

Dernière mise-à-jour : 2020/01/30 03:28

LPI202 - Gestion du Démarrage et de l'Arrêt du Système (9/60)

Ce cours couvre le contenu des trois sections suivantes du cursus LPI 201-450 :

- 202.1 - Customizing SysV-init system startup (3/60)
- 202.2 - System Recovery (4/60)
- 202.3 - Alternate Bootloaders (2/60)

202.1 - Customizing SysV-init system startup (3/60)

Description: Candidates should be able to query and modify the behaviour of system services at various targets / run levels. A thorough understanding of the systemd, SysV Init and the Linux boot process is required. This objective includes interacting with systemd targets and SysV init run levels.

Key Knowledge Areas

- Systemd
- SysV init
- Linux Standard Base Specification (LSB)

Terms and Utilities

- /usr/lib/systemd/
 - /etc/systemd/
 - /run/systemd/
 - systemctl
 - systemd-delta
 - /etc/inittab
-

- /etc/init.d/
- /etc/rc.d/
- chkconfig
- update-rc.d
- init and telinit

202.2 - System Recovery (4/60)

Description: Candidates should be able to properly manipulate a Linux system during both the boot process and during recovery mode. This objective includes using both the init utility and init-related kernel options. Candidates should be able to determine the cause of errors in loading and usage of bootloaders. GRUB version 2 and GRUB Legacy are the bootloaders of interest. Both BIOS and UEFI systems are covered.

Key Knowledge Areas

- BIOS and UEFI
- NVMe booting
- GRUB version 2 and Legacy
- grub shell
- boot loader start and hand off to kernel
- kernel loading
- hardware initialisation and setup
- daemon/service initialisation and setup
- Know the different boot loader install locations on a hard disk or removable device.
- Overwrite standard boot loader options and using boot loader shells.
- Use systemd rescue and emergency modes.

Terms and Utilities

- mount
 - fsck
-

- inittab, telinit and init with SysV init
- The contents of /boot/, /boot/grub/ and /boot/efi/
- EFI System Partition (ESP)
- GRUB
- grub-install
- efibootmgr
- UEFI shell
- initrd, initramfs
- Master boot record
- systemctl

202.3 - Alternate Bootloaders (2/60)

Description: Candidates should be aware of other bootloaders and their major features.

Key Knowledge Areas

- SYSLINUX, ISOLINUX, PXELINUX
- Understanding of PXE for both BIOS and UEFI
- Awareness of systemd-boot and U-Boot

Terms and Utilities

- syslinux
 - extlinux
 - isolinux.bin
 - isolinux.cfg
 - isohdpfx.bin
 - efiboot.img
 - pxelinux.0
-

- pxelinux.cfg/
- uefi/shim.efi
- uefi/grubx64.efi

Présentation

Le processus de démarrage de Linux peut être résumé en trois étapes majeurs :

- Le **firmware** ou **micrologiciel** démarre en effectuant un test rapide du matériel, appelé un **Power-On Self Test** ou **POST**, puis recherche le **Chargeur de Démarrage** (*Bootloader*) à exécuter à partir d'un support bootable,
- Le Chargeur de Démarrage est exécuté et il détermine quel noyau Linux à charger,
- Le noyau se charge en mémoire et commence à exécuter en arrière plan les programmes nécessaires au fonctionnement du système.



A retenir : Il est possible de consulter le défilement des messages lors du démarrage en appuyant sur la touche **Echap** ou simultanément sur les touches **Ctrl** + **Alt** + **F1**. En sachant que la liste des messages se défilent rapidement, il est possible de les consulter **après** le démarrage du système à l'aide de la commande **dmesg** qui lit les derniers messages contenu dans le **Kernel Ring Buffer**. Ces messages sont aussi copiés dans le fichier **/var/log/boot.log**.

Cette description simpliste résume cependant un processus bien plus compliqué que ce cours va détailler.

BIOS, EFI et OpenFirmware

Systèmes à base du BIOS

Au démarrage d'un système à base d'un processeur x86 ou x86-64, le premier programme exécuté a été traditionnellement le BIOS. Le BIOS a pour fonction de :

- Tester les composants et les circuits,

- Faire appel au BIOS de la carte graphique pour initialiser le système d'affichage,
- Détecter les périphériques de stockage,
- Lancer le **Chargeur de Démarrage** du système d'exploitation en utilisant le **bootstrap loader**.

Chargeur de Démarrage

La première partie du Chargeur de Démarrage est en règle générale placé dans le MBR du disque. Le format du MBR est le suivant :

- 446 octets pour le Chargeur de Démarrage,
- 64 octets pour la table de partitions, soit 16 octets par partition décrite,
- 2 octets ayant une valeur fixe en hexadécimale de **AA55**.

Systèmes à base de l'EFI

Depuis 2011, le BIOS est en train d'être remplacé par l'utilisation de l'**UEFI** (**U**nified **E**xtensible **F**irmware **I**nterface ou *Interface micrologicielle extensible unifiée*) issue du développement de l'EFI conçue par Intel pour les processeurs Itanium..

Sous EFI la première partie du gestionnaire de démarrage est un fichier ayant une extension .efi se trouvant dans un sous-répertoire au nom du système d'exploitation à lancer dans une partition appelée **EFI System Partition** ou **ESP**. Cette partition est normalement montée à **/boot/efi** sous Linux.

Pour que EFI fonctionne, le micrologiciel (**firmware**) d'EFI doit avoir connaissance de chaque système d'exploitation à démarrer.



A retenir : Sous Linux c'est l'application **efibootmgr** qui permet de créer et de supprimer des entrées ainsi que de modifier l'ordre de démarrage.



Important : L'UEFI gère parfaitement les **SSD** (*Solid State Drives*) qui utilisent le standard **NVMe** (*Non-Volatile Memory Express*). Linux supporte les SSD depuis le noyau 3.3.

Secure Boot

HERE

Dans le cas de Secure Boot, le gestionnaire de démarrage

Autres Systèmes

Les systèmes utilisant des processeurs autre qu'un x86 ou x86-64 utilisent un logiciel tel [OpenFirmware](#).

Chargeurs de Démarrage

Des gestionnaires de d'amorçage sous Linux, un se distingue comme étant le plus utilisé :

- GRUB (Grand Unified Boot Loader)

Cependant il en existe d'autres :

LILO

LILLO (*Linux LOader*) est configuré par le fichier **/etc/lilo.conf**.

La commande LILLO

La commande **lilo** peut prendre une de plusieurs options. Les options les plus importantes sont :

Option	Description
-M	Permet d'écrire sur le MBR

Option	Description
-d	Permet de réduire ou augmenter le temps d'attente avant le lancement du noyau par défaut
-D	Permet de sélectionner un noyau par défaut en indiquant son label
-u	Permet de désinstaller LILO
-v	Permet d'activer le mode verbose
-m	Permet de modifier le fichier map par défaut (/boot/map)
-i	Permet de spécifier un nouveau fichier à utiliser comme secteur de boot (/boot/boot.b)
-C	Permet de modifier le fichier de configuration par défaut
-q	Permet de créer le fichier /boot/map qui contient l'emplacement des noyaux qui peuvent être booter

Codes Erreur de LILO

Lors du démarrage, LILO permet d'identifier les éventuelles erreurs :

Affichage	Erreur
(rien)	Aucun morceau de LILO n'a été chargé. Soit LILO n'est pas installé, soit la partition sur laquelle son secteur d'amorce se trouve n'est pas active.
L	Le premier morceau du chargeur d'amorce a été chargé et démarré, mais il ne peut charger le second morceau. Les codes d'erreur à deux chiffres indiquent le type de problème. (Voir également la section "Codes d'erreur disque".) Ce cas indique en général une panne de périphérique ou une incohérence de géométrie (c'est à dire de mauvais paramètres disques).
LI	Le premier morceau du chargeur d'amorce a pu charger le second morceau, mais n'a pas réussi à l'exécuter. Cela peut être causé par une incohérence de géométrie ou par le déplacement de /boot/boot.b sans lancer l'installateur de carte.
LIL	Le second morceau du chargeur d'amorce a été démarré, mais il ne trouve pas la table de descripteurs dans le fichier carte. C'est en général dû à une panne de périphérique ou une incohérence de géométrie.
LIL?	Le second morceau du chargeur d'amorce a été chargé à une adresse incorrecte. C'est en général causé par une subtile incohérence de géométrie, ou par le déplacement de /boot/boot.b sans lancer l'installateur de carte.
LIL-	La table de descripteurs est corrompue. Cela peut être dû à une incohérence de géométrie ou au déplacement de /boot/map sans lancer l'installateur.
LILO	Tous les éléments de LILO ont été correctement chargés.

Si le BIOS signale une erreur lorsque LILO essaye de charger une image d'amorce, le code d'erreur correspondant est affiché. Ces codes vont de 0x00 à 0xbb. Reportez-vous au Guide Utilisateur de LILO pour leur explication.



Important : LILO ne gère pas les systèmes UEFI.

Grub Legacy sous RHEL/CentOS 6



A retenir : Grub Legacy a été créé en **1999**.

Dans le cas où le Chargeur de Démarrage **grub** n'est pas installé, il convient de saisir la commande suivante :

```
# grub-install /dev/sda [Entrée]
```

où **sda** est le nom du périphérique où grub doit s'installer dans le MBR. Notez cependant que le MBR a une taille trop petite pour contenir tout le Chargeur de Démarrage. Pour cette raison, le gestionnaire est divisé en deux. Le gestionnaire de niveau 1 est stocké dans le MBR et indique où se trouve le gestionnaire de niveau 2, c'est-à-dire le répertoire `/boot`.

Il est aussi possible d'utiliser la commande :

```
# grub-install '(hd0)' [Entrée]
```

où **hd0** indique à grub le premier disque.

Il est à noter que la première partie du Chargeur de Démarrage peut également être installé dans un **PBR** (*Partition Boot Record*) dans le cas d'un système dual-boot voire multi-boot au quel cas on parle de **chainloading** :

```
# grub-install /dev/sda1 [Entrée]
```

ou :

```
# grub-install 'hd(0,0)' [Entrée]
```

où **(0,0)** est la nomenclature grub pour la première partition du premier disque.

Le gestionnaire de niveau 1 a pour seul but le lancement du gestionnaire de niveau 2. Le gestionnaire de niveau 2 charge le noyau en mémoire, monte l'image **initrd** et charge les modules nécessaires pendant que le noyau monte la partition racine / en lecture seule.

Pour désinstaller grub du MBR, utilisez une disquette DOS pour démarrer la machine puis taper la commande suivante au prompt :

```
A> fdisk /mbr [Entrée]
```

Configurer GRUB Legacy

grub se configure grâce au fichier **/boot/grub/menu.lst**. Pour visualiser ce fichier, il convient de saisir la commande suivante :

```
[root@centos6 ~]# cat /boot/grub/menu.lst
# grub.conf generated by anaconda
#
# Note that you do not have to rerun grub after making changes to this file
# NOTICE:  You have a /boot partition.  This means that
#           all kernel and initrd paths are relative to /boot/, eg.
#           root (hd0,0)
#           kernel /vmlinuz-version ro root=/dev/sda2
#           initrd /initrd-[generic-]version.img
#boot=/dev/sda
default=0
timeout=5
splashimage=(hd0,0)/grub/splash.xpm.gz
hiddenmenu
title CentOS Linux (2.6.32-71.29.1.el6.i686)
    root (hd0,0)
    kernel /vmlinuz-2.6.32-71.29.1.el6.i686 ro root=UUID=e73735d6-c14b-4a40-8735-f34fc868da8a rd_NO_LUKS
rd_NO_LVM rd_NO_MD rd_NO_DM LANG=fr_FR.UTF-8 SYSFONT=latarcyrheb-sun16 KEYBOARDTYPE=pc KEYTABLE=fr-latin9
```

```
crashkernel=auto rhgb quiet
    initrd /initramfs-2.6.32-71.29.1.el6.i686.img
title centos (2.6.32-71.el6.i686)
    root (hd0,0)
    kernel /vmlinuz-2.6.32-71.el6.i686 ro root=UUID=e73735d6-c14b-4a40-8735-f34fc868da8a rd_NO_LUKS rd_NO_LVM
rd_NO_MD rd_NO_DM LANG=fr_FR.UTF-8 SYSFONT=latarcyrheb-sun16 KEYBOARDTYPE=pc KEYTABLE=fr-latin9 crashkernel=auto
rhgb quiet
    initrd /initramfs-2.6.32-71.el6.i686.img
```



Important : Sous les distributions dérivées de Red Hat, par exemple Fedora, le fichier menu.lst s'appelle **grub.conf**.

Ce fichier comporte plusieurs sections :

- * la section des ****paramètres globaux****,
- * une ou plusieurs sections pour chaque système d'opération installé sur la machine.

La Section Globale

Paramètre	Explication
default=0	Ce paramètre désigne le numéro de l'entrée à charger par défaut. La valeur de 0 indique la première section commençant par le mot clef title
timeout=5	Ce paramètre indique le délai en secondes après lequel l'entrée par défaut sera chargée.
splashimage=(hd0,0)/grub/splash.xpm.gz	Ce paramètre indique l'emplacement de l'image de fond du menu de GRUB Legacy
hiddenmenu	Ce paramètre cache le menu de GRUB Legacy pendant le chargement de l'entrée par défaut, sauf si l'utilisateur appuie sur une touche
color	Ce paramètre prend la forme de deux paires de couleurs. Le premier, par exemple white/blue, définit les couleurs de l'avant-plan et de l'arrière-plan des entrées normales du menu, tandis que le deuxième, par exemple yellow/blue définit les couleurs de l'avant-plan et de l'arrière-plan des entrées sélectionnées du menu
password=motdepasse	Cette option n'est présente que dans le cas où un mot de passe a été spécifié pour protéger GRUB Legacy

Une Section spécifique à un OS

Paramètre	Explication
title CentOS Linux (2.6.32-71.29.1.el6.i686)	Ce paramètre indique le début d'une section de configuration d'une entrée pour un système d'exploitation ainsi que le nom qui apparaît dans le menu de GRUB Legacy
root (hd0,0)	Ce paramètre indique la partition contenant le noyau de Linux. Dans l'exemple hd0,0 indique la première partition du premier disque dur. Cette partition est ensuite montée en tant que /boot .
kernel /vmlinuz-2.6.32-71.29.1.el6.i686	Ce paramètre indique le nom du noyau à charger pour démarrer la machine. Son chemin est relatif à la partition de démarrage (hd0,0). Certaines options peuvent être passées au noyau en les spécifiant en tant qu'arguments telles rhgb ou Red Hat Graphical Boot et quiet qui supprime les messages de démarrage qui apparaissent avant le lancement de l'animation graphique activée par l'option rhgb
initrd	Ce paramètre stipule l'emplacement du disque initial chargé en mémoire lors du démarrage. Son chemin est relatif à la partition de démarrage (hd0,0).
lock	Ce paramètre indique que la section est protégée par le mot de passe défini dans les paramètres globaux
rootnoverify (hd0,1)	Ce paramètre indique une section concernant un système d'exploitation non Linux, tel que Windows™

Configurer l'Authentification

Grub Legacy peut être protégé par un mot de passe en incluant la directive suivante dans le fichier `/boot/grub/menu.lst` :

```
password --md5 <mot de passe>
```

Le mot de passe doit être chiffré avec MD5 :

```
[root@centos6 ~]# grub-md5-crypt
Password:
Retype password:
$1$VL0zG$ibdk0my4IHny/XtNIGRhv1
```

Editez ensuite le fichier **/boot/grub/menu.lst** :

```
[root@centos6 ~]# cat /boot/grub/menu.lst
```

```
# grub.conf generated by anaconda
#
# Note that you do not have to rerun grub after making changes to this file
# NOTICE:  You have a /boot partition.  This means that
#           all kernel and initrd paths are relative to /boot/, eg.
#           root (hd0,0)
#           kernel /vmlinuz-version ro root=/dev/sda2
#           initrd /initrd-[generic-]version.img
#boot=/dev/sda
default=0
timeout=5
password --md5 $1$VL0zG$ibdk0my4IHny/XtNIGRhv1
splashimage=(hd0,0)/grub/splash.xpm.gz
hiddenmenu
title CentOS (2.6.32-504.1.3.el6.i686)
    lock
    root (hd0,0)
    kernel /vmlinuz-2.6.32-504.1.3.el6.i686 ro root=UUID=b9f29672-c84e-4d3b-b132-189758a084eb rd_NO_LUKS rd_NO_MD
LANG=fr_FR.UTF-8 SYSFONT=latarcyrheb-sun16 crashkernel=auto KEYBOARDTYPE=pc KEYTABLE=fr-latin9 rd_NO_LVM
rd_NO_DM rhgb quiet
    initrd /initramfs-2.6.32-504.1.3.el6.i686.img
title CentOS (2.6.32-358.18.1.el6.i686)
    lock
    root (hd0,0)
    kernel /vmlinuz-2.6.32-358.18.1.el6.i686 ro root=UUID=b9f29672-c84e-4d3b-b132-189758a084eb rd_NO_LUKS
rd_NO_MD LANG=fr_FR.UTF-8 SYSFONT=latarcyrheb-sun16 crashkernel=auto KEYBOARDTYPE=pc KEYTABLE=fr-latin9
rd_NO_LVM rd_NO_DM rhgb quiet
    initrd /initramfs-2.6.32-358.18.1.el6.i686.img
title CentOS (2.6.32-358.6.1.el6.i686)
    lock
    root (hd0,0)
    kernel /vmlinuz-2.6.32-358.6.1.el6.i686 ro root=UUID=b9f29672-c84e-4d3b-b132-189758a084eb rd_NO_LUKS rd_NO_MD
LANG=fr_FR.UTF-8 SYSFONT=latarcyrheb-sun16 crashkernel=auto KEYBOARDTYPE=pc KEYTABLE=fr-latin9 rd_NO_LVM
rd_NO_DM rhgb quiet
```

```
initrd /initramfs-2.6.32-358.6.1.el6.i686.img
title CentOS (2.6.32-279.el6.i686)
  lock
  root (hd0,0)
  kernel /vmlinuz-2.6.32-279.el6.i686 ro root=UUID=b9f29672-c84e-4d3b-b132-189758a084eb rd_NO_LUKS rd_NO_MD
LANG=fr_FR.UTF-8 SYSFONT=latarcyrheb-sun16 crashkernel=auto KEYBOARDTYPE=pc KEYTABLE=fr-latin9 rd_NO_LVM
rd_NO_DM rhgb quiet
initrd /initramfs-2.6.32-279.el6.i686.img
```



Important : Notez l'addition de la ligne **password -md5 \$1\$VLOzG\$ibdk0my4IHny/XtNIGRhv1** ainsi que le mot clef **lock** sur une ligne située après chaque ligne commençant par **title**.

Modifier la Configuration de GRUB Legacy en Ligne de Commande

Lors du démarrage de GRUB Legacy, il est possible de voir son menu en appuyant sur n'importe quelle touche. Si GRUB Legacy a été protégé par un mot de passe, il convient d'appuyer sur la touche **p** puis de rentrer le mot de passe. A ce stade il est possible d'utiliser deux autres touches :

- la touche **e** pour accéder à l'éditeur de l'interface,
- la touche **c** pour accéder à la ligne de commande.

En mode édition notez l'utilisation des touches suivantes :

- **e** : éditer une ligne,
- **d** : supprimer une ligne,
- **o** : ajouter une ligne après la ligne courante,
- **O** : ajouter une ligne avant la ligne courante,
- **b** : démarrer avec la configuration modifiée,
- **echap** : abandonner les modifications et retourner à l'interface de GRUB.

GRUB 2 sous RHEL/CentOS 7

GRUB 2 est une ré-écriture complète de GRUB Legacy datant de **2005**. Il apporte des améliorations, notamment GRUB 2 sait utiliser des partitions RAID et LVM, mais aussi des modifications importantes.

Le lancement de GRUB 2 se fait en trois étapes :

- Etape 1 : Le **boot.img**, stocké dans les 512 premiers octets du secteur 0 avec la table des partitions, est lancé. Son seul but est de lancer l'étape 1.5,
- Etape 1.5 : Le **core.img**, d'une taille approximative de 25 Ko et stocké dans les secteurs 1 à 62, est lancé. Son travail est de charger des pilotes qui supportent de multiples systèmes de fichiers puis de lancer l'étape 2 dans un des systèmes de fichiers,
- Etape 2 : Contenu dans le répertoire **/boot/grub2/**, il lance le menu pour que l'utilisateur puisse choisir le système d'exploitation à lancer.

Dans le cas où le Chargeur de Démarrage **GRUB 2** n'est pas installé, il convient de saisir la commande suivante :

```
# grub2-install /dev/périphérique [Entrée]
```

où **périphérique** est le nom du périphérique où l'étape 1 de GRUB2 doit s'installer dans le MBR.

