# Administration Système

## Administration d'un Système Linux

Olivier Dalle
Olivier.Dalle@sophia.inria.fr





**Plan**







**Plan**







**Plan**







### Tâches de l'Administration Système

- Installation et maintenance des machines
  - Hardware (configuration, gestion pannes)
  - Noyau (configuration, optimisation)
  - Outils et applications (compil, edit, …)
  - Services (impression, serveur web, …)
- Gestion du Réseau
  - Architecture, configuration
  - Détection et correction des pannes, congestions, …





### Tâches de l'Administration Système (2)

- Gestion du Système et des Utilisateurs
  - Ajout/Suppression d'utilisateurs
  - Login, mot de passe, groupe de travail, répertoire de travail, …
  - Configuration de l'environnement de travail
  - Organisation disque, gestion de l'espace, quotas
  - Surveillance/détection d'incidents, analyse des logs, …

## Tâches de l'Administration Système (3)

- Gestion de la sécurité
  - Sauvegardes des données
  - Limitations d'accès
  - Détection des intrusions
  - Application de « rustines » *(patchs)* pour boucher les « trous » de sécurité
  - Segmentation/cloisonnement du réseau
  - Sécurisation des données et protocoles sensibles





## Tâches de l'Administration Système (4)

- Information des utilisateurs
  - Planification des opération de maintenance, des interruptions de service
  - Mise en place de nouvelles applications ou services
  - Retrait des applications et services obsolètes
- Veille technologique
  - Lutter contre le vieillissement des matériels et logiciels (pbs de maintenance, de support)
  - Surveiller les tendances, anticiper les besoins des utilisateurs





## Les Différents Types de Systèmes

- Indépendant (*Standalone*)
  - Le système fonctionne tout seul sans avoir besoin d'autres systèmes
- Serveur
  - Offre des services qui permettent aux autres systèmes de fonctionner
  - La plupart des serveurs sont aussi de type indépendant





## Les Différents Types de Systèmes (2)

- *Dataless*
  - Ne disposant que d'un espace disque minimal, reservé au fonctionnement du système
  - Fortement dépendant de serveurs : sans réseau, les utilisateurs peuvent éventuellement se loguer, mais ne retrouvent pas leur données
- *Diskless*
  - Pas du tout de disque !
  - Totalement dépendant du réseau : sans réseau la machine ne démarre pas …





## Linux ou Unix, Quelles Différences ?

- Qu'est-ce que Unix ?
  - Système créé en 1969 chez AT&T
  - Fin des 70 : réécriture à Berkeley : BSD 4.1
  - Souche AT&T évolue vers System V
  - Souche BSD reprise chez Sun, DEC et HP
  - Début des 90 : multiples combinaisons des 2 souches, arrivée AIX et OSF/1, SunOS (BSD) devient Solaris (System V) …

  ⇒ La standardisation est indispensable !!





## Linux ou Unix, Quelles Différences ? (2)

- Les Standards du monde Unix :
  - Système V : SVR2, SVR3, SVR4, SVR4.2
  - BSD : 4.2BSD, 4.3BSD, 4.4BSD
  - ANSI/IEEE POSIX : P1003.1 (en-têtes, interface système) et P1003.2 (shells et utils) ; Standards FIPS ; Standard ISO/IEC
  - Spécification XPG : XPG3, XPG4, XPG4 Base, XPG4 Base 95
  - *Single UNIX Specification (v1, v2)*
  - UNIX * : UNIX 93, UNIX 95, UNIX 98

## Linux ou Unix, Quelles Différences ? (3)

- Qu'est-ce que Linux ?
  - Noyau Unix-like
    - Mélange/adaptation de diverses technos (Minix, sockets BSD, IPC System V, VFS Sun, …)
    - Développements spécifiques (ext2fs, procfs)
  - Ensemble de commandes GNU
    - G.N.U = Gnu's Not Unix !
  - Nombreuses « contrib » (utions)
    - Contribution = application « offerte » à la communauté
    - Paquetages : tgz, rpm, deb(ian)
  - Nombreuses « distrib » (utions)
    - Noyau + commandes + contribs + outils config





