




# useradd

- Objet : useradd
- Niveau requis : [débutant, avisé](#)
- Commentaires : *Créer un nouvel utilisateur.*
- Débutant, à savoir : [Utiliser GNU/Linux en ligne de commande, tout commence là !](#) 😊
- Suivi :
  - [à-tester, à-compléter](#)
    - Création par  [smolski](#) le 10-10-2013
    - Testé par <...> le <...> 
- Commentaires sur le forum : [Lien vers le forum concernant ce tuto](#)<sup>1)</sup>

## Nota :

Contributeurs, les  sont là pour vous aider, supprimez-les une fois le problème corrigé ou le champ rempli !

## Introduction

La commande `useradd` permet de créer un nouvel utilisateur<sup>2)</sup>.  
Pour utiliser cette commande, les droits [superutilisateur](#) sont nécessaires..

## Syntaxe

```
useradd
```

### Option -m

L'option `-m` permet de créer en même temps le répertoire personnel.

```
useradd -m <user_créé>
```

Remplacer `<user_créé>` par le *pseudo de l'utilisateur* à créer. 😊

## TP - Créer un utilisateur

Par exemple, créons l'utilisateur et le répertoire *mule* ainsi :

```
useradd -m mule
```



ATTENTION ! Pour pouvoir ouvrir la nouvelle session, il ne faut pas oublier de créer un mot de passe pour l'utilisateur "**mule**"

## Création du mot de passe

Cela se fait avec [la commande PASSWD](#) ainsi :

```
passwd mule
```

### séquence interactive

```
Changing password for mule.  
Nouveau mot de passe :  
Retaper le nouveau mot de passe :  
Mot de passe changé.
```

## Génération du mot de passe

Pour générer des mots de passe en ligne de commande :

- Avec la commande `gpg` pour générer un mot de passe de 16 caractères sans caractères spéciaux :

```
gpg --gen-random --armor 1 16
```

- Avec la commande `pwmake` pour générer un mot de passe avec une entropie de 128 bits avec caractères spéciaux :

```
pwmake 128
```

- Avec la commande `pwgen` pour générer un mot de passe de 16 caractères sans caractères spéciaux :

```
pwgen 16 1
```

- Avec la commande `pwgen` pour générer un mot de passe de 16 caractères avec caractères spéciaux :

```
pwgen -ys 16 1
```

## Stockage des mot de passe

Pour les stocker d'une manière sûre :

- [La commande keepassx.](#)

# Options

## option -c

Les commentaires.

Lors de la création d'un utilisateur, l'administrateur donne généralement des renseignements complémentaires sur l'identité de la personne qui utilisera le compte.

Cette information est généralement constituée du prénom et du nom de la personne utilisatrice.

Ce type d'information étant généralement constituée d'au moins deux mots séparés par un espace, il ne faut pas que la commande useradd considère ces informations comme étant deux arguments distincts.

Pour contrecarrer le principe général de détection des différents arguments sur base de l'espacement, il suffit de placer le prénom et le nom de l'utilisateur entre 2 **doubles quotes** ainsi par exemple : "Pif Paf Pouf".

Cette identité doit être précédée de l'option -c afin d'informer la commande useradd de ce que l'argument suivant est le commentaire à mémoriser.

Voici la commande de création d'un utilisateur avec un argument commentaire :

```
useradd -c "Jean-Francois Challe" jchalle
```

## option -d

Lorsque la commande useradd est utilisée sans précision du répertoire de travail de l'utilisateur, le système crée un répertoire dans /home dont le nom est identique au login.

De nombreux administrateurs placent le répertoire de travail des utilisateurs dans un sous-répertoire de /home.

Dans un environnement réel, il se pourrait que les utilisateurs soient regroupés suivant le type d'activités qu'ils effectuent.

Il existerait alors autant de sous-répertoires dans /home que de groupes d'utilisateurs (cette notion n'a pas de rapport avec la notion UNIX de groupe).

L'option -d de la commande useradd permet, comme le montre l'exemple suivant, de spécifier l'emplacement et le nom du répertoire de travail de l'utilisateur :

```
useradd -d /home/staff/jchalle jchalle
```



Pour que cette commande aboutisse, il faut que le répertoire /home/staff soit créé.