GRUB 2 lit ses entrées de menus à partir du fichier **/boot/grub2/grub.cfg**. Pour visualiser ce fichier, il convient de saisir la commande suivante :

```
[root@centos7 ~]# cat /boot/grub2/grub.cfg
#
# DO NOT EDIT THIS FILE
#
# It is automatically generated by grub2-mkconfig using templates
# from /etc/grub.d and settings from /etc/default/grub
#

### BEGIN /etc/grub.d/00_header ###
set pager=1

if [ -s $prefix/grubenv ]; then
  load_env
```

```
fi
if [ "${next_entry}" ] ; then
    set default="${next_entry}"
    set next_entry=
    save_env next_entry
    set boot_once=true
else
    set default="${saved_entry}"
fi

if [ x"${feature_menuentry_id}" = xy ]; then
    menuentry_id_option="--id"
else
    menuentry_id_option=""
fi

export menuentry_id_option

if [ "${prev_saved_entry}" ]; then
    set saved_entry="${prev_saved_entry}"
    save_env saved_entry
    set prev_saved_entry=
    save_env prev_saved_entry
    set boot_once=true
fi

function savedefault {
    if [ -z "${boot_once}" ]; then
        saved_entry="${chosen}"
        save_env saved_entry
    fi
}

function load_video {
```

```
if [ x$feature_all_video_module = xy ]; then
    insmod all_video
else
    insmod efi_gop
    insmod efi_uqa
    insmod ieee1275_fb
    insmod vbe
    insmod vga
    insmod video_bochs
    insmod video_cirrus
fi
}

terminal_output console
if [ x$feature_timeout_style = xy ] ; then
    set timeout_style=menu
    set timeout=5
# Fallback normal timeout code in case the timeout_style feature is
# unavailable.
else
    set timeout=5
fi
### END /etc/grub.d/00_header ###

### BEGIN /etc/grub.d/10_linux ###
menuentry 'CentOS Linux (3.10.0-229.4.2.el7.x86_64) 7 (Core)' --class centos --class gnu-linux --class gnu --
class os --unrestricted $menuentry_id_option 'gnulinux-3.10.0-123.el7.x86_64-advanced-b35de665-5ec8-4226-
a533-58a1b567ac91' {
    load_video
    set gfxpayload=keep
    insmod gzio
    insmod part_msdos
    insmod xfs
    set root='hd0,msdos1'
```

```
    if [ x$feature_platform_search_hint = xy ]; then
        search --no-floppy --fs-uuid --set=root --hint-bios=hd0,msdos1 --hint-efi=hd0,msdos1 --hint-
baremetal=ahci0,msdos1 --hint='hd0,msdos1' e8d3bd48-1386-411c-9675-41c3f8f1a309
    else
        search --no-floppy --fs-uuid --set=root e8d3bd48-1386-411c-9675-41c3f8f1a309
    fi
    linux16 /vmlinuz-3.10.0-229.4.2.el7.x86_64 root=UUID=b35de665-5ec8-4226-a533-58a1b567ac91 ro
vconsole.keymap=fr crashkernel=auto vconsole.font=latarcyrheb-sun16 rhgb quiet LANG=en_US.UTF-8
    initrd16 /initramfs-3.10.0-229.4.2.el7.x86_64.img
}
menuentry 'CentOS Linux, with Linux 3.10.0-123.el7.x86_64' --class centos --class gnu-linux --class gnu --class
os --unrestricted $menuentry_id_option 'gnulinux-3.10.0-123.el7.x86_64-advanced-b35de665-5ec8-4226-
a533-58a1b567ac91' {
    load_video
    set gfxpayload=keep
    insmod gzio
    insmod part_msdos
    insmod xfs
    set root='hd0,msdos1'
    if [ x$feature_platform_search_hint = xy ]; then
        search --no-floppy --fs-uuid --set=root --hint-bios=hd0,msdos1 --hint-efi=hd0,msdos1 --hint-
baremetal=ahci0,msdos1 --hint='hd0,msdos1' e8d3bd48-1386-411c-9675-41c3f8f1a309
    else
        search --no-floppy --fs-uuid --set=root e8d3bd48-1386-411c-9675-41c3f8f1a309
    fi
    linux16 /vmlinuz-3.10.0-123.el7.x86_64 root=UUID=b35de665-5ec8-4226-a533-58a1b567ac91 ro vconsole.keymap=fr
crashkernel=auto vconsole.font=latarcyrheb-sun16 rhgb quiet LANG=en_US.UTF-8
    initrd16 /initramfs-3.10.0-123.el7.x86_64.img
}
menuentry 'CentOS Linux, with Linux 0-rescue-a2feb9eb09b1488da0f23b99a66350f8' --class centos --class gnu-linux -
-class gnu --class os --unrestricted $menuentry_id_option 'gnulinux-0-rescue-a2feb9eb09b1488da0f23b99a66350f8-
advanced-b35de665-5ec8-4226-a533-58a1b567ac91' {
    load_video
    insmod gzio
```

```
insmod part_msdos
insmod xfs
set root='hd0,msdos1'
if [ x$feature_platform_search_hint = xy ]; then
    search --no-floppy --fs-uuid --set=root --hint-bios=hd0,msdos1 --hint-efi=hd0,msdos1 --hint-
baremetal=ahci0,msdos1 --hint='hd0,msdos1' e8d3bd48-1386-411c-9675-41c3f8f1a309
else
    search --no-floppy --fs-uuid --set=root e8d3bd48-1386-411c-9675-41c3f8f1a309
fi
linux16 /vmlinuz-0-rescue-a2feb9eb09b1488da0f23b99a66350f8 root=UUID=b35de665-5ec8-4226-a533-58alb567ac91 ro
vconsole.keymap=fr crashkernel=auto vconsole.font=latarcyrheb-sun16 rhgb quiet
initrd16 /initramfs-0-rescue-a2feb9eb09b1488da0f23b99a66350f8.img
}
if [ "x$default" = 'CentOS Linux, with Linux 3.10.0-123.el7.x86_64' ]; then default='Advanced options for CentOS
Linux>CentOS Linux, with Linux 3.10.0-123.el7.x86_64'; fi;
### END /etc/grub.d/10_linux ###

### BEGIN /etc/grub.d/20_linux_xen ###
### END /etc/grub.d/20_linux_xen ###

### BEGIN /etc/grub.d/20_ppc_terminfo ###
### END /etc/grub.d/20_ppc_terminfo ###

### BEGIN /etc/grub.d/30_os-prober ###
### END /etc/grub.d/30_os-prober ###

### BEGIN /etc/grub.d/40_custom ###
# This file provides an easy way to add custom menu entries.  Simply type the
# menu entries you want to add after this comment.  Be careful not to change
# the 'exec tail' line above.
### END /etc/grub.d/40_custom ###

### BEGIN /etc/grub.d/41_custom ###
if [ -f ${config_directory}/custom.cfg ]; then
```

```
source ${config_directory}/custom.cfg
elif [ -z "${config_directory}" -a -f $prefix/custom.cfg ]; then
    source $prefix/custom.cfg;
fi
### END /etc/grub.d/41_custom ###
```

Prenons le cas des paramètres de Grub Legacy et comparons-les aux paramètres de GRUB 2 :

Grub Legacy	GRUB 2
title	Menuentry
root (hd0,0)	set root=hd(0,1). Notez que GRUB 2 commence toujours la numérotation des disques à 0 mais numérote les partitions à partir de 1
kernel	linux
initrd	initrd
lock	Ce paramètre n'existe plus sous GRUB 2.
rootnoverify (hd0,1)	Ce paramètre n'existe plus sous GRUB 2. Les paramètres des systèmes d'exploitation non Linux sont définis avec le paramètre root

Notez que ce fichier ne doit pas être modifié manuellement. En effet, il est généré par la commande **grub2-mkconfig** sous RHEL/CentOS 7. La commande grub2-mkconfig prend en argument l'emplacement du fichier destination, par exemple :

- grub2-mkconfig -o /boot/grub2/grub.cfg, ou
- grub2-mkconfig -o /boot/efi/EFI/redhat/grub.cfg

Lors de l'exécution de la commande **grub2-mkconfig** plusieurs fichiers sont lus :

Le fichier /boot/grub2/device.map

```
[root@centos7 ~]# cat /boot/grub2/device.map
# this device map was generated by anaconda
(hd0)      /dev/sda
(hd1)      /dev/sda
```

Le fichier `/etc/default/grub`

Ce fichier contient la configuration par défaut des paramètres de GRUB 2 :

```
[root@centos7 ~]# cat /etc/default/grub
GRUB_TIMEOUT=5
GRUB_DISTRIBUTOR="$(sed 's, release .*$,,g' /etc/system-release)"
GRUB_DEFAULT=saved
GRUB_DISABLE_SUBMENU=true
GRUB_TERMINAL_OUTPUT="console"
GRUB_CMDLINE_LINUX="vconsole.keymap=fr crashkernel=auto vconsole.font=latarcyrheb-sun16 rhgb quiet"
GRUB_DISABLE_RECOVERY="true"
```



Important : Notez que toute modification de ce fichier nécessite l'exécution de la commande **grub2-mkconfig** sous RHEL/CentOS 7 pour que les modifications soient prises en compte.

Dans ce fichier les directives sont :

Directive	Description
GRUB_DEFAULT	Entrée du menu sélectionner par défaut
GRUB_TIMEOUT	Durée de l'affichage du menu avant le démarrage en utilisant la valeur de GRUB_DEFAULT
GRUB_DISTRIBUTOR	Ligne de commande qui génère le texte de l'entrée
GRUB_CMDLINE_LINUX_DEFAULT	Paramètres passés au noyau lors d'un démarrage normal (Hors donc le mode secours)
GRUB_CMDLINE_LINUX	Paramètres passés au noyau peu importe le type de démarrage
GRUB_TERMINAL	Si non commentée, cette directive désactive le démarrage graphique
GRUB_GFXMODE	Indique la résolution utilisée lors d'un démarrage graphique
GRUB_DISABLE_LINUX_UUID	Si true , cette directive empêche l'utilisation de l'UUID de la partition
GRUB_DISABLE_LINUX_RECOVERY	Si true , cette directive empêche la génération des entrées en mode recovery
GRUB_INIT_TUNE	Permet d'obtenir un beep au démarrage de GRUB 2

Directive	Description
GRUB_BADRAM	Permet de spécifier de la mémoire défaillante

Les fichiers du répertoire `/etc/grub.d`

Les fichiers de ce répertoire sont exécutés dans l'ordre alphanumérique et servent à construire les menus de GRUB 2 :

```
[root@centos7 ~]# ls -l /etc/grub.d
total 68
-rwxr-xr-x. 1 root root 8702 Mar 26 09:27 00_header
-rwxr-xr-x. 1 root root 992 Oct 17 2014 00_tuned
-rwxr-xr-x. 1 root root 10114 Mar 26 09:27 10_linux
-rwxr-xr-x. 1 root root 10275 Mar 26 09:27 20_linux_xen
-rwxr-xr-x. 1 root root 2559 Mar 26 09:27 20_ppc_terminfo
-rwxr-xr-x. 1 root root 11169 Mar 26 09:27 30_os-prober
-rwxr-xr-x. 1 root root 214 Mar 26 09:27 40_custom
-rwxr-xr-x. 1 root root 216 Mar 26 09:27 41_custom
-rw-r--r--. 1 root root 483 Mar 26 09:27 README
```

- **Le fichier `/etc/grub.d/10_Linux`,**
 - Le fichier **10_Linux** contient des boucles pour rechercher des noyaux Linux,
- **Le fichier `/etc/grub.d/30_os-prober`,**
 - Ce fichier recherche des éventuels systèmes d'exploitation autre que Linux,
- **Les fichiers `/etc/grub.d/40_custom` et `/etc/grub.d/41_custom`,**
 - Ces deux fichiers sont fournis en tant que modèles à personnaliser.

Configurer l'Authentification

Pour configurer l'authentification sous GRUB 2, il faut créer le fichier **`/etc/grub.d/01_users`** :

```
[root@centos7 ~]# touch /etc/grub.d/01_users
[root@centos7 ~]# chmod 755 /etc/grub.d/01_users
```

Créez deux mots de passe hashés au format **PBKDF2** en utilisant la commande **grub2-mkpasswd-pbkdf2** sous RHEL/CentOS :

```
[root@centos7 ~]# grub2-mkpasswd-pbkdf2
Enter password: pass123
Reenter password: pass123
PBKDF2 hash of your password is
grub.pbkdf2.sha512.10000.0298C1C613A451C46FBC95BB2AC7A41BCEC1C61512EF785BD81E3B65DFF9D57ED4ADF8906C3EF33C22C06FBD
D366E1C118FC41110BD646A4D49EF86EFD0573BF.E14A45900096D773BE99BEA9AB8D4FA81431458952798B997D4FC9E0850426F679897937
1B8EBD331DB33AE8FEAE25E6773156D42F21B884DBA405546782B3BD
[root@centos7 ~]# grub2-mkpasswd-pbkdf2
Enter password: pass456
Reenter password: pass456
PBKDF2 hash of your password is
grub.pbkdf2.sha512.10000.161D4183DC832357403296ED05961FCF494AED9E20DC21C84EA89085EB9EF5AAE4C7D4A276AA5CC21F9C224B
2ECA010B915B4830E9648A7398EB4A91E7E3D252.8277512B849FF727FDD0716D1D4CDC6B92E53918F665282E02133AAD1046EB10273A2BC7
0D76558FFC34A0C0C8BE5132E4C4C02C7C9C1A567BD5365D77350FCF
```

Editez le fichier **/etc/grub.d/01_users** ainsi :

[/etc/grub.d/01_users](#)

```
#!/bin/sh -e
cat <<EOF
set superusers="root"
password_pbkdf2 root
grub.pbkdf2.sha512.10000.0298C1C613A451C46FBC95BB2AC7A41BCEC1C61512EF785BD81E3B65DFF9D57ED4ADF8906C3EF33C22C
06FBDD366E1C118FC41110BD646A4D49EF86EFD0573BF.E14A45900096D773BE99BEA9AB8D4FA81431458952798B997D4FC9E0850426
F6798979371B8EBD331DB33AE8FEAE25E6773156D42F21B884DBA405546782B3BD
password_pbkdf2 trainee
grub.pbkdf2.sha512.10000.161D4183DC832357403296ED05961FCF494AED9E20DC21C84EA89085EB9EF5AAE4C7D4A276AA5CC21F9
C224B2ECA010B915B4830E9648A7398EB4A91E7E3D252.8277512B849FF727FDD0716D1D4CDC6B92E53918F665282E02133AAD1046EB
10273A2BC70D76558FFC34A0C0C8BE5132E4C4C02C7C9C1A567BD5365D77350FCF
EOF
```

Il est aussi possible d'utiliser des mots de passe non cryptés. Modifiez donc le fichier **/etc/grub.d/01_users** ainsi :

[/etc/grub.d/01_users](#)

```
#!/bin/sh -e
cat <<EOF
set superusers="root"
password root fenestros
password trainee trainee
EOF
```

Ouvrez maintenant le fichier **/boot/grub2/grub.cfg** et copier le premier **menuentry** de la section **/etc/grub.d/10_linux** :

```
menuentry 'CentOS Linux (3.10.0-229.4.2.el7.x86_64) 7 (Core)' --class centos --class gnu-linux --class gnu --
class os --unrestricted $menuentry_id_option 'gnulinux-3.10.0-123.el7.x86_64-advanced-b35de665-5ec8-4226-
a533-58a1b567ac91' {
    load_video
    set gfxpayload=keep
    insmod gzio
    insmod part_msdos
    insmod xfs
    set root='hd0,msdos1'
    if [ x$feature_platform_search_hint = xy ]; then
        search --no-floppy --fs-uuid --set=root --hint-bios=hd0,msdos1 --hint-efi=hd0,msdos1 --hint-
baremetal=ahci0,msdos1 --hint='hd0,msdos1' e8d3bd48-1386-411c-9675-41c3f8f1a309
    else
        search --no-floppy --fs-uuid --set=root e8d3bd48-1386-411c-9675-41c3f8f1a309
    fi
    linux16 /vmlinuz-3.10.0-229.4.2.el7.x86_64 root=UUID=b35de665-5ec8-4226-a533-58a1b567ac91 ro
vconsole.keymap=fr crashkernel=auto vconsole.font=latarcyrheb-sun16 rhgb quiet LANG=en_US.UTF-8
    initrd16 /initramfs-3.10.0-229.4.2.el7.x86_64.img
}
```

Collez maintenant ce **menuentry** dans le fichier **/etc/grub.d/40_custom** :

[/etc/grub.d/40_custom](#)

```
#!/bin/sh
exec tail -n +3 $0
# This file provides an easy way to add custom menu entries.  Simply type the
# menu entries you want to add after this comment.  Be careful not to change
# the 'exec tail' line above.
menuentry 'CentOS Linux (3.10.0-229.4.2.el7.x86_64) 7 (Core)' --class centos --class gnu-linux --class gnu -
--class os --unrestricted $menuentry_id_option 'gnulinux-3.10.0-123.el7.x86_64-advanced-b35de665-5ec8-4226-
a533-58a1b567ac91' {
    load_video
    set gfxpayload=keep
    insmod gzio
    insmod part_msdos
    insmod xfs
    set root='hd0,msdos1'
    if [ x$feature_platform_search_hint = xy ]; then
        search --no-floppy --fs-uuid --set=root --hint-bios=hd0,msdos1 --hint-efi=hd0,msdos1 --hint-
baremetal=ahci0,msdos1 --hint='hd0,msdos1' e8d3bd48-1386-411c-9675-41c3f8f1a309
    else
        search --no-floppy --fs-uuid --set=root e8d3bd48-1386-411c-9675-41c3f8f1a309
    fi
    linux16 /vmlinuz-3.10.0-229.4.2.el7.x86_64 root=UUID=b35de665-5ec8-4226-a533-58a1b567ac91 ro
vconsole.keymap=fr crashkernel=auto vconsole.font=latarcyrheb-sun16 rhgb quiet LANG=en_US.UTF-8
    initrd16 /initramfs-3.10.0-229.4.2.el7.x86_64.img
}
```

Modifier le debut du menuentry ainsi :

```
menuentry 'CentOS Linux (3.10.0-229.4.2.el7.x86_64) 7 (Core) pour TRAINEE' --class centos --class gnu-linux --
class gnu --class os --unrestricted $menuentry_id_option 'gnulinux-3.10.0-123.el7.x86_64-advanced-
```

```
b35de665-5ec8-4226-a533-58alb567ac91' --users trainee {
```

Sous RHEL/CentOS, lancez la commande **grub2-mkconfig -o /boot/grub2/grub.cfg** :

```
[root@centos7 ~]# grub2-mkconfig -o /boot/grub2/grub.cfg
Generating grub configuration file ...
Found linux image: /boot/vmlinuz-3.10.0-229.4.2.el7.x86_64
Found initrd image: /boot/initramfs-3.10.0-229.4.2.el7.x86_64.img
Found linux image: /boot/vmlinuz-3.10.0-123.el7.x86_64
Found initrd image: /boot/initramfs-3.10.0-123.el7.x86_64.img
Found linux image: /boot/vmlinuz-0-rescue-a2feb9eb09b1488da0f23b99a66350f8
Found initrd image: /boot/initramfs-0-rescue-a2feb9eb09b1488da0f23b99a66350f8.img
done
```

En examinant le fichier **/boot/grub2/grub.cfg** on doit pouvoir constater la présence de la section **/etc/grub.d/40_custom** :

```
...
### BEGIN /etc/grub.d/40_custom ###
# This file provides an easy way to add custom menu entries.  Simply type the
# menu entries you want to add after this comment.  Be careful not to change
# the 'exec tail' line above.
menuentry 'CentOS Linux (3.10.0-229.4.2.el7.x86_64) 7 (Core) pour TRAINEE' --class centos --class gnu-linux --
class gnu --class os --unrestricted $menuentry_id_option 'gnulinux-3.10.0-123.el7.x86_64-advanced-
b35de665-5ec8-4226-a533-58alb567ac91' --users trainee {
    load_video
    set gfxpayload=keep
    insmod gzio
    insmod part_msdos
    insmod xfs
    set root='hd0,msdos1'
    if [ x$feature_platform_search_hint = xy ]; then
        search --no-floppy --fs-uuid --set=root --hint-bios=hd0,msdos1 --hint-efi=hd0,msdos1 --hint-
baremetal=ahci0,msdos1 --hint='hd0,msdos1' e8d3bd48-1386-411c-9675-41c3f8f1a309
    else
```

```
    search --no-floppy --fs-uuid --set=root e8d3bd48-1386-411c-9675-41c3f8f1a309
fi
linux16 /vmlinuz-3.10.0-229.4.2.el7.x86_64 root=UUID=b35de665-5ec8-4226-a533-58a1b567ac91 ro
vconsole.keymap=fr crashkernel=auto vconsole.font=latarcyrheb-sun16 rhgb quiet LANG=en_US.UTF-8
initrd16 /initramfs-3.10.0-229.4.2.el7.x86_64.img
}
### END /etc/grub.d/40_custom ###
...
```



A faire : Redémarrez votre VM et choisissez l'entrée de GRUB 2 issue du fichier **/etc/grub.d/40_custom** puis constatez que GRUB 2 demande un nom d'utilisateur ainsi qu'un mot de passe quand vous voulez éditer une entrée de GRUB 2.

Modifier la Configuration de GRUB 2 en Ligne de Commande

Lors du démarrage de GRUB 2, trois actions sont possibles à partir du menu :

- Lancer un système d'exploitation en le sélectionnant avec les flèches puis en appuyant sur la touche ,
- Lancer l'éditeur en appuyant sur la touche **e**,
- Lancer l'interface de la ligne de commande GRUB en appuyant sur la touche **c**.

