Recherche *chaîne* vers le haut
??	Recherche l'occurence suivante vers le haut de la dernière recherche
n	Recherche l'occurence suivante de *chaîne* dans le sens de la recherche

Touche(s)	Description
N	Recherche l'occurence précédente de *chaîne* dans le sens de la recherche
:g/*chaîne*/**s**//*chaîne1*/**g**	Recherche et remplace *chaîne* par *chaîne1*

5.2 - Rechercher et remplacer du texte

Recherchez maintenant la chaîne *super* grâce à la commande **/super** suivez par la touche ↵ Entrée. Recherchez ensuite les deux occurrences suivantes en utilisant la commande **//** suivez par la touche ↵ Entrée puis **//** suivez par la touche ↵ Entrée. A ce stade votre curseur doit se trouver sur le dernier mot de la dernière ligne de votre écran.

Recherchez ensuite la chaîne *super* vers le haut en utilisant la commande **?super** suivez par la touche ↵ Entrée. Votre curseur doit se trouver sur la ligne 14.

Appuyez maintenant sur la touche **n**. Votre curseur doit se trouver sur la première ligne. Appuyez ensuite sur la touche **N**. Votre curseur doit se trouver sur la ligne 14.

Appuyez sur la commande **H**. Remplacez maintenant le mot *super* par *wonderful* en utilisant la commande **:g/super/s//wonderful/g** suivez par la touche ↵ Entrée. VI vous confirme le remplacement :

```
  1 This is line Linux is wonderful1
  2 This is line 2
  3 This is line 3
  4 This is line 4
  5 This is line 5
  6 This is line 6
  7 This is line 7
  8 This is line 8
  9 This is line 9
 10 This is line 10
 11 This is line 11
 12 This is line 12
 13 This is line 13
 14 Linux is wonderful
 15 This is line 14
```

```
 16 This is line 15
 17 This is line 16
 18 This is line 17
 19 This is line 18
 20 This is line 19
 21 This is line 20
 22 This is line 21
 23 This is line 22
 24 This is line 23
 25 This is line 24Linux is wonderful
3 substitutions on 3 lines                                    25,1          Top
```

LAB #6 - Suppression de Texte

6.1 - Commandes

Touche(s)	Description
x	Suppression du caractère courant
X	Suppression du caractère à gauche du curseur
5x	Suppression de 5 caractères à partir du caractère courant
dw	Suppression du mot courant
5dw	Suppression de 5 mots à partir du caractère courant
dd ou **:d**	Suppression de la ligne courante
5dd	Suppression de 5 lignes à partir de la ligne courante
:5,7 d	Suppression des lignes 5, 6 et 7

6.2 - Supprimer des lignes

Placez-vous sur la ligne 14 et supprimez-la grâce à la commande **dd**. Vous obtiendrez donc un résultat similaire à celui-ci :

```
  1 This is line Linux is wonderful1
```

```
  2 This is line 2
  3 This is line 3
  4 This is line 4
  5 This is line 5
  6 This is line 6
  7 This is line 7
  8 This is line 8
  9 This is line 9
 10 This is line 10
 11 This is line 11
 12 This is line 12
 13 This is line 13
 14 This is line 14
 15 This is line 15
 16 This is line 16
 17 This is line 17
 18 This is line 18
 19 This is line 19
 20 This is line 20
 21 This is line 21
 22 This is line 22
 23 This is line 23
 24 This is line 24Linux is wonderful
 25 This is line 25
                                                                14,1          All
```

Supprimez maintenant les lignes 4, 5 et 6 grâce à la commande **:4,6 d**. Vous obtiendrez un résultat similaire à celui-ci :

```
  1 This is line Linux is wonderful1
  2 This is line 2
  3 This is line 3
  4 This is line 7
  5 This is line 8
  6 This is line 9
```

```
  7 This is line 10
  8 This is line 11
  9 This is line 12
 10 This is line 13
 11 This is line 14
 12 This is line 15
 13 This is line 16
 14 This is line 17
 15 This is line 18
 16 This is line 19
 17 This is line 20
 18 This is line 21
 19 This is line 22
 20 This is line 23
 21 This is line 24Linux is wonderful
 22 This is line 25
~
~
~
3 fewer lines                                                    4,1           All
```

LAB #7 -Copier, Couper et Coller

7.1 - Commandes

Touche(s)	Description
yy ou **Y**	Copie la ligne courante
V	Sélectionner un bloc
p	Colle la ligne couper ou copier en dessous de la ligne courante
P	Colle la ligne couper ou copier en dessus de la ligne courante
:2,3 co 7	Copie des lignes 2 à 3 à la ligne située en dessous de la ligne 7
:2,3 m 7	Déplace les lignes 2 à 3 à la ligne en dessous de la ligne 7

7.2 - Copier, couper et coller du texte

Placez votre curseur sur la ligne 3. Copiez celle-ci grâce à la commande **yy**. Placez votre curseur sur la ligne 5 et utilisez la commande **p**. Vous obtiendrez le résultat suivant :

```
  1 This is line Linux is wonderful1
  2 This is line 2
  3 This is line 3
  4 This is line 7
  5 This is line 8
  6 This is line 3
  7 This is line 9
  8 This is line 10
  9 This is line 11
 10 This is line 12
 11 This is line 13
 12 This is line 14
 13 This is line 15
 14 This is line 16
 15 This is line 17
 16 This is line 18
 17 This is line 19
 18 This is line 20
 19 This is line 21
 20 This is line 22
 21 This is line 23
 22 This is line 24Linux is wonderful
 23 This is line 25
~
~
3 fewer lines                                                    6,1           All
```

Placez votre curseur sur la ligne 4 et utilisez la commande **Y**. Placez votre curseur sur la ligne 6 et utilisez la commande **P**. Vous obtiendrez le résultat

suivant :

```
  1 This is line Linux is wonderful1
  2 This is line 2
  3 This is line 3
  4 This is line 7
  5 This is line 8
  6 This is line 7
  7 This is line 3
  8 This is line 9
  9 This is line 10
 10 This is line 11
 11 This is line 12
 12 This is line 13
 13 This is line 14
 14 This is line 15
 15 This is line 16
 16 This is line 17
 17 This is line 18
 18 This is line 19
 19 This is line 20
 20 This is line 21
 21 This is line 22
 22 This is line 23
 23 This is line 24Linux is wonderful
 24 This is line 25
~
3 fewer lines                                                    6,1           All
```

Utilisez la commande **:4,5 co 15**. Vous obtiendrez le résultat suivant :

```
  1 This is line Linux is wonderful1
  2 This is line 2
  3 This is line 3
```

```
  4 This is line 7
  5 This is line 8
  6 This is line 7
  7 This is line 3
  8 This is line 9
  9 This is line 10
 10 This is line 11
 11 This is line 12
 12 This is line 13
 13 This is line 14
 14 This is line 15
 15 This is line 16
 16 This is line 7
 17 This is line 8
 18 This is line 17
 19 This is line 18
 20 This is line 19
 21 This is line 20
 22 This is line 21
 23 This is line 22
 24 This is line 23
 25 This is line 24Linux is wonderful
:4,5 co 15                                                  17,1          Top
```

Notez que les lignes 4 et 5 ont été copiées après la ligne 15 :

```
...
 15 This is line 16
 16 This is line 7
 17 This is line 8
 18 This is line 17
...
```

Utilisez la commande **:4,6 m 20**. Vous obtiendrez le résultat suivant :

```
  1 This is line Linux is wonderful1
  2 This is line 2
  3 This is line 3
  4 This is line 3
  5 This is line 9
  6 This is line 10
  7 This is line 11
  8 This is line 12
  9 This is line 13
 10 This is line 14
 11 This is line 15
 12 This is line 16
 13 This is line 7
 14 This is line 8
 15 This is line 17
 16 This is line 18
 17 This is line 19
 18 This is line 7
 19 This is line 8
 20 This is line 7
 21 This is line 20
 22 This is line 21
 23 This is line 22
 24 This is line 23
 25 This is line 24Linux is wonderful
3 lines moved                                                    20,1          Top
```

Notez qu'en supprimant les lignes 4 à 6, la ligne 20 est devenue la ligne 17. Les trois lignes coupées apparaissent donc après la ligne 17 et non après la ligne 20 :

```
...
 17 This is line 19
 18 This is line 7
 19 This is line 8
```

```
 20 This is line 7
...
```

LAB #8 -Configuration d'une Interface VI Personnelle

VI peut être configuré par chaque utilisateur en créant un fichier dénommé **~/.exrc**. Ce fichier doit contenir les commandes des options comme celles-ci sont tapées sur la ligne de commande mais sans le caractère **:**. Le fichier **~/.exrc** est lu par VI lors de son lancement et les options mises en place. Par exemple le fichier suivant activera les options de la *numérotation des lignes* et la *visibilité des caractères cachés* :

```
set nu
set list
```

A faire - Sauvegardez votre fichier vitext et quitter VI. Copiez le texte ci-dessus puis collez-le dans un nouveau fichier dénommé **.exrc** dans votre répertoire personnel.

Ouvrez maintenant le fichier **/home/trainee/vitext** avec VI:

```
[trainee@centos7 ~]$ vi vitext
```

```
  1 This is line Linux is wonderful1$
  2 This is line 2$
  3 This is line 3$
  4 This is line 3$
  5 This is line 9$
  6 This is line 10$
  7 This is line 11$
  8 This is line 12$
  9 This is line 13$
 10 This is line 14$
```

```
 11 This is line 15$
 12 This is line 16$
 13 This is line 7$
 14 This is line 8$
 15 This is line 17$
 16 This is line 18$
 17 This is line 19$
 18 This is line 7$
 19 This is line 8$
 20 This is line 7$
 21 This is line 20$
 22 This is line 21$
 23 This is line 22$
 24 This is line 23$
 25 This is line 24Linux is wonderful$
"vitext" 26L, 442C                                          20,1          Top
```

LAB #9 - 38 Commandes de Base

A faire - Vous êtes actuellement connecté(e) en tant que root dans votre terminal. Avant de procéder plus loin, tapez la commande exit et appuyez sur la touch ↵ Entrée.

9.1 - La commande stty

Dès votre connexion à un système Linux, Il est conseillé de lancer la commande suivante

```
$ stty –a [Entrée]
```

Faites attention à utiliser des minuscules. En effet Linux différencie les minuscules et les majuscules.

```
[trainee@centos7 ~]$ stty -a
speed 38400 baud; rows 26; columns 80; line = 0;
intr = ^C; quit = ^\; erase = ^?; kill = ^U; eof = ^D; eol = <undef>;
eol2 = <undef>; swtch = <undef>; start = ^Q; stop = ^S; susp = ^Z; rprnt = ^R;
werase = ^W; lnext = ^V; flush = ^O; min = 1; time = 0;
-parenb -parodd -cmspar cs8 -hupcl -cstopb cread -clocal -crtscts
-ignbrk -brkint -ignpar -parmrk -inpck -istrip -inlcr -igncr icrnl ixon -ixoff
-iuclc -ixany -imaxbel -iutf8
opost -olcuc -ocrnl onlcr -onocr -onlret -ofill -ofdel nl0 cr0 tab0 bs0 vt0 ff0
isig icanon iexten echo echoe echok -echonl -noflsh -xcase -tostop -echoprt
echoctl echoke
```

Dans l’information qui s’affiche à l’écran, cherchez la chaîne intr =. Si la valeur est «Del», il faut utiliser la touche **Suppr** au lieu de la commande **^C** pour interrompre un programme en cours d’exécution dans un terminal.

Options de la commande

A faire : Utilisez l'option **–help** de la commande **stty** pour visualiser les options de la commande.

9.2 - La commande date

Cette commande affiche la date et l’heure de la machine. La commande peut aussi être utilisée pour régler la date du système :

```
[trainee@centos7 ~]$ date
Thu 29 Sep 04:53:58 CEST 2016
```

Options de la commande

A faire : Utilisez l'option **–help** de la commande **date** pour visualiser les options de la commande.

9.3 - La commande who

Cette commande affiche les utilisateurs connectés au système:

```
[trainee@centos7 ~]$ who
trainee  pts/0        2016-09-28 23:29 (gateway)
```

Options de la commande

A faire : Utilisez l'option **–help** de la commande **who** pour visualiser les options de la commande.

9.4 - La commande df

Cette commande affiche l'espace disque libre sur chacun des unités montés (connectés au système):

```
[trainee@centos7 ~]$ df
Filesystem     1K-blocks    Used Available Use% Mounted on
/dev/sda2       10229760 4274948   5954812  42% /
devtmpfs          236036       0    236036   0% /dev
tmpfs             250388       0    250388   0% /dev/shm
tmpfs             250388    4716    245672   2% /run
```

```
tmpfs             250388       0    250388   0% /sys/fs/cgroup
tmpfs             250388      72    250316   1% /tmp
/dev/sda1         201388  201368        20 100% /boot
tmpfs              50080       0     50080   0% /run/user/1000
```

Les unités sont en blocs. Afin d'*humaniser* la sortie, il est possible d'utiliser l'option **-h**. Une option est aussi connue sous le nom **parameter**, **switch** ou **flag** :

```
[trainee@centos7 ~]$ df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda2       9.8G  4.1G  5.7G  42% /
devtmpfs        231M     0  231M   0% /dev
tmpfs           245M     0  245M   0% /dev/shm
tmpfs           245M  4.7M  240M   2% /run
tmpfs           245M     0  245M   0% /sys/fs/cgroup
tmpfs           245M   72K  245M   1% /tmp
/dev/sda1       197M  197M   20K 100% /boot
tmpfs            49M     0   49M   0% /run/user/1000
```

Options de la commande

A faire : Utilisez l'option **–help** de la commande **df** pour visualiser les options de la commande.

9.5 - La commande free

Cette commande affiche les détails de la mémoire disponible :

```
[trainee@centos7 ~]$ free
              total        used        free      shared  buff/cache   available
```

```
Mem:          500780      192460       11580        3356      296740      245128
Swap:        2096124        3852     2092272
```

Les unités sont en blocs. Afin d'*humaniser* la sortie, il est possible d'utiliser l'option **-h**.

```
[trainee@centos7 ~]$ free -h
              total        used        free      shared  buff/cache   available
Mem:           489M        188M         11M        3.3M        289M        239M
Swap:          2.0G        3.8M        2.0G
```

Options de la commande

A faire : Utilisez l'option **–help** de la commande **free** pour visualiser les options de la commande.

9.6 - La commande whoami

Cette commande affiche le nom associé à l'UID courant effectif, autrement dit, le nom de votre compte courant :

```
[trainee@centos7 ~]$ whoami
trainee
```

Devenez maintenant l'administrateur **root** :

```
[trainee@centos7 ~]$ su -
Password: fenestros
Last login: Wed Sep 28 12:08:28 CEST 2016 on pts/0
```

Important : Notez que le mot de passe saisi ne sera PAS visible.

Saisissez maintenant la commande **whoami** de nouveau :

```
[root@centos7 ~]# whoami
root
[root@centos7 ~]#
```

Important : Notez maintenant que vous êtes root.

Saisissiez en suite la commande **exit** pour redevenir l'utilisateur **trainee** :

```
[root@centos7 ~]# exit
logout
[trainee@centos7 ~]$
```

Options de la commande

A faire : Utilisez l'option **–help** de la commande **whoami** pour visualiser les options de la commande.

9.7 - La commande pwd

Cette commande affiche le répertoire courant de travail :

```
[trainee@centos7 ~]$ pwd
```

```
/home/trainee
```

Options de la commande

A faire : Utilisez la commande **help** avec l'option **pwd** pour visualiser les options de la commande.

9.8 - La commande cd

Cette commande permet de changer de répertoire courant pour le répertoire passé en argument à la commande :

```
[trainee@centos7 ~]$ cd /tmp
[trainee@centos7 tmp]$ pwd
/tmp
[trainee@centos7 tmp]$
```

Options de la commande

A faire : Utilisez la commande **help** avec l'option **cd** pour visualiser les options de la commande.

9.9 - La commande ls

Cette commande permet de lister le contenu d'un répertoire passé en argument à la commande. Si aucun argument n'est spécifié, la commande liste le contenu du répertoire courant :

```
[trainee@centos7 tmp]$ ls
hsperfdata_root  inode  systemd-private-45071248a58f4fd2a4de940a5734c8ec-cups.service-ODc5L6
```

Options de la commande

A faire : Utilisez l'option **–help** de la commande **ls** pour visualiser les options de la commande.

9.10 - La commande lsof

La commande **lsof** affiche des informations sur les fichiers ouverts par des processus :

```
[trainee@centos7 tmp]$ su -
Password: fenestros
Last login: Thu Sep 29 06:24:28 CEST 2016 on pts/0
[root@centos7 ~]# lsof | more
COMMAND     PID   TID           USER   FD      TYPE             DEVICE  SIZE/OFF       NODE NAME
systemd       1                 root  cwd       DIR                8,2      4096        128 /
systemd       1                 root  rtd       DIR                8,2      4096        128 /
systemd       1                 root  txt       REG                8,2   1494056   33607555
/usr/lib/systemd/systemd
systemd       1                 root  mem       REG                8,2     20032   17241167
/usr/lib64/libuuid.so.1.3.0
systemd       1                 root  mem       REG                8,2    252696   20114965
/usr/lib64/libblkid.so.1.1.0
systemd       1                 root  mem       REG                8,2     90632   18751025
/usr/lib64/libz.so.1.2.7
systemd       1                 root  mem       REG                8,2     19888   17326380
/usr/lib64/libattr.so.1.1.0
systemd       1                 root  mem       REG                8,2     19520   17239651
```

```
/usr/lib64/libdl-2.17.so
systemd       1                 root  mem       REG                8,2    153192   18023397
/usr/lib64/liblzma.so.5.0.99
systemd       1                 root  mem       REG                8,2    398272   18105157
/usr/lib64/libpcre.so.1.2.0
systemd       1                 root  mem       REG                8,2   2107816   17239525
/usr/lib64/libc-2.17.so
systemd       1                 root  mem       REG                8,2    142304   17240882
/usr/lib64/libpthread-2.17.so
systemd       1                 root  mem       REG                8,2     88720   17511831
/usr/lib64/libgcc_s-4.8.5-20150702.so.1
systemd       1                 root  mem       REG                8,2     44096   17241011
/usr/lib64/librt-2.17.so
systemd       1                 root  mem       REG                8,2    260784   20114966
/usr/lib64/libmount.so.1.1.0
systemd       1                 root  mem       REG                8,2     91768   18023376
/usr/lib64/libkmod.so.2.2.10
systemd       1                 root  mem       REG                8,2    118792   17326381
/usr/lib64/libaudit.so.1.0.0
systemd       1                 root  mem       REG                8,2     61648   18105139
/usr/lib64/libpam.so.0.83.1
systemd       1                 root  mem       REG                8,2     20024   17326423
/usr/lib64/libcap.so.2.22
systemd       1                 root  mem       REG                8,2    147120   18447456
/usr/lib64/libselinux.so.1
systemd       1                 root  mem       REG                8,2    164440   21744303 /usr/lib64/ld-2.17.so
systemd       1                 root  mem       REG                8,2   1333123   50334524
/etc/selinux/targeted/contexts/files/file_contexts.bin
--More--
```

Options de la commande

A faire : Utilisez l'option **–help** de la commande **lsof** pour visualiser les options de la commande.

9.11 - La commande touch

Cette commande sert à modifier l'horodatage de la date de dernière modification du contenu (**mtime**) et la date du dernier accès (**atime**), d'un ou de plusieurs fichiers passé(s) en argument(s), selon la date courante. Si le(s) fichier(s) n'existe(nt) pas, il(s) est (sont) créé(s) :

```
[root@centos7 ~]# exit
logout
[trainee@centos7 tmp]$ touch test
[trainee@centos7 tmp]$ ls
hsperfdata_root  systemd-private-45071248a58f4fd2a4de940a5734c8ec-cups.service-ODc5L6
inode            test
```

Options de la commande

A faire : Utilisez l'option **–help** de la commande **touch** pour visualiser les options de la commande.

9.12 - La commande echo

Cette commande écrit les arguments vers la sortie standard (autrement dit à l'écran) :

```
[trainee@centos7 tmp]$ echo fenestros
fenestros
```

Options de la commande

A faire : Utilisez la commande **help** avec l'option **echo** pour visualiser les options de la commande.

9.13 - La commande cp

La commande cp permet de copier une source vers une destination ou de multiples sources vers un répertoire :

```
[trainee@centos7 tmp]$ cp test ~
[trainee@centos7 tmp]$ ls -l ~
total 4
drwxr-xr-x. 2 trainee trainee   6 Apr 30 11:54 Desktop
drwxr-xr-x. 2 trainee trainee   6 Apr 30 11:54 Documents
drwxr-xr-x. 2 trainee trainee   6 Apr 30 11:54 Downloads
drwxr-xr-x. 2 trainee trainee   6 Apr 30 11:54 Music
drwxr-xr-x. 2 trainee trainee   6 Apr 30 11:54 Pictures
drwxr-xr-x. 2 trainee trainee   6 Apr 30 11:54 Public
drwxr-xr-x. 2 trainee trainee   6 Apr 30 11:54 Templates
-rw-rw-r--. 1 trainee trainee   0 Sep 29 17:23 test
drwxr-xr-x. 2 trainee trainee   6 Apr 30 11:54 Videos
-rw-rw-r--. 1 trainee trainee 442 Sep 29 00:53 vitext
```

Important : Notez l'utilisation du caractère ~ (tilde) qui est un caractère spécial indiquant le répertoire personnel de l'utilisateur courant, dans ce cas /home/trainee.

Options de la commande

A faire : Utilisez l'option **–help** de la commande **cp** pour visualiser les options de la commande.

9.14 - La commande file

Cette commande permet de connaître le type d'un fichier:

```
[trainee@centos7 tmp]$ file ~/test
/home/trainee/test: empty
```

Important : Notez que la commande vous indique le type de fichier en fonction de son contenu. Dans l'exemple précédent, puisque le fichier est vide, la commande file ne peut pas indiqué le type de fichier.

Redirigez, en utilisant le caractère **>**, la sortie de la commande **echo** vers le fichier **/home/trainee/test** de façon à ce que ce dernier contient le texte **fenestros** :

```
[trainee@centos7 tmp]$ echo "fenestros" > ~/test
```

En utilisant de nouveau la commande **file**, celle-ci est capable de vous indiquer le type de fichier :

```
[trainee@centos7 tmp]$ file ~/test
/home/trainee/test: ASCII text
```

Options de la commande

A faire : Utilisez l'option **–help** de la commande **file** pour visualiser les options de la commande.

9.15 - La commande cat

La commande **cat** permet de concaténer les fichiers passés en argument, ou de l'entrée standard (le **clavier**), vers la sortie standard (l'**écran**). Dans le cas où il n'y a qu'un seul fichier passé en argument, le contenu de celui-ci est affiché à l'écran :

```
[trainee@centos7 tmp]$ cat ~/test
fenestros
```

Options de la commande

A faire : Utilisez l'option **–help** de la commande **cat** pour visualiser les options de la commande.

9.16 - La commande mv

La commande **mv** permet déplacer ou de renommer un fichier ou répertoire.

Utilisez la commande **mv** pour déplacer le fichier **test** de votre répertoire personnel vers le répertoire courant :

```
[trainee@centos7 tmp]$ mv ~/test .
[trainee@centos7 tmp]$ ls -l ~
total 4
```

```
drwxr-xr-x. 2 trainee trainee   6 Apr 30 11:54 Desktop
drwxr-xr-x. 2 trainee trainee   6 Apr 30 11:54 Documents
drwxr-xr-x. 2 trainee trainee   6 Apr 30 11:54 Downloads
drwxr-xr-x. 2 trainee trainee   6 Apr 30 11:54 Music
drwxr-xr-x. 2 trainee trainee   6 Apr 30 11:54 Pictures
drwxr-xr-x. 2 trainee trainee   6 Apr 30 11:54 Public
drwxr-xr-x. 2 trainee trainee   6 Apr 30 11:54 Templates
drwxr-xr-x. 2 trainee trainee   6 Apr 30 11:54 Videos
-rw-rw-r--. 1 trainee trainee 442 Sep 29 00:53 vitext
[trainee@centos7 tmp]$ mv test TeSt
[trainee@centos7 tmp]$ ls -l
total 4
drwxr-xr-x. 2 root    root     80 Sep 28 10:52 hsperfdata_root
drwxr-xr-x. 2 root    root    100 Sep 28 12:30 inode
drwx------. 3 root    root     60 Sep 28 10:49 systemd-private-45071248a58f4fd2a4de940a5734c8ec-cups.service-
ODc5L6
-rw-rw-r--. 1 trainee trainee  10 Sep 29 17:28 TeSt
```

Important : Notez l'utilisation du raccourci . pour indiquer le répertoire courant.

Options de la commande

A faire : Utilisez l'option **–help** de la commande **mv** pour visualiser les options de la commande.

9.17 - La commande mkdir

La commande **mkdir** permet de créer un répertoire.

```
[trainee@centos7 tmp]$ cd ~
[trainee@centos7 ~]$ mkdir testdir
[trainee@centos7 ~]$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  testdir  Videos  vitext
```

Options de la commande

A faire : Utilisez l'option **–help** de la commande **mkdir** pour visualiser les options de la commande.

9.18 - La commande rmdir

La commande **rmdir** permet de supprimer un répertoire **vide** :

```
[trainee@centos7 ~]$ rmdir testdir
[trainee@centos7 ~]$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos  vitext
```

Options de la commande

A faire : Utilisez l'option **–help** de la commande **rmdir** pour visualiser les options de la commande.

9.19 - La commande rm

La commande rm permet de supprimer un répertoire **vide ou non** ou un fichier :

```
[trainee@centos7 ~]$ mkdir testdir1
[trainee@centos7 ~]$ cd /tmp
[trainee@centos7 tmp]$ echo "fenestros" > TeSt
[trainee@centos7 tmp]$ cd ~
[trainee@centos7 ~]$ mv /tmp/TeSt ~/testdir1
[trainee@centos7 ~]$ ls -lR testdir1/
testdir1/:
total 4
-rw-rw-r--. 1 trainee trainee 10 Sep 29 18:13 TeSt
[trainee@centos7 ~]$ rmdir testdir1/
rmdir: failed to remove ‘testdir1/’: Directory not empty
[trainee@centos7 ~]$ rm -rf testdir1/
[trainee@centos7 ~]$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos  vitext
```

Options de la commande

A faire : Utilisez l'option **–help** de la commande **rm** pour visualiser les options de la commande.

9.20 - La commande sort

Cette commande trie dans le canal d’entrée et retourne à l’écran une liste triée.

```
[trainee@centos7 ~]$ touch aac abc bca xyz
```

```
[trainee@centos7 ~]$ ls
aac  bca      Documents  Music     Public     Videos  xyz
abc  Desktop  Downloads  Pictures  Templates  vitext
[trainee@centos7 ~]$ ls | sort
aac
abc
bca
Desktop
Documents
Downloads
Music
Pictures
Public
Templates
Videos
vitext
xyz
[trainee@centos7 ~]$ ls | sort -r
xyz
vitext
Videos
Templates
Public
Pictures
Music
Downloads
Documents
Desktop
bca
abc
aac
```

Important : Notez l'utilisation du caractère spécial |, appelé un tube ou encore un pipe. Un tube est utilisé pour présenter sur l'entrée

standard de la commande qui suit, la sortie standard de la commande qui précède.

Options de la commande

A faire : Utilisez l'option **–help** de la commande **sort** pour visualiser les options de la commande.

9.21 - La commande more

Cette commande affiche le contenu d'un fichier texte et l'envoi page par page au canal de sortie à l'aide de la touche Espace :

```
[trainee@centos7 ~]$ more /etc/services
# /etc/services:
# $Id: services,v 1.55 2013/04/14 ovasik Exp $
#
# Network services, Internet style
# IANA services version: last updated 2013-04-10
#
# Note that it is presently the policy of IANA to assign a single well-known
# port number for both TCP and UDP; hence, most entries here have two entries
# even if the protocol doesn't support UDP operations.
# Updated from RFC 1700, ``Assigned Numbers'' (October 1994).  Not all ports
# are included, only the more common ones.
#
# The latest IANA port assignments can be gotten from
#       http://www.iana.org/assignments/port-numbers
# The Well Known Ports are those from 0 through 1023.
# The Registered Ports are those from 1024 through 49151
# The Dynamic and/or Private Ports are those from 49152 through 65535
```

```
#
# Each line describes one service, and is of the form:
#
# service-name  port/protocol  [aliases ...]   [# comment]

tcpmux          1/tcp                           # TCP port service multiplexer
--More--(0%)
```

Important : L'utilisation de la touche ↵ Entrée permet de défiler le fichier ligne par ligne. L'utilisation de la touche Barre d'espace permet de défiler le fichier écran par écran. L'utilisation de la touche Q permets de revenir au prompt.

Options de la commande

A faire : Utilisez l'option **–help** de la commande **more** pour visualiser les options de la commande.

9.22 - La commande less

La commande **less** produit un résultat similaire à la commande **more**. Utilisez la commande **less** en vous référant à l'aide de la commande avec **less --help**. Laquelle des deux commandes vous semble la plus puissante ?

```
[trainee@centos7 ~]$ less /etc/services
# /etc/services:
# $Id: services,v 1.55 2013/04/14 ovasik Exp $
#
# Network services, Internet style
# IANA services version: last updated 2013-04-10
```

```
#
# Note that it is presently the policy of IANA to assign a single well-known
# port number for both TCP and UDP; hence, most entries here have two entries
# even if the protocol doesn't support UDP operations.
# Updated from RFC 1700, ``Assigned Numbers'' (October 1994).  Not all ports
# are included, only the more common ones.
#
# The latest IANA port assignments can be gotten from
#       http://www.iana.org/assignments/port-numbers
# The Well Known Ports are those from 0 through 1023.
# The Registered Ports are those from 1024 through 49151
# The Dynamic and/or Private Ports are those from 49152 through 65535
#
# Each line describes one service, and is of the form:
#
# service-name  port/protocol  [aliases ...]   [# comment]

tcpmux          1/tcp                           # TCP port service multiplexer
/etc/services
```

Options de la commande

A faire : Utilisez l'option **–help** de la commande **less** pour visualiser les options de la commande.

9.23 - La commande find

Cette commande sert à rechercher un ou des fichiers dans le répertoire courant ou le répertoire spécifié en argument :

```
[trainee@centos7 ~]$ find acc
```

```
find: ‘acc’: No such file or directory
[trainee@centos7 ~]$ find aac
aac
```

Important : Notez que si le fichier n'existe pas le système vous en informe clairement. Notez aussi que ce fichier existe le système vous en informe en vous indiquant son nom.

Options de la commande

A faire : Utilisez l'option **–help** de la commande **find** pour visualiser les options de la commande.

9.24 - La commande su

La commande su permet d'assumer l'identité d'un autre utilisateur du système à condition de connaître son mot de passe. Exécutée sans argument, le système suppose que vous souhaitez devenir **root** :

```
[trainee@centos7 ~]$ su -
Password: fenestros
Last login: Thu Sep 29 17:08:56 CEST 2016 on pts/0
```

Important : Notez que le mot de passe saisi ne sera PAS visible.

