

Chap. 2 : Démarrage du système

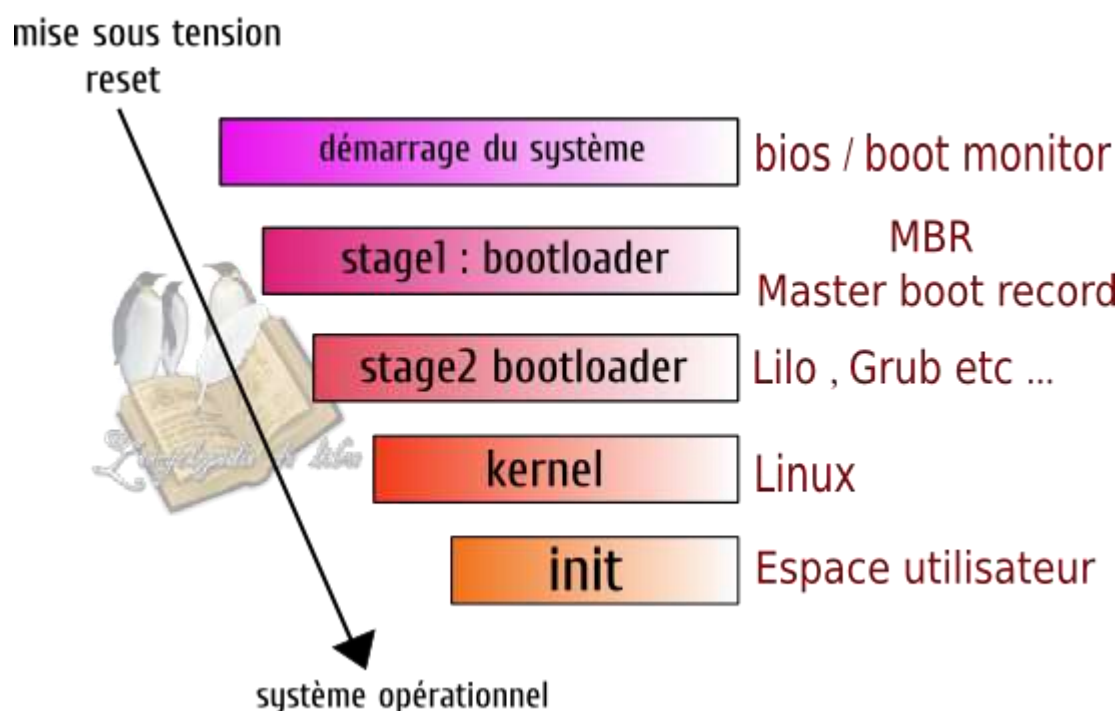
Le démarrage du système implique quatre programmes lancés successivement : le BIOS (*Basic Input Output System*), le chargeur de démarrage, le noyau et le processus init.

Le BIOS est le premier programme exécuté, il réside au niveau du ROM. Il charge les 512 premiers octets, ces 512 octets constituent le secteur d'amorçage ou le (Master Boot Record).

Le MBR contient plusieurs informations sur les partitions du disque. Il inclut aussi le chargeur de démarrage (ou une partie du chargeur de démarrage).

Le chargeur de démarrage Grub (Grand Unified Bootloader) est utilisé sur la plupart des distributions Linux. Il est décomposé en deux parties. La première partie réside sur le MBR, elle charge la deuxième partie qui se trouve dans une partition du disque. Une fois que la deuxième partie du Grub est chargée, une interface est affichée permettant à l'utilisateur de choisir quel système d'exploitation démarrer.

Le chargeur de démarrage exécute le noyau. Ce dernier continue le démarrage de la machine, il détecte et initialise les périphériques, monte la partition racine et démarre le processus init.



1. Processus init

a. Rôle

Le processus init est le premier processus utilisateur créé par le noyau lors du démarrage du système. Il utilise les niveaux d'exécution pour définir l'état du système à un instant donné. Chaque niveau d'exécution contient une liste de services à arrêter ou à démarrer.

Lors du démarrage du système, le processus init détermine, à partir du fichier de configuration /etc/inittab, le niveau d'exécution par défaut, et démarre les applications et les services requis dans ce niveau.