En mode édition notez l'utilisation des touches suivantes :

- **flèches** : se déplacer dans l'écran. L'édition se fait en utilisant simplement les touches du clavier,
- **Ctrl-X** : démarrer avec la configuration modifiée,
- **echap** : abandonner les modifications et retourner à l'interface menu de GRUB 2.



Important : Certaines distributions, telle qu'Ubuntu, cache le menu de GRUB 2 derrière une interface graphique. Afin de voir ce menu, il convient d'appuyer sur la touche **Shift** pendant que la machine démarre.

Chargeurs de Démarrages Alternatifs

Systemd-boot

Un Chargeur de Démarrage étroitement lié à Systemd (voir plus bas), celui-ci connaît actuellement un gain de popularité.

U-boot

Un Chargeur de Démarrage qui peut booter n'importe quelle image à partir de n'importe quel support.

Le Projet Syslinux

SYSLINUX

Un Chargeur de Démarrage pour les systèmes qui utilisent le système de fichier FAT. Par exemple le systèmes sur clefs USB.

EXTLINUX

Un Chargeur de Démarrage de petite taille qui sait booter des systèmes de fichier, EXT2, EXT3, EXT4 et BTRFS.

ISOLINUX

Un Chargeur de Démarrage pour booter des LiveCD et LiveDVD. Dans le cas d'ISOLINUX, deux fichiers sont nécessaires :

- **isolinux.bin** qui contient l'image du Chargeur de Démarrage et
 - **isolinux.cfg** qui contient les paramètres de configuration.
-

PXELINUX

Un Chargeur de Démarrage pour booter à partir d'un serveur réseau. Ce système utilise le standard **PXE** (*Pre-boot Execution Environment*) qui utilise :

- **DHCP** pour attribuer une adresse IP à la machine et
- **BOOTP** pour charger l'image du Chargeur de Démarrage à partir du serveur en utilisant le protocole **TFTP** (*Trivial File Transfer Protocol*). L'image à télécharger doit s'appeler **/tftpboot/pxelinux.0** et chaque machine doit avoir un fichier de configuration dans le répertoire **/tftpboot/pxelinux.cfg/**

Isodhpx

Un Chargeur de Démarrage hybride, appelé **isodhpx.bin**, qui peut être chargé sur un disque **ou** une clef USB. Le fichier isodhpx.bin est créé avec le programme **xorriso**.

Initramfs sous RHEL/CentOS 6

Un fichier Initramfs *INITial Ram File System* est une image d'un système minimal initialisée au démarrage du système.

Ce fichier utilise le système de fichier **cramFS** qui est un système de fichier compressé au format gzip et archivé via cpio.

L'image est chargée en mémoire vive et permet ainsi d'avoir un système minimal pouvant ensuite charger le système de fichier principal.

Examiner l'image existante

Pour examiner une image initramfs, il convient d'abord de la copier vers /tmp et de la décompresser :

```
[root@centos6 ~]# cp /boot/initramfs-2.6.32-71.29.1.el6.i686.img /tmp/custom.gz
[root@centos6 ~]# gunzip /tmp/custom.gz
```

Ensuite il convient d'extraire l'image grâce à la commande **cpio** :

```
[root@centos6 ~]# cd /tmp
[root@centos6 tmp]# mkdir initrd
[root@centos6 tmp]# cd initrd
[root@centos6 initrd]# cpio -cid -I ../custom
59631 blocs
```

Installez maintenant le paquet **tree** :

```
[root@centos6 initrd]# yum install tree
```

Utilisez maintenant la commande **tree** pour examiner le contenu de l'image :

```
[root@centos6 initrd]# tree | more
```

```
.
├── bin
│   ├── basename
│   ├── cat
│   ├── cp
│   ├── dash
│   ├── dd
│   ├── dmesg
│   ├── grep
│   ├── gzip
│   ├── ln
│   ├── loadkeys
│   ├── ls
│   ├── mkdir
│   ├── mknod
│   ├── mount
│   ├── mv
│   ├── plymouth
│   └── plymouthd
```

```
|  └─ readlink
|  └─ rm
|  └─ sed
|  └─ setfont
|  └─ sh -> dash
|  └─ sleep
|  └─ umount
|  └─ uname
└─ cmdline
--Plus--
```

Comme vous pouvez le constater, l'image contient une arborescence Linux minimaliste :

```
[root@centos6 initrd]# ls
bin      dev          emergency  init      initqueue-finished  lib    pre-pivot  pre-udev  sbin
sysroot  usr
cmdline  dracut-004-33.2.el6_0  etc      initqueue  initqueue-settled  mount  pre-trigger  proc      sys    tmp
var
```



A faire : Utilisez le manuel de la commande **cpio** pour comprendre les options utilisées.

Le script **init**

Le script **init** est lancé lors du chargement de l'image :

```
[root@centos6 initrd]# more init
#!/bin/sh
#
# Licensed under the GPLv2
#
```

```
# Copyright 2008-2009, Red Hat, Inc.
# Harald Hoyer <harald@redhat.com>
# Jeremy Katz <katzj@redhat.com>

wait_for_loginit()
{
    if getarg rdinitdebug; then
        set +x
        exec 0<>/dev/console 1<>/dev/console 2<>/dev/console
        # wait for loginit
        i=0
        while [ $i -lt 10 ]; do
            j=$(jobs)
            [ -z "$j" ] && break
            [ -z "${j##*Running*}" ] || break
            sleep 0.1
            i=$((i+1))
        done
        [ $i -eq 10 ] && kill %1 >/dev/null 2>&1

        while pidof -x /sbin/loginit >/dev/null 2>&1; do
            for pid in $(pidof -x /sbin/loginit); do
                kill $HARD $pid >/dev/null 2>&1
            done
        done
    --Plus-- (8%)
}
```



A faire : Passez en revue le contenu du script.

La commande dracut

La commande **dracut** permet de créer facilement une image initramfs. Les options de la commande sont :

```
[root@centos6 initrd]# dracut --help
Usage: /sbin/dracut [OPTION]... <initramfs> <kernel-version>
Creates initial ramdisk images for preloading modules

-f, --force                Overwrite existing initramfs file.
-m, --modules [LIST]       Specify a space-separated list of dracut modules to
                             call when building the initramfs. Modules are located
                             in /usr/share/dracut/modules.d.
-o, --omit [LIST]          Omit a space-separated list of dracut modules.
-a, --add [LIST]           Add a space-separated list of dracut modules.
-d, --drivers [LIST]       Specify a space-separated list of kernel modules to
                             exclusively include in the initramfs.
--add-drivers [LIST]       Specify a space-separated list of kernel
                             modules to add to the initramfs.
--filesystems [LIST]       Specify a space-separated list of kernel filesystem
                             modules to exclusively include in the generic
                             initramfs.
-k, --kmoddir [DIR]        Specify the directory, where to look for kernel
                             modules
--fwdir [DIR]              Specify additional directories, where to look for
                             firmwares, separated by :
--kernel-only              Only install kernel drivers and firmware files
--no-kernel                Do not install kernel drivers and firmware files
--strip                    Strip binaries in the initramfs
--nostrip                  Do not strip binaries in the initramfs (default)
--mdadmconf                Include local /etc/mdadm.conf
--nomdadmconf              Do not include local /etc/mdadm.conf
--lvmconf                  Include local /etc/lvm/lvm.conf
--nolvmconf                Do not include local /etc/lvm/lvm.conf
-h, --help                This message
--debug                    Output debug information of the build process
-v, --verbose              Verbose output during the build process
```

-c, --conf [FILE]	Specify configuration file to use. Default: /etc/dracut.conf
-l, --local	Local mode. Use modules from the current working directory instead of the system-wide installed in /usr/share/dracut/modules.d. Useful when running dracut from a git checkout.
-H, --hostonly	Host-Only mode: Install only what is needed for booting the local host instead of a generic host.
-i, --include [SOURCE] [TARGET]	Include the files in the SOURCE directory into the Target directory in the final initramfs.
-I, --install [LIST]	Install the space separated list of files into the initramfs.

Le fichier de configuration de dracut est **/etc/dracut.conf** :

```
[root@centos6 initrd]# cat /etc/dracut.conf
# Sample dracut config file

# Specific list of dracut modules to use
#dracutmodules+="

# Dracut modules to omit
#omit_dracutmodules+="

# Dracut modules to add to the default
#add_dracutmodules+="

# additional kernel modules to the default
add_drivers+="

# list of kernel filesystem modules to be included in the generic initramfs
#filesystems+="
```

```
# build initrd only to boot current hardware
#hostonly="yes"
#

# install local /etc/mdadm.conf
mdadmconf="yes"

# install local /etc/lvm/lvm.conf
lvmconf="yes"
```

Exécutez maintenant la commande suivante afin de générer le fichier **initramfs** :

```
# dracut -v initramfs [Entrée]
```

Notez la présence de votre nouvelle image **/tmp/initrd/initramfs**.

Déplacez votre fichier initramfs au répertoire /boot :

```
[root@centos6 initrd]# mv initramfs /boot
```

Editez maintenant votre fichier **/boot/grub/menu.lst** et **ajoutez** une **nouvelle section** qui utilise votre initramfs.

```
...
splashimage=(hd0,0)/grub/splash.xpm.gz
hiddenmenu
title CentOS Linux (initramfs)
    root (hd0,0)
    kernel /vmlinuz-2.6.32-71.29.1.el6.i686 ro root=UUID=e73735d6-c14b-4a40-8735-f34fc868da8a rd_NO_LUKS
rd_NO_LVM rd_NO_MD rd_NO_DM LANG=fr_FR.UTF-8 SYSFONT=latarcyrheb-sun16 KEYBOARDTYPE=pc KEYTABLE=fr-latin9
crashkernel=auto rhgb quiet
    initrd /initramfs
title CentOS Linux (2.6.32-71.29.1.el6.i686)
...
```



A faire : Re-démarrez votre machine pour tester votre configuration.

Initramfs sous RHEL/CentOS 7

Le fichier Initramfs *INITial Ram File System* est une archive au format cpio :

```
[root@centos7 ~]# cp /boot/initramfs-3.10.0-514.16.1.el7.x86_64.img /tmp/custom
[root@centos7 tmp]# mkdir initramfs
[root@centos7 tmp]# cd initramfs
[root@centos7 initramfs]# cpio -cid -I ../custom
42 blocks
[root@centos7 initramfs]# ls
early_cpio  kernel
[root@centos7 initramfs]# ls -lR .
.:
total 4
-rw-r--r--. 1 root root  2 Jun 22 11:51 early_cpio
drwxr-xr-x. 3 root root 60 Jun 22 11:51 kernel

./kernel:
total 0
drwxr-xr-x. 3 root root 60 Jun 22 11:51 x86

./kernel/x86:
total 0
drwxr-xr-x. 2 root root 60 Jun 22 11:51 microcode

./kernel/x86/microcode:
total 20
```

```
-rw-r--r--. 1 root root 20480 Jun 22 11:51 GenuineIntel.bin
```

Examiner l'image existante

Pour examiner une image initramfs archivée, il convient d'utiliser la commande **lsinitrd** :

```
[root@centos7 initramfs]# cd ..
[root@centos7 tmp]# ls
custom hspcrdata_root initramfs systemd-private-49209a1668d84a0d8425b08eb83403d9-cups.service-WTdfWP
[root@centos7 tmp]# lsinitrd custom | more
Image: custom: 27M
=====
Early CPIO image
=====
drwxr-xr-x  3 root    root          0 Apr 15 15:36 .
-rw-r--r--  1 root    root          2 Apr 15 15:36 early_cpio
drwxr-xr-x  3 root    root          0 Apr 15 15:36 kernel
drwxr-xr-x  3 root    root          0 Apr 15 15:36 kernel/x86
drwxr-xr-x  2 root    root          0 Apr 15 15:36 kernel/x86/microcode
-rw-r--r--  1 root    root      20480 Apr 15 15:36 kernel/x86/microcode/GenuineIntel.bin
=====
Version: dracut-033-463.el7

Arguments: --kver '3.10.0-514.16.1.el7.x86_64' -f

dracut modules:
bash
nss-softokn
i18n
network
ifcfg
drm
plymouth
```

```

kernel-modules
qemu
resume
rootfs-block
terminfo
udev-rules
biosdevname
systemd
usrmount
base
fs-lib
shutdown
=====
drwxr-xr-x 12 root root          0 Apr 15 15:36 .
crw-r--r--  1 root root          5,  1 Apr 15 15:35 dev/console
crw-r--r--  1 root root          1, 11 Apr 15 15:35 dev/kmsg
crw-r--r--  1 root root          1,  3 Apr 15 15:35 dev/null
lrwxrwxrwx  1 root root          7 Apr 15 15:35 bin -> usr/bin
drwxr-xr-x  2 root root          0 Apr 15 15:35 dev
drwxr-xr-x 11 root root          0 Apr 15 15:36 etc
drwxr-xr-x  2 root root          0 Apr 15 15:35 etc/cmdline.d
--More--

```

Comme vous pouvez le constater, l'image contient une arborescence Linux minimaliste :

Le script **init**

Le script **init** est lancé lors du chargement de l'image. RHEL/CentOS 7 utilise le système de démarrage **systemd**. Pour cette raison le script **init** sous RHEL/CentOS 7 est un lien symbolique vers **/usr/lib/systemd/systemd** :

```

[root@centos7 tmp]# lsinitrd custom | grep init
-rw-r--r--  1 root root          170 Apr 15 15:36 etc/initrd-release
...

```

```
lrwxrwxrwx  1 root    root          22 Apr 15 15:36 usr/sbin/init -> ../lib/systemd/systemd
...
```

Consultez le contenu d'un fichier dans initramfs

Pour consulter le contenu d'un fichier inclus dans l'initramfs, il convient d'utiliser la commande **lsinitrd** de la manière suivante :

```
[root@centos7 tmp]# lsinitrd /boot/initramfs-$(uname -r).img /etc/ld.so.conf
include ld.so.conf.d/*.conf
```

La commande dracut sous RHEL/CentOS

La commande **dracut** permet de créer facilement une image initramfs. Les options de la commande sont :

```
[root@centos7 tmp]# dracut --help
Usage: /sbin/dracut [OPTION]... [<initramfs> [<kernel-version>]]
```

Version: 033-463.el7

Creates initial ramdisk images for preloading modules

--kver [VERSION]	Set kernel version to [VERSION].
-f, --force	Overwrite existing initramfs file.
-a, --add [LIST]	Add a space-separated list of dracut modules.
--rebuild	Append arguments to those of existing image and rebuild
-m, --modules [LIST]	Specify a space-separated list of dracut modules to call when building the initramfs. Modules are located in /usr/lib/dracut/modules.d.
-o, --omit [LIST]	Omit a space-separated list of dracut modules.
--force-add [LIST]	Force to add a space-separated list of dracut modules to the default set of modules, when -H is specified.

-d, --drivers [LIST]	Specify a space-separated list of kernel modules to exclusively include in the initramfs.
--add-drivers [LIST]	Specify a space-separated list of kernel modules to add to the initramfs.
--omit-drivers [LIST]	Specify a space-separated list of kernel modules not to add to the initramfs.
--filesystems [LIST]	Specify a space-separated list of kernel filesystem modules to exclusively include in the generic initramfs.
-k, --kmoddir [DIR]	Specify the directory, where to look for kernel modules
--fwdir [DIR]	Specify additional directories, where to look for firmwares, separated by :
--kernel-only	Only install kernel drivers and firmware files
--no-kernel	Do not install kernel drivers and firmware files
--print-cmdline	Print the kernel command line for the given disk layout
--early-microcode	Combine early microcode with ramdisk
--no-early-microcode	Do not combine early microcode with ramdisk
--kernel-cmdline [PARAMETERS]	Specify default kernel command line parameters
--strip	Strip binaries in the initramfs
--nostrip	Do not strip binaries in the initramfs
--prelink	Prelink binaries in the initramfs
--noprelink	Do not prelink binaries in the initramfs
--hardlink	Hardlink files in the initramfs
--nohardlink	Do not hardlink files in the initramfs
--prefix [DIR]	Prefix initramfs files with [DIR]
--noprefix	Do not prefix initramfs files
--mdadmconf	Include local /etc/mdadm.conf
--nomdadmconf	Do not include local /etc/mdadm.conf
--lvmconf	Include local /etc/lvm/lvm.conf
--nolvmconf	Do not include local /etc/lvm/lvm.conf
--fscks [LIST]	Add a space-separated list of fsck helpers.
--nofscks	Inhibit installation of any fsck helpers.
--ro-mnt	Mount / and /usr read-only by default.

-h, --help	This message
--debug	Output debug information of the build process
--profile	Output profile information of the build process
-L, --stdlog [0-6]	Specify logging level (to standard error) 0 - suppress any messages 1 - only fatal errors 2 - all errors 3 - warnings 4 - info 5 - debug info (here starts lots of output) 6 - trace info (and even more)
-v, --verbose	Increase verbosity level
-q, --quiet	Decrease verbosity level
-c, --conf [FILE]	Specify configuration file to use. Default: /etc/dracut.conf
--confdir [DIR]	Specify configuration directory to use *.conf files from. Default: /etc/dracut.conf.d
--tmpdir [DIR]	Temporary directory to be used instead of default /var/tmp.
-l, --local	Local mode. Use modules from the current working directory instead of the system-wide installed in /usr/lib/dracut/modules.d. Useful when running dracut from a git checkout.
-H, --hostonly	Host-Only mode: Install only what is needed for booting the local host instead of a generic host.
-N, --no-hostonly	Disables Host-Only mode
--hostonly-cmdline	Store kernel command line arguments needed in the initramfs
--no-hostonly-cmdline	Do not store kernel command line arguments needed in the initramfs
--hostonly-i18n	Install only needed keyboard and font files according to the host configuration (default).
--no-hostonly-i18n	Install all keyboard and font files available.
--persistent-policy [POLICY]	

```
Use [POLICY] to address disks and partitions.
POLICY can be any directory name found in /dev/disk.
E.g. "by-uuid", "by-label"
--fstab           Use /etc/fstab to determine the root device.
--add-fstab [FILE] Add file to the initramfs fstab
--mount "[DEV] [MP] [FSTYPE] [FSOPTS]"
                  Mount device [DEV] on mountpoint [MP] with filesystem
                  [FSTYPE] and options [FSOPTS] in the initramfs
--add-device "[DEV]" Bring up [DEV] in initramfs
-i, --include [SOURCE] [TARGET]
                  Include the files in the SOURCE directory into the
                  Target directory in the final initramfs.
                  If SOURCE is a file, it will be installed to TARGET
                  in the final initramfs.
-I, --install [LIST] Install the space separated list of files into the
                  initramfs.
--install-optional [LIST] Install the space separated list of files into the
                  initramfs, if they exist.
--gzip           Compress the generated initramfs using gzip.
                  This will be done by default, unless another
                  compression option or --no-compress is passed.
--bzip2          Compress the generated initramfs using bzip2.
                  Make sure your kernel has bzip2 decompression support
                  compiled in, otherwise you will not be able to boot.
--lzma           Compress the generated initramfs using lzma.
                  Make sure your kernel has lzma support compiled in,
                  otherwise you will not be able to boot.
--xz            Compress the generated initramfs using xz.
                  Make sure that your kernel has xz support compiled
                  in, otherwise you will not be able to boot.
--lzo            Compress the generated initramfs using lzop.
                  Make sure that your kernel has lzo support compiled
                  in, otherwise you will not be able to boot.
--lz4           Compress the generated initramfs using lz4.
```

```
Make sure that your kernel has lz4 support compiled
in, otherwise you will not be able to boot.
--compress [COMPRESSION] Compress the generated initramfs with the
                           passed compression program. Make sure your kernel
                           knows how to decompress the generated initramfs,
                           otherwise you will not be able to boot.
--no-compress             Do not compress the generated initramfs. This will
                           override any other compression options.
--list-modules            List all available dracut modules.
-M, --show-modules        Print included module's name to standard output during
                           build.
--keep                   Keep the temporary initramfs for debugging purposes
--printsiz               Print out the module install size
--sshkey [SSHKEY]        Add ssh key to initramfs (use with ssh-client module)
--logfile [FILE]         Logfile to use (overrides configuration setting)
```

If [LIST] has multiple arguments, then you have to put these in quotes.

