

Version : **2022.01**

Dernière mise-à-jour : 2022/05/04 09:35

LDF601 - Gestion des Paramètres du matériel et les Ressources

Contenu du Module

- **LDF601 - Gestion des Paramètres et les Ressources du Matériel**

- Contenu du Module
- Présentation des Fichiers Spéciaux
- LAB #1 - Commandes
 - 1.1 - La Commande lspci
 - 1.2 - La Commande lsusb
 - 1.3 - La Commande lsblk
 - 1.4 - La Commande dmidecode
- LAB #2 - La Commande sysctl
 - 2.1 - Répertoire /proc
 - 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

- Répertoires
 - ide/scsi
 - acpi
 - bus
 - net
 - sys
- 2.2 - Utilisation de la Commande sysctl
- LAB #3 - Interprétation des informations dans /proc
 - 3.1 - free
 - 3.2 - uptime ou w
 - 3.3 - iostat
 - 3.4 - hdparm
 - 3.5 - vmstat
 - 3.6 - mpstat
 - 3.7 - sar
- Modules usb
- udev
 - La Commande udevadm
- Système de fichiers /sys
- LAB #4 - Limiter les Ressources
 - 4.1 - ulimit
 - 4.2 - Groupes de Contrôle
 - CGroups v1
 - Préparation
 - Présentation
 - Limitation de la Mémoire
 - La Commande cgcreate
 - La Commande cgdelete
 - Le Fichier /etc/cgconfig.conf
 - La Commande cgconfigparser

Présentation des 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@debian11:~# ls -l /dev | more
total 0
crw-r--r--  1 root root      10, 235 Apr 26 13:08 autofs
drwxr-xr-x  2 root root      480 Apr 28 06:26 block
drwxr-xr-x  2 root root      120 Apr 26 13:08 bsg
crw-rw----  1 root disk    10, 234 Apr 28 05:31 btrfs-control
drwxr-xr-x  3 root root       60 Apr 26 13:08 bus
lrwxrwxrwx  1 root root       3 Apr 26 13:08 cdrom -> sr0
drwxr-xr-x  2 root root    2800 Apr 28 06:02 char
crw--w----  1 root tty       5,   1 Apr 30 12:19 console
lrwxrwxrwx  1 root root      11 Apr 26 13:08 core -> /proc/kcore
crw-----  1 root root     10,  62 Apr 26 13:08 cpu_dma_latency
crw-----  1 root root     10, 203 Apr 26 13:08 cuse
drwxr-xr-x  7 root root      140 Apr 26 13:08 disk
brw-rw----  1 root disk   254,   0 Apr 27 15:21 dm-0
brw-rw----  1 root disk   254,   1 Apr 26 16:33 dm-1
brw-rw----  1 root disk   254,   2 Apr 28 06:27 dm-2
```

```

drwxr-xr-x 3 root root      80 Apr 26 13:08 dri
lrwxrwxrwx 1 root root      3 Apr 26 13:08 dvd -> sr0
crw----- 1 root root    10, 61 Apr 28 06:02 ecryptfs
crw-rw---- 1 root video  29,  0 Apr 26 13:08 fb0
lrwxrwxrwx 1 root root     13 Apr 26 13:08 fd -> /proc/self/fd
crw-rw-rw- 1 root root      1,  7 Apr 26 13:08 full
crw-rw-rw- 1 root root    10, 229 Apr 26 13:08 fuse
crw----- 1 root root   248,  0 Apr 26 13:08 hidraw0
crw----- 1 root root    10, 228 Apr 26 13:08 hpet
drwxr-xr-x 2 root root      0 Apr 26 13:08 hugepages
lrwxrwxrwx 1 root root     12 Apr 26 13:08 initctl -> /run/initctl
drwxr-xr-x 4 root root    280 Apr 26 13:08 input
crw-r--r-- 1 root root      1, 11 Apr 26 13:08 kmsg
lrwxrwxrwx 1 root root     28 Apr 26 13:08 log -> /run/systemd/journal/dev-log
crw-rw---- 1 root disk   10, 237 Apr 26 13:08 loop-control
drwxr-xr-x 2 root root    120 Apr 28 06:26 mapper
crw-r----- 1 root kmem      1,  1 Apr 26 13:08 mem
drwxrwxrwt 2 root root     40 Apr 26 13:08 mqueue
drwxr-xr-x 2 root root     60 Apr 26 13:08 net
crw-rw-rw- 1 root root      1,  3 Apr 26 13:08 null
crw----- 1 root root    10, 144 Apr 26 13:08 nvram
crw-r----- 1 root kmem      1,  4 Apr 26 13:08 port
crw----- 1 root root   108,  0 Apr 26 13:08 ppp
crw----- 1 root root    10,  1 Apr 26 13:08 psaux
crw-rw-rw- 1 root tty       5,  2 Apr 30 15:29 ptmx
drwxr-xr-x 2 root root      0 Apr 26 13:08 pts
crw-rw-rw- 1 root root      1,  8 Apr 26 13:08 random
crw-rw-r-- 1 root netdev   10, 242 Apr 26 13:08 rfkill
lrwxrwxrwx 1 root root      4 Apr 26 13:08 rtc -> rtc0
crw----- 1 root root   252,  0 Apr 26 13:08 rtc0
--More--
[q]

```

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,  33 Apr 26 13:46 sdc1  
...  
crw--w----  1 root tty       5,   1 Apr 30 12:19 console  
...
```

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.

LAB #1 - Commandes

1.1 - La Commande lspci

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

```
root@debian11:~# 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 82371SB PIIX3 IDE [Natoma/Triton II]  
00:01.2 USB controller: Intel Corporation 82371SB PIIX3 USB [Natoma/Triton II] (rev 01)  
00:01.3 Bridge: Intel Corporation 82371AB/EB/MB PIIX4 ACPI (rev 03)  
00:02.0 VGA compatible controller: Device 1234:1111 (rev 02)  
00:03.0 Unclassified device [00ff]: Red Hat, Inc. Virtio memory balloon  
00:05.0 SCSI storage controller: Red Hat, Inc. Virtio SCSI  
00:12.0 Ethernet controller: Red Hat, Inc. Virtio network device  
00:1e.0 PCI bridge: Red Hat, Inc. QEMU PCI-PCI bridge
```

00:1f.0 PCI bridge: Red Hat, Inc. QEMU PCI-PCI bridge

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@debian11:~# lspci -v -s 00:03.0
00:03.0 Unclassified device [00ff]: Red Hat, Inc. Virtio memory balloon
  Subsystem: Red Hat, Inc. Virtio memory balloon
  Physical Slot: 3
  Flags: bus master, fast devsel, latency 0, IRQ 10
  I/O ports at e000 [size=64]
  Memory at fe400000 (64-bit, prefetchable) [size=16K]
  Capabilities: [84] Vendor Specific Information: VirtIO: <unknown>
  Capabilities: [70] Vendor Specific Information: VirtIO: Notify
  Capabilities: [60] Vendor Specific Information: VirtIO: DeviceCfg
  Capabilities: [50] Vendor Specific Information: VirtIO: ISR
  Capabilities: [40] Vendor Specific Information: VirtIO: CommonCfg
  Kernel driver in use: virtio-pci
  Kernel modules: virtio_pci
```

ou :

```
root@debian11:~# lspci -vv -s 00:03.0
00:03.0 Unclassified device [00ff]: Red Hat, Inc. Virtio memory balloon
  Subsystem: Red Hat, Inc. Virtio memory balloon
  Physical Slot: 3
  Control: I/O+ Mem+ BusMaster+ SpecCycle- MemWINV- VGASnoop- ParErr- Stepping- SERR+ FastB2B- DisINTx-
  Status: Cap+ 66MHz- UDF- FastB2B- ParErr- DEVSEL=fast >TAbort- <TAbort- <MAbort- >SERR- <PERR- INTx-
  Latency: 0
  Interrupt: pin A routed to IRQ 10
  Region 0: I/O ports at e000 [size=64]
  Region 4: Memory at fe400000 (64-bit, prefetchable) [size=16K]
  Capabilities: [84] Vendor Specific Information: VirtIO: <unknown>
    BAR=0 offset=00000000 size=00000000
```

```
Capabilities: [70] Vendor Specific Information: VirtIO: Notify
      BAR=4 offset=00003000 size=00001000 multiplier=00000004
Capabilities: [60] Vendor Specific Information: VirtIO: DeviceCfg
      BAR=4 offset=00002000 size=00001000
Capabilities: [50] Vendor Specific Information: VirtIO: ISR
      BAR=4 offset=00001000 size=00001000
Capabilities: [40] Vendor Specific Information: VirtIO: CommonCfg
      BAR=4 offset=00000000 size=00001000
Kernel driver in use: virtio-pci
Kernel modules: virtio_pci
```

Les options de cette commande sont :

```
root@debian11:~# lspci --help
lspci: invalid option -- '-'
Usage: lspci [<switches>]
```

Basic display modes:

```
-mm          Produce machine-readable output (single -m for an obsolete format)
-t          Show bus tree
```

Display options:

```
-v          Be verbose (-vv or -vvv for higher verbosity)
-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
-P          Display bridge path in addition to bus and device number
-PP         Display bus path in addition to bus and device number
```

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>[:<class>]          Show only devices with specified ID's
```

Other options:

```
-i <file>      Use specified ID database instead of /usr/share/misc/pci.ids.gz
-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
```

1.2 - La Commande lsusb

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

```
root@debian11:~# lsusb
Bus 001 Device 002: ID 0627:0001 Adomax Technology Co., Ltd QEMU USB Tablet
Bus 001 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub

root@debian11:~# lsusb -vt
/:  Bus 01.Port 1: Dev 1, Class=root_hub, Driver=uhci_hcd/2p, 12M
    ID 1d6b:0001 Linux Foundation 1.1 root hub
    |__ Port 1: Dev 2, If 0, Class=Human Interface Device, Driver=usbhid, 12M
```

