

Rsync: les bases

Comment synchroniser 2 répertoires avec *rsync* ?

- ⇒ [BackUp avec Rsync](#)
 - ⇒ [en local](#)
 - ⇒ [Pourquoi en local ?](#)
 - ⇒ [Options](#)
 - ⇒ [en réseaux](#)
 - ⇒ [en réseaux avec SSH](#)
 - ⇒ [en réseaux avec ssh sur un autre port](#)
- ⇒ [Restauration](#)
 - ⇒ [Delete](#)

1) BackUp avec Rsync

1.1) en local

Synchroniser permet en pratique d'effectuer des sauvegardes , depuis un poste Linux Unix-like vers un NAS ou un serveur. C'est super efficace, ça n'a pas besoin d'une grosse infra et ça nous évite de payer une licence à un éditeur pour une *solution* tel que **VeeamBackUp**.

Comment synchroniser 2 répertoires locaux avec rsync ?

- ⇒ Simplement avec cette commande:

```
rsync options <SRC directory> <DEST directory>
```

En plus clair : `rsync -avP --stats ~/dotfiles/_site/ ~/dev_web/_site`

- ⇒ NotaBene : lorsque le répertoire source est *fermé* par un *slash* cela signifie que c'est le contenu du répertoire qui va être synchroniser avec le répertoire de destination. Si le *slash* n'est pas présent c'est le répertoire lui même qui est synchronisé.

1.1.1) Pourquoi en local ?

Lorsque lance `jekyll serev` , et que je modifie du texte, *jekyll* régénère le site dans `~/dotfiles/_site` et l'URL est `http://localhost:4000` car il lance un serveur web pour que l'on admire le travail :trollface: .

Lorsque je veux publier sur `ordinatous.com` , il faut que j'adapte l'url en `https://site.ordinatous.com` .

1.2) Options

- ⇒ `-v` :(verbose) nous avons un retour console
- ⇒ `-r` :synchronise de manière récursive (descend dans l'arborescence)
- ⇒ `-a` :mode archive , conserve les attributs, liens symboliques, `-r` est facultatif dans ce cas
- ⇒ `-z` :compresse les données durant le transfert , pratique pour une synchro en réseaux
- ⇒ `-h` :human readable , renvoie des données lisible pour nous les humains
- ⇒ `-P` :affiche la progression
- ⇒ `-u` :synchronise les données les plus récentes
- ⇒ `-stats` :donne un résumé à la fin de l'execution

1.3) en réseaux

Comment synchroniser 2 répertoires en réseaux ?

- ⇒ Avec la commande suivante

```
rsync options <SRC directory> [utilisateur]@[nom ou IP de la machine]:<DEST directory>
```

En plus clair :

```
rsync -avzP --stats /home/georges_abitbol/projets/recherche_cancer
georges_abitbol@GEN_SRV_BKP:/backup_abitbol/projet/recherche_cancer
```

1.4) en réseaux avec SSH

⇒ Avec l'option `-e ssh` en reprenant la commande précédente.

```
rsync -avzP -e ssh --stats /home/georges_abitbol/projets/recherche_cancer
georges_abitbol@GEN_SRV_BKP:/backup_abitbol/projet/recherche_cancer
```

1.5) en réseaux avec ssh sur un autre port

Si le port ssh du serveur est 1022 .

```
rsync -avzP -e "ssh -p 1022" --stats
/home/georges_abitbol/projets/recherche_cancer
georges_abitbol@GEN_SRV_BKP:/backup_abitbol/projet/recherche_cancer
```

2) Restauration

Si nous avons synchronisé dans un sens, il est heureusement possible de le faire dans le sens inverse : on restaure les données simplement comme ceci:

```
rsync -auvzp
georges_abitbol@GEN_SRV_BKP:/backup_abitbol/projet/recherche_cancer
/home/georges_abitbol/projets/recherche_cancer
```

⇒ NotaBene : l'option `-u` pas franchement utile car il y a peu de chance que la donnée distante soit plus récente que celle sur le poste.

⇒ NotaBene2 : par défaut `rsync` n'effacera pas les données distantes si elles ne sont plus présente en local donc comment faire ?

2.1) Delete

L'option `-delete` va effacer les données distantes si elles n'existent plus en local.

```
rsync -auvzP -delete -e ssh --stats
/home/georges_abitbol/projets/recherche_cancer
georges_abitbol@GEN_SRV_BKP:/backup_abitbol/projet/recherche_cancer
```

⇒ Attention l'option `-u` va seulement synchroniser les données les plus récentes, et `-delete` va effacer les données distantes si elles sont absentes en local.

⇒ C'est clair ?