## Plan







## L'Organisation du disque

- Un disque dur est généralement découpé en « tranches », les **partitions :**
  - L'espace de stockage de chaque partition peut être **géré** séparément :
    - Zone réservée au fonctionnement du système
    - Zone réservée aux utilisateurs
    - Zone réservée à la mémoire virtuelle (swap)
  - Cohabitation de technologies différentes
    - Systèmes multi-boot (Linux, NT, Solaris, SCO, …)
    - Systèmes de fichiers hétérogènes (traditionnel, journalisé, etc)





## L'Organisation du Disque (2)

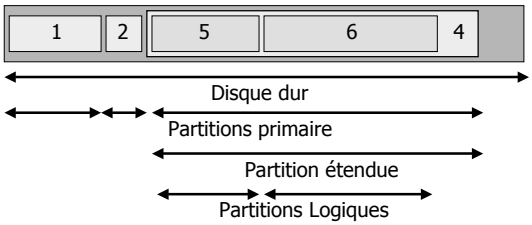
- Disques PC = 3 types de partitions
  - Partitions primaires
    - Subdivision primaire du disque
    - Au maximum 4, numérotées de 1 à 4
  - Partition étendue (une seule)
    - Partition primaire de type « container »
    - Prend la place de l'une des 4 partitions primaires
  - Partitions logiques
    - Subdivision secondaire : possible uniquement dans une partition étendue
    - Numérotées à partir de 5





## L'Organisation du Disque (3)

Exemple de découpage en partitions








## Logical Volume Manager

- Idée : un niveau intermédiaire entre
  - supports physiques (/dev/hdxy, /dev/sdzt,…)
  - supports/container de Syst. de Fichiers

➢ **VolumeGroupe/LogicalVolume**

1. Un VG fournit de l'espace sous forme de *Chunks*
2. Les part. physiques sont ajoutées au VG (apporte des *Chunks)*
3. Des part. logiques (LV) sont créées à partir du VG (consomme des *Chunks)*
4. Les SF sont créés dans les part. logiques

## Intérêt du LVM ?

- Flexibilité !
  - Les partitions logiques peuvent être retaillées à volonté
- Extensibilité (*scalability* )
  - On peut associer plusieurs part. physiques pour créer des part. > taille d'un disque
- Dynamicité/Tolérance aux pannes
  - Toutes les opérations peuvent avoir lieu "à chaud"





## Linux LVM : les commandes (man !)

- Commandes au niveau VG :
  - **vgcreate**, vgdisplay, **vgextend**, vgremove, vgreduce, vgsplit, …
- Commandes au niveau LV :
  - **lvcreate**, lvdisplay, **lvextend**, …
- Commande globales LVM (admin)
  - lvmchange (attrib), lvmsar (stats), …
- Projets en cours
  - ext2resize : idem resize_reiserfs pour ext2/3 (retailler ext2 sans démonter)





## Le Boot

- Etapes du démarrage du système depuis la mise sous tension jusqu'à l'état opérationnel du système
- Chargements successifs de plusieurs programmes :
  - Charger un gros programme comme le noyau est trop compliqué pour se faire en une seule fois
    - Trop gros pour loger dans une mémoire morte
    - Le programme de chargement du noyau est lui-même un programme … qu'il faut charger avec un programme chargeur !
  - La façon de charger le noyau doit être paramétrable :
    - Disquette, CD ou disque dur seul
    - Disquette ou CD puis disque dur
    - Réseau
    - …





## La Séquence de Boot (PC/Linux) : POST (Power On Self Test)

1. Chaque processeur s'initialise
   - self-test
   - Multi-pro : éventuellement élection d'un CPU leader
2. CPU leader exécute instruction en 0xffffffff0
3. Intel/PC : instruction en 0xffffffff0 = saut vers début du programme BIOS (Basic Input / Output System, implanté sur la carte mère)
4. BIOS : POST (Power On Self Test)
5. BIOS : Choix d'un périphérique de Boot