ID 0627:0001 Adomax Technology Co., Ltd

Les options de cette commande sont :

```
root@debian11:~# 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
```

1.3 - La Commande lsblk

Cette commande vous renseigne sur les périphériques de type bloc :

```
root@debian11:~# lsblk
NAME          MAJ:MIN RM  SIZE RO TYPE  MOUNTPOINT
sda           8:0    0   32G  0 disk
├─sda1        8:1    0   31G  0 part  /
└─sda2        8:2    0    1K  0 part
```

```
└─sda5      8:5    0  975M  0 part  [SWAP]
sdb         8:16    0   64G  0 disk
sdc         8:32    0    4G  0 disk
├─sdc1      8:33    0  100M  0 part
├─sdc2      8:34    0  100M  0 part
├─sdc3      8:35    0  100M  0 part
├─sdc4      8:36    0    1K  0 part
├─sdc5      8:37    0  500M  0 part
├─sdc6      8:38    0  200M  0 part
│   └─vg0-lv1 254:0    0  104M  0 lvm
├─sdc7      8:39    0  300M  0 part
│   └─vg0-lv2 254:1    0  112M  0 lvm
├─sdc8      8:40    0  500M  0 part
│   └─md1     9:1     0  996M  0 raid5
├─sdc9      8:41    0  400M  0 part
│   └─vg0-lv2 254:1    0  112M  0 lvm
├─sdc10     8:42    0  500M  0 part
│   └─md1     9:1     0  996M  0 raid5
├─sdc11     8:43    0  500M  0 part
│   └─md1     9:1     0  996M  0 raid5
└─sdc12     8:44    0  200M  0 part
sr0        11:0    1   378M  0 rom
```

Les options de cette commande sont :

```
root@debian11:~# lsblk --help
```

Usage:

```
lsblk [options] [<device> ...]
```

List information about block devices.

Options:

```
-D, --discard          print discard capabilities
```

```
-E, --dedup <column> de-duplicate output by <column>
-I, --include <list> show only devices with specified major numbers
-J, --json           use JSON output format
-O, --output-all    output all columns
-P, --pairs          use key="value" output format
-S, --scsi           output info about SCSI devices
-T, --tree[=<column>] use tree format output
-a, --all            print all devices
-b, --bytes          print SIZE in bytes rather than in human readable format
-d, --nodeps         don't print slaves or holders
-e, --exclude <list> exclude devices by major number (default: RAM disks)
-f, --fs             output info about filesystems
-i, --ascii          use ascii characters only
-l, --list           use list format output
-M, --merge          group parents of sub-trees (usable for RAID's, Multi-path)
-m, --perms          output info about permissions
-n, --noheadings     don't print headings
-o, --output <list>  output columns
-p, --paths          print complete device path
-r, --raw            use raw output format
-s, --inverse        inverse dependencies
-t, --topology       output info about topology
-z, --zoned          print zone model
-x, --sort <column>  sort output by <column>
    --sysroot <dir> use specified directory as system root

-h, --help           display this help
-V, --version        display version
```

Available output columns:

NAME	device name
KNAME	internal kernel device name
PATH	path to the device node
MAJ:MIN	major:minor device number

FSAVAIL	filesystem size available
FSSIZE	filesystem size
FSTYPE	filesystem type
FSUSED	filesystem size used
FSUSE%	filesystem use percentage
FSVER	filesystem version
MOUNTPPOINT	where the device is mounted
LABEL	filesystem LABEL
UUID	filesystem UUID
PTUUID	partition table identifier (usually UUID)
PTTYPE	partition table type
PARTTYPE	partition type code or UUID
PARTTYPENAME	partition type name
PARTLABEL	partition LABEL
PARTUUID	partition UUID
PARTFLAGS	partition flags
RA	read-ahead of the device
RO	read-only device
RM	removable device
HOTPLUG	removable or hotplug device (usb, pcmcia, ...)
MODEL	device identifier
SERIAL	disk serial number
SIZE	size of the device
STATE	state of the device
OWNER	user name
GROUP	group name
MODE	device node permissions
ALIGNMENT	alignment offset
MIN-IO	minimum I/O size
OPT-IO	optimal I/O size
PHY-SEC	physical sector size
LOG-SEC	logical sector size
ROTA	rotational device
SCHED	I/O scheduler name

```
RQ-SIZE request queue size
TYPE device type
DISC-ALN discard alignment offset
DISC-GRAN discard granularity
DISC-MAX discard max bytes
DISC-ZERO discard zeroes data
WSAME write same max bytes
WWN unique storage identifier
RAND adds randomness
PKNAME internal parent kernel device name
HCTL Host:Channel:Target:Lun for SCSI
TRAN device transport type
SUBSYSTEMS de-duplicated chain of subsystems
REV device revision
VENDOR device vendor
ZONED zone model
DAX dax-capable device
```

For more details see `lsblk(8)`.

1.4 - 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@debian11:~# dmidecode
# dmidecode 3.3
Getting SMBIOS data from sysfs.
SMBIOS 2.8 present.
10 structures occupying 443 bytes.
```

Table at 0x000F58C0.

Handle 0x0000, DMI type 0, 24 bytes

BIOS Information

Vendor: SeaBIOS

Version: rel-1.14.0-0-g155821a1990b-prebuilt.qemu.org

Release Date: 04/01/2014

Address: 0xE8000

Runtime Size: 96 kB

ROM Size: 64 kB

Characteristics:

BIOS characteristics not supported

Targeted content distribution is supported

BIOS Revision: 0.0

Handle 0x0100, DMI type 1, 27 bytes

System Information

Manufacturer: QEMU

Product Name: Standard PC (i440FX + PIIX, 1996)

Version: pc-i440fx-5.2

Serial Number: Not Specified

UUID: ce816e48-faa4-4cb9-a03e-f0f58040c52b

Wake-up Type: Power Switch

SKU Number: Not Specified

Family: Not Specified

Handle 0x0300, DMI type 3, 22 bytes

Chassis Information

Manufacturer: QEMU

Type: Other

Lock: Not Present

Version: pc-i440fx-5.2

Serial Number: Not Specified

Asset Tag: Not Specified

Boot-up State: Safe
Power Supply State: Safe
Thermal State: Safe
Security Status: Unknown
OEM Information: 0x00000000
Height: Unspecified
Number Of Power Cords: Unspecified
Contained Elements: 0
SKU Number: Not Specified

Handle 0x0400, DMI type 4, 42 bytes

Processor Information

Socket Designation: CPU 0
Type: Central Processor
Family: Other
Manufacturer: QEMU
ID: 61 0F 00 00 FF FB 8B 07
Version: pc-i440fx-5.2
Voltage: Unknown
External Clock: Unknown
Max Speed: 2000 MHz
Current Speed: 2000 MHz
Status: Populated, Enabled
Upgrade: Other
L1 Cache Handle: Not Provided
L2 Cache Handle: Not Provided
L3 Cache Handle: Not Provided
Serial Number: Not Specified
Asset Tag: Not Specified
Part Number: Not Specified
Core Count: 2
Core Enabled: 2
Thread Count: 1
Characteristics: None

Handle 0x1000, DMI type 16, 23 bytes

Physical Memory Array

Location: Other

Use: System Memory

Error Correction Type: Multi-bit ECC

Maximum Capacity: 4 GB

Error Information Handle: Not Provided

Number Of Devices: 1

Handle 0x1100, DMI type 17, 40 bytes

Memory Device

Array Handle: 0x1000

Error Information Handle: Not Provided

Total Width: Unknown

Data Width: Unknown

Size: 4 GB

Form Factor: DIMM

Set: None

Locator: DIMM 0

Bank Locator: Not Specified

Type: RAM

Type Detail: Other

Speed: Unknown

Manufacturer: QEMU

Serial Number: Not Specified

Asset Tag: Not Specified

Part Number: Not Specified

Rank: Unknown

Configured Memory Speed: Unknown

Minimum Voltage: Unknown

Maximum Voltage: Unknown

Configured Voltage: Unknown

Handle 0x1300, DMI type 19, 31 bytes

```
Memory Array Mapped Address
  Starting Address: 0x000000000000
  Ending Address: 0x000BFFFFFFF
  Range Size: 3 GB
  Physical Array Handle: 0x1000
  Partition Width: 1
```

```
Handle 0x1301, DMI type 19, 31 bytes
Memory Array Mapped Address
  Starting Address: 0x001000000000
  Ending Address: 0x0013FFFFFFF
  Range Size: 1 GB
  Physical Array Handle: 0x1000
  Partition Width: 1
```

```
Handle 0x2000, DMI type 32, 11 bytes
System Boot Information
  Status: No errors detected
```

```
Handle 0x7F00, DMI type 127, 4 bytes
End Of Table
```

Les options de cette commande sont :