Options de la commande

A faire : Utilisez l'option **–help** de la commande **su** pour visualiser les options de la commande.

9.25 - Les commandes locate et updatedb

La commande **locate** sert à rechercher un ou des fichiers dans l'ensemble du système de fichiers en commençant à la racine (/) en spécifiant une chaîne à rechercher en argumant à la commande. La commande locate utilise une base de données afin d'effectuer sa recherche. Pour construire ou mettre à jour cette base de données avant l'utilisation de la commande pour une recherche, il faut utiliser la commande **updatedb** en tant que root.

La commande **updatedb** peut être configurée en éditant son fichier de configuration **/etc/updatedb.conf** :

```
[root@centos7 ~]# cat /etc/updatedb.conf
PRUNE_BIND_MOUNTS = "yes"
PRUNEFS = "9p afs anon_inodefs auto autofs bdev binfmt_misc cgroup cifs coda configfs cpuset debugfs devpts
ecryptfs exofs fuse fuse.sshfs fusectl gfs gfs2 hugetlbfs inotifyfs iso9660 jffs2 lustre mqueue ncpfs nfs nfs4
nfsd pipefs proc ramfs rootfs rpc_pipefs securityfs selinuxfs sfs sockfs sysfs tmpfs ubifs udf usbfs"
PRUNENAMES = ".git .hg .svn"
PRUNEPATHS = "/afs /media /mnt /net /sfs /tmp /udev /var/cache/ccache /var/lib/yum/yumdb /var/spool/cups
/var/spool/squid /var/tmp"
```

L'utilisation des deux commandes est illustrée ci-après :

```
[root@centos7 ~]# updatedb
[root@centos7 ~]# locate aac
/home/trainee/aac
/usr/lib/modules/3.10.0-327.13.1.el7.x86_64/kernel/drivers/scsi/aacraid
/usr/lib/modules/3.10.0-327.13.1.el7.x86_64/kernel/drivers/scsi/aacraid/aacraid.ko
/usr/lib/modules/3.10.0-327.el7.x86_64/kernel/drivers/scsi/aacraid
/usr/lib/modules/3.10.0-327.el7.x86_64/kernel/drivers/scsi/aacraid/aacraid.ko
```

```
/usr/lib/python2.7/site-packages/ipalib/plugins/caacl.py
/usr/lib/python2.7/site-packages/ipalib/plugins/caacl.pyc
/usr/lib/python2.7/site-packages/ipalib/plugins/caacl.pyo
/usr/share/gtk-doc/html/gst-plugins-good-plugins-1.0/gst-plugins-good-plugins-aacparse.html
/usr/share/gupnp-dlna-2.0/dlna-profiles/aac.xml
/usr/share/mime/audio/aac.xml
/usr/src/kernels/3.10.0-327.13.1.el7.x86_64/drivers/scsi/aacraid
/usr/src/kernels/3.10.0-327.13.1.el7.x86_64/drivers/scsi/aacraid/Makefile
/usr/src/kernels/3.10.0-327.13.1.el7.x86_64/include/config/scsi/aacraid.h
/usr/src/kernels/3.10.0-327.el7.x86_64/drivers/scsi/aacraid
/usr/src/kernels/3.10.0-327.el7.x86_64/drivers/scsi/aacraid/Makefile
/usr/src/kernels/3.10.0-327.el7.x86_64/include/config/scsi/aacraid.h
```

La base de données par défaut est **/var/lib/mlocate/mlocate.db** :

```
[root@centos7 ~]# ls -l /var/lib/mlocate/mlocate.db
-rw-r-----. 1 root slocate 3887117 Sep 29 03:46 /var/lib/mlocate/mlocate.db
```

Important : Pour plus d'information concernant le format de la base de données, consultez **man 5 locatedb**.

Options des commandes

A faire : Utilisez l'option **–help** des commandes **updatedb** et **locate** pour visualiser les options des commandes.

9.26 - La commande whereis

La commande **whereis** permet une recherche de l'emplacement des exécutables, des fichiers de configuration et des manuels pour la commande passée en argument :

```
[root@centos7 ~]# whereis passwd
passwd: /usr/bin/passwd /etc/passwd /usr/share/man/man1/passwd.1.gz /usr/share/man/man5/passwd.5.gz
```

Options de la commande

A faire : Utilisez l'option **–help** de la commande **whereis** pour visualiser les options de la commande.

9.27 - La commande which

La commande **which** permet une recherche de l'emplacement d'un exécutable dans le PATH de l'utilisateur courant et retourne le premier qui est trouvé :

```
[root@centos7 ~]# which passwd
/bin/passwd
```

Options de la commande

A faire : Utilisez l'option **–help** de la commande **which** pour visualiser les options de la commande.

9.28 - La commande uptime

Cette commande nous indique l'heure actuelle, la durée depuis laquelle le système fonctionne, le nombre d'utilisateurs actuellement connectés et la charge système moyenne pour les dernières 1 minute, 5 minutes et 15 minutes :

```
[root@centos7 ~]# uptime
 19:18:57 up 1 day,  8:30,  1 user,  load average: 0.10, 0.09, 0.07
```

Options de la commande

A faire : Utilisez l'option **–help** de la commande **uptime** pour visualiser les options de la commande.

9.29 - La commande w

Cette commande reprend les informations de la commande **uptime** et y ajoute des détails sur les utilisateurs connectés via un terminal :

```
[root@centos7 ~]# w
 19:28:55 up 1 day,  8:40,  1 user,  load average: 0.00, 0.01, 0.05
USER     TTY      FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
trainee  pts/0    gateway          15:22    7.00s  0.16s  0.12s sshd: trainee [priv]
```

La valeur JCPU indique le temps processeur utilisé par tous les processus attachés au terminal de la connexion. Cette valeur n'inclut pas les temps des anciens processus en arrière plan.

La valeur PCPU indique le temps processeur utilisé par les processus attachés au terminal de la connexion et actuellement en cours (autrement dit le processus identifié dans la colonne **WHAT**).

Options de la commande

A faire : Utilisez l'option **–help** de la commande **w** pour visualiser les options de la commande.

9.30 - La commande uname

Cette commande affiche des informations sur le système :

```
[root@centos7 ~]# uname -a
Linux centos7.fenestros.loc 3.10.0-327.13.1.el7.x86_64 #1 SMP Thu Mar 31 16:04:38 UTC 2016 x86_64 x86_64 x86_64
GNU/Linux
[root@centos7 ~]# uname -s
Linux
[root@centos7 ~]# uname -n
centos7.fenestros.loc
[root@centos7 ~]# uname -r
3.10.0-327.13.1.el7.x86_64
[root@centos7 ~]# uname -v
#1 SMP Thu Mar 31 16:04:38 UTC 2016
[root@centos7 ~]# uname -m
x86_64
[root@centos7 ~]# uname -p
x86_64
[root@centos7 ~]# uname -i
x86_64
[root@centos7 ~]# uname -o
GNU/Linux
```

Options de la commande

A faire : Utilisez l'option **–help** de la commande **uname** pour visualiser les options de la commande.

9.31 - La commande du

La commande du peut être utilisée pour afficher la taille des fichiers contenus dans les répertoires passés en arguments. L'utilisation suivante de la commande avec les options -s et -h sur la racine du système affiche la somme des sous-répertoires avec un affichage *humanisé* en Ko, Mo et Go :

```
[root@centos7 ~]# du -sh /* 2>/dev/null
0   /bin
187M    /boot
0   /dev
33M /etc
20M /home
0   /lib
0   /lib64
0   /lost+found
0   /media
0   /mnt
173M    /opt
0   /proc
43M /root
8.7M    /run
0   /sbin
0   /srv
0   /sys
64K /tmp
3.6G    /usr
228M    /var
```

Important : Notez l'utilisation de la redirection **2>/dev/null**. Cette chaîne envoie les erreurs éventuelles, contenues dans le canal 2 appelé le canal des erreurs, à **/dev/null** de façon à ce que les erreurs n'apparaissent pas à l'écran. Le canal des erreurs sera couvert dans le cours **La Ligne de Commande**.

Options de la commande

A faire : Utilisez l'option **–help** de la commande **du** pour visualiser les options de la commande.

9.32 - La commande clear

Cette commande est utilisée pour effacer le contenu de l'écran courant du terminal :

```
[root@centos7 ~]# clear

[root@centos7 ~]#
```

9.33 - La commande exit

Cette commande ferme le terminal courant :

```
[root@centos7 ~]# exit
logout
[trainee@centos7 ~]$
```

Options de la commande

A faire : Utilisez la commande **help** avec l'option **exit** pour visualiser les options de la commande.

9.34 - La commande logout

Cette commande est utilisée pour se déconnecter d'un terminal de connexion en écrivant les données umtp et wmtp dans les fichiers de journalisation.

Options de la commande

A faire : Utilisez la commande **help** avec l'option **logout** pour visualiser les options de la commande.

9.35 - La commande sleep

Cette commande pause le terminal pour le nombre de secondes passé en argument.

Options de la commande

A faire : Utilisez l'option **–help** de la commande **sleep** pour visualiser les options de la commande.

9.36 - La Commande wall

La commande **wall** envoie un message à tous les utilisateurs connectés, dont l’autorisation mesg(1) est positionnée à yes. Le message peut être fourni en tant qu’argument sur la ligne de commande, ou il peut être envoyé sur l’entrée standard de wall. Quand l’entrée standard est un terminal, le message doit se terminer par la touche EOF (Ctrl-D).La longueur du message est limitée à 20 lignes. Pour chaque invocation de wall, une notification sera écrite dans syslog avec la facilité LOG_USER et le niveau LOG_INFO.

Ouvrez donc un deuxième terminal via ssh en tant que trainee puis dans le premier envoyez un message :

```
[trainee@centos7 ~]$ su -
Password: fenestros
[root@centos7 ~]# wall this is a message from root
[root@centos7 ~]#
Broadcast message from root@centos7.fenestros.loc (pts/0) (Fri Sep 30 05:35:26 2016):

this is a message from root
[Enter]
[root@centos7 ~]#
```

Constatez que le message est affiché dans le deuxième terminal :

```
[trainee@centos7 ~]$
Broadcast message from root@centos7.fenestros.loc (pts/0) (Fri Sep 30 05:35:26 2016):

this is a message from root
```

Consultez maintenant la fin du fichier **/var/log/messages** :

```
[root@centos7 ~]# tail /var/log/messages
Sep 30 05:31:08 centos7 sh: Sleeping '' ''
Sep 30 05:34:16 centos7 sh: Sleeping '' ''
Sep 30 05:35:13 centos7 systemd: Started Session 311 of user trainee.
Sep 30 05:35:13 centos7 systemd-logind: New session 311 of user trainee.
Sep 30 05:35:13 centos7 systemd: Starting Session 311 of user trainee.
Sep 30 05:35:13 centos7 dbus[515]: [system] Activating service name='org.freedesktop.problems' (using
servicehelper)
Sep 30 05:35:13 centos7 dbus-daemon: dbus[515]: [system] Activating service name='org.freedesktop.problems'
(using servicehelper)
Sep 30 05:35:13 centos7 dbus[515]: [system] Successfully activated service 'org.freedesktop.problems'
Sep 30 05:35:13 centos7 dbus-daemon: dbus[515]: [system] Successfully activated service
'org.freedesktop.problems'
Sep 30 05:35:26 centos7 wall[27550]: wall: user root broadcasted 1 lines (29 chars)
```

Important : La commande wall ignore la variable d’environnement TZ. L’heure affichée dans la première page est basée sur les paramètres de régionalisation du système.

9.37 - La Commande seq

La commande **seq** affiche une séquence de nombres du PREMIER au DERNIER par pas d'un INCREMENT. La commande prend la forme suivante :

- **seq** [options] PREMIER
- **seq** [options] PREMIER DERNIER
- **seq** [options] PREMIER INCREMENT DERNIER

Par exemple :

```
[root@centos7 ~]# seq 10
1
2
3
4
5
6
7
8
9
10
[root@centos7 ~]# seq 20 30
20
21
22
23
24
```

```
25
26
27
28
29
30
[root@centos7 ~]# seq 20 10 90
20
30
40
50
60
70
80
90
[root@centos7 ~]#
```

Options de la commande

A faire : Utilisez l'option **–help** de la commande **seq** pour visualiser les options de la commande.

9.38 - La Commande screen

La commande **screen** est un « multiplexeur de terminaux » permettant d'ouvrir jusqu'à 10 (numérotés de 0 à 9) terminaux dans une même console, de passer de l'un à l'autre et de les récupérer plus tard.

La commande **screen** n'est pas installée par défaut sous RHEL/CentOS 7. Installez donc le paquet du même nom que la commande :

```
[root@centos7 ~]#  which screen
```

```
/usr/bin/which: no screen in (/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin:/root/bin)
[root@centos7 ~]#  yum install screen
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
 * base: ftp.rezopole.net
 * extras: centos.mirror.ate.info
 * updates: centos.mirror.fr.planethoster.net
Resolving Dependencies
--> Running transaction check
---> Package screen.x86_64 0:4.1.0-0.23.20120314git3c2946.el7_2 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
 Package    Arch       Version                                   Repository   Size
================================================================================
Installing:
 screen     x86_64     4.1.0-0.23.20120314git3c2946.el7_2      updates     552 k

Transaction Summary
================================================================================
Install  1 Package

Total download size: 552 k
Installed size: 914 k
Is this ok [y/d/N]: y
Downloading packages:
screen-4.1.0-0.23.20120314git3c2946.el7_2.x86_64.rpm      | 552 kB  00:00:01
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : screen-4.1.0-0.23.20120314git3c2946.el7_2.x86_64              1/1
```

```
  Verifying  : screen-4.1.0-0.23.20120314git3c2946.el7_2.x86_64              1/1

Installed:
  screen.x86_64 0:4.1.0-0.23.20120314git3c2946.el7_2

Complete!
[root@centos7 ~]# which screen
/bin/screen
```

Créez maintenant une session avec screen :

```
[root@centos7 ~]# screen -S mysession
```

Important - Vous verrez en tant que titre de votre terminal la chaîne **[screen 0: root@centos7:~]**.

Appuyez maintenant sur CTRL A puis relachez la touche A et appuyez sur la touche C pour créer un deuxième screen imbriqué dans la même session.

Important - Vous verrez en tant que titre de votre terminal la chaîne **[screen 1: root@centos7:~]**.

Pour repasser au premier screen, appuyez sur CTRL A A. Ceci permet de basculer entre les deux derniers screens.

Pour voir les screens actifs, utilisez la commande suivante :

```
[root@centos7 ~]# screen -ls
There is a screen on:
        25812.mysession (Attached)
1 Socket in /var/run/screen/S-root.
```

Dans votre screen, saisissez les commandes suivantes :

```
[root@centos7 ~]# sleep 9999 &
[1] 2272
[root@centos7 ~]# jobs
[1]+  Running                 sleep 9999 &
[root@centos7 ~]#
```

Pour détacher le screen, appuyez sur CTRL A puis relachez la touche A et appuyez sur la touche D :

```
[root@centos7 ~]# screen -S mysession
[detached from 25812.mysession]
[root@centos7 ~]#
```

Pour rattacher le screen, saisissez la commande suivante :

```
[root@centos7 ~]# screen -r
```

Utilisez la commande jobs pour vérifier si le processus créé par la commande sleep est toujours en cours de fonctionnement :

```
[root@centos7 ~]# jobs
[1]+  Running                 sleep 9999 &
```

Pour naviguer entre les screens il convient d'appuyer sur CTRL A puis relachez la touche A et appuyez sur la touche N ou d'appuyer sur CTRL A puis relachez la touche A et appuyez sur la touche P.

Détachez de nouveau le screen actuel en appuyant sur CTRL A puis en relachant la touche A et en appuyant sur la touche D :

```
[root@centos7 ~]# screen -S mysession
[detached from 25812.mysession]
[root@centos7 ~]#
```

Créez maintenant un autre screen, cette fois-ci, non imbriqué :

```
[root@centos7 ~]# screen -S mysession1
```

Constatez le résultat :

```
[root@centos7 ~]# screen -ls
There are screens on:
        14942.mysession1        (Attached)
        25812.mysession (Detached)
2 Sockets in /var/run/screen/S-root.
```

Ratachez maintenant le screen **mysession** :

```
[root@centos7 ~]# screen -r 25812
```

Constatez de nouveau le résultat :

```
[root@centos7 ~]# sleep 9999 &
[1] 2272
[root@centos7 ~]# jobs
[1]+  Running                 sleep 9999 &
[root@centos7 ~]# jobs
[1]+  Running                 sleep 9999 &
[root@centos7 ~]# screen -ls
There is a screen on:
        25812.mysession (Attached)
1 Socket in /var/run/screen/S-root.

[root@centos7 ~]# screen -ls
There are screens on:
        14942.mysession1        (Attached)
        25812.mysession (Attached)
2 Sockets in /var/run/screen/S-root.
```

Options de la commande

A faire : Utilisez l'option **–help** de la commande **screen** pour visualiser les options de la commande.

LAB #10 - Options et Arguments

Les options sous Linux peuvent être exprimées au format court ou au format long. Plusieurs différences sont importantes à noter.

Premièrement les options courtes sont précédées par un simple tiré **-**, tandis que les options longues sont précédées par deux tirés **–**.

Un exemple est l'option de l'aide pour la plupart des commandes bash :

- -h
- –help

Deuxièmement les options courtes peuvent être combinées tandis que les options longues ne peuvent pas l'être. Par exemple, la ligne de commande **ls -l -a -i** peut être aussi écrite **ls -lai**, **ls -lia** ou encore **ls -ali** :

```
[root@centos7 ~]# ls -lai /tmp
total 244
  11071 drwxrwxrwt. 10 root root    260 Sep 30 06:24 .
    128 dr-xr-xr-x. 18 root root   4096 Dec 10  2015 ..
  13635 drwxrwxrwt.  2 root root     40 Sep 28 10:48 .font-unix
  24858 drwxr-xr-x.  2 root root    100 Sep 30 06:05 hsperfdata_root
  13597 drwxrwxrwt.  2 root root     40 Sep 28 10:48 .ICE-unix
 287056 drwxr-xr-x.  2 root root    100 Sep 28 12:30 inode
  24861 srw-------.  1 root root      0 Sep 28 10:52 .java_pid3213
  25344 srw-------.  1 root root      0 Sep 28 10:52 .java_pid3302
  16615 drwx------.  3 root root     60 Sep 28 10:49 systemd-private-45071248a58f4fd2a4de940a5734c8ec-
cups.service-ODc5L6
```

```
  13595 drwxrwxrwt.  2 root root     40 Sep 28 10:48 .Test-unix
  13590 drwxrwxrwt.  2 root root     40 Sep 28 10:48 .X11-unix
  13629 drwxrwxrwt.  2 root root     40 Sep 28 10:48 .XIM-unix
8098838 -rw-------.  1 root root 242724 Sep 30 05:45 yum_save_tx.2016-09-30.05-45.FF7KO4.yumtx
[root@centos7 ~]# ls -ali /tmp
total 244
  11071 drwxrwxrwt. 10 root root    260 Sep 30 06:24 .
    128 dr-xr-xr-x. 18 root root   4096 Dec 10  2015 ..
  13635 drwxrwxrwt.  2 root root     40 Sep 28 10:48 .font-unix
  24858 drwxr-xr-x.  2 root root    100 Sep 30 06:05 hsperfdata_root
  13597 drwxrwxrwt.  2 root root     40 Sep 28 10:48 .ICE-unix
 287056 drwxr-xr-x.  2 root root    100 Sep 28 12:30 inode
  24861 srw-------.  1 root root      0 Sep 28 10:52 .java_pid3213
  25344 srw-------.  1 root root      0 Sep 28 10:52 .java_pid3302
  16615 drwx------.  3 root root     60 Sep 28 10:49 systemd-private-45071248a58f4fd2a4de940a5734c8ec-
cups.service-ODc5L6
  13595 drwxrwxrwt.  2 root root     40 Sep 28 10:48 .Test-unix
  13590 drwxrwxrwt.  2 root root     40 Sep 28 10:48 .X11-unix
  13629 drwxrwxrwt.  2 root root     40 Sep 28 10:48 .XIM-unix
8098838 -rw-------.  1 root root 242724 Sep 30 05:45 yum_save_tx.2016-09-30.05-45.FF7KO4.yumtx
```

La commande **ls -l -all -inode** ne peut pas être écrite **ls -l -allinode** :

```
[root@centos7 ~]# ls -l --all --inode /tmp
total 244
  11071 drwxrwxrwt. 10 root root    260 Sep 30 06:24 .
    128 dr-xr-xr-x. 18 root root   4096 Dec 10  2015 ..
  13635 drwxrwxrwt.  2 root root     40 Sep 28 10:48 .font-unix
  24858 drwxr-xr-x.  2 root root    100 Sep 30 06:05 hsperfdata_root
  13597 drwxrwxrwt.  2 root root     40 Sep 28 10:48 .ICE-unix
 287056 drwxr-xr-x.  2 root root    100 Sep 28 12:30 inode
  24861 srw-------.  1 root root      0 Sep 28 10:52 .java_pid3213
  25344 srw-------.  1 root root      0 Sep 28 10:52 .java_pid3302
  16615 drwx------.  3 root root     60 Sep 28 10:49 systemd-private-45071248a58f4fd2a4de940a5734c8ec-
```

```
cups.service-ODc5L6
  13595 drwxrwxrwt.  2 root root     40 Sep 28 10:48 .Test-unix
  13590 drwxrwxrwt.  2 root root     40 Sep 28 10:48 .X11-unix
  13629 drwxrwxrwt.  2 root root     40 Sep 28 10:48 .XIM-unix
8098838 -rw-------.  1 root root 242724 Sep 30 05:45 yum_save_tx.2016-09-30.05-45.FF7KO4.yumtx

[root@centos7 ~]# ls -l --allinode /tmp
ls: unrecognized option '--allinode'
Try 'ls --help' for more information
```

Important : Les options prenant un argument ne sont pas combinées avec les autres options.

LAB #11 - Expressions Régulières

La manipulation de fichiers textes utilise des **expressions régulières**. Sous Linux il existe deux types d'expressions régulières :

- expressions régulières basiques - IEEE POSIX Basic Regular Expressions, appelées **ERb**,
  - utilisées par les commandes **vi**, **grep**, **expr** et **sed**,
- expressions régulières étendues - IEEE POSIX Extended Regular Expressions, appelées **ERe**,
  - utilisées par les commandes **egrep** (grep -E) et **awk**.

Les expressions régulières utilisent des caractères spéciaux. Certains caractères sont communs aux Erb et aux Ere :

Caractère spécial	Description
^	Trouver la chaîne au début de la ligne
$	Trouver la chaîne à la fin de la ligne
\	Annuler l'effet spécial du caractère suivant
[ ]	Trouver n'importe quel des caractères entre les crochets
[^]	Exclure les caractères entre crochets

Caractère spécial	Description
.	Trouver n'importe quel caractère sauf à la fin de la ligne
*	Trouver 0 ou plus du caractère qui précède
\<	Trouver la chaîne au début d'un mot
\>	Trouver la chaîne à la fin d'un mot

ERb

Certains caractères spéciaux sont spécifiques aux ERb :

Caractère spécial	Description
\{x,y\}	Trouver de **x** à **y** occurrences de ce qui précède
\{x\}	Trouver exactement le nombre **x** d'occurrences de ce qui précède
\{x,\}	Trouver le nombre **x** ou plus d'occurrences de ce qui précède
\(ERb)	Mémoriser une ERb
\1	Rappeler la première ERb mémorisée
\2, \3 …	Rappeler la deuxième ERb mémorisée, rappeler la troisième ERb mémorisée etc

ERe

Certains caractères spéciaux sont spécifiques aux ERe :

Caractère spécial	Description
?	Trouver 0 ou 1 occurrence de ce qui précède
+	Trouver 1 ou **n** d'occurrences de ce qui précède
{x,y}	Trouver de **x** à **y** occurrences de ce qui précède
{x}	Trouver exactement le nombre **x** d'occurrences de ce qui précède
{x,}	Trouver le nombre **x** ou plus d'occurrences de ce qui précède
()	Faire un **ET** des expressions régulières entre les paranthèses
|	Faire un **OU** des expressions régulières se trouvant de chaque côté du pipe

Outils de Manipulation de Fichiers Texte

Présentation des Commandes grep, egrep et fgrep

La commande grep

La commande grep peut être utilisée pour rechercher des lignes contenant une chaîne de caractères dans un jeu de fichiers.

Par défaut, la commande grep est sensible à la casse. Pour rendre cette commande insensible à la casse, il faut utiliser l'option **-i**.

La commande grep peut être aussi utilisée pour faire l'inverse, autrement dit de montrer les lignes qui ne contiennent pas la chaîne recherchée. Dans ce cas, il faut utiliser l'option **–v**.

La commande grep peut être utilisée avec des **Expressions Régulières basiques**. Ceci est utile pour rechercher dans le contenu de fichiers.

Options de la commande

A faire : Utilisez l'option **–help** de la commande **grep** pour visualiser les options de la commande.

La Commande egrep

La commande **egrep** est identique à la commande **grep -E**. Dans les deux cas, l'utilisation des expressions régulières est étendue aux ERe.

Options de la commande

A faire : Utilisez l'option **–help** de la commande **egrep** pour visualiser les options de la commande.

La Commande fgrep

La commande **fgrep** est identique à la commande **grep -F**. Dans les deux cas et par défaut la recherche concerne une chaîne de caractères interprétés dans un sens littéral sans utilisation de caractères spéciaux ni d'expressions régulières.

Options de la commande

A faire : Utilisez l'option **–help** de la commande **fgrep** pour visualiser les options de la commande.

LAB #12 - Utiliser grep, egrep et fgrep

Téléchargez le fichier **greptest** vers votre machine virtuelle en double-cliquant sur le titre ci-dessous :

greptest

```
fenestrOS
fenestros
555-5555
f
.fenestros
.fe
£
```

Placez ce fichier dans le répertoire **/tmp** :

```
[root@centos7 ~]# mv /home/trainee/Downloads/greptest /tmp/greptest
```

Recherchez maintenant toute ligne du fichier **/tmp/greptest** contenant au moins une lettre :

```
[root@centos7 ~]# grep '[a-zA-Z]' /tmp/greptest
fenestrOS
fenestros
f
.fenestros
.fe
```

Recherchez maintenant toute ligne contenant au moins une lettre ou un chiffre :

```
[root@centos7 ~]# grep '[a-zA-Z0-9]' /tmp/greptest
fenestrOS
fenestros
555-5555
f
.fenestros
.fe
```

Important : Notez la présence de la ligne 555-5555.

Recherchez maintenant toute ligne contenant un numéro de téléphone au format NNN-NNNN :

```
[root@centos7 ~]# grep '[0-9]\{3\}-[0-9]\{4\}' /tmp/greptest
555-5555
```

Recherchez maintenant toute ligne contenant exactement un caractère :

```
[root@centos7 ~]# grep '^.$' /tmp/greptest
f
£
```

Important : Notez l'utilisation des caractères spéciaux le début de ligne : ^, n'importe quel caractère : . et la fin de ligne : $.

Recherchez maintenant toute ligne commençant par un point :

```
[root@centos7 ~]# grep '^\.' /tmp/greptest
.fenestros
.fe
```

Important : Notez l'utilisation du caractère d'échappement \ pour annuler l'effet du caractère spécial **.**

La commande grep peut aussi être utilisée pour rechercher une chaîne dans tous les fichiers d'un répertoire spécifié : **grep -rnw 'directory' -e “pattern”**. Vous pouvez aussi spécifier les extentions des fichiers dans lesquels vous voulez rechercher la chaîne : **grep –include={*.doc,*.xls} -rnw 'directory' -e “pattern”**. Dernièrement vous pouvez exclure des fichiers de la recherche de la façon suivante : **grep –exclude=*.doc -rnw 'directory' -e “pattern”**.

Téléchargez le fichier **greptest** modifié vers votre machine virtuelle en double-cliquant sur le titre ci-dessous :

greptest

```
# Starting comment
fenestrOS
```

```
fenestros
# Another comment
555-5555
f

.fenestros

.fe

£
# End comment
```

Placez ce fichier dans le répertoire **/tmp** :

```
[root@centos7 ~]# mv /home/trainee/Downloads/greptest /tmp/greptest
```

Utilisez maintenant la commande **grep** avec l'option **-E** pour supprimer les lignes de commentaires ainsi que les lignes vides :

```
[root@centos7 ~]# grep -E -v '^(#|$)' /tmp/greptest
fenestrOS
fenestros
555-5555
f
.fenestros
.fe
£
```

Important : Notez l'utilisation des parenthèses pour faire un regroupement ainsi que le pipe pour représenter un OU. L'expression '^(#|$)' indique donc “toute ligne commençant par le caractère #” OU “toute ligne où le début de la ligne est aussi la fin de la ligne”.

Utilisez maintenant la commande **egrep** pour envoyer le contenu du fichier **/tmp/greptest**, sans commentaires et sans lignes vides, dans le fichier **/tmp/greptest1** :

```
[root@centos7 ~]# egrep -v '^(#|$)'  /tmp/greptest > /tmp/greptest1
[root@centos7 ~]# cat /tmp/greptest1
fenestrOS
fenestros
555-5555
f
.fenestros
.fe
£
```

Important : Cette commande est particulièrement utile face à un fichier de configuration de plusieurs centaines de lignes dont certaines contiennent des directives activées d'autres sont vides ou en commentaires. De cette façon vous pouvez généré facilement un fichier ne contenant que les directives activées.

Téléchargez le fichier **greptest** modifié vers votre machine virtuelle en double-cliquant sur le titre ci-dessous :

greptest

```
# Starting comment
^ This line will be used to demonstrate the use of fgrep
fenestrOS
fenestros
# Another comment
555-5555
f

.fenestros
```

```
.fe

£
# End comment
```

Placez ce fichier dans le répertoire **/tmp** :

```
[root@centos7 ~]# mv /home/trainee/Downloads/greptest /tmp/greptest
```

Utilisez maintenant la commande **fgrep** pour rechercher la ligne commençant par le caractère **^** :

```
[root@centos7 ~]# fgrep '^' /tmp/greptest
^ voici une ligne pour la recherche fgrep
```

Comparez le résultat ci-dessus avec celui de la commande grep :

```
[root@centos7 ~]# grep '^' /tmp/greptest
# Commentaire du début
^ voici une ligne pour la recherche fgrep
fenestrOS
fenestros
# Un autre commentaire
555-5555
f

.fenestros

.fe

£
# Commentaire de la fin
```

En effet, la ligne de commande en utilisant la commande grep devrait être :

```
[root@centos7 ~]# grep '^\^' /tmp/greptest
^ voici une ligne pour la recherche fgrep
```

Présentation de la Commande sed

La commande **sed** ou *Stream EDitor* est un éditeur de texte non-intéractif. Les actions spécifiées par la commande sed sont exécutées par défaut sur chaque ligne du fichier. La commande sed ne modifie pas le fichier d'origine et sa sortie standard est le canal 1.

