Systèmes hétérogènes

Procédure de démarrage d'un PC, login sous Linux, et changement mot de passe root

Auteurs :
Mohamed Amine NASSEH
Alexandre LEONARDI

Encadrant :
Roland AGOPIAN

12 octobre 2016

Résumé

Présentation en 3 parties liées au démarrage d'un ordinateur :
— mécanisme de démarrage d'un PC jusqu'à la recherche d'un OS
— mécanisme de démarrage d'un OS de type Linux
— procédure de changement de mot de passe root pour un système Linux en cas d'oubli

Table des matières

1 Bootstrap : Mécanisme de démarrage d'un PC

Bootstrap, qui se traduit littéralement par lanière de botte, est une expression des *Aventures du baron de Münchhausen* où le baron s'échappe de sables mouvants en tirant sur les lanières de ses propres bottes jusqu'à s'en extraire. Accessoirement c'est aussi le nom du processus de démarrage d'un ordinateur.

1.1 Impulsion initiale

Sur un appui du bouton de démarrage, un circuit électrique est fermé et une impulsion est envoyée de la carte mère à la PSU (Power Supply Unit, l'alimentation). La PSU va réaliser un test de bon fonctionnement et, si le résultat est positif, répondre par un signal électrique sur le connecteur alimentant le CPU (Central Processing Unit, le processeur).

Pendant cette première phrase, deux actions importantes sont effectuées consécutivement :

1. les registres du CPU sont remis à une valeur par défaut stockée dans un vecteur de reset. Il s'agit d'un pointeur, stocké dans de la mémoire non-volatile, par exemple une mémoire flash de type NVRAM (Non-Volatile Random-Access Memory) ou bien une mémoire de type EEPROM (Electrically Erasable Programmable Read-Only Memory). Cette adresse est dépendante du modèle de CPU.
2. le CPU exécute l'instruction maintenant pointée par ses registres. Cette instruction est la première instruction du BIOS, qui est un firmware (un logiciel de bas-niveau permettant le contrôle direct du matériel, stocké dans de la mémoire non-volatile) qui va prendre en charge la suite de l'initialisation de l'ordinateur.

1.2 BIOS : Basic Input/Output System

Le BIOS lui-même va effectuer un certain nombre de tâches avant de laisser la place à un système d'exploitation. Le BIOS dispose de sa propre mémoire non-volatile appelée CMOS (Complimentary Metal Oxide Semiconductor).

La première tâche du BIOS est d'initialiser le contrôleur du FSB (Front Side Bus, le bus système). Le FSB est le bus qui gère les échanges CPU/RAM (Random Access Memory, la mémoire vive) et pour l'initialiser, le BIOS récupère sa fréquence dans la CMOS qui correspond au nombre de bits de données échangeables par seconde au travers du FSB.

L'étape suivante consiste à récupérer le coefficient multiplicateur du CPU qui, appliqué à la fréquence de fonctionnement du FSB, donne la fréquence de fonctionnement du CPU. Cette fréquence permet d'initialiser le contrôleur du CPU. À ce stade, le BIOS est prêt à réaliser le POST.

1.2.1 POST : Power-On Self Test

Le POST est une procédure qui correspond à un certain nombre de vérifications matérielles faites par le BIOS :

— Intégrité du BIOS lui-même (checksum du CMOS et du BIOS, état des batteries)
— Bon fonctionnement de la RAM et du CPU
— Bon fonctionnement des disques durs

— Initialisation d'un clavier et d'une souris (si applicable)
— Présence de BIOS spécifiques à certains composants tels qu'une carte vidéo (le cas échéant, ils sont eux-mêmes exécutés)

Lorsqu'une erreur survient durant le POST, elle est signalée soit par un affichage à l'écran quand c'est possible, soit à défaut par un signal sonore, c'est-à-dire une suite de bips répondant à un code. Ce code n'est pas standard et dépend du constructeur du BIOS (typiquement Award, AMI ou Phœnix).

Le code "un bip" correspond néanmoins de manière standard à passage du POST avec succès. Dans ce cas le BIOS va tenter de trouver une partition bootable. Pour ce faire il va passer en revue l'ensemble des systèmes de stockage auxquels il a accès (CD-ROMs, disques durs, clés USB, disquettes (sait-on jamais...)) dans un ordre prédéterminé et stocké dans le CMOS.

1.2.2 MBR : Master Boot Record

Un disque (ou une clé USB, etc) est bootable si le premier secteur de mémoire contient un MBR. Un MBR est un secteur contenant le code du bootloader sur 448o et la table des partitions du disque sur 64o (la taille du secteur d'un disque est de 512o de manière standard). Le bootloader est un programme dont l'objectif est de trouver et charger en mémoire le noyau du système d'exploitation pour que celui-ci puisse ensuite prendre la main.