For example:

```
# dracut --add-drivers "module1 module2" ...
```

Le fichier de configuration de dracut est **/etc/dracut.conf** :

```
# PUT YOUR CONFIG HERE OR IN separate files named *.conf
# in /etc/dracut.conf.d
# SEE man dracut.conf(5)

# Sample dracut config file

#logfile=/var/log/dracut.log
#fileloglvl=6

# Exact list of dracut modules to use. Modules not listed here are not going
```

```
# to be included. If you only want to add some optional modules use
# add_dracutmodules option instead.
#dracutmodules+="

# dracut modules to omit
#omit_dracutmodules+="

# dracut modules to add to the default
#add_dracutmodules+="

# additional kernel modules to the default
add_drivers+="ehci-hcd ohci-hcd usb-storage scsi_mod sd_mod"

# list of kernel filesystem modules to be included in the generic initramfs
#filesystems+="

# build initrd only to boot current hardware
#hostonly="yes"
#

# install local /etc/mdadm.conf
#mdadmconf="no"

# install local /etc/lvm/lvm.conf
#lvmconf="no"

# A list of fsck tools to install. If it's not specified, module's hardcoded
# default is used, currently: "umount mount /sbin/fsck* xfs_db xfs_check
# xfs_repair e2fsck jfs_fsck reiserfsck btrfsck". The installation is
# opportunistic, so non-existing tools are just ignored.
#fscks=""

# inhibit installation of any fsck tools
#nofscks="yes"
```

```
# mount / and /usr read-only by default
#ro_mnt="no"

# set the directory for temporary files
# default: /var/tmp
#tmpdir=/tmp
```

Exécutez maintenant la commande suivante afin de générer le fichier **initramfs** :

```
[root@centos7 tmp]# dracut -v initramfs
Executing: /sbin/dracut -v initramfs
dracut module 'busybox' will not be installed, because command 'busybox' could not be found!
dracut module 'busybox' will not be installed, because command 'busybox' could not be found!
*** Including module: bash ***
*** Including module: nss-softokn ***
*** Including module: il8n ***
*** Including module: network ***
*** Including module: ifcfg ***
*** Including module: drm ***
*** Including module: plymouth ***
*** Including module: kernel-modules ***
*** Including module: qemu ***
*** Including module: resume ***
*** Including module: rootfs-block ***
*** Including module: terminfo ***
*** Including module: udev-rules ***
Skipping udev rule: 91-permissions.rules
*** Including module: biosdevname ***
*** Including module: systemd ***
*** Including module: usrmount ***
*** Including module: base ***
*** Including module: fs-lib ***
*** Including module: shutdown ***
*** Including modules done ***
```

```
*** Installing kernel module dependencies and firmware ***
*** Installing kernel module dependencies and firmware done ***
*** Resolving executable dependencies ***
*** Resolving executable dependencies done***
*** Hardlinking files ***
*** Hardlinking files done ***
*** Stripping files ***
*** Stripping files done ***
*** Generating early-microcode cpio image contents ***
*** Constructing GenuineIntel.bin ****
*** Store current command line parameters ***
*** Creating image file ***
*** Creating microcode section ***
*** Created microcode section ***
*** Creating image file done ***
*** Creating initramfs image file '/tmp/initramfs' done ***
```

Notez la présence de votre nouvelle image **/tmp/initrd/initramfs** :

```
[root@centos7 initrd]# ls -l /tmp/initrd/initramfs
-rw-----. 1 root root 16373488 Jun 10 10:23 /tmp/initrd/initramfs
```

Déplacez votre fichier initramfs au répertoire /boot :

```
[root@centos7 initrd]# mv initramfs /boot
```

Ouvrez votre fichier **/boot/grub2/grub.cfg** et cherchez la section **10_linux**. Copiez la première section **menuentry** :

```
...
### BEGIN /etc/grub.d/10_linux ###
menuentry 'CentOS Linux (3.10.0-229.4.2.el7.x86_64) 7 (Core)' --class centos --class gnu-linux --class gnu --
class os --unrestricted $menuentry_id_option 'gnulinux-3.10.0-123.el7.x86_64-advanced-b35de665-5ec8-4226-
a533-58a1b567ac91' {
    load_video
```

```
set gfxpayload=keep
insmod gzio
insmod part_msdos
insmod xfs
set root='hd0,msdos1'
if [ x$feature_platform_search_hint = xy ]; then
    search --no-floppy --fs-uuid --set=root --hint-bios=hd0,msdos1 --hint-efi=hd0,msdos1 --hint-
baremetal=ahci0,msdos1 --hint='hd0,msdos1' e8d3bd48-1386-411c-9675-41c3f8f1a309
else
    search --no-floppy --fs-uuid --set=root e8d3bd48-1386-411c-9675-41c3f8f1a309
fi
linux16 /vmlinuz-3.10.0-229.4.2.el7.x86_64 root=UUID=b35de665-5ec8-4226-a533-58a1b567ac91 ro
vconsole.keymap=fr crashkernel=auto vconsole.font=latarcyrheb-sun16 rhgb quiet LANG=en_US.UTF-8
initrd16 /initramfs-3.10.0-229.4.2.el7.x86_64.img
}
...
```

Créez maintenant le fichier **/etc/grub.d/09_redhat**, collez la section précédemment copiée à la place indiquée entre les lignes **cat « EOF** et **EOF** puis modifiez la dernière ligne **initrd16** :

```
#!/bin/sh -e
cat << EOF
menuentry 'CentOS Linux, with Linux 3.10.0-229.4.2.el7.x86_64 et initramfs' --class centos --class gnu-linux --
class gnu --class os --unrestricted $menuentry_id_option 'gnulinux-3.10.0-229.4.2.el7.x86_64-advanced-
b35de665-5ec8-4226-a533-58a1b567ac91' {
    load_video
    set gfxpayload=keep
    insmod gzio
    insmod part_msdos
    insmod xfs
    set root='hd0,msdos1'
    if [ x$feature_platform_search_hint = xy ]; then
        search --no-floppy --fs-uuid --set=root --hint-bios=hd0,msdos1 --hint-efi=hd0,msdos1 --hint-
baremetal=ahci0,msdos1 --hint='hd0,msdos1' e8d3bd48-1386-411c-9675-41c3f8f1a309
```

```
    else
        search --no-floppy --fs-uuid --set=root e8d3bd48-1386-411c-9675-41c3f8f1a309
    fi
    linux16 /vmlinuz-3.10.0-229.4.2.el7.x86_64 root=UUID=b35de665-5ec8-4226-a533-58a1b567ac91 ro
vconsole.keymap=fr crashkernel=auto vconsole.font=latarcyrheb-sun16 rhgb quiet
    initrd16 /initramfs
}
EOF
```

Rendez ce fichier exécutable :

```
[root@centos7 initrd]# chmod +x /etc/grub.d/09_redhat
```

Avant de continuer, supprimer le fichier **/etc/grub.d/01_users** et éditez le contenu du fichier **/etc/grub.d/40_custom** ainsi :

```
#!/bin/sh
exec tail -n +3 $0
# This file provides an easy way to add custom menu entries.  Simply type the
# menu entries you want to add after this comment.  Be careful not to change
# the 'exec tail' line above.
```

Mettez à jour grub afin que celui-ci prend en compte le nouveau fichier :

```
[root@centos7 initrd]# grub2-mkconfig -o /boot/grub2/grub.cfg
Generating grub configuration file ...
Found linux image: /boot/vmlinuz-3.10.0-229.4.2.el7.x86_64
Found initrd image: /boot/initramfs-3.10.0-229.4.2.el7.x86_64.img
Found linux image: /boot/vmlinuz-3.10.0-123.el7.x86_64
Found initrd image: /boot/initramfs-3.10.0-123.el7.x86_64.img
Found linux image: /boot/vmlinuz-0-rescue-a2feb9eb09b1488da0f23b99a66350f8
Found initrd image: /boot/initramfs-0-rescue-a2feb9eb09b1488da0f23b99a66350f8.img
done
```

Contrôlez le fichier **/boot/grub2/grub.cfg** :

```
...  
### BEGIN /etc/grub.d/09_redhat ###  
menuentry 'CentOS Linux, with Linux 3.10.0-229.4.2.el7.x86_64 et initramfs' --class centos --class gnu-linux --  
class gnu --class os --unrestricted 'gnulinux-3.10.0-229.4.2.el7.x86_64-advanced-b35de665-5ec8-4226-  
a533-58a1b567ac91' {  
    load_video  
    set gfxpayload=keep  
    insmod gzio  
    insmod part_msdos  
    insmod xfs  
    set root='hd0,msdos1'  
    if [ x = xy ]; then  
        search --no-floppy --fs-uuid --set=root --hint-bios=hd0,msdos1 --hint-efi=hd0,msdos1 --hint-  
baremetal=ahci0,msdos1 --hint='hd0,msdos1' e8d3bd48-1386-411c-9675-41c3f8f1a309  
    else  
        ...  
    fi  
}
```



A faire : Re-démarrez votre machine pour tester votre configuration.

Processus de Démarrage du Noyau Linux

Le processus de démarrage du Noyau Linux peut être divisé en 6 étapes :

Etape	Description
Chargement, configuration et exécution du chargeur du noyau	Le fichier bootsect.s est chargé en mémoire par le BIOS. Une fois configuré celui-ci charge le reste du noyau en mémoire
Configuration des paramètres et bascule vers le mode 32 bits	Le fichier boot.s met en place un IDT (<i>Interrupt Descriptor Table</i>) temporaire et GDT (<i>Global Descriptor Table</i>) temporaire et gère le basculement vers le mode 32 bits
Décompression du Noyau	Le fichier head.s décompresse le noyau

Etape	Description
Initialisation du noyau et de la mémoire	Le fichier head.s crée un GDT et IDT définitif
Configuration du noyau	Le fichier main.c met en place les contraintes de mémoire et configure la mémoire virtuelle
Création du processus Init	Le fichier main.c crée le processus init

La fonction **init_post()** essaie ensuite d'exécuter un des processus suivant dans l'ordre :

- /sbin/init
- /etc/init
- /bin/init
- /bin/sh

Dans le cas d'un échec à ce stade le message **Kernel Panic** sera affiché.

Processus Init

Le premier processus lancé par le noyau est **Init**. L'exécutable lancé est **/sbin/init**. Son rôle est de d'initialiser le système et de lancer certains autres services. Les tâches accomplies par init sont :

- le montage de /proc et de /sys,
- configuration des paramètres du noyau présents dans **/etc/sysctl.conf**,
- l'activation de SELinux,
- la mise à l'heure du système,
- la définition des consoles textes,
- la définition du nom de la machine,
- la détection des périphériques USB,
- la mise en place du support RAID et LVM,
- l'activation des quotas de disque,
- le montages des systèmes de fichiers,
- le re-montage du système de fichiers racine en lecture/écriture,
- l'activation du swap,
- le lancement de syslog,
- le chargement des modules du noyau,

- le nettoyage des fichiers temporaires,
- la définition des variables d'environnement tels PATH et RUNLEVEL

Démarrer RHEL/CentOS 5 avec SysVinit

Niveaux d'exécution sous RHEL/CentOS 5

Il existe 8 niveaux d'exécution ou **RUNLEVELS** sous RHEL/CentOS 5. Quatre des 8 sont réservés :

RUNLEVEL	Description
0	Arrêt de la machine
1	Mode mono-utilisateur pour la maintenance
6	Redémarrage de la machine
S ou s	Mode mono-utilisateur avec seul la partition racine montée

Les autres quatre RUNLEVELS sont définis par chaque distribution. Par exemple, sous RedHat, ils sont :

RUNLEVEL	Description
2	Mode multi-utilisateur sans NFS
3	Mode multi-utilisateur
4	Non-utilisé
5	Mode multi-utilisateur avec session graphique

Il existe aussi 3 pseudo-niveaux d'exécution **a**, **b** et **c**. Ces pseudo-niveaux permettent à init de faire quelquechose sans changer de niveau d'exécution.

Pour connaître le niveau d'exécution actuel de la machine, saisissez la commande suivante :

```
[root@centos5 ~]# runlevel  
N 5
```

La lettre N indique que le système n'a pas changé de niveau d'exécution depuis son démarrage.

Pour modifier le niveau d'exécution courant, il convient d'utiliser la commande **init** ou **telinit** suivie du numéro du nouveau niveau d'exécution. Ces commandes peuvent prendre plusieurs options :

Option	Description
Q ou q	Demande à Init de relire le fichier /etc/inittab
-t	Permet de modifier le temps accordé par Init aux processus entre l'envoi du signal SIGTERM et l'envoi du signal SIGKILL

Inittab

Le fichier **/etc/inittab** permet de définir les services à démarrer en fonction du RUNLEVEL :

```
[root@centos5 ~]# cat /etc/inittab
#
# inittab          This file describes how the INIT process should set up
#                  the system in a certain run-level.
#
# Author:          Miquel van Smoorenburg, <miquels@drinkel.nl.mugnet.org>
#                  Modified for RHS Linux by Marc Ewing and Donnie Barnes
#
# Default runlevel. The runlevels used by RHS are:
#  0 - halt (Do NOT set initdefault to this)
#  1 - Single user mode
#  2 - Multiuser, without NFS (The same as 3, if you do not have networking)
#  3 - Full multiuser mode
#  4 - unused
#  5 - X11
#  6 - reboot (Do NOT set initdefault to this)
#
id:5:initdefault:
```

```
# System initialization.
si::sysinit:/etc/rc.d/rc.sysinit

l0:0:wait:/etc/rc.d/rc 0
l1:1:wait:/etc/rc.d/rc 1
l2:2:wait:/etc/rc.d/rc 2
l3:3:wait:/etc/rc.d/rc 3
l4:4:wait:/etc/rc.d/rc 4
l5:5:wait:/etc/rc.d/rc 5
l6:6:wait:/etc/rc.d/rc 6

# Trap CTRL-ALT-DELETE
ca::ctrlaltdel:/sbin/shutdown -t3 -r now

# When our UPS tells us power has failed, assume we have a few minutes
# of power left.  Schedule a shutdown for 2 minutes from now.
# This does, of course, assume you have powerd installed and your
# UPS connected and working correctly.
pf::powerfail:/sbin/shutdown -f -h +2 "Power Failure; System Shutting Down"

# If power was restored before the shutdown kicked in, cancel it.
pr:12345:powerokwait:/sbin/shutdown -c "Power Restored; Shutdown Cancelled"

# Run gettys in standard runlevels
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

# Run xdm in runlevel 5
x:5:respawn:/etc/X11/prefdm -nodaemon
```

Dans l'exemple ci-dessus, chaque ligne non-commentée est composée de quatre champs, séparés par le caractère :

Champ	Nom	Description
1	Identifiant	Identifiant unique de la ligne composé de 1 à 4 caractères
2	RUNLEVELS	Liste des niveaux d'exécution concernés par la ligne
3	Action	Méthode utilisé pour lancer la commande se trouvant dans le champ 4
4	Commande	Commande à lancer

Le champ **action** prend une des directives suivantes :

Directive	Description
respawn	Le processus est relancé en cas d'arrêt de celui-ci
mingetty	Assure la gestion du terminal texte
once	Le processus n'est exécuté qu'une fois
wait	Le processus n'est exécuté qu'une fois. Init attend la fin du processus avant de passer à la ligne suivante
boot	Le processus est exécuté au démarrage de la machine. Le champ RUNLEVELS est sans importance
bootwait	Le processus est exécuté au démarrage de la machine. Le champ RUNLEVELS est sans importance. Init attend la fin du processus avant de passer à la ligne suivante
off	Revient à commenter la ligne
ondemand	La même chose que respawn mais la commande est exécuté dans un des 3 pseudo-niveaux d'exécution
initdefault	Définit le niveau d'exécution par défaut
sysinit	La commande est exécutée au démarrage de la machine avant les lignes boot et bootwait
powerfail	La commande est exécutée quand init reçoit un signal SIGPWR d'un onduleur
powerwait	La commande est exécutée quand init reçoit un signal SIGPWR d'un onduleur. Init attend la fin du processus avant de passer à la ligne suivante
powerokwait	La commande est exécutée si Init reçoit un signal de rétablissement du courant
powerfailnow	La commande est exécutée quand Init reçoit un signal comme quoi la batterie de l'onduleur est presque vide
ctrlaltdel	La commande est exécutée quand Init reçoit un signal SIGINT. Ce signal est envoyé par la combinaison de touches [CTRL] [ALT] [SUPPR]
kbrequest	La commande est exécutée suivant des séquences de touches saisies au clavier

L'analyse de notre fichier d'exemple sous RHEL/CentOS 5 indique :

Ligne	Description
id:5:initdefault:	Le niveau d'exécution par défaut est 5
si::sysinit:/etc/rc.d/rc.sysinit	La commande /etc/rc.d/rc.sysinit est lancée au démarrage de la machine
l0:0:wait:/etc/rc.d/rc 0	La commande /etc/rc.d/rc est lancée pour le niveau d'exécution 0
l1:1:wait:/etc/rc.d/rc 1	La commande /etc/rc.d/rc est lancée pour le niveau d'exécution 1
l2:2:wait:/etc/rc.d/rc 2	La commande /etc/rc.d/rc est lancée pour le niveau d'exécution 2
l3:3:wait:/etc/rc.d/rc 3	La commande /etc/rc.d/rc est lancée pour le niveau d'exécution 3
l4:4:wait:/etc/rc.d/rc 4	La commande /etc/rc.d/rc est lancée pour le niveau d'exécution 4
l5:5:wait:/etc/rc.d/rc 5	La commande /etc/rc.d/rc est lancée pour le niveau d'exécution 5
l6:6:wait:/etc/rc.d/rc 6	La commande /etc/rc.d/rc est lancée pour le niveau d'exécution 6
ca::ctrlaltdel:/sbin/shutdown -t3 -r now	La commande /sbin/shutdown -t3 -r now est lancée si les touches [CTRL] [ALT] [SUPPR] sont appuyées simultanément
pf::powerfail:/sbin/shutdown -f -h +2 "Power Failure; System Shutting Down"	La commande /sbin/shutdown -f -h +2 est lancée quand Init reçoit le signal SIGPWR
pr:12345:powerokwait:/sbin/shutdown -c "Power Restored; Shutdown Cancelled"	La commande /sbin/shutdown -c est lancée quand Init reçoit un signal de rétablissement du courant
1:2345:respawn:/sbin/mingetty tty1	Le terminal tty1 est initialisé par la commande /sbin/mingetty. Le terminal est disponible en appuyant sur les touches [CTRL] [ALT] [F1]
2:2345:respawn:/sbin/mingetty tty2	Le terminal tty2 est initialisé par la commande /sbin/mingetty. Le terminal est disponible en appuyant sur les touches [CTRL] [ALT] [F2]
3:2345:respawn:/sbin/mingetty tty3	Le terminal tty3 est initialisé par la commande /sbin/mingetty. Le terminal est disponible en appuyant sur les touches [CTRL] [ALT] [F3]
4:2345:respawn:/sbin/mingetty tty4	Le terminal tty4 est initialisé par la commande /sbin/mingetty. Le terminal est disponible en appuyant sur les touches [CTRL] [ALT] [F4]
5:2345:respawn:/sbin/mingetty tty5	Le terminal tty5 est initialisé par la commande /sbin/mingetty. Le terminal est disponible en appuyant sur les touches [CTRL] [ALT] [F5]
6:2345:respawn:/sbin/mingetty tty6	Le terminal tty6 est initialisé par la commande /sbin/mingetty. Le terminal est disponible en appuyant sur les touches [CTRL] [ALT] [F6]
x:5:respawn:/etc/X11/prefdm -nodaemon	xdm est lancé dans le niveau d'exécution 5

Scripts de Démarrage

Naviguez à **/etc/rc.d** et saisissez la commande **ls** :

```
[root@centos5 ~]# cd /etc/rc.d
[root@centos5 rc.d]# ls
init.d  rc0.d  rc2.d  rc4.d  rc6.d  rc.sysinit
rc      rc1.d  rc3.d  rc5.d  rc.local
```

rc.sysinit sous RHEL/CentOS 5

D'après l'étude du fichier **inittab** nous savons que le script rc.sysinit est exécuté en premier.

Répertoire init.d

Le répertoire **/etc/rc.d/init.d** contient les scripts permettant de lancer les services du système :

```
[root@centos5 rc.d]# ls init.d/*
init.d/abrttd          init.d/cpuspeed      init.d/iptables      init.d/netfs          init.d/portreserve
init.d/sandbox         init.d/vboxadd
init.d/acpid           init.d/crond          init.d/irqbalance    init.d/network        init.d/postfix
init.d/saslauthd       init.d/vboxadd-service
init.d/atd             init.d/cups           init.d/jexec          init.d/NetworkManager init.d/psacct
init.d/single          init.d/vboxadd-x11
init.d/auditd          init.d/dnsmasq        init.d/kdump          init.d/nfs             init.d/rdisc
init.d/smartd          init.d/wpa_supplicant
init.d/autofs          init.d/firstboot      init.d/killall         init.d/nfslock        init.d/restorecond
init.d/snmpd           init.d/ypbind
init.d/avahi-daemon    init.d/functions      init.d/lvm2-monitor   init.d/nscd            init.d/rpcbind
init.d/snmptrapd
```

init.d/bluetooth	init.d/haldaemon	init.d/mdmonitor	init.d/nslcd	init.d/rpcgssd
init.d/ssh				
init.d/certmonger	init.d/halt	init.d/messagebus	init.d/ntpd	init.d/rpcidmapd
init.d/sss				
init.d/cgconfig	init.d/httpd	init.d/microcode_ctl	init.d/ntpdate	init.d/rpcsvcgssd
init.d/sysstat				
init.d/cgred	init.d/ip6tables	init.d/netconsole	init.d/oddjobd	init.d/rsyslog
init.d/udev-post				

Linux Standard Base

Linux Standard Base (LSB) fut introduit par le **Linux Foundation** dans un but de permettre la portabilité des scripts init entre distributions différentes.

