

ADMINISTRER UN SYSTEME SERVEUR (2)

- Services
- Partages
- Démarrage

- Le démarrage d'un PC
- Le démarrage du système Linux
- Le processus init
- Les scripts /etc/rc
- Les sessions
- L'arrêt du système

Démarrage d'un PC

- Organisation d'un disque de PC
 - Plusieurs systèmes peuvent cohabiter
=> plusieurs partitions
 - 1 seul est actif à la fois, chargé en mémoire au démarrage

Démarrage d'un PC

- Organisation d'un disque de PC
 - La structure du MBR
 - Les 446 premiers octets contiennent un programme de chargement (loader) qui va démarrer l'exécution du programme de chargement propre au système actif
 - Les 64 octets qui suivent décrivent les partitions: taille, localisation, type et statut. Cette table ne contient que 4 entrées (primaires)
 - Les 2 derniers octets constituent le magic number, une valeur numérique que certains système utilisent pour vérifier la signature du secteur

Démarrage d'un PC

- Le processus de démarrage commence par l'exécution d'un programme de chargement (BIOS dans ROM). Ce programme charge le programme de chargement du MBR qui prend la relève (loader)
- Plusieurs programmes de chargement existent peuvent être installés dans la structure du MBR, ce programme choisit le système d'exploitation de la partition active ou offre la possibilité d'en démarrer un autre.

Démarrage d'un PC

- Les principaux chargeurs sont :
 - Linux LOader : tous les systèmes pour PC
 - Boot manager (IBM OS2) : charge un exécutable sur une partition séparée c'est lui qui propose le choix du système ensuite
 - NT Loader : recherche les systèmes à proposer dans un fichier de sa propre partition.
- Qu'est ce que LILO ?
 - Permet un démarrage automatique ou interactif, choix du noyau à charger et définitions des paramètres



Démarrage d'un système Linux

GTR

- Les composants de LILO
 - La commande `/sbin/lilo` installe les programmes de démarrage, installe le fichier `map`, configurée par `lilo.conf`
 - Le fichier `/boot/boot.b` (boot loader) contient les programmes de démarrage du noyau linux (chargeur) et permet de démarrer les autres systèmes
 - Le fichier `/boot/map` mémorise l'emplacement physique des blocs qui composent les programmes de démarrage
 - Le fichier de configuration `lilo.conf` spécifie les différents noyaux et définit les options et précise l'emplacement des programmes de démarrage



Démarrage d'un système Linux

GTR

- Le fichier `/etc/lilo.conf`
 - Un exemple :


```
# paramètres globaux
boot=/dev/hda
Delay=100
#paramètres de l' 'image
image=/boot/vmlinuz
root=/dev/hda1
label=linux
#paramètres de l' 'image
other=/dev/hda4
label=dos
```



Démarrage d'un système Linux

GTR

- Le fichier `/etc/lilo.conf`
 - Structure du fichier : en ligne de type `nom=valeur` image pour linux et other pour autre OS
 - Les paramètres globaux :
 - `boot` localisation de lilo
 - `install` chemin d'accès du chargeur
 - `map` chemin du fichier des blocs
 - `delay` en 10^{ème} de secondes
 - `default` nom de l'image chargée par défaut
 - Les paramètres spécifiques :
 - `image` chemin d'accès d'un noyau linux
 - `label` identificateur affiché
 - `root` la partition qui contient le SGF racine



Démarrage d'un système Linux

GTR

- La commande `lilo`
 - `lilo` installe le chargeur et crée le fichier `map`
- Mise en œuvre
 - destruction possible du système existant, sauvegarder !
 - `Dd if=/dev/hda of=/mnt/floppy/MBR bs=512 count=1`
 - `fdisk /mbr`
 - `dd if=/mnt/floppy/MBR of=/dev/hda bs=446 count=1`
- Utilisation avancée
 - Options `message=`, `append=`, `ro`, `password=`, ...



Démarrage d'un système Linux

GTR

- Démarrer à partir de la disquette réparation
 - `mkbootdisk`
 - `dd if=/mnt/cdrom/images/rescue.img of=/dev/fd0`
 - Booter avec la disquette de boot et taper `rescue`
- Démarrer avec LOADLIN
 - Copier l'utilitaire `LOADLIN` sur le disque DOS (`rawrite`)
 - Copier le noyau sur le disque dos (`vmlinuz`)
 - Configuration de windows



Le processus init

GTR

- L'activation des processus : `init`
- Principe de fonctionnement
 - Plusieurs niveaux de fonctionnement
 - A chaque niveau correspond des services
 - `init` exécute les services du niveau donné
 - niveau par défaut est donné par `inittab`
 - Il est possible de passer à LILO des paramètres pour `init`



Le processus init

GTR

- Les niveaux
 - 0: arrêt le système
 - 1: maintenance (nb restreints de services)
 - 2: mode multi-utilisateur : SGF, services, réseau sauf NFS
 - 3: service de partage de fichiers (NFS)
 - 4: multi-utilisateur spécifique (peu utilisé)
 - 5: connexion graphique : xdm
 - 6: arrêt et redémarrage du système
 - s,S: 1 seul utilisateur, processus fondamentaux démarrés
 - 7,8,9: A définir par l'utilisateur



