

# Gestion des utilisateurs et groupes

# Identification et authentification

- L'identification, c'est savoir qui est connecté, afin de déterminer les droits de la personne qui se connecte. Un utilisateur est identifié par un login.
- L'authentification, c'est apporter la preuve de qui on est, par exemple via un secret partagé entre l'utilisateur et le système, et connus d'eux seuls. L'utilisateur est authentifié par un mot de passe.



# Identification des utilisateurs et groupes

- Comprendre l'identification des utilisateurs est crucial pour :
  - Résoudre les problèmes d'autorisations
  - Configurer correctement les autorisations de fichiers et de répertoires
  - Gérer les comptes utilisateurs
  - Assurer la sécurité du système
  - Auditer l'accès au système

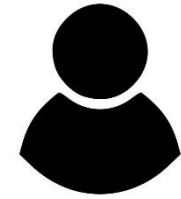


# Identification des utilisateurs et groupes

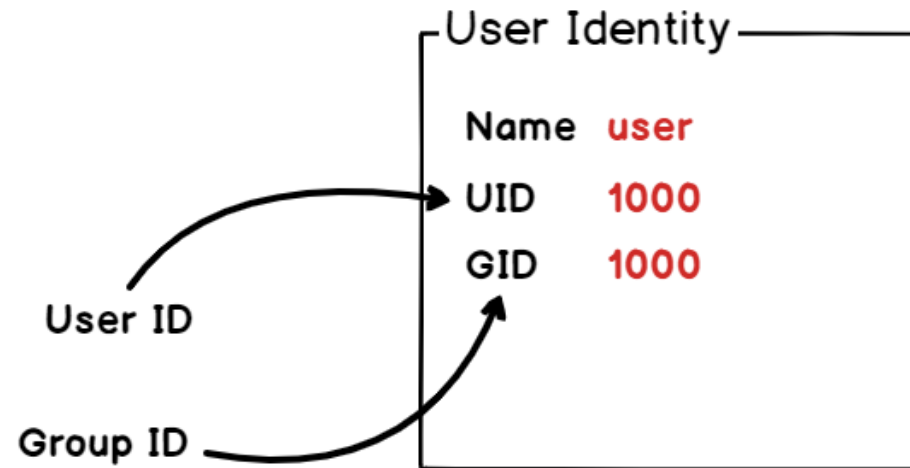
- La commande whoami, qui affiche le nom d'utilisateur actuellement connecté
- La commande id, qui fournit des informations détaillées sur l'utilisateur et le groupe
- La commande groups, qui répertorie tous les groupes auxquels un utilisateur appartient



# L'utilisateur



- Un utilisateur est l'association d'un nom de connexion, le login, à un UID et au moins un GID.



- UID : User ID.
- GID : Group ID.

- Les UID et les GID sont en principe uniques. Le login est unique. L'UID identifie l'utilisateur (ou le compte applicatif) tout au long de sa connexion. Il est utilisé pour le contrôle de ses droits et de ceux des processus qu'il a lancé

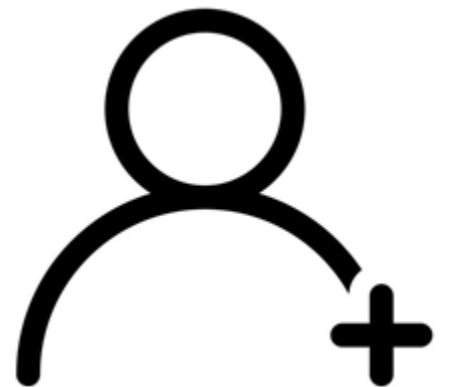
# Le groupe

- Chaque utilisateur fait partie d'au moins un groupe. Un groupe regroupe des utilisateurs. Comme pour les logins, le GID du groupe accompagne toujours l'utilisateur pour le contrôle de ses droits
- La commande `id` donne les informations sur un utilisateur : uid, gid et liste des groupes secondaires auxquels appartient l'utilisateur



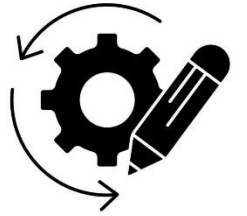
# Créer et sécuriser un compte utilisateur

- Parmi les tâches administratives fondamentales pour la gestion d'un système Linux **la création d'un nouveau compte utilisateur** en utilisant la commande **useradd** et le sécuriser en **définissant un mot de passe** avec la commande **passwd**
- Tout d'abord, créons un nouvel utilisateur nommé `etudiant1`. La commande `useradd` nécessite des privilèges root pour créer des utilisateurs, vous devrez donc préfixer la commande avec `sudo`.
- Exécuter les commandes suivantes dans votre terminal :
  - `sudo useradd <option> etudiant1`
  - `sudo passwd etudiant1`