```
root@debian11:~# 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
  -H, --handle HANDLE     Only display the entry of given handle
  -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
--no-sysfs         Do not attempt to read DMI data from sysfs files
--oem-string N     Only display the value of the given OEM string
-V, --version      Display the version and exit
```

LAB #2 - La commande sysctl

2.1 - 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@debian11:~# ls /proc
1      128  24      34952  46798  54400  55364  56363  57690  68      acpi      dynamic_debug  key-users  mtrr
sys
10     13   244     4       46804  54401  55367  56440  57691  71      buddyinfo  execdomains    kmsg       net
sysrq-trigger
11     15   25      42378  46805  54405  55368  56894  57772  72      bus        fb              kpagecgroup
pagetypeinfo sysvipc
116    16   27      42390  46806  54406  55385  56910  57773  7576    cgroups    filesystems     kpagecount
partitions thread-self
119    17   28      42391  46808  54407  55392  56911  57781  7708    cmdline    fs              kpageflags  pressure
timer_list
12     176  287     42392  46812  54408  55549  56912  57785  7709    consoles  interrupts      loadavg
sched_debug tty
120    177  29      42401  46813  55      55563  57      57792  7883    cpuinfo     iomem           locks       schedstat
uptime
121    18   3       42402  50      55345  55565  57148  57798  7894    crypto      ioports         mdstat      self
version
122    2    30      46777  51      55347  55569  57149  57800  8       devices     irq             meminfo     slabinfo
```

vmallocinfo													
123	20	31	46780	52	55348	55571	57150	58	8904	diskstats	kallsyms	misc	softirqs
vmstat													
126	214	32	46782	53	55350	56	57308	59	8905	dma	kcore	modules	stat
zoneinfo													
127	23	34924	46797	54	55353	56135	57689	6	9	driver	keys	mounts	swaps

Fichiers

Processeur

```
root@debian11:~# cat /proc/cpuinfo
processor       : 0
vendor_id      : GenuineIntel
cpu family     : 15
model          : 6
model name     : Common KVM processor
stepping       : 1
microcode      : 0x1
cpu MHz        : 2399.982
cache size     : 16384 KB
physical id    : 0
siblings       : 2
core id        : 0
cpu cores      : 2
apicid         : 0
initial apicid : 0
fpu            : yes
fpu_exception  : yes
cpuid level    : 13
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 ht syscall nx lm constant_tsc nopl xtopology cpuid tsc_known_freq pni cx16 x2apic hypervisor lahf_lm
cpuid_fault pti
bugs          : cpu_meltdown spectre_v1 spectre_v2 spec_store_bypass l1tf mds swapgs itlb_multihit
bogomips      : 4799.96
clflush size  : 64
cache_alignment : 128
address sizes  : 40 bits physical, 48 bits virtual
power management:

processor      : 1
vendor_id     : GenuineIntel
cpu family    : 15
model         : 6
model name    : Common KVM processor
stepping      : 1
microcode     : 0x1
cpu MHz       : 2399.982
cache size    : 16384 KB
physical id   : 0
siblings      : 2
core id       : 1
cpu cores     : 2
apicid        : 1
initial apicid : 1
fpu           : yes
fpu_exception : yes
cpuid level   : 13
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 ht syscall nx lm constant_tsc nopl xtopology cpuid tsc_known_freq pni cx16 x2apic hypervisor lahf_lm
cpuid_fault pti
bugs          : cpu_meltdown spectre_v1 spectre_v2 spec_store_bypass l1tf mds swapgs itlb_multihit
bogomips      : 4799.96
clflush size  : 64
```

```
cache_alignment : 128
address sizes   : 40 bits physical, 48 bits virtual
power management:
```

Interruptions système

```
root@debian11:~# cat /proc/interrupts
          CPU0           CPU1
 0:         29             0   IO-APIC  2-edge    timer
 1:          0             9   IO-APIC  1-edge    i8042
 6:          0             3   IO-APIC  6-edge    floppy
 8:          1             0   IO-APIC  8-edge    rtc0
 9:          0             0   IO-APIC  9-fasteoi  acpi
10:          0       177277   IO-APIC 10-fasteoi  virtio0
11:         40             0   IO-APIC 11-fasteoi  uhci_hcd:usb1
12:         15             0   IO-APIC 12-edge    i8042
14:          0             0   IO-APIC 14-edge    ata_piix
15:          0       352265   IO-APIC 15-edge    ata_piix
24:          0             0   PCI-MSI 294912-edge  virtio2-config
25:       24163             0   PCI-MSI 294913-edge  virtio2-input.0
26:          0       18954   PCI-MSI 294914-edge  virtio2-output.0
27:          0             0   PCI-MSI 81920-edge  virtio1-config
28:          0             0   PCI-MSI 81921-edge  virtio1-control
29:          0             0   PCI-MSI 81922-edge  virtio1-event
30:       74023             0   PCI-MSI 81923-edge  virtio1-request
31:          0       71408   PCI-MSI 81924-edge  virtio1-request
NMI:          0             0   Non-maskable interrupts
LOC:    1701175       2400564   Local timer interrupts
SPU:          0             0   Spurious interrupts
PMI:          0             0   Performance monitoring interrupts
IWI:          1             0   IRQ work interrupts
RTR:          0             0   APIC ICR read retries
RES:    284026       248715   Rescheduling interrupts
```

CAL:	47419	21994	Function call interrupts
TLB:	23047	13223	TLB shutdowns
TRM:	0	0	Thermal event interrupts
THR:	0	0	Threshold APIC interrupts
DFR:	0	0	Deferred Error APIC interrupts
MCE:	0	0	Machine check exceptions
MCP:	1140	1140	Machine check polls
ERR:	0		
MIS:	0		
PIN:	0	0	Posted-interrupt notification event
NPI:	0	0	Nested posted-interrupt event
PIW:	0	0	Posted-interrupt wakeup event



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@debian11:~# cat /proc/dma
2: floppy
4: cascade
```

Plages d'entrée/sortie

```
root@debian11:~# cat /proc/ioports
```

```
0000-0cf7 : PCI Bus 0000:00
 0000-001f : dma1
 0020-0021 : pic1
 0040-0043 : timer0
 0050-0053 : timer1
 0060-0060 : keyboard
 0064-0064 : keyboard
 0070-0077 : 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+
 03f2-03f2 : floppy
 03f4-03f5 : floppy
 03f6-03f6 : 0000:00:01.1
   03f6-03f6 : ata_piix
 03f7-03f7 : floppy
 0510-051b : QEMU0002:00
   0510-051b : fw_cfg_io
 0600-063f : 0000:00:01.3
   0600-0603 : ACPI PM1a_EVT_BLK
   0604-0605 : ACPI PM1a_CNT_BLK
   0608-060b : ACPI PM_TMR
 0700-070f : 0000:00:01.3
   0700-0708 : piix4_smbus
0cf8-0cff : PCI conf1
0d00-ffff : PCI Bus 0000:00
```

```
afe0-afe3 : ACPI GPE0_BLK
c000-cfff : PCI Bus 0000:02
d000-dfff : PCI Bus 0000:01
e000-e03f : 0000:00:03.0
e040-e07f : 0000:00:05.0
e080-e09f : 0000:00:01.2
    e080-e09f : uhci_hcd
e0a0-e0bf : 0000:00:12.0
e0c0-e0cf : 0000:00:01.1
    e0c0-e0cf : ata_piix
```



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

Périphériques

```
root@debian11:~# 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
21 sg
```

```
29 fb
99 ppdev
128 ptm
136 pts
153 spi
180 usb
189 usb_device
226 drm
248 hidraw
249 aux
250 cec
251 bsg
252 rtc
253 dax
254 gpiochip
```

Block devices:

```
2 fd
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 mdp
254 device-mapper
259 blkext
```

Modules

```
root@debian11:~# cat /proc/modules | more
ufs 94208 0 - Live 0xffffffffc0cde000
qnx4 16384 0 - Live 0xffffffffc07a3000
hfsplus 126976 0 - Live 0xffffffffc0d2b000
hfs 73728 0 - Live 0xffffffffc0cc4000
minix 45056 0 - Live 0xffffffffc0cb8000
vfat 20480 0 - Live 0xffffffffc0cb2000
msdos 20480 0 - Live 0xffffffffc0cac000
fat 86016 2 vfat,msdos, Live 0xffffffffc0c8f000
jfs 212992 0 - Live 0xffffffffc0cf6000
xfs 1773568 0 - Live 0xffffffffc0add000
xts 16384 1 - Live 0xffffffffc0ad2000
dm_crypt 53248 1 - Live 0xffffffffc0ac4000
cbc 16384 0 - Live 0xffffffffc0abf000
aes_generic 36864 3 - Live 0xffffffffc0ab5000
libaes 16384 1 aes_generic, Live 0xffffffffc0ab0000
crypto_simd 16384 0 - Live 0xffffffffc0aab000
cryptd 24576 1 crypto_simd, Live 0xffffffffc0a9f000
glue_helper 16384 0 - Live 0xffffffffc08dd000
ecb 16384 2 - Live 0xffffffffc0883000
ecryptfs 122880 0 - Live 0xffffffffc0a80000
btrfs 1568768 0 - Live 0xffffffffc0900000
blake2b_generic 20480 0 - Live 0xffffffffc087d000
raid1 53248 0 - Live 0xffffffffc08f2000
dm_raid 45056 0 - Live 0xffffffffc08e6000
```

```
raid456 180224 1 dm_raid, Live 0xfffffffffc08b0000
async_raid6_recov 24576 1 raid456, Live 0xfffffffffc08a7000
async_memcpy 20480 2 raid456,async_raid6_recov, Live 0xfffffffffc089f000
async_pq 20480 2 raid456,async_raid6_recov, Live 0xfffffffffc0899000
async_xor 20480 3 raid456,async_raid6_recov,async_pq, Live 0xfffffffffc0893000
async_tx 20480 5 raid456,async_raid6_recov,async_memcpy,async_pq,async_xor, Live 0xfffffffffc088a000
md_mod 180224 3 raid1,dm_raid,raid456, Live 0xfffffffffc0850000
xor 24576 2 btrfs,async_xor, Live 0xfffffffffc0845000
raid6_pq 122880 4 btrfs,raid456,async_raid6_recov,async_pq, Live 0xfffffffffc0822000
libcrc32c 16384 3 xfs,btrfs,raid456, Live 0xfffffffffc07b3000
dm_snapshot 53248 0 - Live 0xfffffffffc07b9000
dm_bufio 36864 1 dm_snapshot, Live 0xfffffffffc07a9000
dm_mod 163840 10 dm_crypt,dm_raid,dm_snapshot,dm_bufio, Live 0xfffffffffc07f6000
rfkill 28672 0 - Live 0xfffffffffc07cb000
joydev 28672 0 - Live 0xfffffffffc0725000
evdev 28672 2 - Live 0xfffffffffc071d000
virtio_balloon 24576 0 - Live 0xfffffffffc06f7000
sg 36864 0 - Live 0xfffffffffc06cc000
serio_raw 20480 0 - Live 0xfffffffffc06ba000
pcspkr 16384 0 - Live 0xfffffffffc06b0000
qemu_fw_cfg 20480 0 - Live 0xfffffffffc06aa000
parport_pc 40960 0 - Live 0xfffffffffc0679000
ppdev 24576 0 - Live 0xfffffffffc0664000
lp 20480 0 - Live 0xfffffffffc066b000
parport 69632 3 parport_pc,ppdev,lp, Live 0xfffffffffc0652000
fuse 167936 1 - Live 0xfffffffffc0628000
--More--
[q]
```

Statistiques de l'utilisation des disques

