

Version : **2024.01**

Dernière mise-à-jour : 2024/04/05 05:54

Topic 1 : System Architecture

Contenu du Module

- **Topic 1: System Architecture**

- Contenu du Module
- Présentation
- Processus de Démarrage du Noyau Linux
- Processus Init
- Démarrer RHEL/CentOS 5 avec SysVinit
 - Niveaux d'exécution sous RHEL/CentOS 5
 - Inittab
 - Scripts de Démarrage
 - rc.sysinit sous RHEL/CentOS 5
 - Répertoire init.d
 - Linux Standard Base
 - Codes Retour Standardisés
 - Scripts
 - Répertoires rcx.d
 - rc.local
 - La Commande chkconfig
- Démarrer RHEL/CentOS 6 avec Upstart
 - Inittab
 - Initialisation du Système
 - Runlevels
 - [CTL]-[ALT]-[DEL]
 - mingetty

- Gestion des Services
 - La Commande initctl
- Jobs
- Événements
- Etats
- Démarrer et Arrêter les Jobs
 - La Commande status
 - La Commande start
 - La Commande stop
 - La Commande restart
- Démarrer RHEL/CentOS 7
 - La Commande systemctl
 - Fichiers de Configuration
 - La Commande systemd-analyze
 - Gestion des Services
- Fichiers Spéciaux
- Commandes
 - La Commande lspci
 - La Commande lsusb
 - La Commande dmidecode
- Répertoire /proc
 - Répertoires
 - ide/scsi
 - acpi
 - bus
 - net
 - sys
 - La Commande sysctl
 - Fichiers
 - Processeur
 - Interruptions système
 - Canaux DMA
 - Plages d'entrée/sortie
 - Périphériques

- Modules
- Statistiques de l'utilisation des disques
- Partitions
- Espaces de pagination
- Statistiques d'utilisation du processeur
- Statistiques d'utilisation de la mémoire
- Version du noyau
- Interprétation des informations dans /proc
 - Commandes
 - free
 - uptime ou w
 - iostat
 - vmstat
 - mpstat
 - sar
 - Utilisation des commandes en production
 - Identifier un système limité par le processeur
 - Identifier un système ayant un problème de mémoire
 - Identifier un système ayant un problème d'E/S
- Modules usb
- udev
 - La Commande udevadm
- Système de fichiers /sys
- Limiter les Ressources
 - ulimit
 - Groupes de Contrôle
 - LAB #1 - Travailler avec les cgroups sous RHEL/CentOS 7

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

Charger de Démarrage (*Bootloader*) à exécuter à partir d'un support bootable,

- Le Charger 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.

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
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 quelque chose 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

Ligne	Description
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/abrtcd          init.d/cpuspeed      init.d/iptables      init.d/netfs          init.d/portreserve
init.d/sandbox         init.d/vboxadd        init.d/crond          init.d/network        init.d/postfix
init.d/acpid           init.d/crond          init.d/irqbalance     init.d/NetworkManager init.d/psacct
init.d/atd              init.d/cups           init.d/jexec           init.d/nfs             init.d/rdisc
init.d/single          init.d/dnsmasq        init.d/kdump           init.d/nfslock         init.d/restorecond
init.d/auditd          init.d/wpa_supplicant init.d/killall         init.d/nscd            init.d/rpcbind
init.d/smartd          init.d/firstboot      init.d/lvm2-monitor   init.d/nslcd           init.d/rpcgssd
init.d/autofs           init.d/ypbind          init.d/mdmonitor       init.d/ntpd            init.d/rpcidmapd
init.d/snmpd           init.d/functions      init.d/microcode_ctl   init.d/ntpddate        init.d/rpcsvcgssd
init.d/avahi-daemon    init.d/haldaemon      init.d/messagebus      init.d/oddjobd         init.d/rsyslog
init.d/snmpttrapd      init.d/halt           init.d/netconsole      init.d/udev-post
init.d/bluetooth       init.d/ip6tables      init.d/netconsole      init.d/udev-post
init.d/ssh             init.d/ip6tables      init.d/netconsole      init.d/udev-post
init.d/certmonger      init.d/ip6tables      init.d/netconsole      init.d/udev-post
init.d/sss             init.d/ip6tables      init.d/netconsole      init.d/udev-post
init.d/cgconfig        init.d/ip6tables      init.d/netconsole      init.d/udev-post
init.d/sysstat         init.d/ip6tables      init.d/netconsole      init.d/udev-post
init.d/cgred           init.d/ip6tables      init.d/netconsole      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=$1~/s$|^[0-9.]*$/?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
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/ssh ] && . /etc/sysconfig/ssh
--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.
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/S82abrt			
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/S82abrtd			
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/K01smartrd 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/K01smartrd rc4.d/K15httpd rc4.d/K50snmpd rc4.d/K70vboxadd-x11 rc4.d/K75ntpdate
```

```
rc4.d/K86cgred          rc4.d/K95cgconfig
rc4.d/K02oddjobd        rc4.d/K16abrttd      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_suplicant
rc5.d/K89rdisc
rc5.d/K10psacct          rc5.d/K50netconsole  rc5.d/K69rpcsvcgssd   rc5.d/K75ntpddate    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
cgred	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
cron	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
nsld	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

ntpd	0:arrêt	1:arrêt	2:arrêt	3:arrêt	4:arrêt	5:arrêt	6:arrêt
oddjobd	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 tiers 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@centos ~]# 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	loaded	active	plugged	PRO/1000 MT Desktop
Adapter				
- .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	loaded	active	mounted	NFSD configuration
filesystem				
run-media-trainee-VBOXADDITIONS_4.3.28_100309.mount	loaded	active	mounted	
/run/media/trainee/VBOXADDITIONS_4.3.28_100309				

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 Reporting Tool	loaded	active	running	ABRT Automated Bug
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 Generate Color Profiles	loaded	active	running	Manage, Install and
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 firewall daemon	loaded	active	running	firewalld - dynamic
gdm.service	loaded	active	running	GNOME Display Manager
gssproxy.service	loaded	active	running	GSSAPI Proxy Daemon

iscsi-shutdown.service sessions on shutdown	loaded active exited	Logout off all iSCSI
kdump.service arming	loaded failed failed	Crash recovery kernel
kmod-static-nodes.service static device nodes for the current ker	loaded active exited	Create list of required
ksm.service	loaded active exited	Kernel Samepage Merging
ksmtuned.service (KSM) Tuning Daemon	loaded active running	Kernel Samepage Merging
libstoragemgmt.service server daemon	loaded active running	libstoragemgmt plug-in
libvirtd.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 Arbitrary Executable File Formats File System	loaded inactive dead
systemd-ask-password-console.path Dispatch Password Requests to Cons ole Directory Watch	loaded inactive dead
abrt-vmcore.service Harvest vmcores for ABRT	loaded inactive dead
alsa-restore.service Restore Sound Card State	loaded inactive dead
alsa-store.service Store Sound Card State	loaded inactive dead
apparmor.service apparmor.service	not-found inactive dead
auth-rpcgss-module.service Kernel Module supporting RPCSEC_GS S	loaded inactive dead

brandbot.service Flexible Branding Service	loaded inactive dead
cpupower.service Configure CPU power related settings	loaded inactive dead
dm-event.service Device-mapper event daemon	loaded inactive dead
dmraid-activation.service Activation of DM RAID sets	loaded inactive dead
dracut-shutdown.service Restore /run/initramfs	loaded inactive dead
ebtables.service Ethernet Bridge Filtering tables	loaded inactive dead
emergency.service Emergency Shell	loaded inactive dead
exim.service exim.service	not-found inactive dead
getty@tty1.service Getty on tty1	loaded inactive dead
hypervkvpd.service Hyper-V KVP daemon	loaded inactive dead
hypervvssd.service Hyper-V VSS daemon	loaded inactive dead
ip6tables.service IPv6 firewall with ip6tables	loaded inactive dead
iptables.service IPv4 firewall with iptables	loaded inactive dead
irqbalance.service irqbalance daemon	loaded inactive dead
iscsi.service Login and scanning of iSCSI devices	loaded inactive dead
iscsid.service Open-iSCSI	loaded inactive dead

iscsiuio.service iSCSI UserSpace I/O driver --More--	loaded inactive dead
--	----------------------

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 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--
```

```
[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 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--
```

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.
```

Fichiers Spéciaux

Dans l'ordinateur les périphériques sont reliés à un **contrôleur** qui communique avec le processeur à l'aide d'un **bus**. Le contrôleur ainsi que les périphériques nécessitent des pilotes. Sous Linux, les pilotes sont généralement fournis sous la forme d'un **module**. Chaque périphérique est représenté par un **fichier spécial** dans le répertoire **/dev** et c'est dans ce fichier que le système trouve les informations nécessaires pour s'adresser au pilote.



Important : Les périphériques qui nécessitent à ce que l'ordinateur soit éteint afin des les brancher/débrancher sont appelés communément **Cold Plug Devices**. Les périphériques qui peuvent être brancher/débrancher à chaud sont appelés des **Hot Plug Devices**.

Consultez le contenu du répertoire /dev :

```
[root@centos7 ~]# ls -l /dev | more
total 0
crw-----. 1 root    root      10, 235 Oct 28 09:40 autofs
drwxr-xr-x. 2 root    root        140 Oct 28 09:40 block
drwxr-xr-x. 2 root    root         80 Oct 28 09:40 bsg
crw-----. 1 root    root      10, 234 Oct 28 09:40 btrfs-control
drwxr-xr-x. 3 root    root         60 Oct 28 09:40 bus
lrwxrwxrwx. 1 root    root          3 Oct 28 09:40 cdrom -> sr0
```

```

drwxr-xr-x. 2 root    root      2800 Oct 28 09:41 char
crw-----. 1 root    root        5,   1 Oct 28 09:40 console
lrwxrwxrwx. 1 root    root        11 Oct 28 09:40 core -> /proc/kcore
drwxr-xr-x. 3 root    root        80 Oct 28 09:40 cpu
crw-----. 1 root    root       10,  61 Oct 28 09:40 cpu_dma_latency
crw-----. 1 root    root       10,  62 Oct 28 09:40 crash
drwxr-xr-x. 5 root    root       100 Oct 28 09:40 disk
drwxr-xr-x. 2 root    root        60 Oct 28 09:40 dri
lrwxrwxrwx. 1 root    root        13 Oct 28 09:40 fd -> /proc/self/fd
crw-rw-rw-. 1 root    root        1,   7 Oct 28 09:40 full
crw-rw-rw-. 1 root    root     10, 229 Oct 28 09:41 fuse
crw-----. 1 root    root     10, 228 Oct 28 09:40 hpet
drwxr-xr-x. 3 root    root         0 Oct 28 09:40 hugepages
prw-----. 1 root    root         0 Oct 28 09:40 initctl
drwxr-xr-x. 3 root    root       240 Oct 28 09:40 input
crw-r--r--. 1 root    root        1,  11 Oct 28 09:40 kmsg
--More--

```

On peut noter dans la sortie de la commande que certains fichiers sont de type **bloc (b)**, tandis que d'autre sont de type **caractère (c)**.

```

...
brw-rw---- 1 root    disk        8,   1 Nov  1 08:39 sda1
...
crw-rw-rw- 1 root    tty         5,   0 Nov  1 08:39 tty
...

```

La différence entre les deux repose sur le type de communication entre le système et le module. Dans le premier cas le système accède au périphérique par des coordonnées du bloc de données sur le support tandis que dans le deuxième cas la communication d'échange de données se fait octet par octet sans utiliser des tampons.

Les deux informations clefs du fichier spécial sont situées à la place de la taille d'un fichier normal et se nomment le **majeur** et le **mineur** :

- le **majeur** identifie le pilote du périphérique et donc son contrôleur,
- le **mineur** identifie le périphérique ou une particularité du périphérique telle une partition d'un disque.