Option	Rôle
<b>-m</b>	Crée aussi le répertoire personnel. Elle est parfois comprise par défaut, mais il vaut mieux vérifier si le répertoire personnel est présent après l'utilisation de la commande si vous n'utilisez pas cette option.
<b>-u</b>	Précise l'UID numérique de l'utilisateur, pour le forcer. Autrement l'UID est calculé selon les règles du fichier login.defs et les UID existants.
<b>-g</b>	Précise le groupe principal de l'utilisateur, par GID ou par son nom (variable GROUP).
<b>-G</b>	Précise les groupes additionnels (secondaires, de l'utilisateur) séparés par des virgules (variable GROUPS).
<b>-d</b>	Chemin du répertoire personnel. Généralement /home/<login>, mais n'importe quel chemin peut être précisé (variable HOME/<login>).
<b>-c</b>	Un commentaire associé au compte. Il peut être quelconque mais est parfois utilisé par certaines commandes comme <b>finger</b> . Son contenu peut être modifié par l'utilisateur avec la commande <b>chfn</b> .
<b>-k</b>	Chemin du répertoire contenant le squelette de l'arborescence du répertoire utilisateur. C'est généralement /etc/skel (variable SKEL).
<b>-s</b>	Shell (commande de connexion) par défaut de l'utilisateur (variable SHELL). L'utilisateur peut le changer via la commande <b>chsh</b> .

# Modification



- Utilisez la commande `usermod` pour modifier un compte. Elle prend la même syntaxe et options que `useradd` mais dispose aussi d'une syntaxe complémentaire.

Option	Rôle
<b>-L</b>	Lock du compte, comme <code>passwd -l</code> .
<b>-U</b>	Unlock du compte, comme <code>passwd -u</code> .
<b>-e &lt;n&gt;</b>	Expire : le mot de passe expire n jours après le 01/01/1970.
<b>-u &lt;UID&gt;</b>	Modifie l'UID associé au login. Le propriétaire des fichiers appartenant à l'ancien UID au sein du répertoire personnel est modifié en conséquence.
<b>-l &lt;login&gt;</b>	Modifie le nom de login.
<b>-m</b>	Move : implique la présence de <code>-d</code> pour préciser un nouveau répertoire personnel. Le contenu de l'ancien répertoire est déplacé dans le nouveau.

# Modification

- Utilisez la commande `usermod` pour modifier un compte. Elle prend la même syntaxe et options que `useradd` mais dispose aussi d'une syntaxe complémentaire.
- La commande `usermod` change ce qu'on veut avec les options :
  - `-g groupe` : change le groupe principal
  - `-c infos` : change les informations
  - `-u UID` : change l'UID de l'utilisateur
  - `-s shell` : change son shell de connexion
- `sudo usermod -u 1500 etudiant1`



# Suppression

- Supprimez un utilisateur avec la commande `userdel`.
- `sudo userdel user1`
- Par défaut le répertoire personnel n'est pas supprimé. Vous devez pour ceci passer l'option `-r`.
- `sudo userdel -r user1`



# Où sont stockées informations de l'utilisateur

- /etc/passwd : login, UID, GID, GECOS, home dir, shell
- /etc/shadow : mot de passe, dates de validité des mots de passe. . . '
- /etc/group : nom des groupes, GID, listes de membres
- /etc/gshadow : mots de passe, administrateurs. . .
- /home/username : répertoire personnel de l'utilisateur, contenant ses fichiers et préférences



# Changer le mot de passe

- Les mots de passe permettent d'authentifier les utilisateurs. Les mots de passe sont cryptés (MD5, DES par exemple) et ne sont pas directement lisibles sous leur forme cryptée par l'utilisateur afin que personne ne puisse tenter de le décrypter via un quelconque traitement.
- La commande passwd permet de gérer les mots de passe mais aussi les autorisations de connexion
- passwd etudiant1



# Changer de mot de passe

Option	Rôle
<b>-l</b>	Lock : verrouille le compte en rajoutant un ! devant le mot de passe crypté.
<b>-u</b>	Unlock : déverrouille le compte. Il n'est pas possible de déverrouiller un compte qui n'a pas de mot de passe, il faut utiliser en plus f pour cela.
<b>-d</b>	(root) Supprime le mot de passe du compte.
<b>-n &lt;j&gt;</b>	(root) Durée de vie minimale en jours du mot de passe.
<b>-x &lt;j&gt;</b>	(root) Durée de vie maximale en jours du mot de passe.
<b>-w &lt;j&gt;</b>	(root) Nombre de jours avant avertissement.
<b>-i &lt;j&gt;</b>	(root) Délai de grâce avant désactivation si le mot de passe est expiré.
<b>-S</b>	(root) Statut du compte.