Lors de l'arrêt du système, init bascule vers le niveau d'exécution numéro 0. Ce niveau d'exécution est configuré de telle sorte que toutes les applications et tous les services seront arrêtés.

b. Niveaux d'exécution

Un niveau d'exécution, ou runlevel, est un état dans lequel se trouve Unix/Linux. Cet état est contrôlé par init. Chaque état dispose de sa propre configuration.

Niveau	Effet
0	Halt : stoppe le système d'exploitation, éteint la machine.

1	Mode mono utilisateur utilisé pour la maintenance, mode console.
2	Multi-Utilisateur, sans réseau, console.
3	Multi-Utilisateur, avec réseau, console.
4	Idem que le 3, laissé à la convenance de l'administrateur.
5	Multi-Utilisateur, avec réseau, avec environnement graphique X Window.
6	Reboot : redémarrage de la machine.
S,s	Single user mode, le mode le plus bas en cas de soucis.

Le niveau d'exécution par défaut est positionné dans /etc/inittab sur la ligne initdefault.

id:5:initdefault:

Remplacez 5 par le niveau souhaité au démarrage.

c. /etc/inittab

Le comportement du processus init et des runlevels est défini dans le fichier /etc/inittab. La syntaxe d'une ligne est la suivant :

Id : [niveau] : action : commande

Champ	Description
Id	Identifiant de ligne sur quatre caractères, juste indicatif.
Niveau	Indique si la commande doit être prise en compte pour le niveau demandé.
Action	Type d'action à effectuer selon les circonstances pour cette ligne.
Commande	La commande à exécuter avec ses paramètres et les redirections.

L'action est très importante car elle définit les activités de init lors du démarrage et du changement de niveau. Voici les principales :

Action	Signification
initdefault	Définit le niveau par défaut lors du boot et du lancement d'init.
sysinit	Exécuté une seule et unique fois lors du démarrage du système.
boot	Idem mais après sysinit.
bootwait	Idem, mais init attend la fin de l'exécution de la commande avant de continuer à dérouler le fichier inittab.
off.	La ligne est ignorée
once	La commande est exécutée à chaque changement de niveau pour les niveaux spécifiés.
Wait	Idem, mais init attend la fin de l'exécution avant de continuer.

respawn	La commande est lancée pour les niveaux concernés. Si le processus se termine, il est automatiquement relancé. C'est le cas pour les terminaux si un utilisateur s'en déconnecte
powerwait	La commande est lancée si le serveur passe sur alimentation de secours (UPS).
powerfail	Idem, mais sans attente de la fin d'exécution de la commande.
powerokwait	La commande est lancée lorsque le courant est rétabli.
powerfailnow	Commande de dernier recours lorsque l'alimentation de secours est presque vide.
ctrlaltdel	Init reçoit un signal SIGINT issu d'une séquence [Alt][Ctrl][Suppr].

EXEMPLE

Voici un exemple de fichier `/etc/inittab` :

```
# Le niveau d'exécution par défaut (le 2)
id:2:initdefault:
# Le premier script à démarrer
si::sysinit:/etc/init.d/rcS
# /etc/init.d exécute les scripts S et K lors
# d'un changement de mode de démarrage.
#
# Runlevel 0 est l'arrêt.
# Runlevel 1 est le mode simple utilisateur.
# Runlevels 2-5 sont des modes multi-utilisateur.
# Runlevel 6 est le mode de redémarrage.
l0:0:wait:/etc/init.d/rc 0
l1:1:wait:/etc/init.d/rc 1
l2:2:wait:/etc/init.d/rc 2
l3:3:wait:/etc/init.d/rc 3
l4:4:wait:/etc/init.d/rc 4
l5:5:wait:/etc/init.d/rc 5
l6:6:wait:/etc/init.d/rc 6
# Interceptor les touches CTRL-ALT-DELETE pour un redémarrage
ca::ctrlaltdel:/sbin/shutdown -t5 -rf now
# Création des différentes consoles (CTRL ALT F[1-6])
1:2345:respawn:/sbin/mingetty tty1
2:2345:respawn:/sbin/mingetty tty2
3:2345:respawn:/sbin/mingetty tty3
4:2345:respawn:/sbin/mingetty tty4
5:2345:respawn:/sbin/mingetty tty5
6:2345:respawn:/sbin/mingetty tty6
```