## option -s

Les systèmes UNIX sont généralement livrés avec plusieurs interpréteurs de commandes, chacun ayant ses caractéristiques propres.

Selon l'utilisateur et le travail qui sera réalisé, un type d'interpréteur de commandes peut être choisi au moment de la création du compte.

Les systèmes Linux disposent des interpréteurs suivants dans le répertoire `/bin` :

- `ash` ;
- `bash` ;
- `bsh` ;
- `csh` ;
- `ksh` ;
- `sh` ;
- `tcsh` ;
- `zsh` ;

L'option `-s` de la commande `useradd` permet à l'administrateur de spécifier le shell par défaut qui sera employé par l'utilisateur.

Voici la syntaxe de la commande `useradd` permettant de configurer un compte pour qu'il utilise le korn shell :

```
useradd -s /bin/ksh jchalle
```

## option -g

Tous les utilisateurs font partie d'au moins un groupe.

Par défaut, la commande `useradd` crée un groupe dont le nom sera le même que celui du login.

Au moyen de l'option `-g`, il est possible de spécifier un autre groupe par défaut à la condition que ce groupe existe.

Voici la commande permettant de créer un nouvel utilisateur dont le groupe par défaut sera celui du super-utilisateur :

```
useradd -g root jchalle
```

Pour affiner les autorisations d'accès d'un utilisateur, il est possible de le placer dans plusieurs groupes.

Cette opération peut être effectuée au moment de la création du compte par l'intermédiaire de l'option `-G` de la commande `adduser` :

```
useradd -G adm,sys jchalle
```

Lorsque plusieurs utilisateurs forment un groupe de travail penché sur un même projet, il est intéressant qu'ils puissent se partager des informations.

Pour cela, chaque utilisateur est placé dans un groupe qui lui est propre mais également dans un groupe où tous les acteurs du projet se trouvent.

Une autre utilité de la commande consiste à donner, à certains utilisateurs, des droits correspondants à ceux du [superutilisateur](#).

Dans ce cas, il suffit d'inclure également le nouvel utilisateur dans le groupe `root`.

## option -u

À chaque utilisateur du système est associé un entier unique appelé User IDentifier. Pour les utilisateurs privilégiés du système, cet UID a une valeur comprise entre 0 et 99.

Lorsque la commande useradd doit attribuer un UID par défaut, elle donne toujours une valeur supérieure à 99.

De plus, si d'autres comptes ont été créés, la commande useradd choisit comme valeur d'UID, celle d'une unité supérieure à la plus grande valeur utilisée.

L'administrateur a la possibilité d'imposer une valeur d'UID au moment de la création d'un compte. Cette possibilité permet au gestionnaire du système d'attribuer, par exemple, un UID compris entre 1000 et 2000 aux utilisateurs du service commercial et un UID compris en 3000 et 4000 aux utilisateurs du service recherche et développement.

Voici un exemple de commande useradd attribuant l'UID 1000 au nouvel utilisateur créé :

```
useradd -u 1000 jchalle
```

## option -e

Dans certains cas, des comptes sont créés pour une durée limitée.

C'est, par exemple, le cas lorsqu'une société reçoit un stagiaire. Après une certaine date, le compte ne doit plus être utilisé.

L'option -e de la commande useradd permet de fixer la date à partir de laquelle le compte expire.

Cette date peut être révisée à la hausse au moyen de l'option -f.

L'option -e spécifie simplement une date d'expiration alors que l'option -f donne le nombre de jours de sursis accordé à l'utilisateur.

Voici un exemple de commande créant un compte utilisateur dont l'expiration sera impérativement le 15 janvier 2005 :

```
useradd -e 2005-01-15 -f 0 jchalle
```

## Lien

Voir :

- [La commande adduser](#)
- [La commande id](#)

<sup>1)</sup>

N'hésitez pas à y faire part de vos remarques, succès, améliorations ou échecs !

<sup>2)</sup>

à utiliser avec l'option -m pour créer en même temps le répertoire personnel

From:

<http://debian-facile.org/> - **Documentation - Wiki**

Permanent link:

<http://debian-facile.org/doc:systeme:useradd>



Last update: **25/08/2022 19:47**