# Changer de mot de passe

Dans l'exemple suivant le compte etudiant1 est modifié comme ceci :

- Il doit attendre 5 jours après saisie d'un nouveau mot de passe pour pouvoir le changer,
  - Son mot de passe est valide 45 jours,
  - Il est prévenu 7 jours avant qu'il doit changer de mot de passe,
  - S'il ne change pas de mot de passe après 45 jours, il aura encore 5 jours avant d'être désactivé.
- 
- `# passwd -n 5 -x 45 -w 7 -i 5 etudiant1`

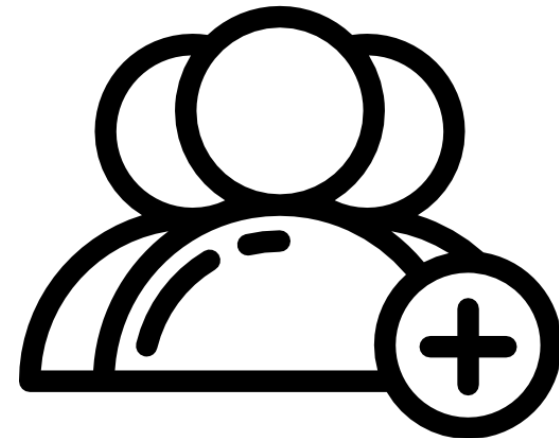
# La commande chage

- La commande chage (abréviation de "change age") permet de gérer les politiques de mots de passe des utilisateurs, de visualiser et de modifier les informations d'expiration des mots de passe pour les comptes utilisateurs.
- On prend comme exemple etudiant1, visualisons ses paramètres d'expiration de mot de passe par défaut. Utiliser la commande chage avec l'option -l (list) :
- `sudo chage -l etudiant1`

```
user@user-virtual-machine:~$ sudo chage -l etudiant1
Dernière modification du mot de passe                : nov. 13, 2025
Le mot de passe expire                                : jamais
Mot de passe inactif                                  : jamais
Le compte expire                                      : jamais
Nombre minimal de jours entre deux changements de mot de passe : 0
Nombre maximal de jours entre deux changements du mot de passe  : 99999
Nombre de jours d'avertissements avant que le mot de passe n'expire : 7
```

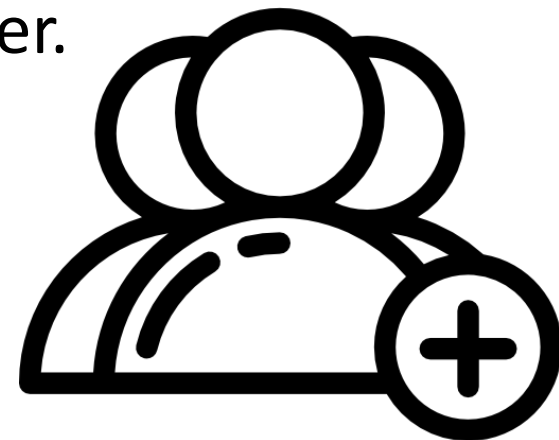
# Ajout du groupe

- La commande `sudo` élève vos privilèges pour effectuer cette tâche administrative. **groupadd** est la commande pour **créer le groupe**, et `coursA` est le nom choisi pour le nouveau groupe comme exemple.
  - `sudo groupadd coursA`



# Ajout du groupe

- La commande `sudo` élève vos privilèges pour effectuer cette tâche administrative. **groupadd** est la commande pour **créer le groupe**, et `coursA` est le nom choisi pour le nouveau groupe comme exemple.
  - `sudo groupadd coursA`
- Pour confirmer que le groupe a été créé, vous pouvez vérifier le fichier `/etc/group`. Ce fichier stocke les informations sur tous les groupes du système. Nous pouvons utiliser la commande `grep` pour rechercher notre groupe nouvellement créé dans ce fichier.
  - `Grep coursA /etc/group`



# Modification du groupe

- La commande `groupmod` permet de modifier un groupe. Ses paramètres sont les suivants :

Option	Rôle
<b>-n &lt;nom&gt;</b>	Renomme le groupe.
<b>-g &lt;GID&gt;</b>	Modifie le GID. Attention, le groupe d'appartenance des fichiers concernés n'est pas modifié.
<b>-A &lt;user&gt;</b>	Ajoute l'utilisateur spécifié dans le groupe (groupe secondaire).
<b>-R &lt;user&gt;</b>	Supprime l'utilisateur spécifié du groupe.