Le processus `init` utilise une série de scripts stockés sous les répertoires `/etc/rcN.d` où `N` est le niveau d'exécution. Par exemple le répertoire `/etc/rc3.d` contient les services du niveau d'exécution 3.

d. Changement de niveau :

Vous pouvez changer de niveau à la volée après le démarrage de la machine avec la commande `/sbin/init` ou `/sbin/telinit`, cette dernière étant un simple lien symbolique vers `init`. La commande suivante passe au niveau 5.

```
# telinit 5
```

Les valeurs `q`, `u` ou `-t` peuvent être précisées :

Q ou q : `init` relit le fichier `/etc/inittab`, s'il a été modifié, en corrigeant ses tables internes.

U ou u : init se relance sans relire inittab et sans changer de niveau. Si des services ont été rajoutés ou supprimés du niveau en cours, init prend en compte la modification.

t : quand init a terminé l'arrêt des services (ou plutôt quand le script rc l'a fait, voir un peu plus loin), init envoie le signal SIGTERM à tous les processus restants, leur demandant de se terminer proprement, attend le nombre de secondes spécifié (5 par défaut), puis envoie SIGKILL.

Le niveau d'exécution actuel est visible avec la commande **/sbin/runlevel**.

La première valeur retournée est le niveau précédent le niveau actuel. Un N signifie qu'il n'y a pas de précédent niveau. La seconde valeur est le niveau actuel.

runlevel

N 5

2. Gestion des services

Un serveur est conçu pour être constamment en activité et écouter les requêtes des clients, c'est le cas des serveurs de courriers électroniques, serveurs web ou serveurs de fichiers. Un serveur est aussi appelé *démon* ou *service*.

Les niveaux d'exécution sont utilisés par le système lors du démarrage pour savoir quels services démarrer et quels services arrêter.

Par exemple les fichiers dans le répertoire `/etc/rc3.d` sont des liens symboliques vers des scripts stockés dans le répertoire `/etc/init.d`. Ces scripts contiennent les procédures pour démarrer, arrêter ou déterminer l'état de chaque service :

```
lrwxrwxrwx. 1 root root 15 janv. 10 10:33 K15httpd -> ../init.d/httpd
lrwxrwxrwx. 1 root root 13 janv. 10 10:34 K20nfs -> ../init.d/nfs
lrwxrwxrwx. 1 root root 23 janv. 25 06:27 K32clamd.amavisd ->../init.d/clamd.amavisd
lrwxrwxrwx. 1 root root 15 janv. 10 10:44 K50snmpd -> ../init.d/snmpd
lrwxrwxrwx. 1 root root 19 janv. 10 10:44 K50snmptrapd ->../init.d/snmptrapd
lrwxrwxrwx. 1 root root 14 janv. 10 11:52 K74ntpd -> ../init.d/ntpd
lrwxrwxrwx. 1 root root 17 janv. 10 11:52 K75ntpddate -> ../init.d/ntpddate
lrwxrwxrwx. 1 root root 16 janv. 10 10:42 K84btseed -> ../init.d/btseed
lrwxrwxrwx. 1 root root 17 janv. 10 10:31 K90network -> ../init.d/network
lrwxrwxrwx. 1 root root 19 janv. 10 11:52 K95firstboot ->../init.d/firstboot
lrwxrwxrwx. 1 root root 18 janv. 10 10:29 S06cpuspeed -> ../init.d/cpuspeed
lrwxrwxrwx. 1 root root 19 janv. 10 10:28 S08ip6tables ->../init.d/ip6tables
lrwxrwxrwx. 1 root root 18 janv. 10 10:28 S08iptables -> ../init.d/iptables
lrwxrwxrwx. 1 root root 16 janv. 10 10:34 S11auditd -> ../init.d/auditd
lrwxrwxrwx. 1 root root 21 janv. 10 10:28 S11portreserve ->../init.d/portreserve
lrwxrwxrwx. 1 root root 17 janv. 10 10:31 S12rsyslog -> ../init.d/rsyslog
lrwxrwxrwx. 1 root root 14 janv. 10 10:33 S25cups -> ../init.d/cups
lrwxrwxrwx. 1 root root 24 janv. 10 10:44 S27NetworkManager ->../init.d/NetworkManager
lrwxrwxrwx. 1 root root 14 janv. 10 10:34 S55sshd -> ../init.d/sshd
lrwxrwxrwx. 1 root root 17 janv. 11 15:35 S65dovecot -> ../init.d/dovecot
lrwxrwxrwx. 1 root root 22 janv. 25 14:59 S78spamassassin ->../init.d/spamassassin
lrwxrwxrwx. 1 root root 17 janv. 25 09:20 S79amavisd -> ../init.d/amavisd
lrwxrwxrwx. 1 root root 17 janv. 11 10:15 S80postfix -> ../init.d/postfix
lrwxrwxrwx. 1 root root 15 janv. 10 10:32 S90crond -> ../init.d/crond
lrwxrwxrwx. 1 root root 13 janv. 10 10:40 S95atd -> ../init.d/atd
lrwxrwxrwx. 1 root root 11 janv. 10 10:31 S99local -> ../rc.local
```

