SFTP

Pour se connecter, rien de sorcier, c'est quasi comme en ssh : sftp [-P port] login@host.

Vous verrez que les commandes sont les mêmes que celles que vous connaissez déjà en bash. La seule différence est que sftp permet de gérer deux dimensions à la fois : les fichiers distants (sur le serveur) et les fichiers locaux (sur le client).

Lorsque nous voulons effectuer une action en local, il suffit de préfixer la commande avec l. Les détails dans le tableau ci-dessous.

Commande	**Description**
pwd	Affiche le dossier courant
lpwd	Affiche le dossier courant local (client)
cd	Change de répertoire
lcd	Change de répertoire en local
ls	Liste les fichiers et répertoires du dossier courant
lls	Liste les fichiers et répertoires du dossier courant sur le client
mkdir	Créer un répertoire
lmkdir	Créer un répertoire en local
get	Télécharger des fichiers depuis le serveur vers le client (l'option -r permet d'envoyer un répertoire)
put	Envoyer des fichiers depuis le client vers le serveur (l'option -r permet d'envoyer un répertoire)
rm	Supprime un fichier
rmdir	Supprime un répertoire
exit	Je pense que celui-ci se passe d'explications
help	Afficher l'aide
LFTP	le client qui vous veut du bien

Conseillé par @BXT dans les commentaires, LFTP est un client bien plus puissant que le sftp natif. Ce client supporte de nombreux protocoles, mais le cas qui nous intéresse ici est bien l'usage du sftp. Pour se connecter avec ce protocole, il va falloir lui spécifier dans l'url :

lftp [-p port] protocol:*login:password@host # exemple lftp sftp:*buzut:azerty@127.0.0.10

Chose assez étrange, même si vous utilisez une connexion par clef (et que vous ne spécifiez pas de mot de passe), l'oiseau vous le demande quand même, vous n'avez cas spécifier n'importe quoi ou juste "entrer".

Mais alors, pourquoi utiliser lftp plutôt que sftp ? Le premier supporte l'auto-complétion, l'effacement récursif (vous pouvez faire rm -r un_dossier/), sans vous embêter à d'abord effacer manuellement tout son contenu. Il supporte aussi le mirrorring un peu à la rsync en upload et en download. Pratique pour ne répercuter que les modifications !

-R pour inverser (de la machine locale vers le serveur) lftp > mirror [-R] repertoire_source repertoire_dest

Dans les options intéressantes du mode mirror, on notera –delete qui permet d'effacer les fichiers du répertoire de destination qui ne sont pas présents dans le répertoire source – comme dans rsync – et

–parallel[=N] qui permet de paralléliser les téléchargements.

On n'hésitera donc pas longtemps à troquer sftp contre lftp pour tout autre chose qu'un trivial upload.

Se connecter en sftp

Connexion depuis un terminal

Dans un terminal, vérifiez que la commande sftp est installée en écrivant :

sftp

La commande pour vous connecter est la suivante :

sftp -P xxxx user@pcc-xxx-xxx-xxx-xxx.ovh.com xxxx étant le numéro de port de connexion

Le mot de passe de l'utilisateur vous sera ensuite demandé. Une fois connecté, vous pourrez lister vos datastores à l'aide de la commande ls :

sftp> ls pcc-000714

Parcourez la liste des datastores trouvés avec la commande précédente :

sftp> pcc-000714

Utilisez la commande put pour exporter des fichiers de votre datastore vers votre poste local.

sftp> put /home/ubuntu-16.04.3-server-amd64.iso /datastore/pcc-000714/ubuntu-16.04.3-server-amd64.iso

Utilisez la commande get pour importer des fichiers de votre poste local vers votre datastore.

sftp> get /datastore/pcc-00714/ubuntu-16.04.3-server-amd64.iso /home/

La commande ***exit*** vous permet de fermer la connexion.

From:
http://chanterie37.fr/fablab37110/ - **Castel'Lab le Fablab MJC de Château-Renault**

Permanent link:
http://chanterie37.fr/fablab37110/doku.php?id=start:raspberry:sftp

Last update: **2023/01/27 16:08**