## La Séquence de Boot (PC/Linux): Le BIOS Passe le Relai

6. Le BIOS charge le MBR (Master Boot Record) du périphérique de boot
   - 1er secteur (512 octets) du périphérique
7. Le BIOS inspecte le MBR:
   - Vérification (nombre magique, table partitions)
   - Recherche le Secteur de Boot
   - NB: Deux scenarios selon que le MBR est installé ou non par Linux … Attention à la cohabitation avec d'autres OS !
8. Le BIOS charge le secteur de BOOT
   - Selon scénario, il peut s'agir du (début du) programme chargeur de Linux (LILO) ou du chargeur d'un autre OS





## La Séquence de Boot (PC/Linux): LILO Prend la Main et Charge le Noyau

9. Deux cas de figure :
   - Le programme du Boot sector termine le chargement de LILO (scénario MBR LILO)
   - Le programme du Boot sector déclenche le chargement d'un autre chargeur (NT, OS/2), qui à son tour chargera le 1er secteur de LILO
     - 1er secteur de LILO installé dans le secteur de Boot secondaire (en début de partition Linux)
     - On se retrouve dans la même situation que l'autre scénario : LILO termine son propre chargement
10. LILO charge le fichier contenant le noyau et lance son exécution

## La Séquence de Boot (PC/Linux): Le Noyau Lance Init

9. Le fichier noyau est généralement compressé!
   - Les 1ères instructions sont un programme de décompression qui
     1. Décompresse le reste du noyau
     2. Lance l'exécution du noyau décompressé
10. Le noyau commence son exécution :
    1. Initialisation du noyau, détection périph, …
    2. Montage de la racine en read-only
    3. Le noyau lance /sbin/init (processus 1)
11. Le processus init
    1. Lit le fichier /etc/inittab
    2. Exécute les scripts *rc (Run Control)*





## La Séquence de Boot (PC/Linux): Niveaux d'Exécution

- Le système (processus init) se trouve toujours dans un état (ou niveau) d'exécution appelé *runlevel :*
  - S, 0, 1, 2, 3, 4, 5, 6
  - Sous Linux (RedHat)
    - 0 – Halt
    - 1 – Single User mode
    - 2 – Multi-User mode, without NFS
    - 3 – Full multi-user mode
    - 4 – Unused
    - 5 – Full multi-user mode (X-based login)
    - 6 – Reboot (transitoire seulement)





## La Séquence de Boot (PC/Linux): Changement de *runlevel*.

- Lorsque *init* change de niveau, un certain nombre de tâches sont déclenchées :
  - En fonction des règles présentes dans le fichier /etc/inittab
  - En fonction des scripts de Run Control des répertoires /etc/rc.d/rc**n**.d (en fait, inittab aussi …)
    - SXXyyyyy : lors de l'entrée dans le niveau
    - KXXyyyyy : lors de la sortie du niveau

    Exemple :
    /etc/rc.d/rc3.d/S25netfs : montage partitions distantes
    /etc/rc.d/rc3.d/K20nfs : arrêt du service NFS





## La Séquence de Boot (Intel/Linux): Le fichier /etc/inittab

- Décrit les règles de fonctionnement du processus init
- Contient des entrées de la forme :

  id:niveau:action:process
  - id : identifiant (1 – 4 caractères)
  - niveau : indique le niveau d'exécution auquel la règle s'applique
  - action : une action parmis un ensemble d'actions prédéfinies
  - process : le processus à exécuter





## La Séquence de Boot (Intel/Linux): Le fichier /etc/inittab (2)

- Les actions possibles
  - respawn : relancer le processus
  - once : le processus est lancé une fois que le niveau d'exécution est atteint.
  - wait : idem once + init attend que le processus se termine
  - boot : exécuté lors du boot (runlevel ignoré)
  - bootwait : idem boot + attente terminaison
  - off : rien du tout …
  - ondemand : lorsque le nivequ « ondemand » correpondant (a,b ou c) est atteint





## La Séquence de Boot (Intel/Linux): Le fichier /etc/inittab (3)

- Les actions possibles
  - initdefault : indique le niveau d'exécution par défaut (process ignoré)
  - sysinit : exécuté durant le boot, mais avant boot et bootwait
  - powerwait, powerfail, powerokwait, powerfailnow : incident d'alimentation …
  - ctrlaltdel : action à exécuter lorsque init reçoit SIGINT (via ctrl+alt+del)
  - kbrequest : lorsqu'une combinaison spéciale est frappée au clavier