Le préfixe du nom de chaque lien définit son ordre de lancement ou son ordre d'arrêt. Le nom est sous la forme suivante :

[SK]nnservice

S : start.

K : kill (stop).

nn : ordre numérique de démarrage ou d'arrêt. (00=premier, 99=dernier).

service : nom du service.

Les services peuvent être lancés à partir des scripts du répertoire `/etc/init.d`. Ces scripts peuvent être appelés avec différents arguments :

- start : pour démarrer le service ;

- stop : pour arrêter le service ;
- restart : pour arrêter puis redémarrer le service ;
- reload : pour envoyer un signal SIGHUP au processus du service en cours d'exécution. Ce signal le force à relire ses fichiers de configuration ;
- status : indique si le service est en cours d'exécution.

EXEMPLE

La séquence de commandes ci-dessous démarre puis arrête le service postfix.

```
# /etc/init.d/postfix start
Démarrage de postfix :
# /etc/init.d/postfix stop
Arrêt de postfix :
```

3. Gestion des services sous Debian et ses dérivés

Sous Débian et ses dérivés on utilise la commande update-rc.d pour maintenir les liens symboliques vers les scripts de démarrage dans les répertoires /etc/rcN.d

SYNTAXE

update-rc.d nom_service { start | stop } numéro_sequence numéros_niveaux_execution .

La commande update-rc.d a comme arguments :

- un numéro de séquence, utilisé par init pour décider de l'ordre d'exécution des scripts ;
- la liste des niveaux d'exécution applicables, impérativement suivie par un point.

EXEMPLE

Pour configurer le service d'impression cups de telle sorte qu'il démarre dans les niveaux d'exécution 2, 3, 4 et 5 avec un ordre de 80 et s'arrête dans les niveaux d'exécution S, 1 et 6 avec un ordre de 20 :

```
# update-rc.d cups start 80 2 3 4 5 . stop 20 S 1 6 .
Adding system startup for /etc/init.d/cups ...
/etc/rc1.d/K20cups -> ../init.d/cups
/etc/rc6.d/K20cups -> ../init.d/cups
/etc/rc8.d/K20cups -> ../init.d/cups
/etc/rc2.d/S80cups -> ../init.d/cups
/etc/rc3.d/S80cups -> ../init.d/cups
/etc/rc4.d/S80cups -> ../init.d/cups
/etc/rc5.d/S80cups -> ../init.d/cups
```

Quand on appelle update-rc.d avec l'option remove les liens dans les répertoires /etc/rcN.d qui pointent vers le script correspondant du répertoire /etc/init.d seront supprimés. Ce script doit auparavant avoir été lui-même supprimé. S'il est toujours présent dans /etc/init.d, update-rc.d affiche un message d'erreur indiquant qu'il faut utiliser l'option -f pour forcer la suppression du script, comme l'illustre l'exemple suivant.