Si plusieurs actions sont spécifiées dans la ligne de commande, chacune doit être précédée par l'option **-e**.

La syntaxe de la commande sed est la suivante :

```
sed [adresse] commande [arguments]
```

L'**adresse** permet de stipuler les lignes concernées par la **commande**.

La syntaxe d'une adresse peut être :

adresse	Lignes concernées
a	La ligne numéro **a**
$	La dernière ligne
/ERb/	Les lignes qui correspondent à l'ERb
a,b	De la ligne numéro **a** jusqu'à la ligne numéro **b**
/ERb1/, /ERb2/	Toutes les lignes entre la première occurrence correspondant à l'ERb1 jusqu'à la première occurrence correspondant à l'ERb2

Le commandes de sed sont :

commande	Description
d	Ne pas afficher la ou les ligne(s)
p	Afficher la ou les ligne(s)
s	Effectuer une substitution
w	Ecrire le ou les ligne(s) dans un fichier

commande	Description
=	Afficher le numéro de la ligne spécifiée
!	Exécuter la commande ci-dessus sur toutes les lignes sauf celle spécifiées dans l'adresse

Options de la commande

A faire : Utilisez l'option **–help** de la commande **sed** pour visualiser les options de la commande.

LAB #13 - Utiliser la Commande sed

La commande **d** de sed permet de ne pas afficher certaines lignes à l'écran. Dans l'exemple qui suit, les 10 premières lignes du fichier **/etc/services** ne sont pas affichées à l'écran :

```
[root@centos7 ~]# sed '1,10d' /etc/services | more
# are included, only the more common ones.
#
# The latest IANA port assignments can be gotten from
#       http://www.iana.org/assignments/port-numbers
# The Well Known Ports are those from 0 through 1023.
# The Registered Ports are those from 1024 through 49151
# The Dynamic and/or Private Ports are those from 49152 through 65535
#
# Each line describes one service, and is of the form:
#
# service-name  port/protocol  [aliases ...]   [# comment]

tcpmux          1/tcp                           # TCP port service multiplexer
tcpmux          1/udp                           # TCP port service multiplexer
rje             5/tcp                           # Remote Job Entry
```

```
rje             5/udp                           # Remote Job Entry
echo            7/tcp
echo            7/udp
discard         9/tcp           sink null
discard         9/udp           sink null
systat          11/tcp          users
systat          11/udp          users
daytime         13/tcp
--Plus--
```

Dans l'exemple qui suit, sed n'affiche pas de lignes de commentaires, c'est-à-dire les lignes commençant par le caractère **#** :

```
[root@centos7 ~]# sed '/^#/d' /etc/services | more

tcpmux          1/tcp                           # TCP port service multiplexer
tcpmux          1/udp                           # TCP port service multiplexer
rje             5/tcp                           # Remote Job Entry
rje             5/udp                           # Remote Job Entry
echo            7/tcp
echo            7/udp
discard         9/tcp           sink null
discard         9/udp           sink null
systat          11/tcp          users
systat          11/udp          users
daytime         13/tcp
daytime         13/udp
qotd            17/tcp          quote
qotd            17/udp          quote
msp             18/tcp                          # message send protocol
msp             18/udp                          # message send protocol
chargen         19/tcp          ttytst source
chargen         19/udp          ttytst source
ftp-data        20/tcp
ftp-data        20/udp
```

```
ftp             21/tcp
ftp             21/udp          fsp fspd
--Plus--
```

Important : Notez que l'ERb est entourée des caractères / et /.

La commande sed vous permet d'afficher à l'écran certaines lignes spécifiées en utilisant la commande **p** :

```
[root@centos7 ~]# sed '1,2p' /etc/passwd
root:x:0:0:root:/root:/bin/bash
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
...
```

Important : Notez que sed affiche également tout le contenu du fichier. Ceci implique que les lignes 1 et 2 s'affichent deux fois.

Pour n'afficher que les lignes spécifiées, il convient d'utiliser l'option **-n** :

```
[root@centos7 ~]# sed -n '1,2p' /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
```

La commande **w** permet d'écrire dans un fichier. Par exemple pour écrire dans le fichier **/tmp/sedtest** toutes les lignes du fichier **/etc/services** ne commençant pas par le caractère **#**, il convient d'utiliser la commande suivante :

```
[root@centos7 ~]# sed -n '/^#/!w /tmp/sedtest' /etc/services
[root@centos7 ~]# more /tmp/sedtest

tcpmux          1/tcp                           # TCP port service multiplexer
tcpmux          1/udp                           # TCP port service multiplexer
rje             5/tcp                           # Remote Job Entry
rje             5/udp                           # Remote Job Entry
echo            7/tcp
echo            7/udp
discard         9/tcp           sink null
discard         9/udp           sink null
systat          11/tcp          users
systat          11/udp          users
daytime         13/tcp
daytime         13/udp
qotd            17/tcp          quote
qotd            17/udp          quote
msp             18/tcp                          # message send protocol
msp             18/udp                          # message send protocol
chargen         19/tcp          ttytst source
chargen         19/udp          ttytst source
ftp-data        20/tcp
ftp-data        20/udp
ftp             21/tcp
ftp             21/udp          fsp fspd
--Plus--(0%)
```

La commande **s** permet de procéder à une substitution :

```
[root@centos7 ~]# echo "user1,user2,user3" > /tmp/sedtest1
[root@centos7 ~]# cat /tmp/sedtest1 | sed 's/,/ /g'
user1 user2 user3
```

Important : Notez que dans cet exemple, la commande s est suivi par un argument qui prend la forme /ce qui est à remplacer (caractère, chaîne ou ERb)/chaîne de remplacement/g. Le caractère g force le remplacement de toutes les occurrences. Sans elle, uniquement la première occurrence serait remplacée. Dans le cas de l'exemple, on remplace donc les virgules par des espaces.

Présentation de La Commande awk

Le processeur de texte **awk** est un **filtre**. Une **action** awk est fournie sur la ligne de commande entourée de ' ou de “ :

```
awk [-F séparateur] 'critère {action}' [fichier1 ... fichiern]
```

Important : Le couple critère {action} s'appelle une clause.

Dans le cas de l'utilisation d'un **script** awk, la syntaxe de la commande devient :

```
awk [-F séparateur] -f script [fichier1 ... fichiern]
```

Découpage en champs

awk sait identifier les champs de la ligne soit parce que ceux-ci sont séparés par un espace ou par une tabulation soit parce que la ligne de commande lui a identifié le séparateur grâce à l'option **-F**.

awk stocke les informations de la ligne dans des variables :

Variable	Description
$0	Contient toute la ligne

Variable	Description
$1, $2 ...	Contient le premier champ de la ligne, contient le deuxième champ de la ligne ...

Par exemple :

```
[root@centos7 ~]# ls -l | awk '{print $8 $3 $4}'

2015rootroot
19:09270users
2015rootroot
2015rootroot
```

Comme vous pouvez constater, awk a extrait du résultat de la commande **ls -l** les champs **nom de l'élément**, **le propriétaire** et le **groupe**.

Afin de le rendre un peu plus lisible, saisissez la commande suivante :

```
[root@centos7 ~]# ls -l | awk '{print $8 " " $3 " " $4}'
2015 root root
19:09 270 users
2015 root root
2015 root root
```

Critères

Les **critères** conditionnent l'exécution d'une **action** dans une **clause**.

Plusieurs types de critères sont possibles. Les plus utilisées sont les suivantes :

Une expression régulière valide pour la ligne

- Format:
- /expression régulière/ {instruction}

- Exemple:
- /ERe/ {print $0}

Une expression régulière valide pour un champ

- Format:
- $n ~/expression régulière/ {instruction}
- $n!~/expression régulière/ {instruction}
- Exemple:
- $1 ~/ERe/ {print $0}
- $1!~/ERe/ {print $0}

awk sélectionne des lignes en utilisant un opérateur de correspondance ou de non-correspondance :

Opérateur	Condition
~	Correspondance
!~	Non-correspondance

Une comparaison

- Format:
- $n opérateur critère de comparaison {action}
- Exemple:
- $1 > 20 {print $0}

Les opérateurs sont :

Opérateur	Condition
<	Inférieur
<=	Inférieur ou égal
==	Egal
!=	Différent

Opérateur	Condition
>	Supérieur
>=	Supérieur ou égal

Un opérateur logique

- Format:
- test1 opérateur logique test2 {action}
- Exemple:
- $1 ~/ERe/ && $2 > 20 {print $0}

Les opérateurs sont :

Opérateur logique	Condition
||	OU
&&	ET
!	NON

Une variable interne

- Format:
- expression1, expression2 {action}
- Exemple:
- NR==7, NR==10 {print $0}

Les variables sont :

Variable	Description
NR	Nombre total de lignes
NF	Nombre total de champs
FILENAME	Le nom du fichier en entrée
FS	Le séparateur de champs en entrée. Par défaut un **espace** ou une **tabulation**

Variable	Description
RS	Le séparateur de lignes en entrée. Par défaut une **nouvelle ligne**
OFS	Le séparateur de champs en sortie. Par défaut un **espace**
ORS	Le séparateur de lignes en sortie. Par défaut une **nouvelle ligne**
OFMT	Le format numérique. Par défaut ”%.6g“

Scripts awk

Quand un programme awk comporte plusieurs **clauses** composées de **critères** et d'**actions**, il convient de d'écrire un **script awk**. Ce script comporte trois sections :

- La section **BEGIN**
  - Cette section est exécutée avant la lecture du script
- La section **principale**
  - Cette section contient les clauses
- La section **END**
  - Cette sectione est exécutée une fois à la fin du script

Par exemple :

```
[root@centos7 ~]# cd /tmp
[root@centos7 tmp]# cat > scriptawk
BEGIN {
  print "Liste des systèmes de fichiers montés"}
{print $0}
END {
  print "====================================="}
[^D]
```

Important : Dans l'exemple ci-dessus, la ligne [^D] indique que vous devez appuyer simultanément sur les touches CTRL et D.

Ensuite saisissez la commande suivante :

```
[root@centos7 tmp]# awk -f scriptawk /etc/fstab
Liste des systèmes de fichiers montés

#
# /etc/fstab
# Created by anaconda on Sun Mar  8 12:38:10 2015
#
# Accessible filesystems, by reference, are maintained under '/dev/disk'
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info
#
UUID=b35de665-5ec8-4226-a533-58a1b567ac91 /                       xfs     defaults        1 1
UUID=e8d3bd48-1386-411c-9675-41c3f8f1a309 /boot                   xfs     defaults        1 2
UUID=11a4d11d-81e4-46a7-82e0-7796cd597dc9 swap                    swap    defaults        0 0
====================================
```

Important : Notez l'utilisation de l'option -f qui applique le script awk au fichier donné en argument.

La Fonction printf

La fonction intégrée **printf** permet de formater des affichages. Elle a la syntaxe suivante :

```
printf ("chaine",expression1,expression2,...,expressionn)
```

chaine contient autant de formats qu'il y a d'expressions.

Les formats de printf sont, par exemple :

Format	Description
%30s	Affichage d'une chaîne (s=string) sur 30 positions avec cadrage à droite

Format	Description
%-30s	Affichage d'une chaîne (s=string) sur 30 positions avec cadrage à gauche
%4d	Affichage d'un entier sur 4 positions avec cadrage à droite
%-4d	Affichage d'un entier sur 4 positions avec cadrage à gauche

Structures de Contrôle

awk peut utiliser des structures de contrôle.

if

La syntaxe de la commande if est la suivante :

```
if condition {

    commande
    commande
    ...
}

else {

    commande
    commande
    ...
}
```

ou dans le cas d'une seule commande :

```
if condition
```

```
    commande

else

    commande
```

for

La syntaxe de la structure de contrôle **for** est la suivante :

```
for variable in liste_variables {

    commande
    commande
    ...

}
```

ou dans le cas d'une seule commande :

```
for variable in liste_variables

    commande
```

ou dans le cas d'un tableau :

```
for clef dans tableau {

    print clef , tableau[clef]

}
```

while

La syntaxe de la structure de contrôle **while** est la suivante :

```
while condition {

     commande
     commande
     ...

}
```

do-while

La syntaxe de la structure de contrôle **do-while** est la suivante :

```
do {

     commande
     commande
     ...

} while condition
```

Options de la commande

A faire : Utilisez l'option **–help** de la commande **awk** pour visualiser les options de la commande.

LAB #14 - Utiliser la Commande awk

Pour illustrer l'utilisation des tableaux, téléchargez d'abord le fichier **sales.txt** en cliquant sur le titre du fichier exemple ci-dessous :

sales.txt

```
# FenestrOs.com
# Annual sales by French department
# 83
Desktops§100
Portables§50
Servers§21
Ipads§4

# 06
Desktops§99
Portables§60
Servers§8
Ipads§16

# 13
Desktops§130
Portables§65
Servers§12
Ipads§56
```

Ce fichier contient des statistiques de vente par type de PC et par département.

Téléchargez maintenant le script awk **sales.awk** en cliquant sur le titre du fichier exemple ci-dessous :

sales.awk

```
# BEGIN
BEGIN {
    FS="§"
}
# TABLE
$1 !~ /^#/ && $1 !~ /^$/ {
    sales[$1]+=$2
}
# END
END {
    for (pc in sales)
    printf("PC Type :  %s \t Sales (06+13+83) : %10d\n",pc,sales[pc]);
}
```

Ce script comporte 13 lignes et a pour but de calculer le nombre total de PC vendus dans les trois départements cités dans le fichier **sales.txt** :

```
 1  # BEGIN
 2  BEGIN {
 3      FS="§"
 4  }
 5  # TABLE
 6  $1 !~ /^#/ && $1 !~ /^$/ {
 7      sales[$1]+=$2
 8  }
 9  # END
10  END {
11      for (pc in sales)
12      printf("PC Type :  %s \t Sales (06+13+83) : %10d\n",pc,sales[pc]);
13  }
```

Dans ce script vous noterez :

- La ligne **3**,

    - Cette ligne se trouve dans la section **BEGIN**. Elle spécifie le séparateur de champs.
- La ligne **6**,
    - Cette ligne évite le traitement de toute ligne commençant par le caractère **#** ainsi que toute ligne vide.
- La ligne **7**,
    - Ce tableau a pour clef la valeur de **$1**, c'est-à-dire, les noms des différents types de PC. Le valeurs du tableau sont le nombre de PC vendus, ici représenté par **$2**. Les caractères **+=** indique qu'à chaque traitement de ligne, le nombre de PC vendus sur la ligne doit être rajouté à la valeur déjà présente dans le tableau.
- La ligne **11**,
    - Cette ligne démarre une boucle **for**.7
- La ligne **12**,
    - Cette ligne utilise **printf** afin d'imprimer à l'écran les valeurs calculées et stockées dans le tableau.

Appliquez maintenant votre script awk au fichier **sales.txt** :

```
[root@centos7 ~]# awk -f /home/trainee/Downloads/sales.awk /home/trainee/Downloads/sales.txt
PC Type :  Portables     Sales (06+13+83) :        175
PC Type :  Ipads         Sales (06+13+83) :         76
PC Type :  Desktops      Sales (06+13+83) :        329
PC Type :  Servers       Sales (06+13+83) :         41
```

LAB #15 -Autres Commandes Utiles

15.1 - La Commande expand

La commande **expand** convertit des tabulations dans un fichier en espaces et envoie le résultat à la sortie standard. Sans fichier en argument ou avec le caractère **-**, la commande prend son entrée de l'entrée standard.

Téléchargez le fichier suivant :

expand.txt

```
un  deux    trois    quatre    cinq
```

```
un  deux    trois    quatre    cinq
un  deux    trois    quatre    cinq
un  deux    trois    quatre    cinq
un  deux    trois    quatre    cinq
un  deux    trois    quatre    cinq
un  deux    trois    quatre    cinq
un  deux    trois    quatre    cinq
un  deux    trois    quatre    cinq
un  deux    trois    quatre    cinq
```

Déplacez le fichier vers /root :

```
[root@centos7 ~]# mv /home/trainee/Downloads/expand.txt /root/expand
```

Visualisez le fichier avec la commande cat :

```
[root@centos7 ~]# cat expand
un  deux    trois    quatre    cinq
un  deux    trois    quatre    cinq
un  deux    trois    quatre    cinq
un  deux    trois    quatre    cinq
un  deux    trois    quatre    cinq
un  deux    trois    quatre    cinq
un  deux    trois    quatre    cinq
un  deux    trois    quatre    cinq
un  deux    trois    quatre    cinq
un  deux    trois    quatre    cinq
```

Utilisez les option **-vet** de la commande cat pour visualiser les caractères invisibles :

```
[root@centos7 ~]# cat -vet expand
un^Ideux^Itrois^Iquatre^Icinq$
un^Ideux^Itrois^Iquatre^Icinq$
```

```
un^Ideux^Itrois^Iquatre^Icinq$
un^Ideux^Itrois^Iquatre^Icinq$
un^Ideux^Itrois^Iquatre^Icinq$
un^Ideux^Itrois^Iquatre^Icinq$
un^Ideux^Itrois^Iquatre^Icinq$
un^Ideux^Itrois^Iquatre^Icinq$
un^Ideux^Itrois^Iquatre^Icinq$
un^Ideux^Itrois^Iquatre^Icinq$
```

Important : Comme vous pouvez constater, les tabulations sont représentées par ^I et les fins de lignes par $.

Utilisez maintenant la commande **expand** pour convertir les tabulations en espaces en envoyant le résultat dans le fichier **expand1** :

```
[root@centos7 ~]# expand expand > expand1
```

Visualisez le fichier avec la commande cat et les options **-vet** :

```
[root@centos7 ~]# cat -vet expand1
un      deux    trois   quatre  cinq$
un      deux    trois   quatre  cinq$
un      deux    trois   quatre  cinq$
un      deux    trois   quatre  cinq$
un      deux    trois   quatre  cinq$
un      deux    trois   quatre  cinq$
un      deux    trois   quatre  cinq$
un      deux    trois   quatre  cinq$
un      deux    trois   quatre  cinq$
un      deux    trois   quatre  cinq$
```

Important : Comme vous pouvez constater, les tabulations ont été converties en espaces.

Options de la commande

A faire : Utilisez l'option **–help** de la commande **expand** pour visualiser les options de la commande.

15.2 - La Commande unexpand

La commande **unexpand** convertit des espaces dans un fichier en tabulations et envoie le résultat à la sortie standard. Sans fichier en argument ou avec le caractère **-**, la commande prend son entrée de l'entrée standard.

Utilisez la commande **unexpand** sur le fichier **expand1** et envoyez le résultat dans le fichier **expand2** :

```
[root@centos7 ~]# cat -vet expand1
un      deux    trois   quatre  cinq$
un      deux    trois   quatre  cinq$
un      deux    trois   quatre  cinq$
un      deux    trois   quatre  cinq$
un      deux    trois   quatre  cinq$
un      deux    trois   quatre  cinq$
un      deux    trois   quatre  cinq$
un      deux    trois   quatre  cinq$
un      deux    trois   quatre  cinq$
un      deux    trois   quatre  cinq$
un      deux    trois   quatre  cinq$
[root@centos7 ~]# unexpand -a expand1 > expand2
[root@centos7 ~]# cat -vet expand2
un^Ideux^Itrois^Iquatre^Icinq$
un^Ideux^Itrois^Iquatre^Icinq$
un^Ideux^Itrois^Iquatre^Icinq$
un^Ideux^Itrois^Iquatre^Icinq$
un^Ideux^Itrois^Iquatre^Icinq$
un^Ideux^Itrois^Iquatre^Icinq$
```

```
un^Ideux^Itrois^Iquatre^Icinq$
un^Ideux^Itrois^Iquatre^Icinq$
un^Ideux^Itrois^Iquatre^Icinq$
un^Ideux^Itrois^Iquatre^Icinq$
```

Important : Notez que les espaces ont été remplacés par des tabulations.

Options de la commande

A faire : Utilisez l'option **–help** de la commande **unexpand** pour visualiser les options de la commande.

15.3 - La Commande cut

Chaque ligne est divisée en colonnes. Dans une ligne le premier caractère est dans la colonne numéro **un**, le deuxième dans la colonne deux et ainsi de suite. Dans une ligne il peut y avoir des champs séparés par des tabulations.

La commande **cut** permet de sélectionner des colonnes et des champs dans un fichier. La commande permet aussi d'utiliser une critère de séparation de champs autre que la tabulation en spécifiant cette critère en utilisant l'option **-d**.

Par exemple, pour sélectionner les 7 premières colonnes du fichier **/etc/passwd** la commande est :

```
[root@centos7 ~]# cut -c1-7 /etc/passwd
root:x:
bin:x:1
daemon:
adm:x:3
```

```
lp:x:4:
sync:x:
shutdow
halt:x:
mail:x:
operato
games:x
ftp:x:1
nobody:
dbus:x:
polkitd
unbound
colord:
usbmuxd
avahi:x
avahi-a
saslaut
qemu:x:
libstor
rpc:x:3
rpcuser
nfsnobo
rtkit:x
radvd:x
ntp:x:3
chrony:
abrt:x:
pulse:x
gdm:x:4
gnome-i
postfix
sshd:x:
tcpdump
trainee
```

```
vboxadd
tss:x:5
```

Pour sélectionner les colonnes 1 à 5, les colonnes 10 à 15 et les colonnes 30 et après, il convient d’utiliser la commande suivante :

```
[root@centos7 ~]# cut -c1-5,10-15,30- /etc/passwd
root:0:rootsh
bin:x:bin:/gin
daemo2:2:dain/nologin
adm:x:adm:/nologin
lp:x:lp:/vabin/nologin
sync:0:syncnc
shutdx:6:0::/sbin/shutdown
halt:0:haltalt
mail:12:maiail:/sbin/nologin
operax:11:0t:/sbin/nologin
games2:100:es:/sbin/nologin
ftp:x50:FTP:/sbin/nologin
nobod99:99:/nologin
dbus::81:Syus:/:/sbin/nologin
polki:999:9lkitd:/:/sbin/nologin
unbou:998:9 resolver:/etc/unbound:/sbin/nologin
color997:99ord:/var/lib/colord:/sbin/nologin
usbmu:113:1r:/:/sbin/nologin
avahi0:70:ASD Stack:/var/run/avahi-daemon:/sbin/nologin
avahioipd:x IPv4LL Stack:/var/lib/avahi-autoipd:/sbin/nologin
saslax:996:user":/run/saslauthd:/sbin/nologin
qemu:7:107:bin/nologin
libstemgmt:on account for libstoragemgmt:/var/run/lsm:/sbin/nologin
rpc:x32:Rpcar/lib/rpcbind:/sbin/nologin
rpcus:29:29ser:/var/lib/nfs:/sbin/nologin
nfsno:x:655mous NFS User:/var/lib/nfs:/sbin/nologin
rtkit72:172proc:/sbin/nologin
radvd5:75:rbin/nologin
```

```
ntp:x38::/eologin
chron994:99rony:/sbin/nologin
abrt:3:173:in/nologin
pulse71:171stem Daemon:/var/run/pulse:/sbin/nologin
gdm:x42::/vin/nologin
gnometial-s::/run/gnome-initial-setup/:/sbin/nologin
postf:89:89ostfix:/sbin/nologin
sshd::74:Prted SSH:/var/empty/sshd:/sbin/nologin
tcpdu:72:72gin
train:1000:home/trainee:/bin/bash
vboxa:992:1xadd:/bin/false
tss:x59:Acche trousers package to sandbox the tcsd daemon:/dev/null:/sbin/nologin
```

Pour sélectionner les champs 2, 4 et 6 du fichier, il convient d'utiliser la commande suivante :

```
[root@centos7 ~]# cut -d: -f2,4,6 /etc/passwd
x:0:/root
x:1:/bin
x:2:/sbin
x:4:/var/adm
x:7:/var/spool/lpd
x:0:/sbin
x:0:/sbin
x:0:/sbin
x:12:/var/spool/mail
x:0:/root
x:100:/usr/games
x:50:/var/ftp
x:99:/
x:81:/
x:998:/
x:997:/etc/unbound
x:996:/var/lib/colord
x:113:/
```

```
x:70:/var/run/avahi-daemon
x:170:/var/lib/avahi-autoipd
x:76:/run/saslauthd
x:107:/
x:994:/var/run/lsm
x:32:/var/lib/rpcbind
x:29:/var/lib/nfs
x:65534:/var/lib/nfs
x:172:/proc
x:75:/
x:38:/etc/ntp
x:993:/var/lib/chrony
x:173:/etc/abrt
x:171:/var/run/pulse
x:42:/var/lib/gdm
x:991:/run/gnome-initial-setup/
x:89:/var/spool/postfix
x:74:/var/empty/sshd
x:72:/
x:1000:/home/trainee
x:1:/var/run/vboxadd
x:59:/dev/null
```

Options de la commande

A faire : Utilisez l'option **–help** de la commande **cut** pour visualiser les options de la commande.

15.4 - La Commande uniq

La commande suivante permet d'extraire du fichier /etc/passwd les GID utilisés en tant que groupes principaux des utilisateurs :

```
[root@centos7 ~]# cut -d: -f4 /etc/passwd | sort -n | uniq
0
1
2
4
7
12
29
32
38
42
50
59
70
72
74
75
76
81
89
99
100
107
113
170
171
172
173
991
993
994
996
```

```
997
998
1000
65534
```

Important : Notez l'utilisation de la commande uniq qui permet de supprimer les doublons dans la sortie triée.

Options de la commande

A faire : Utilisez l'option **–help** de la commande **uniq** pour visualiser les options de la commande.

15.5 - La Commande tr

La commande **tr** permet de substituer des caractères pour d'autres. Cette commande n'accepte que des données en provenance de son entrée standard et non en provenance d'un fichier.

```
[root@centos7 ~]# cat /etc/passwd | tr "[a-z]" "[A-Z]"
ROOT:X:0:0:ROOT:/ROOT:/BIN/BASH
BIN:X:1:1:BIN:/BIN:/SBIN/NOLOGIN
DAEMON:X:2:2:DAEMON:/SBIN:/SBIN/NOLOGIN
ADM:X:3:4:ADM:/VAR/ADM:/SBIN/NOLOGIN
LP:X:4:7:LP:/VAR/SPOOL/LPD:/SBIN/NOLOGIN
SYNC:X:5:0:SYNC:/SBIN:/BIN/SYNC
SHUTDOWN:X:6:0:SHUTDOWN:/SBIN:/SBIN/SHUTDOWN
HALT:X:7:0:HALT:/SBIN:/SBIN/HALT
MAIL:X:8:12:MAIL:/VAR/SPOOL/MAIL:/SBIN/NOLOGIN
```

```
OPERATOR:X:11:0:OPERATOR:/ROOT:/SBIN/NOLOGIN
GAMES:X:12:100:GAMES:/USR/GAMES:/SBIN/NOLOGIN
FTP:X:14:50:FTP USER:/VAR/FTP:/SBIN/NOLOGIN
NOBODY:X:99:99:NOBODY:/:/SBIN/NOLOGIN
DBUS:X:81:81:SYSTEM MESSAGE BUS:/:/SBIN/NOLOGIN
POLKITD:X:999:998:USER FOR POLKITD:/:/SBIN/NOLOGIN
UNBOUND:X:998:997:UNBOUND DNS RESOLVER:/ETC/UNBOUND:/SBIN/NOLOGIN
COLORD:X:997:996:USER FOR COLORD:/VAR/LIB/COLORD:/SBIN/NOLOGIN
USBMUXD:X:113:113:USBMUXD USER:/:/SBIN/NOLOGIN
AVAHI:X:70:70:AVAHI MDNS/DNS-SD STACK:/VAR/RUN/AVAHI-DAEMON:/SBIN/NOLOGIN
AVAHI-AUTOIPD:X:170:170:AVAHI IPV4LL STACK:/VAR/LIB/AVAHI-AUTOIPD:/SBIN/NOLOGIN
SASLAUTH:X:996:76:"SASLAUTHD USER":/RUN/SASLAUTHD:/SBIN/NOLOGIN
QEMU:X:107:107:QEMU USER:/:/SBIN/NOLOGIN
LIBSTORAGEMGMT:X:995:994:DAEMON ACCOUNT FOR LIBSTORAGEMGMT:/VAR/RUN/LSM:/SBIN/NOLOGIN
RPC:X:32:32:RPCBIND DAEMON:/VAR/LIB/RPCBIND:/SBIN/NOLOGIN
RPCUSER:X:29:29:RPC SERVICE USER:/VAR/LIB/NFS:/SBIN/NOLOGIN
NFSNOBODY:X:65534:65534:ANONYMOUS NFS USER:/VAR/LIB/NFS:/SBIN/NOLOGIN
RTKIT:X:172:172:REALTIMEKIT:/PROC:/SBIN/NOLOGIN
RADVD:X:75:75:RADVD USER:/:/SBIN/NOLOGIN
NTP:X:38:38::/ETC/NTP:/SBIN/NOLOGIN
CHRONY:X:994:993::/VAR/LIB/CHRONY:/SBIN/NOLOGIN
ABRT:X:173:173::/ETC/ABRT:/SBIN/NOLOGIN
PULSE:X:171:171:PULSEAUDIO SYSTEM DAEMON:/VAR/RUN/PULSE:/SBIN/NOLOGIN
GDM:X:42:42::/VAR/LIB/GDM:/SBIN/NOLOGIN
GNOME-INITIAL-SETUP:X:993:991::/RUN/GNOME-INITIAL-SETUP/:/SBIN/NOLOGIN
POSTFIX:X:89:89::/VAR/SPOOL/POSTFIX:/SBIN/NOLOGIN
SSHD:X:74:74:PRIVILEGE-SEPARATED SSH:/VAR/EMPTY/SSHD:/SBIN/NOLOGIN
TCPDUMP:X:72:72::/:/SBIN/NOLOGIN
TRAINEE:X:1000:1000:TRAINEE:/HOME/TRAINEE:/BIN/BASH
VBOXADD:X:992:1::/VAR/RUN/VBOXADD:/BIN/FALSE
TSS:X:59:59:ACCOUNT USED BY THE TROUSERS PACKAGE TO SANDBOX THE TCSD DAEMON:/DEV/NULL:/SBIN/NOLOGIN
```

Options de la commande

A faire : Utilisez l'option **–help** de la commande **tr** pour visualiser les options de la commande.