## La Séquence de Boot (Intel/Linux): Le fichier /etc/inittab (3)

- Exemple d'inittab (ancien linux) :

*id:1:initdefault:*
*rc::bootwait:/etc/rc*
*1:1:respawn:/etc/getty 9600 tty1*
*2:1:respawn:/etc/getty 9600 tty2*
*3:1:respawn:/etc/getty 9600 tty3*
*4:1:respawn:/etc/getty 9600 tty4*





## Plan







## Types d'Installation

- A partir des sources, en partant de rien
  - Long … long … long ! (For the braves ! :-)
  - Le meilleur moyen de vraiment comprendre, c'est de tout faire soit même !
  - Un système sur mesure = entièrement sous contrôle = sécurité
  - http://www.linuxfromscratch.org
- A partir d'une distribution
  - **RedHat**, Mandrake, Debian, Suse, Slackware





## Plan de Découpage en Partitions

- [impératif] Une partition pour la racine :
  - Tout ce qu'il faut pour booter en mode « single »
  - Idéalement prévue pour le minimum, cad de 30 à 100 Mo
- [recommandé] Une partition pour le swap
  - taille : au moins autant que la RAM, au max 2 fois la RAM (inutile d'en mettre plus)
  - recommandé mais pas indispensable : un fichier de swap peut toujours être crée sur une partition non dédiée…





## Plan de Découpage en Partitions (2)

- [recommandé] Une partition /usr
  - taille : minimum 400 Mo, maximum (actuellement) 1.5 Go
- [optionnel] Des partitions séparée pour /var, /usr/local et /opt
- [optionnel (linux)] Une partition /boot séparée
- [recommandé] Une partition /home avec ce qui reste





## Outils de Découpage en Partitions

- Outils linux :
  - fdisk : historiquement le premier. Pas très convivial, plutôt réservé aux utilisateurs avertis (habitués)
  - cfdisk : outil plus récent, plus convivial. Interface similaire au fdisk de msdos.
  - DiskDrake (mandrake), **DiskDruide** (RedHat) : outils de distribution conviviaux mais propriétaires

## Outils de Découpage en Partitions (2)

- Outils des autres systèmes (windows, …)
  - fdisk : l'outil rudimentaire du monde MSDOS
  - FIPS, Partition Magic : des outils avancés, qui permettent généralement le découpage de partitions (FAT) existantes





## Création des Partitions

- Méthode :
  1. Détruire les éventuelles partitions inutiles
  2. Créer les partitions linux
     1. Racine et swap
     2. Autres partitions
  3. Enregistrer (écrire la nouvelle table de partitions)
  4. Rebooter
- Si cohabition avec système « alien » :
  - Supprimer et retailler les partitions à partir du système alien, rebooter, PUIS terminer depuis linux
  - Installer partition linux à la suite des partitions aliens





## Création des Partitions (2)

- Création de la partition racine Linux :
  - De préférence une partition primaire
  - De préférence APRES les partitions « aliens »
  - Type = Linux FS (0x81)
  - Note: si LILO sur cette partition, activer le drapeau « bootable »





## Création des Partitions (3)

- Création de la partition de swap Linux :
  - Partition primaire ou secondaire
  - Si choix possible, préférer :
    - le disque le plus rapide de la machine.
    - le disque le moins sollicité (en fonctionnement linux)
  - Type = Linux swap (0x82)





## Création de Comptes

- Création manuelle :
  - Créer groupe utilisateur (/etc/group)

    group:passwd:gid:logins
  - Créer un homedir
    - (attention aux permissions !)
  - Créer entrée fichier /etc/passwd

    login:passwd:pid:gid:gecos:homedir:shell
  - Créer entrée /etc/shadow

    (Si système « shadow passwd » installé, voir plus loin)

    login:passwd:last:min:max:avis:inactif:expire:drapeau





## Création de Comptes (2)

- Création assistée :
  - useradd [-c comment] [-d homedir]
    [-e expire] [-f inactive]
    [-g initial_group] [-G group[, …]]
    [-m [-k skeleton_dir] | -M] [-p passwd]
    [-s shell] [-u uid [-o]] [-n] [-r] login
  - useradd -D … (destruction)