EXEMPLE

```
# update-rc.d atd remove
update-rc.d: /etc/init.d/atd exists during rc.d purge (use -f to force)
# update-rc.d -f atd remove
Removing any system startup links for /etc/init.d/atd ...
/etc/rc1.d/K11atd
/etc/rc2.d/S89atd
/etc/rc3.d/S89atd
/etc/rc4.d/S89atd
/etc/rc5.d/S89atd
```

4. Upstart

Plusieurs distributions Linux récentes, y compris les dernières versions d'Ubuntu et de Fedora, ont remplacé le système classique init d'UNIX système V, SysV init, basé sur les niveaux d'exécution, par un nouveau système Upstart. Upstart est un système de gestion de services fonctionnant avec les événements. Il supervise les services pendant que le système fonctionne. Ainsi il démarre ou arrête des services en réponse aux événements tels que l'ajout d'un périphérique.

Pour un besoin de compatibilité, il émule également les niveaux d'exécution traditionnels de init.

Un système qui utilise Upstart remplace à la fois le fichier `/etc/inittab` et les répertoires spécifiques des niveaux d'exécution par les scripts placés dans le répertoire `/etc/init` (ce répertoire a été appelé `/etc/event.d` sur des versions antérieures d'Upstart).

5. Récupération du système

Des problèmes de MBR ou de système de fichiers de la partition racine empêchent Linux de démarrer. L'administrateur doit donc utiliser un CD de dépannage permettant de démarrer le système à partir d'un noyau Linux sur ce CD, ensuite procéder à la réparation des problèmes de démarrage du système.

a. Récupération du chargeur de démarrage Grub

Si on n'obtient pas l'écran de Grub au démarrage du système, il se peut que le MBR (Master Boot Record) soit endommagé, il faut donc réinstaller Grub sur le MBR du disque d'amorçage

PROCEDURE

Démarrage du système à partir du premier CD d'installation de la distribution. La distribution CentOS 6.0 propose le menu suivant permettant un démarrage en mode dépannage (*figure 1*). Ce mode est aussi disponible dans les autres distributions Linux.

