

M1-DNR2I TP3-2 :

Utilitaires

Gaétan Richard

30 janvier 2012

1 ssh and co

En se basant sur **ssh** la commande **scp** permet de copier des fichiers depuis ou vers une machine distante.

Question 1 Copiez les fichiers `.pdf` (ou autres) de votre répertoire dans le répertoire `/tmp/` de la machine de votre voisin.

Question 2 Faire de même en copiant des fichiers depuis une machine distante sur votre machine.

Dans un troisième temps, copiez des fichiers présents sur votre machine sur une machine distante en vous connectant en utilisant le login de votre voisin (celui-ci devra taper son mot de passe).

Question 3 Observez les droits utilisés et les propriétaires obtenus. Commenter.

Il existe aussi le protocole `sftp` qui permet de faire du transfert de fichier au dessus de `ssh`. Ce protocole sert par exemple à déposer des fichiers sur votre espace perso.

Question 4 En utilisant le logiciel `gftp`, déposez un fichier sur votre espace personnel en utilisant le protocole `sftp` (qui peut être présent sous la forme `SSH2`).

2 Screen

Même si **ssh** permet de travailler sur une machine distante, il est sensible à des déconnexions. Pour contrer cela, il existe un logiciel qui permet de maintenir un shell sur la machine distante même en cas de déconnexion : **screen**.

Connectez-vous sur une machine distante, lancez un **sleep 10000** puis tuez le processus **ssh** sur votre machine.

Question 5 *Qu'arrive-t-il à la commande lancée sur la machine distante ?*

La commande **screen** lance une autre commande (par défaut un shell) et ajoute un certain nombre de fonctionnalités. Celles-ci sont accessibles à l'aide de la combinaison **CTRL+a**.

Question 6 *Trouver le combinaison de touches pour détacher un screen. Que se passe-t-il alors pour la commande en cours d'exécution ?*

Question 7 *En regardant le manuel, essayez de reprendre la main sur ce screen.*

Ces deux fonctionnalités permettent de maintenir une commande en cas de déconnexion.

Connectez-vous sur une machine distante, lancer un screen dans lequel vous exécuterez la commande **sleep 10000**. Tuez ensuite la connexion sur votre machine.

Question 8 *Connectez-vous de nouveau à la machine distante. Qu'advient-il de votre commande.*

screen permet également d'avoir plusieurs fenêtres et de séparer les fenêtres.

Question 9 *À l'aide du manuel et de l'aide en ligne obtenue avec **CTRL+a** ", créez trois fenêtres dans votre screen et naviguez entre les différentes fenêtres.*

3 Archives

Pour transférer des fichiers, il est utile de regrouper des fichiers sous forme d'une archive. Sous Linux, on utilise souvent la commande **tar**.

Question 10 *À l'aide du manuel de cette commande, choisissez un répertoire et faites-en une archive (extension **tar**).*

Question 11 *Listez le contenu de cette archive puis extrayez-là dans un autre dossier.*

Pour réduire la taille, on utilise souvent une compression **gz** (option **z**) ou **bz2** (option **j**).

Question 12 *En choisissant un répertoire conséquent, comparer le temps de compression / décompression ainsi que le ratio obtenu pour chacune des méthodes (vous pourrez utiliser la commande **time**).*

L'avantage de **tar** est qu'il respecte la hiérarchie des fichiers et dossiers.

Question 13 *Créez une archive qui se décompresse par défaut dans un répertoire à votre nom. Pourquoi ce comportement est-il vraiment utile et important lors de l'envoi d'archives ?*

Question 14 Ajoutez un dossier caché `.svn` dans le répertoire à archiver (ou utilisez un répertoire contenant un projet sous gestionnaire de version). Créez une archive et regardez les fichiers présents. Commentez.

Dans le cadre d'un dossier sous `svn`, il existe la commande `svn export` pour faire une archive des fichiers sous gestionnaire de version.

Question 15 En regardant le manuel de `tar`, indiquez les options à ajouter pour exclure les fichiers cachés.

4 `rsync`

`rsync` est une commande permettant de synchroniser complètement un répertoire avec un autre. La syntaxe des chemins est la même que pour `scp`.

Question 16 À l'aide de la page de `man`, utilisez `rsync` pour synchroniser deux répertoires situés dans `/tmp/` sur deux machines différentes.

Question 17 Quel sont les avantages de `rsync` par rapport à une simple copie ?