Le BIOS va donc déclencher l'exécution du bootloader qui a plusieurs tâches à accomplir :

1. Trouver, à l'aide de la table des partitions, la partition active (celle qui contient l'OS)
2. Trouver le secteur de début de la partition active (le secteur de démarrage)
3. Charger l'information contenue dans le secteur de démarrage en mémoire
4. Transférer le contrôle du code à exécuter au secteur de démarrage

Il est à noter qu'il n'y a pas de MBR sur une disquette : le secteur de démarrage est nécessairement le premier secteur de la disquette.

Par ailleurs, la taille du bootloader étant limité à 448o par la taille d'un secteur de disque, il existe des bootloaders chaînés (tels que GRUB, GRand Unified Bootloader, utilisé fréquemment par des distributions Linux) pour le cas où cet espace serait insuffisant. Le premier secteur du bootloader contient alors les informations nécessaires pour accéder aux secteurs suivants (à l'image d'une liste chaînée en C).

À partir de ce stade, le BIOS rend la main au noyau du système d'exploitation.

2 Mécanisme de démarrage d'un système Linux

Le démarrage d'un système Linux démarre à proprement parler quand le bootloader rend la main, après que toute la partie matérielle du PC ait été testée. À ce stade, deux éléments sont nécessaires au bon déroulement des opérations : une image du noyau Linux, encore appelé `kernel`, et une image `initramfs` (ou bien `initrd` dans de plus anciennes versions du kernel) qui permettra de simuler un système de fichiers temporaire dans la RAM.

Ces deux images ont été récupérées sur le disque et chargées en mémoire par le bootloader.

2.1 Kernel

La première tâche du kernel est une première phase de configuration matérielle écrite en assembleur qui permet à l'image de déclencher sa propre décompression.

Le kernel va ensuite configurer le matériel de l'ordinateur pour le rendre utilisable : configuration de la mémoire, du CPU, des différents systèmes d'Entrée/Sortie, ... Il va également décompresser et monter l'image de `initramfs` mentionnée plus tôt. Cela va servir à émuler le système de fichier root pour permettre au kernel de terminer son initialisation sans avoir à effectivement monter un disque physique.

`initramfs` fournit notamment au kernel les drivers nécessaires à l'utilisation des différents périphériques connectés à l'ordinateur et évite la situation où l'on aurait besoin d'un driver pour aller chercher un driver sur un disque physique.

À ce stade, le kernel est démarré et opérationnel mais encore aucune opération ne peut être effectuée. Le système de fichier apporté par `initramfs` est démonté et le véritable système de fichier root est monté à sa place. Puis le programme `init` est appelé.

2.2 Init

`init` est le nom le plus couramment donné au premier programme appelé par le kernel après son initialisation mais cela n'est pas obligatoire. Il va terminer le démarrage du système Linux en le rendant utilisable par un usager. Son nom complet est `/sbin/init`. Comme ce fichier n'est pas standard, le chapitre qui suit est susceptible de ne pas être exact pour toute distribution de Linux ; il a été rédigé grâce à la documentation de RedHat.

Une fois le programme `init` lancé, il devient le processus parent de tous les autres processus automatiques déclenchés à l'initialisation du système. Il commence par appeler le script `/etc/rc.d/rc.sysinit` qui initialise le PATH, la partition de swap, ainsi que d'autres étapes qui peuvent être spécifiques au système entrain de démarrer (initialisation de l'horloge par exemple *via* `/etc/sysconfig/clock`).

`init` appelle ensuite `/etc/inittab` qui est un script qui décrit la configuration que doit adopter le système en fonction du runlevel choisi. Les runlevels représentent des états du système associés à un entier, certains standard (0 correspond à l'arrêt du système) et d'autres non (2 n'a pas de signification par défaut). Le runlevel 5 correspond à un système multi-utilisateurs avec interface graphique.

C'est ensuite `/etc/rc.d/init.d/functions` qui est lu et utilisé par `init` pour configurer la manière de démarrer, tuer, et déterminer le PID d'un programme.

Enfin, l'ensemble des processus tournant en tâche de fond en fonction du runlevel choisi, qui ont éventuellement été configurés par l'utilisateur, sont lancés. Le dossier

`/etc/rc.d/rcX.d/`, où X est l'entier correspondant au runlevel, contient des liens symboliques vers l'ensemble de ces processus.

À ce stade, tous les processus pour qu'un utilisateur puisse se servir du système ont été lancés.