Commandes

La Commande lspci

Cette commande vous renseigne sur les adaptateurs reliés aux bus PCI, AGP et PCI express :

```
[root@centos7 ~]# lspci
00:00.0 Host bridge: Intel Corporation 440FX - 82441FX PMC [Natoma] (rev 02)
00:01.0 ISA bridge: Intel Corporation 82371SB PIIX3 ISA [Natoma/Triton II]
00:01.1 IDE interface: Intel Corporation 82371AB/EB/MB PIIX4 IDE (rev 01)
00:02.0 VGA compatible controller: InnoTek Systemberatung GmbH VirtualBox Graphics Adapter
00:03.0 Ethernet controller: Intel Corporation 82540EM Gigabit Ethernet Controller (rev 02)
00:04.0 System peripheral: InnoTek Systemberatung GmbH VirtualBox Guest Service
00:05.0 Multimedia audio controller: Intel Corporation 82801AA AC'97 Audio Controller (rev 01)
00:06.0 USB controller: Apple Inc. KeyLargo/Intrepid USB
00:07.0 Bridge: Intel Corporation 82371AB/EB/MB PIIX4 ACPI (rev 08)
00:0b.0 USB controller: Intel Corporation 82801FB/FBM/FR/FW/FRW (ICH6 Family) USB2 EHCI Controller
00:0d.0 SATA controller: Intel Corporation 82801HM/HEM (ICH8M/ICH8M-E) SATA Controller [AHCI mode] (rev 02)
```

Pour obtenir de l'information sur un adaptateur spécifique, il convient d'utiliser la même commande avec l'option **-v** en spécifiant l'identifiant concerné :

```
[root@centos7 ~]# lspci -v -s 00:03.0
00:03.0 Ethernet controller: Intel Corporation 82540EM Gigabit Ethernet Controller (rev 02)
    Subsystem: Intel Corporation PR0/1000 MT Desktop Adapter
    Flags: bus master, 66MHz, medium devsel, latency 64, IRQ 19
    Memory at f0000000 (32-bit, non-prefetchable) [size=128K]
    I/O ports at d010 [size=8]
    Capabilities: [dc] Power Management version 2
    Capabilities: [e4] PCI-X non-bridge device
    Kernel driver in use: e1000
```

ou :

```
[root@centos7 ~]# lspci -vv -s 00:03.0
00:03.0 Ethernet controller: Intel Corporation 82540EM Gigabit Ethernet Controller (rev 02)
  Subsystem: Intel Corporation PR0/1000 MT Desktop Adapter
  Control: I/O+ Mem+ BusMaster+ SpecCycle- MemWINV- VGASnoop- ParErr- Stepping- SERR- FastB2B- DisINTx-
  Status: Cap+ 66MHz+ UDF- FastB2B- ParErr- DEVSEL=medium >TAbort- <TAbort- <MAbort- >SERR- <PERR- INTx-
  Latency: 64 (63750ns min)
  Interrupt: pin A routed to IRQ 19
  Region 0: Memory at f0000000 (32-bit, non-prefetchable) [size=128K]
  Region 2: I/O ports at d010 [size=8]
  Capabilities: [dc] Power Management version 2
    Flags: PMEClk- DSI+ D1- D2- AuxCurrent=0mA PME(D0-,D1-,D2-,D3hot-,D3cold-)
    Status: D0 NoSoftRst- PME-Enable- DSel=0 DScale=0 PME-
  Capabilities: [e4] PCI-X non-bridge device
    Command: DPERE- ER0+ RBC=512 OST=1
    Status: Dev=ff:1f.0 64bit- 133MHz- SCD- USC- DC=simple DMMRBC=2048 DMOST=1 DMCRS=8 RSCEM- 266MHz- 533MHz-
  Kernel driver in use: e1000
```

Options de la commande

Les options de cette commande sont :

```
[root@centos7 ~]# lspci --help
lspci: invalid option -- '-'
Usage: lspci [<switches>]

Basic display modes:
-m      Produce machine-readable output (single -m for an obsolete format)
-t      Show bus tree

Display options:
-v      Be verbose (-vv for very verbose)
```

```
-k      Show kernel drivers handling each device
-x      Show hex-dump of the standard part of the config space
-xxx     Show hex-dump of the whole config space (dangerous; root only)
-xxxx    Show hex-dump of the 4096-byte extended config space (root only)
-b      Bus-centric view (addresses and IRQ's as seen by the bus)
-D      Always show domain numbers
```

Resolving of device ID's to names:

```
-n      Show numeric ID's
-nn     Show both textual and numeric ID's (names & numbers)
-q      Query the PCI ID database for unknown ID's via DNS
-qq     As above, but re-query locally cached entries
-Q      Query the PCI ID database for all ID's via DNS
```

Selection of devices:

```
-s [[:<domain>]:]<bus>:][<slot>][.[:<func>]]  Show only devices in selected slots
-d [<vendor>]:<device>                      Show only devices with specified ID's
```

Other options:

```
-i <file> Use specified ID database instead of /usr/share/hwdata/pci.ids
-p <file> Look up kernel modules in a given file instead of default modules.pcimap
-M       Enable 'bus mapping' mode (dangerous; root only)
```

PCI access options:

```
-A <method> Use the specified PCI access method (see '-A help' for a list)
-O <par>=<val> Set PCI access parameter (see '-O help' for a list)
-G       Enable PCI access debugging
-H <mode> Use direct hardware access (<mode> = 1 or 2)
-F <file> Read PCI configuration dump from a given file
```

La Commande lsusb

Cette commande vous renseigne sur les adaptateurs reliés au bus usb :

```
[root@centos7 ~]# lsusb
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 002 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub

[root@centos7 ~]# lsusb -vt
/: Bus 02.Port 1: Dev 1, Class=root_hub, Driver=ohci-pci/8p, 12M
/: Bus 01.Port 1: Dev 1, Class=root_hub, Driver=ehci-pci/8p, 480M
```

Options de la commande

Les options de cette commande sont :

```
[root@centos7 ~]# lsusb --help
Usage: lsusb [options]...
List USB devices
  -v, --verbose
        Increase verbosity (show descriptors)
  -s [[bus]:][devnum]
        Show only devices with specified device and/or
        bus numbers (in decimal)
  -d vendor:[product]
        Show only devices with the specified vendor and
        product ID numbers (in hexadecimal)
  -D device
        Selects which device lsusb will examine
  -t, --tree
        Dump the physical USB device hierarchy as a tree
  -V, --version
        Show version of program
  -h, --help
        Show usage and help
```

La Commande dmidecode

La commande **dmidecode** lit la table **DMI** (*Desktop Management Interface*) aussi appelée **SMBIOS** (*System Management BIOS*) et fourni les informations sur :

- l'état du matériel actuel,
- les extensions possibles.

```
[root@centos7 ~]# dmidecode
# dmidecode 2.12
SMBIOS 2.5 present.
10 structures occupying 450 bytes.
Table at 0x000E1000.

Handle 0x0000, DMI type 0, 20 bytes
BIOS Information
    Vendor: innotek GmbH
    Version: VirtualBox
    Release Date: 12/01/2006
    Address: 0xE0000
    Runtime Size: 128 kB
    ROM Size: 128 kB
    Characteristics:
        ISA is supported
        PCI is supported
        Boot from CD is supported
        Selectable boot is supported
        8042 keyboard services are supported (int 9h)
        CGA/mono video services are supported (int 10h)
        ACPI is supported

Handle 0x0001, DMI type 1, 27 bytes
System Information
    Manufacturer: innotek GmbH
```

Product Name: VirtualBox
Version: 1.2
Serial Number: 0
UUID: DDEE345C-66F8-4250-B5A2-85CEF2C1C54D
Wake-up Type: Power Switch
SKU Number: Not Specified
Family: Virtual Machine

Handle 0x0008, DMI type 2, 15 bytes

Base Board Information

Manufacturer: Oracle Corporation
Product Name: VirtualBox
Version: 1.2
Serial Number: 0
Asset Tag: Not Specified
Features:
 Board is a hosting board
Location In Chassis: Not Specified
Chassis Handle: 0x0003
Type: Motherboard
Contained Object Handles: 0

Handle 0x0003, DMI type 3, 13 bytes

Chassis Information

Manufacturer: Oracle Corporation
Type: Other
Lock: Not Present
Version: Not Specified
Serial Number: Not Specified
Asset Tag: Not Specified
Boot-up State: Safe
Power Supply State: Safe
Thermal State: Safe
Security Status: None

```
Handle 0x0007, DMI type 126, 42 bytes
Inactive

Handle 0x0005, DMI type 126, 15 bytes
Inactive

Handle 0x0006, DMI type 126, 28 bytes
Inactive

Handle 0x0002, DMI type 11, 7 bytes
OEM Strings
  String 1: vboxVer_4.3.28
  String 2: vboxRev_100309

Handle 0x0008, DMI type 128, 8 bytes
OEM-specific Type
  Header and Data:
    80 08 08 00 01 15 21 00

Handle 0xFEFF, DMI type 127, 4 bytes
End Of Table
```

Options de la commande

Les options de cette commande sont :

```
[root@centos7 ~]# dmidecode --help
Usage: dmidecode [OPTIONS]
Options are:
  -d, --dev-mem FILE      Read memory from device FILE (default: /dev/mem)
  -h, --help              Display this help text and exit
  -q, --quiet             Less verbose output
  -s, --string KEYWORD    Only display the value of the given DMI string
```

```

-t, --type TYPE      Only display the entries of given type
-u, --dump           Do not decode the entries
    --dump-bin FILE  Dump the DMI data to a binary file
    --from-dump FILE Read the DMI data from a binary file
-V, --version        Display the version and exit

```

Répertoire /proc

Le répertoire /proc contient des fichiers et des répertoires virtuels. Le contenu de ces fichiers est créé dynamiquement lors de la consultation. Seul root peut consulter la totalité des informations dans le répertoire /proc.

```

[root@centos7 ~]# ls /proc
1      217    260    2833   3085   465    5898      fb      sched_debug
10     22619  2605   2879   3091   478    5933      filesystems  scsi
11     22650  2606   2883   3104   482    5940      fs        self
12     228    261    2884   3108   484    60        interrupts  slabinfo
1215   229    262    29     3124   485    604       iomem     softirqs
1217   22908  263    2901   3172   488    698       ioports   stat
1227   230    264    2906   3185   490    7         irq       swaps
1228   232    265    2910   3304   492    7110      kallsyms  sys
1230   233    2671   2913   3496   494    8         kcore     sysrq-trigger
1264   23387  2676   2924   3543   495    83        keys      sysvipc
13     235    27     2926   363    5      9         key-users timer_list
1372   23580  2749   2931   370    502    acpi      kmsg      timer_stats
14     23916  2752   2937   372    503    asound    kpagecount tty
15     23998  2759   2948   38     508    buddyinfo kpageflags uptime
16     24     2762   2955   381    509    bus       loadavg   version
17     24013  2764   2967   40     519    cgroups   locks     vmallocinfo
1732   2427   2765   2971   41     521    cmdline   mdstat    vmstat
1744   248    2767   2973   414    525    consoles  meminfo   zoneinfo
17788  2486   2768   2983   431    530    cpuinfo   misc
18     25     2782   2985   432    536    crypto    modules

```

19	2586	2798	2988	433	549	devices	mounts
2	259	28	2998	434	586	diskstats	mtrr
20	2595	2802	3	436	587	dma	net
20638	2597	2806	3058	454	5894	driver	pagetypeinfo
21	26	2818	3084	458	5897	execdomains	partitions

Répertoires

ide/scsi

Ce répertoire contient des répertoires dans lesquels se trouvent des informations sur la capacité, le type et la géométrie des disques.

acpi

Ce répertoire contient des informations sur la gestion de l'énergie, les températures, les vitesses de ventilateurs, la charge des batteries.

bus

Ce répertoire contient un sous-répertoire par bus.

net

Ce répertoire contient des informations sur le réseau.

sys

Ce répertoire contient des paramètres du noyau. Certains des fichiers dans ce répertoire sont accessibles en écriture par root en temps réel. Par

exemple pour éviter des attaques réseau **DoS** utilisant la commande **ping**, saisissez la commande suivante :

```
# echo 1 > /proc/sys/net/ipv4/icmp_echo_ignore_all [Entrée]
```

Cette commande a pour résultat d'ignorer les requêtes ping.

La commande sysctl

Les fichiers dans le répertoire **/proc/sys** peuvent être administrés par la commande **sysctl** en temps réel.

