

LO22

Aucun document autorisé

Baala, Sibille

1. Quelle est le rôle d'une "inode" (nœud d'index) sous Unix?
2. Histoires de droits.

Tout cet exercice se déroule dans votre répertoire courant et vous êtes un utilisateur différent de root (le super-utilisateur).

2.1) Vous voulez modifier le contenu d'un fichier existant, quelles sont les conditions pour que vous puissiez y parvenir.

2.2) Afin de vous prémunir des destructions malencontreuses par rm, vous décidez de protéger les fichiers de votre répertoire MesSauvegardes en adoptant des droits appropriés. Donnez et justifiez votre ou vos commandes.

2.3) Vous désirez **maintenant** protéger le contenu de vos fichiers contre vos autres maladresses (modification intempestive,...), comment faites-vous?

2.4) Dans le répertoire MesSauvegardesC vous sauvegardez tous vos fichiers liés au langage C de votre arborescence ( sources C et entête):  
Comment faites-vous cela en une seule commande (ou à la rigueur en 2 commandes similaires)?

Au besoin vous préciserez les limites de fonctionnement de votre commande ou les nécessités d'organisation de votre arborescence.

2.5) Admettons maintenant que le répertoire MesSauvegardesC est protégé comme MesSauvegardes, donnez le contenu du script SauverMesSources permettant de sauvegarder vos sources C, tout en les protégeant ensuite des mauvaises manipulations déjà citées.

3. Un utilisateur saisit au niveau système la ligne suivante :  
( cat < f\_donnee | wc ) > f\_resultat

Que fait-elle?

Vous souhaitez conserver les messages de stderr dans erreur.log, que devient cette ligne de commande? Au besoin, expliquez simplement vos choix éventuels.

Et la ligne suivante, que fait-elle?

( cat | wc ) < f\_donnee > f\_resultat

Proposer une autre forme d'écriture.

4. Dans votre répertoire se trouve différents fichiers : des sources C (.c et .h), des fichiers divers et variés, et des répertoires.

Vous désirez pouvoir protéger ou déprotéger les sources C de votre répertoire (sans toucher aux sources des sous-répertoires). Pour ce, vous allez écrire 2 shells-scripts, AutoriserW et InterdireW qui respectivement autorise ou interdit la modification du contenu des sources C.

4.1 Donner le contenu ces 2 scripts.

4.2 Fusionner ces 2 scripts dans un seul fichier contenant 2 fonctions AutoriserW et InterdireW, ainsi que le "corps principal".

Ecrire le contenu du fichier, sachant que selon le nom de lancement (d'exécution) du script, la commande ajoutera ou retirera les droits d'écriture sur les fichiers sources.

Préciser les 2 noms donnés à ce fichier script.

5. Mort ou vif!

Le script TesterProcessus vérifie l'existence des processus dont les identificateurs de processus sont contenus dans le fichier "Processus.pid". Si tous les processus existent alors TesterProcessus affiche le message "Tous les Processus sont vivants" et retourne 0 sinon TesterProcessus retourne 1 et affiche sur la sortie d'erreur le message "Il manque au moins un processus".

Le fichier Processus.pid est constitué d'une suite de lignes composée chacune comme suit:  
**numéro\_de\_pid Nom\_du\_Processus**

Donner le contenu de TesterProcessus.

*Eléments du manuel et +:*

*La commande "kill -0 <num\_pid> retourne 0 si le processus <num\_pid> est actif, retourne une valeur différente de 0 si cet identificateur ne correspond pas à un processus existant.*

*In fich autreNom permet de donner un nom supplémentaire (ici autreNom) au fichier de nom fich.*