3 Procédure de modification de mot de passe root sous Linux

On trouve différentes solutions sur le net permettant d'arriver à cette solution, plus ou moins complexes de mises en œvre. En voici trois.

3.1 Mode utilisateur unique

Cette méthode marche pour tout bootloader, mais nous allons supposer qu'il s'agit d'un utilisateur utilisant GRUB.

Dans un premier temps, se placer sur la ligne correspondant à votre partition de boot classique (cf FIGURE 1) et appuyer sur "e" .

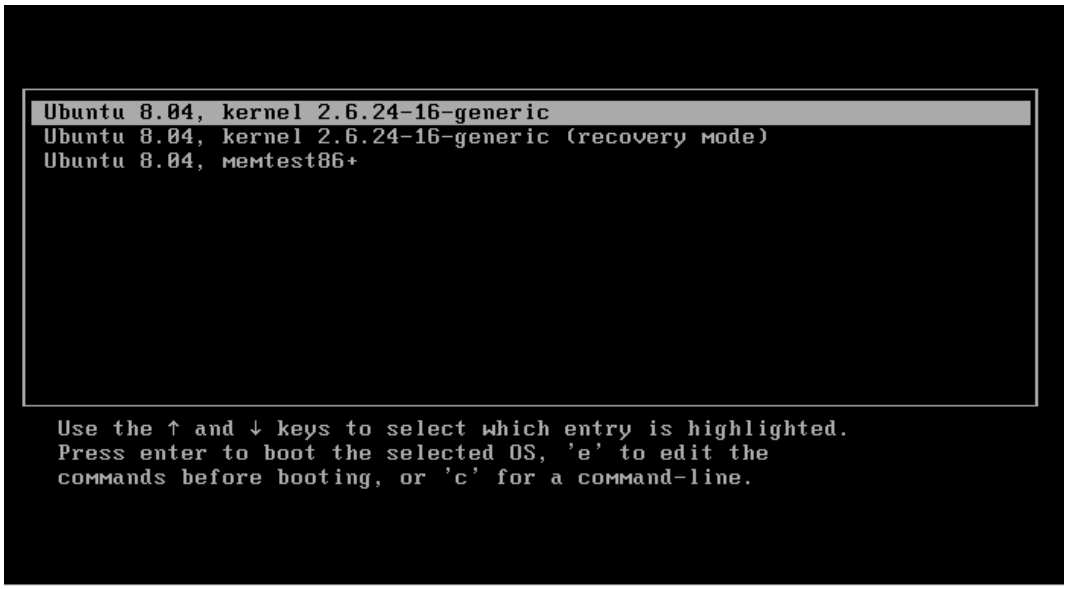

FIGURE 1 – Menu initial de GRUB

Puis se déplacer vers la ligne *kernel* (cf FIGURE 2) et appuyer de nouveau sur "e" pour l'éditer.

Il suffit maintenant de rajouter `single` à la fin de la ligne de commande, en prenant bien soin qu'il y ait un espace entre la fin du texte original et le début de `single`. Il est possible que votre système demande le mot de passe root pour se connecter en mode utilisateur unique ; dans ce cas rajouter encore `init=/bin/bash` après le `single`. Appuyer sur "entrée" pour valider les modifications.

Il suffit maintenant d'appuyer sur "b" pour booter en mode utilisateur unique et ainsi avoir accès à une invite de commande. Sur certains systèmes vous aurez besoin de monter es partitions `/` et |/proc|. Pour cela entrer la commande suivante : `mount -o remount,rw / mount -o remount,rw /proc`

Entrer maintenant la commande `passwd`. Le système va vous demander d'entrer un nouveau mot de passe root, puis de le confirmer (cf FIGURE 3). Il ne reste plus qu'à redémarrer et vous connecter avec le nouveau mot de passe.

3.2 Démarrer sur un disque bootable

Vous pouvez mettre en place cette solution avec le CD d'installation Linux que vous avez utilisé ou en en créant un nouveau (cela peut également être fait sur une

