Généralités administration système sur Linux Debian

Listes commandes utiles

- *sudo* : permet d'executer la commande qu'elle précède avec les droits sur superutilisateur (SuperUtilisateurDO).
- *cd mondossier* : permet de se rendre dans *mondossier . cd ../* permet de remonter d'un niveau dans l'arborescence.
- *mkdir mondossier* : permet de créer un dossier nommé *mondossier.*
- *apt update* : permet de mettre à jour le fichier cache contenant les paquets disponibles sur les dépôts référérencés.
- *apt install monpaquet* : installe le paquet mon paquet.
- *apt upgrade* : permet de mettre à jour tout les paquets.
- *cp dossiersource dossier destination* : permet de copier dossiersource dans dossier destination.
- *dir* : liste les fichiers du dossier actuellement parcouru.
- *ls -l* : similaire à dir, donne d'autres informations comme les droits.
- *chown user monfichier* : change l'utilisateur propriétaire de monfichier pour user.
- *chgrp group monfichier* : change le groupe de monfichier pour group.
- *chmod droits fichier* : change les droits de fichier (voir plus bas, chapitre ACL).
- *nano monfichier* : permet de lancer un éditeur de texte sur monfichier.

La plupart des commandes peuvent être adjointe d'arguments. Par exemple en ajoutant -h ou -help après une commande, on accède souvent à une aide listant les arguments qu'il est possible d'ajouter.

Chemins

Les '/' ont une importance capitale quand on lance une commande. La racine du système est /.

- Si je tape *cd /var* je vais atterrir dans le dossier var, quelque soit le dossier où j'étais actuellement. On parle de c**hemin absolu.**
- Si je suis dans le dossier */etc* et que je tape *cd var*, je dis au système que je veux aller dans */etc/var*, pas dans */var*. Si un dossier */etc/var* existe, alors tout se passera bien, s'il n'existe pas, le système me renverra une erreur. On parle de **chemin relatif.**

Les '/' ont également une signification en fin de chemin. Ainsi /var/www designe le dossier /etc/var

et /var/www/ désigne son contenu. Ainsi ces deux commandes bien que très similaires ne feront pas la même chose :

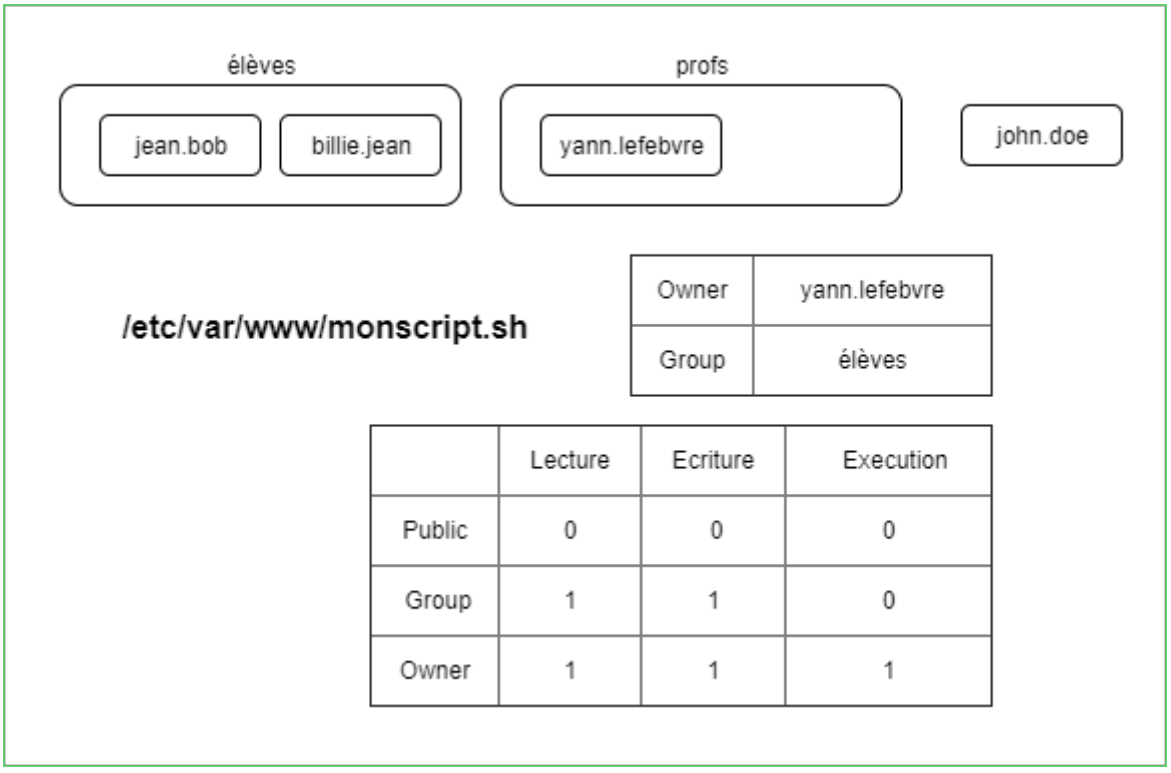
- *cp /var/log /tmp* va copier (et donc créer) le dossier /tmp/log et tout son contenu. On finit donc avec un dossier /tmp/log avec tout le contenu de /var/log.
- *cp /var/log/ /tmp* va copier le contenu de /var/log dans le dossier /tmp. On finit donc avec un dossier /tmp ayant pour contenu ce lui de /var/log*

Droits sur les fichiers (ACL)

Une des forces des systèmes Unix est la gestion des utilisateurs et des droits qui est à la fois simple et très sécurisée (sauf bien sûr si on fait n'importe quoi ...).

Pour utiliser le système il faut déjà disposer d'un utilisateur valide sur le système. Il s'identifie très généralement grâce à un login et un mot de passe. Cet utilisateur peut ensuite faire partie d'un ou plusieurs groupes. Ainsi on va pouvoir gérer des droits dans le système avec des groupes, ce qui permet de faire des réglages "de masse" sans avoir avec les traiter un par un. Par exemple j'ai des comptes pour plusieurs employés que j'ai regroupé dans un groupe "salariés". Je peux maintenant leur donner accès à un dossier en particulier de mon système en une seule commande.

Chaque fichier et chaque dossier du système appartient à un utilisateur et à un groupe. Une table permet ensuite de définir qui peut faire quoi sur ce fichier (Lire/Ecrire/Executer).



- Si je suis john.doe :
  - nano /etc/var/www/monscript.sh => échec
  - sh /etc/var/www/monscript.sh => échec
-

Si je suis billie.jean :

- nano /etc/var/www/monscript.sh => OK
- sh /etc/var/www/monscript.sh => échec

- Si je suis yann.lefebvre
  - nano /etc/var/www/monscript.sh => OK
  - sh /etc/var/www/monscript.sh => OK

Revision #9
Created 4 November 2019 10:04:40 by Admin
Updated 9 December 2019 13:30:06 by Admin