La commande **sysctl** applique les règles consignés dans le fichier **/etc/sysctl.conf** au démarrage de la machine.

Saisissez la commande :

```
[root@centos7 ~]# cat /etc/sysctl.conf
# System default settings live in /usr/lib/sysctl.d/00-system.conf.
# To override those settings, enter new settings here, or in an /etc/sysctl.d/<name>.conf file
#
# For more information, see sysctl.conf(5) and sysctl.d(5).

[root@centos7 ~]# cat /usr/lib/sysctl.d/00-system.conf
# Kernel sysctl configuration file
#
# For binary values, 0 is disabled, 1 is enabled.  See sysctl(8) and
# sysctl.conf(5) for more details.

# Disable netfilter on bridges.
net.bridge.bridge-nf-call-ip6tables = 0
net.bridge.bridge-nf-call-iptables = 0
net.bridge.bridge-nf-call-arptables = 0

# Controls the maximum shared segment size, in bytes
kernel.shmmax = 4294967295
```

```
# Controls the maximum number of shared memory segments, in pages
```

```
[root@centos7 ~]# ls -l /etc/sysctl.d/
```

```
total 0
```

```
lrwxrwxrwx. 1 root root 14 Jun  4 09:54 99-sysctl.conf -> ../sysctl.conf
```

```
[root@centos7 ~]# cat /etc/sysctl.d/99-sysctl.conf
```

```
# System default settings live in /usr/lib/sysctl.d/00-system.conf.
```

```
# To override those settings, enter new settings here, or in an /etc/sysctl.d/<name>.conf file
```

```
#
```

```
# For more information, see sysctl.conf(5) and sysctl.d(5).
```

Options de la commande

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

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

Usage:

```
sysctl [options] [variable[=value] ...]
```

Options:

-a, --all	display all variables
-A	alias of -a
-X	alias of -a
--deprecated	include deprecated parameters to listing
-b, --binary	print value without new line
-e, --ignore	ignore unknown variables errors
-N, --names	print variable names without values
-n, --values	print only values of a variables
-p, --load[=<file>]	read values from file
-f	alias of -p
--system	read values from all system directories

```
-r, --pattern <expression>
                        select setting that match expression
-q, --quiet            do not echo variable set
-w, --write            enable writing a value to variable
-o                    does nothing
-x                    does nothing
-d                    alias of -h

-h, --help            display this help and exit
-V, --version          output version information and exit
```

For more details see `sysctl(8)`.



Important : Consultez la page de la traduction du manuel de **sysctl** [ici](#) pour comprendre la commande.

Fichiers

Processeur

```
[root@centos7 ~]# cat /proc/cpuinfo
processor      : 0
vendor_id     : GenuineIntel
cpu family    : 6
model         : 55
model name    : Intel(R) Celeron(R) CPU  N2840  @ 2.16GHz
stepping      : 8
microcode     : 0x19
cpu MHz       : 2167.721
cache size    : 6144 KB
```

```
physical id : 0
siblings   : 1
core id    : 0
cpu cores  : 1
apicid     : 0
initial apicid : 0
fpu        : yes
fpu_exception : yes
cpuid level : 5
wp         : yes
flags      : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush mmx fxsr sse sse2
syscall nx rdtscp lm constant_tsc rep_good nopl pni monitor ssse3 lahf_lm
bogomips   : 4335.44
clflush size : 64
cache_alignment : 64
address sizes : 36 bits physical, 48 bits virtual
power management:
```

Interruptions système

```
[root@centos7 ~]# cat /proc/interrupts
      CPU0
 0:   189   IO-APIC-edge   timer
 1:  5890   IO-APIC-edge   i8042
 8:    0   IO-APIC-edge   rtc0
 9:    1   IO-APIC-fasteoi  acpi
12: 22452   IO-APIC-edge   i8042
14:    0   IO-APIC-edge   ata_piix
15: 13891   IO-APIC-edge   ata_piix
19: 25816   IO-APIC-fasteoi ehci_hcd:usb1, enp0s3
20: 119352   IO-APIC-fasteoi vboxguest
21: 33754   IO-APIC-fasteoi ahci, snd_intel8x0
22:    0   IO-APIC-fasteoi ohci_hcd:usb2
```

NMI:	0	Non-maskable interrupts
LOC:	1630315	Local timer interrupts
SPU:	0	Spurious interrupts
PMI:	0	Performance monitoring interrupts
IWI:	83206	IRQ work interrupts
RTR:	0	APIC ICR read retries
RES:	0	Rescheduling interrupts
CAL:	0	Function call interrupts
TLB:	0	TLB shutdowns
TRM:	0	Thermal event interrupts
THR:	0	Threshold APIC interrupts
MCE:	0	Machine check exceptions
MCP:	48	Machine check polls
ERR:	0	
MIS:	0	



Important : Un pilote de périphérique demande au processeur de fournir un service en utilisant un IRQ. Quand la demande est faite, le processeur interrompt ses activités et passe le contrôle au pilote identifié par l'IRQ. Techniquement l'attribution d'un IRQ à un périphérique doit être exclusive. Dans le cas où deux périphériques demandent un service en même temps, c'est le périphérique ayant l'IRQ le plus bas qui est prioritaire.

Canaux DMA

```
[root@centos7 ~]# cat /proc/dma
4: cascade
```

Plages d'entrée/sortie

```
[root@centos7 ~]# cat /proc/ioproports | more
0000-001f : dma1
0020-0021 : pic1
0040-0043 : timer0
0050-0053 : timer1
0060-0060 : keyboard
0064-0064 : keyboard
0070-0071 : rtc_cmos
    0070-0071 : rtc0
0080-008f : dma page reg
00a0-00a1 : pic2
00c0-00df : dma2
00f0-00ff : fpu
0170-0177 : 0000:00:01.1
    0170-0177 : ata_piix
01f0-01f7 : 0000:00:01.1
    01f0-01f7 : ata_piix
0376-0376 : 0000:00:01.1
    0376-0376 : ata_piix
03c0-03df : vga+
03f6-03f6 : 0000:00:01.1
    03f6-03f6 : ata_piix
0cf8-0cff : PCI conf1
4000-4003 : ACPI PM1a_EVT_BLK
--More--
```



Si deux périphériques ont le même port, les **deux** périphériques seront inutilisables.

Périphériques

```
[root@centos7 ~]# cat /proc/devices
```

Character devices:

```
1 mem
4 /dev/vc/0
4 tty
4 ttyS
5 /dev/tty
5 /dev/console
5 /dev/ptmx
6 lp
7 vcs
10 misc
13 input
14 sound
29 fb
99 ppdev
116 alsa
128 ptm
136 pts
162 raw
180 usb
188 ttyUSB
189 usb_device
202 cpu/msr
203 cpu/cpuid
226 drm
250 hidraw
251 usbmon
252 bsg
253 watchdog
254 rtc
```

Block devices:

```
259 blkext
    8 sd
    9 md
   11 sr
   65 sd
   66 sd
   67 sd
   68 sd
   69 sd
   70 sd
   71 sd
  128 sd
  129 sd
  130 sd
  131 sd
  132 sd
  133 sd
  134 sd
  135 sd
 253 device-mapper
 254 mdp
```

Modules

```
[root@centos7 ~]# cat /proc/modules | more
tcp_lp 12663 0 - Live 0xfffffffffa059d000
lp 17759 0 - Live 0xfffffffffa0593000
nls_utf8 12557 1 - Live 0xfffffffffa0527000
isofs 39844 1 - Live 0xfffffffffa0588000
bnep 19704 2 - Live 0xfffffffffa0516000
bluetooth 372662 7 bnep, Live 0xfffffffffa052c000
rfkill 26536 3 bluetooth, Live 0xfffffffffa051f000
```

```
fuse 87741 3 - Live 0xfffffffffa04ff000
ip6t_rpfilter 12546 1 - Live 0xfffffffffa04fa000
ip6t_REJECT 12939 2 - Live 0xfffffffffa04f5000
ipt_REJECT 12541 2 - Live 0xfffffffffa04f0000
xt_conntrack 12760 7 - Live 0xfffffffffa04e6000
ebtable_nat 12807 0 - Live 0xfffffffffa04e1000
ebtable_broute 12731 0 - Live 0xfffffffffa04eb000
bridge 115385 1 ebtable_broute, Live 0xfffffffffa04c3000
stp 12976 1 bridge, Live 0xfffffffffa04be000
llc 14552 2 bridge,stp, Live 0xfffffffffa04b5000
ebtable_filter 12827 0 - Live 0xfffffffffa04b0000
ebtables 30913 3 ebtable_nat,ebtable_broute,ebtable_filter, Live 0xfffffffffa04a3000
ip6table_nat 12864 1 - Live 0xfffffffffa049e000
nf_conntrack_ipv6 18738 5 - Live 0xfffffffffa0498000
nf_defrag_ipv6 34651 1 nf_conntrack_ipv6, Live 0xfffffffffa048a000
--More--
```

Statistiques de l'utilisation des disques

```
[root@centos7 ~]# cat /proc/diskstats
 11      0 sr0 21 0 132 157 0 0 0 0 157 157
  8      0 sda 19700 638 2178341 386424 13324 1605 547086 356748 0 164849 740706
  8      1 sda1 2129 0 9438 2181 2057 0 4122 392 0 2527 2569
  8      2 sda2 17213 598 2165719 382957 11202 1605 542964 356316 0 164048 736814
  8      3 sda3 170 31 1608 775 0 0 0 0 0 775 775
```

Partitions

```
[root@centos7 ~]# cat /proc/partitions
major minor #blocks name
```

11	0	56876	sr0
8	0	20971520	sda
8	1	204800	sda1
8	2	7168000	sda2
8	3	3072000	sda3

Espaces de pagination

```
[root@centos7 ~]# cat /proc/swaps
Filename                Type              Size    Used    Priority
/dev/sda3                partition        3071996    0      -1
```

Statistiques d'utilisation du processeur

```
[root@centos7 ~]# cat /proc/loadavg
1.82 1.07 0.81 3/379 25349
```

Statistiques d'utilisation de la mémoire

```
[root@centos7 ~]# cat /proc/meminfo
MemTotal:        1791624 kB
MemFree:         72272 kB
MemAvailable:    769436 kB
Buffers:         404 kB
Cached:         790132 kB
SwapCached:      0 kB
Active:         1002556 kB
Inactive:       529520 kB
Active(anon):    620404 kB
Inactive(anon):  131808 kB
```

```
Active(file):      382152 kB
Inactive(file):    397712 kB
Unevictable:       0 kB
Mlocked:           0 kB
SwapTotal:         3071996 kB
SwapFree:          3071992 kB
Dirty:             35168 kB
Writeback:         0 kB
AnonPages:         741568 kB
Mapped:            138756 kB
Shmem:             10664 kB
Slab:              98304 kB
SReclaimable:      59848 kB
SUnreclaim:        38456 kB
KernelStack:       5808 kB
PageTables:        24140 kB
NFS_Unstable:      0 kB
Bounce:            0 kB
WritebackTmp:      0 kB
CommitLimit:       3967808 kB
Committed_AS:      3006392 kB
VmallocTotal:      34359738367 kB
VmallocUsed:        28612 kB
VmallocChunk:      34359706684 kB
HardwareCorrupted: 0 kB
AnonHugePages:     256000 kB
HugePages_Total:   0
HugePages_Free:    0
HugePages_Rsvd:    0
HugePages_Surp:    0
Hugepagesize:      2048 kB
DirectMap4k:       75712 kB
DirectMap2M:       1759232 kB
```

Version du noyau

```
[root@centos7 ~]# cat /proc/version
Linux version 3.10.0-229.4.2.el7.x86_64 (builder@kbuilder.dev.centos.org) (gcc version 4.8.2 20140120 (Red Hat 4.8.2-16) (GCC) ) #1 SMP Wed May 13 10:06:09 UTC 2015
```

Interprétation des informations dans /proc

Les informations brutes stockées dans /proc peuvent être interprétées grâce à l'utilisation des commandes dites de *gestion des performances* :

- free,
- uptime et w,
- iostat,
- vmstat,
- mpstat,
- sar.