15.6 - La Commande paste

La commande **paste** concatène les lignes de n fichiers. Par exemple :

```
[root@centos7 ~]# paste -d: /etc/passwd /etc/shadow
root:x:0:0:root:/root:/bin/bash:root:$6$r4px/s9L2uwGSFnI$NkK5mzNF.CMAFFqMcO.i.tnrMZQDkriDLYwICsimsPaDWKFwUHS3NhDw
ZY5e7P3glIu.gTBtaOE.SOOW/D.AU/:16502:0:99999:7:::
bin:x:1:1:bin:/bin:/sbin/nologin:bin:*:16231:0:99999:7:::
daemon:x:2:2:daemon:/sbin:/sbin/nologin:daemon:*:16231:0:99999:7:::
adm:x:3:4:adm:/var/adm:/sbin/nologin:adm:*:16231:0:99999:7:::
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin:lp:*:16231:0:99999:7:::
sync:x:5:0:sync:/sbin:/bin/sync:sync:*:16231:0:99999:7:::
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown:shutdown:*:16231:0:99999:7:::
halt:x:7:0:halt:/sbin:/sbin/halt:halt:*:16231:0:99999:7:::
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin:mail:*:16231:0:99999:7:::
operator:x:11:0:operator:/root:/sbin/nologin:operator:*:16231:0:99999:7:::
games:x:12:100:games:/usr/games:/sbin/nologin:games:*:16231:0:99999:7:::
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin:ftp:*:16231:0:99999:7:::
nobody:x:99:99:Nobody:/:/sbin/nologin:nobody:*:16231:0:99999:7:::
dbus:x:81:81:System message bus:/:/sbin/nologin:dbus:!!:16502::::::
polkitd:x:999:998:User for polkitd:/:/sbin/nologin:polkitd:!!:16502::::::
unbound:x:998:997:Unbound DNS resolver:/etc/unbound:/sbin/nologin:unbound:!!:16502::::::
colord:x:997:996:User for colord:/var/lib/colord:/sbin/nologin:colord:!!:16502::::::
usbmuxd:x:113:113:usbmuxd user:/:/sbin/nologin:usbmuxd:!!:16502::::::
avahi:x:70:70:Avahi mDNS/DNS-SD Stack:/var/run/avahi-daemon:/sbin/nologin:avahi:!!:16502::::::
avahi-autoipd:x:170:170:Avahi IPv4LL Stack:/var/lib/avahi-autoipd:/sbin/nologin:avahi-autoipd:!!:16502::::::
```

```
saslauth:x:996:76:"Saslauthd user":/run/saslauthd:/sbin/nologin:saslauth:!!:16502::::::
qemu:x:107:107:qemu user:/:/sbin/nologin:qemu:!!:16502::::::
libstoragemgmt:x:995:994:daemon account for
libstoragemgmt:/var/run/lsm:/sbin/nologin:libstoragemgmt:!!:16502::::::
rpc:x:32:32:Rpcbind Daemon:/var/lib/rpcbind:/sbin/nologin:rpc:!!:16502:0:99999:7:::
rpcuser:x:29:29:RPC Service User:/var/lib/nfs:/sbin/nologin:rpcuser:!!:16502::::::
nfsnobody:x:65534:65534:Anonymous NFS User:/var/lib/nfs:/sbin/nologin:nfsnobody:!!:16502::::::
rtkit:x:172:172:RealtimeKit:/proc:/sbin/nologin:rtkit:!!:16502::::::
radvd:x:75:75:radvd user:/:/sbin/nologin:radvd:!!:16502::::::
ntp:x:38:38::/etc/ntp:/sbin/nologin:ntp:!!:16502::::::
chrony:x:994:993::/var/lib/chrony:/sbin/nologin:chrony:!!:16502::::::
abrt:x:173:173::/etc/abrt:/sbin/nologin:abrt:!!:16502::::::
pulse:x:171:171:PulseAudio System Daemon:/var/run/pulse:/sbin/nologin:pulse:!!:16502::::::
gdm:x:42:42::/var/lib/gdm:/sbin/nologin:gdm:!!:16502::::::
gnome-initial-setup:x:993:991::/run/gnome-initial-setup/:/sbin/nologin:gnome-initial-setup:!!:16502::::::
postfix:x:89:89::/var/spool/postfix:/sbin/nologin:postfix:!!:16502::::::
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin:sshd:!!:16502::::::
tcpdump:x:72:72::/:/sbin/nologin:tcpdump:!!:16502::::::
trainee:x:1000:1000:trainee:/home/trainee:/bin/bash:trainee:$6$tMd44tmmFiitAS7.$sJSua3jhyKm2k0mIifYuTpUO0d6q6/gS3
PDyuxbHadHVYLsoVslZ3Pn8m5X93rr64oj.KK8OL6J.gvhxbQBrZ1:16502:0:99999:7:::
vboxadd:x:992:1::/var/run/vboxadd:/bin/false:vboxadd:!!:16691::::::
tss:x:59:59:Account used by the trousers package to sandbox the tcsd
daemon:/dev/null:/sbin/nologin:tss:!!:16723::::::
```

Options de la commande

A faire : Utilisez l'option **–help** de la commande **paste** pour visualiser les options de la commande.

15.7 - La Commande split

La commande **split** est utilisée pour découper de grands fichiers en petit morceaux d'une taille fixe ou d'un nombre de lignes fixe.

Créez d'abord un fichier d'une taille de 250Mo :

```
[root@centos7 ~]# dd if=/dev/zero of=/file bs=1024k count=250
250+0 records in
250+0 records out
262144000 bytes (262 MB) copied, 0.215447 s, 1.2 GB/s
```

Utilisez maintenant la commande **split** pour diviser ce fichier en morceaux de 50 Mo :

```
[root@centos7 ~]# split -b 50m /file filepart
[root@centos7 ~]# ls -l | grep filepart
-rw-r--r--.  1 root root  52428800 Oct 19 15:31 filepartaa
-rw-r--r--.  1 root root  52428800 Oct 19 15:31 filepartab
-rw-r--r--.  1 root root  52428800 Oct 19 15:31 filepartac
-rw-r--r--.  1 root root  52428800 Oct 19 15:31 filepartad
-rw-r--r--.  1 root root  52428800 Oct 19 15:31 filepartae
```

Important : Notez que cinq morceaux ont été créés dans le répertoire courant. Si aucune taille n'est spécifiée, split devise le fichier en morceaux de 1 000 lignes par défaut.

Reconstruisez simplement le fichier avec la commande cat :

```
[root@centos7 ~]# cat fileparta* > newfile
[root@centos7 ~]# ls -l | grep newf
-rw-r--r--.  1 root root  262144000 Oct 19 15:31 newfile
```

Options de la commande

A faire : Utilisez l'option **–help** de la commande **split** pour visualiser les options de la commande.

15.8 - La Commande diff

La commande **diff** indique les modifications à apporter à deux fichiers pour que ceux-ci soient identique.

Pour commencer, copiez le fichier **/etc/passwd** vers le répertoire **/root** :

```
[root@centos7 ~]# cp /etc/passwd /root
```

Modifiez ensuite le fichier la ligne **trainee** du fichier /root/passwd ainsi :

```
...
trainee10:x:1000:1000:trainee:/home/trainee:/bin/bash
...
```

Supprimez la ligne **tcpdump** dans le fichier /root/passwd et ajoutez en fin de fichier la ligne suivante :

```
...
Linux est super!
```

Comparez maintenant les deux fichiers :

```
[root@centos7 ~]# diff /etc/passwd /root/passwd
38,39c38
< trainee:x:1000:1000:trainee:/home/trainee:/bin/bash
< vboxadd:x:992:1::/var/run/vboxadd:/bin/false
```

```
---
> trainee10:x:1000:1000:trainee:/home/trainee:/bin/bash
40a40
> Linux est super!
```

Dans cette sortie on constate le caractère **<** et le caractère **>**. Le premier indique le premier fichier qui a suivi la commande **diff** tandis que le deuxième indique le deuxième fichier.

Le message **38,39c38** indique qu'il faut changer la ligne 39 dans /etc/passwd afin que celle-ci corresponde à la ligne 39 dans /root/passwd.

Le message **40a40** indique qu'à la ligne 40 dans /etc/passwd il faut ajouter la ligne 40 de /root/passwd.

Options de la commande

A faire : Utilisez l'option **–help** de la commande **diff** pour visualiser les options de la commande.

15.9 - La Commande cmp

La commande **cmp** compare les fichiers caractère par caractère. Par défaut la commande s'arrête à la première différence rencontrée :

```
[root@centos7 ~]# cmp /root/passwd /etc/passwd
/root/passwd /etc/passwd differ: byte 1873, line 38
```

L'option **-l** de la commande indique toutes les différences en trois colonnes :

```
[root@centos7 ~]# cmp -l /root/passwd /etc/passwd | more
cmp: EOF on /root/passwd
1873  61  72
1874  60 170
```

```
1876 170  61
1877  72  60
1878  61  60
1880  60  72
1881  60  61
1882  72  60
1883  61  60
1885  60  72
1886  60 164
1887  72 162
1888 164 141
1889 162 151
1890 141 156
1891 151 145
1892 156 145
1893 145  72
1894 145  57
1895  72 150
1896  57 157
1897 150 155
1898 157 145
1899 155  57
1900 145 164
1901  57 162
1902 164 141
1903 162 151
--More--
```

La première colonne représente le numéro de caractère, la deuxième la valeur octale ASCII du caractère dans le fichier /root/passwd et la troisième la valeur octale ASCII du caractère dans le fichier /etc/passwd.

Options de la commande

A faire : Utilisez l'option **–help** de la commande **cmp** pour visualiser les options de la commande.

15.10 - La commande patch

La commande **patch** est utilisée pour appliquer des modifications à un fichier à partir d'un fichier patch qui contient les différences entre le contenu de l'ancienne version du fichier et la nouvelle version.

La commande patch n'est pas installée par défaut sous RHEL/CentOS 7 :

```
[root@centos7 ~]# yum install patch
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
 * base: mirror.ibcp.fr
 * extras: mirrors.atosworldline.com
 * updates: mirror.ibcp.fr
Resolving Dependencies
--> Running transaction check
---> Package patch.x86_64 0:2.7.1-8.el7 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=================================================================================================
=====================================
 Package                            Arch                                Version
Repository                         Size
=================================================================================================
=====================================
Installing:
 patch                              x86_64                              2.7.1-8.el7
```

```
base                              110 k

Transaction Summary
================================================================================================================
=====================================
Install  1 Package

Total download size: 110 k
Installed size: 210 k
Is this ok [y/d/N]: y
Downloading packages:
patch-2.7.1-8.el7.x86_64.rpm
| 110 kB  00:00:00
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : patch-2.7.1-8.el7.x86_64
1/1
  Verifying  : patch-2.7.1-8.el7.x86_64
1/1

Installed:
  patch.x86_64 0:2.7.1-8.el7

Complete!
```

Rappelez-vous maintenant des modifications apportées au fichier /tmp/greptest :

```
[root@centos7 ~]# cat /tmp/greptest
# Commentaire du début
^ voici une ligne pour la recherche fgrep
fenestrOS
fenestros
```

```
# Un autre commentaire
555-5555
f

.fenestros

.fe

£
# Commentaire de la fin
[root@centos ~]# cat /tmp/greptest1
fenestrOS
fenestros
555-5555
f
.fenestros
.fe
£
```

Afin de créer un fichier de patch, il convient d'utiliser la commande **diff** avec l'option **-u**

```
[root@centos7 ~]# cd /tmp
[root@centos7 tmp]# diff -u greptest greptest1 > greptest.patch
```

L'examen du fichier de patch démontre les modifications à apporter au fichier **greptest** :

```
[root@centos7 tmp]# cat greptest.patch
--- greptest    2012-05-13 15:49:23.143130290 +0200
+++ greptest1   2012-05-13 15:26:12.498027492 +0200
@@ -1,14 +1,7 @@
-# Commentaire du début
-^ voici une ligne pour la recherche fgrep
 fenestrOS
 fenestros
```

```
-# Un autre commentaire
 555-5555
 f
-
 .fenestros
-
 .fe
-
 £
-# Commentaire de la fin
```

Procédez maintenant à l'application du fichier patch :

```
[root@centos7 tmp]# patch < greptest.patch
patching file greptest
```

Contrôlez maintenant le contenu du fichier **greptest** :

```
[root@centos7 tmp]# cat greptest
fenestrOS
fenestros
555-5555
f
.fenestros
.fe
£
```

Options de la commande

A faire : Utilisez l'option **–help** de la commande **patch** pour visualiser les options de la commande.

15.11 - La commande strings

La commande **strings** est utilisée pour trouver toutes les chaînes de caractères qui peuvent être imprimés dans un ou plusieurs fichiers objets ou exécutables passés en argument. Un fichier objet est un fichier intermédiaire intervenant dans le processus de compilation.

Sous Linux et Unix, le format d'un fichier objet est le format **ELF**, (*Executable and Linkable Format*). Ce format est aussi utilisé pour :

- les exécutables,
- les bibliothèques partagés,
- les core dumps.

Sans option, la commande **strings** trouve toutes les chaînes d'une longueur de 4 caractères ou plus suivies par un caractère non-imprimable :

```
[root@centos7 ~]# strings /usr/bin/passwd | more
/lib64/ld-linux-x86-64.so.2
{1~`
libuser.so.1
g_value_get_int64
is_selinux_enabled
_ITM_deregisterTMCloneTable
g_free
g_value_array_get_nth
__gmon_start__
g_value_get_string
g_type_check_value_holds
g_value_get_long
_Jv_RegisterClasses
freecon
_ITM_registerTMCloneTable
lu_ent_set_string
lu_ent_get_first_value_strdup
lu_error_free
lu_user_lock
_fini
```

```
_init
lu_strerror
lu_ent_free
lu_ent_new
lu_user_modify
lu_prompt_console
lu_start
lu_user_lookup_name
--More--
```

L'option **-t** de la commande retourne, en plus des chaînes concernées, la position de décalage pour chaque ligne sur laquelle une ou plusieurs chaînes se trouvent :

```
[root@centos7 ~]# strings -t d /usr/bin/passwd | more
    568 /lib64/ld-linux-x86-64.so.2
    648 {1~`
   2833 libuser.so.1
   2846 g_value_get_int64
   2864 is_selinux_enabled
   2883 _ITM_deregisterTMCloneTable
   2911 g_free
   2918 g_value_array_get_nth
   2940 __gmon_start__
   2955 g_value_get_string
   2974 g_type_check_value_holds
   2999 g_value_get_long
   3016 _Jv_RegisterClasses
   3036 freecon
   3044 _ITM_registerTMCloneTable
   3070 lu_ent_set_string
   3088 lu_ent_get_first_value_strdup
   3118 lu_error_free
   3132 lu_user_lock
   3145 _fini
```

```
   3151 _init
   3157 lu_strerror
   3169 lu_ent_free
   3181 lu_ent_new
   3192 lu_user_modify
   3207 lu_prompt_console
   3225 lu_start
   3234 lu_user_lookup_name
--More--
```

L'option **-t** prend un de trois arguments qui indique le système de numérotation à utiliser :

Argument	Système de Numérotation
d	Décimal
o	Octal
x	Héxadécimal

L'option **-n** de la commande permet de modifier le nombre de caractères minimales dans les chaînes recherchées :

```
[root@centos7 ~]# strings -t d -n 15 /usr/bin/passwd | more
    568 /lib64/ld-linux-x86-64.so.2
   2846 g_value_get_int64
   2864 is_selinux_enabled
   2883 _ITM_deregisterTMCloneTable
   2918 g_value_array_get_nth
   2955 g_value_get_string
   2974 g_type_check_value_holds
   2999 g_value_get_long
   3016 _Jv_RegisterClasses
   3044 _ITM_registerTMCloneTable
   3070 lu_ent_set_string
   3088 lu_ent_get_first_value_strdup
   3207 lu_prompt_console
   3234 lu_user_lookup_name
```

```
   3261 lu_ent_set_long
   3303 lu_user_removepass
   3322 libgobject-2.0.so.0
   3342 libglib-2.0.so.0
   3372 poptSetOtherOptionHelp
   3464 poptHelpOptions
   3503 audit_log_acct_message
   3599 libpam_misc.so.0
   3640 audit_log_user_avc_message
   3667 audit_log_user_message
   3690 libselinux.so.1
   3706 selinux_set_callback
   3727 selinux_check_access
   3759 libpthread.so.0
--More--
```

Dans le cas de l'utilisation de la commande avec plus d'un fichier, l'option **-f** devient très utile. Par exemple, imaginons que vous souhaitez connaître les détails disponibles des Copyright des fichiers dans /bin :

```
[root@centos7 ~]# strings -f /bin/* | grep "(c)"
/bin/btrace: # Copyright (c) 2005 Silicon Graphics, Inc.
/bin/cdda-player: (c) 1997,98 Gerd Knorr <kraxel@goldbach.in-berlin.de>
/bin/cdda-player: (c) 2005, 2006 Rocky Bernstein <rocky@gnu.org>
/bin/cd-drive: Copyright (c) 2003-2005, 2007-2008, 2011-2013 R. Bernstein
/bin/cd-info: Copyright (c) 2003-2005, 2007-2008, 2011-2013 R. Bernstein
/bin/cd-read: Copyright (c) 2003-2005, 2007-2008, 2011-2013 R. Bernstein
/bin/chcat:         if len(c) > 0 and ( c[0] == "+" or c[0] == "-" ):
/bin/chcat:                 if len(c) > 0 and c[0] == "+":
/bin/chcat:                 if len(c) > 0 and c[0] == "-":
/bin/crash:        (c) a pointer to the first item pointed to by the start address.
/bin/crash:        (c) a list_head that is embedded within a data structure that is
/bin/crash:                 switch(c)
/bin/csh: @(#) Copyright (c) 1991 The Regents of the University of California.
/bin/diffpp: # Copyright (c) 1996-1998 Markku Rossi
```

```
/bin/drill: Copyright (c) 2004-2008 NLnet Labs.
/bin/festival_server: ##                        Copyright (c) 1996                          ##
/bin/festival_server_control: ##                        Copyright (c) 1996                          ##
/bin/gcm-calibrate: Copyright (c)
/bin/gprof: @(#) Copyright (c) 1983 Regents of the University of California.
/bin/hpijs: Copyright (c) 2001-2004, Hewlett-Packard Co.
/bin/ibus-setup: # Copyright (c) 2007-2010 Peng Huang <shawn.p.huang@gmail.com>
/bin/ibus-setup: # Copyright (c) 2007-2010 Red Hat, Inc.
/bin/ibus-table-createdb: # Copyright (c) 2008-2009 Yu Yuwei <acevery@gmail.com>
/bin/iso-info: Copyright (c) 2003-2005, 2007-2008, 2011-2013 R. Bernstein
/bin/iso-read: Copyright (c) 2003-2005, 2007-2008, 2011-2013 R. Bernstein
/bin/lsusb.py: # (c) Kurt Garloff <garloff@suse.de>, 2/2009, GPL v2 or v3.
/bin/mmc-tool: Copyright (c) 2003-2005, 2007-2008, 2011-2013 R. Bernstein
/bin/orca: __copyright__ = "Copyright (c) 2010-2012 The Orca Team" \
/bin/orca:                 "Copyright (c) 2012 Igalia, S.L."
/bin/pinentry: # Copyright (c) 2006 SUSE LINUX Products GmbH, Nuernberg, Germany.
/bin/pinentry: # Copyright (c) 2009 Fedora Project
/bin/ping: @(#) Copyright (c) 1989 The Regents of the University of California.
/bin/ping6: @(#) Copyright (c) 1989 The Regents of the University of California.
/bin/pod2usage: # Copyright (c) 1996-2000 by Bradford Appleton. All rights reserved.
/bin/qemu-img: qemu-img version 1.5.3, Copyright (c) 2004-2008 Fabrice Bellard
/bin/repoclosure: # seth vidal 2005 (c) etc etc
/bin/repodiff: # (c) 2007 Red Hat. Written by skvidal@fedoraproject.org
/bin/repo-graph: # (c) 2005 Panu Matilainen <pmatilai@laiskiainen.org>
/bin/repomanage: # (c) Copyright Seth Vidal 2004
/bin/repoquery: # (c) pmatilai@laiskiainen.org
/bin/repo-rss: # seth vidal 2005 (c) etc etc
/bin/repotrack: # (c) 2005 seth vidal skvidal at phy.duke.edu
/bin/rngtest: Copyright (c) 2004 by Henrique de Moraes Holschuh
/bin/saytime: ;;                        Copyright (c) 1996,1997                        ;;
/bin/sbcdec: Copyright (c) 2004-2010  Marcel Holtmann
/bin/sbcenc: Copyright (c) 2004-2010  Marcel Holtmann
/bin/screen: Copyright (c) 2010 Juergen Weigert, Sadrul Habib Chowdhury
/bin/screen: Copyright (c) 2008, 2009 Juergen Weigert, Michael Schroeder, Micah Cowan, Sadrul Habib Chowdhury
```

```
/bin/screen: Copyright (c) 1993-2002, 2003, 2005, 2006, 2007 Juergen Weigert, Michael Schroeder
/bin/screen: Copyright (c) 1987 Oliver Laumann
/bin/sliceprint: # Copyright (c) 1996-1999 Markku Rossi
/bin/soundstretch:     Copyright (c) Olli Parviainen
/bin/ssh-copy-id: # Copyright (c) 1999-2013 Philip Hands <phil@hands.com>
/bin/tcptraceroute: #   Copyright (c)  2007             Dmitry Butskoy
/bin/tcsh: @(#) Copyright (c) 1991 The Regents of the University of California.
/bin/text2wave: ;;                       Copyright (c) 1996,1997                        ;;
/bin/traceroute: Copyright (c) 2008  Dmitry Butskoy,   License: GPL v2 or any later
/bin/traceroute6: Copyright (c) 2008  Dmitry Butskoy,   License: GPL v2 or any later
/bin/verifytree: # copyright (c) 2008 Red Hat, Inc - written by Seth Vidal and Will Woods
/bin/wavpack:  Copyright (c) 1998 - 2009 Conifer Software.  All Rights Reserved.
/bin/wvdial: Copyright (c) 1997-2005 Net Integration Technologies, Inc.
/bin/wvdialconf: Copyright (c) 1997-2005 Net Integration Technologies, Inc.
/bin/wvgain:  Copyright (c) 2005 - 2009 Conifer Software.  All Rights Reserved.
/bin/wvunpack:  Copyright (c) 1998 - 2009 Conifer Software.  All Rights Reserved.
/bin/yum-debug-dump: ## (c) 2008 Red Hat. Written by skvidal@fedoraproject.org
/bin/yum-debug-restore: ## (c) 2008 Red Hat. Written by skvidal@fedoraproject.org
/bin/zip: ight (c)H
/bin/zip: Copyright (c) 1990-2008 Info-ZIP.  All rights reserved.
/bin/zip: Copyright (c) 1990-2008 Info-ZIP - Type '%s "-L"' for software license.
/bin/zipcloak: Copyright (c) 1990-2008 Info-ZIP.  All rights reserved.
/bin/zipcloak: Copyright (c) 1990-2008 Info-ZIP - Type '%s "-L"' for software license.
/bin/zipnote: Copyright (c) 1990-2008 Info-ZIP.  All rights reserved.
/bin/zipnote: Copyright (c) 1990-2008 Info-ZIP - Type '%s "-L"' for software license.
/bin/zipsplit: Copyright (c) 1990-2008 Info-ZIP.  All rights reserved.
/bin/zipsplit: Copyright (c) 1990-2008 Info-ZIP - Type '%s "-L"' for software license.
```

Important : Notez que l'option -f a pour conséquence d'imprimer le nom du fichier contenant la chaîne au début de chaque ligne.

Options de la commande

A faire : Utilisez l'option **–help** de la commande **strings** pour visualiser les options de la commande.

15.12 - La commande comm

La commande **comm** est utilisée pour comparer deux fichiers texte. La sortie de la commande sépare les lignes en trois catégories :

- Les lignes présentes seulement dans le premier fichier,
- Les lignes présentes seulement dans le deuxième fichier,
- Les lignes présentes dans la deux fichiers.

Utilisez la commande **comm** pour comparer les fichiers **/etc/passwd** et **/root/passwd** :

```
[root@centos7 ~]# comm /etc/passwd /root/passwd
        root:x:0:0:root:/root:/bin/bash
        bin:x:1:1:bin:/bin:/sbin/nologin
        daemon:x:2:2:daemon:/sbin:/sbin/nologin
        adm:x:3:4:adm:/var/adm:/sbin/nologin
        lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
        sync:x:5:0:sync:/sbin:/bin/sync
        shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
        halt:x:7:0:halt:/sbin:/sbin/halt
        mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
        operator:x:11:0:operator:/root:/sbin/nologin
        games:x:12:100:games:/usr/games:/sbin/nologin
        ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
        nobody:x:99:99:Nobody:/:/sbin/nologin
        dbus:x:81:81:System message bus:/:/sbin/nologin
        polkitd:x:999:998:User for polkitd:/:/sbin/nologin
```

```
        unbound:x:998:997:Unbound DNS resolver:/etc/unbound:/sbin/nologin
        colord:x:997:996:User for colord:/var/lib/colord:/sbin/nologin
        usbmuxd:x:113:113:usbmuxd user:/:/sbin/nologin
        avahi:x:70:70:Avahi mDNS/DNS-SD Stack:/var/run/avahi-daemon:/sbin/nologin
        avahi-autoipd:x:170:170:Avahi IPv4LL Stack:/var/lib/avahi-autoipd:/sbin/nologin
        saslauth:x:996:76:"Saslauthd user":/run/saslauthd:/sbin/nologin
        qemu:x:107:107:qemu user:/:/sbin/nologin
        libstoragemgmt:x:995:994:daemon account for libstoragemgmt:/var/run/lsm:/sbin/nologin
        rpc:x:32:32:Rpcbind Daemon:/var/lib/rpcbind:/sbin/nologin
        rpcuser:x:29:29:RPC Service User:/var/lib/nfs:/sbin/nologin
        nfsnobody:x:65534:65534:Anonymous NFS User:/var/lib/nfs:/sbin/nologin
        rtkit:x:172:172:RealtimeKit:/proc:/sbin/nologin
        radvd:x:75:75:radvd user:/:/sbin/nologin
        ntp:x:38:38::/etc/ntp:/sbin/nologin
        chrony:x:994:993::/var/lib/chrony:/sbin/nologin
        abrt:x:173:173::/etc/abrt:/sbin/nologin
        pulse:x:171:171:PulseAudio System Daemon:/var/run/pulse:/sbin/nologin
        gdm:x:42:42::/var/lib/gdm:/sbin/nologin
        gnome-initial-setup:x:993:991::/run/gnome-initial-setup/:/sbin/nologin
        postfix:x:89:89::/var/spool/postfix:/sbin/nologin
        sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
        tcpdump:x:72:72::/:/sbin/nologin
    trainee10:x:1000:1000:trainee:/home/trainee:/bin/bash
trainee:x:1000:1000:trainee:/home/trainee:/bin/bash
    tss:x:59:59:Account used by the trousers package to sandbox the tcsd daemon:/dev/null:/sbin/nologin
comm: file 2 is not in sorted order
    Linux est super!
vboxadd:x:992:1::/var/run/vboxadd:/bin/false
comm: file 1 is not in sorted order
tss:x:59:59:Account used by the trousers package to sandbox the tcsd daemon:/dev/null:/sbin/nologin
```

Pour afficher uniquement les lignes présentes dans les deux fichiers, il convient d'utiliser les options **-1** et **-2** :

```
[root@centos7 ~]# comm -12 /etc/passwd /root/passwd
```

```
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:99:99:Nobody:/:/sbin/nologin
dbus:x:81:81:System message bus:/:/sbin/nologin
polkitd:x:999:998:User for polkitd:/:/sbin/nologin
unbound:x:998:997:Unbound DNS resolver:/etc/unbound:/sbin/nologin
colord:x:997:996:User for colord:/var/lib/colord:/sbin/nologin
usbmuxd:x:113:113:usbmuxd user:/:/sbin/nologin
avahi:x:70:70:Avahi mDNS/DNS-SD Stack:/var/run/avahi-daemon:/sbin/nologin
avahi-autoipd:x:170:170:Avahi IPv4LL Stack:/var/lib/avahi-autoipd:/sbin/nologin
saslauth:x:996:76:"Saslauthd user":/run/saslauthd:/sbin/nologin
qemu:x:107:107:qemu user:/:/sbin/nologin
libstoragemgmt:x:995:994:daemon account for libstoragemgmt:/var/run/lsm:/sbin/nologin
rpc:x:32:32:Rpcbind Daemon:/var/lib/rpcbind:/sbin/nologin
rpcuser:x:29:29:RPC Service User:/var/lib/nfs:/sbin/nologin
nfsnobody:x:65534:65534:Anonymous NFS User:/var/lib/nfs:/sbin/nologin
rtkit:x:172:172:RealtimeKit:/proc:/sbin/nologin
radvd:x:75:75:radvd user:/:/sbin/nologin
ntp:x:38:38::/etc/ntp:/sbin/nologin
chrony:x:994:993::/var/lib/chrony:/sbin/nologin
abrt:x:173:173::/etc/abrt:/sbin/nologin
pulse:x:171:171:PulseAudio System Daemon:/var/run/pulse:/sbin/nologin
gdm:x:42:42::/var/lib/gdm:/sbin/nologin
gnome-initial-setup:x:993:991::/run/gnome-initial-setup/:/sbin/nologin
```

```
postfix:x:89:89::/var/spool/postfix:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
tcpdump:x:72:72::/:/sbin/nologin
comm: file 2 is not in sorted order
comm: file 1 is not in sorted order
```

Options de la commande

A faire : Utilisez l'option **–help** de la commande **comm** pour visualiser les options de la commande.

15.13 - La commande head

La commande **head** permet d'afficher les **x** premières lignes d'un fichier. Sans options, la valeur de **x** est de 10 par défaut :

```
[root@centos7 ~]# head /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
```

Avec l'option **-n**, la valeur de **x** peut être spécifiée :

```
[root@centos7 ~]# head -n 15 /etc/passwd
```

```
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:99:99:Nobody:/:/sbin/nologin
dbus:x:81:81:System message bus:/:/sbin/nologin
polkitd:x:999:998:User for polkitd:/:/sbin/nologin
```

La commande **head** peut également être utilisée pour afficher les premiers **y** octets en utilisant l'option **-c** :

```
[root@centos7 ~]# head -c 150 /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7[root@centos7 ~]#
```

Dans le cas où le **y** est négatif, la commande **head** affiche tous les octets du fichier sauf les derniers **y** octets :

```
lp:x:4:7[root@centos7 ~]# head -c -150 /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
```

```
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:99:99:Nobody:/:/sbin/nologin
dbus:x:81:81:System message bus:/:/sbin/nologin
polkitd:x:999:998:User for polkitd:/:/sbin/nologin
unbound:x:998:997:Unbound DNS resolver:/etc/unbound:/sbin/nologin
colord:x:997:996:User for colord:/var/lib/colord:/sbin/nologin
usbmuxd:x:113:113:usbmuxd user:/:/sbin/nologin
avahi:x:70:70:Avahi mDNS/DNS-SD Stack:/var/run/avahi-daemon:/sbin/nologin
avahi-autoipd:x:170:170:Avahi IPv4LL Stack:/var/lib/avahi-autoipd:/sbin/nologin
saslauth:x:996:76:"Saslauthd user":/run/saslauthd:/sbin/nologin
qemu:x:107:107:qemu user:/:/sbin/nologin
libstoragemgmt:x:995:994:daemon account for libstoragemgmt:/var/run/lsm:/sbin/nologin
rpc:x:32:32:Rpcbind Daemon:/var/lib/rpcbind:/sbin/nologin
rpcuser:x:29:29:RPC Service User:/var/lib/nfs:/sbin/nologin
nfsnobody:x:65534:65534:Anonymous NFS User:/var/lib/nfs:/sbin/nologin
rtkit:x:172:172:RealtimeKit:/proc:/sbin/nologin
radvd:x:75:75:radvd user:/:/sbin/nologin
ntp:x:38:38::/etc/ntp:/sbin/nologin
chrony:x:994:993::/var/lib/chrony:/sbin/nologin
abrt:x:173:173::/etc/abrt:/sbin/nologin
pulse:x:171:171:PulseAudio System Daemon:/var/run/pulse:/sbin/nologin
gdm:x:42:42::/var/lib/gdm:/sbin/nologin
gnome-initial-setup:x:993:991::/run/gnome-initial-setup/:/sbin/nologin
postfix:x:89:89::/var/spool/postfix:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
tcpdump:x:72:72::/:/sbin/nologin
trainee:x:1000:1000:trainee:/home/trainee:/bin/[root@centos7 ~]#
```

Les valeurs **x** et **y** acceptent des multiplicateurs :

```
[root@centos7 ~]# head -c 1b /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:99:99:Nobody:/:/sbin/nologin[root@centos7 ~]#
[root@centos7 ~]# head -c 512 /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:99:99:Nobody:/:/sbin/nologin[root@centos7 ~]#
```

Les multiplicateurs les plus utilisés sont :

Multiplicateur	Valeur en octets
b	512
KB	1000

Multiplicateur	Valeur en octets
K	1024
MB	1000*1000
M	1024*1024
GB	1000*1000*1000
G	1024*1024*1024

Options de la commande

A faire : Utilisez l'option **–help** de la commande **head** pour visualiser les options de la commande.