```
root@debian11:~# cat /proc/diskstats
11          0 sr0 77 0 4235 23 0 0 0 0 0 68 23 0 0 0 0 0 0
```

```

8      0 sda 12242 8305 1103196 42502 11683 13369 1910368 718254 0 105044 783887 0 0 0 0 2936 23131
8      1 sda1 11481 8305 1067918 41666 10597 13369 1910368 717761 0 103920 759427 0 0 0 0 0 0
8      2 sda2 11 0 14 38 0 0 0 0 0 64 38 0 0 0 0 0 0
8      5 sda5 201 0 14692 296 0 0 0 0 0 460 296 0 0 0 0 0 0
8     16 sdb 227 0 8976 27 0 0 0 0 0 72 27 0 0 0 0 0 0
8     32 sdc 5023 507625 4247435 13154 1528 253646 2060274 14491 0 12120 31223 0 0 0 0 141 3577
8     33 sdc1 144 0 9972 5 0 0 0 0 0 68 5 0 0 0 0 0 0
8     34 sdc2 138 0 9924 6 0 0 0 0 0 64 6 0 0 0 0 0 0
8     35 sdc3 138 0 9924 6 0 0 0 0 0 64 6 0 0 0 0 0 0
8     36 sdc4 24 0 566 43 0 0 0 0 0 124 43 0 0 0 0 0 0
8     37 sdc5 941 127001 1035648 1742 19 0 26 1281 0 3504 3024 0 0 0 0 0 0
8     38 sdc6 278 0 24508 243 0 0 0 0 0 352 243 0 0 0 0 0 0
8     39 sdc7 220 8 19832 268 0 0 0 0 0 376 268 0 0 0 0 0 0
8     40 sdc8 1539 253750 2054126 3152 41 0 48 3380 0 6968 6533 0 0 0 0 0 0
8     41 sdc9 142 4 19720 211 0 0 0 0 0 332 211 0 0 0 0 0 0
8     42 sdc10 807 126861 1033214 5955 734 126795 1019952 4901 0 8468 10857 0 0 0 0 0 0
8     43 sdc11 281 1 9545 835 715 126851 1040248 4581 0 5564 5417 0 0 0 0 0 0
8     44 sdc12 148 0 13300 287 0 0 0 0 0 368 287 0 0 0 0 0 0
254    0 dm-0 210 0 8440 72 0 0 0 0 0 108 72 0 0 0 0 0 0
254    1 dm-1 242 0 8440 164 0 0 0 0 0 168 164 0 0 0 0 0 0
9      1 md1 57 0 2304 0 0 0 0 0 0 0 0 0 0 0 0 0

```

Partitions

```

root@debian11:~# cat /proc/partitions
major minor #blocks name

```

```

11      0      387072 sr0
8       0    33554432 sda
8       1    32552960 sda1
8       2           1 sda2
8       5     998400 sda5
8      16    67108864 sdb

```

8	32	4194304	sd
8	33	102400	sd1
8	34	102400	sd2
8	35	102400	sd3
8	36	1	sd4
8	37	512000	sd5
8	38	204800	sd6
8	39	307200	sd7
8	40	512000	sd8
8	41	409600	sd9
8	42	512000	sd10
8	43	512000	sd11
8	44	204800	sd12
254	0	106496	dm-0
254	1	114688	dm-1
9	1	1019904	md1

Espaces de pagination

```
root@debian11:~# cat /proc/swaps
```

Filename	Type	Size	Used	Priority
/dev/sda5	partition	998396	0	-2

Statistiques d'utilisation du processeur

```
root@debian11:~# cat /proc/loadavg
```

```
0.00 0.00 0.00 1/292 57813
```

Statistiques d'utilisation de la mémoire

```
root@debian11:~# cat /proc/meminfo
MemTotal:      4025596 kB
MemFree:       2604608 kB
MemAvailable:  3626000 kB
Buffers:       28244 kB
Cached:        1159028 kB
SwapCached:    0 kB
Active:        447852 kB
Inactive:      807036 kB
Active(anon):   748 kB
Inactive(anon): 68892 kB
Active(file):   447104 kB
Inactive(file): 738144 kB
Unevictable:    0 kB
Mlocked:       0 kB
SwapTotal:     998396 kB
SwapFree:      998396 kB
Dirty:         0 kB
Writeback:     0 kB
AnonPages:     67648 kB
Mapped:        65484 kB
Shmem:         2024 kB
KReclaimable:  82492 kB
Slab:          120008 kB
SReclaimable:  82492 kB
SUnreclaim:    37516 kB
KernelStack:   4688 kB
PageTables:    3504 kB
NFS_Unstable:  0 kB
Bounce:        0 kB
WritebackTmp:  0 kB
CommitLimit:   3011192 kB
Committed_AS:  1645372 kB
VmallocTotal:  34359738367 kB
```

```
VmallocUsed:      27688 kB
VmallocChunk:      0 kB
Percpu:           2080 kB
HardwareCorrupted: 0 kB
AnonHugePages:    26624 kB
ShmemHugePages:    0 kB
ShmemPmdMapped:    0 kB
FileHugePages:     0 kB
FilePmdMapped:     0 kB
HugePages_Total:   0
HugePages_Free:    0
HugePages_Rsvd:    0
HugePages_Surp:    0
Hugepagesize:      2048 kB
Hugetlb:           0 kB
DirectMap4k:       114536 kB
DirectMap2M:       4079616 kB
```

Version du noyau

```
root@debian11:~# cat /proc/version
Linux version 5.10.0-13-amd64 (debian-kernel@lists.debian.org) (gcc-10 (Debian 10.2.1-6) 10.2.1 20210110, GNU ld
(GNU Binutils for Debian) 2.35.2) #1 SMP Debian 5.10.106-1 (2022-03-17)
```

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.

2.2 - Utilisation de 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@debian11:~# cat /etc/sysctl.conf
#
# /etc/sysctl.conf - Configuration file for setting system variables
# See /etc/sysctl.d/ for additional system variables.
# See sysctl.conf (5) for information.
#

#kernel.domainname = example.com

# Uncomment the following to stop low-level messages on console
#kernel.printk = 3 4 1 3

#####
# Functions previously found in netbase
#

# Uncomment the next two lines to enable Spoof protection (reverse-path filter)
# Turn on Source Address Verification in all interfaces to
# prevent some spoofing attacks
#net.ipv4.conf.default.rp_filter=1
#net.ipv4.conf.all.rp_filter=1

# Uncomment the next line to enable TCP/IP SYN cookies
# See http://lwn.net/Articles/277146/
# Note: This may impact IPv6 TCP sessions too
#net.ipv4.tcp_syncookies=1

# Uncomment the next line to enable packet forwarding for IPv4
#net.ipv4.ip_forward=1

# Uncomment the next line to enable packet forwarding for IPv6
# Enabling this option disables Stateless Address Autoconfiguration
# based on Router Advertisements for this host
#net.ipv6.conf.all.forwarding=1
```

```
#####  
# Additional settings - these settings can improve the network  
# security of the host and prevent against some network attacks  
# including spoofing attacks and man in the middle attacks through  
# redirection. Some network environments, however, require that these  
# settings are disabled so review and enable them as needed.  
#  
# Do not accept ICMP redirects (prevent MITM attacks)  
#net.ipv4.conf.all.accept_redirects = 0  
#net.ipv6.conf.all.accept_redirects = 0  
# _or_  
# Accept ICMP redirects only for gateways listed in our default  
# gateway list (enabled by default)  
# net.ipv4.conf.all.secure_redirects = 1  
#  
# Do not send ICMP redirects (we are not a router)  
#net.ipv4.conf.all.send_redirects = 0  
#  
# Do not accept IP source route packets (we are not a router)  
#net.ipv4.conf.all.accept_source_route = 0  
#net.ipv6.conf.all.accept_source_route = 0  
#  
# Log Martian Packets  
#net.ipv4.conf.all.log_martians = 1  
#  
  
#####  
# Magic system request Key  
# 0=disable, 1=enable all, >1 bitmask of sysrq functions  
# See https://www.kernel.org/doc/html/latest/admin-guide/sysrq.html  
# for what other values do  
#kernel.sysrq=438  
  
root@debian11:~# ls -l /etc/sysctl.d/
```

```
total 4
lrwxrwxrwx 1 root root 14 Mar 20 20:55 99-sysctl.conf -> ../sysctl.conf
-rw-r--r-- 1 root root 639 Apr 6 2021 README.sysctl
```

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

```
root@debian11:~# 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 the given variable(s)
-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)`.

LAB#3 - 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,
- hdparm,
- vmstat,
- mpstat,
- sar.

3.1 - La Commande free

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

```
root@debian11:~# free -m
```

	total	used	free	shared	buff/cache	available
Mem:	3931	147	2543	1	1240	3541
Swap:	974	0	974			

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

- 3931 Mo de mémoire physique totale,
- 147 Mo de mémoire physique utilisée et 2543 Mo de mémoire physique libre,
- 974 Mo de mémoire swap totale et 0 Mo de swap utilisé

Les options de cette commande sont :

```
root@debian11:~# free --help
```

Usage:

```
free [options]
```

Options:

```
-b, --bytes      show output in bytes
--kilo          show output in kilobytes
--mega          show output in megabytes
--giga          show output in gigabytes
--tera          show output in terabytes
--peta          show output in petabytes
-k, --kibi       show output in kibibytes
-m, --mebi       show output in mebibytes
-g, --gibi       show output in gibibytes
--tebi          show output in tebibytes
--pebi          show output in pebibytes
-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)`.