Les scripts init qui sont conformes au standard LSB doivent fournir :

- au moins les actions **start**, **stop**, **restart**, **force-reload** et **status**,
- des codes retours standardisés,
- des informations sur des dépendances.

Les scripts init conformes au standard LSB peuvent aussi fournir :

- les actions **reload** et **try-restart**,
- des messages de journalisation en utilisant les fonctions Init.d **log_success_msg**, **log_failure_msg** et **log_warning_msg**.

Les fonctions proposées par défaut par LSB sont contenues dans le fichier **/lib/lsb/init-functions** :

```
[root@centos5 ~]# cat /lib/lsb/init-functions
#!/bin/sh

# LSB initscript functions, as defined in the LSB Spec 1.1.0
#
# Lawrence Lim <llim@redhat.com> - Tue, 26 June 2007
```

```
# Updated to the latest LSB 3.1 spec
# http://refspecs.freestandards.org/LSB_3.1.0/LSB-Core-generic/LSB-Core-generic_lines.txt

start_daemon () {
    /etc/redhat-lsb/lsb_start_daemon "$@"
}

killproc () {
    /etc/redhat-lsb/lsb_killproc "$@"
}

pidofproc () {
    /etc/redhat-lsb/lsb_pidofproc "$@"
}

log_success_msg () {
    /etc/redhat-lsb/lsb_log_message success "$@"
}

log_failure_msg () {
    /etc/redhat-lsb/lsb_log_message failure "$@"
}

log_warning_msg () {
    /etc/redhat-lsb/lsb_log_message warning "$@"
}
```

Ces fonctions sont des scripts se trouvant dans le répertoire **/etc/redhat-lsb** :

```
[root@centos5 ~]# ls -l /etc/redhat-lsb/
total 16
-rwxr-xr-x. 1 root root 70 22 févr. 2013 lsb_killproc
-rwxr-xr-x. 1 root root 243 22 févr. 2013 lsb_log_message
-rwxr-xr-x. 1 root root 59 22 févr. 2013 lsb_pidofproc
```

```
-rwxr-xr-x. 1 root root 650 22 févr. 2013 lsb_start_daemon
```

L'examen de ces scripts nous démontre que le système appelle le fichier **/etc/init.d/functions** :

```
[root@centos5 ~]# cat /etc/redhat-lsb/lsb_killproc
#!/bin/bash

. /etc/init.d/functions

LSB=LSB-1.1 killproc $*
exit $?

[root@centos ~]#
[root@centos ~]# cat /etc/redhat-lsb/lsb_pidofproc
#!/bin/bash

. /etc/init.d/functions

pidofproc $*
exit $?
```

Le fichier **/etc/init.d/functions** contient les fonctions à exécuter, par exemple :

```
...
# A function to stop a program.
killproc() {
    local RC killlevel= base pid pid_file= delay try binary=

    RC=0; delay=3; try=0
    # Test syntax.
    if [ "$#" -eq 0 ]; then
        echo $"Usage: killproc [-p pidfile] [ -d delay] {program} [-signal]"
        return 1
    fi
    if [ "$1" = "-p" ]; then
```

```

        pid_file=$2
        shift 2
    fi
    if [ "$1" = "-b" ]; then
        if [ -z $pid_file ]; then
            echo "$-b option can be used only with -p"
            echo "Usage: killproc -p pidfile -b binary program"
            return 1
        fi
        binary=$2
        shift 2
    fi
    if [ "$1" = "-d" ]; then
        delay=$(echo $2 | awk -v RS=' ' -v IGNORECASE=1 '{if($1!~/^[0-9.]+[smhd]?$/ ) exit
1;d=$1~/s$/?1:$1~/m$/?60:$1~/h$/?60*60:$1~/d$/?24*60*60:-1;if(d==-1) exit 1;delay+=d*$1} END
{printf("%d",delay+0.5)}')
        if [ "$?" -eq 1 ]; then
            echo "Usage: killproc [-p pidfile] [-d delay] {program} [-signal]"
            return 1
        fi
        shift 2
    ...

```

et

```

...
pidofproc() {
    local RC pid pid_file=

    # Test syntax.
    if [ "$#" = 0 ]; then
        echo "Usage: pidofproc [-p pidfile] {program}"
        return 1
    fi

```

```
    if [ "$1" = "-p" ]; then
        pid_file=$2
        shift 2
    fi
    fail_code=3 # "Program is not running"

    # First try "/var/run/*.pid" files
    __pids_var_run "$1" "$pid_file"
    RC=$?
    if [ -n "$pid" ]; then
        echo $pid
        return 0
    fi

    [ -n "$pid_file" ] && return $RC
    __pids_pidof "$1" || return $RC
}
...
```

Codes Retour Standardisés

Les codes retour standardisés sont :

Code Retour	Description
0	Le programme fonctionne et le service est correctement démarré
1	Le programme est mort et le fichier pid dans /var/run existe
2	Le programme est mort et le fichier verrou dans /var/lock existe
3	Le programme ne fonctionne pas et le service n'est pas correctement démarré
4	Le statut du programme ou du service est inconnu
5 - 99	Réservés pour LSB
100-149	Réserves pour la distribution
150-199	Réservés pour l'application

Code Retour	Description
200-254	Réservés

Scripts

LSB stipule un format *rigide* de script qui commence par une section délimitée par deux clauses :

- **### BEGIN INIT INFO,**
- **### END INIT INFO.**

Par exemple :

```
[root@centos5 ~]# more /etc/rc.d/init.d/sshd
#!/bin/bash
#
# sshd      Start up the OpenSSH server daemon
#
# chkconfig: 2345 55 25
# description: SSH is a protocol for secure remote shell access. \
#              This service starts up the OpenSSH server daemon.
#
# processname: sshd
# config: /etc/ssh/ssh_host_key
# config: /etc/ssh/ssh_host_key.pub
# config: /etc/ssh/ssh_random_seed
# config: /etc/ssh/sshd_config
# pidfile: /var/run/sshd.pid

### BEGIN INIT INFO
# Provides: sshd
# Required-Start: $local_fs $network $syslog
# Required-Stop: $local_fs $syslog
# Should-Start: $syslog
# Should-Stop: $network $syslog
```

```
# Default-Start: 2 3 4 5
# Default-Stop: 0 1 6
# Short-Description: Start up the OpenSSH server daemon
# Description:      SSH is a protocol for secure remote shell access.
#                  This service starts up the OpenSSH server daemon.
### END INIT INFO

# source function library
. /etc/rc.d/init.d/functions

# pull in sysconfig settings
[ -f /etc/sysconfig/sshd ] && . /etc/sysconfig/sshd
--Plus-- (20%)
```

Les lignes se trouvant entre les deux clauses ont un format spécifique :

```
# {MotClef}: valeur1 [valeur2...]
```



Important : Notez qu'à part la deuxième ligne de la description, chaque ligne **doit** commencer par le caractère **#** suivi par un espace.

Les Mots Clefs sont :

Mot Clef	Description
Provides	Indique le service -(<i>boot facilities</i> en anglais) démarré par le script. Le nom doit être unique.
Required-start	Indique d'autres services qui doivent être démarrés avant le démarrage de celui indiqué par le mot-clef Provides .
Required-Stop	Indique d'autres services qui doivent être arrêtés après l'arrêt de celui indiqué par le mot-clef Provides .
Should-Start	Indique d'autres services qui, s'ils sont présents sur le système, être démarrés avant le démarrage de celui indiqué par le mot-clef Provides .
Should-Stop	Indique d'autres services qui, s'ils sont présents sur le système, doivent être arrêtés après l'arrêt de celui indiqué par le mot-clef Provides .
Default-Start	Indique les niveaux d'exécution dans lesquels le service doit être démarré.

Mot Clef	Description
Default-Stop	Indique les niveaux d'exécution dans lesquels le service doit être arrêté.
Short-Description	Indique une description du service en une seule ligne .
Description	Indique une description multi-lignes du service.

Il existe des groupements de services, appelés en anglais des *Virtual Facilities*. Les plus importants sont :

Virtual Facility	Description
\$local_fs	Tous les systèmes de fichiers locaux doivent être montés.
\$network	La carte Ethernet doit fonctionner
\$named	Les daemons, si présents, responsables de la résolution des noms tels DNS, NIS+ ou LDAP doivent être démarrés.
\$portmap	Les daemons qui fournissent le <i>SunRPC/ONCRPC port mapping</i> doivent être démarrés.
\$remote_fs	Tous les systèmes de fichiers doivent être montés.
\$syslog	Syslog, Syslog-ng ou Rsyslog doit être démarré.
\$time	L'heure du système doit avoir été fixé soit par NTP, soit par rdate soit par l'horloge système.

Répertoires rcx.d

Les répertoires **rc0.d** à **rc6.d** contiennent des liens vers les scripts du répertoire **init.d**.

Pour mieux comprendre, saisissez les commandes suivantes :

```
[root@centos5 rc.d]# for rep in rc[345].d; do echo "dans $rep :"; ls $rep/S*; done
dans rc3.d :
rc3.d/S00microcode_ctl  rc3.d/S10network      rc3.d/S13irqbalance    rc3.d/S24avahi-daemon  rc3.d/S25netfs
rc3.d/S30vboxadd        rc3.d/S80postfix
rc3.d/S01sysstat        rc3.d/S11auditd       rc3.d/S13rpcbind        rc3.d/S24nfslock       rc3.d/S26acpid
rc3.d/S30vboxadd-x11    rc3.d/S82abrttd
rc3.d/S02lvm2-monitor   rc3.d/S11portreserve  rc3.d/S15mdmonitor     rc3.d/S24rpcgssd       rc3.d/S26haldaemon
rc3.d/S35vboxadd-service rc3.d/S90crond
rc3.d/S08ip6tables      rc3.d/S12rsyslog      rc3.d/S22messagebus    rc3.d/S24rpcidmapd     rc3.d/S26udev-post
rc3.d/S50bluetooth      rc3.d/S95atd
```

rc3.d/S08iptables	rc3.d/S13cpuspeed	rc3.d/S23NetworkManager	rc3.d/S25cups	rc3.d/S28autofs
rc3.d/S55sshd	rc3.d/S99local			
dans rc4.d :				
rc4.d/S00microcode_ctl	rc4.d/S10network	rc4.d/S13irqbalance	rc4.d/S24avahi-daemon	rc4.d/S25netfs
rc4.d/S30vboxadd	rc4.d/S90crond			
rc4.d/S01sysstat	rc4.d/S11auditd	rc4.d/S13rpcbind	rc4.d/S24nfslock	rc4.d/S26acpid
rc4.d/S35vboxadd-service	rc4.d/S95atd			
rc4.d/S02lvm2-monitor	rc4.d/S11portreserve	rc4.d/S15mdmonitor	rc4.d/S24rpcgssd	rc4.d/S26haldaemon
rc4.d/S50bluetooth	rc4.d/S99local			
rc4.d/S08ip6tables	rc4.d/S12rsyslog	rc4.d/S22messagebus	rc4.d/S24rpcidmapd	rc4.d/S26udev-post
rc4.d/S55sshd				
rc4.d/S08iptables	rc4.d/S13cpuspeed	rc4.d/S23NetworkManager	rc4.d/S25cups	rc4.d/S28autofs
rc4.d/S80postfix				
dans rc5.d :				
rc5.d/S00microcode_ctl	rc5.d/S10network	rc5.d/S13irqbalance	rc5.d/S24avahi-daemon	rc5.d/S25netfs
rc5.d/S30vboxadd	rc5.d/S80postfix			
rc5.d/S01sysstat	rc5.d/S11auditd	rc5.d/S13rpcbind	rc5.d/S24nfslock	rc5.d/S26acpid
rc5.d/S30vboxadd-x11	rc5.d/S82abrt			
rc5.d/S02lvm2-monitor	rc5.d/S11portreserve	rc5.d/S15mdmonitor	rc5.d/S24rpcgssd	rc5.d/S26haldaemon
rc5.d/S35vboxadd-service	rc5.d/S90crond			
rc5.d/S08ip6tables	rc5.d/S12rsyslog	rc5.d/S22messagebus	rc5.d/S24rpcidmapd	rc5.d/S26udev-post
rc5.d/S50bluetooth	rc5.d/S95atd			
rc5.d/S08iptables	rc5.d/S13cpuspeed	rc5.d/S23NetworkManager	rc5.d/S25cups	rc5.d/S28autofs
rc5.d/S55sshd	rc5.d/S99local			



Important : Notez que chaque répertoire correspondant à un niveau d'exécution contient des liens pointant vers un script dans le répertoire **/etc/init.d**. La lettre **S** indique au script **rc** que le script dans **/etc/rc.d/init.d** doit être exécutée avec l'option **start**. De cette façon les processus sont lancés dans le niveau d'exécution spécifié. Le numéro qui suit la lettre **S** indique l'ordre de lancement par le script **rc**. Si deux scripts dans un répertoire **/etc/rc.d/rcX.d** ont le même numéro, l'ordre alphabétique prime. Notez aussi la présence du lien **S99local** qui lance le script **rc.local** en dernier. Le script **rc.local** est lancé dans les niveaux d'exécution **2, 3, 4 et 5**. C'est dans ce script que **root** peut ajouter des commandes.

Rappelez la commande précédente et modifiez la lettre S en **K** :

```
[root@centos5 rc.d]# for rep in rc[345].d; do echo "dans $rep :"; ls $rep/K*; done
dans rc3.d :
rc3.d/K01certmonger rc3.d/K10saslauthd rc3.d/K50snmpd rc3.d/K73ypbind rc3.d/K80kdump
rc3.d/K87restorecond rc3.d/K95firstboot
rc3.d/K01smartd rc3.d/K15httpd rc3.d/K50snmptrapd rc3.d/K74nscd rc3.d/K80sssd
rc3.d/K88nslcd
rc3.d/K02oddjobd rc3.d/K50dnsmasq rc3.d/K60nfs rc3.d/K74ntpd rc3.d/K84wpa_supplciant
rc3.d/K89rdisc
rc3.d/K10psacct rc3.d/K50netconsole rc3.d/K69rpcsvcgssd rc3.d/K75ntpdate rc3.d/K86cgred
rc3.d/K95cgconfig
dans rc4.d :
rc4.d/K01certmonger rc4.d/K10saslauthd rc4.d/K50netconsole rc4.d/K69rpcsvcgssd rc4.d/K74ntpd
rc4.d/K84wpa_supplciant rc4.d/K89rdisc
rc4.d/K01smartd rc4.d/K15httpd rc4.d/K50snmpd rc4.d/K70vboxadd-x11 rc4.d/K75ntpdate
rc4.d/K86cgred rc4.d/K95cgconfig
rc4.d/K02oddjobd rc4.d/K16abrtcd rc4.d/K50snmptrapd rc4.d/K73ypbind rc4.d/K80kdump
rc4.d/K87restorecond rc4.d/K95firstboot
rc4.d/K10psacct rc4.d/K50dnsmasq rc4.d/K60nfs rc4.d/K74nscd rc4.d/K80sssd
rc4.d/K88nslcd
dans rc5.d :
rc5.d/K01certmonger rc5.d/K10saslauthd rc5.d/K50snmpd rc5.d/K73ypbind rc5.d/K80kdump
rc5.d/K87restorecond rc5.d/K95firstboot
rc5.d/K01smartd rc5.d/K15httpd rc5.d/K50snmptrapd rc5.d/K74nscd rc5.d/K80sssd
rc5.d/K88nslcd
rc5.d/K02oddjobd rc5.d/K50dnsmasq rc5.d/K60nfs rc5.d/K74ntpd rc5.d/K84wpa_supplciant
rc5.d/K89rdisc
rc5.d/K10psacct rc5.d/K50netconsole rc5.d/K69rpcsvcgssd rc5.d/K75ntpdate rc5.d/K86cgred
rc5.d/K95cgconfig
```



Important : Ici le principe est le même sauf que la lettre **K** indique au script **rc** que le script dans **/etc/rc.d/init.d** doit être lancé avec l'option **stop**.

rc.local

Le script rc.local est lancé dans les niveaux d'exécution **2, 3, 4 et 5**. C'est dans ce script que **root** peut ajouter des commandes.

La commande chkconfig

Pour avoir une vue globale des services lancés par niveau d'exécution nous pouvons utiliser la commande **chkconfig**. Saisissez la commande suivante :

```
[root@centos5 rc.d]# chkconfig --list
NetworkManager 0:arrêt 1:arrêt 2:marche 3:marche 4:marche 5:marche 6:arrêt
abrtd           0:arrêt 1:arrêt 2:arrêt 3:marche 4:arrêt 5:marche 6:arrêt
acpid           0:arrêt 1:arrêt 2:marche 3:marche 4:marche 5:marche 6:arrêt
atd             0:arrêt 1:arrêt 2:arrêt 3:marche 4:marche 5:marche 6:arrêt
auditd          0:arrêt 1:arrêt 2:marche 3:marche 4:marche 5:marche 6:arrêt
autofs          0:arrêt 1:arrêt 2:arrêt 3:marche 4:marche 5:marche 6:arrêt
avahi-daemon    0:arrêt 1:arrêt 2:arrêt 3:marche 4:marche 5:marche 6:arrêt
bluetooth       0:arrêt 1:arrêt 2:arrêt 3:marche 4:marche 5:marche 6:arrêt
certmonger      0:arrêt 1:arrêt 2:arrêt 3:arrêt 4:arrêt 5:arrêt 6:arrêt
cgconfig        0:arrêt 1:arrêt 2:arrêt 3:arrêt 4:arrêt 5:arrêt 6:arrêt
cgmred          0:arrêt 1:arrêt 2:arrêt 3:arrêt 4:arrêt 5:arrêt 6:arrêt
cpuspeed        0:arrêt 1:marche 2:marche 3:marche 4:marche 5:marche 6:arrêt
crond           0:arrêt 1:arrêt 2:marche 3:marche 4:marche 5:marche 6:arrêt
cups            0:arrêt 1:arrêt 2:marche 3:marche 4:marche 5:marche 6:arrêt
dnsmasq         0:arrêt 1:arrêt 2:arrêt 3:arrêt 4:arrêt 5:arrêt 6:arrêt
firstboot       0:arrêt 1:arrêt 2:arrêt 3:arrêt 4:arrêt 5:arrêt 6:arrêt
haldaemon       0:arrêt 1:arrêt 2:arrêt 3:marche 4:marche 5:marche 6:arrêt
httpd           0:arrêt 1:arrêt 2:arrêt 3:arrêt 4:arrêt 5:arrêt 6:arrêt
ip6tables       0:arrêt 1:arrêt 2:marche 3:marche 4:marche 5:marche 6:arrêt
iptables        0:arrêt 1:arrêt 2:marche 3:marche 4:marche 5:marche 6:arrêt
irqbalance      0:arrêt 1:arrêt 2:arrêt 3:marche 4:marche 5:marche 6:arrêt
```

kdump	0:arrêt	1:arrêt	2:arrêt	3:arrêt	4:arrêt	5:arrêt	6:arrêt
lvm2-monitor	0:arrêt	1:marche	2:marche	3:marche	4:marche	5:marche	6:arrêt
mdmonitor	0:arrêt	1:arrêt	2:marche	3:marche	4:marche	5:marche	6:arrêt
messagebus	0:arrêt	1:arrêt	2:marche	3:marche	4:marche	5:marche	6:arrêt
microcode_ctl	0:arrêt	1:arrêt	2:marche	3:marche	4:marche	5:marche	6:arrêt
netconsole	0:arrêt	1:arrêt	2:arrêt	3:arrêt	4:arrêt	5:arrêt	6:arrêt
netfs	0:arrêt	1:arrêt	2:arrêt	3:marche	4:marche	5:marche	6:arrêt
network	0:arrêt	1:arrêt	2:marche	3:marche	4:marche	5:marche	6:arrêt
nfs	0:arrêt	1:arrêt	2:arrêt	3:arrêt	4:arrêt	5:arrêt	6:arrêt
nfslock	0:arrêt	1:arrêt	2:arrêt	3:marche	4:marche	5:marche	6:arrêt
nscd	0:arrêt	1:arrêt	2:arrêt	3:arrêt	4:arrêt	5:arrêt	6:arrêt
nsd	0:arrêt	1:arrêt	2:arrêt	3:arrêt	4:arrêt	5:arrêt	6:arrêt
ntpd	0:arrêt	1:arrêt	2:arrêt	3:arrêt	4:arrêt	5:arrêt	6:arrêt
ntpddate	0:arrêt	1:arrêt	2:arrêt	3:arrêt	4:arrêt	5:arrêt	6:arrêt
odddjobd	0:arrêt	1:arrêt	2:arrêt	3:arrêt	4:arrêt	5:arrêt	6:arrêt
portreserve	0:arrêt	1:arrêt	2:marche	3:marche	4:marche	5:marche	6:arrêt
postfix	0:arrêt	1:arrêt	2:marche	3:marche	4:marche	5:marche	6:arrêt
psacct	0:arrêt	1:arrêt	2:arrêt	3:arrêt	4:arrêt	5:arrêt	6:arrêt
rdisc	0:arrêt	1:arrêt	2:arrêt	3:arrêt	4:arrêt	5:arrêt	6:arrêt
restorecond	0:arrêt	1:arrêt	2:arrêt	3:arrêt	4:arrêt	5:arrêt	6:arrêt
rpcbind	0:arrêt	1:arrêt	2:marche	3:marche	4:marche	5:marche	6:arrêt
rpcgssd	0:arrêt	1:arrêt	2:arrêt	3:marche	4:marche	5:marche	6:arrêt
rpcidmapd	0:arrêt	1:arrêt	2:arrêt	3:marche	4:marche	5:marche	6:arrêt
rpcsvcgssd	0:arrêt	1:arrêt	2:arrêt	3:arrêt	4:arrêt	5:arrêt	6:arrêt
rsyslog	0:arrêt	1:arrêt	2:marche	3:marche	4:marche	5:marche	6:arrêt
saslauthd	0:arrêt	1:arrêt	2:arrêt	3:arrêt	4:arrêt	5:arrêt	6:arrêt
smartd	0:arrêt	1:arrêt	2:arrêt	3:arrêt	4:arrêt	5:arrêt	6:arrêt
snmpd	0:arrêt	1:arrêt	2:arrêt	3:arrêt	4:arrêt	5:arrêt	6:arrêt
snmptrapd	0:arrêt	1:arrêt	2:arrêt	3:arrêt	4:arrêt	5:arrêt	6:arrêt
sshd	0:arrêt	1:arrêt	2:marche	3:marche	4:marche	5:marche	6:arrêt
sssd	0:arrêt	1:arrêt	2:arrêt	3:arrêt	4:arrêt	5:arrêt	6:arrêt
sysstat	0:arrêt	1:marche	2:marche	3:marche	4:marche	5:marche	6:arrêt
udev-post	0:arrêt	1:marche	2:marche	3:marche	4:marche	5:marche	6:arrêt
vboxadd	0:arrêt	1:arrêt	2:marche	3:marche	4:marche	5:marche	6:arrêt

vboxadd-service	0:arrêt	1:arrêt	2:marche	3:marche	4:marche	5:marche	6:arrêt
vboxadd-x11	0:arrêt	1:arrêt	2:arrêt	3:marche	4:arrêt	5:marche	6:arrêt
wpa_supplicant	0:arrêt	1:arrêt	2:arrêt	3:arrêt	4:arrêt	5:arrêt	6:arrêt
ypbind	0:arrêt	1:arrêt	2:arrêt	3:arrêt	4:arrêt	5:arrêt	6:arrêt