15.14 - La commande tail

La commande **tail** permet d'afficher les **x** dernières lignes d'un fichier. Sans options, la valeur de **x** est de 10 par défaut :

```
[root@centos7 ~]# tail /etc/passwd
abrt:x:173:173::/etc/abrt:/sbin/nologin
pulse:x:171:171:PulseAudio System Daemon:/var/run/pulse:/sbin/nologin
gdm:x:42:42::/var/lib/gdm:/sbin/nologin
gnome-initial-setup:x:993:991::/run/gnome-initial-setup/:/sbin/nologin
postfix:x:89:89::/var/spool/postfix:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
tcpdump:x:72:72::/:/sbin/nologin
trainee:x:1000:1000:trainee:/home/trainee:/bin/bash
vboxadd:x:992:1::/var/run/vboxadd:/bin/false
tss:x:59:59:Account used by the trousers package to sandbox the tcsd daemon:/dev/null:/sbin/nologin
```

Avec l'option **-n**, la valeur de **x** peut être spécifiée :

```
[root@centos7 ~]# tail -n 15 /etc/passwd
nfsnobody:x:65534:65534:Anonymous NFS User:/var/lib/nfs:/sbin/nologin
rtkit:x:172:172:RealtimeKit:/proc:/sbin/nologin
radvd:x:75:75:radvd user:/:/sbin/nologin
ntp:x:38:38::/etc/ntp:/sbin/nologin
chrony:x:994:993::/var/lib/chrony:/sbin/nologin
abrt:x:173:173::/etc/abrt:/sbin/nologin
pulse:x:171:171:PulseAudio System Daemon:/var/run/pulse:/sbin/nologin
gdm:x:42:42::/var/lib/gdm:/sbin/nologin
gnome-initial-setup:x:993:991::/run/gnome-initial-setup/:/sbin/nologin
postfix:x:89:89::/var/spool/postfix:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
tcpdump:x:72:72::/:/sbin/nologin
trainee:x:1000:1000:trainee:/home/trainee:/bin/bash
vboxadd:x:992:1::/var/run/vboxadd:/bin/false
tss:x:59:59:Account used by the trousers package to sandbox the tcsd daemon:/dev/null:/sbin/nologin
```

La commande **tail** peut également être utilisée pour afficher les derniers **y** octets en utilisant l'option **-c** :

```
[root@centos7 ~]# tail -c 150 /etc/passwd
bash
vboxadd:x:992:1::/var/run/vboxadd:/bin/false
tss:x:59:59:Account used by the trousers package to sandbox the tcsd daemon:/dev/null:/sbin/nologin
```

Dans le cas où le **y** est positif, la commande **tail** affiche tous les octets du fichier à partir de la position de **y**ième octet :

```
[root@centos7 ~]# tail -c +150 /etc/passwd
7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
```

```
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:99:99:Nobody:/:/sbin/nologin
dbus:x:81:81:System message bus:/:/sbin/nologin
polkitd:x:999:998:User for polkitd:/:/sbin/nologin
unbound:x:998:997:Unbound DNS resolver:/etc/unbound:/sbin/nologin
colord:x:997:996:User for colord:/var/lib/colord:/sbin/nologin
usbmuxd:x:113:113:usbmuxd user:/:/sbin/nologin
avahi:x:70:70:Avahi mDNS/DNS-SD Stack:/var/run/avahi-daemon:/sbin/nologin
avahi-autoipd:x:170:170:Avahi IPv4LL Stack:/var/lib/avahi-autoipd:/sbin/nologin
saslauth:x:996:76:"Saslauthd user":/run/saslauthd:/sbin/nologin
qemu:x:107:107:qemu user:/:/sbin/nologin
libstoragemgmt:x:995:994:daemon account for libstoragemgmt:/var/run/lsm:/sbin/nologin
rpc:x:32:32:Rpcbind Daemon:/var/lib/rpcbind:/sbin/nologin
rpcuser:x:29:29:RPC Service User:/var/lib/nfs:/sbin/nologin
nfsnobody:x:65534:65534:Anonymous NFS User:/var/lib/nfs:/sbin/nologin
rtkit:x:172:172:RealtimeKit:/proc:/sbin/nologin
radvd:x:75:75:radvd user:/:/sbin/nologin
ntp:x:38:38::/etc/ntp:/sbin/nologin
chrony:x:994:993::/var/lib/chrony:/sbin/nologin
abrt:x:173:173::/etc/abrt:/sbin/nologin
pulse:x:171:171:PulseAudio System Daemon:/var/run/pulse:/sbin/nologin
gdm:x:42:42::/var/lib/gdm:/sbin/nologin
gnome-initial-setup:x:993:991::/run/gnome-initial-setup/:/sbin/nologin
postfix:x:89:89::/var/spool/postfix:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
tcpdump:x:72:72::/:/sbin/nologin
trainee:x:1000:1000:trainee:/home/trainee:/bin/bash
vboxadd:x:992:1::/var/run/vboxadd:/bin/false
tss:x:59:59:Account used by the trousers package to sandbox the tcsd daemon:/dev/null:/sbin/nologin
```

Les valeurs **x** et **y** acceptent des multiplicateurs :

```
[root@centos7 ~]# tail -c 1b /etc/passwd
PulseAudio System Daemon:/var/run/pulse:/sbin/nologin
```

```
gdm:x:42:42::/var/lib/gdm:/sbin/nologin
gnome-initial-setup:x:993:991::/run/gnome-initial-setup/:/sbin/nologin
postfix:x:89:89::/var/spool/postfix:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
tcpdump:x:72:72::/:/sbin/nologin
trainee:x:1000:1000:trainee:/home/trainee:/bin/bash
vboxadd:x:992:1::/var/run/vboxadd:/bin/false
tss:x:59:59:Account used by the trousers package to sandbox the tcsd daemon:/dev/null:/sbin/nologin
[root@centos7 ~]# tail -c 512 /etc/passwd
PulseAudio System Daemon:/var/run/pulse:/sbin/nologin
gdm:x:42:42::/var/lib/gdm:/sbin/nologin
gnome-initial-setup:x:993:991::/run/gnome-initial-setup/:/sbin/nologin
postfix:x:89:89::/var/spool/postfix:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
tcpdump:x:72:72::/:/sbin/nologin
trainee:x:1000:1000:trainee:/home/trainee:/bin/bash
vboxadd:x:992:1::/var/run/vboxadd:/bin/false
tss:x:59:59:Account used by the trousers package to sandbox the tcsd daemon:/dev/null:/sbin/nologin
```

Les multiplicateurs les plus utilisés sont :

Multiplicateur	Valeur en octets
b	512
KB	1000
K	1024
MB	1000*1000
M	1024*1024
GB	1000*1000*1000
G	1024*1024*1024

Une option intéressante pour la surveillance des fichiers de journalisation est **-f**. Cette option met à jour l'affichage au fur et au mesure que le fichier est mis à jour :

```
[root@centos7 ~]# tail -f /var/log/messages
Oct 19 15:58:06 centos7 dbus-daemon: dbus[542]: [system] Activating service name='org.freedesktop.PackageKit'
(using servicehelper)
Oct 19 15:58:06 centos7 dbus[542]: [system] Activating service name='org.freedesktop.PackageKit' (using
servicehelper)
Oct 19 15:58:06 centos7 dbus-daemon: dbus[542]: [system] Successfully activated service
'org.freedesktop.PackageKit'
Oct 19 15:58:06 centos7 dbus[542]: [system] Successfully activated service 'org.freedesktop.PackageKit'
Oct 19 16:00:01 centos7 systemd: Created slice user-0.slice.
Oct 19 16:00:01 centos7 systemd: Starting Session 17 of user root.
Oct 19 16:00:01 centos7 systemd: Started Session 17 of user root.
Oct 19 16:01:01 centos7 systemd: Created slice user-0.slice.
Oct 19 16:01:01 centos7 systemd: Starting Session 18 of user root.
Oct 19 16:01:01 centos7 systemd: Started Session 18 of user root.
^C
```

Options de la commande

A faire : Utilisez l'option **–help** de la commande **tail** pour visualiser les options de la commande.

LAB #16 - Utiliser les commandes ifconfig, grep, tr et cut pour isoler l'adresse IPv4

```
[root@centos7 ~]# ifconfig enp0s3
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.2.15  netmask 255.255.255.0  broadcast 10.0.2.255
        inet6 fe80::a00:27ff:febd:f523  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:bd:f5:23  txqueuelen 1000  (Ethernet)
        RX packets 7462  bytes 6158140 (5.8 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
```

```
        TX packets 4130  bytes 555287 (542.2 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

[root@centos7 ~]# ifconfig enp0s3 | grep "inet"
        inet 10.0.2.15  netmask 255.255.255.0  broadcast 10.0.2.255
        inet6 fe80::a00:27ff:febd:f523  prefixlen 64  scopeid 0x20<link>

[root@centos7 ~]# ifconfig enp0s3 | grep "inet" | grep -v "inet6"
        inet 10.0.2.15  netmask 255.255.255.0  broadcast 10.0.2.255

[root@centos7 ~]# ifconfig enp0s3 | grep "inet" | grep -v "inet6" | tr -s " " ":"
:inet:10.0.2.15:netmask:255.255.255.0:broadcast:10.0.2.255

[root@centos7 ~]# ifconfig enp0s3 | grep "inet" | grep -v "inet6" | tr -s " " ":" | cut -d: -f3
10.0.2.15
```

Important : Notez l'utilisation de l'option -s avec la commande tr. Cette option permet de remplacer une suite de x caractères identiques par un seul caractère.

LAB #17 - Utiliser les commandes ip, grep, awk et sed pour isoler l'adresse IPv4

```
[root@centos7 ~]# ip addr show enp0s3
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 08:00:27:03:97:dd brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global noprefixroute dynamic enp0s3
       valid_lft 85977sec preferred_lft 85977sec
    inet6 fe80::1629:eb59:3b24:a1e4/64 scope link noprefixroute
       valid_lft forever preferred_lft forever

[root@centos7 ~]# ip addr show enp0s3 | grep "inet"
```

```
    inet 10.0.2.15/24 brd 10.0.2.255 scope global noprefixroute dynamic enp0s3
    inet6 fe80::1629:eb59:3b24:a1e4/64 scope link noprefixroute

[root@centos7 ~]# ip addr show enp0s3 | grep "inet" | grep -v "inet6"
    inet 10.0.2.15/24 brd 10.0.2.255 scope global noprefixroute dynamic enp0s3

[root@centos7 ~]# ip addr show enp0s3 | grep "inet" | grep -v "inet6" | awk '{ print $2; }'
10.0.2.15/24

[root@centos7 ~]# ip addr show enp0s3 | grep "inet" | grep -v "inet6" | awk '{ print $2; }' | sed 's/\/.*$//'
10.0.2.15
```

Le Shell

Un shell est un **interpréteur de commandes** ou en anglais un **Command Line Interpreter (C.L.I)**. Il est utilisé comme interface pour donner des instructions ou **commandes** au système d'exploitation.

Le mot shell est générique. Il existe de nombreux shells dans le monde Unix, par exemple :

Shell	**Nom**	**Date de Sortie**	**Inventeur**	**Commande**	**Commentaires**
tsh	Thompson Shell	1971	Ken Thompson	sh	Le premier shell
sh	Bourne Shell	1977	Stephen Bourne	sh	Le shell commun à tous les Unix. Sous RHEL/CentOS 7 : /usr/bin/sh
csh	C-Shell	1978	Bill Joy	csh	Le shell BSD. Sous RHEL/CentOS 7 : /usr/bin/csh
tcsh	Tenex C-Shell	1979	Ken Greer	tcsh	Un dérivé du shell csh. Sous RHEL/CentOS 7 : /usr/bin/tcsh
ksh	Korn Shell	1980	David Korn	ksh	Uniquement libre depuis 2005. Sous RHEL/CentOS 7 : /usr/bin/ksh
bash	Bourne Again Shell	1987	Brian Fox	bash	Le shell par défaut de Linux et de MacOS X. Sous RHEL/CentOS 7 : /usr/bin/bash
zsh	Z Shell	1990	Paul Falstad	zsh	Zsh est plutôt orienté pour l'interactivité avec l'utilisateur. Sous RHEL/CentOS 7 : /usr/bin/zsh

Sous RHEL/CentOS 7 le shell **/bin/sh** est un lien symbolique vers **/bin/bash** :

```
[trainee@centos7 ~]$ ls -l /bin/sh
lrwxrwxrwx. 1 root root 4 30 sept. 06:01 /bin/sh -> bash
```

Le Shell /bin/bash

Ce module concerne l'utilisation du shell **bash** sous Linux. Le shell **bash** permet de:

- Rappeler des commandes
- Générer la fin de noms de fichiers
- Utiliser des alias
- Utiliser les variables tableaux
- Utiliser les variables numériques et l'arithmétique du langage C
- Gérer des chaînes de caractères
- Utiliser les fonctions

Une commande commence toujours par un mot clef. Ce mot clef est interprété par le shell selon le type de commande et dans l'ordre qui suit :

1. Les alias
2. Les fonctions
3. Les commandes internes au shell
4. Les commandes externes au shell

Les Commandes Internes et Externes au shell

Les commandes internes au shell sont des commandes telles **cd**. Pour vérifier le type de commande, il faut utiliser la commande **type** :

```
[trainee@centos7 ~]$ type cd
cd is a shell builtin
```

Les commandes externes au shell sont des binaires exécutables ou des scripts, généralement situés dans /bin, /sbin, /usr/bin ou /usr/sbin :

```
[trainee@centos7 ~]$ type passwd
passwd is /usr/bin/passwd
```

Les alias

Les alias sont des noms permettant de désigner une commande ou une suite de commandes et ne sont spécifiques qu'au shell qui les a créés ainsi qu'à l'environnement de l'utilisateur :

```
[trainee@centos7 ~]$ type ls
ls is aliased to `ls --color=auto'
```

Important : Notez que dans ce cas l'alias **ls** est en effet un alias qui utilise la **commande** ls elle-même.

Un alias se définit en utilisant la commande **alias** :

```
[trainee@centos7 ~]$ alias dir='ls -l'
[trainee@centos7 ~]$ dir
total 4
-rw-rw-r--. 1 trainee trainee   0 29 sept. 18:20 aac
-rw-rw-r--. 1 trainee trainee   0 29 sept. 18:20 abc
-rw-rw-r--. 1 trainee trainee   0 29 sept. 18:20 bca
drwxr-xr-x. 2 trainee trainee   6 30 avril 11:54 Desktop
drwxr-xr-x. 2 trainee trainee   6 30 avril 11:54 Documents
drwxr-xr-x. 2 trainee trainee   6 30 avril 11:54 Downloads
drwxr-xr-x. 2 trainee trainee   6 30 avril 11:54 Music
drwxr-xr-x. 2 trainee trainee   6 30 avril 11:54 Pictures
drwxr-xr-x. 2 trainee trainee   6 30 avril 11:54 Public
drwxr-xr-x. 2 trainee trainee   6 30 avril 11:54 Templates
drwxr-xr-x. 2 trainee trainee   6 30 avril 11:54 Videos
-rw-rw-r--. 1 trainee trainee 442 29 sept. 00:53 vitext
```

```
-rw-rw-r--. 1 trainee trainee   0 29 sept. 18:20 xyz
```

Important : Notez que la commande **dir** existe vraiment. Le fait de créer un alias qui s'appelle **dir** implique que l'alias sera exécuté à la place de la commande **dir**.

La liste des alias définis peut être visualisée en utilisant la commande **alias** :

```
[trainee@centos7 ~]$ alias
alias dir='ls -l'
alias egrep='egrep --color=auto'
alias fgrep='fgrep --color=auto'
alias grep='grep --color=auto'
alias l.='ls -d .* --color=auto'
alias ll='ls -l --color=auto'
alias ls='ls --color=auto'
alias vi='vim'
alias which='alias | /usr/bin/which --tty-only --read-alias --show-dot --show-tilde'
```

Important : Notez que cette liste contient, sans distinction, les alias définis dans les fichiers de démarrage du système ainsi que l'alias **dir** créé par **trainee** qui n'est que disponible à **trainee** dans le terminal courant.

Pour forcer l'exécution d'une commande et non l'alias il faut faire précéder la commande par le caractère **** :

```
[trainee@centos7 ~]$ \dir
aac  bca      Documents  Music     Public     Videos  xyz
abc  Desktop  Downloads  Pictures  Templates  vitext
```

Pour supprimer un alias, il convient d'utiliser la commande **unalias** :

```
[trainee@centos7 ~]$ unalias dir
[trainee@centos7 ~]$ dir
aac  bca      Documents  Music     Public     Videos  xyz
abc  Desktop  Downloads  Pictures  Templates  vitext
```

Le shell des utilisateurs est défini par **root** dans le dernier champs du fichier **/etc/passwd** :

```
[trainee@centos7 ~]$ cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:99:99:Nobody:/:/sbin/nologin
avahi-autoipd:x:170:170:Avahi IPv4LL Stack:/var/lib/avahi-autoipd:/sbin/nologin
systemd-bus-proxy:x:999:997:systemd Bus Proxy:/:/sbin/nologin
systemd-network:x:998:996:systemd Network Management:/:/sbin/nologin
dbus:x:81:81:System message bus:/:/sbin/nologin
polkitd:x:997:995:User for polkitd:/:/sbin/nologin
abrt:x:173:173::/etc/abrt:/sbin/nologin
usbmuxd:x:113:113:usbmuxd user:/:/sbin/nologin
colord:x:996:993:User for colord:/var/lib/colord:/sbin/nologin
libstoragemgmt:x:995:992:daemon account for libstoragemgmt:/var/run/lsm:/sbin/nologin
setroubleshoot:x:994:991::/var/lib/setroubleshoot:/sbin/nologin
rpc:x:32:32:Rpcbind Daemon:/var/lib/rpcbind:/sbin/nologin
rtkit:x:172:172:RealtimeKit:/proc:/sbin/nologin
chrony:x:993:990::/var/lib/chrony:/sbin/nologin
```

```
unbound:x:992:989:Unbound DNS resolver:/etc/unbound:/sbin/nologin
tss:x:59:59:Account used by the trousers package to sandbox the tcsd daemon:/dev/null:/sbin/nologin
geoclue:x:991:988:User for geoclue:/var/lib/geoclue:/sbin/nologin
ntp:x:38:38::/etc/ntp:/sbin/nologin
sssd:x:990:987:User for sssd:/:/sbin/nologin
rpcuser:x:29:29:RPC Service User:/var/lib/nfs:/sbin/nologin
nfsnobody:x:65534:65534:Anonymous NFS User:/var/lib/nfs:/sbin/nologin
pulse:x:171:171:PulseAudio System Daemon:/var/run/pulse:/sbin/nologin
gdm:x:42:42::/var/lib/gdm:/sbin/nologin
gnome-initial-setup:x:989:984::/run/gnome-initial-setup/:/sbin/nologin
avahi:x:70:70:Avahi mDNS/DNS-SD Stack:/var/run/avahi-daemon:/sbin/nologin
postfix:x:89:89::/var/spool/postfix:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
tcpdump:x:72:72::/:/sbin/nologin
trainee:x:1000:1000:trainee:/home/trainee:/bin/bash
vboxadd:x:988:1::/var/run/vboxadd:/bin/false
named:x:25:25:Named:/var/named:/sbin/nologin
```

Cependant l'utilisateur peut changer son shell grâce à la commande **chsh**. Les shells disponibles aux utilisateurs du système sont inscrits dans le fichier **/etc/shells**. Saisissez la commande **cat /etc/shells** :

```
[trainee@centos7 ~]$ cat /etc/shells
/bin/sh
/bin/bash
/sbin/nologin
/usr/bin/sh
/usr/bin/bash
/usr/sbin/nologin
/bin/tcsh
/bin/csh
```

Ensuite utilisez la commande **echo** pour afficher le shell actuel de **trainee** :

```
[trainee@centos7 ~]$ echo $SHELL
```

```
/bin/bash
```

Important : Notez sous RHEL/CentOS 7 que le système nous informe que le shell courant de l'utiisateur **trainee** est **/bin/bash** et non **/usr/bin/bash**. Ceci est du au fait que le répertoire /bin est un lien symbolique pointant vers le répertoire /usr/bin.

Changez ensuite le shell de **trainee** en utilisant la commande **chsh** en indiquant la valeur de **/bin/sh** pour le nouveau shell :

```
[trainee@centos7 ~]$ chsh
Changing shell for trainee.
New shell [/bin/bash]: /bin/sh
Password: trainee
Shell changed.
```

Important : Notez que le mot de passe saisi ne sera **pas** visible.

Vérifiez ensuite le shell actif pour **trainee** :

```
[trainee@centos7 ~]$ echo $SHELL
/bin/bash
```

Dernièrement contrôlez le shell stipulé dans le fichier **/etc/passwd** pour **trainee** :

```
[trainee@centos7 ~]$ cat /etc/passwd | grep trainee
trainee:x:1000:1000:trainee:/home/trainee:/bin/sh
```

Important : Vous noterez que le shell actif est toujours **/bin/bash** tandis que le shell stipulé dans le fichier /etc/passwd est le **/bin/sh**. Le shell **/bin/sh** ne deviendra le shell actif de **trainee** que lors de sa prochaine connexion au système.

Modifiez votre shell à **/bin/bash** de nouveau en utilisant la commande chsh :

```
[trainee@centos7 ~]$ chsh
Changing shell for trainee.
New shell [/bin/sh]: /bin/bash
Password: trainee
Shell changed.
```

Important : Notez que le mot de passe saisi ne sera **pas** visible.

Le Prompt

Le prompt d'un utilisateur dépend de son statut :

- **$** pour un utilisateur normal,
- **#** pour root.

Rappeler des Commandes

Le shell **/bin/bash** permet le rappel des dernières commandes saisies. Afin de connaître la liste des commandes mémorisées, utilisez la commande history :

```
[trainee@centos7 ~]$ history | more
    1  su -
    2  df -h
    3  su -
    4  exit
    5  su -
    6  su -
```

```
    7  vi vitext
    8  view vitext
    9  vi vitext
   10  locale
   11  LANG=en_GB.UTF-8
   12  export LANG
   13  locale
   14  vi vitext
   15  vi .exrc
   16  vi vitext
   17  clear
   18  stty -a
   19  date
   20  locale
   21  who
   22  df
   23  df -h
--More--
```

Important: L'historique est spécifique à chaque utilisateur.

L'historique des commandes est en mode **emacs** par défaut. De ce fait, le rappel de la dernière commande se fait en utilisant la touche **[Flèche vers le haut]** ou bien les touches **[CTRL]-[P]** et le rappel de la commande suivante se fait en utilisant la touche **[Flèche vers le bas]** ou bien les touches **[CTRL]-[N]** :

Caractère de Contrôle	**Définition**
[CTRL]-[P] (= flèche vers le haut)	Rappelle la commande précédente
[CTRL]-[N] (= flèche vers le bas)	Rappelle la commande suivante

Pour se déplacer dans la ligne de l'historique :

Caractère de Contrôle	Définition
[CTRL]-[A]	Se déplacer au début de la ligne
[CTRL]-[E]	Se déplacer à la fin de la ligne
[CTRL]-[B]	Se déplacer un caractère à gauche
[CTRL]-[F]	Se déplacer un caractère à droite
[CTRL]-[D]	Supprimer le caractère sous le curseur

Pour rechercher dans l'historique il convient d'utiliser les touches :

Caractère de Contrôle	Définition
[CTRL]-[R] *chaine*	Recherche en arrière de *chaine* dans l'historique. L'utilisation successive de la combinaison de touches par la suite recherche d'autres occurences de *chaine*
[CTRL]-[S] *chaine*	Recherche en avant de *chaine* dans l'historique. L'utilisation successive de la combinaison de touches par la suite recherche d'autres occurences de *chaine*
[CTRL]-[G]	Sortir du mode recherche

Il est aussi possible de rappeler la dernière commande de l'historique en utilisant les caractères **!!**:

```
[trainee@centos7 ~]$ ls
aac  bca      Documents  Music     Public     Videos  xyz
abc  Desktop  Downloads  Pictures  Templates  vitext
[trainee@centos7 ~]$ !!
ls
aac  bca      Documents  Music     Public     Videos  xyz
abc  Desktop  Downloads  Pictures  Templates  vitext
```

Vous pouvez rappeler une commande spécifique de l'historique en utilisant le caractère **!** suivi du numéro de la commande à rappeler :

```
[trainee@centos7 ~]$ !123
ls
aac  bca      Documents  Music     Public     Videos  xyz
abc  Desktop  Downloads  Pictures  Templates  vitext
```

Le paramétrage de la fonction du rappel des commandes est fait pour tous les utilisateurs dans le fichier **/etc/profile**. Dans ce fichier, les variables concernant le rappel des commandes peuvent être définis. Le plus important est **HISTSIZE** :

```
[trainee@centos7 ~]$ cat /etc/profile | grep HISTSIZE
HISTSIZE=1000
export PATH USER LOGNAME MAIL HOSTNAME HISTSIZE HISTCONTROL
```

Vous noterez que dans le cas précédent, la valeur de **HISTSIZE** est de **1000**. Ceci implique que les dernières mille commandes sont mémorisées.

Les commandes mémorisées sont stockées dans le fichier **~/.bash_history**. Les commandes de la session en cours ne sont sauvegardées dans ce fichier qu'à la fermerture de la session :

```
[trainee@centos7 ~]$ nl .bash_history | more
     1  su -
     2  df -h
     3  su -
     4  exit
     5  su -
     6  su -
     7  vi vitext
     8  view vitext
     9  vi vitext
    10  locale
    11  LANG=en_GB.UTF-8
    12  export LANG
    13  locale
    14  vi vitext
    15  vi .exrc
    16  vi vitext
    17  clear
    18  stty -a
    19  date
    20  locale
    21  who
```

```
    22  df
    23  df -h
--More--
```

Important : Notez l'utilisation de la commande **nl** pour numéroter les lignes de l'affichage du contenu du fichier **.bash_history**.

Générer les fins de noms de fichiers

Le shell /bin/bash permet la génération des fins de noms de fichiers. Celle-ci est accomplie grâce à l'utilisation de la touche **[Tab]**. Dans l'exemple qui suit, la commande saisie est :

```
$ ls .b [Tab][Tab][Tab]
```

```
[trainee@centos7 ~]$ ls .bash
.bash_history  .bash_logout   .bash_profile  .bashrc
```

Important : Notez qu'en appuyant sur la touche Tab trois fois le shell propose 3 ou 4 possibilités de complétion de nom de fichier. En effet, sans plus d'information, le shell ne sait pas quel fichier est concerné.

La même possibilité existe pour la génération des fins de noms de commandes. Dans ce cas saisissez la commande suivante :

```
$ mo [Tab][Tab]
```

Appuyez sur la touche Tab deux fois. Vous obtiendrez une fenêtre similaire à celle-ci :

```
[trainee@centos7 ~]$ mo
mobj_dump        modutil          mount.cifs       mount.nfs4       mousetweaks
```

```
modifyrepo       mokutil          mount.fuse       mountpoint
modinfo          more             mount.glusterfs  mountstats
modprobe         mount            mount.nfs        mount.vboxsf
```

Le shell interactif

Lors de l'utilisation du shell, nous avons souvent besoin d'exécuter une commande sur plusieurs fichiers au lieu de les traiter individuellement. A cette fin nous pouvons utiliser les caractères spéciaux.

Caractère Spéciaux	Description
*	Représente 0 ou plus de caractères
?	Représente un caractère
[abc]	Représente un caractère parmi ceux entre crochets
[!abc]	Représente un caractère ne trouvant pas parmi ceux entre crochets
?(expression1|expression2| ...)	Représente 0 ou 1 fois l'expression1 ou 0 ou 1 fois l'expression2 …
***(expression1|expression2| ...)**	Représente 0 à x fois l'expression1 ou 0 à x fois l'expression2 …
+(expression1|expression2| ...)	Représente 1 à x fois l'expression1 ou 1 à x fois l'expression2 …
@(expression1|expression2| ...)	Représente 1 fois l'expression1 ou 1 fois l'expression2 …
!(expression1|expression2| ...)	Représente 0 fois l'expression1 ou 0 fois l'expression2 …

Caractère *

Dans votre répertoire individuel, créez un répertoire **training**. Ensuite créez dans ce répertoire 5 fichiers nommés respectivement f1, f2, f3, f4 et f5 :

```
[trainee@centos7 ~]$ mkdir training
[trainee@centos7 ~]$ cd training
[trainee@centos7 training]$ touch f1 f2 f3 f4 f5
```

Afin de démontrer l'utilisation du caractère spécial *, saisissez la commande suivante :

```
[trainee@centos7 training]$ echo f*
```

```
f1 f2 f3 f4 f5
```

Important : Notez que le caractère * remplace un caractère ou une suite de caractères.

Caractère ?

Créez maintenant les fichiers f52 et f62 :

```
[trainee@centos7 training]$ touch f52 f62
```

Saisissez ensuite la commande suivante :

```
[trainee@centos7 training]$ echo f?2
f52 f62
```

Important : Notez que le caractère **?** remplace **un seul** caractère.