Commandes

free

La commande **free** permet de donner l'état de la mémoire totale, libre, partagée, swap et bufferisée. Saisissez donc la commande suivante :

```
[root@centos7 ~]# free -m
```

	total	used	free	shared	buff/cache	available
Mem:	1749	859	76	6	813	707
Swap:	2999	3	2996			

Dans le cas de cet exemple, nous pouvons constater que l'affichage montre :

- 1749 Mo de mémoire physique totale,
- 859 Mo de mémoire physique utilisée et 76 Mo de mémoire physique libre,
- 2999 Mo de mémoire swap totale et 3 Mo de swap utilisé

Les options de cette commande sont :

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

Usage:

free [options]

Options:

-b, --bytes	show output in bytes
-k, --kilo	show output in kilobytes
-m, --mega	show output in megabytes
-g, --giga	show output in gigabytes
--tera	show output in terabytes
-h, --human	show human-readable output
--si	use powers of 1000 not 1024
-l, --lohi	show detailed low and high memory statistics
-t, --total	show total for RAM + swap
-s N, --seconds N	repeat printing every N seconds
-c N, --count N	repeat printing N times, then exit
-w, --wide	wide output
--help	display this help and exit
-V, --version	output version information and exit

For more details see free(1).

uptime ou w

Chacune des ces commandes indique la charge moyenne du ou des processeurs depuis 1 minute, 5 minutes et 15 minutes :

```
[root@centos7 ~]# uptime
14:43:46 up 4:05, 2 users, load average: 0.92, 1.23, 0.95

[root@centos7 ~]# w
14:43:49 up 4:05, 2 users, load average: 0.92, 1.23, 0.95
USER      TTY      FROM          LOGIN@  IDLE   JCPU   PCPU WHAT
trainee   :0        :0             Wed09   ?xdm?  40:01  0.74s gdm-session-worker [pam/gdm-password]
trainee   pts/0     :0             Wed10   5.00s  0.65s 26.90s /usr/libexec/gnome-terminal-server
```

Les valeurs **load average** ou *charge moyenne* indiquent le nombre moyen de processus en cours de traitement ou en attente pour la période concernée.

Par exemple si les valeurs sur un système muni d'un seul processeur étaient **3,48 4,00 3,85** ceci indiquerait que le processeur a du mal à traiter les processus mettant en moyenne :

- 2,48 processus en attente dans la dernière minute,
- 3,00 processus en attente dans les dernières 5 minutes,
- 2,85 processus en attente dans les dernières 15 minutes.

Les options de ces commandes sont :

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

Usage:
  uptime [options]

Options:
  -p, --pretty    show uptime in pretty format
  -h, --help      display this help and exit
  -s, --since     system up since
  -V, --version   output version information and exit

For more details see uptime(1).
```

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

Usage:

```
w [options]
```

Options:

```
-h, --no-header      do not print header
-u, --no-current     ignore current process username
-s, --short          short format
-f, --from           show remote hostname field
-o, --old-style       old style output
-i, --ip-addr        display IP address instead of hostname (if possible)

--help              display this help and exit
-V, --version        output version information and exit
```

For more details see w(1).

iostat

La commande **iostat** affiche des statistiques sur l'utilisation des disques, des terminaux et des lecteurs de cartouche :

```
[root@centos7 ~]# iostat
```

```
Linux 3.10.0-229.4.2.el7.x86_64 (centos7.fenestros.loc)      11/02/2015      _x86_64_      (1 CPU)
```

avg-cpu:	%user	%nice	%system	%iowait	%steal	%idle
	9.00	0.14	2.67	0.29	0.00	87.90

Device:	tps	kB_read/s	kB_wrtn/s	kB_read	kB_wrtn
sdd0	0.00	0.00	0.00	66	0
sda	2.56	95.04	37.09	1409026	549881

Au-dessous de la première ligne indiquant la version du noyau du système et son nom d'hôte ainsi que la date actuelle, iostat affiche une vue

d'ensemble de l'utilisation CPU moyenne du système depuis le dernier démarrage. Le rapport d'utilisation du CPU inclut les pourcentages suivants :

- Pourcentage de temps passé en mode utilisateur (exécutant des applications, etc.)
- Pourcentage de temps passé en mode utilisateur (pour les processus qui ont modifié leur priorité de programmation à l'aide de la commande `nice`)
- Pourcentage de temps passé en mode noyau
- Pourcentage de temps passé en inactivité

Notez la valeur de **%iowait**. Dans le cas où ce pourcentage est trop élevé, ceci indique que le processeur passe son temps à attendre les entrées et les sorties de disque.

Pour surveiller la vitesse des entrées et des sorties du disque, vous pouvez utiliser la commande **hdparm**.

Sous RHEL/CentOS 7, la commande **hdparm** n'est pas disponible par défaut. Installez donc le paquet **hdparm** :

```
[root@centos7 ~]# yum install hdparm
```

```
[root@centos7 ~]# hdparm -t /dev/sda
```

```
/dev/sda:
```

```
Timing buffered disk reads: 254 MB in 3.02 seconds = 84.19 MB/sec
```

Au-dessous du rapport d'utilisation du CPU de la sortie de la commande **iostat** figure le rapport d'utilisation des périphériques. Ce dernier contient une ligne pour chaque périphérique disque du système et inclut les informations suivantes :

- La spécification du périphérique, apparaissant sous la forme `dev<major-number>-sequence-number` où `<major-number>` correspond au nombre majeur du périphérique et `<sequence-number>` correspond à un numéro de séquence commençant par zéro.
- Le nombre de transferts (ou opérations d'E/S) par seconde.
- Le nombre de blocs de 512 octets lus par seconde.
- Le nombre de blocs de 512 octets écrits par seconde.
- Le nombre total de blocs de 512 octets lus par seconde.
- Le nombre total de blocs de 512 octets écrits par seconde.

Les options de cette commande sont :

```
[root@centos7 ~]# iostat --help
Usage: iostat [ options ] [ <interval> [ <count> ] ]
Options are:
[ -c ] [ -d ] [ -h ] [ -k | -m ] [ -N ] [ -t ] [ -V ] [ -x ] [ -y ] [ -z ]
[ -j { ID | LABEL | PATH | UUID | ... } ]
[ [ -T ] -g <group_name> ] [ -p [ <device> [,...] | ALL ] ]
[ <device> [...] | ALL ]
```

vmstat

La commande **vmstat** affiche des statistiques sur la mémoire, la pagination et la charge ponctuelle du processeur :

```
[root@centos7 ~]# vmstat 1 10
procs -----memory----- ---swap-- -----io----- -system-- -----cpu-----
 r  b   swpd   free   buff  cache   si   so    bi    bo    in   cs  us  sy  id  wa  st
 5  0    4600 352560    20 620332    0    0   114   37   157  236  9   3 88   0   0
 1  0    4600 352736    20 620368    0    0    0    0   231  310 29   7 63   0   0
 0  0    4600 352736    20 620368    0    0    0    0   223  318 32   7 60   0   0
 1  0    4600 352776    20 620328    0    0    0    0   250  339 34   9 57   0   0
 0  0    4600 352776    20 620328    0    0    0    0   215  285 30   9 61   0   0
 0  0    4600 352776    20 620328    0    0    0    0   234  326 31   7 61   0   0
 1  0    4600 352776    20 620328    0    0    0    0   219  306 31   8 62   0   0
 0  0    4600 352776    20 620328    0    0    0    0   223  298 29   7 63   0   0
 0  0    4600 352776    20 620328    0    0    0    0   223  314 32   7 60   0   0
 1  0    4600 352776    20 620332    0    0    0    0   238  309 34   7 59   0   0
```

La première ligne subdivise le champ en six catégories à savoir : processus, mémoire, swap, E/S, système et CPU sur lesquelles elle donne des statistiques. La seconde ligne identifie de manière encore plus détaillée chacun des champs, permettant ainsi de parcourir simplement et rapidement l'ensemble des données lors de la recherche de statistiques spécifiques.

Les champs relatifs aux processus sont les suivants :

- r — Le nombre de processus exécutables attendant d'avoir accès au CPU

- b — Le nombre de processus exécutables dans un état de veille qui ne peut être interrompu

Les champs relatifs à la mémoire sont les suivants :

- swpd — La quantité de mémoire virtuelle utilisée
- free — La quantité de mémoire libre
- buff — La quantité de mémoire utilisée par les tampons (ou buffers)
- cache — La quantité de mémoire utilisée comme cache de pages

Les champs relatifs au swap sont les suivants :

- si — La quantité de mémoire chargée depuis le disque
- so — La quantité de mémoire déchargée sur le disque

Les champs relatifs aux Entrées/Sorties (E/S) sont les suivants :

- bi — Blocs envoyés vers un périphérique blocs
- bo — Blocs reçus d'un périphérique blocs

Les champs relatifs au système sont les suivants :

- in — Nombre d'interruptions par seconde
- cs — Nombre de changements de contexte par seconde

Les champs relatifs au CPU sont les suivants :

- us — Le pourcentage de temps pendant lequel le CPU exécute un code de niveau utilisateur
- sy — Le pourcentage de temps pendant lequel le CPU exécute un code de niveau système
- id — Le pourcentage de temps pendant lequel le CPU était inoccupé
- wa — Attente d'E/S

Les options de cette commande sont :

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

Usage:

```
vmstat [options] [delay [count]]
```

Options:

```
-a, --active          active/inactive memory
-f, --forks           number of forks since boot
-m, --slabs           slabinfo
-n, --one-header      do not redisplay header
-s, --stats           event counter statistics
-d, --disk            disk statistics
-D, --disk-sum        summarize disk statistics
-p, --partition <dev> partition specific statistics
-S, --unit <char>     define display unit
-w, --wide            wide output
-t, --timestamp       show timestamp

-h, --help            display this help and exit
-V, --version          output version information and exit
```

For more details see `vmstat(8)`.



Important : Par défaut la commande `vmstat` affiche des informations depuis le démarrage du système.

mpstat

La commande **mpstat** affiche des statistiques détaillées sur le CPU :

```
[root@centos7 ~]# mpstat
Linux 3.10.0-229.4.2.el7.x86_64 (centos7.fenestros.loc)      11/02/2015      _x86_64_      (1 CPU)

03:19:32 PM  CPU      %usr   %nice    %sys %iowait    %irq   %soft  %steal  %guest  %gnice   %idle
```

03:19:32 PM	all	9.38	0.14	2.52	0.31	0.00	0.24	0.00	0.00	0.00	87.42
-------------	-----	------	------	------	------	------	------	------	------	------	-------

Dans le cas où vous avez plusieurs processeurs ou coeurs, vous pouvez visualiser ces mêmes informations par unité de traitement :

```
[root@centos7 ~]# mpstat -P ALL
Linux 3.10.0-229.4.2.el7.x86_64 (centos7.fenestros.loc)      11/02/2015      _x86_64_      (1 CPU)

03:20:06 PM  CPU      %usr   %nice    %sys %iowait    %irq   %soft  %steal  %guest  %gnice   %idle
03:20:06 PM  all       9.42    0.14    2.53    0.31    0.00    0.24    0.00    0.00    0.00   87.36
03:20:06 PM    0       9.42    0.14    2.53    0.31    0.00    0.24    0.00    0.00    0.00   87.36
```

Pour afficher 5 jeux de statistiques à des intervalles de 2 secondes pour tous les unités de traitement, il convient d'utiliser la commande suivante :

```
[root@centos7 ~]# mpstat -P ALL 2 5
Linux 3.10.0-229.4.2.el7.x86_64 (centos7.fenestros.loc)      11/02/2015      _x86_64_      (1 CPU)

03:20:43 PM  CPU      %usr   %nice    %sys %iowait    %irq   %soft  %steal  %guest  %gnice   %idle
03:20:45 PM  all      27.46    0.00    7.04    0.00    0.00    0.00    0.00    0.00    0.00   65.49
03:20:45 PM    0      27.46    0.00    7.04    0.00    0.00    0.00    0.00    0.00    0.00   65.49