Options de la commande

Les options de la commande **chkconfig** sont :

```
[root@centos5 ~]# chkconfig --help
chkconfig version 1.3.47 - Copyright (C) 1997-2000 Red Hat, Inc.
Ce logiciel peut être librement distribué selon les termes de la licence publique GNU (GPL).

utilisation :  chkconfig [--list] [--type <type>] [nom]
               chkconfig --add <nom>
               chkconfig --del <nom>
               chkconfig --override <name>
               chkconfig [--level <niveaux>] [--type <type>] <nom> <on|off|reset|resetpriorities>
```

Les options les plus importantes sont :

Option	Description
- -add (nom)	Permet d'ajouter un service aux runlevels définis par le service lui-même
- -del (nom)	Permet de supprimer un service aux runlevels définis par le service lui-même
- -level [niveau] (nom) [on off reset]	Permet d'activer, de désactiver ou de réinitialiser un service inscrit

Démarrer RHEL/CentOS 6 avec Upstart

Upstart est un processus Init mieux adapté aux exigences des périphériques modernes de type plug'n'play. **Upstart** peut faire tout ce que fait le processus Init **SysVinit** mais supporte d'avantage d'**actions**. Par exemple, Upstart est capable de démarrer un service quand un périphérique spécifique est branché à chaud. Upstart est néanmoins compatible avec les scripts de démarrage du système **SysVinit** et ceux-ci fonctionnent

normalement en attendant que les éditeurs de logiciels tierces portent les scripts SysVinit vers Upstart.

Inittab

Sous Upstart, le fichier `/etc/inittab` ne permet **plus** de définir les services à démarrer en fonction du RUNLEVEL, seule la définition d'**initdefault** y est spécifiée :

```
[root@centos6 ~]# cat /etc/inittab
# inittab is only used by upstart for the default runlevel.
#
# ADDING OTHER CONFIGURATION HERE WILL HAVE NO EFFECT ON YOUR SYSTEM.
#
# System initialization is started by /etc/init/rcS.conf
#
# Individual runlevels are started by /etc/init/rc.conf
#
# Ctrl-Alt-Delete is handled by /etc/init/control-alt-delete.conf
#
# Terminal gettys are handled by /etc/init/tty.conf and /etc/init/serial.conf,
# with configuration in /etc/sysconfig/init.
#
# For information on how to write upstart event handlers, or how
# upstart works, see init(5), init(8), and initctl(8).
#
# Default runlevel. The runlevels used are:
# 0 - halt (Do NOT set initdefault to this)
# 1 - Single user mode
# 2 - Multiuser, without NFS (The same as 3, if you do not have networking)
# 3 - Full multiuser mode
# 4 - unused
# 5 - X11
# 6 - reboot (Do NOT set initdefault to this)
#
```

```
id:5:initdefault:
```

```
[root@centos6 ~]# cat /etc/inittab
# inittab is only used by upstart for the default runlevel.
#
# ADDING OTHER CONFIGURATION HERE WILL HAVE NO EFFECT ON YOUR SYSTEM.
#
# System initialization is started by /etc/init/rcS.conf
#
# Individual runlevels are started by /etc/init/rc.conf
#
# Ctrl-Alt-Delete is handled by /etc/init/control-alt-delete.conf
#
# Terminal gettys are handled by /etc/init/tty.conf and /etc/init/serial.conf,
# with configuration in /etc/sysconfig/init.
#
# For information on how to write upstart event handlers, or how
# upstart works, see init(5), init(8), and initctl(8).
#
# Default runlevel. The runlevels used are:
# 0 - halt (Do NOT set initdefault to this)
# 1 - Single user mode
# 2 - Multiuser, without NFS (The same as 3, if you do not have networking)
# 3 - Full multiuser mode
# 4 - unused
# 5 - X11
# 6 - reboot (Do NOT set initdefault to this)
#
id:5:initdefault:
```

Sous Upstart, les définitions des autres valeurs habituellement incluses dans `/etc/inittab` se trouvent dans des fichiers de configuration supplémentaires qui se trouvent dans le répertoire **/etc/init/** :

```
[root@centos6 ~]# ls -l /etc/init
```

```
total 68
-rw-r--r--. 1 root root 412 22 juil. 2014 control-alt-delete.conf
-rw-r--r--. 1 root root 130 12 mars 2014 init-system-dbus.conf
-rw-r--r--. 1 root root 463 22 juil. 2014 kexec-disable.conf
-rw-r--r--. 1 root root 560 22 juil. 2014 plymouth-shutdown.conf
-rw-r--r--. 1 root root 357 22 juil. 2014 prefdm.conf
-rw-r--r--. 1 root root 505 22 juil. 2014 quit-plymouth.conf
-rw-r--r--. 1 root root 417 22 juil. 2014 rc.conf
-rw-r--r--. 1 root root 1046 22 juil. 2014 rcS.conf
-rw-r--r--. 1 root root 430 22 juil. 2014 rcS-emergency.conf
-rw-r--r--. 1 root root 725 22 juil. 2014 rcS-sulogin.conf
-rw-r--r--. 1 root root 2915 23 nov. 2013 readahead-collector.conf
-rw-r--r--. 1 root root 1559 23 nov. 2013 readahead.conf
-rw-r--r--. 1 root root 726 23 nov. 2013 readahead-disable-services.conf
-rw-r--r--. 1 root root 1302 22 juil. 2014 serial.conf
-rw-r--r--. 1 root root 791 22 juil. 2014 splash-manager.conf
-rw-r--r--. 1 root root 473 22 juil. 2014 start-ttys.conf
-rw-r--r--. 1 root root 335 22 juil. 2014 tty.conf
```

Initialisation du Système

L'initialisation du système est configurée dans le fichier **/etc/init/rcS.conf** :

```
[root@centos6 init]# cat /etc/init/rcS.conf
# rcS - runlevel compatibility
#
# This task runs the old sysv-rc startup scripts.

start on startup

stop on runlevel

task
```

```
# Note: there can be no previous runlevel here, if we have one it's bad
# information (we enter rc1 not rcS for maintenance). Run /etc/rc.d/rc
# without information so that it defaults to previous=N runlevel=S.
console output
exec /etc/rc.d/rc.sysinit
post-stop script
    if [ "$UPSTART_EVENTS" = "startup" ]; then
        [ -f /etc/inittab ] && runlevel=$(/bin/awk -F ':' '$3 == "initdefault" && $1 !~ "^#" { print $2 }'
/etc/inittab)
        [ -z "$runlevel" ] && runlevel="3"
        for t in $(cat /proc/cmdline); do
            case $t in
                -s|single|S|s) runlevel="S" ;;
                [1-9])      runlevel="$t" ;;
            esac
        done
        exec telinit $runlevel
    fi
end script
```

Runlevels

La gestion des Runlevels est configurée dans le fichier **/etc/init/rc.conf** :

```
[root@centos6 init]# cat /etc/init/rc.conf
# rc - System V runlevel compatibility
#
# This task runs the old sysv-rc runlevel scripts. It
# is usually started by the telinit compatibility wrapper.

start on runlevel [0123456]

stop on runlevel [!$RUNLEVEL]
```

```
task
```

```
export RUNLEVEL
console output
exec /etc/rc.d/rc $RUNLEVEL
```

[CTL]-[ALT]-[DEL]

Le comportement associé avec la combinaison de touches [CTL]-[ALT]-[DEL] est configuré dans le fichier **/etc/init/control-alt-delete.conf** :

```
[root@centos6 init]# cat /etc/init/control-alt-delete.conf
# control-alt-delete - emergency keypress handling
#
# This task is run whenever the Control-Alt-Delete key combination is
# pressed. Usually used to shut down the machine.

start on control-alt-delete

exec /sbin/shutdown -r now "Control-Alt-Delete pressed"
```

mingetty

Le génération des terminaux getty est spécifiée dans les fichiers **/etc/init/tty.conf** et **/etc/init/serial.conf** :

```
[root@centos6 init]# cat /etc/init/tty.conf
# tty - getty
#
# This service maintains a getty on the sepcified device.

stop on runlevel [016]
```

```
respawn
instance $TTY
exec /sbin/mingetty $TTY
[root@centos rc.d]# cat /etc/init/serial.conf
# Automatically start a configured serial console
#
# How this works:
#
# On boot, a udev helper examines /dev/console. If a serial console is the
# primary console (last console on the commandline in grub), the event
# 'fedora.serial-console-available <port name> <speed>' is emitted, which
# triggers this script. It waits for the runlevel to finish, ensures
# the proper port is in /etc/securetty, and starts the getty.
#
# If your serial console is not the primary console, or you want a getty
# on serial even if it's not the console, create your own event by copying
# /etc/init/tty.conf, and changing the getty line in that file.

start on fedora.serial-console-available DEV=* and stopped rc RUNLEVEL=[2345]
stop on runlevel [016]

instance $DEV
respawn
pre-start exec /sbin/securetty $DEV
exec /sbin/agetty /dev/$DEV $SPEED vt100-nav
```

La configuration des terminaux fait partie du fichier **/etc/sysconfig/init**

```
[root@centos6 init]# cat /etc/sysconfig/init
# color => new RH6.0 bootup
# verbose => old-style bootup
# anything else => new style bootup without ANSI colors or positioning
BOOTUP=color
# column to start "[ OK ]" label in
```

```
RES_COL=60
# terminal sequence to move to that column. You could change this
# to something like "tput hpa ${RES_COL}" if your terminal supports it
MOVE_TO_COL="echo -en \\033[${RES_COL}G"
# terminal sequence to set color to a 'success' color (currently: green)
SETCOLOR_SUCCESS="echo -en \\033[0;32m"
# terminal sequence to set color to a 'failure' color (currently: red)
SETCOLOR_FAILURE="echo -en \\033[0;31m"
# terminal sequence to set color to a 'warning' color (currently: yellow)
SETCOLOR_WARNING="echo -en \\033[0;33m"
# terminal sequence to reset to the default color.
SETCOLOR_NORMAL="echo -en \\033[0;39m"
# Set to anything other than 'no' to allow hotkey interactive startup...
PROMPT=yes
# Set to 'yes' to allow probing for devices with swap signatures
AUTOSWAP=no
# What ttys should gettys be started on?
ACTIVE_CONSOLES=/dev/tty[1-6]
# Set to '/sbin/sulogin' to prompt for password on single-user mode
# Set to '/sbin/sushell' otherwise
SINGLE=/sbin/sushell
```

Gestion des Services

Sous Upstart, les services sont appelés des **jobs**. Les scripts de démarrage de jobs au format Upstart sont placés dans le répertoire **/etc/init/** et ont une forme **nom.conf** où *nom* est le nom du job :

```
[root@centos6 ~]# ls /etc/init
control-alt-delete.conf  rcS-sulogin.conf
init-system-dbus.conf    readahead-collector.conf
kexec-disable.conf       readahead.conf
plymouth-shutdown.conf   readahead-disable-services.conf
prefdm.conf              serial.conf
```

```
quit-plymouth.conf    splash-manager.conf
rc.conf               start-ttys.conf
rcS.conf              tty.conf
rcS-emergency.conf
```

Pour créer un fichier job, il convient de respecter un certain format. Par exemple créez le fichier **/etc/init/testjob.conf** :

[testjob.conf](#)

```
description "Un job pour tester Upstart"
author "Linux E-Learning"
start on runlevel [2345]
exec echo Le job test a été lancé le `date` >> /var/log/testjob.log
```

La Commande initctl

Pour obtenir une liste de tous les jobs et leurs états, il convient d'utiliser la commande `initctl`. Vérifiez donc que le job `testjob` se trouve dans la sortie de cette commande :

```
[root@centos6 ~]# initctl list
rc stop/waiting
tty (/dev/tty3) start/running, process 1833
tty (/dev/tty2) start/running, process 1828
tty (/dev/tty6) start/running, process 1854
tty (/dev/tty5) start/running, process 1845
tty (/dev/tty4) start/running, process 1838
plymouth-shutdown stop/waiting
control-alt-delete stop/waiting
rcS-emergency stop/waiting
readahead-collector stop/waiting
kexec-disable stop/waiting
```

```
quit-plymouth stop/waiting
testjob stop/waiting
rcS stop/waiting
prefdm start/running, process 1814
init-system-dbus stop/waiting
readahead stop/waiting
splash-manager stop/waiting
start-ttys stop/waiting
readahead-disable-services stop/waiting
rcS-sulogin stop/waiting
serial stop/waiting
```

Option de la Commande **initctl**

Les options de la commande **initctl** sont :

```
[root@centos6 ~]# initctl --help
Usage: initctl [OPTION]... COMMAND [OPTION]... [ARG]...

Options:
  --system          use D-Bus system bus to connect to init daemon
  --dest=NAME       destination well-known name on system bus
  -q, --quiet        reduce output to errors only
  -v, --verbose      increase output to include informational messages
  --help            display this help and exit
  --version          output version information and exit
```

For a list of commands, try `initctl help'.

Report bugs at <<https://launchpad.net/upstart/+bugs>>

Jobs

Il existe trois types de jobs sous Upstart :

- **task**
- **service**
 - un service job peut fonctionner en arrière plan
- **abstract**
 - un abstract job est un service qui fonctionnent jusqu'à ce que l'administrateur l'arrête

Événements

Un événement ou *event* en anglais est un signal envoyé vers un job pour déclencher une action. Par exemple :

- **starting**
- **started**
- **stopping**
- **stopped**

Etats

L'objectif d'un job est de démarrer ou de s'arrêter. Entre ces deux objectifs se trouvent des états intermédiaires :

- **waiting** - l'état initial d'un job
 - **starting** - le job est sur le point de démarrer
 - **pre-start** - la section pre-start est chargée
 - **spawned** - une section script est sur le point de démarrer
 - **post-start** - les opérations détaillées dans la section post-start ont lieu
 - **running** - le job est opérationnel
 - **pre-stop** - la section pre-stop est chargée
 - **stopping** - le job est en cours d'arrêt
 - **killed** - le job est arrêté
-

- **post-stop** - les opérations détaillées dans la section post-stop ont lieu

Démarrer et Arrêter les Jobs

Compte tenu de la compatibilité avec le système SysVinit, les commandes traditionnellement utilisées avec ce dernier sont compris par Upstart. Ceci étant Upstart fournit ses propres commandes pour le contrôle des jobs.

La Commande status

Pour voir le status d'un job spécifique, utilisez la commande suivante :

```
[root@centos6 ~]# status testjob
testjob stop/waiting
```

Options de la Commande status

Les options de la commande **status** sont :

```
[root@centos6 ~]# status --help
Usage: status [OPTION]... JOB [KEY=VALUE]...
Query status of job.

Options:
  --system          use D-Bus system bus to connect to init daemon
  --dest=NAME       destination well-known name on system bus
  -q, --quiet        reduce output to errors only
  -v, --verbose      increase output to include informational messages
  --help            display this help and exit
  --version          output version information and exit
```

JOB is the name of the job that is to be queried, this may be followed by zero or more environment variables to distinguish between job instances.

Report bugs at <<https://launchpad.net/upstart/+bugs>>

La Commande start

```
[root@centos6 ~]# start testjob
testjob start/running, process 2981
[root@centos6 ~]# cat /var/log/testjob.log
Le job test a été lancé le Tue Jun 30 15:26:10 CEST 2015
```

Options de la Commande start

Les options de la commande **start** sont :

```
[root@centos6 ~]# start --help
Usage: start [OPTION]... JOB [KEY=VALUE]...
Start job.
```

Options:

-n, --no-wait	do not wait for job to start before exiting
--system	use D-Bus system bus to connect to init daemon
--dest=NAME	destination well-known name on system bus
-q, --quiet	reduce output to errors only
-v, --verbose	increase output to include informational messages
--help	display this help and exit
--version	output version information and exit

JOB is the name of the job that is to be started, this may be followed by zero

or more environment variables to be defined in the new job.

The environment may also serve to distinguish between job instances, and thus decide whether a new instance will be started or an error returned if an existing instance is already running.

Report bugs at <<https://launchpad.net/upstart/+bugs>>

La Commande stop

Pour arrêter un job, utilisez la commande **stop**.

Options de la Commande stop

Les options de la commande **stop** sont :

```
[root@centos6 ~]# stop --help
Usage: stop [OPTION]... JOB [KEY=VALUE]...
Stop job.
```

Options:

-n, --no-wait	do not wait for job to stop before exiting
--system	use D-Bus system bus to connect to init daemon
--dest=NAME	destination well-known name on system bus
-q, --quiet	reduce output to errors only
-v, --verbose	increase output to include informational messages
--help	display this help and exit
--version	output version information and exit

JOB is the name of the job that is to be stopped, this may be followed by zero or more environment variables to be passed to the job's pre-stop and post-stop

processes.

The environment also serves to distinguish between job instances, and thus decide which of multiple instances will be stopped.

Report bugs at <<https://launchpad.net/upstart/+bugs>>

La Commande restart

Pour redémarrer un job, utilisez la commande **restart**.

Options de la Commande restart

Les options de la commande **restart** sont :

```
[root@centos6 ~]# restart --help
Usage: restart [OPTION]... JOB [KEY=VALUE]...
Restart job.
```

Options:

-n, --no-wait	do not wait for job to restart before exiting
--system	use D-Bus system bus to connect to init daemon
--dest=NAME	destination well-known name on system bus
-q, --quiet	reduce output to errors only
-v, --verbose	increase output to include informational messages
--help	display this help and exit
--version	output version information and exit

JOB is the name of the job that is to be restarted, this may be followed by zero or more environment variables to be defined in the job after restarting.

The environment also serves to distinguish between job instances, and thus decide which of multiple instances will be restarted.

Report bugs at <<https://launchpad.net/upstart/+bugs>>



Il est aussi possible d'utiliser les commandes **initctl start**, **initctl stop** et **initctl restart** pour gérer les jobs.

Démarrer RHEL/CentOS 7

RHEL/CentOS 7, comme beaucoup d'autres distributions, ont abandonné **Upstart** pour **Systemd**. Ce dernier prend une approche différente au démarrage de Linux. En effet, **SysVinit** et **Upstart** sont des systèmes de démarrage **séquentiels**. **Systemd** essaie, par contre, de démarrer autant de services en parallèle que possible. Ceci est rendu possible car la majorité d'architectures matérielles modernes sont multi-cœurs. Si un service dépend d'un autre qui n'est pas encore démarré ce premier est mis en attente dans une mémoire tampon. Qui plus est, les services qui ne sont pas nécessaires au démarrage de la machine, tel cups, ne sont démarrés ultérieurement que si nécessaire. Lors de démarrage, les partitions sont montées en parallèle. Dernièrement, **Systemd** remplace les scripts de démarrage traditionnels avec des binaires compilés, beaucoup plus rapides que leur prédécesseurs.