# Ajouter un utilisateur à un groupe secondaire avec usermod

- Dans cette étape, vous allez ajouter l'utilisateur `etudiant1` au groupe `coursA` créé. Sous Linux, chaque utilisateur a un *groupe primaire* et peut appartenir à plusieurs *groupes secondaires*
- Nous utiliserons la commande `usermod` avec les options `-aG` :
- `-G` : Spécifie la nouvelle liste des groupes secondaires.
- `-a` : Signifie "append" (ajouter). C'est une option très importante. Elle ajoute l'utilisateur aux groupes spécifiés *sans* le retirer de ses groupes actuels. Si vous omettez `-a`, l'utilisateur sera retiré de tous les autres groupes secondaires non listés dans la commande.
- Dans votre terminal, exécutez la commande suivante pour ajouter l'utilisateur `etudiant1` au groupe `coursA` :
- `sudo usermod -aG coursA etudiant1`

# Afficher groupes

- Pour afficher tous les groupes auxquels un utilisateur particulier appartient. C'est une tâche courante pour les administrateurs système afin de vérifier les permissions et les configurations.
- Pour voir tous les groupes auxquels l'utilisateur etudiant1 appartient, exécutez la commande suivante :
- `groups etudiant1`
- Utiliser `grep` pour trouver tous les groupes secondaires dont l'utilisateur etudiant1 est membre. En recherchant le nom d'utilisateur dans le fichier `/etc/group`, on peut voir toutes les entrées de groupe où etudiant1 est listé comme membre.
- Exécuter la commande dans le terminal :
- `grep etudiant1 /etc/group`

# Suppression du groupe

- La commande **groupdel** **supprime un groupe**. (La commande vérifie d'abord si le groupe que vous voulez supprimer est le groupe principal d'un utilisateur alors le groupe ne peut pas être supprimé)
- `sudo groupdel coursA`
- Pour confirmer que le groupe a été supprimé avec succès, la commande `grep` est utilisée pour vérifier son existence.
- `grep coursA /etc/group`



# Gérer les groupes windows

- Afficher la liste des comptes utilisateurs locaux: `net user`
- Créer un nouvel utilisateur local : `net user [nom] /add`
- Supprimer un utilisateur: `net user [nom] /delete`
- Modifier le mot de passe de l'utilisateur : `net user [nom] *`
- Afficher les groupes locaux : `net localgroup`
- Afficher les membres d'un groupe : `net localgroup [groupe]`
- Ajouter un utilisateur à un groupe : `net localgroup [groupe] [utilisateur] /add`
- Retirer un utilisateur d'un groupe : `net localgroup [groupe] [utilisateur] /delete`
- Afficher le nom de l'utilisateur connecté : `whoami`
- Afficher les SID des comptes utilisateurs: `wmic useraccount get name,sid`

# TP 1

1. Ajouter un nouveau compte d'utilisateur user1 avec le répertoire personnel
2. Attribuer un mot de passe a user1
3. Créer un nouveau groupe tp1 et ajouter user1 au groupe et vérifier dans le fichier /etc/group
4. Vérifier la configuration avec `ls -ld /home/user1`
5. Créer un script de gestion automatisée de ces étapes ainsi que l'affichage des informations ci-dessous:

## # Affichage des informations

- Utilisateur ... : créé avec succès."
- Mot de passe défini."
- Ajouté au groupe :....
- Répertoire personnel : .....
- Groupes actuels :....

# TP 2

- Appliquons une politique plus stricte avec chage. Nous allons configurer le compte user1 de manière à ce que :
  - -m 7: L'utilisateur doit attendre au moins 7 jours avant de changer à nouveau son mot de passe.
  - -M 90: Le mot de passe expire après 90 jours.
  - -W 14: L'utilisateur recevra un avertissement 14 jours avant l'expiration de son mot de passe.
- 
1. Appliquer tous ces paramètres avec une seule commande chage
  2. Verrouiller le compte user1 et vérifier le status
  3. Déverrouiller le compte user1
  4. Donner une date d'expiration temporaire du compte user1 avec chage (à partir du 31 décembre 2025)

# TP 3

- Créer un script de génération massive de x utilisateurs ayant comme préfix 'etudiant' et avec un mot de passe en commun 'pass'
- Pour la création de 3 nouveaux utilisateurs par exemple, la sortie doit être sous ce format:
- Utilisateur etudiant1 créé avec mot de passe par défaut
- Utilisateur etudiant2 créé avec mot de passe par défaut
- Utilisateur etudiant3 créé avec mot de passe par défaut