3.2 - Les Commandes uptime et w

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

```
root@debian11:~# uptime
15:51:36 up 4 days,  2:43,  1 user,  load average: 0.10, 0.03, 0.01

root@debian11:~# w
15:51:38 up 4 days,  2:43,  1 user,  load average: 0.10, 0.03, 0.01
USER      TTY      FROM            LOGIN@   IDLE   JCPU   PCPU WHAT
trainee   pts/0    10.0.2.1        10:54    2.00s  0.43s  0.06s sshd: trainee [priv]
```

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@debian11:~# 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@debian11:~# 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).

3.3 - La Commande iostat

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

```
root@debian11:~# iostat
-bash: iostat: command not found
root@debian11:~# apt -y install sysstat
```

```
root@debian11:~# iostat
Linux 5.10.0-13-amd64 (debian11)      05/01/2022      _x86_64_      (2 CPU)

avg-cpu:  %user   %nice %system %iowait  %steal   %idle
           0.21    0.00    0.11    0.31    0.00   99.38

Device            tps    kB_read/s    kB_wrtn/s    kB_dscd/s    kB_read    kB_wrtn    kB_dscd
dm-0               0.01         0.29         0.00         0.00       4220         0         0
dm-1               0.02         0.29         0.00         0.00       4220         0         0
```

md1	0.00	0.08	0.00	0.00	1152	0	0
sda	1.64	37.79	65.45	0.00	551666	955604	0
sdb	0.02	0.31	0.00	0.00	4488	0	0
sdc	0.45	145.46	70.56	0.00	2123717	1030137	0
sr0	0.01	0.15	0.00	0.00	2117	0	0

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 :

- **%user** - Pourcentage de temps passé en mode utilisateur (exécutant des applications, etc.)
- **%nice** - Pourcentage de temps passé en mode utilisateur (pour les processus qui ont modifié leur priorité de programmation à l'aide de la commande **nice**)
- **%system** - Pourcentage de temps passé en mode noyau
- **%steal** - Pourcentage du temps passé par des CPU virtuels en attendant que l'hyperviseur s'occupe d'un autre CPU virtuel.
- **%iowait** - Pourcentage du temps passé à attendre les entrées et les sorties des disques.
- **%idle** - 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.

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.
- Le nombre de transferts (ou opérations d'E/S) par seconde.
- Le nombre de blocs de KB lus par seconde.
- Le nombre de blocs de KB écrits par seconde.
- Le nombre de blocs de KB jetés par seconde.
- Le nombre total de KB lus.
- Le nombre total de KB écrits.
- Le nombre total de KB jetés.

Dernièrement, pour voir les statistiques étendues des disques, utilisez la commande suivante :

```
root@debian11:~# iostat -d -x
```

Linux 5.10.0-13-amd64 (debian11)		05/01/2022		_x86_64_		(2 CPU)							
Device		r/s	rkB/s	rrqm/s	%rrqm	r_await	rareq-sz	w/s	wkB/s	wrqm/s	%wrqm	w_await	wareq-sz
sz	d/s	dkB/s	drqm/s	%drqm	d_await	dareq-sz	f/s	f_await	aqu-sz	%util			
dm-0		0.01	0.29	0.00	0.00	0.34	20.10	0.00	0.00	0.00	0.00	0.00	
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00			
dm-1		0.02	0.29	0.00	0.00	0.68	17.44	0.00	0.00	0.00	0.00	0.00	
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00			
md1		0.00	0.08	0.00	0.00	0.00	20.21	0.00	0.00	0.00	0.00	0.00	
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00			
sda		0.84	37.71	0.57	40.42	3.47	45.06	0.80	65.31	0.91	53.23	61.25	
81.27	0.00	0.00	0.00	0.00	0.00	0.00	0.20	7.89	0.05	0.72			
sdb		0.02	0.31	0.00	0.00	0.12	19.77	0.00	0.00	0.00	0.00	0.00	
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00			
sdc		0.34	145.15	34.70	99.02	2.62	422.80	0.10	70.41	17.34	99.40	9.48	
674.17	0.00	0.00	0.00	0.00	0.00	0.00	0.01	25.37	0.00	0.08			
sr0		0.01	0.14	0.00	0.00	0.30	27.50	0.00	0.00	0.00	0.00	0.00	
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00			

Les options de cette commande sont :

```
root@debian11:~# iostat --help
Usage: iostat [ options ] [ <interval> [ <count> ] ]
Options are:
[ -c ] [ -d ] [ -h ] [ -k | -m ] [ -N ] [ -s ] [ -t ] [ -V ] [ -x ] [ -y ] [ -z ]
[ { -f | +f } <directory> ] [ -j { ID | LABEL | PATH | UUID | ... } ]
[ --dec={ 0 | 1 | 2 } ] [ --human ] [ --pretty ] [ -o JSON ]
[ [ -H ] -g <group_name> ] [ -p [ <device> [,...] | ALL ] ]
[ <device> [...] | ALL ]
```

3.4 - La Commande hdparm

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

```

root@debian11:~# hdparm -t /dev/sdc

root@debian11:~# apt -y install hdparm

root@debian11:~# hdparm -t /dev/sdc

/dev/sdc:
Timing buffered disk reads: 2788 MB in 3.00 seconds = 928.32 MB/sec

```

3.5 - La Commande vmstat

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

```

root@debian11:~# 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
 0  0      0 3229368 26228 621512    0    0   22   33   14   27  0  0 99  0  0
 0  0      0 3229360 26228 621552    0    0    0    0   34   42  0  0 100  0  0
 0  0      0 3229360 26228 621552    0    0    0    0   22   31  0  0 100  0  0
 0  0      0 3229360 26228 621552    0    0    0    0   29   43  0  0 100  0  0
 0  0      0 3229360 26228 621552    0    0    0    0   28   40  0  0 100  0  0
 0  0      0 3229360 26228 621552    0    0    0    0   29   41  0  0 100  0  0
 0  0      0 3229392 26236 621544    0    0    0   12   33   52  0  0 95  6  0
 0  0      0 3229392 26236 621552    0    0    0    0   40   50  0  0 100  0  0
 0  0      0 3229392 26236 621552    0    0    0    0   29   40  0  0 100  0  0
 0  0      0 3229392 26236 621552    0    0    0    0   36   60  0  0 100  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@debian11:~# 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.

3.6 - La Commande `mpstat`

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

```
root@debian11:~# mpstat
Linux 5.10.0-13-amd64 (debian11)      04/30/2022      _x86_64_      (2 CPU)
```

04:10:45 PM	CPU	%usr	%nice	%sys	%iowait	%irq	%soft	%steal	%guest	%gnice	%idle
04:10:45 PM	all	0.04	0.00	0.02	0.12	0.00	0.00	0.00	0.00	0.00	99.83

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

```
root@debian11:~# mpstat -P ALL
Linux 5.10.0-13-amd64 (debian11)          04/30/2022      _x86_64_          (2 CPU)
```

04:11:05 PM	CPU	%usr	%nice	%sys	%iowait	%irq	%soft	%steal	%guest	%gnice	%idle
04:11:05 PM	all	0.04	0.00	0.02	0.12	0.00	0.00	0.00	0.00	0.00	99.83
04:11:05 PM	0	0.04	0.00	0.02	0.13	0.00	0.00	0.00	0.00	0.00	99.81
04:11:05 PM	1	0.04	0.00	0.02	0.10	0.00	0.00	0.00	0.00	0.00	99.84

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@debian11:~# mpstat -P ALL 2 5
Linux 5.10.0-13-amd64 (debian11)          04/30/2022      _x86_64_          (2 CPU)
```

04:11:24 PM	CPU	%usr	%nice	%sys	%iowait	%irq	%soft	%steal	%guest	%gnice	%idle
04:11:26 PM	all	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	100.00
04:11:26 PM	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	100.00
04:11:26 PM	1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	100.00

04:11:26 PM	CPU	%usr	%nice	%sys	%iowait	%irq	%soft	%steal	%guest	%gnice	%idle
04:11:28 PM	all	0.00	0.00	0.25	0.00	0.00	0.00	0.00	0.00	0.00	99.75
04:11:28 PM	0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	100.00
04:11:28 PM	1	0.00	0.00	0.50	0.00	0.00	0.00	0.00	0.00	0.00	99.50

04:11:28 PM	CPU	%usr	%nice	%sys	%iowait	%irq	%soft	%steal	%guest	%gnice	%idle
04:11:30 PM	all	0.00	0.00	0.25	0.00	0.00	0.00	0.00	0.00	0.00	99.75
04:11:30 PM	0	0.00	0.00	0.50	0.00	0.00	0.00	0.00	0.00	0.00	99.50
04:11:30 PM	1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	100.00

