

SCR.1.2 TP 09 ⊥ :

La commande find

On se place dans SCR.1.2/TP09/ et on y place l'archive `arch_for_find.tar.gz`

On désarchive. On se place dans TREE. Tout le TP doit être réalisé **sans bouger de TREE**.

Dans TREE, il y a un répertoire `lib/kernel/config/`

Sans bouger de TREE, on crée, dans `lib/kernel/config/`, les répertoires `AA/BB/CC/XX` et `AA/BB/DD/YY`

Note. Les lignes de commandes qu'on construit doivent fonctionner quelque soit l'arborescence sur laquelle on les applique.

I. Tout le TP doit être réalisé **sans bouger de TREE**.

On donne la liste, dans toute l'arborescence commençant par TREE, sans utiliser de pipeline et sans utiliser `xargs` :

1. des répertoires dont le nom se termine par `.install`
2. des fichiers réguliers dont le nom se termine par `.d`
3. des répertoires vides.
4. des fichiers réguliers dont la taille est inférieure à 1000 octets.
5. des fichiers réguliers dont la taille est supérieure à 30000 octets.
6. des fichiers réguliers dont la taille est 1024 octets.
7. des fichiers réguliers dont le contenu occupe 2 blocs.
8. des fichiers réguliers dont le contenu occupe plus de 30 blocs.
9. des fichiers réguliers dont le contenu occupe moins de 3 blocs.
10. des fichiers réguliers dont le contenu occupe plus d'un bloc et moins de 4 blocs.
11. des éléments qui ont une date de dernière modification postérieure à celle du fichier `atom.install`
12. des éléments dans toute l'arborescence commençant à partir de `lib/krb5/syn/` qui ont été modifiés il y a moins de 4 jours.
13. des éléments dans toute l'arborescence commençant à partir de `lib/kernel/install.d` qui ont été modifiés il y a plus de 3 jours.
14. des éléments qui ont été modifiés il y a 3 jours et dont le temps de dernier accès remonte à 2 jours.
15. des répertoires qui sont modifiables par le groupe ou par les autres utilisateurs. On n'utilisera pas d'opérateurs.
16. des répertoires dont le contenu est lisible par tous. On n'utilisera pas d'opérateurs.
17. des fichiers réguliers dont les permissions sont exactement : lisibles et modifiables par le propriétaire, et seulement lisibles par les autres utilisateurs. On n'utilisera pas d'opérateurs.

II. Tout le TP doit être réalisé **sans bouger de TREE**.

À l'aide de la commande `find` et sans utiliser de pipeline et sans utiliser `xargs` :

1. On copie, dans **chacun** des répertoires `lib/kernel/config/AA/BB/CC/`, `lib/kernel/config/AA/BB/CC/XX`, `lib/kernel/config/AA/BB/DD/`, et `lib/kernel/config/AA/BB/DD/YY`, tous les fichiers réguliers de toute l'arborescence TREE dont le nom commence par un chiffre, et qui se trouvent à une profondeur d'au plus 4 (donc moins de 5.)
2. On donne la liste de tous les éléments dans toute l'arborescence commençant par TREE dont le temps de dernier accès est passé d'au moins 30 minutes.

3. On positionne les permissions d'accès à 0700 pour tous les répertoires dans l'arborescence commençant par **TREE** dont le nom commence par une majuscule, et qui se trouvent à une profondeur d'au moins 6 (donc plus de 5.)
4. On donne la liste de tous les éléments dans toute l'arborescence commençant par **TREE** dont le temps de changement de statut remonte à au moins 15 minutes.

III. Tout le TP doit être réalisé **sans bouger de TREE**.

Faire une copie de **TREE** en appelant la copie **TRee**

À l'aide de la commande **find**, sans utiliser de pipeline et sans utiliser **xargs**, on cherche :

1. dans l'arborescence commençant par **/**, les fichiers dont le nom se termine par **.h** et on arrête la recherche dès qu'on en trouve un. On ne veut pas voir les messages d'erreurs à l'écran.
2. dans l'arborescence commençant au répertoire domicile, les éléments dont la date de dernière modification est plus récente que celle du fichier **transm.d** se trouvant dans **TRee**. Pour chacun d'eux, on veut afficher le nom complet du fichier suivi, sur la même ligne, du symbole **--** suivi, sur la même ligne, de la date de dernière modification, et avec un fichier par ligne. On ne veut pas voir les messages d'erreurs à l'écran.