

# Administration GNU/Linux

## TP 3 et 4 : Administration Debian GNU/Linux

Nathaël Pajani  
&  
David Odin

IRC3 — 2013

### Objectifs :

- Se familiariser avec un système GNU/Linux.
- Gagner en autonomie et en aisance avec les systèmes GNU/Linux.
- Explorer les services proposés par un serveur et les mettre en place.
- Initiation à l'administration d'un système.

### Consignes :

- Le travail est à faire OBLIGATOIREMENT sur les clés USB fournies par nos soins sur les postes de l'école et en binôme.
- Le rendu est composé des diverses validations qui devront être réalisées au cours du TP sur un serveur dédié et de la clé USB utilisée.
- Le travail est donc terminé à la fin de la deuxième séance, mais il vous est possible de faire des recherches entre les deux séances.

### Note :

Être un bon administrateur système implique un boulot à plein temps. Nous ne sommes administrateurs systèmes qu'à temps (très) partiel, vous aurez donc plein de questions auxquelles nous ne saurons pas répondre immédiatement, mais n'hésitez pas à les poser, nous chercherons les réponses ensemble.

## Partie 1 : CPE : Commencez Par Exister : `cpe.start` (trente minutes max)

La première étape consiste à démarrer le PC qui se trouve devant vous en utilisant le système d'exploitation fourni sur la clé USB et non pas l'un des systèmes d'exploitation pré-installés.

Pour utiliser le système d'exploitation de la clé USB, vous devez démarrer le PC avec la clé connectée en face avant sur l'un des deux ports USB, puis lors de l'apparition du menu de sélection du système d'exploitation de GRUB, appuyer sur la touche 'c'.

GRUB (GRand Unified Bootloader) est ce que l'on appelle un *BootLoader*, ou chargeur de démarrage. C'est un programme qui sert à charger puis exécuter l'image d'un système d'exploitation, tout en fournissant des possibilités plus intéressantes que le BIOS (Basic Input Output System) qui se trouve généralement sur les ordinateurs, comme la possibilité de choisir entre plusieurs options, ou de disposer d'une ligne de commande pour charger une image manuellement. Plus d'informations ici : <http://www.gnu.org/software/grub/>

La ligne de commande que vous venez d'ouvrir va vous permettre de charger l'image qui se trouve sur la clé USB, à l'aide des trois commandes suivantes, qui demandent à GRUB d'utiliser l'image de noyau *vmlinuz* qui se trouve sur la première partition du deuxième disque (la clé USB) :

- **set root=(hd1,msdos1)**
- **linux /vmlinuz**
- **boot**

Lorsque l'invite de login apparaît, connectez vous en utilisant le seul utilisateur existant actuellement : *root* avec son mot de passe (*cpe*) et effectuer vos premières étapes d'administration de votre serveur : la création des comptes utilisateurs. Vous devez créer les comptes suivants, en utilisant la commande **adduser** :

- *admin*, avec le mot de passe de votre choix
- *eve*, avec comme mot de passe *cpe*

Changez maintenant le mot de passe root mais souvenez vous en bien, vous serez désormais les seuls à pouvoir administrer cette machine !

L'étape suivante est la dernière que vous effectuerez sur le poste qui se trouve devant vous, et consiste à lui donner un nom plus sexy que *usblive*, qui est actuellement le même sur toutes les machines. Choisissez si possible un mot simple, mais ayant une signification. Certains nomment les machines qu'ils administrent d'après des galaxies, des personnages de BD, ... à vous de choisir. La modification se fait avec les étapes suivantes :

- Remplacez le contenu du fichier */etc/hostname* par votre nouveau nom d'hôte.
- Utilisez la commande *hostname* pour changer le hostname courant : **hostname example**
- Modifiez le nom de la machine dans le fichier */etc/hosts* sur la première ligne, par exemple en utilisant la commande **sed** avec l'option **"-i"** pour changer immédiatement le contenu du fichier.

Le nouveau nom n'apparaît pas encore dans l'invite de commande, et c'est normal. Pourquoi ? Comment y remédier ?

Et enfin, déconnectez vous (commande *exit*), et changez de poste.

