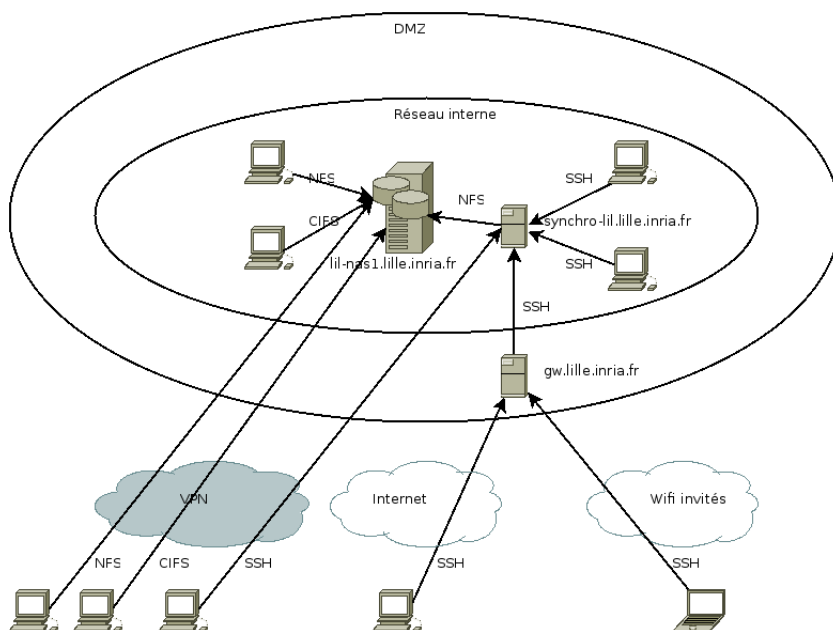


Sauvegardes Inria Lille - LG - Nov 2010

Disclaimer, sources, etc

Ce document a été fait très rapidement (pardonnez moi pour le style), et n'est pas garanti sans erreur. Il a été réalisé à partir des infos/images trouvées sur le site des moyens infos (MI) inria lille : <https://docmi.lille.inria.fr/> On suppose par la suite que l'on dispose d'un login inria et d'une machine Linux.

1 Plan du réseau inria Lille



Contenu Nous allons utiliser la machine de sauvegarde `synchro-lil.lille.inria.fr` :

1. Savoir s'y connecter : de l'intérieur, de l'extérieur.
2. Sauvegarder ses données sur son compte inria.

2 Se connecter à la machine synchro

Inside De l'intérieur du réseau, un simple ssh :

```
ssh gonnord@synchro-lil.lille
```

Outside - avec le VPN c'est plus facile! Avec le VPN, on se retrouve dans le réseau local, donc, pareil!

REMARQUE 1 Comment installer le VPN inria : https://dsi.inria.fr/services_offerts/vpn/Guide_utilisation/index_html (MI généraux inria)

Et si on ne veut pas le VPN, on fait un tunnel ssh avec la machine relais `gw.lille.inria.fr` et c'est expliqué là : <https://docmi.lille.inria.fr/accesdepuisexterieur/passerelle-ssh>

3 La synchronisation avec rsync et unison

3.1 Rsync

La commande rsync permet la synchronisation distante de répertoires, elle permet d'optimiser la bande passante utilisée en ne copiant que les fichiers distants modifiés depuis la dernière sauvegarde.

Un exemple

```
rsync -avzP -e ssh /home/laure/demosync/ gonnord@synchro-lil.lille.inria.fr:~/demo \
--exclude-from '/home/laure/bin/exclude.rsync'
```

(récursif)

avec dans `exclude.rsync` :

```
*.o
*~
```

REMARQUE 2 *Adapté aux sauvegardes dans un seul sens !*

3.2 Unison

Cf. <https://docmi.lille.inria.fr/donnees/synchronisation-avec-unison>

A la différence d'une copie (dans un sens seulement) la synchronisation s'occupe des deux côtés. On regarde les fichiers dans les deux arborescences pour voir s'il y a eu des changements et si oui dans quel sens il faut faire la copie. Les copies elles-mêmes utilisent l'algorithme de rsync.

Utilisation : dans le cas de plusieurs machines de travail, par exemple.

Démo Voici un fichier de config (`~/unison/testinria.prf`) pour la synchro machine perso / machine inria, à partir de la machine perso.

```
root = /home/laure/demosync/demounison
root = ssh://gonnord@synchro-lil.lille.inria.fr/demoUNISON
ignore = Name *~
ignore = Name .*~
ignore = Name *.o
```

Le seul souci avec unison c'est qu'il faut que des deux côtés de la synchro la version d'unison soit exactement la même, du coup, des fois, il faut compiler une version différente de celle du package linux dispo (le 1er nov 2010, la version d'unison sur synchro est la 2.27.57).

REMARQUE 3 *Au passage, unison a été écrit en OCaml! <http://www.cis.upenn.edu/~bcpierce/unison/>*

3.3 Récupération de fichiers grâce au système de sauvegarde

Un système de sauvegarde incrémental efficace a été mis en place par les MI, toutes les caractéristiques sont décrites sur la page :

<https://docmi.lille.inria.fr/donnees/nas>

La récupération des données est très facile :

<https://docmi.lille.inria.fr/donnees/snapshots>

4 Automatiser tout cela ?

L'outil `cron` et son copain `anacron` sont bien utiles pour automatiser des tâches à lancer toutes les heures, tous les mois, tous les reboot. La syntaxe des fichiers de config étant atroce, on peut utiliser par exemple l'utilitaire `gnome-schedule`.

Une synchro par jour (je suppose que l'on est dans le réseau local, et si on est à l'extérieur, le vpn est lancé – cela s'automatise aussi!)

1. créer un script shell accessible et exécutable : chez moi `/home/laure/bin/synchroinria.sh`.
2. utiliser `gnome-schedule` : `new, recurrent task, everyday`. Donner un nom à la tâche, mettre : `/home/laure/bin/synchroinria.sh` dans `command`, mettre "x application" (on aura besoin de donner le mot de passe!). Puis `add`.
3. on peut tester en exécutant cette tâche une fois (`run task`).