03:20:45 PM  CPU      %usr   %nice    %sys %iowait    %irq   %soft  %steal  %guest  %gnice   %idle
03:20:47 PM  all      24.83    0.00    5.37    0.00    0.00    0.67    0.00    0.00    0.00   69.13
03:20:47 PM    0      24.83    0.00    5.37    0.00    0.00    0.67    0.00    0.00    0.00   69.13

03:20:47 PM  CPU      %usr   %nice    %sys %iowait    %irq   %soft  %steal  %guest  %gnice   %idle
03:20:49 PM  all      17.95    0.00    4.49    0.00    0.00    0.00    0.00    0.00    0.00   77.56
03:20:49 PM    0      17.95    0.00    4.49    0.00    0.00    0.00    0.00    0.00    0.00   77.56

03:20:49 PM  CPU      %usr   %nice    %sys %iowait    %irq   %soft  %steal  %guest  %gnice   %idle
03:20:51 PM  all      18.18    0.00    3.90    0.00    0.00    0.00    0.00    0.00    0.00   77.92
03:20:51 PM    0      18.18    0.00    3.90    0.00    0.00    0.00    0.00    0.00    0.00   77.92

03:20:51 PM  CPU      %usr   %nice    %sys %iowait    %irq   %soft  %steal  %guest  %gnice   %idle
03:20:53 PM  all      24.14    0.00    6.21    0.00    0.00    0.00    0.00    0.00    0.00   69.66
```

```
03:20:53 PM    0   24.14    0.00    6.21    0.00    0.00    0.00    0.00    0.00    0.00    0.00   69.66

Average:  CPU    %usr    %nice    %sys %iowait    %irq    %soft    %steal    %guest    %gnice    %idle
Average:   all   22.39    0.00    5.36    0.00    0.00    0.13    0.00    0.00    0.00    72.12
Average:    0   22.39    0.00    5.36    0.00    0.00    0.13    0.00    0.00    0.00    72.12
```

Les options de cette commande sont :

```
[root@centos7 ~]# mpstat --help
Usage: mpstat [ options ] [ <interval> [ <count> ] ]
Options are:
[ -A ] [ -u ] [ -V ] [ -I { SUM | CPU | SCPU | ALL } ]
[ -P { <cpu> [,...] | ON | ALL } ]
```

sar

La commande **sar** permet de surveiller toutes les ressources du système selon l'option qui est passée en argument à la commande. Quelques options importantes sont :

Option	Description
-u	Pourcentage d'utilisation du CPU
-q	Nombre de processus en attente
-r	Utilisation de la mémoire centrale
-w	Surveillance du swapping
-p	Surveillance de la pagination
-b	Utilisation des tampons
-d	Utilisation des disques

Sous RHEL/CentOS 7 la commande **/usr/lib64/sa/sadc** permet de collecter les informations :

```
[root@centos7 ~]# ls /usr/lib64/sa
```

sa1 sa2 sadc

Le script **/usr/lib64/sa/sa1** exécute la commande **sadc**. Ce script prend deux options :

Option	Description
-t	L'interval entre les collectes
-n	Nombre de collectes

Le script **/usr/lib64/sa/sa2** exécute la commande **sar** et consigne les informations dans un fichier au format **/var/log/sa/sar<jj>**.

Pour pouvoir fonctionner correctement, ces scripts doivent être appelés par **cron**.

Modifiez le fichier **/etc/cron.d/sysstat** ainsi :

```
[root@centos7 ~]# cat /etc/cron.d/sysstat
# Run system activity accounting tool every 10 minutes
# */10 * * * * root /usr/lib64/sa/sa1 1 1
*/2 * * * * root /usr/lib64/sa/sa1 1 1
# 0 * * * * root /usr/lib64/sa/sa1 600 6 &
# Generate a daily summary of process accounting at 23:53
53 23 * * * root /usr/lib64/sa/sa2 -A
```

Attendez deux minutes puis saisissez les commandes suivantes :

```
[root@centos7 ~]# sar
Linux 3.10.0-229.4.2.el7.x86_64 (centos7.fenestros.loc)      11/02/2015      _x86_64_      (1 CPU)

02:20:02 PM      CPU      %user      %nice      %system      %iowait      %steal      %idle
02:30:01 PM      all       12.95        0.01         3.36         0.01         0.00       83.67
02:40:01 PM      all       27.64        0.50         7.84         0.06         0.00       63.96
02:50:01 PM      all       11.56        2.46         4.59         1.18         0.00       80.21
03:00:01 PM      all         0.45        0.01         0.64         0.03         0.00       98.88
03:10:01 PM      all         0.38        0.01         0.61         0.01         0.00       98.98
03:20:01 PM      all       25.80        0.02         6.68         0.85         0.00       66.66
```

03:28:02 PM	all	21.59	0.06	6.84	0.21	0.00	71.31
03:30:01 PM	all	22.88	0.01	5.76	0.00	0.00	71.35
03:32:01 PM	all	11.34	0.01	2.99	0.11	0.00	85.55
Average:	all	13.21	0.41	4.01	0.30	0.00	82.06

```
[root@centos7 ~]# sar -u 5 3
Linux 3.10.0-229.4.2.el7.x86_64 (centos7.fenestros.loc)      11/02/2015      _x86_64_      (1 CPU)
```

	CPU	%user	%nice	%system	%iowait	%steal	%idle
03:34:01 PM	all	29.73	0.00	14.19	0.00	0.00	56.08
03:34:11 PM	all	17.62	0.00	4.40	0.00	0.00	77.98
03:34:16 PM	all	4.75	0.00	1.30	0.00	0.00	93.95
Average:	all	15.55	0.00	5.68	0.00	0.00	78.78

```
[root@centos7 ~]# sar -r 5 3
Linux 3.10.0-229.4.2.el7.x86_64 (centos7.fenestros.loc)      11/02/2015      _x86_64_      (1 CPU)
```

	kbmemfree	kbmemused	%memused	kbbuffers	kbcached	kbcommit	%commit	kbactive	kbinact	kbdirty
03:34:48 PM	137708	1653916	92.31	20	561940	2950052	60.66	776232	539396	12
03:34:53 PM	130988	1660636	92.69	20	566552	3000196	61.69	779820	542436	20
03:35:03 PM	131360	1660264	92.67	20	566584	2999016	61.66	779288	542472	48
Average:	133352	1658272	92.56	20	565025	2983088	61.33	778447	541435	27

```
[root@centos7 ~]# sar -w 5 3
Linux 3.10.0-229.4.2.el7.x86_64 (centos7.fenestros.loc)      11/02/2015      _x86_64_      (1 CPU)
```

	proc/s	cswch/s
03:35:51 PM	0.00	608.04
03:35:56 PM	1.06	531.75
03:36:01 PM	25.94	1226.28
Average:	8.36	767.19

```
[root@centos7 ~]# sar -b 5 3
Linux 3.10.0-229.4.2.el7.x86_64 (centos7.fenestros.loc)      11/02/2015      _x86_64_      (1 CPU)
```

03:36:09 PM	tps	rtps	wtps	bread/s	bwrtn/s
03:36:14 PM	0.00	0.00	0.00	0.00	0.00
03:36:19 PM	0.00	0.00	0.00	0.00	0.00
03:36:24 PM	2.32	0.00	2.32	0.00	31.79
Average:	0.89	0.00	0.89	0.00	12.25

```
[root@centos7 ~]# sar -d 5 3
```

```
Linux 3.10.0-229.4.2.el7.x86_64 (centos7.fenestros.loc) 11/02/2015 _x86_64_ (1 CPU)
```

03:36:46 PM	DEV	tps	rd_sec/s	wr_sec/s	avgrq-sz	avgqu-sz	await	svctm	%util
03:36:51 PM	dev11-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
03:36:51 PM	dev8-0	0.31	0.00	2.45	8.00	0.00	0.00	0.00	0.00
03:36:51 PM	DEV	tps	rd_sec/s	wr_sec/s	avgrq-sz	avgqu-sz	await	svctm	%util
03:36:56 PM	dev11-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
03:36:56 PM	dev8-0	4.31	0.00	55.38	12.86	0.04	9.36	1.93	0.83
03:36:56 PM	DEV	tps	rd_sec/s	wr_sec/s	avgrq-sz	avgqu-sz	await	svctm	%util
03:37:01 PM	dev11-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
03:37:01 PM	dev8-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Average:	DEV	tps	rd_sec/s	wr_sec/s	avgrq-sz	avgqu-sz	await	svctm	%util
Average:	dev11-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Average:	dev8-0	1.49	0.00	18.63	12.53	0.01	8.73	1.80	0.27

```
[root@centos7 ~]# sar -p 5 3
```

```
Linux 3.10.0-229.4.2.el7.x86_64 (centos7.fenestros.loc) 11/02/2015 _x86_64_ (1 CPU)
```

03:37:16 PM	CPU	%user	%nice	%system	%iowait	%steal	%idle
03:37:21 PM	all	38.30	0.00	7.31	0.00	0.00	54.39
03:37:26 PM	all	45.81	0.00	9.58	0.00	0.00	44.61
03:37:31 PM	all	33.33	0.00	7.67	0.00	0.00	59.00
Average:	all	39.11	0.00	8.18	0.00	0.00	52.71

```
[root@centos7 ~]# sar -v 5 3
Linux 3.10.0-229.4.2.el7.x86_64 (centos7.fenestros.loc)      11/02/2015      _x86_64_      (1 CPU)

03:38:44 PM dentunusd      file-nr  inode-nr      pty-nr
03:38:49 PM      136266      5792      122548          2
03:38:54 PM      136266      5792      122548          2
03:38:59 PM      136266      5792      122548          2
Average:          136266      5792      122548          2
```

Les options de cette commande sont :

```
[root@centos7 ~]# sar --help
Usage: sar [ options ] [ <interval> [ <count> ] ]
Options are:
[ -A ] [ -B ] [ -b ] [ -C ] [ -d ] [ -H ] [ -h ] [ -p ] [ -q ] [ -R ]
[ -r ] [ -S ] [ -t ] [ -u [ ALL ] ] [ -V ] [ -v ] [ -W ] [ -w ] [ -y ]
[ -I { <int> [,...] | SUM | ALL | XALL } ] [ -P { <cpu> [,...] | ALL } ]
[ -m { <keyword> [,...] | ALL } ] [ -n { <keyword> [,...] | ALL } ]
[ -j { ID | LABEL | PATH | UUID | ... } ]
[ -f [ <filename> ] | -o [ <filename> ] | -[0-9]+ ]
[ -i <interval> ] [ -s [ <hh:mm:ss> ] ] [ -e [ <hh:mm:ss> ] ]
```

Utilisation des commandes en production

Identifier un système limité par le processeur

Dans ce cas utilisez les commandes suivantes :

- uptime ou w
- vmstat
- mpstat -P ALL
- sar -u

- iostat -c

Identifier un système ayant un problème de mémoire

Dans ce cas utilisez les commandes suivantes :

- free
- sar -B

Identifier un système ayant un problème d'E/S

Utilisez la commande :

- iostat -d -x

Modules usb

L'**USB** (*Universal Serial Bus*) est un bus de données qui peut offrir des taux de transfert jusqu'à 480Mb/s sous la version 2.0 et jusqu'à 4.8 Gb/s sous la version 3.0. Les modules nécessaires pour les contrôleurs USB sont :

Version USB	Module	Nom Complet
1.0\1.1	UHCI	<i>Universal Controller Host Interface</i>
	OHCI	<i>Open Controller Host Interface</i>
2.0	EHCI	<i>Enhanced Host Controller Interface</i>
3.0	XHCI	<i>Extensible Host Controller Interface</i>

Le tableau suivant liste les modules couramment chargés en fonction du périphérique utilisé :

Module	Type de Périphérique
usb_storage	Supports de masse

Module	Type de Périphérique
usbhid	Périphériques HID (<i>Human Interface Device</i>)
snd-usb-audio	Cartes son usb
usbvidéo	Cartes vidéo et d'acquisition
irda-usb	Périphériques infrarouges
usbnet	Cartes réseaux usb

Les modules peuvent être chargés par un des moyens suivants :

- INITrd,
- Le processus init,
- kmod, d'une manière dynamique et transparente lors du branchement du périphérique, en utilisant le fichier **/lib/modules/2.6.18-194.3.1.el5/modules.usbmap**,
- udev,
- manuellement.



A faire : Branchez une clef USB avant de continuer. Si vous utilisez VirtualBox, activez la clef dans votre machine virtuelle grâce aux menus **Périphériques > Périphériques USB > le_nom_de_votre_clef**.

udev

Depuis le noyau Linux 2.6 Linux est capable de détecter des périphériques branchés à chaud. Cette technologie s'appelle le **hotplugging**. Le **hotplugging** est obtenu grâce à l'utilisation de trois composants :

- Udev,
- HAL,
- Dbus.

Les rôles de chaque composant sont les suivants :

- Udev se charge de créer et supprimer d'une manière dynamique les nœuds dans le répertoire **/dev**,
- HAL obtient des informations à partir d'Udev et crée un fichier au format XML représentant le périphérique branché. Il informe ensuite Nautilus en utilisant le Dbus,
- Dbus joue le rôle d'un bus système qui est utilisé pour la communication inter-processus.

Lors de démarrage de Linux, Udev joue un rôle important :

- Au démarrage **tmpfs** est monté sur **/dev**,
- Udev copie les éventuels nœuds statiques de **/lib/udev/devices** vers **/dev**,
- le démon **udevd** collecte des données appelées **uevents** du noyau et cherche une règle correspondante dans le répertoire **/lib/udev/rules.d/**,
- Udev crée les nœuds et liens symboliques spécifiés dans la règle identifiée,
- Udev stocke les règles contenues dans **/lib/udev/rules.d/*.rules** en mémoire,
- En cas de modification des ces règles, Udev met à jour la mémoire.

Udev repose sur le filesystem **sysfs** monté sur **/sys** qui permet de rendre les périphériques visibles à Udev dans l'*User Space*. Par exemple, lors du branchement d'une clé USB, Udev crée **/dev/sdb1** automatiquement et utilise les informations contenues dans le fichier **/lib/modules/`uname -r`/modules.alias** pour trouver le pilote nécessaire :

Le fichier de configuration principal d'Udev est **/etc/udev/udev.conf** :