Le processus init

GTR

- Le fichier /etc/inittab

Etiquette:Niveaux>Action:Commande

- Id: 4 caractères max identificateur de lignes
- Niveau: précise le niveau de la commande



Le processus init

GTR

- Le fichier /etc/inittab

Etiquette:Niveaux>Action:Commande

- Action: contexte dans lequel la commande est exécutée
 - sysinit 1 fois, démarrage à froid, fondamental
 - boot 1ère fois que l'on active l'un des niveaux
 - bootwait idem mais attend avant de poursuivre
 - initdefault 1ère ligne recherchée, niveau de démarrage
 - off ignoré
 - once activé à chq fois que init exécute inittab
 - wait idem avec attente
 - respawn once + si fin réexécuté



Le processus init

GTR

- Le fichier /etc/inittab

Etiquette:Niveaux>Action:Commande

- Commande : la ligne à exécuter, chemin absolu
- Les commandes :
 - runlevel affiche le niveau de fonctionnement
 - init change de niveau



Les scripts de démarrage

GTR

- Présents dans /etc/rc*
 - Vérification des FS
 - Montage des FS
 - Activation des quotas
 - Activation du swap secondaire
 - Lancement des daemons (crond, lpd)
 - Le réseau est activé (ifconfig, route, inetd)



Les scripts de démarrage

GTR

- Le script rc.sysinit
 - Initialisation de PATH
 - Initialisation du swap
 - Détermination du nom
 - Vérification du FS principal root et montage /proc
 - Montage des FS sauf NFS
 - Suppression des fichiers de verrous de services
 - Initialisation du port série
 - Chargement de pilotes dynamiques, les modules
 - Création de dmesg qui contient la config du syst



Les scripts de démarrage

GTR

- Init exécute ensuite les scripts spécifiques au niveau demandé
- Par défaut niveau=3 : principaux daemons crond lpd xinetd sendmail
- Le script rc.local
 - Modifié par l'administrateur pour exécuter des commandes propres au site tels des logiciels spécifiques.



Les scripts de démarrage

GTR

- Le script /etc/rc.d/rc exécute les actions du niveau passé en argument.
 - Deux boucles de programmes. Les scripts K* du répertoire /etc/rc.d/rc<n>.d qui permettent d'arrêter les services et les scripts S* du même répertoire pour les démarrer. Le nombre qui suit S ou K donne l'ordre d'exécution. Le reste du nom indique le nom du service. Ex: S40crond
 - Certains répertoires contiennent que des scripts K et d'autres que des scripts S
 - Commandes de configuration : chkconfig, ntsysv



Les scripts de démarrage

GTR

- Ajouter un script K* ou S*
 - L'administrateur qui souhaite ajouter un nouveau service n'a pas à modifier le fichier inittab mais doit intégrer un nouveau script S* ou K*
 - Ecrire un script Snnxx qui démarre le service et le placer dans le répertoire /etc/rc.d/rc3.d et un script Knnxx qui l'arrête dans /etc/rc.d/rc0.d en choisissant un nombre nn de sorte qu'il soit exécuté dans le bon ordre
 - Ecrire un script dans /etc/rc.d/init.d avec pour argument start ou stop. Créer les liens symboliques depuis les répertoires vers le script.
 - Intérêt : un seul exemplaire à modifier du script.
 - Le script K* n'est pas exécuté si le service n'est pas activé



Les étapes de la connexion

GTR

- A la saisie du nom de login, la commande getty exécute la commande login qui valide ou rejette la connexion en examinant le fichier /etc/passwd
- En cas d'échec, un message est affiché si le fichier /etc/nologin existe. (cf man login pour bien paramétrer en fonction des règles de sécurité)
- Si valide, le processus exécute la commande du fichier /etc/passwd généralement une console.
- Le shell exécute alors le fichier /etc/profile ou le .profile de l'utilisateur et le .bashrc (si \$TERM=bash)
- Le processus qui exécute le shell est toujours celui qui exécutait la commande getty et login. Init crée donc un nouveau processus qui exécute à nouveau la commande getty pour cette liaison.



Les étapes de la connexion

GTR

- Le fichier termcap
 - Ce fichier décrit les séquences de contrôle des attributs de la vidéo de la quasi totalité des terminaux de type texte (déplacement curseur ...)
 - Le contenu est difficile à lire. Un terminal y est décrit en plusieurs lignes. L'administrateur ne fait que vérifier si l'ensemble de son parc de terminaux sont bien reconnus.
 - Le programme qui a besoin d'utiliser ses séquences de contrôle doit trouver son nom dans la variable TERM.
 - Terminfo base de données représente une alternative à termcap



L'arrêt du système

GTR

- Les utilisateurs sont prévenus de l'arrêt
- Les applications sont arrêtées proprement
- Les sessions des utilisateurs sont stoppées
- L'intégrité du FS est assurée
- L'arrêt du système est réalisé par la commande shutdown qui avertit les utilisateurs toutes les 5min de l'arrêt imminent avant de basculer dans le niveau de fonctionnement 0 qui correspond à l'arrêt du système.
- La commande nous renvoie directement aux scripts de /etc/rc.d/rc0.d/*
- Shutdown -h now : halt
- Shutdown -r now : reboot