04:11:30 PM	CPU	%usr	%nice	%sys	%iowait	%irq	%soft	%steal	%guest	%gnice	%idle
-------------	-----	------	-------	------	---------	------	-------	--------	--------	--------	-------

```

04:11:32 PM all 0.25 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 99.75
04:11:32 PM 0 0.50 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 99.50
04:11:32 PM 1 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 100.00

04:11:32 PM CPU %usr %nice %sys %iowait %irq %soft %steal %guest %gnice %idle
04:11:34 PM all 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 100.00
04:11:34 PM 0 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 100.00
04:11:34 PM 1 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 100.00

Average: CPU %usr %nice %sys %iowait %irq %soft %steal %guest %gnice %idle
Average: all 0.05 0.00 0.10 0.00 0.00 0.00 0.00 0.00 0.00 0.00 99.85
Average: 0 0.10 0.00 0.10 0.00 0.00 0.00 0.00 0.00 0.00 0.00 99.80
Average: 1 0.00 0.00 0.10 0.00 0.00 0.00 0.00 0.00 0.00 0.00 99.90

```

Les options de cette commande sont :

```

root@debian11:~# mpstat --help
Usage: mpstat [ options ] [ <interval> [ <count> ] ]
Options are:
[ -A ] [ -n ] [ -T ] [ -u ] [ -V ]
[ -I { SUM | CPU | SCPU | ALL } ] [ -N { <node_list> | ALL } ]
[ --dec={ 0 | 1 | 2 } ] [ -o JSON ] [ -P { <cpu_list> | ALL } ]

```

3.7 - La Commande sar

La commande **sar** (**S**ystem **A**ctivity **R**epoter) 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

Option	Description
-w	Surveillance du swapping
-p	Surveillance de la pagination
-b	Utilisation des tampons
-d	Utilisation des disques

Sous Debian 11 la commande **/usr/lib/sysstat/sadc** permet de collecter les informations :

```
root@debian8:~# ls /usr/lib/sysstat/  
debian-sa1 sa1 sa2 sadc
```

Le script **/usr/lib/sysstat/sa1** exécute la commande **sadc**. Sous Debian 11, c'est le script **debian-sa1** qui est appelé à la place de **sa1** afin de rectifier une bogue dans ce dernier (Bug#499461). Ce script, tout comme le script **sa1**, prend deux options :

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

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

Sous Debian 11, l'intervalle entre les collectes des informations est configuré par les **timers** de systemd au lieu des cron jobs :

```
root@debian11:~# cat /usr/lib/systemd/system/sysstat-collect.timer  
# /lib/systemd/system/sysstat-collect.timer  
# (C) 2014 Tomasz Torcz <tomek@pipebreaker.pl>  
#  
# sysstat-12.5.2 systemd unit file:  
#     Activates activity collector every 10 minutes  
  
[Unit]  
Description=Run system activity accounting tool every 10 minutes  
  
[Timer]  
OnCalendar=*:00/10
```

```
[Install]
WantedBy=sysstat.service
```

La valeur de **OnCalendar** indique un collecte toutes les dix minutes.

Pour modifier l'intervalle entre les collectes, il faut créer un fichier **override** dans le répertoire **/etc/systemd/system/** en utilisant la commande **systemctl edit** :

```
root@debian11:~# EDITOR=/usr/bin/vi
root@debian11:~# export EDITOR
root@debian11:~# systemctl edit sysstat-collect.timer
### Editing /etc/systemd/system/sysstat-collect.timer.d/override.conf
### Anything between here and the comment below will become the new contents of the file

[Unit]
Description=Run system activity accounting tool every 2 minutes

[Timer]
OnCalendar=
OnCalendar=*:00/2

[Install]
WantedBy=sysstat.service

### Lines below this comment will be discarded

### /lib/systemd/system/sysstat-collect.timer
# # /lib/systemd/system/sysstat-collect.timer
# # (C) 2014 Tomasz Torcz <tomek@pipebreaker.pl>
# #
# # sysstat-12.5.2 systemd unit file:
# #     Activates activity collector every 10 minutes
#
# [Unit]
```



```

    Active: inactive (dead)
    Trigger: n/a
    Triggers: ● sysstat-collect.service
root@debian11:~# systemctl start sysstat-collect.timer
root@debian11:~# systemctl status sysstat-collect.timer
● sysstat-collect.timer - Run system activity accounting tool every 2 minutes
   Loaded: loaded (/lib/systemd/system/sysstat-collect.timer; disabled; vendor preset: enabled)
   Drop-In: /etc/systemd/system/sysstat-collect.timer.d
            └─override.conf
   Active: active (waiting) since Sat 2022-04-30 16:30:54 CEST; 2s ago
   Trigger: Sat 2022-04-30 16:32:00 CEST; 1min 2s left
   Triggers: ● sysstat-collect.service

Apr 30 16:30:54 debian11 systemd[1]: Started Run system activity accounting tool every 2 minutes. <<<<<<<<<<<<
NOTEZ "2 minutes"

```

Saisissez la commande suivante :

<code>

```

root@debian11:~# sar
Linux 5.10.0-13-amd64 (debian11)      04/30/2022      _x86_64_      (2 CPU)

```

Statistiques d'Utilisation du CPU

Visualisez maintenant les statistiques d'utilisation du CPU:

```

root@debian11:~# sar -u 5 3
Linux 5.10.0-13-amd64 (debian11)      04/30/2022      _x86_64_      (2 CPU)

04:33:15 PM    CPU    %user   %nice   %system   %iowait   %steal   %idle
04:33:20 PM    all     0.00    0.00    0.00    0.00    0.00   100.00
04:33:25 PM    all     0.00    0.00    0.00    0.00    0.00   100.00

```

04:33:30 PM	all	0.00	0.00	0.00	0.00	0.00	100.00
Average:	all	0.00	0.00	0.00	0.00	0.00	100.00

D'avantage de statistiques peuvent être obtenues en utilisant l'option **ALL** :

```

root@debian11:~# sar -u ALL 5 3
Linux 5.10.0-13-amd64 (debian11)      04/30/2022      _x86_64_      (2 CPU)

04:33:53 PM    CPU    %usr    %nice    %sys    %iowait    %steal    %irq    %soft    %guest    %gnice
%idle
04:33:58 PM    all     0.00     0.00     0.00     0.00     0.00     0.00     0.00     0.00     0.00
100.00
04:34:03 PM    all     0.10     0.00     0.00     0.00     0.00     0.00     0.00     0.00     0.00
99.90
04:34:08 PM    all     0.00     0.00     0.00     0.00     0.00     0.00     0.00     0.00     0.00
100.00
Average:        all     0.03     0.00     0.00     0.00     0.00     0.00     0.00     0.00     0.00
99.97

```

Pour consulter les statistiques d'un coeur spécifique, utilisez l'option **-P** :

```

root@debian11:~# sar -u -P 0 5 3
Linux 5.10.0-13-amd64 (debian11)      04/30/2022      _x86_64_      (2 CPU)

04:35:14 PM    CPU    %user    %nice    %system    %iowait    %steal    %idle
04:35:19 PM      0     0.00     0.00     0.00     0.00     0.00    100.00
04:35:24 PM      0     0.00     0.00     0.00     0.00     0.00    100.00
04:35:29 PM      0     0.00     0.00     0.00     0.00     0.00    100.00
Average:         0     0.00     0.00     0.00     0.00     0.00    100.00

```

```

root@debian11:~# sar -u -P 1 5 3
Linux 5.10.0-13-amd64 (debian11)      04/30/2022      _x86_64_      (2 CPU)

04:34:39 PM    CPU    %user    %nice    %system    %iowait    %steal    %idle

```

04:34:44 PM	1	0.00	0.00	0.00	0.00	0.00	100.00
04:34:49 PM	1	0.00	0.00	0.00	0.00	0.00	100.00
04:34:54 PM	1	0.00	0.00	0.00	0.00	0.00	100.00
Average:	1	0.00	0.00	0.00	0.00	0.00	100.00

Statistiques d'Utilisation de la Mémoire et du Swap

Utilisez l'option **-r** pour visualiser les statistiques concernant la mémoire :

```
root@debian11:~# sar -r 5 3
Linux 5.10.0-13-amd64 (debian11)      04/30/2022      _x86_64_      (2 CPU)

04:35:58 PM kbmemfree  kbavail  kbmempused  %mempused  kbbuffers  kbcached  kbcommit   %commit  kbactive  kbinact
kldirty
04:36:03 PM   2977608   3629252    115520     2.87     33556    789044   1645752   32.76   463612   428012
308
04:36:08 PM   2977664   3629308    115448     2.87     33556    789068   1645752   32.76   463612   428016
356
04:36:13 PM   2977664   3629332    115440     2.87     33564    789068   1645752   32.76   463636   428016
308
Average:        2977645   3629297    115469     2.87     33559    789060   1645752   32.76   463620   428015
324
```

HERE

Utilisez l'option **-S** pour visualiser les statistiques concernant le Swap :

```
root@debian11:~# sar -S 5 3
Linux 5.10.0-13-amd64 (debian11)      04/30/2022      _x86_64_      (2 CPU)

04:39:41 PM kbswpfree  kbswpused  %swpused  kbswpcad  %swpcad
04:39:46 PM   998396         0     0.00         0     0.00
04:39:51 PM   998396         0     0.00         0     0.00
```

04:39:56 PM	998396	0	0.00	0	0.00
Average:	998396	0	0.00	0	0.00

Statistiques des E/S

Utilisez l'option **-b** pour visualiser les statistiques concernant les E/S :

```
root@debian11:~# sar -b 5 3
Linux 5.10.0-13-amd64 (debian11)      04/30/2022      _x86_64_      (2 CPU)

04:40:19 PM      tps      rtps      wtps      dtps      bread/s      bwrtn/s      bdscd/s
04:40:24 PM      0.40      0.00      0.40      0.00      0.00      11.20      0.00
04:40:29 PM      0.00      0.00      0.00      0.00      0.00      0.00      0.00
04:40:34 PM      0.00      0.00      0.00      0.00      0.00      0.00      0.00
Average:          0.13      0.00      0.13      0.00      0.00      3.73      0.00
```