```
[root@centos7 ~]# cat /etc/udev/udev.conf
# see udev(7) for details

#udev_log="info"
```

Les fichiers de règles se trouvent dans **/lib/udev/rules.d/** :

```
[root@centos7 ~]# ls /lib/udev/rules.d/
10-dm.rules          61-gnome-bluetooth-rfkill.rules  77-mm-ericsson-mbm.rules      90-
iprutils.rules       62-multipath.rules              77-mm-huawei-net-port-types.rules  90-
11-dm-lvm.rules      63-md-raid-arrays.rules         77-mm-longcheer-port-types.rules  90-
pulseaudio.rules
11-dm-mpath.rules
vconsole.rules
```

13-dm-disk.rules	64-btrfs.rules	77-mm-nokia-port-types.rules	91-
drm-modeset.rules			
40-libgphoto2.rules	65-libwacom.rules	77-mm-pcmcia-device-blacklist.rules	95-
cd-devices.rules			
40-redhat.rules	65-md-incremental.rules	77-mm-platform-serial-whitelist.rules	95-
dm-notify.rules			
40-usb_modeswitch.rules	65-sane-backends.rules	77-mm-simtech-port-types.rules	95-
udev-late.rules			
42-usb-hid-pm.rules	69-cd-sensors.rules	77-mm-telit-port-types.rules	95-
upower-battery-recall-dell.rules			
50-rbd.rules	69-dm-lvm-metad.rules	77-mm-usb-device-blacklist.rules	95-
upower-battery-recall-fujitsu.rules			
50-udev-default.rules	69-libmtp.rules	77-mm-usb-serial-adapters-greylist.rules	95-
upower-battery-recall-gateway.rules			
56-hpmud.rules	69-xorg-vmmouse.rules	77-mm-x22x-port-types.rules	95-
upower-battery-recall-ibm.rules			
60-alias-kmsg.rules	70-power-switch.rules	77-mm-zte-port-types.rules	95-
upower-battery-recall-lenovo.rules			
60-cdrom_id.rules	70-printers.rules	77-nm-olpc-mesh.rules	95-
upower-battery-recall-toshiba.rules			
60-fprint-autosuspend.rules	70-spice-vdagentd.rules	78-sound-card.rules	95-
upower-csr.rules			
60-keyboard.rules	70-touchpad-quirks.rules	80-drivers.rules	95-
upower-hid.rules			
60-net.rules	70-uaccess.rules	80-kvm.rules	95-
upower-wup.rules			
60-persistent-alsa.rules	70-wacom.rules	80-mm-candidate.rules	97-
bluetooth-serial.rules			
60-persistent-input.rules	71-biosdevname.rules	80-net-name-slot.rules	98-
kexec.rules			
60-persistent-serial.rules	71-seat.rules	80-udisks2.rules	98-
rdma.rules			
60-persistent-storage.rules	73-idrac.rules	81-kvm-rhel.rules	99-
qemu-guest-agent.rules			

60-persistent-storage-tape.rules	73-seat-late.rules	85-regulatory.rules	99-
systemd.rules			
60-persistent-v4l.rules	75-net-description.rules	85-usbmuxd.rules	
60-raw.rules	75-probe_mtd.rules	90-alsa-restore.rules	
61-accelerometer.rules	75-tty-description.rules	90-alsa-tools-firmware.rules	



Important : Il vous est possible d'ajouter des règles si besoin est. Dans ce cas, créez un fichier **99-local.rules** et éditez-le au lieu d'éditer les fichiers existants.

Comme indique le nom de chaque fichier, le contenu est composé de règles à l'attention d'udev. Le fichier des règles par défaut est le **50-udev-default.rules** :

```
[root@centos7 ~]# cat /lib/udev/rules.d/50-udev-default.rules | more
# do not edit this file, it will be overwritten on update

SUBSYSTEM=="virtio-ports", KERNEL=="vport*", ATTR{name}=="?*", SYMLINK+="virtio-ports/${attr{name}}"

# select "system RTC" or just use the first one
SUBSYSTEM=="rtc", ATTR{hctosys}=="1", SYMLINK+="rtc"
SUBSYSTEM=="rtc", KERNEL=="rtc0", SYMLINK+="rtc", OPTIONS+="link_priority=-100"

SUBSYSTEM=="usb", ENV{DEVTYPE}=="usb_device", IMPORT{builtin}="usb_id", IMPORT{builtin}="hwdb --subsystem=usb"
SUBSYSTEM=="input", ENV{ID_INPUT}=="", IMPORT{builtin}="input_id"
ENV{MODALIAS}!="", IMPORT{builtin}="hwdb --subsystem=$env{SUBSYSTEM}"

ACTION!="add", GOTO="default_permissions_end"

SUBSYSTEM=="tty", KERNEL=="ptmx", GROUP="tty", MODE="0666"
SUBSYSTEM=="tty", KERNEL=="tty", GROUP="tty", MODE="0666"
SUBSYSTEM=="tty", KERNEL=="tty[0-9]*", GROUP="tty", MODE="0620"
SUBSYSTEM=="tty", KERNEL=="sclp_line[0-9]*", GROUP="tty", MODE="0620"
```

```

SUBSYSTEM=="tty", KERNEL=="ttysclp[0-9]*", GROUP="tty", MODE="0620"
SUBSYSTEM=="tty", KERNEL=="3270/tty[0-9]*", GROUP="tty", MODE="0620"
SUBSYSTEM=="vc", KERNEL=="vcs*|vcsa*", GROUP="tty"
KERNEL=="tty[A-Z]*[0-9]|ppox[0-9]*|ircomm[0-9]*|noz[0-9]*|rfcomm[0-9]*", GROUP="dialout"

SUBSYSTEM=="mem", KERNEL=="mem|kmem|port", GROUP="kmem", MODE="0640"

SUBSYSTEM=="input", KERNEL=="js[0-9]*", MODE="0664"

SUBSYSTEM=="video4linux", GROUP="video"
SUBSYSTEM=="misc", KERNEL=="agpgart", GROUP="video"
--More--

```

Chaque règle prend la forme suivante :

KEY, [KEY, ...] NAME [, SYMLINK]

Chaque KEY est un champ au format **type=valeur** qui doit correspondre à un périphérique unique. La valeur de type peut prendre plusieurs formes :

Type	Description	Exemples
BUS	Type de bus	usb, scsi, ide
KERNEL	Le nom par défaut du périphérique donné par le noyau	hda, ttyUSB0, lp0
SUBSYSTEM	Le nom noyau du sous-système, généralement identique à la valeur du BUS	usb, scsi
DRIVER	Le nom du pilote qui contrôle le périphérique	usb-storage
ID	Le numéro du périphérique sur son bus	PCI bus id, USB id
PLACE	Ne concerne que les périphériques USB et donne la position topologique du périphérique sur son bus	S/O
SYSFS{filename}	Le nom du fichier dans /sys pour le périphérique. Ce fichier contient le fabricant, le label, le numéro de série et UUID du périphérique. La vérification de jusqu'à 5 fichiers est possible par règle	S/O
PROGRAM	Ceci permet à Udev d'appeler un programme externe pour nommer un périphérique	S/O
RESULT	Valeur à comparer au résultat de PROGRAM	S/O

NAME et SYMLINK sont utilisées pour stipuler ce que Udev doit faire avec le périphérique :

Type	Description	Exemples
NAME	Le nome du nœud dans /dev	S/O
SYMLINK	Le ou les lien(s) symbolique(s) qui pointe(nt) vers le NAME	S/O

La commande udevadm

Pour obtenir de l'information sur un périphérique il convient d'utiliser la commande **udevadm** :

[illegible]

```
E: MAJOR=8
E: MINOR=0
E: MPATH_SBIN_PATH=/sbin
E: SUBSYSTEM=block
E: TAGS=:systemd:
E: USEC_INITIALIZED=957366
```

Les options de la commande

Les options de la commande udevadm sont :

```
[root@centos7 ~]# udevadm --help
Usage: udevadm [--help] [--version] [--debug] COMMAND [COMMAND OPTIONS]
  info          query sysfs or the udev database
  trigger       request events from the kernel
  settle        wait for the event queue to finish
  control       control the udev daemon
  monitor       listen to kernel and udev events
  hwdb          maintain the hardware database index
  test          test an event run
  test-builtin  test a built-in command

[root@centos7 ~]# udevadm info --help
Usage: udevadm info OPTIONS
  --query=<type>      query device information:
                      name          name of device node
                      symlink       pointing to node
                      path          sys device path
                      property      the device properties
                      all           all values
  --path=<syspath>    sys device path used for query or attribute walk
  --name=<name>       node or symlink name used for query or attribute walk
  --root            prepend dev directory to path names
```

```
--attribute-walk      print all key matches while walking along the chain
                        of parent devices
--device-id-of-file=<file> print major:minor of device containing this file
--export              export key/value pairs
--export-prefix       export the key name with a prefix
--export-db           export the content of the udev database
--cleanup-db          cleanup the udev database
--help
```

Système de fichiers /sys

Le système de fichiers virtuel **/sys** a été introduit avec le noyau Linux **2.6**. Son rôle est de décrire le matériel pour udev.

Saisissez la commande suivante :

```
[root@centos7 ~]# ls -l /sys
total 0
drwxr-xr-x.  2 root root 0 Oct 28 09:40 block
drwxr-xr-x. 24 root root 0 Oct 28 09:40 bus
drwxr-xr-x. 45 root root 0 Oct 28 09:40 class
drwxr-xr-x.  4 root root 0 Oct 28 09:40 dev
drwxr-xr-x. 11 root root 0 Oct 28 09:40 devices
drwxr-xr-x.  5 root root 0 Oct 28 09:40 firmware
drwxr-xr-x.  6 root root 0 Oct 28 09:40 fs
drwxr-xr-x.  2 root root 0 Oct 28 09:40 hypervisor
drwxr-xr-x.  9 root root 0 Oct 28 09:40 kernel
drwxr-xr-x. 146 root root 0 Oct 28 09:40 module
drwxr-xr-x.  2 root root 0 Oct 28 09:40 power
```

Chaque répertoire contient des informations :

- **block**

- contient des informations sur les périphériques bloc
- **bus**
 - contient des informations sur les bus de données
- **class**
 - contient des informations sur des classes de matériel
- **devices**
 - contient des informations sur la position des périphériques sur les bus
- **firmware**
 - contient, entre autre, des informations sur l'ACPI
- **module**
 - contient des informations sur les modules du noyau
- **power**
 - contient des informations sur la gestion de l'énergie
- **fs**
 - contient des informations sur les systèmes de fichiers

Pour illustrer ceci, saisissez la commande suivante :