Caractères [ ]

L'utilisation peut prendre plusieurs formes différentes :

Joker	Description
[xyz]	Représente le caractère x ou y ou z
[m-t]	Représente le caractère m ou n .... t
[!xyz]	Représente un caractère autre que x ou y ou z

Joker	Description
[!m-t]	Représente un caractère autre que m ou n .... t

Afin de démontrer l'utilisation des caractères **[** et **]**, créez le fichier a100 :

```
[trainee@centos7 training]$ touch a100
```

Ensuite saisissez les commandes suivantes et notez le résultat :

```
[trainee@centos7 training]$ echo [a-f]*
a100 f1 f2 f3 f4 f5 f52 f62
```

Important : Notez ici que tous les fichiers commençant par les lettres **a**, **b**, **c**, **d**, **e** ou **f** sont affichés à l'écran.

```
[trainee@centos7 training]$ echo [af]*
a100 f1 f2 f3 f4 f5 f52 f62
```

Important : Notez ici que tous les fichiers commençant par les lettres **a** ou **f** sont affichés à l'écran.

```
[trainee@centos7 training]$ echo [!a]*
f1 f2 f3 f4 f5 f52 f62
```

Important : Notez ici que tous les fichiers sont affichés à l'écran, à l'exception d'un fichier commençant par la lettre **a** .

```
[trainee@centos7 training]$ echo [a-b]*
```

```
a100
```

Important : Notez ici que seul le fichier commençant par la lettre **a** est affiché à l'écran car il n'existe pas de fichiers commençant par la lettre **b**.

```
[trainee@centos7 training]$ echo [a-f]
[a-f]
```

Important : Notez que dans ce cas, il n'existe pas de fichiers dénommés **a**, **b**, **c**, **d**, **e** ou **f**. Pour cette raison, n'ayant trouvé aucune correspondance entre le filtre utilisé et les objets dans le répertoire courant, le commande **echo** retourne le filtre passé en argument, c'est-à-dire **[a-f]**.

L'option extglob

Activez l'option **extglob** du shell bash afin de pouvoir utiliser **?(expression), *(expression), +(expression), @(expression) et !(expression)** :

```
[trainee@centos7 training]$ shopt -s extglob
```

La commande **shopt** est utilisée pour activer ou désactiver les options du comportement optional du shell. La liste des options peut être visualisée en exécutant la commande **shopt** sans options :

```
[trainee@centos7 training]$ shopt
autocd          off
cdable_vars     off
cdspell         off
checkhash       off
checkjobs       off
```

```
checkwinsize    on
cmdhist         on
compat31        off
compat32        off
compat40        off
compat41        off
direxpand       off
dirspell        off
dotglob         off
execfail        off
expand_aliases  on
extdebug        off
extglob         on
extquote        on
failglob        off
force_fignore   on
globstar        off
gnu_errfmt      off
histappend      on
histreedit      off
histverify      off
hostcomplete    off
huponexit       off
interactive_comments    on
lastpipe        off
lithist         off
login_shell     on
mailwarn        off
no_empty_cmd_completion off
nocaseglob      off
nocasematch     off
nullglob        off
progcomp        on
promptvars      on
```

```
restricted_shell    off
shift_verbose   off
sourcepath      on
xpg_echo        of
```

?(expression)

Créez les fichiers f, f.txt, f123.txt, f123123.txt, f123123123.txt :

```
[trainee@centos7 training]$ touch f f.txt f123.txt f123123.txt f123123123.txt
```

Saisissez la commande suivante :

```
[trainee@centos7 training]$ ls f?(123).txt
f123.txt  f.txt
```

Important : Notez ici que la commande affiche les fichiers ayant un nom contenant 0 ou 1 occurence de la chaîne **123**.

***(expression)**

Saisissez la commande suivante :

```
[trainee@centos7 training]$ ls f*(123).txt
f123123123.txt  f123123.txt  f123.txt  f.txt
```

Important : Notez ici que la commande affiche les fichiers ayant un nom contenant de 0 jusqu'à x occurences de la chaîne **123**.

+(expression)

Saisissez la commande suivante :

```
[trainee@centos7 training]$ ls f+(123).txt
f123123123.txt  f123123.txt  f123.txt
```

Important : Notez ici que la commande affiche les fichiers ayant un nom contenant entre 1 et x occurences de la chaîne **123**.

@(expression)

Saisissez la commande suivante :

```
[trainee@centos7 training]$ ls f@(123).txt
f123.txt
```

Important : Notez ici que la commande affiche les fichiers ayant un nom contenant 1 seule occurence de la chaîne **123**.

!(expression)

Saisissez la commande suivante :

```
[trainee@centos7 training]$ ls f!(123).txt
f123123123.txt  f123123.txt  f.txt
```

Important : Notez ici que la commande n'affiche que les fichiers ayant un nom qui ne contient **pas** la chaîne **123**.

Caractères d'Échappement

Afin d'utiliser un caractère spécial dans un contexte littéral, il faut utiliser un caractère d'échappement. Il existe trois caractères d'échappement :

Caractère	Description
\	Protège le caractère qui le suit
' '	Protège tout caractère, à l'exception du caractère ' lui-même, se trouvant entre les deux '
” “	Protège tout caractère, à l'exception des caractères **“** lui-même, **$**, \ et ', se trouvant entre les deux **“**

Afin d'illustrer l'utilisation des caractères d'échappement, considérons la commande suivante :

```
$ echo * est un caractère spécial [Entrée]
```

Lors de la saisie de cette commande dans votre répertoire **training**, vous obtiendrez une fenêtre similaire à celle-ci :

```
[trainee@centos7 training]$ echo * est un caractère spécial
a100 f f1 f123123123.txt f123123.txt f123.txt f2 f3 f4 f5 f52 f62 f.txt est un caractère spécial

[trainee@centos7 training]$ echo \* est un caractère spécial
* est un caractère spécial

[trainee@centos7 training]$ echo "* est un caractère spécial"
* est un caractère spécial

[trainee@centos7 training]$ echo '* est un caractère spécial'
* est un caractère spécial
```

Codes Retour

Chaque commande retourne un code à la fin de son exécution. La variable spéciale **$?** sert à stocker le code retour de la dernière commande exécutée.

Par exemple :

```
[trainee@centos7 training]$ cd ..
[trainee@centos7 ~]$ mkdir codes
[trainee@centos7 ~]$ echo $?
0
[trainee@centos7 ~]$ touch codes/exit.txt
[trainee@centos7 ~]$ rmdir codes
rmdir: failed to remove ‘codes’: Directory not empty
[trainee@centos7 ~]$ echo $?
1
```

Dans cette exemple la création du répertoire **codes** s'est bien déroulée. Le code retour stocké dans la variable $? est un zéro.

La suppression du répertoire a rencontré une erreur car **codes** contenait le fichier **retour**. Le code retour stocké dans la variable $? est un **un**.

Si le code retour est **zéro** la dernière commande s'est déroulée sans erreur.

Si le code retour est **autre que zéro** la dernière commande s'est déroulée avec une erreur.

Redirections

Votre dialogue avec le système Linux utilise des canaux d’entrée et de sortie. On appelle le clavier, le **canal d’entrée standard** et l’écran, le **canal de sortie standard** :

Autrement dit, en tapant une commande sur le clavier, vous voyez le résultat de cette commande à l’écran.

Parfois, cependant il est utile de re-diriger le canal de sortie standard vers un fichier. De cette façon, le résultat d'une commande telle **free** peut être stocké dans un fichier pour une consultation ultérieure :

Cet effet est obtenu en utilisant une **redirection** :

```
[trainee@centos7 ~]$ pwd
/home/trainee
[trainee@centos7 ~]$ cd training
[trainee@centos7 training]$ free > file
[trainee@centos7 training]$ cat file
              total        used        free      shared  buff/cache   available
Mem:         500780      192692       38916        4824      269172      260472
Swap:       2096124           0     2096124
```

Si le fichier cible n'existe pas, il est créé et son contenu sera le résultat de la commande free.

Par contre si le fichier existe déjà, il sera écrasé :

```
[trainee@centos7 training]$ date > file
[trainee@centos7 training]$ cat file
Mon 28 Nov 15:48:09 CET 2016
```

Pour ajouter des données supplémentaires au même fichier cible, il faut utiliser une **double redirection** :

```
[trainee@centos7 training]$ free >> file
[trainee@centos7 training]$ cat file
Mon 28 Nov 15:48:09 CET 2016
              total        used        free      shared  buff/cache   available
Mem:         500780      192792       38516        4824      269472      260376
Swap:       2096124           0     2096124
```

De cette façon, la date du jour sera rajoutée à la fin de votre fichier après les informations de la commande free.

Important : Notez que la sortie standard ne peut être redirigée que dans **une seule direction**.

Les canaux d’entrées et de sorties sont numérotés :

- 0 = Le Canal d’entrée Standard
- 1 = Le Canal de Sortie Standard
- 2 = Le Canal d’erreur

La commande suivante créera un fichier nommé **errorlog** qui contient les messages d’erreur de l’exécution de la commande **rmdir** :

```
[trainee@centos7 training]$ cd ..
[trainee@centos7 ~]$ rmdir training/ 2>errorlog
[trainee@centos7 ~]$ cat errorlog
rmdir: failed to remove ‘training/’: Directory not emptyy
```

En effet l'erreur est générée parce que le répertoire **training** n'est pas vide.

Nous pouvons également réunir des canaux. Pour mettre en application ceci, il faut comprendre que le shell traite les commandes de **gauche à droite**.

Dans l’exemple suivant, nous réunissons le canal de sortie et le canal d’erreurs :

```
[trainee@centos7 ~]$ free > file 2>&1
```

La syntaxe **2>&1** envoie la sortie du canal 2 au même endroit que le canal 1, à savoir le fichier dénommé **file**.

Il est possible de modifier le canal d'entrée standard afin de lire des informations à partir d’un fichier. Dans ce cas la redirection est obtenue en utilisant le caractère **<** :

```
$ wc -w < errorlog [Entrée]
```

Dans cet exemple la commande wc compte le nombre de mots (-w) dans le fichier errorlog et l'affiche à l'écran :

```
[trainee@centos7 ~]$ wc -w < errorlog
8
```

D'autres redirections existent :

Caractères	Définition
&>	Rediriger les canaux 1 et 2 au même endroit
<<	Permet d'utiliser le texte taper ensuite en tant que entrée standard. Par exemple *programme* << EOF utilisera le texte taper après en tant qu'entrée standard jusqu'à l'apparition de EOF sur une ligne seule.
<>	Permet d'utiliser le fichier specifié en tant que entrée standard et sortie standard

Pipes

Il est aussi possible de relier des commandes avec un pipe **|** .

Dans ce cas, le canal de sortie de la commande à gauche du tube est envoyé au canal d'entrée de la commande à droite du pipe :

```
$ ls | wc -w [Entrée]
```

Cette commande, lancée dans votre répertoire personnel, prend la sortie de la commande **ls** et demande à la commande **wc** de compter le nombre de mots inclus dans la sortie de ls :

```
[trainee@centos7 ~]$ ls | wc -w
17
```

Important : Il est à noter qu'il est possible de relier plusieurs tubes dans la même commande.

Rappelez-vous que la sortie standard ne peut être redirigée que dans une seule direction. Afin de pouvoir rediriger la sortie standard vers un fichier **et**

la visualiser à l'écran, nous devons utiliser la commande **tee** avec un pipe :

```
[trainee@centos7 ~]$ date | tee file1
Mon 28 Nov 16:14:24 CET 2016
[trainee@centos7 ~]$ cat file1
Mon 28 Nov 16:14:24 CET 2016
```

Cette même technique nous permet de créer **deux fichiers** :

```
$ date | tee file1 > file2 [Entrée]
```

```
[trainee@centos7 ~]$ date | tee file1 > file2
[trainee@centos7 ~]$ cat file1
Mon 28 Nov 16:15:57 CET 2016
[trainee@centos7 ~]$ cat file2
Mon 28 Nov 16:15:57 CET 2016
```

Important : Par défaut la commande tee écrase le fichier de destination. Pour ajouter des données supplémentaires au même fichier cible, il convient d'utiliser l'option **-a** de la commande tee.

Substitutions de Commandes

Il est parfois intéressant, notamment dans les scripts, de remplacer une commande par sa valeur de sa sortie. Afin d'illustrer ce point, considérons les commandes suivantes :

```
[trainee@centos7 ~]$ echo date
date
[trainee@centos7 ~]$ echo $(date)
Mon 28 Nov 16:19:35 CET 2016
```

```
[trainee@centos7 ~]$ echo `date`
Mon 28 Nov 16:19:35 CET 2016
```

Important : Notez le format de chaque substitution **$(commande)** ou **`commande`**. Sur un clavier français, l'anti-côte est accessible en utilisant les touches Alt Gr et 77.

Chainage de Commandes

Il est possible de regrouper des commandes à l'aide d'un sous-shell :

```
$ (ls -l; ps; who) > list [Entrée]
```

Cet exemple envoie le résultat des trois commandes vers le fichier **list** en les traitant en tâches de fond.

Les commandes peuvent être aussi chainées en fonction du code retour de la commande précédente.

&& est utilisé afin de s'assurer que la deuxième commande s'exécute dans le cas où la valeur du statut de sortie est 0, autrement dit qu'il n'y a pas eu d'erreurs.

|| est utilisé afin de s'assurer de l'inverse.

Le syntaxe de cette commande est :

```
Commande1 && Commande2
```

Dans ce cas, Commande 2 est exécutée uniquement dans le cas où Commande1 s'est exécuté sans erreur

Ou :

```
Commande1 || Commande2
```

Dans ce cas, Commande2 est exécuté si Commande1 a rencontré une erreur.

Affichage des variables du shell

Une variable du shell peut être affichée grâce à la commande :

```
$ echo $VARIABLE [Entrée]
```

Les variables principales

Variable	Description
BASH	Le chemin complet du shell.
BASH_VERSION	La version du shell.
EUID	EUID de l'utilisateur courant.
UID	UID de l'utilisateur courant.
PPID	Le PID du processus père.
PWD	Le répertoire courant.
OLDPWD	Le répertoire avant la dernière commande cd. Même chose que la commande **cd -**.
RANDOM	Un nombre aléatoire entre 0 et 32767
SECONDS	Le nombre de scondes écoules depuis le lancement du shell
LINES	Le nombre de lignes de l'écran.
COLUMNS	La largeur de l'écran.
HISTFILE	Le fichier historique
HISTFILESIZE	La taille du fichier historique
HISTSIZE	Le nombre de commandes mémorisées dans le fichier historique
HISTCMD	Le numéro de la commande courante dans l'historique
HISTCONTROL	**ignorespace** ou **ignoredups** ou **ignoreboth**
HOME	Le répertoire de connexion.
HOSTTYPE	Le type de machine.
OSTYPE	Le système d'exploitation.

Variable	Description
MAIL	Le fichier contenant le courrier.
MAILCHECK	La fréquence de vérification du courrier en secondes.
PATH	Le chemin de recherche des commandes.
PROMPT_COMMAND	La commande exécutée avant chaque affichage du prompt.
PS1	Le prompt par défaut.
PS2	Le deuxième prompt par défaut
PS3	Le troisième prompt par défaut
PS4	Le quatrième prompt par défaut
SHELL	Le shell de préférence.
SHLVL	Le nombre d'instances du shell.
TMOUT	Le nombre de secondes moins 60 d'inactivité avant que le shell exécute la commande **exit**.

Les Variables de Régionalisation et d'Internationalisation

L'**Internationalisation**, aussi appelé **i18n** car il y a 18 lettres entre la lettre **I** et la lettre **n** dans le mot *Internationalization*, consiste à adapter un logiciel aux paramètres variant d'une région à l'autre :

- longueur des mots,
- accents,
- écriture de gauche à droite ou de droite à gauche,
- unité monétaire,
- styles typographiques et modèles rédactionnels,
- unités de mesures,
- affichage des dates et des heures,
- formats d'impression,
- format du clavier,
- etc …

Le **Régionalisation**, aussi appelé **l10n** car il y a 10 lettres entre la lettre **L** et la lettre n du mot *Localisation*, consiste à modifier l'internalisation en fonction d'une région spécifique.

Le code pays complet prend la forme suivante : **langue-PAYS.jeu_de_caractères**. Par exemple, pour la langue anglaise les valeurs de langue-PAYS

sont :

- en_GB = Great Britain,
- en_US = USA,
- en_AU = Australia,
- en_NZ = New Zealand,
- en_ZA = South Africa,
- en_CA = Canada.

Les variables système les plus importants contenant les informations concernant le régionalisation sont :

Variable	Description
LC_ALL	Avec une valeur non nulle, celle-ci prend le dessus sur la valeur de toutes les autres variables d'internationalisation
LANG	Fournit une valeur par défaut pour les variables d'environnement dont la valeur est nulle ou non définie.
LC_CTYPE	Détermine les paramètres régionaux pour l'interprétation de séquence d'octets de données texte en caractères.

Par exemple :

```
[trainee@centos7 ~]$ echo $LC_ALL
en_GB.UTF-8
[trainee@centos7 ~]$ echo $LC_CTYPE

[trainee@centos7 ~]$ echo $LANG
en_GB.UTF-8

[trainee@centos7 ~]$ locale
LANG=en_GB.UTF-8
LC_CTYPE="en_GB.UTF-8"
LC_NUMERIC="en_GB.UTF-8"
LC_TIME="en_GB.UTF-8"
LC_COLLATE="en_GB.UTF-8"
LC_MONETARY="en_GB.UTF-8"
LC_MESSAGES="en_GB.UTF-8"
LC_PAPER="en_GB.UTF-8"
```

```
LC_NAME="en_GB.UTF-8"
LC_ADDRESS="en_GB.UTF-8"
LC_TELEPHONE="en_GB.UTF-8"
LC_MEASUREMENT="en_GB.UTF-8"
LC_IDENTIFICATION="en_GB.UTF-8"
LC_ALL=en_GB.UTF-8
```

Les variables spéciales

Variable	Description
$LINENO	Contient le numéro de la ligne courante du script ou de la fonction
$$	Contient le PID du shell en cours
$PPID	Contient le PID du processus parent du shell en cours
$0	Contient le nom du script en cours tel que ce nom ait été saisi sur la ligne de commande
$1, $2 ...	Contient respectivement le premier argument, deuxième argument etc passés au script
$#	Contient le nombre d'arguments passés au script
$*	Contient l'ensemble des arguments passés au script
$@	Contient l'ensemble des arguments passés au script

La Commande env

La commande **env** envoie sur la sortie standard les valeurs des variables système de l'environnement de l'utilisateur qui l'invoque :

```
[trainee@centos7 ~]$ env
XDG_SESSION_ID=1
HOSTNAME=centos7.fenestros.loc
SELINUX_ROLE_REQUESTED=
TERM=xterm-256color
SHELL=/bin/bash
HISTSIZE=1000
SSH_CLIENT=10.0.2.2 33896 22
SELINUX_USE_CURRENT_RANGE=
```

```
SSH_TTY=/dev/pts/0
LC_ALL=en_GB.UTF-8
USER=trainee
LS_COLORS=rs=0:di=38;5;27:ln=38;5;51:mh=44;38;5;15:pi=40;38;5;11:so=38;5;13:do=38;5;5:bd=48;5;232;38;5;11:cd=48;5
;232;38;5;3:or=48;5;232;38;5;9:mi=05;48;5;232;38;5;15:su=48;5;196;38;5;15:sg=48;5;11;38;5;16:ca=48;5;196;38;5;226
:tw=48;5;10;38;5;16:ow=48;5;10;38;5;21:st=48;5;21;38;5;15:ex=38;5;34:*.tar=38;5;9:*.tgz=38;5;9:*.arc=38;5;9:*.arj
=38;5;9:*.taz=38;5;9:*.lha=38;5;9:*.lz4=38;5;9:*.lzh=38;5;9:*.lzma=38;5;9:*.tlz=38;5;9:*.txz=38;5;9:*.tzo=38;5;9:
*.t7z=38;5;9:*.zip=38;5;9:*.z=38;5;9:*.Z=38;5;9:*.dz=38;5;9:*.gz=38;5;9:*.lrz=38;5;9:*.lz=38;5;9:*.lzo=38;5;9:*.x
z=38;5;9:*.bz2=38;5;9:*.bz=38;5;9:*.tbz=38;5;9:*.tbz2=38;5;9:*.tz=38;5;9:*.deb=38;5;9:*.rpm=38;5;9:*.jar=38;5;9:*
.war=38;5;9:*.ear=38;5;9:*.sar=38;5;9:*.rar=38;5;9:*.alz=38;5;9:*.ace=38;5;9:*.zoo=38;5;9:*.cpio=38;5;9:*.7z=38;5
;9:*.rz=38;5;9:*.cab=38;5;9:*.jpg=38;5;13:*.jpeg=38;5;13:*.gif=38;5;13:*.bmp=38;5;13:*.pbm=38;5;13:*.pgm=38;5;13:
*.ppm=38;5;13:*.tga=38;5;13:*.xbm=38;5;13:*.xpm=38;5;13:*.tif=38;5;13:*.tiff=38;5;13:*.png=38;5;13:*.svg=38;5;13:
*.svgz=38;5;13:*.mng=38;5;13:*.pcx=38;5;13:*.mov=38;5;13:*.mpg=38;5;13:*.mpeg=38;5;13:*.m2v=38;5;13:*.mkv=38;5;13
:*.webm=38;5;13:*.ogm=38;5;13:*.mp4=38;5;13:*.m4v=38;5;13:*.mp4v=38;5;13:*.vob=38;5;13:*.qt=38;5;13:*.nuv=38;5;13
:*.wmv=38;5;13:*.asf=38;5;13:*.rm=38;5;13:*.rmvb=38;5;13:*.flc=38;5;13:*.avi=38;5;13:*.fli=38;5;13:*.flv=38;5;13:
*.gl=38;5;13:*.dl=38;5;13:*.xcf=38;5;13:*.xwd=38;5;13:*.yuv=38;5;13:*.cgm=38;5;13:*.emf=38;5;13:*.axv=38;5;13:*.a
nx=38;5;13:*.ogv=38;5;13:*.ogx=38;5;13:*.aac=38;5;45:*.au=38;5;45:*.flac=38;5;45:*.mid=38;5;45:*.midi=38;5;45:*.m
ka=38;5;45:*.mp3=38;5;45:*.mpc=38;5;45:*.ogg=38;5;45:*.ra=38;5;45:*.wav=38;5;45:*.axa=38;5;45:*.oga=38;5;45:*.spx
=38;5;45:*.xspf=38;5;45:
MAIL=/var/spool/mail/trainee
PATH=/usr/local/bin:/usr/bin:/usr/local/sbin:/usr/sbin:/home/trainee/.local/bin:/home/trainee/bin
PWD=/home/trainee
LANG=fr_FR.UTF-8
SELINUX_LEVEL_REQUESTED=
HISTCONTROL=ignoredups
SHLVL=1
HOME=/home/trainee
LOGNAME=trainee
SSH_CONNECTION=10.0.2.2 33896 192.168.1.99 22
LESSOPEN=||/usr/bin/lesspipe.sh %s
XDG_RUNTIME_DIR=/run/user/1000
_=/usr/bin/env
OLDPWD=/home/trainee/training
```

La commande peut aussi être utilisée pour fixer une variable lors de l'exécution d'une commande. Par exemple, pour lancer **xterm** avec la variable EDITOR fixée à **vi** :

```
$ env EDITOR=vim xterm
```

Options du Shell Bash

Pour visualiser les options du shell bash, il convient d'utiliser la commande **set** :

```
$ set -o [Entrée]
```

Par exemple :

```
[trainee@centos7 ~]$ set -o
allexport       off
braceexpand     on
emacs           on
errexit         off
errtrace        off
functrace       off
hashall         on
histexpand      on
history         on
ignoreeof       off
interactive-comments    on
keyword         off
monitor         on
noclobber       off
noexec          off
noglob          off
nolog           off
notify          off
```

```
nounset         off
onecmd          off
physical        off
pipefail        off
posix           off
privileged      off
verbose         off
vi              off
xtrace          off
```

Pour activer une option il convient de nouveau à utiliser la commande **set** :

set -o allexport [Entrée]

Par exemple :

```
[trainee@centos7 ~]$ set -o allexport
[trainee@centos7 ~]$ set -o
allexport       on
braceexpand     on
...
```

Notez que l'option **allexport** a été activée.

Pour désactiver une option, on utilise la commande **set** avec l'option **+o** :

```
$ set +o allexport [Entrée]
```

```
[trainee@centos7 ~]$ set +o allexport
[trainee@centos7 ~]$ set -o
allexport       off
braceexpand     on
...
```

Parmi les options, voici la description des plus intéressantes :

Option	Valeur par Défaut	Description
allexport	off	Le shell export automatiquement toute variable
emacs	on	L'édition de la ligne de commande est au style emacs
history	on	L'historique des commandes est activé
noclobber	off	Les simples re-directions n'écrasent pas le fichier de destination
noglob	off	Désactive l'expansion des caractères génériques
nounset	off	Le shell retourne une erreur lors de l'expansion d'une variable inconnue
verbose	off	Affiche les lignes de commandes saisies
vi	off	L'édition de la ligne de commande est au style vi

Exemples

noclobber

```
[trainee@centos7 ~]$ set -o noclobber
[trainee@centos7 ~]$ pwd > file
-bash: file: cannot overwrite existing file
[trainee@centos7 ~]$ pwd > file
-bash: file: cannot overwrite existing file
[trainee@centos7 ~]$ pwd >| file
[trainee@centos7 ~]$ set +o noclobber
```

Important : Notez que l'option **noclobber** peut être contournée en utilisant la redirection suivi par le caractère **|**.

noglob

```
[trainee@centos7 ~]$ set -o noglob
[trainee@centos7 ~]$ echo *
*
[trainee@centos7 ~]$ set +o noglob
[trainee@centos7 ~]$ echo *
aac abc bca codes Desktop Documents Downloads errorlog file file1 Music Pictures Public Templates training Videos
vitext xyz
```

Important : Notez que l'effet du caractère spécial est annulé sous l'influence de l'option **noglob**.

nounset

```
[trainee@centos7 ~]$ set -o nounset
[trainee@centos7 ~]$ echo $FENESTROS
-bash: FENESTROS: unbound variable
[trainee@centos7 ~]$ set +o nounset
[trainee@centos7 ~]$ echo $FENESTROS

[trainee@centos7 ~]$
```

Important : Notez que la variable inexistante **$FENESTROS** est identifiée comme telle sous l'influence de l'option **nounset**. Or le comportement habituel de Linux est de retourner une ligne vide qui n'indique pas si la variable n’existe pas ou si elle est simplement vide.

Les Scripts Shell

Le but de la suite de cette unité est de vous amener au point où vous êtes capable de comprendre et de déchiffrer les scripts, notamment les scripts de démarrage ainsi que les scripts de contrôle des services.

Écrire des scripts compliqués est en dehors de la portée de cette unité car il nécessite une approche programmation qui ne peut être adressée que lors d'une formation dédiée à l'écriture des scripts.

Exécution

Un script shell est un fichier dont le contenu est lu en entrée standard par le shell. Le contenu du fichier est lu et exécuté d'une manière séquentielle. Afin qu'un script soit exécuté, il suffit qu'il puisse être lu au quel cas le script est exécuté par un shell fils soit en l'appelant en argument à l'appel du shell :

/bin/bash myscript

soit en redirigeant son entrée standard :

/bin/bash < myscript

Dans le cas où le droit d'exécution est positionné sur le fichier script et à condition que celui-ci se trouve dans un répertoire spécifié dans le PATH de l'utilisateur qui le lance, le script peut être lancé en l'appelant simplement par son nom :

myscript

Pour lancer le script sans qu'il soit dans un répertoire du PATH, il convient de se placer dans le répertoire contenant le script et de le lancer ainsi :

./myscript

Dans le cas où le script doit être exécuté par le shell courant, dans les mêmes conditions que l'exemple précédent, et non par un shell fils, il convient de le lancer ainsi :

. myscript

Dans un script il est fortement conseillé d'inclure des commentaires. Les commentaires permettent à d'autres personnes de comprendre le script. Toute ligne de commentaire commence avec le caractère **#**.

Il existe aussi un **pseudo commentaire** qui est placé au début du script. Ce pseudo commentaire permet de stipuler quel shell doit être utilisé pour l'exécution du script. L'exécution du script est ainsi rendu indépendant du shell de l'utilisateur qui le lance. Le pseudo commentaire commence avec les caractères **#!**. Chaque script commence donc par une ligne similaire à celle-ci :

```
#!/bin/sh
```

Puisque un script contient des lignes de commandes qui peuvent être saisies en shell intéractif, il est souvent issu d'une procédure manuelle. Afin de faciliter la création d'un script il existe une commande, **script**, qui permet d'enregistrer les textes sortis sur la sortie standard, y compris le prompt dans un fichier dénommé **typescript**. Afin d'illustrer l'utilisation de cette commande, saisissez la suite de commandes suivante :

```
[trainee@centos7 ~]$ script
Script started, file is typescript
[trainee@centos7 ~]$ pwd
/home/trainee
[trainee@centos7 ~]$ ls
aac  bca    Desktop    Downloads  fichier1  file   Music     Public     training    Videos  xyz
abc  codes  Documents  errorlog   fichier2  file1  Pictures  Templates  typescript  vitext
[trainee@centos7 ~]$ exit
exit
Script done, file is typescript
[trainee@centos7 ~]$ cat typescript
Script started on Tue 29 Nov 2016 03:58:33 CET
[trainee@centos7 ~]$ pwd
/home/trainee
[trainee@centos7 ~]$ ls
aac  bca    Desktop    Downloads  fichier1  file   Music     Public     training    Videos  xyz
abc  codes  Documents  errorlog   fichier2  file1  Pictures  Templates  typescript  vitext
[trainee@centos7 ~]$ exit
exit

Script done on Tue 29 Nov 2016 03:58:40 CET
```

Cette procédure peut être utilisée pour enregistrer une suite de commandes longues et compliquées afin d'écrire un script.