Figure 1. Menu de démarrage en mode dépannage de la distribution CentOS

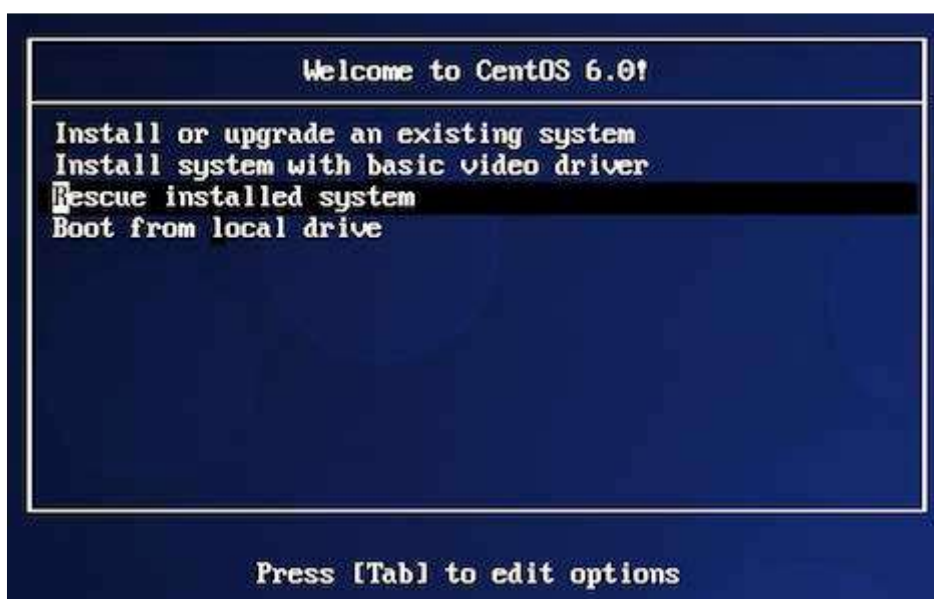


Figure 2. Configuration de la langue



Figure 3. Configuration du type de clavier

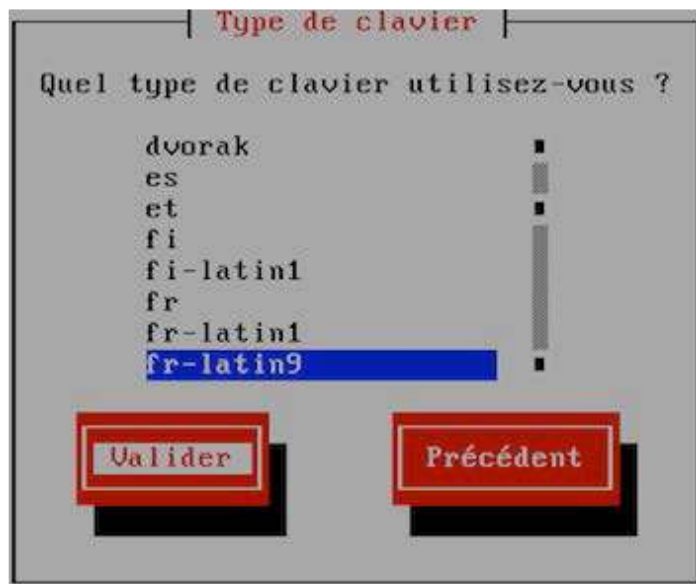


Figure 4. Montage du système : le mode dépannage cherche l'installation Linux et la monte sous le répertoire /mnt/sysimage



- Changement de répertoire racine (chroot) : le prompt « # » apparaît, on définit le répertoire /mnt/sysimage comme la nouvelle racine « / » du système :

```
# chroot /mnt/sysimage
```
- Installation du Grub sur le MBR avec les paramètres définis dans le fichier de configuration grub.conf :

```
# /sbin/grub-install /dev/sda
Installation finished. No error reported.
This is the contents of the device map /boot/grub/device.map.
Check if this is correct. If any of the lines are incorrect, fix it and re-run the
script 'grub-install.'
# this device map was generated by anaconda
(hd0) /dev/sda
```

Lors du redémarrage du système on devra voir l'écran du Grub permettant de sélectionner le système d'exploitation à démarrer.

Dans le cas où la partition de démarrage est endommagée, la commande grub-install ne fonctionne pas car elle ne peut pas localiser le fichier de configuration grub.conf. On doit donc utiliser le shell Grub pour localiser la partition contenant l'installation précédente du Grub puis le réinstaller sur le MBR :

- Exécution de la commande grub, le prompt change indiquant que le shell Grub est lancé :

```
# grub
GNU GRUB version 0.97 (640K lower / 3072K upper memory)
[ Minimal BASH-like line editing is supported. For the first word, TAB
lists possible command completions. Anywhere else TAB lists the possible
completions of a device/filename.]
grub >
```

- Recherche de la partition contenant l'installation du Grub :

```
grub> find /grub/stage1
find grub/stage1
(hd0,0)
```

Le résultat de la commande précédente indique que l'installation du Grub est sur la première partition du premier disque dur.

- Réinstallation du Grub sur le MBR :

```
grub> setup (hd0)
setup (hd0)
Checking if "/boot/grub/stage1" exists... No
Checking if "/grub/stage2" exists... yes
Checking if "/grub/e2fs_stage1_5" exists... yes
Running "embed /grub/e2fs_stage1_5 (hd0)"... 26 sectors are embedded.
succeeded
Running "install /grub/stage1 (hd0) (hd0)1+26 p (hd0,0)/grub/stage2
/grub/grub.conf"... Succeeded
Done.
```

b. Récupération de la partition racine

Si le système de fichiers de la partition racine n'est pas démonté correctement, la commande fsck est lancée automatiquement lors de redémarrage du système afin de vérifier et corriger les systèmes de fichiers.

On peut aussi vérifier le système de fichiers manuellement avec le mode de dépannage lancé à partir du premier CD d'installation de la distribution.

Il est recommandé d'exécuter la commande fsck sur un système de fichiers démonté. Comme on ne peut pas démonter le système de fichiers racine d'un système en cours d'exécution, la solution est de remonter la partition racine en lecture seule :

```
# mount -o remount,ro /
```

Ensuite vérifier et corriger le système de fichiers de la partition racine :

```
# fsck /
```