```
[root@centos7 ~]# cat /sys/block/sda/sda1/size
409600
```

Ce chiffre correspond aux nombre de secteurs.

Limitation des ressources

ulimit

Les ressources disponibles aux utilisateurs peuvent être limitées par l'utilisation de la commande **ulimit**.

La commande **ulimit** gère deux types de limite, la limite *hard* en utilisant l'option **-H** et la limite *soft* en utilisant l'option **-S**. Seul root peut positionner une limite *hard* et ceci à condition que la limite ne dépasse pas les ressources réelles.

La limite *soft* est la limite imposée à l'utilisateur par défaut tandis que la limite *hard* est la limite que l'utilisateur peut atteindre en utilisant la commande ulimit lui-même.

L'utilisateur root peut paramétrer les limites accordées en éditant la fichier **/etc/security/limits.conf** :

```
[root@centos7 ~]# cat /etc/security/limits.conf
# /etc/security/limits.conf
#
#This file sets the resource limits for the users logged in via PAM.
#It does not affect resource limits of the system services.
#
#Also note that configuration files in /etc/security/limits.d directory,
#which are read in alphabetical order, override the settings in this
#file in case the domain is the same or more specific.
#That means for example that setting a limit for wildcard domain here
#can be overridden with a wildcard setting in a config file in the
#subdirectory, but a user specific setting here can be overridden only
#with a user specific setting in the subdirectory.
#
#Each line describes a limit for a user in the form:
#
#<domain>          <type>  <item>  <value>
#
#Where:
#<domain> can be:
#      - a user name
#      - a group name, with @group syntax
#      - the wildcard *, for default entry
#      - the wildcard %, can be also used with %group syntax,
#        for maxlogin limit
#
#<type> can have the two values:
#      - "soft" for enforcing the soft limits
#      - "hard" for enforcing hard limits
```

```
#
#<item> can be one of the following:
#   - core - limits the core file size (KB)
#   - data - max data size (KB)
#   - fsize - maximum filesize (KB)
#   - memlock - max locked-in-memory address space (KB)
#   - nofile - max number of open file descriptors
#   - rss - max resident set size (KB)
#   - stack - max stack size (KB)
#   - cpu - max CPU time (MIN)
#   - nproc - max number of processes
#   - as - address space limit (KB)
#   - maxlogins - max number of logins for this user
#   - maxsyslogins - max number of logins on the system
#   - priority - the priority to run user process with
#   - locks - max number of file locks the user can hold
#   - sigpending - max number of pending signals
#   - msgqueue - max memory used by POSIX message queues (bytes)
#   - nice - max nice priority allowed to raise to values: [-20, 19]
#   - rtprio - max realtime priority
#
#<domain>      <type>  <item>          <value>
#
#*              soft    core             0
#*              hard    rss              10000
#@student       hard    nproc             20
#@faculty       soft    nproc             20
#@faculty       hard    nproc             50
#ftp            hard    nproc             0
#@student       -       maxlogins          4

# End of file
```



Important : La valeur de la limite peut être un **nombre** ou le mot **unlimited**.

Par exemple, si root inscrit les deux ligne suivantes dans le fichier `/etc/security/limits.conf` :

```
...  
trainee          soft    nofile    1024  
trainee          hard    nofile    4096  
...
```

la limite du nombre de fichiers ouverts simultanément par trainee est de 1 024. Par contre, trainee a la possibilité d'augmenter cette limite jusqu'à 4 096 en utilisant la commande suivante :

```
$ ulimit -n 4096
```

Pour consulter la liste des limites actuelles, il convient d'utiliser la commande `ulimit` avec l'option **-a** :

```
[root@centos7 ~]# ulimit -a  
core file size          (blocks, -c) 0  
data seg size           (kbytes, -d) unlimited  
scheduling priority     (-e) 0  
file size               (blocks, -f) unlimited  
pending signals         (-i) 6929  
max locked memory       (kbytes, -l) 64  
max memory size         (kbytes, -m) unlimited  
open files              (-n) 1024  
pipe size               (512 bytes, -p) 8  
POSIX message queues    (bytes, -q) 819200  
real-time priority      (-r) 0  
stack size              (kbytes, -s) 8192  
cpu time                (seconds, -t) unlimited  
max user processes      (-u) 6929
```

virtual memory	(kbytes, -v) unlimited
file locks	(-x) unlimited

Options de la commande

Les options de **ulimit** sont :

```
[root@centos7 ~]# help ulimit
ulimit: ulimit [-SHacdefilmnpqrstuvx] [limit]
  Modify shell resource limits.
  Provides control over the resources available to the shell and processes
  it creates, on systems that allow such control.
  Options:
    -S      use the `soft' resource limit
    -H      use the `hard' resource limit
    -a      all current limits are reported
    -b      the socket buffer size
    -c      the maximum size of core files created
    -d      the maximum size of a process's data segment
    -e      the maximum scheduling priority (`nice')
    -f      the maximum size of files written by the shell and its children
    -i      the maximum number of pending signals
    -l      the maximum size a process may lock into memory
    -m      the maximum resident set size
    -n      the maximum number of open file descriptors
    -p      the pipe buffer size
    -q      the maximum number of bytes in POSIX message queues
    -r      the maximum real-time scheduling priority
    -s      the maximum stack size
    -t      the maximum amount of cpu time in seconds
    -u      the maximum number of user processes
    -v      the size of virtual memory
    -x      the maximum number of file locks
```

```
If LIMIT is given, it is the new value of the specified resource; the
special LIMIT values `soft', `hard', and `unlimited' stand for the
current soft limit, the current hard limit, and no limit, respectively.
Otherwise, the current value of the specified resource is printed. If
no option is given, then -f is assumed.
Values are in 1024-byte increments, except for -t, which is in seconds,
-p, which is in increments of 512 bytes, and -u, which is an unscaled
number of processes.
Exit Status:
Returns success unless an invalid option is supplied or an error occurs.
```

Groupes de Contrôle

Les **Groupes de Contrôles** (*Control Groups*) aussi appelés **CGroups**, sont une nouvelle façon sous RHEL/CentOS 7 de contrôler et de limiter des ressources. Les groupes de contrôle permettent l'allocation de ressources, même d'une manière dynamique pendant que le système fonctionne, telles le temps processeur, la mémoire système, la bande réseau, ou une combinaison de ces ressources parmi des groupes de tâches (processus) définis par l'utilisateur et exécutés sur un système.

Les groupes de contrôle sont organisés de manière hiérarchique, comme des processus. Par contre, la comparaison entre les deux démontre que tandis que les processus se trouvent dans une arborescence unique descendant tous du processus init et héritant de l'environnement de leurs parents, les contrôles groupes peuvent être multiples donnant lieu à des arborescences ou **hiérarchies** multiples qui héritent de certains attributs de leurs groupes de contrôle parents.

Ces hiérarchies multiples et séparés sont nécessaires parce que chaque hiérarchie est attaché à un ou plusieurs **sous-système(s)** aussi appelés des **Contrôleurs de Ressources** ou simplement des **Contrôleurs**. Les contrôleurs disponibles sous RHEL/CentOS 7 sont :

- **blkio** - utilisé pour établir des limites sur l'accès des entrées/sorties à partir et depuis des périphériques blocs,
- **cpu** - utilisé pour fournir aux tâches des groupes de contrôle accès au CPU grâce au planificateur,
- **cpuacct** - utilisé pour produire des rapports automatiques sur les ressources CPU utilisées par les tâches dans un groupe de contrôle,
- **cpuset** - utilisé pour assigner des CPU individuels sur un système multicoeur et des noeuds de mémoire à des tâches dans un groupe de contrôle,
- **devices** - utilisé pour autoriser ou pour refuser l'accès des tâches aux périphériques dans un groupe de contrôle,
- **freezer** - utilisé pour suspendre ou pour réactiver les tâches dans un groupe de contrôle,

- **memory** - utilisé pour établir les limites d'utilisation de la mémoire par les tâches d'un groupe de contrôle et pour générer des rapports automatiques sur les ressources mémoire utilisées par ces tâches,
- **net_cls** - utilisé pour repérer les paquets réseau avec un identifiant de classe (*classid*) afin de permettre au contrôleur de trafic Linux, **tc**, d'identifier les paquets provenant d'une tâche particulière d'un groupe de contrôle.
- **perf_event** - utilisé pour permettre le monitoring des CGroups avec l'outil perf,
- **hugetlb** - utilisé pour limiter des ressources sur des pages de mémoire virtuelle de grande taille.

Pour visualiser les hiérarchies, il convient d'utiliser la commande suivante :

```
[root@centos7 ~]# lssubsys -am
cpuset /sys/fs/cgroup/cpuset
cpu,cpuacct /sys/fs/cgroup/cpu,cpuacct
memory /sys/fs/cgroup/memory
devices /sys/fs/cgroup/devices
freezer /sys/fs/cgroup/freezer
net_cls /sys/fs/cgroup/net_cls
blkio /sys/fs/cgroup/blkio
perf_event /sys/fs/cgroup/perf_event
hugetlb /sys/fs/cgroup/hugetlb
```

Sous RHEL/CentOS 7, **Systemd** organise les processus dans chaque CGroup. Par exemple tous les processus démarrés par le serveur Apache se trouveront dans le même CGroup, y compris les scripts CGI. Ceci implique que la gestion des ressources en utilisant des hiérarchies est couplé avec l'arborescence des unités de Systemd.

En haut de l'arborescence des unités de Systemd se trouve la tranche root - **-.slice**, dont dépend :

- le **system.slice** - l'emplacement des services système,
- le **user.slice** - l'emplacement des sessions des utilisateurs,
- le **machine.slice** - l'emplacement des machines virtuelles et conteneurs.

En dessous des tranches peuvent se trouver :

- des **scopes** - des processus créés par **fork**,
- des **services** - des processus créés par une **Unité**.

Les slices peuvent être visualisés avec la commande suivante :

```
[root@centos7 ~]# systemctl list-units --type=slice
UNIT                                LOAD    ACTIVE SUB    DESCRIPTION
-.slice                            loaded active active Root Slice
system-getty.slice                loaded active active system-getty.slice
system.slice                      loaded active active System Slice
user-0.slice                      loaded active active user-0.slice
user-1000.slice                   loaded active active user-1000.slice
user.slice                        loaded active active User and Session Slice
```

LOAD = Reflects whether the unit definition was properly loaded.

ACTIVE = The high-level unit activation state, i.e. generalization of SUB.

SUB = The low-level unit activation state, values depend on unit type.

6 loaded units listed. Pass --all to see loaded but inactive units, too.

To show all installed unit files use 'systemctl list-unit-files'.

L'arborescence des unités de Systemd est la suivante :

```
[root@centos7 ~]# systemd-cgls
├─1 /usr/lib/systemd/systemd --switched-root --system --deserialize 21
├─user.slice
│   └─user-1000.slice
│       └─session-2.scope
│           ├── 5577 sshd: trainee [priv]
│           ├── 6004 sshd: trainee@pts/0
│           ├── 6167 -bash
│           ├── 6217 su -
│           ├── 6245 -bash
│           ├── 13457 systemd-cgls
│           └─13459 systemd-cgls
└─system.slice
    └─upower.service
```