Sur votre nouveau poste, vous ne pouvez pas vous connecter en tant qu'administrateur, mais seulement en tant qu'élève, ce que vous allez faire. Vous pouvez changer le mot de passe de votre utilisateur si vous le désirez.

Utilisez un navigateur pour créer votre compte sur le serveur d'évaluation en vous connectant sur **cpe.start**, renseignez les différents champs et répondez aux questions.

## Partie 2 : CPE : Comment Poursuivre Efficacement : cpe.mouv (trente minutes max)

Vous vous trouvez désormais bien loin de la machine que vous allez devoir administrer. L'objectif est de vous mettre en situation réelle (ou presque, on évite habituellement de faire des expérimentations sur une machine en production). Attention à ce que vous faites, un autre groupe utilise la machine que vous administrez, et la machine sur laquelle vous êtes actuellement connectés est administrée par quelqu'un d'autre.

Reste à vous connecter à la machine que vous devez administrer. Une option serait d'avoir noté l'adresse IP de la machine que vous avez configurée, mais il existe des moyens plus performants : les serveurs DNS. Cependant, plutôt que d'installer un serveur DNS local utilisable et suffisamment dynamique, nous avons utilisé un petit script qui a mis à jour la table *hosts* de votre machine. Le résultat est le même.

Vous pouvez donc utiliser le nom d'hôte pour vous connecter à votre serveur.

La connexion se fait en **SSH**, qui est à la fois un protocole de communication sécurisé, et le programme permettant d'utiliser ce protocole, qui permet d'obtenir un shell sécurisé sur une machine. Pour l'utiliser :

- **ssh -p port -l user host**
- **ssh user@host**

Note : le protocole SSH inclue aussi un protocole de transfert de fichiers, utilisable avec la commande *scp* :

- **scp -P port local\_file user@host:distant\_file**
- **scp -P port -r local\_dir user@host:distant\_dir**
- **scp -P port user@host:distant\_file local\_file**
- ...

Ces commandes vous permettent donc d'administrer une machine distante. Cependant, si vous fermez le terminal utilisé sur votre machine locale (ou si votre administrateur fait planter votre machine ...), le shell distant est automatiquement fermé, et tout travail en cours est perdu. Fort heureusement, les systèmes GNU/Linux sont créés par des développeurs, pour des développeurs, et il existe une solution (plusieurs en réalité, mais nous n'en présenterons qu'une) qui permet de conserver un (ou plusieurs) shell ouvert même lorsque le terminal de contrôle est fermé : *screen*.

Lors de la première utilisation, il suffit d'exécuter la commande **screen**. Lorsque vous voulez vous re-connecter à une session existante, il faut utiliser la commande **screen -x**. Testez rapidement les effets de "*Ctrl-a d*", "*Ctrl-a c*", "*Ctrl-a I*" et "*Ctrl-a Ctrl-a*".

## Partie 3 : CPE : Contenu Pratique Essentiel : cpe.mini

Consigne supplémentaire :  
conf : tout doit être dans /etc et data dans /srv

Ne soyez pas en retard.  
ntpdate et cron

Soyez à jour  
apt-get, apt-get update, apt-get upgrade, fichier sources.list

Un peu de sécurité.  
iptables

Sauvez votre travail.  
mettre en place un backup incrémental pour /etc + /srv et rsync en démon over ssh pour "commiter" le développement d'un site web.

## **Partie 4 : CPE : C'est Pour Évaluation : cpe.eval**

Donnez vous de la visibilité.  
Installation apache + php + mysql + imagemagick + phpmyadmin, apache : virtual hosts, htaccess, mettre wordpress, mettre mediawiki

Communiquez.  
postfix + dovecot + postgrey + spamassassin + squirrelmail ou roundcube ?

Backups automatiques avec reporting.  
mettre en place backup incrémental pour mysql report par mail

Gestion des sources.  
git + gitweb : rendre téléchargeable les sources suivantes (bla.tar.gz) sous gitweb à l'adresse suivante ....

## **Partie 5 : CPE : Comment Puis-je Étudier : cpe.info**

Pour aller plus loin.  
webalizer / darkstat, unison ?, motd (esthétique), autre ?