Au lieu de parler de scripts de démarrage et de niveaux d'exécution, **Systemd** utilise la terminologie **Unités** (*Units*) et **Cibles** (*Targets*). Une Cible est en quelque sorte une **grande étape** dans le démarrage du système tandis qu'une Unité peut être :

- un automount - (*.automount*),
- une périphérique - *Device* - (*.device*),
- un montage d'un périphérique - *Mount* - (*.mount*),
- un chemin - *Path* - (*.path*)
- un socket - *Socket* - (*.socket*),
- un service - *Service* - (*.service*),
- une instantanée - *Snapshot* - (*.snapshot*),
- une cible - *Target* - (*.target*).



Important : Dans le contexte d'une Unité, le type **cible** regroupe des Unités multiples afin qu'elles puissent être démarrées en même temps. Par exemple **network.target** regroupe toutes les Unités nécessaires pour démarrer toutes les interfaces réseaux en même temps.

La Commande `systemctl`

Pour visualiser la liste des Unités, il convient d'utiliser la commande **`systemctl`** avec l'option **`list-units`** :

```
[root@centos7 ~]# systemctl list-units
```

UNIT	LOAD	ACTIVE	SUB	DESCRIPTION
proc-sys-fs-binfmt_misc.automount	loaded	active	waiting	Arbitrary Executable File
Formats File System Automount Point				
sys-devices-pci0000:...t1-target1:0:0-1:0:0:0-block-sr0.device	loaded	active	plugged	VBOX_CD-ROM
sys-devices-pci0000:00-0000:00:03.0-net-enp0s3.device	loaded	active	plugged	PRO/1000 MT Desktop
Adapter				
sys-devices-pci0000:00-0000:00:05.0-sound-card0.device	loaded	active	plugged	82801AA AC'97 Audio
Controller				
sys-devices-pci0000:...rget2:0:0-2:0:0:0-block-sda-sda1.device	loaded	active	plugged	VBOX_HARDDISK
sys-devices-pci0000:...rget2:0:0-2:0:0:0-block-sda-sda2.device	loaded	active	plugged	VBOX_HARDDISK
sys-devices-pci0000:...rget2:0:0-2:0:0:0-block-sda-sda3.device	loaded	active	plugged	VBOX_HARDDISK
sys-devices-pci0000:...t2-target2:0:0-2:0:0:0-block-sda.device	loaded	active	plugged	VBOX_HARDDISK
sys-devices-platform-serial8250-tty-ttyS0.device	loaded	active	plugged	
/sys/devices/platform/serial8250/tty/ttyS0				
sys-devices-platform-serial8250-tty-ttyS1.device	loaded	active	plugged	
/sys/devices/platform/serial8250/tty/ttyS1				
sys-devices-platform-serial8250-tty-ttyS2.device	loaded	active	plugged	
/sys/devices/platform/serial8250/tty/ttyS2				
sys-devices-platform-serial8250-tty-ttyS3.device	loaded	active	plugged	
/sys/devices/platform/serial8250/tty/ttyS3				
sys-module-configfs.device	loaded	active	plugged	/sys/module/configfs
sys-module-fuse.device	loaded	active	plugged	/sys/module/fuse

sys-subsystem-net-devices-enp0s3.device Adapter	loaded	active	plugged	PR0/1000 MT Desktop
- .mount	loaded	active	mounted	/
boot.mount	loaded	active	mounted	/boot
dev-hugepages.mount	loaded	active	mounted	Huge Pages File System
dev-mqueue.mount	loaded	active	mounted	POSIX Message Queue File System
proc-fs-nfsd.mount filesystem	loaded	active	mounted	NFSD configuration
run-media-trainee-VBOXADDITIONS_4.3.28_100309.mount /run/media/trainee/VBOXADDITIONS_4.3.28_100309	loaded	active	mounted	
run-user-1000-gvfs.mount	loaded	active	mounted	/run/user/1000/gvfs
sys-fs-fuse-connections.mount	loaded	active	mounted	FUSE Control File System
sys-kernel-config.mount	loaded	active	mounted	Configuration File System
sys-kernel-debug.mount	loaded	active	mounted	Debug File System
var-lib-nfs-rpc_pipefs.mount	loaded	active	mounted	RPC Pipe File System
brandbot.path	loaded	active	waiting	Flexible branding
cups.path	loaded	active	waiting	CUPS Printer Service Spool
systemd-ask-password-plymouth.path to Plymouth Directory Watch	loaded	active	waiting	Forward Password Requests
systemd-ask-password-wall.path to Wall Directory Watch	loaded	active	waiting	Forward Password Requests
session-1.scope	loaded	active	running	Session 1 of user trainee
abrt-ccpp.service	loaded	active	exited	Install ABRT coredump hook
abrt-oops.service	loaded	active	running	ABRT kernel log watcher
abrt-xorg.service	loaded	active	running	ABRT Xorg log watcher
abrttd.service	loaded	active	running	ABRT Automated Bug Reporting Tool
accounts-daemon.service	loaded	active	running	Accounts Service
alsa-state.service (restore and store)	loaded	active	running	Manage Sound Card State
atd.service	loaded	active	running	Job spooling tools
auditd.service	loaded	active	running	Security Auditing Service
avahi-daemon.service	loaded	active	running	Avahi mDNS/DNS-SD Stack

bluetooth.service	loaded active running	Bluetooth service
chronyd.service	loaded active running	NTP client/server
colord.service	loaded active running	Manage, Install and
Generate Color Profiles		
crond.service	loaded active running	Command Scheduler
cups.service	loaded active running	CUPS Printing Service
dbus.service	loaded active running	D-Bus System Message Bus
firewalld.service	loaded active running	firewalld - dynamic
firewall daemon		
gdm.service	loaded active running	GNOME Display Manager
gssproxy.service	loaded active running	GSSAPI Proxy Daemon
iscsi-shutdown.service	loaded active exited	Logout off all iSCSI
sessions on shutdown		
kdump.service	loaded failed failed	Crash recovery kernel
arming		
kmod-static-nodes.service	loaded active exited	Create list of required
static device nodes for the current ker		
ksm.service	loaded active exited	Kernel Samepage Merging
ksmtuned.service	loaded active running	Kernel Samepage Merging
(KSM) Tuning Daemon		
libstoragemgmt.service	loaded active running	libstoragemgmt plug-in
server daemon		
libvirt.service	loaded active running	Virtualization daemon
lvm2-lvmetad.service	loaded active running	LVM2 metadata daemon

Pour consulter la liste des Unités inactifs, utilisez la commande suivante :

```
[root@centos7 ~]# systemctl list-units --all | grep inactive | more
```

proc-sys-fs-binfmt_misc.mount	loaded inactive dead
Arbitrary Executable File Formats	
File System	
systemd-ask-password-console.path	loaded inactive dead
Dispatch Password Requests to Cons	
ole Directory Watch	

abrt-vmcore.service	loaded inactive dead
Harvest vmcores for ABRT	
alsa-restore.service	loaded inactive dead
Restore Sound Card State	
alsa-store.service	loaded inactive dead
Store Sound Card State	
apparmor.service	not-found inactive dead
apparmor.service	
auth-rpcgss-module.service	loaded inactive dead
Kernel Module supporting RPCSEC_GS	
S	
brandbot.service	loaded inactive dead
Flexible Branding Service	
cpupower.service	loaded inactive dead
Configure CPU power related settin	
gs	
dm-event.service	loaded inactive dead
Device-mapper event daemon	
dmraid-activation.service	loaded inactive dead
Activation of DM RAID sets	
dracut-shutdown.service	loaded inactive dead
Restore /run/initramfs	
ebtables.service	loaded inactive dead
Ethernet Bridge Filtering tables	
emergency.service	loaded inactive dead
Emergency Shell	
exim.service	not-found inactive dead
exim.service	
getty@tty1.service	loaded inactive dead
Getty on tty1	
hypervkvpd.service	loaded inactive dead
Hyper-V KVP daemon	
hypervvssd.service	loaded inactive dead
Hyper-V VSS daemon	

ip6tables.service	loaded inactive dead
IPv6 firewall with ip6tables	
iptables.service	loaded inactive dead
IPv4 firewall with iptables	
irqbalance.service	loaded inactive dead
irqbalance daemon	
iscsi.service	loaded inactive dead
Login and scanning of iSCSI device	
s	
iscsid.service	loaded inactive dead
Open-iSCSI	
iscsiuio.service	loaded inactive dead
iSCSI UserSpace I/O driver	
--More--	

Pour consulter la liste des fichiers Unités, utilisez la commande suivante :

```
[root@centos7 ~]# systemctl list-unit-files | more
UNIT FILE                                STATE
proc-sys-fs-binfmt_misc.automount      static
dev-hugepages.mount                    static
dev-mqueue.mount                       static
proc-fs-nfsd.mount                     static
proc-sys-fs-binfmt_misc.mount           static
sys-fs-fuse-connections.mount           static
sys-kernel-config.mount                 static
sys-kernel-debug.mount                  static
tmp.mount                               disabled
var-lib-nfs-rpc_pipefs.mount            static
brandbot.path                           disabled
cups.path                               enabled
systemd-ask-password-console.path       static
systemd-ask-password-plymouth.path       static
systemd-ask-password-wall.path           static
```

```
session-1.scope          static
session-8.scope          static
abrt-ccpp.service        enabled
abrt-oops.service        enabled
abrt-pstoreoops.service  disabled
abrt-vmcore.service      enabled
abrt-xorg.service        enabled
abrttd.service           enabled
accounts-daemon.service  enabled
alsa-restore.service     static
alsa-state.service       static
alsa-store.service       static
anaconda-direct.service  static
--More--
```

Options de la Commande systemctl

Les options de la commande **systemctl** sont :

```
[root@centos7 ~]# systemctl --help
systemctl [OPTIONS...] {COMMAND} ...
```

Query or send control commands to the systemd manager.

-h --help	Show this help
--version	Show package version
-t --type=TYPE	List only units of a particular type
--state=STATE	List only units with particular LOAD or SUB or ACTIVE state
-p --property=NAME	Show only properties by this name
-a --all	Show all loaded units/properties, including dead/empty ones. To list all units installed on the system, use the 'list-unit-files' command instead.
--reverse	Show reverse dependencies with 'list-dependencies'

```
-l --full          Don't ellipsize unit names on output
--fail            When queueing a new job, fail if conflicting jobs are
                  pending
--irreversible     When queueing a new job, make sure it cannot be implicitly
                  cancelled
--ignore-dependencies
                  When queueing a new job, ignore all its dependencies
--show-types       When showing sockets, explicitly show their type
-i --ignore-inhibitors
                  When shutting down or sleeping, ignore inhibitors
--kill-who=WHO     Who to send signal to
-s --signal=SIGNAL Which signal to send
-H --host=[USER@]HOST
                  Show information for remote host
-P --privileged    Acquire privileges before execution
-q --quiet          Suppress output
--no-block         Do not wait until operation finished
--no-wall          Don't send wall message before halt/power-off/reboot
--no-reload        When enabling/disabling unit files, don't reload daemon
                  configuration
--no-legend        Do not print a legend (column headers and hints)
--no-pager         Do not pipe output into a pager
--no-ask-password  Do not ask for system passwords
--system          Connect to system manager
--user            Connect to user service manager
--global          Enable/disable unit files globally
--runtime         Enable unit files only temporarily until next reboot
-f --force         When enabling unit files, override existing symlinks
                  When shutting down, execute action immediately
--root=PATH       Enable unit files in the specified root directory
-n --lines=INTEGER Number of journal entries to show
-o --output=STRING Change journal output mode (short, short-monotonic,
                  verbose, export, json, json-pretty, json-sse, cat)
```

--plain Print unit dependencies as a list instead of a tree

Unit Commands:

list-units	List loaded units
list-sockets	List loaded sockets ordered by address
start [NAME...]	Start (activate) one or more units
stop [NAME...]	Stop (deactivate) one or more units
reload [NAME...]	Reload one or more units
restart [NAME...]	Start or restart one or more units
try-restart [NAME...]	Restart one or more units if active
reload-or-restart [NAME...]	Reload one or more units if possible, otherwise start or restart
reload-or-try-restart [NAME...]	Reload one or more units if possible, otherwise restart if active
isolate [NAME]	Start one unit and stop all others
kill [NAME...]	Send signal to processes of a unit
is-active [NAME...]	Check whether units are active
is-failed [NAME...]	Check whether units are failed
status [NAME... PID...]	Show runtime status of one or more units
show [NAME... JOB...]	Show properties of one or more units/jobs or the manager
set-property [NAME] [ASSIGNMENT...]	Sets one or more properties of a unit
help [NAME... PID...]	Show manual for one or more units
reset-failed [NAME...]	Reset failed state for all, one, or more units
list-dependencies [NAME]	Recursively show units which are required or wanted by this unit or by which this unit is required or wanted

Unit File Commands:

list-unit-files	List installed unit files
enable [NAME...]	Enable one or more unit files
disable [NAME...]	Disable one or more unit files

reenable [NAME...]	Reenable one or more unit files
preset [NAME...]	Enable/disable one or more unit files based on preset configuration
is-enabled [NAME...]	Check whether unit files are enabled
mask [NAME...]	Mask one or more units
unmask [NAME...]	Unmask one or more units
link [PATH...]	Link one or more units files into the search path
get-default	Get the name of the default target
set-default NAME	Set the default target

Job Commands:

list-jobs	List jobs
cancel [JOB...]	Cancel all, one, or more jobs

Snapshot Commands:

snapshot [NAME]	Create a snapshot
delete [NAME...]	Remove one or more snapshots

Environment Commands:

show-environment	Dump environment
set-environment [NAME=VALUE...]	Set one or more environment variables
unset-environment [NAME...]	Unset one or more environment variables

Manager Lifecycle Commands:

daemon-reload	Reload systemd manager configuration
daemon-reexec	Reexecute systemd manager

System Commands:

default	Enter system default mode
rescue	Enter system rescue mode
emergency	Enter system emergency mode
halt	Shut down and halt the system

```
poweroff      Shut down and power-off the system
reboot        Shut down and reboot the system
kexec         Shut down and reboot the system with kexec
exit          Request user instance exit
switch-root [ROOT] [INIT] Change to a different root file system
suspend       Suspend the system
hibernate     Hibernate the system
hybrid-sleep  Hibernate and suspend the system
lines 95-123/123 (END)
```

Fichiers de Configuration

Les Cibles et les Unités sont configurées par des fichiers se trouvant dans le répertoire **/etc/systemd/system** :

```
[root@centos7 ~]# ls -l /etc/systemd/system
total 12
drwxr-xr-x. 2 root root  54 Mar  8 13:57 basic.target.wants
drwxr-xr-x. 2 root root  30 Mar  8 13:53 bluetooth.target.wants
lrwxrwxrwx. 1 root root  41 Mar  8 13:53 dbus-org.bluez.service -> /usr/lib/systemd/system/bluetooth.service
lrwxrwxrwx. 1 root root  41 Mar  8 13:48 dbus-org.fedoraproject.FirewallD1.service ->
/usr/lib/systemd/system/firewalld.service
lrwxrwxrwx. 1 root root  44 Mar  8 13:48 dbus-org.freedesktop.Avahi.service -> /usr/lib/systemd/system/avahi-
daemon.service
lrwxrwxrwx. 1 root root  44 Mar  8 13:57 dbus-org.freedesktop.ModemManager1.service ->
/usr/lib/systemd/system/ModemManager.service
lrwxrwxrwx. 1 root root  46 Mar  8 13:49 dbus-org.freedesktop.NetworkManager.service ->
/usr/lib/systemd/system/NetworkManager.service
lrwxrwxrwx. 1 root root  57 Mar  8 13:49 dbus-org.freedesktop.nm-dispatcher.service ->
/usr/lib/systemd/system/NetworkManager-dispatcher.service
lrwxrwxrwx. 1 root root  36 Mar  8 14:05 default.target -> /lib/systemd/system/graphical.target
drwxr-xr-x. 2 root root  85 Mar  8 13:47 default.target.wants
lrwxrwxrwx. 1 root root  35 Mar  8 13:54 display-manager.service -> /usr/lib/systemd/system/gdm.service
drwxr-xr-x. 2 root root  31 Mar  8 13:47 getty.target.wants
```

```
drwxr-xr-x. 2 root root 63 Jun 4 14:59 graphical.target.wants
drwxr-xr-x. 2 root root 4096 Jun 4 10:00 multi-user.target.wants
drwxr-xr-x. 2 root root 29 Mar 8 13:48 nfs.target.wants
drwxr-xr-x. 2 root root 25 Mar 8 13:50 printer.target.wants
drwxr-xr-x. 2 root root 30 Jun 4 10:00 remote-fs.target.wants
drwxr-xr-x. 2 root root 4096 Mar 8 13:50 sockets.target.wants
drwxr-xr-x. 2 root root 35 Mar 8 13:57 spice-vdagentd.target.wants
drwxr-xr-x. 2 root root 4096 Mar 8 13:49 sysinit.target.wants
drwxr-xr-x. 2 root root 83 Mar 8 13:49 system-update.target.wants
```

ainsi que par des fichiers se trouvant dans le répertoire **/lib/systemd/system** et **/usr/lib/systemd/system** :

```
[root@centos7 ~]# ls -l /lib/systemd/system | more
total 1208
-rw-r--r--. 1 root root 275 Mar 24 04:56 abrt-ccpp.service
-rw-r--r--. 1 root root 380 Mar 24 04:56 abrt-d.service
-rw-r--r--. 1 root root 361 Mar 24 04:56 abrt-oops.service
-rw-r--r--. 1 root root 266 Mar 24 04:56 abrt-pstoreoops.service
-rw-r--r--. 1 root root 262 Mar 24 04:56 abrt-vmcore.service
-rw-r--r--. 1 root root 311 Mar 24 04:56 abrt-xorg.service
-rw-r--r--. 1 root root 421 Jun 10 2014 accounts-daemon.service
-rw-r--r--. 1 root root 501 Mar 5 20:37 alsa-restore.service
-rw-r--r--. 1 root root 558 Mar 5 20:37 alsa-state.service
-rw-r--r--. 1 root root 412 Mar 5 20:37 alsa-store.service
-rw-r--r--. 1 root root 645 Mar 26 11:43 anaconda-direct.service
-rw-r--r--. 1 root root 185 Mar 26 11:43 anaconda-nm-config.service
-rw-r--r--. 1 root root 660 Mar 26 11:43 anaconda-noshell.service
-rw-r--r--. 1 root root 387 Mar 26 11:43 anaconda.service
-rw-r--r--. 1 root root 684 Mar 26 11:43 anaconda-shell@.service
-rw-r--r--. 1 root root 322 Mar 26 11:43 anaconda-sshd.service
-rw-r--r--. 1 root root 312 Mar 26 11:43 anaconda.target
drwxr-xr-x. 2 root root 4096 Jun 4 15:33 anaconda.target.wants
-rw-r--r--. 1 root root 498 Mar 26 11:43 anaconda-tmux@.service
-rw-r--r--. 1 root root 275 Jun 10 2014 arp-ethers.service
```

```
-rw-r--r--. 1 root root 205 Oct 7 2014 atd.service
-rw-r-----. 1 root root 669 Mar 5 22:59 auditd.service
-rw-r--r--. 1 root root 663 Mar 6 05:17 auth-rpcgss-module.service
lrwxrwxrwx. 1 root root 14 Jun 4 09:52 autovt@.service -> getty@.service
-rw-r--r--. 1 root root 1044 Mar 5 23:03 avahi-daemon.service
-rw-r--r--. 1 root root 874 Mar 5 23:03 avahi-daemon.socket
-rw-r--r--. 1 root root 546 May 12 21:44 basic.target
drwxr-xr-x. 2 root root 4096 Jun 4 10:07 basic.target.wants
--More--
```

```
[root@centos7 ~]# ls -l /usr/lib/systemd/system | more
total 1208
-rw-r--r--. 1 root root 275 Mar 24 04:56 abrt-ccpp.service
-rw-r--r--. 1 root root 380 Mar 24 04:56 abrttd.service
-rw-r--r--. 1 root root 361 Mar 24 04:56 abrt-oops.service
-rw-r--r--. 1 root root 266 Mar 24 04:56 abrt-pstoreoops.service
-rw-r--r--. 1 root root 262 Mar 24 04:56 abrt-vmcore.service
-rw-r--r--. 1 root root 311 Mar 24 04:56 abrt-xorg.service
-rw-r--r--. 1 root root 421 Jun 10 2014 accounts-daemon.service
-rw-r--r--. 1 root root 501 Mar 5 20:37 alsa-restore.service
-rw-r--r--. 1 root root 558 Mar 5 20:37 alsa-state.service
-rw-r--r--. 1 root root 412 Mar 5 20:37 alsa-store.service
-rw-r--r--. 1 root root 645 Mar 26 11:43 anaconda-direct.service
-rw-r--r--. 1 root root 185 Mar 26 11:43 anaconda-nm-config.service
-rw-r--r--. 1 root root 660 Mar 26 11:43 anaconda-noshell.service
-rw-r--r--. 1 root root 387 Mar 26 11:43 anaconda.service
-rw-r--r--. 1 root root 684 Mar 26 11:43 anaconda-shell@.service
-rw-r--r--. 1 root root 322 Mar 26 11:43 anaconda-sshd.service
-rw-r--r--. 1 root root 312 Mar 26 11:43 anaconda.target
drwxr-xr-x. 2 root root 4096 Jun 4 15:33 anaconda.target.wants
-rw-r--r--. 1 root root 498 Mar 26 11:43 anaconda-tmux@.service
-rw-r--r--. 1 root root 275 Jun 10 2014 arp-ethers.service
-rw-r--r--. 1 root root 205 Oct 7 2014 atd.service
-rw-r-----. 1 root root 669 Mar 5 22:59 auditd.service
```

```
-rw-r--r--. 1 root root 663 Mar 6 05:17 auth-rpcgss-module.service
lrwxrwxrwx. 1 root root 14 Jun 4 09:52 autovt@.service -> getty@.service
-rw-r--r--. 1 root root 1044 Mar 5 23:03 avahi-daemon.service
-rw-r--r--. 1 root root 874 Mar 5 23:03 avahi-daemon.socket
-rw-r--r--. 1 root root 546 May 12 21:44 basic.target
drwxr-xr-x. 2 root root 4096 Jun 4 10:07 basic.target.wants
--More--
```

Par exemple, sous RHEL/CentOS 7, le service **sshd** est configuré par le fichier **/usr/lib/systemd/system/sshd.service** :

```
[root@centos7 ~]# cat /usr/lib/systemd/system/sshd.service
[Unit]
Description=OpenSSH server daemon
After=network.target sshd-keygen.service
Wants=sshd-keygen.service

[Service]
EnvironmentFile=/etc/sysconfig/ssh
ExecStart=/usr/sbin/sshd -D $OPTIONS
ExecReload=/bin/kill -HUP $MAINPID
KillMode=process
Restart=on-failure
RestartSec=42s

[Install]
WantedBy=multi-user.target
```

Dans le fichier on peut noter la présence des lignes suivantes :

- **ExecStart=/usr/sbin/sshd -D \$OPTIONS**,
 - Cette ligne définit l'exécutable à lancer,
- **After=network.target sshd-keygen.service**,
 - Cette ligne indique les services qui devraient être démarrés avant le démarrage de sshd,
- **WantedBy=multi-user.target**,

- Cette ligne indique la Cible dans laquelle le service doit être démarré,
- **Restart=on-failure**,
 - Cette ligne indique quand le service doit être re-démarré.