Statistiques des E/S par Disque

Utilisez l'option **-d** pour visualiser les statistiques concernant les E/S par disque :

```
root@debian11:~# sar -d 5 3
Linux 5.10.0-13-amd64 (debian11)      05/01/2022      _x86_64_      (2 CPU)

02:48:27 PM      DEV      tps      rkB/s      wkB/s      dkB/s      areq-sz      aqu-sz      await      %util
02:48:32 PM      sr0      0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
02:48:32 PM      sda      0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
02:48:32 PM      sdb      0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
02:48:32 PM      sdc      0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
02:48:32 PM      dm-0     0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
02:48:32 PM      dm-1     0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
02:48:32 PM      md1      0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
```

02:48:32 PM	DEV	tps	rkB/s	wkB/s	dkB/s	areq-sz	aqu-sz	await	%util
02:48:37 PM	sr0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
02:48:37 PM	sda	1.80	0.00	7.20	0.00	4.00	0.05	26.44	0.56
02:48:37 PM	sdb	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
02:48:37 PM	sdc	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
02:48:37 PM	dm-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
02:48:37 PM	dm-1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
02:48:37 PM	md1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
02:48:37 PM	DEV	tps	rkB/s	wkB/s	dkB/s	areq-sz	aqu-sz	await	%util
02:48:42 PM	sr0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
02:48:42 PM	sda	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
02:48:42 PM	sdb	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
02:48:42 PM	sdc	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
02:48:42 PM	dm-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
02:48:42 PM	dm-1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
02:48:42 PM	md1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Average:	DEV	tps	rkB/s	wkB/s	dkB/s	areq-sz	aqu-sz	await	%util
Average:	sr0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Average:	sda	0.60	0.00	2.40	0.00	4.00	0.02	26.44	0.19
Average:	sdb	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Average:	sdc	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Average:	dm-0	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Average:	dm-1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Average:	md1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.000

Dernièrement, vérifiez que le répertoire **/var/log/sysstat/** contient un fichier **saXX** :

```
root@debian11:~# ls -l /var/log/sysstat/
total 16
-rw-r--r-- 1 root root 14284 Apr 30 16:44 sa30
```

Les options de la commande sar sont :

```
root@debian11:~# sar --help
Usage: sar [ options ] [ <interval> [ <count> ] ]
Main options and reports (report name between square brackets):
  -B      Paging statistics [A_PAGE]
  -b      I/O and transfer rate statistics [A_IO]
  -d      Block devices statistics [A_DISK]
  -F [ MOUNT ]
           Filesystems statistics [A_FS]
  -H      Hugepages utilization statistics [A_HUGE]
  -I { <int_list> | SUM | ALL }
           Interrupts statistics [A_IRQ]
  -m { <keyword> [,...] | ALL }
           Power management statistics [A_PWR_...]
           Keywords are:
           CPU      CPU instantaneous clock frequency
           FAN      Fans speed
           FREQ     CPU average clock frequency
           IN       Voltage inputs
           TEMP     Devices temperature
           USB      USB devices plugged into the system
  -n { <keyword> [,...] | ALL }
           Network statistics [A_NET_...]
           Keywords are:
           DEV      Network interfaces
           EDEV     Network interfaces (errors)
           NFS      NFS client
           NFSD     NFS server
           SOCK     Sockets (v4)
           IP       IP traffic          (v4)
           EIP      IP traffic          (v4) (errors)
           ICMP     ICMP traffic        (v4)
           EICMP    ICMP traffic        (v4) (errors)
           TCP      TCP traffic          (v4)
           ETCP     TCP traffic          (v4) (errors)
```

```

    UDP      UDP traffic      (v4)
    SOCK6    Sockets (v6)
    IP6      IP traffic      (v6)
    EIP6     IP traffic      (v6) (errors)
    ICMP6    ICMP traffic    (v6)
    EICMP6   ICMP traffic    (v6) (errors)
    UDP6     UDP traffic      (v6)
    FC       Fibre channel HBAs
    SOFT     Software-based network processing
-q [ <keyword> [,...] | PSI | ALL ]
    System load and pressure-stall statistics
    Keywords are:
    LOAD     Queue length and load average statistics [A_QUEUE]
    CPU      Pressure-stall CPU statistics [A_PSI_CPU]
    IO       Pressure-stall I/O statistics [A_PSI_IO]
    MEM      Pressure-stall memory statistics [A_PSI_MEM]
-r [ ALL ]
    Memory utilization statistics [A_MEMORY]
-S      Swap space utilization statistics [A_MEMORY]
-u [ ALL ]
    CPU utilization statistics [A_CPU]
-v      Kernel tables statistics [A_KTABLES]
-W      Swapping statistics [A_SWAP]
-w      Task creation and system switching statistics [A_PCSW]
-y      TTY devices statistics [A_SERIAL]

```

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
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 (systemd),
- kmod, d'une manière dynamique et transparente lors du branchement du périphérique,
- udev,
- manuellement.

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 **udev** 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/sdx1** 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@debian11:~# cat /etc/udev/udev.conf
# see udev.conf(5) for details
#
# udevd is also started in the initrd.  When this file is modified you might
# also want to rebuild the initrd, so that it will include the modified configuration.

#udev_log=info
#children_max=
#exec_delay=
#event_timeout=180
#timeout_signal=SIGKILL
#resolve_names=early
```

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

```
root@debian11:~# ls /lib/udev/rules.d/
39-usbmuxd.rules          70-joystick.rules        77-mm-usb-device-blacklist.rules
40-usb_modeswitch.rules  70-mouse.rules           77-mm-usb-serial-adapters-greylist.rules
50-firmware.rules        70-power-switch.rules    77-mm-x22x-port-types.rules
50-udev-default.rules    70-printers.rules        77-mm-zte-port-types.rules
55-dm.rules              70-touchpad.rules        78-sound-card.rules
56-lvm.rules             70-uaccess.rules         80-debian-compat.rules
60-autosuspend.rules    71-ipp-usb.rules         80-drivers.rules
60-block.rules           71-seat.rules            80-ifupdown.rules
60-cdrom_id.rules        73-seat-late.rules       80-libinput-device-groups.rules
60-drm.rules             73-special-net-names.rules
60-evdev.rules           75-net-description.rules  80-mm-candidate.rules
60-fido-id.rules         75-probe_mtd.rules       80-net-setup-link.rules
60-input-id.rules        77-mm-broadmobi-port-types.rules
60-libgphoto2-6.rules    77-mm-cinterion-port-types.rules
60-libopenni2-0.rules    77-mm-dell-port-types.rules
60-libsane1.rules        77-mm-dlink-port-types.rules
60-persistent-alsa.rules 77-mm-ericsson-mbm.rules 80-udisks2.rules
60-persistent-input.rules 77-mm-fibocom-port-types.rules
60-persistent-storage-dm.rules
60-persistent-storage.rules
60-persistent-storage-tape.rules
60-persistent-v4l.rules  77-mm-foxconn-port-types.rules
60-qemu-guest-agent.rules 77-mm-haier-port-types.rules
60-sensor.rules          77-mm-huawei-net-port-types.rules
60-serial.rules          77-mm-longcheer-port-types.rules
64-btrfs-dm.rules        77-mm-mtk-port-types.rules
64-btrfs.rules           77-mm-nokia-port-types.rules
64-xorg-xkb.rules        77-mm-pcmcia-device-blacklist.rules
65-libwacom.rules        77-mm-qdl-device-blacklist.rules
69-cd-sensors.rules      77-mm-quectel-port-types.rules
69-lvm-metad.rules       77-mm-sierra.rules       90-alsa-restore.rules
                          77-mm-simtech-port-types.rules
                          77-mm-telit-port-types.rules
                          77-mm-tplink-port-types.rules
                          90-console-setup.rules
                          90-libinput-fuzz-override.rules
                          90-nm-thunderbolt.rules
                          90-pipewire-alsa.rules
                          90-pulseaudio.rules
                          95-cd-devices.rules
                          95-dm-notify.rules
                          95-upower-csr.rules
                          95-upower-hidpp.rules
                          95-upower-hid.rules
                          95-upower-wup.rules
                          96-e2scrub.rules
                          99-libsane1.rules
                          99-systemd.rules
```

69-wacom.rules

77-mm-ublox-port-types.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@debian11:~# cat /lib/udev/rules.d/50-udev-default.rules | more
# do not edit this file, it will be overwritten on update

# run a command on remove events
ACTION=="remove", ENV{REMOVE_CMD}!="", RUN+="$env{REMOVE_CMD}"
ACTION=="remove", GOTO="default_end"

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",
GOTO="default_hwdb_imported"
ENV{MODALIAS}!="", IMPORT{builtin}="hwdb --subsystem=$env{SUBSYSTEM}"
LABEL="default_hwdb_imported"

ACTION!="add", GOTO="default_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]|ttymxc[0-9]*|pppox[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", GROUP="input"
SUBSYSTEM=="input", KERNEL=="js[0-9]*", MODE="0664"

SUBSYSTEM=="video4linux", GROUP="video"
SUBSYSTEM=="graphics", GROUP="video"
SUBSYSTEM=="drm", KERNEL!="renderD*", GROUP="video"
SUBSYSTEM=="dvb", GROUP="video"
SUBSYSTEM=="media", GROUP="video"
SUBSYSTEM=="cec", GROUP="video"

SUBSYSTEM=="drm", KERNEL=="renderD*", GROUP="render", MODE="0660"
SUBSYSTEM=="kfd", GROUP="render", MODE="0660"

# When using static_node= with non-default permissions, also update
# tmpfiles.d/static-nodes-permissions.conf.in to keep permissions synchronized.

SUBSYSTEM=="sound", GROUP="audio", \
    OPTIONS+="static_node=snd/seq", OPTIONS+="static_node=snd/timer"

SUBSYSTEM=="usb", ENV{DEVTYPE}=="usb_device", MODE="0664"
--More--
[q]
```