## Les Comptes « Système »

- Uid = 0
  - Super-utilisateur = tous les droits :
    - Fichiers et systèmes de fichiers
    - Périphériques (via fichiers)
    - Processus
  - En principe : login = root
  - MAIS éventuellement d'autres noms de login
    - Permet de donner l'accès root à plusieurs utilisateurs sans leur donner le même mdp
    - Permet d'attribuer des homedirs différents
    - …





## Les Comptes « Système » (2)

- Uid entre 1 et 99
  - Comptes sans possibilité de connexion (shell = /bin/false dans /etc/passwd)
    - daemon : pour donner une identité aux processus systèmes d'arrière-plan
    - bin : propriétaire des programmes
    - sys : propriétaire de certains fichiers système
    - adm : propriétaire de certains fichiers d'administration
    - lp : propriétaire du « spool » d'impression
  - Comptes avec connexion spéciale
    - halt : exécute le programme /sbin/halt qui arrête le système
    - Sync : exécute le programme sync





## Le Système « Shadow Password »

- Principe : les mots de passe sont stockés dans /etc/shadow plutôt que /etc/passwd
  - Champ n°2 devient inutile dans /etc/passwd (En général remplacé par 'x')
  - /etc/shadow est protégé en lecture
- Système plus « sécure »
  - Le fichier passwd **doit** être lisible de tous
  - MAIS l'accès à la forme codée des mots de passe présente un risque (attaque par dictionnaire …)
  - Shadow supprime le risque sans remettre en cause l'accès aux informations du fichier /etc/passwd
  - Shadow permet d'ajouter des informations de sécurité tout en restant compatible avec ancienne version





## Le Système « Shadow Password » (2)

- Format d'une entrée dans /etc/shadow

login:passwd:lastchg:min:max:avis:inactif:expire:drapeau

  - login : login
  - passwd : mot de passe codé
  - lastchg : date dernier changement mdp
  - min, max : nb jours minimum et maximum entre 2 changements
  - avis : nb jours en dessous duquel il faut averir l'utilisateur (qu'il doit changer de mdp)
  - inactif : nb de jours d'inactivité autorisés
  - expire : date d'expiration du compte
  - drapeau : inutilisé pour l'instant





## Installation de Paquetages

- Principe du paquetage : Ensemble de fichiers + scripts [ + règles de dépendance ]
- RedHat, Mandrake :
  - RPM (RedHat Package Manager)
  - Commandes : rpm, rpmdrake, …
- Debian :
  - fichiers .deb
  - Commandes : dselect (menu), installpkg, ar (!), …
- Slackware :
  - Fichiers .tgz = .tar.gz
  - Pas de règles de dépendance !
  - Commandes : setup (menu), pkginstall, tar (!), …





## Déroulement Typique d'une Installation par Distribution

1. Boot noyau d'install
2. Choix préliminaires :
   - Langue, type de clavier, …
3. Partitionnement
4. Création du/des SF + montage
5. Choix des paquetages à installer
6. Préconfiguration
7. Dépaquetage/installation
8. Postconfiguration

## Installation à partir de MSDOS ou CD non « bootable »

- Le CDROM RedHat contient un répertoire « dosutils »
  - FIPS pour redimensionner les partitions DOS
  - rawrite pour fabriquer des disquettes de démarrage à partir de fichiers présents sur le CD (répertoire « images »)
  - LOADLIN pour charger linux depuis MSDOS
    - Pour ne pas avoir à passer par LILO





## Installation Typique à Partir de Disquettes (RH)

- Une, Deux ou Trois Disquettes à créer (rawrite), selon type d'installation
  - installation à partir du CD
    - boot.img (normal),
  - Installation à partir du réseau (NFS, HTTP, …)
    - bootnet.img (install réseau)
  - Installation sur portable (CD pcmcia) :
    - pcmcia.img
  - Matériel exotique :
    - drivers.img (en plus d'une des 3 précédentes)