```

└─3478 /usr/libexec/upowerd
└─polkit.service
└─822 /usr/lib/polkit-1/polkitd --no-debug
└─wpa_supplicant.service
└─821 /usr/sbin/wpa_supplicant -u -f /var/log/wpa_supplicant.log -c /etc/wpa_supplicant/wpa_supplicant.conf -
u -f /var/log/wpa_supplicant.log -P /var/run/wpa_suppli
└─crond.service
└─793 /usr/sbin/crond -n
└─atd.service
└─789 /usr/sbin/atd -f
└─tuned.service
└─762 /usr/bin/python -Es /usr/sbin/tuned -l -P
└─simplegateway.service
└─760 /bin/sh /opt/JWrapper-Remote Access/JWAppsSharedConfig/SimpleGatewayService/service_launch.sh
└─3202 /opt/JWrapper-Remote Access/JWrapper-Linux64JRE-00028603412-complete/bin/Remote Access -cp
/opt/JWrapper-Remote Access/JWrapper-JWrapper-00041369502-complet
└─3384 /opt/JWrapper-Remote Access/JWrapper-Linux64JRE-00028603412-complete/bin/Remote Access Monitoring -cp
/opt/JWrapper-Remote Access/JWrapper-JWrapper-00041369
└─13111 /bin/sh /opt/JWrapper-Remote Access/JWAppsSharedConfig/SimpleGatewayService/service_launch.sh
└─13458 sleep 1
└─postfix.service
└─1810 /usr/libexec/postfix/master -w
└─1833 pickup -l -t unix -u
└─1834 qmgr -l -t unix -u
└─cups.service
└─756 /usr/sbin/cupsd -f
└─sshd.service
└─755 /usr/sbin/sshd -D
└─docker.service
└─750 /bin/sh -c /usr/bin/docker-current daemon $OPTIONS $DOCKER_STORAGE_OPTIONS
$DOCKER_NETWORK_OPTIONS $ADD_REGISTRY $
└─753 /usr/bin/docker-current daemon --selinux-enabled
└─754 /usr/bin/forward-journald -tag docker
└─NetworkManager.service

```

```
| | 678 /usr/sbin/NetworkManager --no-daemon
| | 1968 /sbin/dhclient -d -q -sf /usr/libexec/nm-dhcp-helper -pf /var/run/dhclient-enp0s3.pid -lf
/var/lib/NetworkManager/dhclient-3b386b69-23e8-4940-80e0-e16d346d43
| | abrt-xorg.service
| |   586 /usr/bin/abrt-watch-log -F Backtrace /var/log/Xorg.0.log -- /usr/bin/abrt-dump-xorg -xD
| | abrt-oops.service
| |   582 /usr/bin/abrt-watch-log -F BUG: WARNING: at WARNING: CPU: INFO: possible recursive locking detected
ernel BUG at list_del corruption list_add corruption do_IR
| | abrttd.service
| |   581 /usr/sbin/abrttd -d -s
| | dbus.service
| |   513 /bin/dbus-daemon --system --address=systemd: --nofork --nopidfile --systemd-activation
| | chronyd.service
| |   525 /usr/sbin/chronyd
| | firewalld.service
| |   510 /usr/bin/python -Es /usr/sbin/firewalld --nofork --nopid
| | avahi-daemon.service
| |   507 avahi-daemon: running [centos7.local]
| |   530 avahi-daemon: chroot help
| | ModemManager.service
| |   506 /usr/sbin/ModemManager
| | smartd.service
| |   505 /usr/sbin/smartd -n -q never
| | libstoragemgmt.service
| |   494 /usr/bin/lsmd -d
| | gssproxy.service
| |   517 /usr/sbin/gssproxy -D
| | systemd-logind.service
| |   492 /usr/lib/systemd/systemd-logind
| | rsyslog.service
| |   488 /usr/sbin/rsyslogd -n
| | alsa-state.service
| |   487 /usr/sbin/alsactl -s -n 19 -c -E ALSA_CONFIG_PATH=/etc/alsa/alsactl.conf --
initfile=/lib/alsa/init/00main rdaemon
```

```
└─auditd.service
  └─463 /sbin/auditd -n
  └─473 /sbin/audispd
  └─475 /usr/sbin/sedispach
└─systemd-udevd.service
  └─381 /usr/lib/systemd/systemd-udevd
└─lvm2-lvmetad.service
  └─378 /usr/sbin/lvmetad -f
└─systemd-journald.service
  └─349 /usr/lib/systemd/systemd-journald
└─system-getty.slice
  └─getty@tty1.service
    └─798 /sbin/agetty --noclear tty1 linux
```

En utilisant Systemd, plusieurs ressources peuvent être limitées :

- **CPUShares** - par défaut 1024,
- **MemoryLimit** - limite exprimée en Mo ou en Go. Pas de valeur par défaut,
- **BlockIOWeight** - valeur entre 10 et 1000. Pas de valeur par défaut,
- **StartupCPUShares** - comme CPUShares mais uniquement appliqué pendant le démarrage,
- **StartupBlockIOWeight** - comme BlockIOWeight mais uniquement appliqué pendant le démarrage,
- **CPUQuota** - utilisé pour limiter le temps CPU, même quand le système ne fait rien.



Important : Consultez le manuel `systemd.resource-control(5)` pour voir les paramètres CGroup qui peuvent être passés à `systemctl`.

LAB #1 - Travailler avec les cgroups sous RHEL/CentOS 7

Créez un service appelé **foo** :

```
[root@centos7 ~]# vi /etc/systemd/system/foo.service
```

```
[root@centos7 ~]# cat /etc/systemd/system/foo.service
[Unit]
Description=The foo service that does nothing useful
After=remote-fs.target nss-lookup.target

[Service]
ExecStart=/usr/bin/shasum /dev/zero
ExecStop=/bin/kill -WINCH ${MAINPID}

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

Consultez le statut du service foo :

```
[root@centos7 ~]# systemctl status foo.service
● foo.service - The foo service that does nothing useful
   Loaded: loaded (/etc/systemd/system/foo.service; disabled; vendor preset: disabled)
   Active: inactive (dead)
```

Démarrez et activez le service :

```
[root@centos7 ~]# systemctl start foo.service
[root@centos7 ~]# systemctl enable foo.service
Created symlink from /etc/systemd/system/multi-user.target.wants/foo.service to /etc/systemd/system/foo.service.
[root@centos7 ~]# systemctl status foo.service
● foo.service - The foo service that does nothing useful
   Loaded: loaded (/etc/systemd/system/foo.service; enabled; vendor preset: disabled)
   Active: active (running) since Wed 2016-06-15 13:13:17 CEST; 24s ago
 Main PID: 22256 (shasum)
    CGroup: /system.slice/foo.service
            └─22256 /usr/bin/shasum /dev/zero

Jun 15 13:13:17 centos7.fenestros.loc systemd[1]: Started The foo service that does nothing useful.
```

```
Jun 15 13:13:17 centos7.fenestros.loc systemd[1]: Starting The foo service that does nothing useful...
```

Notez que notre service a été placé dans la tranche **system.slice** :

```
[root@centos7 ~]# systemctl show -p Slice foo.service
Slice=system.slice
```

Utilisez ps pour voir le pourcentage du CPU utilisé par ce service :

```
[root@centos7 ~]# ps -p 22256 -o pid,comm,cputime,%cpu
  PID COMMAND          TIME %CPU
22256 shalsum           00:00:46 94.8
```

Fixez maintenant la valeur de CPUShares pour ce service à 250 :

```
[root@centos7 ~]# systemctl set-property foo.service CPUShares=250
```

Cette limite est permanente et a été inscrite dans le fichier **50-CPUShares.conf** qui se trouve dans le répertoire **/etc/systemd/system/foo.service.d** :

```
[root@centos7 ~]# ls /etc/systemd/system/foo.service.d
50-CPUShares.conf
[root@centos7 ~]# cat /etc/systemd/system/foo.service.d/50-CPUShares.conf
[Service]
CPUShares=250
```



Important : En utilisant l'option **-runtime** avec la commande **systemctl set-property** il est possible d'appliquer la limite d'une manière provisoire.

Appliquez cette modification en rechargeant systemd et en re-démarrant le service foo.service :

```
[root@centos7 ~]# systemctl daemon-reload
[root@centos7 ~]# systemctl restart foo.service
```

Vérifiez maintenant que la limite a été appliquée :

```
[root@centos7 ~]# cat /sys/fs/cgroup/cpu/system.slice/foo.service/cpu.shares
250
[root@centos7 ~]# systemctl show -p MainPID foo.service
MainPID=27233
[root@centos7 ~]# cat /proc/27233/cgroup | grep foo
4:cpuacct,cpu:/system.slice/foo.service
1:name=systemd:/system.slice/foo.service
```

Créez maintenant le service **bar** :

```
[root@centos7 ~]# vi /etc/systemd/system/bar.service
[root@centos7 ~]# cat /etc/systemd/system/bar.service
[Unit]
Description=The bar service that does nothing useful
After=remote-fs.target nss-lookup.target

[Service]
ExecStart=/usr/bin/md5sum /dev/zero
ExecStop=/bin/kill -WINCH ${MAINPID}

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

Fixez maintenant la limite de CPUShares pour ce service à 2000 :

```
[root@centos7 ~]# systemctl set-property bar.service CPUShares=2000
Failed to set unit properties on bar.service: Unit bar.service is not loaded.
[root@centos7 ~]# systemctl start bar.service
```

```
[root@centos7 ~]# systemctl enable bar.service
Created symlink from /etc/systemd/system/multi-user.target.wants/bar.service to /etc/systemd/system/bar.service.
[root@centos7 ~]# systemctl set-property bar.service CPUShares=2000
```

Appliquez la limite :

```
[root@centos7 ~]# systemctl daemon-reload
[root@centos7 ~]# systemctl restart bar.service
[root@centos7 ~]# systemctl status bar.service
● bar.service - The bar service that does nothing useful
   Loaded: loaded (/etc/systemd/system/bar.service; enabled; vendor preset: disabled)
   Drop-In: /etc/systemd/system/bar.service.d
            └─50-CPUShares.conf
   Active: active (running) since Wed 2016-06-15 13:37:54 CEST; 9s ago
 Main PID: 29515 (md5sum)
    CGroup: /system.slice/bar.service
            └─29515 /usr/bin/md5sum /dev/zero

Jun 15 13:37:54 centos7.fenestros.loc systemd[1]: Started The bar service that does nothing useful.
Jun 15 13:37:54 centos7.fenestros.loc systemd[1]: Starting The bar service that does nothing useful...
```

Re-démarrer les services foo et bar :

```
[root@centos7 ~]# systemctl restart foo.service
[root@centos7 ~]# systemctl status foo.service
● foo.service - The foo service that does nothing useful
   Loaded: loaded (/etc/systemd/system/foo.service; enabled; vendor preset: disabled)
   Drop-In: /etc/systemd/system/foo.service.d
            └─50-CPUShares.conf
   Active: active (running) since Wed 2016-06-15 13:50:08 CEST; 11s ago
 Main PID: 652 (shasum)
    CGroup: /system.slice/foo.service
            └─652 /usr/bin/shasum /dev/zero
```

```
Jun 15 13:50:08 centos7.fenestros.loc systemd[1]: Started The foo service that does nothing useful.
Jun 15 13:50:08 centos7.fenestros.loc systemd[1]: Starting The foo service that does nothing useful...
[root@centos7 ~]# systemctl restart bar.service
[root@centos7 ~]# systemctl status bar.service
● bar.service - The bar service that does nothing useful
   Loaded: loaded (/etc/systemd/system/bar.service; enabled; vendor preset: disabled)
   Drop-In: /etc/systemd/system/bar.service.d
            └─50-CPUShares.conf
   Active: active (running) since Wed 2016-06-15 13:50:38 CEST; 12s ago
 Main PID: 810 (md5sum)
   CGroup: /system.slice/bar.service
           └─810 /usr/bin/md5sum /dev/zero

Jun 15 13:50:38 centos7.fenestros.loc systemd[1]: Started The bar service that does nothing useful.
Jun 15 13:50:38 centos7.fenestros.loc systemd[1]: Starting The bar service that does nothing useful...
```

Utilisez ps pour voir le pourcentage du CPU utilisé par les deux services :

```
[root@centos7 ~]# ps -p 652,810 -o pid,comm,cputime,%cpu
  PID COMMAND          TIME %CPU
  652 sh1sum            00:00:08  9.7
  810 md5sum            00:00:45 78.9
```