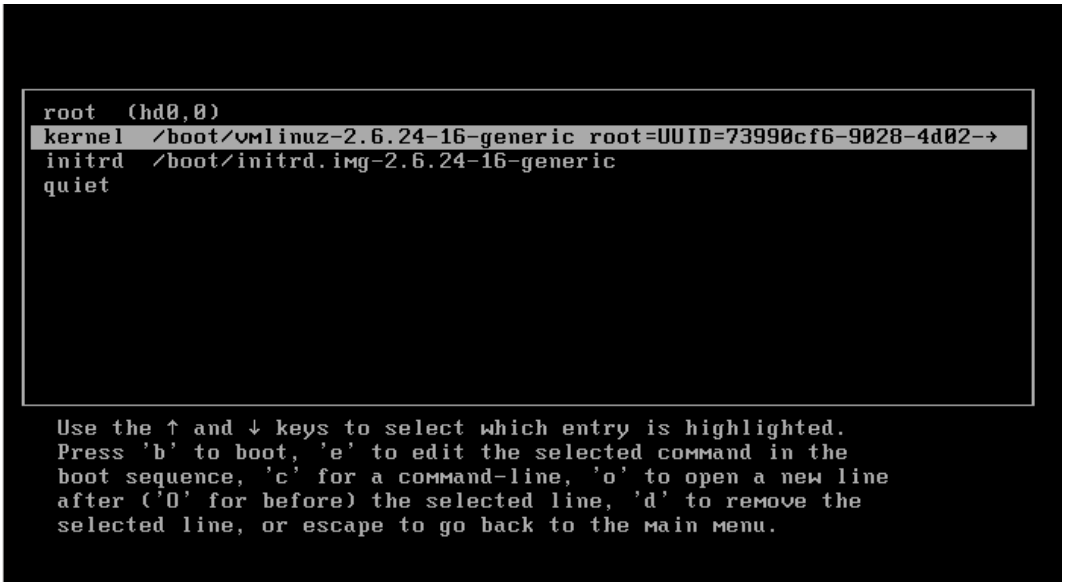

FIGURE 2 – Menu d'édition de GRUB

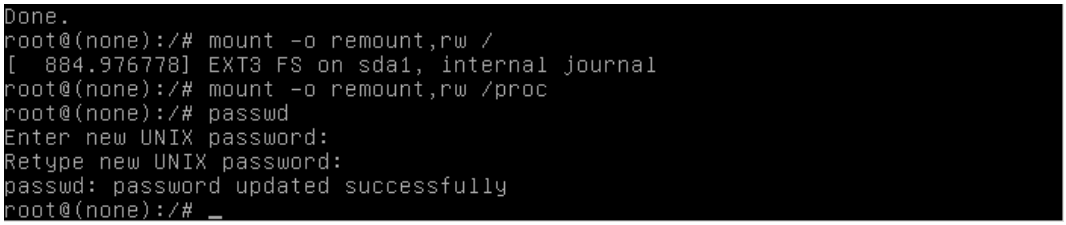

FIGURE 3 – Changement de mot de passe root

clé USB, à l'aide par exemple de Linux Live USB Creator et d'un .iso de votre distribution Linux). Démarrer sur ce disque comme lors de l'installation du système d'exploitation.

Une fois sur un bureau, plutôt que de lancer l'installation, ouvrir une console.

Taper `mkdir mountplace` pour créer un dossier nommé "mountplace" où le disque sera plus tard monté.

Taper `mount /dev/hdaX mountplace` (où X est le numéro associé à votre partition root) pour monter la partition concernée et pouvoir y accéder.

Taper `cd mountplace/etc` pour vous déplacer dans le dossier etc de votre partition.

Ouvrez le fichier *shadow* avec l'éditeur de texte de votre choix, par exemple grâce à la commande `vi shadow`

Il faut maintenant trouver une ligne contenant les informations de l'utilisateur root, qui devrait ressembler à `root:dsfDSDF!s:12581:0:99999:7:::`. Vous allez devoir enlever la partie entre le premier et le deuxième deux-points. Avec l'exemple donné, vous obtiendriez `root::12581:0:99999:7:::` (on voit que *dsfDSDF!s* a disparu).

Vous pouvez maintenant sauvegarder votre modification et quitter votre éditeur de texte, taper `cd` pour quitter le dossier etc, `unmount mountplace` pour démonter la partition et enfin redémarrer votre ordinateur.

Prenez soin de retirer le CD ou la clé USB que vous avez utilisé avant de redémarrer. Vous n'aurez pas besoin de votre mot de passe pour vous connecter à votre

compte. Pensez à en configurer un nouveau dès que possible !

3.3 Monter le disque sur un autre ordinateur

Cette dernière méthode est la plus complexe de mise en œuvre mais la plus sûre de marcher (à moins que le disque dur sur lequel votre système est installé soit encrypté, et que vous ayez oublié ce mot de passe également... ce qui confinerait à de la mauvaise volonté).

Dans un premier temps, mettre votre ordinateur hors-tension, ouvrir le boitier et retirer le disque dur (vous pouvez laisser passer quelques secondes ou minutes après la mise hors-tension pour que les condensateurs présents dans votre machine se déchargent).