Pour illustrer l'écriture et l'exécution d'un script, éditez le fichier **myscript** avec **vi** :

```
$ vi myscript [Entrée]
```

Éditez votre fichier ainsi :

```
pwd
ls
```

Sauvegardez votre fichier. Lancez ensuite votre script en passant le nom du fichier en argument à /bin/bash :

```
[trainee@centos7 ~]$ vi myscript
[trainee@centos7 ~]$ /bin/bash myscript
/home/trainee
aac  codes  Downloads  fichier2  myscript    Public       typescript  xyz
abc  Desktop    errorlog   file      Music    Templates  Videos
bca  Documents  fichier1   file1     Pictures    training   vitext
```

Lancez ensuite le script en redirigeant son entrée standard :

```
[trainee@centos7 ~]$ /bin/bash < myscript
/home/trainee
aac  codes  Downloads  fichier2  myscript    Public       typescript  xyz
abc  Desktop    errorlog   file      Music    Templates  Videos
bca  Documents  fichier1   file1     Pictures    training   vitext
```

Pour lancer le script en l'appelant simplement par son nom, son chemin doit être inclus dans votre PATH:

```
[trainee@centos7 ~]$ echo $PATH
/usr/local/bin:/usr/bin:/usr/local/sbin:/usr/sbin:/home/trainee/.local/bin:/home/trainee/bin
```

Dans le cas de RHEL/CentOS, même si PATH contient $HOME/bin, le répertoire n'existe pas :

```
[trainee@centos7 ~]$ ls
aac  codes      Downloads  fichier2  myscript  Public     typescript  xyz
abc  Desktop    errorlog   file      Music      Templates  Videos
bca  Documents  fichier1   file1     Pictures   training   vitext
```

Créez donc ce répertoire :

```
[trainee@centos7 ~]$ mkdir bin
```

Ensuite déplacez votre script dans ce répertoire et rendez-le exécutable pour votre utilisateur :

```
[trainee@centos7 ~]$ mv myscript ~/bin
[trainee@centos7 ~]$ chmod u+x ~/bin/myscript
```

Exécutez maintenant votre script en l'appelant par son nom à partir du répertoire **/tmp** :

```
[trainee@centos7 tmp]$ myscript
/tmp
hsperfdata_root  systemd-private-e526abcf335b40949dfc725f28456502-cups.service-u0xGiL
```

Placez-vous dans le répertoire contenant le script et saisissez les commandes suivantes :

- ./myscript
- . myscript

```
[trainee@centos7 tmp]$ cd ~/bin
[trainee@centos7 bin]$ ./myscript
/home/trainee/bin
myscript
[trainee@centos7 bin]$ . myscript
/home/trainee/bin
myscript
```

A faire : Notez bien la différence entre les sorties de cette dernière commande et la précédente. Expliquez pourquoi.

La commande read

La commande **read** lit son entrée standard et affecte les mots saisis dans la ou les variable(s) passée(s) en argument. La séparation entre le contenu des variables est l'espace. Par conséquent il est intéressant de noter les exemples suivants :

```
[trainee@centos7 bin]$ read var1 var2 var3 var4
fenestros edu is great!
[trainee@centos7 bin]$ echo $var1
fenestros
[trainee@centos7 bin]$ echo $var2
edu
[trainee@centos7 bin]$ echo $var3
is
[trainee@centos7 bin]$ echo $var4
great!
```

Important: Notez que chaque champs a été placé dans une variable différente. Notez aussi que par convention les variables déclarées par des utilisateurs sont en miniscules afin de les distinguer des variables système qui sont en majuscules.

```
[trainee@centos7 bin]$ read var1 var2
fenestros edu is great!
[trainee@centos7 bin]$ echo $var1
fenestros
[trainee@centos7 bin]$ echo $var2
```

```
edu is great!
```

Important : Notez que dans le deuxième cas, le reste de la ligne après le mot *fenestros* est mis dans **$var2**.

Code de retour

La commande **read** renvoie un code de retour de **0** dans le cas où elle ne reçoit pas l'information **fin de fichier** matérialisée par les touches Ctrl+D.
Le contenu de la variable **var** peut être vide et la valeur du code de retour **0** grâce à l'utilisation de la touche Entrée :

```
[trainee@centos7 bin]$ read var
```

↵ Entrée

```
[trainee@centos7 bin]$ echo $?
0
[trainee@centos7 bin]$ echo $var

[trainee@centos7 bin]$
```

Le contenu de la variable **var** peut être vide et la valeur du code de retour **autre que 0** grâce à l'utilisation des touches Ctrl+D :

```
[trainee@centos7 bin]$ read var
```

Ctrl+D

```
[trainee@centos7 bin]$ echo $?
1
[trainee@centos7 bin]$ echo $var
```

```
[trainee@centos7 bin]$
```

La variable IFS

La variable IFS contient par défaut les caractères Espace, Tab et Entrée :

```
[trainee@centos7 bin]$ echo "$IFS" | od -c
0000000      \t  \n  \n
0000004
```

> **Important** : La commande **od** (*Octal Dump*) renvoie le contenu d'un fichier ou de l'entrée standard au format octal. Ceci est utile afin de visualiser les caractères non-imprimables. L'option **-c** permet de sélectionner des caractères ASCII ou des backslash dans le fichier ou dans le contenu fourni à l'entrée standard.

La valeur de cette variable définit donc le séparateur de mots lors de la saisie des contenus des variables avec la commande **read**. La valeur de la variable **IFS** peut être modifiée :

```
[trainee@centos7 bin]$ OLDIFS="$IFS"
[trainee@centos7 bin]$ IFS=":"
[trainee@centos7 bin]$ echo "$IFS" | od -c
0000000   :  \n
0000002
```

De cette façon l'espace redevient un caractère normal :

```
[trainee@centos7 bin]$ read var1 var2 var3
fenestros:edu is:great!
[trainee@centos7 bin]$ echo $var1
fenestros
[trainee@centos7 bin]$ echo $var2
```

```
edu is
[trainee@centos7 bin]$ echo $var3
great!
```

Restaurez l'ancienne valeur de IFS avec la commande IFS="$OLDIFS"

```
[trainee@centos7 bin]$ IFS="$OLDIFS"
[trainee@centos7 bin]$ echo "$IFS" | od -c
0000000      \t  \n  \n
0000004
```

La commande test

La commande **test** peut être utilisée avec deux syntaxes :

test *expression*

ou

[Espace*expression*Espace**]**

Tests de Fichiers

Test	Description
-f fichier	Retourne vrai si fichier est d'un type standard
-d fichier	Retourne vrai si fichier est d'un type répertoire
-r fichier	Retourne vrai si l'utilisateur peut lire fichier
-w fichier	Retourne vrai si l'utilisateur peut modifier fichier
-x fichier	Retourne vrai si l'utilisateur peut exécuter fichier
-e fichier	Retourne vrai si fichier existe
-s fichier	Retourne vrai si fichier n'est pas vide
fichier1 -nt fichier2	Retourne vrai si fichier1 est plus récent que fichier2

Test	Description
fichier1 -ot fichier2	Retourne vrai si fichier1 est plus ancien que fichier2
fichier1 -ef fichier2	Retourne vrai si fichier1 est identique à fichier2

Testez si le fichier **a100** est un fichier ordinaire :

```
[trainee@centos7 bin]$ cd ../training/
[trainee@centos7 training]$ test -f a100
[trainee@centos7 training]$ echo $?
0
[trainee@centos7 training]$ [ -f a100 ]
[trainee@centos7 training]$ echo $?
0
```

Testez si le fichier a101 existe :

```
[trainee@centos7 training]$ [ -f a101 ]
[trainee@centos7 training]$ echo $?
1
```

Testez si /home/trainee/training est un répertoire :

```
[trainee@centos7 training]$ [ -d /home/trainee/training ]
[trainee@centos7 training]$ echo $?
0
```

Tests de chaînes de caractère

Test	Description
-n chaîne	Retourne vrai si chaîne n'est pas de longueur 0
-z chaîne	Retourne vrai si chaîne est de longueur 0
string1 = string2	Retourne vrai si string1 est égale à string2

Test	Description
string1 != string2	Retourne vrai si string1 est différente de string2
string1	Retourne vrai si string1 n'est pas vide

Testez si les deux chaînes sont égales :

```
[trainee@centos7 training]$ string1="root"
[trainee@centos7 training]$ string2="fenestros"
[trainee@centos7 training]$ [ $string1 = $string2 ]
[trainee@centos7 training]$ echo $?
1
```

Testez si la string1 n'a pas de longueur 0 :

```
[trainee@centos7 training]$ [ -n $string1 ]
[trainee@centos7 training]$ echo $?
0
```

Testez si la string1 a une longueur de 0 :

```
[trainee@centos7 training]$ [ -z $string1 ]
[trainee@centos7 training]$ echo $?
1
```

Tests sur des nombres

Test	Description
value1 -eq value2	Retourne vrai si value1 est égale à value2
value1 -ne value2	Retourne vrai si value1 n'est pas égale à value2
value1 -lt value2	Retourne vrai si value1 est inférieure à value2
value1 -le value2	Retourne vrai si value1 est inférieur ou égale à value2
value1 -gt value2	Retourne vrai si value1 est supérieure à value2

Test	Description
value1 -ge value2	Retourne vrai si value1 est supérieure ou égale à value2

Comparez les deux nombres **value1** et **value2** :

```
[trainee@centos7 training]$ read value1
35
[trainee@centos7 training]$ read value2
23
[trainee@centos7 training]$ [ $value1 -lt $value2 ]
[trainee@centos7 training]$ echo $?
1
[trainee@centos7 training]$ [ $value2 -lt $value1 ]
[trainee@centos7 training]$ echo $?
0
[trainee@centos7 training]$ [ $value2 -eq $value1 ]
[trainee@centos7 training]$ echo $?
1
```

Les opérateurs

Test	Description
!expression	Retourne vrai si expression est fausse
expression1 -a expression2	Représente un et logique entre expression1 et expression2
expression1 -o expression2	Représente un ou logique entre expression1 et expression2
\(expression\)	Les parenthèses permettent de regrouper des expressions

Testez si $file n'est pas un répertoire :

```
[trainee@centos7 training]$ file=a100
[trainee@centos7 training]$ [ ! -d $file ]
[trainee@centos7 training]$ echo $?
```

```
0
```

Testez si $directory est un répertoire **et** si l'utilisateur à le droit de le traverser :

```
[trainee@centos7 training]$ directory=/usr
[trainee@centos7 training]$ [ -d $directory -a -x $directory ]
[trainee@centos7 training]$ echo $?
0
```

Testez si l'utilisateur peut écrire dans le fichier a100 **et** /usr est un répertoire **ou** /tmp est un répertoire :

```
[trainee@centos7 training]$ [ -w a100 -a \( -d /usr -o -d /tmp \) ]
[trainee@centos7 training]$ echo $?
0
```

Tests d'environnement utilisateur

Test	Description
-o option	Retourne vrai si l'option du shell “option” est activée

```
[trainee@centos7 training]$ [ -o allexport ]
[trainee@centos7 training]$ echo $?
1
```

La commande [[expression]]

La commande **[[Espaceexpression Espace]]** est une amélioration de la commande **test**. Les opérateurs de la commande test sont compatibles avec la commande [[expression]] sauf **-a** et **-o** qui sont remplacés par **&&** et **||** respectivement :

Test	Description
!expression	Retourne vrai si expression est fausse
expression1 && expression2	Représente un **et** logique entre expression1 et expression2

Test	Description
expression1 || expression2	Représente un **ou** logique entre expression1 et expression2
(expression)	Les parenthèses permettent de regrouper des expressions

D'autres opérateurs ont été ajoutés :

Test	Description
string = modele	Retourne vrai si chaîne correspond au modèle
string != modele	Retourne vrai si chaîne ne correspond pas au modèle
string1 < string2	Retourne vrai si string1 est lexicographiquement avant string2
string1 > string2	Retourne vrai si string1 est lexicographiquement après string2

Testez si l'utilisateur peut écrire dans le fichier a100 **et** /usr est un répertoire **ou** /tmp est un répertoire :

```
[trainee@centos7 training]$ [[ -w a100 && ( -d /usr || -d /tmp ) ]]
[trainee@centos7 training]$ echo $?
0
```

Opérateurs du shell

Opérateur	Description
Commande1 && Commande2	Commande 2 est exécutée si la première commande renvoie un code vrai
Commande1 || Commande2	Commande 2 est exécutée si la première commande renvoie un code faux

```
[trainee@centos7 training]$ [[ -d /root ]] && echo "The root directory exists"
The root directory exists
[trainee@centos7 training]$ [[ -d /root ]] || echo "The root directory exists"
[trainee@centos7 training]$
```

L'arithmétique

La commande expr

La commande **expr** prend la forme :

expr Espace value1 Espace *opérateur* Espace value2 Entrée

ou

expr Tab value1 Tab *opérateur* Tab value2 Entrée

ou

expr Espace chaîne Espace : Espace *expression_régulière* Entrée

ou

expr Tab chaîne Tab : Tab *expression_régulière* Entrée

Opérateurs Arithmétiques

Opérateur	Description
+	Addition
-	Soustraction
*	Multiplication
/	Division
%	Modulo
\(\)	Parenthèses

Opérateurs de Comparaison

Opérateur	Description
\<	Inférieur
\<=	Inférieur ou égal

Opérateur	Description
\>	Supérieur
\>=	Supérieur ou égal
=	égal
!=	inégal

Opérateurs Logiques

Opérateur	Description
\\|	ou logique
\&	et logique

Ajoutez 2 à la valeur de $x :

```
[trainee@centos7 training]$ x=2
[trainee@centos7 training]$ expr $x + 2
4
```

Si les espaces sont retirés, le résultat est tout autre :

```
[trainee@centos7 training]$ expr $x+2
2+2
```

Les opérateurs doivent être protégés :

```
[trainee@centos7 training]$ expr $x * 2
expr: syntax error
[trainee@centos7 training]$ expr $x \* 2
4
```

Mettez le résultat d'un calcul dans une variable :

```
[trainee@centos7 training]$ resultat=`expr $x + 10`
```

```
[trainee@centos7 training]$ echo $resultat
12
```

La commande let

La commande let est l'équivalent de la commande ((expression)). La commande ((expression)) est une amélioration de la commande **expr** :

- plus grand nombre d'opérateurs
- pas besoin d'espaces ou de tabulations entre les arguments
- pas besoin de préfixer les variables d'un $
- les caractères spéciaux du shell n'ont pas besoin d'être protégés
- les affectations se font dans la commande
- exécution plus rapide

Opérateurs Arithmétiques

Opérateur	Description
+	Addition
-	Soustraction
*	Multiplication
/	Division
%	Modulo
^	Puissance

Opérateurs de comparaison

Opérateur	Description
<	Inférieur
<=	Inférieur ou égal
>	Supérieur
>=	Supérieur ou égal

Opérateur	Description
==	égal
!=	inégal

Opérateurs Logiques

Opérateur	Description
&&	et logique
||	ou logique
!	négation logique

Opérateurs travaillant sur les bits

Opérateur	Description
~	négation binaire
>>	décalage binaire à droite
<<	décalage binaire à gauche
&	et binaire
|	ou binaire
^	ou exclusif binaire

```
[trainee@centos7 training]$ x=2
[trainee@centos7 training]$ ((x=$x+10))
[trainee@centos7 training]$ echo $x
12
[trainee@centos7 training]$ ((x=$x+20))
[trainee@centos7 training]$ echo $x
32
```

Structures de contrôle

If

La syntaxe de la commande If est la suivante :

```
if condition
then
    commande(s)
else
    commande(s)
fi
```

ou :

```
if condition
then
    commande(s)
    commande(s)
fi
```

ou encore :

```
if condition
then
     commande(s)
elif condition
then
     commande(s)
elif condition
then
     commande(s)
else
    commande(s)
```

```
fi
```

Créez le script **user_check** suivant :

```
#!/bin/bash
if [ $# -ne 1 ] ; then
  echo "Mauvais nombre d'arguments"
  echo "Usage : $0 nom_utilisateur"
  exit 1
fi
if grep "^$1:" /etc/passwd > /dev/null
then
  echo "Utilisateur $1 est défini sur ce système"
else
  echo "Utilisateur $1 n'est pas défini sur ce système"
fi
exit 0
```

Testez-le :

```
[trainee@centos7 training]$ chmod 770 user_check
[trainee@centos7 training]$ ./user_check
Mauvais nombre d'arguments
Usage : ./user_check nom_utilisateur
[trainee@centos7 training]$ ./user_check root
Utilisateur root est défini sur ce système
[trainee@centos7 training]$ ./user_check mickey mouse
Mauvais nombre d'arguments
Usage : ./user_check nom_utilisateur
[trainee@centos7 training]$ ./user_check "mickey mouse"
Utilisateur mickey mouse n'est pas défini sur ce système
```

case

La syntaxe de la commande case est la suivante :

```
case $variable in
modele1) commande
   ...
   ;;
modele2) commande
   ...
   ;;
modele3 | modele4 | modele5 ) commande
   ...
   ;;
esac
```

Exemple

```
  case "$1" in
      start)
          start
          ;;
      stop)
          stop
          ;;
      restart|reload)
          stop
          start
          ;;
      status)
          status
          ;;
```

```
      *)
          echo $"Usage: $0 {start|stop|restart|status}"
          exit 1
esac
```

Important : L'exemple indique que dans le cas où le premier argument qui suit le nom du script contenant la clause **case** est **start**, la fonction *start* sera exécutée. La fonction *start* n'a pas besoin d'être définie dans **case** et est donc en règle générale définie en début de script. La même logique est appliquée dans le cas où le premier argument est **stop**, **restart** ou **reload** et **status**. Dans tous les autres cas, représentés par une étoile, **case** affichera la ligne **Usage: $0 {start|stop|restart|status}** où $0 est remplacé par le nom du script.

Boucles

for

La syntaxe de la commande for est la suivante :

```
for variable in liste_variables
do
    commande(s)
done
```

while

La syntaxe de la commande while est la suivante :

```
while condition
do
    commande(s)
```

```
done
```

Exemple

```
U=1
while [ $U -lt $MAX_ACCOUNTS ]
do
useradd fenestros"$U" -c fenestros"$U" -d /home/fenestros"$U" -g staff -G audio,fuse -s /bin/bash 2>/dev/null
useradd fenestros"$U"$ -g machines -s /dev/false -d /dev/null 2>/dev/null
echo "Compte fenestros$U créé"
let U=U+1
done
```

Scripts de Démarrage

Quand Bash est appelé en tant que shell de connexion, il exécute des scripts de démarrage dans l'ordre suivant :

- **/etc/profile**,
- **~/.bash_profile** ou **~/.bash_login** ou **~/.profile** selon la distribution,

Dans le cas de RHEL/CentOS, le système exécute le fichier **~/.bash_profile**.

Quand un shell de login se termine, Bash exécute le fichier **~/.bash_logout** si celui-ci existe.

Quand bash est appelé en tant que shell interactif qui n'est pas un shell de connexion, il exécute le script **~/.bashrc**.

LAB #1- Scripts de Démarrage

A faire : En utilisant vos connaissances acquises dans ce module, expliquez les scripts suivants ligne par ligne.

~/.bash_profile

```
[trainee@centos7 training]$ cat ~/.bash_profile
# .bash_profile

# Get the aliases and functions
if [ -f ~/.bashrc ]; then
    . ~/.bashrc
fi

# User specific environment and startup programs

PATH=$PATH:$HOME/.local/bin:$HOME/bin

export PATH
```

~/.bashrc

```
[trainee@centos7 training]$ cat ~/.bashrc
# .bashrc

# Source global definitions
if [ -f /etc/bashrc ]; then
    . /etc/bashrc
fi

# Uncomment the following line if you don't like systemctl's auto-paging feature:
# export SYSTEMD_PAGER=

# User specific aliases and functions
```

Gestion des Processus

Un processus est un fichier binaire (binary file) qui est chargé en mémoire centrale. Une fois chargé la mémoire exécute le programme en langage machine. Quand le programme est chargé, il a besoin du système d’exploitation qui lui fournit des informations pour qu’il puisse s’exécuter correctement. Ces informations sont appelées des **données d’identification**.

L’ensemble des **données d’identification** est appelé l’**environnement de processus** :

- Un numéro de processus unique (PID),
- Un numéro de processus parent (PPID),
- Un numéro d'utilisateur (UID),
- Un numéro de groupe (GID),
- La durée de traitement,
- La priorité du processus,
- Le répertoire de travail actif,
- Les fichiers ouverts.

Ces informations sont stockés dans le répertoire **/proc**. Le répertoire /proc contient des fichiers et des répertoires virtuels. Le contenu de ces fichiers est créé dynamiquement lors de la consultation. Seul root peut consulter la totalité des informations dans le répertoire /proc.

Saisissez la commande suivante :

```
[root@centos7 ~]# cd /proc; ls -d [0-9]*
1     1347  20   2421  27    3005  3132  3249  3319  3697  407   436  495  571
10    14    21   246   2740  3074  3144  3252  3359  370   4070  454  498  596
1015  1410  218  2481  28    3076  3160  3256  3395  373   4071  458  5    6
11    15    22   25    2849  3085  3196  3266  3408  3756  4077  466  502  60
12    16    220  258   2890  3088  3201  3273  3418  3759  41    478  505  625
1217  1677  221  259   29    3090  3202  3274  3428  3760  4111  482  506  7
1219  1687  222  26    2902  3092  3219  3285  3434  38    4112  484  509  700
1228  1688  223  260   2923  3094  3224  3289  3436  381   4113  485  520  8
1229  17    224  261   2931  3095  3228  3291  3460  3873  431   488  521  83
1231  18    227  262   2932  3108  3231  3304  3477  3883  432   489  523  9
```

```
13    19    229  263   2997  3124  3239  3309  362   40    433   491  525
1327  2     24   264   3     3128  3243  3311  3639  4007  434   492  552
```

Chaque répertoire fait référence à un PID d'un processus. Les données de l'**environnement de processus** y sont présentes, par exemple :

```
[root@centos7 proc]# cd 1 ; ls -l
total 0
dr-xr-xr-x. 2 root root 0 Oct 27 16:20 attr
-rw-r--r--. 1 root root 0 Oct 27 16:20 autogroup
-r--------. 1 root root 0 Oct 27 16:20 auxv
-r--r--r--. 1 root root 0 Oct 27 16:06 cgroup
--w-------. 1 root root 0 Oct 27 16:20 clear_refs
-r--r--r--. 1 root root 0 Oct 27 16:06 cmdline
-rw-r--r--. 1 root root 0 Oct 27 16:06 comm
-rw-r--r--. 1 root root 0 Oct 27 16:20 coredump_filter
-r--r--r--. 1 root root 0 Oct 27 16:20 cpuset
lrwxrwxrwx. 1 root root 0 Oct 27 16:20 cwd -> /
-r--------. 1 root root 0 Oct 27 16:06 environ
lrwxrwxrwx. 1 root root 0 Oct 27 16:06 exe -> /usr/lib/systemd/systemd
dr-x------. 2 root root 0 Oct 27 16:06 fd
dr-x------. 2 root root 0 Oct 27 16:20 fdinfo
-rw-r--r--. 1 root root 0 Oct 27 16:20 gid_map
-r--------. 1 root root 0 Oct 27 16:20 io
-r--r--r--. 1 root root 0 Oct 27 16:20 limits
-rw-r--r--. 1 root root 0 Oct 27 16:06 loginuid
-r--r--r--. 1 root root 0 Oct 27 16:06 maps
-rw-------. 1 root root 0 Oct 27 16:20 mem
-r--r--r--. 1 root root 0 Oct 27 16:06 mountinfo
-r--r--r--. 1 root root 0 Oct 27 16:06 mounts
-r--------. 1 root root 0 Oct 27 16:20 mountstats
dr-xr-xr-x. 6 root root 0 Oct 27 16:06 net
dr-x--x--x. 2 root root 0 Oct 27 16:20 ns
-r--r--r--. 1 root root 0 Oct 27 16:20 numa_maps
-rw-r--r--. 1 root root 0 Oct 27 16:20 oom_adj
```

```
-r--r--r--. 1 root root 0 Oct 27 16:20 oom_score
-rw-r--r--. 1 root root 0 Oct 27 16:20 oom_score_adj
-r--r--r--. 1 root root 0 Oct 27 16:20 pagemap
-r--r--r--. 1 root root 0 Oct 27 16:20 personality
-rw-r--r--. 1 root root 0 Oct 27 16:20 projid_map
lrwxrwxrwx. 1 root root 0 Oct 27 16:06 root -> /
-rw-r--r--. 1 root root 0 Oct 27 16:20 sched
-r--r--r--. 1 root root 0 Oct 27 16:06 sessionid
-r--r--r--. 1 root root 0 Oct 27 16:20 smaps
-r--r--r--. 1 root root 0 Oct 27 16:20 stack
-r--r--r--. 1 root root 0 Oct 27 16:06 stat
-r--r--r--. 1 root root 0 Oct 27 16:20 statm
-r--r--r--. 1 root root 0 Oct 27 16:06 status
-r--r--r--. 1 root root 0 Oct 27 16:20 syscall
dr-xr-xr-x. 3 root root 0 Oct 27 16:06 task
-rw-r--r--. 1 root root 0 Oct 27 16:20 uid_map
-r--r--r--. 1 root root 0 Oct 27 16:20 wchan
```

Important - Vous n'avez pas besoin de consulter le contenu des fichiers et des répertoires. Il convient tout simplement de savoir que ces données existent. Naviguez donc à /root en ligne de commande.

Les Types de Processus

Il existe trois types de processus :

- **interactif** qui est lancé par le shell dans une console en premier plan ou en tâche de fond
- **batch** qui est lancé par le système au moment propice
- **daemon** qui est lancé au démarrage par le système (lpd, dns etc)

Un processus peut être dans un de neuf états ou *process states* :

- *user mode* - le processus s'exécute en mode utilisateur,
- *kernel mode*- le processus s'exécute en mode noyau,
- *waiting* – le processus est en attente pour une ressource autre que le processeur,
- *sleeping* - le processus est endormi,
- *runnable* – le processus dispose de toutes le ressources nécessaire à son exécution sauf le processeur,
- *swap* - le processus est endormi dans la mémoire virtuelle,
- *new* - le processus est nouveau,
- *elected* – le processus a le contrôle du processeur,
- *zombie* – le processus a terminé son exécution et est prêt à mourir.

Les Commandes relatives aux Processus

La Commande ps

Cette commande affiche les processus de l’utilisateur attaché au terminal :

```
[root@centos7 1]# cd ~
[root@centos7 ~]# ps
  PID TTY          TIME CMD
 4070 pts/0    00:00:00 su
 4077 pts/0    00:00:00 bash
 4285 pts/0    00:00:00 ps
```

Pour plus de détails, il convient d'utiliser l'option **-l** :

```
[root@centos7 ~]# ps -l
F S   UID   PID  PPID  C PRI  NI ADDR SZ WCHAN  TTY          TIME CMD
4 S     0  4070  3760  0  80   0 - 50611 wait   pts/0    00:00:00 su
4 S     0  4077  4070  0  80   0 - 29027 wait   pts/0    00:00:00 bash
0 R     0  4309  4077  0  80   0 - 30319 -      pts/0    00:00:00 ps
```

On note dans cette sortie :

F	Drapeaux du processus. La valeur 4 indique que le processus utilise les privilèges de root
S	État du processus S (sleeping), R (In run queue), Z (zombie), N (low priority), D (uninterruptible sleep), T (Traced)
UID	Numéro de l'Utilisateur
PID	Numéro Unique de Processus
PPID	PID du processus parent
C	Facteur de priorité du processus
PRI	Priorité du processus
NI	La valeur de nice
ADDR	Adresse mémoire du processus
SZ	Utilisation de la mémoire virtuelle
WCHAN	Nom de la fonction du noyau dans laquelle le processus est endormi
TTY	Nom du terminal depuis lequel le processus a été lancé
TIME	Durée d'exécution du processus
CMD	Commande exécutée

Pour visualiser la table des processus, utilisez la commande ps avec les options l et x - la commande affiche tous les processus avec un affichage long :

```
[root@centos7 ~]# ps lx | more
F   UID   PID  PPID PRI  NI    VSZ   RSS WCHAN  STAT TTY        TIME COMMAND
4     0     1     0  20   0  59600  7028 ep_pol Ss   ?          0:03 /usr/lib/systemd/systemd --switched-root
 --system --deserialize 24
1     0     2     0  20   0      0     0 kthrea S    ?          0:00 [kthreadd]
1     0     3     2  20   0      0     0 smpboo S    ?          0:01 [ksoftirqd/0]
1     0     5     2   0 -20      0     0 worker S<   ?          0:00 [kworker/0:0H]
1     0     6     2  20   0      0     0 worker S    ?          0:00 [kworker/u2:0]
1     0     7     2 -100  -      0     0 smpboo S    ?          0:00 [migration/0]
1     0     8     2  20   0      0     0 rcu_gp S    ?          0:00 [rcu_bh]
1     0     9     2  20   0      0     0 rcu_no S    ?          0:00 [rcuob/0]
1     0    10     2  20   0      0     0 rcu_gp S    ?          0:01 [rcu_sched]
1     0    11     2  20   0      0     0 rcu_no S    ?          0:02 [rcuos/0]
5     0    12     2 -100  -      0     0 smpboo S    ?          0:00 [watchdog/0]
```

```
1     0    13     2   0 -20      0     0 rescue S<   ?          0:00 [khelper]
5     0    14     2  20   0      0     0 devtmp S    ?          0:00 [kdevtmpfs]
1     0    15     2   0 -20      0     0 rescue S<   ?          0:00 [netns]
1     0    16     2   0 -20      0     0 rescue S<   ?          0:00 [writeback]
1     0    17     2   0 -20      0     0 rescue S<   ?          0:00 [kintegrityd]
1     0    18     2   0 -20      0     0 rescue S<   ?          0:00 [bioset]
1     0    19     2   0 -20      0     0 rescue S<   ?          0:00 [kblockd]
1     0    20     2  20   0      0     0 hub_th S    ?          0:00 [khubd]
1     0    21     2   0 -20      0     0 rescue S<   ?          0:00 [md]
1     0    24     2  20   0      0     0 watchd S    ?          0:00 [khungtaskd]
--More--
```

On note dans cette sortie certaines informations supplémentaires :

VSZ	La même chose que SZ dans l'exemple ci-dessus
RSS	La mémoire utilisée en kilobytes par le processus
STAT	La même chose que S dans l'exemple ci-dessus

Avec des options a,u et x la commande affiche le résultat suivant :

```
[root@centos7 ~]# ps aux | more
USER       PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1  0.3  0.3  59600  7028 ?        Ss   16:06   0:03 /usr/lib/systemd/systemd --switched-root --s
ystem --deserialize 24
root         2  0.0  0.0      0     0 ?        S    16:06   0:00 [kthreadd]
root         3  0.1  0.0      0     0 ?        S    16:06   0:01 [ksoftirqd/0]
root         5  0.0  0.0      0     0 ?        S<   16:06   0:00 [kworker/0:0H]
root         6  0.0  0.0      0     0 ?        S    16:06   0:00 [kworker/u2:0]
root         7  0.0  0.0      0     0 ?        S    16:06   0:00 [migration/0]
root         8  0.0  0.0      0     0 ?        S    16:06   0:00 [rcu_bh]
root         9  0.0  0.0      0     0 ?        S    16:06   0:00 [rcuob/0]
root        10  0.1  0.0      0     0 ?        S    16:06   0:01 [rcu_sched]
root        11  0.2  0.0      0     0 ?        R    16:06   0:02 [rcuos/0]
root        12  0.0  0.0      0     0 ?        S    16:06   0:00 [watchdog/0]
```

```
root        13  0.0  0.0      0     0 ?        S<   16:06   0:00 [khelper]
root        14  0.0  0.0      0     0 ?        S    16:06   0:00 [kdevtmpfs]
root        15  0.0  0.0      0     0 ?        S<   16:06   0:00 [netns]
root        16  0.0  0.0      0     0 ?        S<   16:06   0:00 [writeback]
root        17  0.0  0.0      0     0 ?        S<   16:06   0:00 [kintegrityd]
root        18  0.0  0.0      0     0 ?        S<   16:06   0:00 [bioset]
root        19  0.0  0.0      0     0 ?        S<   16:06   0:00 [kblockd]
root        20  0.0  0.0      0     0 ?        S    16:06   0:00 [khubd]
root        21  0.0  0.0      0     0 ?        S<   16:06   0:00 [md]
root        24  0.0  0.0      0     0 ?        S    16:06   0:00 [khungtaskd]
--More--
```

On note dans cette sortie certaines informations supplémentaires :

USER	L'utilisateur du processus
%CPU	Ressources du microprocesseur utilisées par le processus
%MEM	Ressources en mémoire vive utilisées par le processus

Options de la commande ps

Les options de cette commande sont :

```
[root@centos7 ~]# ps --help

Usage:
 ps [options]

 Try 'ps --help <simple|list|output|threads|misc|all>'
  or 'ps --help <s|l|o|t|m|a>'
 for additional help text.

For more details see ps(1).
```

La Commande pgrep

La commande **pgrep** permet de rechercher un processus en fonction de son nom et d’autres propriétés puis d'afficher son PID sur la sortie standard.