Système de Démarrage

Systemd utilise des Cibles d'une manière similaire à ce que **SysVinit** utilise des niveaux d'exécution. Pour rendre la transition plus facile, il existe des **Cibles** qui simulent les niveaux d'exécution de **SysVinit** :

- runlevel0.target,
- runlevel1.target,
- runlevel2.target,
- runlevel3.target,
- runlevel4.target,
- runlevel5.target,
- runlevel6.target.

Ceci étant dans RHEL/CentOS 7 il y principalement deux Cibles finales :

- **multi-user.target** qui est l'équivalent du niveau d'exécution 3,
- **graphical.target** qui est l'équivalent du niveau d'exécution 5.

Chaque Cible est décrite par un fichier de configuration :

```
[root@centos7 ~]# cat /usr/lib/systemd/system/graphical.target
# This file is part of systemd.
#
# systemd is free software; you can redistribute it and/or modify it
# under the terms of the GNU Lesser General Public License as published by
# the Free Software Foundation; either version 2.1 of the License, or
# (at your option) any later version.

[Unit]
Description=Graphical Interface
```

```
Documentation=man:systemd.special(7)
Requires=multi-user.target
After=multi-user.target
Conflicts=rescue.target
Wants=display-manager.service
AllowIsolate=yes

[Install]
Alias=default.target
```

Dans ce fichier on peut noter la présence des lignes suivantes :

- **Requires=multi-user.target,**
 - Cette ligne indique que le **graphical.target** ne peut pas être atteint si le **multi-user.target** n'a pas été atteint auparavant,
- **After=multi-user.target,**
 - Cette ligne indique le **multi-user.target** doit d'abord être lancé,
- **Conflicts=rescue.target,**
 - Cette ligne indique la Cible en conflit avec le **graphical.target**,
- **Wants=display-manager.service,**
 - Cette ligne indique quel service doit être démarré.

Dernièrement, sous RHEL/CentOS 7, la Cible par défaut peut être modifiée en éditant le lien symbolique **/etc/systemd/system/default.target** :

```
[root@centos7 ~]# ls -l /etc/systemd/system/default.target
lrwxrwxrwx. 1 root root 36 Mar  8 14:05 /etc/systemd/system/default.target ->
/lib/systemd/system/graphical.target
```

La Commande systemd-analyze

Pour avoir une évaluation du temps de démarrage, il convient d'utiliser la commande suivante :

```
[root@centos7 ~]# systemd-analyze
```

Startup finished in 769ms (kernel) + 4.643s (initrd) + 40.147s (userspace) = 45.560s

L'option **blame** de la commande `systemd-analyze` permet de voir le temps de démarrage de chaque Unité afin de pouvoir se concentrer sur les plus lentes :

```
[root@centos7 ~]# systemd-analyze blame
12.274s firewalld.service
10.302s tuned.service
 9.676s accounts-daemon.service
 8.875s gssproxy.service
 8.860s ModemManager.service
 8.598s vboxadd-x11.service
 7.829s kdump.service
 7.089s vboxadd.service
 6.398s plymouth-quit-wait.service
 5.593s NetworkManager-wait-online.service
 5.379s avahi-daemon.service
 5.104s abrt-ccpp.service
 5.065s postfix.service
 4.684s systemd-logind.service
 4.385s sysstat.service
 4.306s rtkit-daemon.service
 3.927s systemd-udev-settle.service
 3.396s ksmtuned.service
 3.084s rhel-dmesg.service
 2.811s libvirtd.service
 2.428s chronyd.service
 2.401s vboxadd-service.service
 2.349s nfs-config.service
 2.266s var-lib-nfs-rpc_pipefs.mount
 2.229s rhel-loadmodules.service
 2.104s rsyslog.service
 1.357s network.service
 1.283s lvm2-monitor.service
```

```
1.246s rpcbind.service
1.069s systemd-fsck-root.service
1.007s colord.service
944ms systemd-tmpfiles-setup-dev.service
872ms systemd-tmpfiles-clean.service
791ms rhel-readonly.service
780ms NetworkManager.service
743ms dmraid-activation.service
723ms gdm.service
720ms ksm.service
718ms polkit.service
716ms proc-fs-nfsd.mount
669ms auditd.service
660ms boot.mount
608ms systemd-udev-trigger.service
601ms kmod-static-nodes.service
565ms netcf-transaction.service
520ms systemd-vconsole-setup.service
497ms systemd-sysctl.service
487ms sys-kernel-debug.mount
302ms dev-disk-by\x2duuid-11a4d11d\x2d81e4\x2d46a7\x2d82e0\x2d7796cd597dc9.swap
297ms systemd-tmpfiles-setup.service
283ms dev-mqueue.mount
282ms dev-hugepages.mount
261ms rhel-import-state.service
243ms udisks2.service
239ms systemd-user-sessions.service
235ms rpc-statd-notify.service
217ms systemd-random-seed.service
173ms plymouth-read-write.service
161ms systemd-udevd.service
147ms upower.service
142ms systemd-fsck@dev-disk-by\x2duuid-e8d3bd48\x2d1386\x2d411c\x2d9675\x2d41c3f8f1a309.service
110ms plymouth-start.service
```

```
96ms sys-fs-fuse-connections.mount
82ms bluetooth.service
73ms iscsi-shutdown.service
69ms systemd-remount-fs.service
63ms systemd-hostnamed.service
53ms systemd-update-utmp.service
38ms systemd-journal-flush.service
33ms sys-kernel-config.mount
31ms systemd-update-utmp-runlevel.service
```

lines 43-71/71 (END)

L'option **critical-chain** permet de voir l'enchaînement des événements qui amènent au chargement de l'Unité passée en argument :

```
[root@centos7 ~]# systemd-analyze critical-chain sshd.service
```

The time after the unit is active or started is printed after the "@" character.

The time the unit takes to start is printed after the "+" character.

```
sshd.service @32.037s
└─network.target @31.990s
   └─network.service @30.621s +1.357s
      └─NetworkManager.service @24.242s +780ms
         └─firewalld.service @11.954s +12.274s
            └─basic.target @11.937s
               └─sockets.target @11.937s
                  └─dbus.socket @11.936s
                     └─sysinit.target @11.784s
                        └─systemd-update-utmp.service @11.726s +53ms
                           └─auditd.service @11.051s +669ms
                              └─systemd-tmpfiles-setup.service @10.734s +297ms
                                 └─rhel-import-state.service @10.470s +261ms
                                    └─local-fs.target @10.464s
                                       └─boot.mount @9.798s +660ms
                                          └─systemd-fsck@dev-disk-by\x2duuid-
e8d3bd48\x2d1386\x2d411c\x2d9675\x2d41c3f8f1a309.service @9.654s +142ms
```

```
└dev-disk-by\x2duuid-e8d3bd48\x2d1386\x2d411c\x2d9675\x2d41c3f8f1a309.device
```

```
@9.650s
```

Options de la Commande

Les options de la commande **systemd-analyze** sont :

```
[root@centos7 ~]# systemd-analyze --help
systemd-analyze [OPTIONS...] {COMMAND} ...
```

Process systemd profiling information

-h --help	Show this help
--version	Show package version
--system	Connect to system manager
--user	Connect to user service manager
--order	When generating a dependency graph, show only order
--require	When generating a dependency graph, show only requirement
--from-pattern=GLOB, --to-pattern=GLOB	When generating a dependency graph, filter only origins or destinations, respectively
--fuzz=TIMESPAN	When printing the tree of the critical chain, print also services, which finished TIMESPAN earlier, than the latest in the branch. The unit of TIMESPAN is seconds unless specified with a different unit, i.e. 50ms
--no-pager	Do not pipe output into a pager

Commands:

time	Print time spent in the kernel before reaching userspace
blame	Print list of running units ordered by time to init
critical-chain	Print a tree of the time critical chain of units
plot	Output SVG graphic showing service initialization
dot	Output dependency graph in dot(1) format

```
set-log-level LEVEL Set logging threshold for systemd
dump                Output state serialization of service manager
```

Gestion des Services

Pour obtenir le détail sur un service donné, il convient d'utiliser la commande **systemctl** :

```
[root@centos7 ~]# systemctl status sshd.service
sshd.service - OpenSSH server daemon
   Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled)
   Active: active (running) since Thu 2015-06-11 11:01:52 CEST; 2h 45min ago
 Main PID: 1212 (sshd)
    CGroup: /system.slice/sshd.service
            └─1212 /usr/sbin/sshd -D

Jun 11 11:01:52 centos7.fenestros.loc systemd[1]: Started OpenSSH server daemon.
Jun 11 11:01:53 centos7.fenestros.loc sshd[1212]: Server listening on 0.0.0.0 port 22.
Jun 11 11:01:53 centos7.fenestros.loc sshd[1212]: Server listening on :: port 22.
```

Pour arrêter une Unité de service, utilisez la commande suivante :

```
[root@centos7 ~]# systemctl stop sshd.service
[root@centos7 ~]# systemctl status sshd.service
sshd.service - OpenSSH server daemon
   Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled)
   Active: inactive (dead) since Thu 2015-06-11 13:58:59 CEST; 16s ago
  Process: 1212 ExecStart=/usr/sbin/sshd -D $OPTIONS (code=exited, status=0/SUCCESS)
 Main PID: 1212 (code=exited, status=0/SUCCESS)

Jun 11 11:01:52 centos7.fenestros.loc systemd[1]: Started OpenSSH server daemon.
Jun 11 11:01:53 centos7.fenestros.loc sshd[1212]: Server listening on 0.0.0.0 port 22.
Jun 11 11:01:53 centos7.fenestros.loc sshd[1212]: Server listening on :: port 22.
Jun 11 13:58:59 centos7.fenestros.loc systemd[1]: Stopping OpenSSH server daemon...
```

```
Jun 11 13:58:59 centos7.fenestros.loc sshd[1212]: Received signal 15; terminating.  
Jun 11 13:58:59 centos7.fenestros.loc systemd[1]: Stopped OpenSSH server daemon.
```

Pour démarrer un service, utilisez la commande suivante :

```
[root@centos7 ~]# systemctl start ssh.service  
[root@centos7 ~]# systemctl status sshd.service  
sshd.service - OpenSSH server daemon  
   Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled)  
   Active: active (running) since Thu 2015-06-11 14:00:17 CEST; 6s ago  
 Main PID: 6624 (sshd)  
    CGroup: /system.slice/sshd.service  
            └─6624 /usr/sbin/sshd -D  
  
Jun 11 14:00:17 centos7.fenestros.loc systemd[1]: Starting OpenSSH server daemon...  
Jun 11 14:00:17 centos7.fenestros.loc systemd[1]: Started OpenSSH server daemon.  
Jun 11 14:00:17 centos7.fenestros.loc sshd[6624]: Server listening on 0.0.0.0 port 22.  
Jun 11 14:00:17 centos7.fenestros.loc sshd[6624]: Server listening on :: port 22.
```

Pour désactiver un service au prochain démarrage du système, utilisez l'option **disable** :

```
[root@centos7 ~]# systemctl disable sshd.service  
rm '/etc/systemd/system/multi-user.target.wants/sshd.service'  
[root@centos7 ~]# systemctl status sshd.service  
sshd.service - OpenSSH server daemon  
   Loaded: loaded (/usr/lib/systemd/system/sshd.service; disabled)  
   Active: active (running) since Thu 2015-06-11 14:00:17 CEST; 1min 59s ago  
 Main PID: 6624 (sshd)  
    CGroup: /system.slice/sshd.service  
            └─6624 /usr/sbin/sshd -D  
  
Jun 11 14:00:17 centos7.fenestros.loc systemd[1]: Starting OpenSSH server daemon...  
Jun 11 14:00:17 centos7.fenestros.loc systemd[1]: Started OpenSSH server daemon.  
Jun 11 14:00:17 centos7.fenestros.loc sshd[6624]: Server listening on 0.0.0.0 port 22.
```

```
Jun 11 14:00:17 centos7.fenestros.loc sshd[6624]: Server listening on :: port 22.
```

Pour activer un service au prochain démarrage du système, utilisez l'option **enable** :

```
[root@centos7 ~]# systemctl enable sshd.service
ln -s '/usr/lib/systemd/system/sshd.service' '/etc/systemd/system/multi-user.target.wants/sshd.service'
[root@centos7 ~]# systemctl status sshd.service
sshd.service - OpenSSH server daemon
   Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled)
   Active: active (running) since Thu 2015-06-11 14:00:17 CEST; 3min 27s ago
 Main PID: 6624 (sshd)
    CGroup: /system.slice/sshd.service
            └─6624 /usr/sbin/sshd -D

Jun 11 14:00:17 centos7.fenestros.loc systemd[1]: Starting OpenSSH server daemon...
Jun 11 14:00:17 centos7.fenestros.loc systemd[1]: Started OpenSSH server daemon.
Jun 11 14:00:17 centos7.fenestros.loc sshd[6624]: Server listening on 0.0.0.0 port 22.
Jun 11 14:00:17 centos7.fenestros.loc sshd[6624]: Server listening on :: port 22.
```

Arrêt du Système

La commande shutdown

Lors de l'arrêt de la machine, Linux procède, entre autre, aux tâches suivantes :

- Il prévient les utilisateurs,
- Il arrête tous les services,
- Il inscrit toutes les données sur disque,
- Il démonte les systèmes de fichiers.

La commande utilisée pour arrêter le système est la commande **shutdown** :

```
shutdown [-t sec] [-akrhHPfnc] heure [message]
```

Options de la commande

Les options de cette commande sont :

```
Usage:      shutdown [-akrhHPfnc] [-t secs] time [warning message]
            -a:      use /etc/shutdown.allow
            -k:      don't really shutdown, only warn.
            -r:      reboot after shutdown.
            -h:      halt after shutdown.
            -P:      halt action is to turn off power.
            -H:      halt action is to just halt.
            -f:      do a 'fast' reboot (skip fsck).
            -F:      Force fsck on reboot.
            -n:      do not go through "init" but go down real fast.
            -c:      cancel a running shutdown.
            -t secs: delay between warning and kill signal.
            ** the "time" argument is mandatory! (try "now") **
```

Parmi les options les plus importantes, on note :

Option	Description
-h	Arrêter le système
-r	Re-démarrer le système
-c	Annuler l'opération shutdown en cours
-f	Re-démarrer rapidement sans vérifier les systèmes de fichiers
-F	Forcer la vérification des systèmes de fichiers lors du prochain démarrage

L'option **heure** peut prendre plusieurs valeurs :

Valeur	Description
hh:mm	L'heure à laquelle l'opération aura lieu
+m	Nombre de minutes avant que l'opération aura lieu
now	L'opération est immédiate



Important : Si l'opération est programmée pour dans moins de 5 minutes, les connexions supplémentaires sont interdites, y comprises les tentatives de connexion de root. Notez aussi que l'utilisation de la commande **shutdown** peut être accordée à d'autres utilisateurs de root en utilisant le fichier **/etc/shutdown.allow**.

Dans votre VM, ouvrez deux terminaux. Dans le premier passez en tant que root. Planifiez ensuite un redémarrage de la VM dans 30 minutes :

```
[root@centos7 ~]# shutdown -r +30

Broadcast message from trainee@centos.fenestros.loc
(/dev/pts/0) at 10:12 ...

The system is going down for reboot in 30 minutes!
```

Dans le deuxième terminal est passez en tant que route. Annulez ensuite le shutdown :

```
[trainee@centos7 ~]$ su -
Mot de passe :
[root@centos7 ~]# shutdown -c
```

Retournez au premier terminal et constatez le message affiché :

```
[root@centos7 ~]# shutdown -r +30

Broadcast message from trainee@centos.fenestros.loc
(/dev/pts/0) at 10:12 ...
```

```
The system is going down for reboot in 30 minutes!
```

```
shutdown: Shutdown cancelled  
[root@centos7 ~]#
```

La commande reboot

Cette commande redémarre le système. Quand le système fonctionne normalement, l'exécution de reboot appelle la commande **shutdown -r**.

Options de la commande

Les options de cette commande sont :

```
[root@centos7 ~]# reboot --help  
Usage: reboot [OPTION]...  
Reboot the system.  
  
Options:  
-n, --no-sync          don't sync before reboot or halt  
-f, --force            force reboot or halt, don't call shutdown(8)  
-p, --poweroff         switch off the power when called as halt  
-w, --wtmp-only        don't actually reboot or halt, just write wtmp  
                        record  
-q, --quiet            reduce output to errors only  
-v, --verbose          increase output to include informational messages  
--help                display this help and exit  
--version              output version information and exit
```

This command is intended to instruct the kernel to reboot or halt the system; when run without the -f option, or when in a system runlevel other than 0 or 6,

it will actually execute `/sbin/shutdown`.

Report bugs to [<upstart-devel@lists.ubuntu.com>](mailto:upstart-devel@lists.ubuntu.com)

La commande halt

Cette commande arrête le système. Quand le système fonctionne normalement, l'exécution de `halt` appelle la commande **`shutdown -h`**.

Options de la commande

Les options de cette commande sont :

```
[root@centos7 ~]# halt --help
Usage: halt [OPTION]...
Halt the system.
```

Options:

<code>-n, --no-sync</code>	don't sync before reboot or halt
<code>-f, --force</code>	force reboot or halt, don't call <code>shutdown(8)</code>
<code>-p, --poweroff</code>	switch off the power when called as <code>halt</code>
<code>-w, --wtmp-only</code>	don't actually reboot or halt, just write <code>wtmp</code> record
<code>-q, --quiet</code>	reduce output to errors only
<code>-v, --verbose</code>	increase output to include informational messages
<code>--help</code>	display this help and exit
<code>--version</code>	output version information and exit

This command is intended to instruct the kernel to reboot or halt the system; when run without the `-f` option, or when in a system runlevel other than 0 or 6, it will actually execute `/sbin/shutdown`.

Report bugs to <upstart-devel@lists.ubuntu.com>

La commande **poweroff**

Cette commande arrête le système et coupe l'alimentation électrique. Elle est l'équivalente de la commande **halt -p**. Quand le système fonctionne normalement, l'exécution de **poweroff** appelle la commande **shutdown -hP**.

Options de la commande

Les options de cette commande sont :

```
[root@centos7 ~]# poweroff --help
```

```
Usage: poweroff [OPTION]...
```

```
Power off the system.
```

Options:

-n, --no-sync	don't sync before reboot or halt
-f, --force	force reboot or halt, don't call shutdown(8)
-p, --poweroff	switch off the power when called as halt
-w, --wtmp-only	don't actually reboot or halt, just write wtmp record
-q, --quiet	reduce output to errors only
-v, --verbose	increase output to include informational messages
--help	display this help and exit
--version	output version information and exit

This command is intended to instruct the kernel to reboot or halt the system; when run without the -f option, or when in a system runlevel other than 0 or 6, it will actually execute /sbin/shutdown.

Report bugs to <upstart-devel@lists.ubuntu.com>

<html>

Copyright © 2004-2017 Hugh Norris.

Ce(tte) oeuvre est mise à disposition selon les termes de la Licence Creative Commons Attribution - Pas d'Utilisation Commerciale - Pas de Modification 3.0 France.

</html>