Ensuite, brancher le disque dur sur un autre ordinateur, de préférence utilisant un système Linux aussi, car le format de fichier utilisé par Linux n'est pas reconnu par Windows et Mac.

Une fois le second ordinateur démarré, la démarche est la même qu'avec un disque bootable :

1. `mkdir mountplace` pour créer un répertoire où sera monté le disque que vous avez déplacé
2. `mount /dev/hdaX mountplace` où hdaX est le nom du disque que vous avez déplacé
3. `cd mountplace/etc` pour vous déplacer dans le répertoire etc de votre disque
4. `vi shadow` (ou tout autre éditeur de texte selon votre préférence) pour ouvrir le fichier shadow
5. localiser la ligne correspondant à l'utilisateur root, ressemblant à `root:dsfDSDF!s:12581:0:99999:7:::`
6. retirer la partie entre le premier et le deuxième deux-points, pour obtenir un résultat ressemblant à `root:s:12581:0:99999:7:::`
7. Sauvegarder la modification et quitter l'éditeur de texte
8. `cd` pour quitter le dossier etc
9. `unmount mountplace` pour démonter votre disque dur

Une fois tout ceci fait, vous pouvez éteindre l'ordinateur, débrancher le disque dur et le rebrancher dans l'ordinateur d'origine, puis remettre ce dernier sous tension.

Vous devriez alors pouvoir vous connecter à votre compte sans mot de passe. Là encore, pensez à renseigner un nouveau mot de passe root dès que possible.

4 Conclusion

Pour terminer ce rapport, il est bon de préciser que dans chacun des points abordés (démarrage d'un ordinateur jusqu'au chargement du noyau d'un OS, démarrage d'un système Linux, changement de mot de passe root sous Linux si celui-ci a été oublié), il est possible que l'explication donnée ne soit pas exhaustive.

Cela est dû à la multiplicité des possibilités en terme de matériel et de logiciel. De fait, tout couvrir en détail représenterait un travail bien plus titanesque que ce qui est possible dans le cadre de ce projet.

Un exemple de ce qui n'a pas été traité est le couple UEFI/GPT. UEFI (Unified Extensible Firmware Interface) est un remplaçant à la technologie BIOS dont les spécifications ont été publiées en 2005 et qui tend à être prédominant sur les nouvelles cartes mères. GPT (GUID Partition Table, GUID signifiant Globally Unique IDentifiers) est un standard de tables de partitions qui vise à remplacer MBR et fait lui-même partie du standard UEFI.

Ces deux technologies prennent une approche différente, mais comme d'autres, les traiter ici demanderait bien trop de temps. Il reste néanmoins que le présent rapport donne une vue d'ensemble de ce qui existe et apporte un degré de compréhension non-négligeable sur la procédure de démarrage d'un ordinateur Linux.

Références

[1] Post procedure. http://bios-setup.org/post-procedure/. Accessed : 2016-10-13.

[2] La procédure d'amorçage d'un pc. http://www.courstechinfo.be/OS/Boot.html. Accessed : 2016-10-12.

[3] How computer boots up? http://www.engineersgarage.com/tutorials/how-computer-pc-boots-up?page=1. Accessed : 2016-10-12.

[4] Inside the linux boot process. http://www.ibm.com/developerworks/library/l-linuxboot/index.html. Accessed : 2016-10-13.

[5] Recover - reset forgotten linux root password. https://linuxconfig.org/recover-reset-forgotten-linux-root-password. Accessed : 2016-10-13.

[6] How to reset forgotten root passwords. http://linuxgazette.net/107/tomar.html. Accessed : 2016-10-13.

[7] The boot process. http://www.opsschool.org/en/latest/boot_process_101.html. Accessed : 2016-10-13.

[8] Troubleshooting bios beep codes. http://www.pcguide.com/ts/x/sys/beep/index.htm. Accessed : 2016-10-13.

[9] Red hat enterprise linux 3 : Reference guide. https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/3/html/Reference_Guide/s1-boot-init-shutdown-sysv.html. Accessed : 2016-10-13.

[10] Master boot record. https://technet.microsoft.com/en-us/library/cc976786.aspx. Accessed : 2016-10-13.

[11] The system boot process explained. http://www.webopedia.com/DidYouKnow/Hardware_Software/BootProcess.asp. Accessed : 2016-10-12.

[12] Reset vector. https://fr.wikipedia.org/wiki/Front_side_bus. Accessed : 2016-10-13.

[13] Reset vector. https://en.wikipedia.org/wiki/Reset_vector. Accessed : 2016-10-13.