Par exemple, la commande suivante affiche le PID du processus sshd appartenant à root :

```
[root@centos7 ~]# pgrep -u root sshd
1219
```

Tandis que la commande suivante affiche tous les PID des processus appartenant à root ou à trainee :

```
[root@centos7 ~]# pgrep -u root,trainee | more
1
2
3
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
24
```

```
25
26
--More--
```

Option de la commande pgrep

```
[root@centos7 ~]# pgrep --help

Usage:
 pgrep [options] <pattern>

Options:
 -d, --delimiter <string>  specify output delimiter
 -l, --list-name           list PID and process name
 -a, --list-full           list PID and full command line
 -v, --inverse             negates the matching
 -w, --lightweight         list all TID
 -c, --count               count of matching processes
 -f, --full                use full process name to match
 -g, --pgroup <PGID,...>   match listed process group IDs
 -G, --group <GID,...>     match real group IDs
 -n, --newest              select most recently started
 -o, --oldest              select least recently started
 -P, --parent <PPID,...>   match only child processes of the given parent
 -s, --session <SID,...>   match session IDs
 -t, --terminal <tty,...>  match by controlling terminal
 -u, --euid <ID,...>       match by effective IDs
 -U, --uid <ID,...>        match by real IDs
 -x, --exact               match exactly with the command name
 -F, --pidfile <file>      read PIDs from file
 -L, --logpidfile          fail if PID file is not locked
 --ns <PID>                match the processes that belong to the same
                           namespace as <pid>
```

```
 --nslist <ns,...>         list which namespaces will be considered for
                           the --ns option.
                           Available namespaces: ipc, mnt, net, pid, user, uts

 -h, --help     display this help and exit
 -V, --version  output version information and exit

For more details see pgrep(1).
```

La Commande pstree

Cette commande affiche les processus en forme d'arborescence, démontrant ainsi les processus parents en enfants :

```
[root@centos7 ~]# pstree
systemd─┬─ModemManager───2*[{ModemManager}]
        ├─NetworkManager─┬─dhclient
        │                └─3*[{NetworkManager}]
        ├─3*[VBoxClient───VBoxClient───{VBoxClient}]
        ├─VBoxClient───VBoxClient───2*[{VBoxClient}]
        ├─VBoxService───7*[{VBoxService}]
        ├─2*[abrt-watch-log]
        ├─abrtd
        ├─accounts-daemon───2*[{accounts-daemon}]
        ├─alsactl
        ├─at-spi-bus-laun─┬─dbus-daemon───{dbus-daemon}
        │                 └─3*[{at-spi-bus-laun}]
        ├─at-spi2-registr───{at-spi2-registr}
        ├─atd
        ├─auditd─┬─audispd─┬─sedispatch
        │        │         └─{audispd}
        │        └─{auditd}
        ├─avahi-daemon───avahi-daemon
        ├─bluetoothd
```

```
        ├─chronyd
        ├─colord───2*[{colord}]
        ├─crond
        ├─cupsd
        ├─2*[dbus-daemon───{dbus-daemon}]
        ├─dbus-launch
        ├─dconf-service───2*[{dconf-service}]
        ├─evolution-addre───4*[{evolution-addre}]
        ├─evolution-calen───4*[{evolution-calen}]
        ├─evolution-sourc───2*[{evolution-sourc}]
        ├─firewalld───{firewalld}
        ├─gconfd-2
        ├─gdm─┬─gdm-simple-slav─┬─Xorg
        │     │                 ├─gdm-session-wor─┬─gnome-session─┬─abrt-applet───{abrt-applet}
        │     │                 │                 │               ├─gnome-settings-───4*[{gnome-settings-}]
        │     │                 │                 │               ├─gnome-shell─┬─firefox───42*[{firefox}]
        │     │                 │                 │               │             ├─gedit───4*[{gedit}]
        │     │                 │                 │               │             └─5*[{gnome-shell}]
        │     │                 │                 │               ├─seapplet
        │     │                 │                 │               ├─ssh-agent
        │     │                 │                 │               ├─tracker-miner-f───4*[{tracker-miner-f}]
        │     │                 │                 │               └─3*[{gnome-session}]
        │     │                 │                 └─2*[{gdm-session-wor}]
        │     │                 └─2*[{gdm-simple-slav}]
        │     └─2*[{gdm}]
        ├─gnome-keyring-d───4*[{gnome-keyring-d}]
        ├─gnome-shell-cal───4*[{gnome-shell-cal}]
        ├─gnome-terminal-─┬─bash───su───bash───pstree
        │                 ├─gnome-pty-helpe
        │                 └─3*[{gnome-terminal-}]
...
```

Options de la commande pstree

```
[root@centos7 ~]# pstree --help
pstree: unrecognized option '--help'
Usage: pstree [ -a ] [ -c ] [ -h | -H PID ] [ -l ] [ -n ] [ -p ] [ -g ] [ -u ]
              [ -A | -G | -U ] [ PID | USER ]
       pstree -V
Display a tree of processes.

  -a, --arguments     show command line arguments
  -A, --ascii         use ASCII line drawing characters
  -c, --compact       don't compact identical subtrees
  -h, --highlight-all highlight current process and its ancestors
  -H PID,
  --highlight-pid=PID highlight this process and its ancestors
  -g, --show-pgids    show process group ids; implies -c
  -G, --vt100         use VT100 line drawing characters
  -l, --long          don't truncate long lines
  -n, --numeric-sort  sort output by PID
  -N type,
  --ns-sort=type      sort by namespace type (ipc, mnt, net, pid, user, uts)
  -p, --show-pids     show PIDs; implies -c
  -s, --show-parents  show parents of the selected process
  -S, --ns-changes    show namespace transitions
  -u, --uid-changes   show uid transitions
  -U, --unicode       use UTF-8 (Unicode) line drawing characters
  -V, --version       display version information
  -Z,
  --security-context   show SELinux security contexts
  PID    start at this PID; default is 1 (init)
  USER   show only trees rooted at processes of this user
```

La Commande top

Cette commande indique les processus en mémoire :

```
[root@centos7 ~]# top

top - 16:28:28 up 21 min,  2 users,  load average: 1.50, 1.21, 0.86
Tasks: 160 total,   3 running, 157 sleeping,   0 stopped,   0 zombie
%Cpu(s):  8.3 us,  2.7 sy,  0.0 ni, 89.0 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
KiB Mem :  1791624 total,   114728 free,   697212 used,   979684 buff/cache
KiB Swap:  3071996 total,  3071996 free,        0 used.   921100 avail Mem

  PID USER      PR  NI    VIRT    RES    SHR S %CPU %MEM     TIME+ COMMAND
 3249 trainee   20   0 1595324 235224  40572 S 13.6 13.1   3:43.96 gnome-shell
 1410 root      20   0  336640  50172   8980 R  4.4  2.8   1:15.09 Xorg
 3756 trainee   20   0  626148  19044  12284 S  0.9  1.1   0:06.53 gnome-terminal-
 3883 trainee   20   0 1049588 208252  49496 S  0.9 11.6   0:20.22 firefox
 4904 root      20   0  130024   1780   1240 R  0.6  0.1   0:00.09 top
    3 root      20   0       0      0      0 S  0.3  0.0   0:01.18 ksoftirqd/0
  525 dbus      20   0   38480   3104   1444 S  0.3  0.2   0:01.53 dbus-daemon
  596 root      20   0  338392   1072    756 S  0.3  0.1   0:01.03 VBoxService
    1 root      20   0   59600   7028   3968 S  0.0  0.4   0:03.35 systemd
    2 root      20   0       0      0      0 S  0.0  0.0   0:00.00 kthreadd
    5 root       0 -20       0      0      0 S  0.0  0.0   0:00.00 kworker/0:0H
    6 root      20   0       0      0      0 S  0.0  0.0   0:00.17 kworker/u2:0
    7 root      rt   0       0      0      0 S  0.0  0.0   0:00.00 migration/0
    8 root      20   0       0      0      0 S  0.0  0.0   0:00.00 rcu_bh
    9 root      20   0       0      0      0 S  0.0  0.0   0:00.00 rcuob/0
   10 root      20   0       0      0      0 S  0.0  0.0   0:01.38 rcu_sched
   11 root      20   0       0      0      0 R  0.0  0.0   0:02.55 rcuos/0
...
```

Pour afficher l'aide de la commande **top**, appuyez sur la touche **h** :

```
Help for Interactive Commands - procps-ng version 3.3.10
Window 1:Def: Cumulative mode Off.  System: Delay 20.0 secs; Secure mode Off.

  Z,B,E,e   Global: 'Z' colors; 'B' bold; 'E'/'e' summary/task memory scale
  l,t,m     Toggle Summary: 'l' load avg; 't' task/cpu stats; 'm' memory info
  0,1,2,3,I Toggle: '0' zeros; '1/2/3' cpus or numa node views; 'I' Irix mode
  f,F,X     Fields: 'f'/'F' add/remove/order/sort; 'X' increase fixed-width

  L,&,<,> . Locate: 'L'/'&' find/again; Move sort column: '<'/'>' left/right
  R,H,V,J . Toggle: 'R' Sort; 'H' Threads; 'V' Forest view; 'J' Num justify
  c,i,S,j . Toggle: 'c' Cmd name/line; 'i' Idle; 'S' Time; 'j' Str justify
  x,y     . Toggle highlights: 'x' sort field; 'y' running tasks
  z,b     . Toggle: 'z' color/mono; 'b' bold/reverse (only if 'x' or 'y')
  u,U,o,O . Filter by: 'u'/'U' effective/any user; 'o'/'O' other criteria
  n,#,^O  . Set: 'n'/'#' max tasks displayed; Show: Ctrl+'O' other filter(s)
  C,...   . Toggle scroll coordinates msg for: up,down,left,right,home,end

  k,r       Manipulate tasks: 'k' kill; 'r' renice
  d or s    Set update interval
  W,Y       Write configuration file 'W'; Inspect other output 'Y'
  q         Quit
          ( commands shown with '.' require a visible task display window )
Press 'h' or '?' for help with Windows,
Type 'q' or <Esc> to continue
```

Important - Pour revenir à l'affichage précédent, appuyez sur la touche **q** ou **echap**.

Au lancement, le temps de rafraîchissement de la liste est de 3 secondes. Pour modifier ce temps à 1 seconde, appuyez sur la touche **s** puis la touche 1 et validez :

```
[root@centos7 ~]# top
```

```
...
top - 16:31:05 up 24 min,  2 users,  load average: 0.90, 1.10, 0.87
Tasks: 161 total,   3 running, 158 sleeping,   0 stopped,   0 zombie
%Cpu(s): 11.3 us,  7.2 sy,  0.0 ni, 81.5 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
KiB Mem :  1791624 total,   113084 free,   698800 used,   979740 buff/cache
KiB Swap:  3071996 total,  3071996 free,        0 used.   919492 avail Mem
Change delay from 3.0 to 1
...
```

Pour trier la liste selon l'utilisation de la mémoire, appuyez sur la touche **M** :

```
[root@centos7 ~]# top
...
top - 16:32:41 up 26 min,  2 users,  load average: 0.65, 0.96, 0.84
Tasks: 161 total,   2 running, 159 sleeping,   0 stopped,   0 zombie
%Cpu(s):  2.9 us,  1.0 sy,  0.0 ni, 96.1 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
KiB Mem :  1791624 total,   114468 free,   697428 used,   979728 buff/cache
KiB Swap:  3071996 total,  3071996 free,        0 used.   920864 avail Mem

  PID USER      PR  NI    VIRT    RES    SHR S %CPU %MEM     TIME+ COMMAND
 3249 trainee   20   0 1597372 237248  40572 S  4.1 13.2   5:10.83 gnome-shell
 3883 trainee   20   0 1049588 207680  49496 S  0.0 11.6   0:20.28 firefox
 1410 root      20   0  336640  50172   8980 S  1.8  2.8   1:41.56 Xorg
 3873 trainee   20   0 1023212  30228  15840 S  0.0  1.7   0:21.83 gedit
 3359 trainee   20   0 1054704  24780  15860 S  0.0  1.4   0:01.06 nautilus
 3144 trainee   20   0 1000480  23484  14756 S  0.0  1.3   0:01.54 gnome-settings-
  482 root      20   0  329596  23200   6252 S  0.0  1.3   0:01.38 firewalld
 3756 trainee   20   0  626148  19044  12284 S  0.5  1.1   0:07.65 gnome-terminal-
 3231 trainee   20   0  687500  18164  12164 S  0.0  1.0   0:00.09 goa-daemon
  506 root      20   0  550176  16092   5612 S  0.0  0.9   0:01.02 tuned
 3418 trainee   20   0  564964  15820   4776 S  0.0  0.9   0:00.98 tracker-store
 1015 root      20   0  103740  15652   3244 S  0.0  0.9   0:00.06 dhclient
  625 polkitd   20   0  518564  14932   4788 S  0.0  0.8   0:10.60 polkitd
 3477 trainee   20   0  450108  13964   9392 S  0.0  0.8   0:00.33 abrt-applet
```

```
 1217 root      20   0  477048  13688   8796 S  0.0  0.8   0:00.37 libvirtd
 3395 trainee   20   0  894520  13624   8476 S  0.0  0.8   0:00.38 evolution-calen
 3460 trainee   39  19  677116  12672   7388 S  0.0  0.7   0:00.32 tracker-miner-f
```

Pour ne pas visualiser les processus zombies ou les processus en attente, appuyez sur la touche i :

```
[root@centos7 ~]# top
...
top - 16:33:45 up 27 min,  2 users,  load average: 0.72, 0.94, 0.85
Tasks: 160 total,   3 running, 157 sleeping,   0 stopped,   0 zombie
%Cpu(s): 41.2 us,  7.2 sy,  0.0 ni, 51.3 id,  0.0 wa,  0.0 hi,  0.4 si,  0.0 st
KiB Mem :  1791624 total,   109348 free,   702524 used,   979752 buff/cache
KiB Swap:  3071996 total,  3071996 free,        0 used.   915768 avail Mem

  PID USER      PR  NI    VIRT    RES    SHR S %CPU %MEM     TIME+ COMMAND
 3249 trainee   20   0 1597372 237248  40572 S 51.3 13.2   5:31.72 gnome-shell
 1410 root      20   0  347244  57824   8980 R 11.5  3.2   1:48.83 Xorg
 3873 trainee   20   0 1023212  30228  15840 S  1.6  1.7   0:24.10 gedit
 3756 trainee   20   0  626292  19044  12284 S  0.5  1.1   0:08.03 gnome-terminal-
 4904 root      20   0  130024   1780   1240 R  0.2  0.1   0:00.30 top
  596 root      20   0  338392   1072    756 S  0.2  0.1   0:01.28 VBoxService
 5080 root      20   0       0      0      0 S  0.2  0.0   0:00.23 kworker/0:1
```

Pour quitter top, appuyez sur la touche **q**.

Options de la commande top

```
[root@centos7 ~]# top --help
top: inappropriate '-help'
Usage:
  top -hv | -bcHiOSs -d secs -n max -u|U user -p pid(s) -o field -w [cols]
```

Les Commandes fg, bg et jobs

Normalement les commandes s'exécutent en avant plan. Vous pouvez également lancer des processus en arrière plan (en tâche de fond). Si vous lancez une commande en tâche de fond, il faut rajouter **(espace)&** à la fin de la commande :

```
# sleep 9999 &
```

> Notez qu'un processus en arrière plan est dit **asynchrone** car il se poursuit indépendamment de son parent qui est le shell. En avant plan le processus est dit **synchrone**.

Linux numérote tous les processus qui sont placés en tâches de fond. On parle donc d'un **numéro de tâche**.

La commande **jobs** permet de se renseigner sur les processus en arrière plan :

```
[root@centos7 ~]# sleep 9999 &
[1] 9870
[root@centos7 ~]# jobs -l
[1]+  9870 Running                 sleep 9999 &
```

> **Important** - Notez que le numéro de tâche est indiqué entre [crochets] tandis que le PID ne l'est pas. Le signe **+** qui suit le numéro de tâche [1] indique que la tâche est la dernière a avoir été manipulée.

Si on souhaite envoyer un processus en arrière plan de façon à libérer le shell pour d'autres commandes, il faut d'abord suspendre le processus en question. Normalement on suspend un processus en utilisant la combinaison de touches CtrlZ.

Par exemple :

```
[root@centos7 ~]# sleep 1234
```

```
^Z
[2]+  Stopped                 sleep 1234
```

Un fois suspendu, on utilise la commande **bg** (background) suivi par **%** et le numéro de tâche pour envoyer le processus en arrière plan :

```
[root@centos7 ~]# bg %2
[2]+ sleep 1234 &
[root@centos7 ~]# jobs -l
[1]-  9870 Running                 sleep 9999 &
[2]+  9930 Running                 sleep 1234 &
```

Important - Notez que lors du passage en arrière plan, le processus reprend son exécution normalement. Le caractère **-** qui suit le numéro de tâche [1] indique que la tâche est l'avant-dernière a avoir été manipulée.

Pour ramener le processus en avant plan, il faut de nouveau interrompre le processus concerné. Or cette fois-ci, nous ne pouvons pas utiliser la commande CtrlZ. Il faut donc envoyer un **signal** au processus en utilisant la commande **kill** avec l'opérateur **–stop**.

```
[root@centos7 ~]# kill -stop %2
[root@centos7 ~]# jobs -l
[1]-  9870 Running                 sleep 9999 &
[2]+  9930 Stopped (signal)        sleep 1234
```

Pour reprendre le processus en arrière plan, sans le ramener en avant plan, on utilise la commande kill avec l'option **-cont** :

```
[root@centos7 ~]# kill -cont %2
[root@centos7 ~]# jobs -l
[1]-  9870 Running                 sleep 9999 &
[2]+  9930 Running                 sleep 1234 &
```

Pour ramener le processus en avant plan, on utilise la commande fg :

```
[root@centos7 ~]# kill -stop %2
[root@centos7 ~]# jobs -l
[1]-  9870 Running                 sleep 9999 &
[2]+  9930 Stopped (signal)        sleep 1234
[root@centos7 ~]# fg %2
sleep 1234
^C
[root@centos7 ~]#
```

Important - Notez l'utilisation des touches CtrlC pour tuer le processus en avant plan.

Options de la commande jobs

Les options de la commande jobs sont :

```
[root@centos7 ~]# help jobs
jobs: jobs [-lnprs] [jobspec ...] or jobs -x command [args]
    Display status of jobs.
    Lists the active jobs.  JOBSPEC restricts output to that job.
    Without options, the status of all active jobs is displayed.
    Options:
      -l    lists process IDs in addition to the normal information
      -n    list only processes that have changed status since the last
        notification
      -p    lists process IDs only
      -r    restrict output to running jobs
      -s    restrict output to stopped jobs
    If -x is supplied, COMMAND is run after all job specifications that
    appear in ARGS have been replaced with the process ID of that job's
```

```
    process group leader.
    Exit Status:
    Returns success unless an invalid option is given or an error occurs.
    If -x is used, returns the exit status of COMMAND.
```

La Commande wait

Cette commande permet de doter un processus asynchrone du comportement d'un processus synchrone. Elle est utilisée pour attendre jusqu'à ce qu'un processus en tâche de fond soit terminé :

```
[root@centos7 ~]# jobs -l
[1]+  9870 Running                 sleep 9999 &
[root@centos7 ~]# wait %1
^C
[root@centos7 ~]# jobs -l
[1]+  9870 Running                 sleep 9999 &
```

Important - Notez que l'utilisation des touches CtrlC tue le processus généré par la commande **wait** et non le processus généré par la commande **sleep**.

La Commande nice

Cette commande affiche ou modifie la priorité d'un processus. La priorité par défaut de nice est 10. La valeur de nice la plus prioritaire est -20. La valeur la moins prioritaire est 19 :

```
[root@centos7 ~]# nice -n -20 sleep 1234
^Z
[2]+  Stopped                 nice -n -20 sleep 1234
```

```
[root@centos7 ~]# ps lx | grep sleep
0     0  9870  4077  20   0 107892   616 hrtime S    pts/0      0:00 sleep 9999
0     0 10282   552  20   0 107892   612 hrtime S    ?          0:00 sleep 60
4     0 10283  4077   0 -20 107892   612 signal T<   pts/0      0:00 sleep 1234
0     0 10394  4077  20   0 112640   960 pipe_w S+   pts/0      0:00 grep --color=auto sleep

[root@centos7 ~]# nice -n 19 sleep 5678
^Z
[3]+  Stopped                 nice -n 19 sleep 5678

[root@centos7 ~]# ps lx | grep sleep
0     0  9870  4077  20   0 107892   616 hrtime S    pts/0      0:00 sleep 9999
4     0 10283  4077   0 -20 107892   612 signal T<   pts/0      0:00 sleep 1234
0     0 10402   552  20   0 107892   612 hrtime S    ?          0:00 sleep 60
0     0 10403  4077  39  19 107892   616 signal TN   pts/0      0:00 sleep 5678
0     0 10405  4077  20   0 112640   960 pipe_w S+   pts/0      0:00 grep --color=auto sleep
```

Comme vous pouvez constater la 6ième colonne contient la valeur de nice qui s'applique à la priorité dans la colonne 5.

Important - Notez que seul root peut lancer des processus avec une valeur négative.

Options de la commande

Les options de cette commande sont :

```
[root@centos7 ~]# nice --help
Usage: nice [OPTION] [COMMAND [ARG]...]
Run COMMAND with an adjusted niceness, which affects process scheduling.
With no COMMAND, print the current niceness.  Niceness values range from
-20 (most favorable to the process) to 19 (least favorable to the process).
```

```
Mandatory arguments to long options are mandatory for short options too.
  -n, --adjustment=N   add integer N to the niceness (default 10)
      --help     display this help and exit
      --version  output version information and exit

NOTE: your shell may have its own version of nice, which usually supersedes
the version described here.  Please refer to your shell's documentation
for details about the options it supports.

GNU coreutils online help: <http://www.gnu.org/software/coreutils/>
For complete documentation, run: info coreutils 'nice invocation'
```

La Commande renice

Cette commande modifie la priorité d’un processus déjà en cours. La valeur de la priorité ne peut être modifiée que par le propriétaire du processus ou par root.

```
[root@centos7 ~]# jobs -l
[1]   9870 Running                 sleep 9999 &
[2]- 10283 Stopped                 nice -n -20 sleep 1234
[3]+ 10403 Stopped                 nice -n 19 sleep 5678
[root@centos7 ~]# bg %2
[2]- nice -n -20 sleep 1234 &
[root@centos7 ~]# bg %3
[3]+ nice -n 19 sleep 5678 &
[root@centos7 ~]# jobs -l
[1]   9870 Running                 sleep 9999 &
[2]- 10283 Running                 nice -n -20 sleep 1234 &
[3]+ 10403 Running                 nice -n 19 sleep 5678 &
[root@centos7 ~]# renice +5 10283
10283 (process ID) old priority -20, new priority 5
[root@centos7 ~]# renice -5 10403
10403 (process ID) old priority 19, new priority -5
```

```
[root@centos7 ~]# ps lx | grep sleep
0     0  9870  4077  20   0 107892   616 hrtime S    pts/0      0:00 sleep 9999
4     0 10283  4077  25   5 107892   612 restar SN   pts/0      0:00 sleep 1234
0     0 10403  4077  15  -5 107892   616 restar S<   pts/0      0:00 sleep 5678
0     0 10570   552  20   0 107892   616 hrtime S    ?          0:00 sleep 60
0     0 10648  4077  20   0 112640   960 pipe_w S+   pts/0      0:00 grep --color=auto sleep
```

Important -Notez que seul root peut décrémenter la valeur de priorité avec la commande renice.

Options de la commande

Les options de cette commande sont :

```
[r[root@centos7 ~]# renice --help

Usage:
 renice [-n] <priority> [-p|--pid] <pid>...
 renice [-n] <priority>  -g|--pgrp <pgid>...
 renice [-n] <priority>  -u|--user <user>...

Options:
 -g, --pgrp <id>        interpret argument as process group ID
 -n, --priority <num>   specify the nice increment value
 -p, --pid <id>         interpret argument as process ID (default)
 -u, --user <name|id>   interpret argument as username or user ID
 -h, --help             display help text and exit
 -V, --version          display version information and exit

For more information see renice(1).
```

La Commande nohup

Cette commande permet à un processus de poursuivre son exécution après la déconnexion. Un processus enfant meurt quand le processus parent meure ou se termine. Comme une connexion et un processus, quand vous vous déconnectez, vos processus se terminent. Pour éviter de rester connecté après avoir lancé un processus long, vous utiliserez la commande nohup :

```
nohup lp ventes.txt &
```

Options de la commande

Les options de cette commande sont :

```
[root@centos7 ~]# nohup --help
Usage: nohup COMMAND [ARG]...
  or:  nohup OPTION
Run COMMAND, ignoring hangup signals.

      --help     display this help and exit
      --version  output version information and exit

If standard input is a terminal, redirect it from /dev/null.
If standard output is a terminal, append output to 'nohup.out' if possible,
'$HOME/nohup.out' otherwise.
If standard error is a terminal, redirect it to standard output.
To save output to FILE, use 'nohup COMMAND > FILE'.

NOTE: your shell may have its own version of nohup, which usually supersedes
the version described here.  Please refer to your shell's documentation
for details about the options it supports.

GNU coreutils online help: <http://www.gnu.org/software/coreutils/>
```

```
For complete documentation, run: info coreutils 'nohup invocation'
```

La Commande kill

La commande kill envoie des signaux aux processus. La liste des signaux possibles peut être afficher avec l'option **-l** :

```
[root@centos7 ~]# kill -l
 1) SIGHUP   2) SIGINT     3) SIGQUIT     4) SIGILL     5) SIGTRAP
 6) SIGABRT  7) SIGBUS     8) SIGFPE     9) SIGKILL    10) SIGUSR1
11) SIGSEGV 12) SIGUSR2    13) SIGPIPE    14) SIGALRM    15) SIGTERM
16) SIGSTKFLT   17) SIGCHLD    18) SIGCONT    19) SIGSTOP    20) SIGTSTP
21) SIGTTIN 22) SIGTTOU    23) SIGURG    24) SIGXCPU    25) SIGXFSZ
26) SIGVTALRM   27) SIGPROF    28) SIGWINCH    29) SIGIO    30) SIGPWR
31) SIGSYS  34) SIGRTMIN    35) SIGRTMIN+1    36) SIGRTMIN+2    37) SIGRTMIN+3
38) SIGRTMIN+4  39) SIGRTMIN+5    40) SIGRTMIN+6    41) SIGRTMIN+7    42) SIGRTMIN+8
43) SIGRTMIN+9  44) SIGRTMIN+10    45) SIGRTMIN+11    46) SIGRTMIN+12    47) SIGRTMIN+13
48) SIGRTMIN+14 49) SIGRTMIN+15    50) SIGRTMAX-14    51) SIGRTMAX-13    52) SIGRTMAX-12
53) SIGRTMAX-11 54) SIGRTMAX-10    55) SIGRTMAX-9    56) SIGRTMAX-8    57) SIGRTMAX-7
58) SIGRTMAX-6  59) SIGRTMAX-5    60) SIGRTMAX-4    61) SIGRTMAX-3    62) SIGRTMAX-2
63) SIGRTMAX-1  64) SIGRTMAX
```

Important - Vous constaterez que chaque signal possède un numéro. Ces numéros de signaux sont utilisés à la place des options. Par exemple, **-19** à la place de l'option **-stop**.

Parmi les numéros de signaux les plus utiles on trouve :

Numéro	Description
-1	Le signal Hang Up est envoyé à tous les enfants d'un processus quand il se termine
-2	Interruption du processus - équivalent à CtrlC
-3	La même chose que -2 mais avec la génération d'un fichier de déboggage

Numéro	Description
-9	Le signal qui tue un processus brutalement
-15	Le signal envoyé par défaut par la commande **kill**. Le processus se termine normalement

Options de la commande kill

```
[root@centos7 ~]# help kill
kill: kill [-s sigspec | -n signum | -sigspec] pid | jobspec ... or kill -l [sigspec]
    Send a signal to a job.
    Send the processes identified by PID or JOBSPEC the signal named by
    SIGSPEC or SIGNUM.  If neither SIGSPEC nor SIGNUM is present, then
    SIGTERM is assumed.
    Options:
      -s sig    SIG is a signal name
      -n sig    SIG is a signal number
      -l    list the signal names; if arguments follow `-l' they are
        assumed to be signal numbers for which names should be listed
    Kill is a shell builtin for two reasons: it allows job IDs to be used
    instead of process IDs, and allows processes to be killed if the limit
    on processes that you can create is reached.
    Exit Status:
    Returns success unless an invalid option is given or an error occurs.
```

La Commande pkill

La commande pkill permet d'envoyer des signaux aux processus identifiés par leur nom. Par exemple la commande suivante force syslog de relire son fichier de configuration :

```
[root@centos ~]# pkill -HUP rsyslogd
```

Options de la commande pkill

```
[root@centos7 ~]# pkill --help

Usage:
 pkill [options] <pattern>

Options:
 -<sig>, --signal <sig>    signal to send (either number or name)
 -e, --echo                display what is killed
 -c, --count               count of matching processes
 -f, --full                use full process name to match
 -g, --pgroup <PGID,...>   match listed process group IDs
 -G, --group <GID,...>     match real group IDs
 -n, --newest              select most recently started
 -o, --oldest              select least recently started
 -P, --parent <PPID,...>   match only child processes of the given parent
 -s, --session <SID,...>   match session IDs
 -t, --terminal <tty,...>  match by controlling terminal
 -u, --euid <ID,...>       match by effective IDs
 -U, --uid <ID,...>        match by real IDs
 -x, --exact               match exactly with the command name
 -F, --pidfile <file>      read PIDs from file
 -L, --logpidfile          fail if PID file is not locked
 --ns <PID>                match the processes that belong to the same
                           namespace as <pid>
 --nslist <ns,...>         list which namespaces will be considered for
                           the --ns option.
                           Available namespaces: ipc, mnt, net, pid, user, uts

 -h, --help     display this help and exit
 -V, --version  output version information and exit
```

```
For more details see pgrep(1).
```

Copyright © 2024 Hugh Norris.