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 nom 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** :

```
root@debian11:~# udevadm info --query=all -n /dev/sdc
P: /devices/pci0000:00/0000:00:05.0/virtio1/host2/target2:0:0/2:0:0:1/block/sdc
N: sdc
L: 0
S: disk/by-id/scsi-0QEMU_QEMU_HARDDISK_drive-scsi1
S: disk/by-path/pci-0000:00:05.0-scsi-0:0:0:1
E: DEVPATH=/devices/pci0000:00/0000:00:05.0/virtio1/host2/target2:0:0/2:0:0:1/block/sdc
```

```
E: DEVNAME=/dev/sdc
E: DEVTYPEDisk
E: MAJOR=8
E: MINOR=32
E: SUBSYSTEM=block
E: USEC_INITIALIZED=1712598
E: ID_SCSI=1
E: ID_VENDOR=QEMU
E: ID_VENDOR_ENC=QEMU\x20\x20\x20\x20
E: ID_MODEL=QEMU_HARDDISK
E: ID_MODEL_ENC=QEMU\x20HARDDISK\x20\x20\x20
E: ID_REVISION=2.5+
E: ID_TYPE=disk
E: ID_SERIAL=0QEMU_QEMU_HARDDISK_drive-scsi1
E: ID_SERIAL_SHORT=drive-scsi1
E: ID_BUS=scsi
E: ID_PATH=pci-0000:00:05.0-scsi-0:0:0:1
E: ID_PATH_TAG=pci-0000_00_05_0-scsi-0_0_0_1
E: ID_PART_TABLE_UUID=304308a3
E: ID_PART_TABLE_TYPE=dos
E: DEVLINKS=/dev/disk/by-id/scsi-0QEMU_QEMU_HARDDISK_drive-scsi1 /dev/disk/by-path/pci-0000:00:05.0-scsi-0:0:0:1
E: TAGS=:systemd:
E: CURRENT_TAGS=:systemd:
```

Les options de la commande udevadm sont :

```
root@debian11:~# udevadm --help
udevadm [--help] [--version] [--debug] COMMAND [COMMAND OPTIONS]
```

Send control commands or test the device manager.

Commands:

info	Query sysfs or the udev database
trigger	Request events from the kernel

settle	Wait for pending udev events
control	Control the udev daemon
monitor	Listen to kernel and udev events
test	Test an event run
test-builtin	Test a built-in command

See the udevadm(8) man page for details.

```
root@debian11:~# udevadm info --help
udevadm info [OPTIONS] [DEVPATH|FILE]
```

Query sysfs or the udev database.

-h --help	Print this message
-V --version	Print version of the program
-q --query=TYPE	Query device information:
name	Name of device node
symlink	Pointing to node
path	sysfs device path
property	The device properties
all	All values
-p --path=SYSPATH	sysfs device path used for query or attribute walk
-n --name=NAME	Node or symlink name used for query or attribute walk
-r --root	Prepend dev directory to path names
-a --attribute-walk	Print all key matches walking along the chain of parent devices
-d --device-id-of-file=FILE	Print major:minor of device containing this file
-x --export	Export key/value pairs
-P --export-prefix	Export the key name with a prefix
-e --export-db	Export the content of the udev database
-c --cleanup-db	Clean up the udev database
-w --wait-for-initialization[=SECONDS]	Wait for device to be initialized

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@debian11:~# ls -l /sys
total 0
drwxr-xr-x  2 root root 0 Apr 30 16:32 block
drwxr-xr-x 33 root root 0 Apr 30 16:21 bus
drwxr-xr-x 51 root root 0 Apr 30 16:21 class
drwxr-xr-x  4 root root 0 Apr 30 16:21 dev
drwxr-xr-x 14 root root 0 Apr 26 13:08 devices
drwxr-xr-x  6 root root 0 Apr 30 16:21 firmware
drwxr-xr-x 10 root root 0 Apr 26 13:08 fs
drwxr-xr-x  2 root root 0 Apr 30 16:50 hypervisor
drwxr-xr-x 15 root root 0 Apr 26 13:08 kernel
drwxr-xr-x 151 root root 0 Apr 30 16:21 module
drwxr-xr-x  3 root root 0 Apr 30 16:50 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@debian11:~# cat /sys/block/sdc/sdc1/size
204800
```

Ce chiffre correspond au nombre de secteurs.

LAB #4 - Limitation des ressources

4.1 - 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 le fichier **/etc/security/limits.conf** :

```
root@debian11:~# cat /etc/security/limits.conf
# /etc/security/limits.conf
#
#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
#      - NOTE: group and wildcard limits are not applied to root.
#        To apply a limit to the root user, <domain> must be
#        the literal username root.
#
#<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
# - chroot - change root to directory (Debian-specific)
#
#<domain>      <type> <item>      <value>
#
#*              soft   core          0
#root           hard   core          100000
#*              hard   rss            10000
#@student       hard   nproc         20
#@faculty       soft   nproc         20
#@faculty       hard   nproc         50
#ftp            hard   nproc         0
#ftp            -      chroot         /ftp
#@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@debian11:~# ulimit -a
real-time non-blocking time (microseconds, -R) unlimited
core file size              (blocks, -c) 0
data seg size               (kbytes, -d) unlimited
scheduling priority         (-e) 0
file size                   (blocks, -f) unlimited
pending signals             (-i) 15545
max locked memory           (kbytes, -l) 503199
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) 15545
virtual memory              (kbytes, -v) unlimited
file locks                  (-x) unlimited
```

Les options de **ulimit** sont :

```
root@debian11:~# help ulimit
ulimit: ulimit [-SHabcdefiklmnpqrstuvxPT] [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
- k the maximum number of kqueues allocated for this process
- 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
- P the maximum number of pseudoterminals
- R the maximum time a real-time process can run before blocking
- T the maximum number of threads

Not all options are available on all platforms.

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.

4.2 - Groupes de Contrôle

CGroups v1

Préparation

Debian 11 utilise CGroups v2 par défaut. Pour revenir à l'utilisation de CGroups v1, éditez le fichier **/etc/boot/grub** et ajoutez la directive **systemd.unified_cgroup_hierarchy=0** à la ligne **GRUB_CMDLINE_LINUX_DEFAULT** :

```
root@debian11:~# vi /etc/default/grub
root@debian11:~# cat /etc/default/grub
# If you change this file, run 'update-grub' afterwards to update
# /boot/grub/grub.cfg.
# For full documentation of the options in this file, see:
#   info -f grub -n 'Simple configuration'

GRUB_DEFAULT=0
GRUB_TIMEOUT=5
GRUB_DISTRIBUTOR=`lsb_release -i -s 2> /dev/null || echo Debian`
GRUB_CMDLINE_LINUX_DEFAULT="quiet systemd.unified_cgroup_hierarchy=0"
GRUB_CMDLINE_LINUX=""

# Uncomment to enable BadRAM filtering, modify to suit your needs
# This works with Linux (no patch required) and with any kernel that obtains
# the memory map information from GRUB (GNU Mach, kernel of FreeBSD ...)
#GRUB_BADRAM="0x01234567,0xfefefefe,0x89abcdef,0xefefefef"

# Uncomment to disable graphical terminal (grub-pc only)
#GRUB_TERMINAL=console

# The resolution used on graphical terminal
# note that you can use only modes which your graphic card supports via VBE
```

```
# you can see them in real GRUB with the command `vbeinfo`
#GRUB_GFXMODE=640x480

# Uncomment if you don't want GRUB to pass "root=UUID=xxx" parameter to Linux
#GRUB_DISABLE_LINUX_UUID=true

# Uncomment to disable generation of recovery mode menu entries
#GRUB_DISABLE_RECOVERY="true"

# Uncomment to get a beep at grub start
#GRUB_INIT_TUNE="480 440 1"

root@debian11:~# grub-mkconfig -o /boot/grub/grub.cfg
Generating grub configuration file ...
Found background image: /usr/share/images/desktop-base/desktop-grub.png
Found linux image: /boot/vmlinuz-5.10.0-13-amd64
Found initrd image: /boot/initrd.img-5.10.0-13-amd64
done
```

Redémarrez ensuite votre VM :

```
root@debian11:~# reboot
```

Présentation

Les **Groupes de Contrôles** (*Control Groups*) aussi appelés **CGroups**, sont une façon 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 Debian 11 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.

Commencez par installer le paquet **cgroup-tools** :

```
root@debian11:~# apt -y install cgroup-tools
```

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

```
root@debian11:~# lssubsys -am
cpuset /sys/fs/cgroup/cpuset
cpu,cpuacct /sys/fs/cgroup/cpu,cpuacct
blkio /sys/fs/cgroup/blkio
memory /sys/fs/cgroup/memory
devices /sys/fs/cgroup/devices
freezer /sys/fs/cgroup/freezer
net_cls,net_prio /sys/fs/cgroup/net_cls,net_prio
perf_event /sys/fs/cgroup/perf_event
```

```
hugetlb /sys/fs/cgroup/hugetlb
pids /sys/fs/cgroup/pids
rdma /sys/fs/cgroup/rdma
```

Sous Debian 11, **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@debian11:~# 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-lvm2\x2dpvscan.slice         loaded active active system-lvm2\x2dpvscan.slice
system-modprobe.slice               loaded active active system-modprobe.slice
system-systemd\x2dcryptsetup.slice  loaded active active Cryptsetup Units Slice
system.slice                         loaded active active System Slice
user-1000.slice                     loaded active active User Slice of UID 1000
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.

8 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@debian11:~# systemd-cgls
Control group /:
-.slice
├─user.slice
│   └─user-1000.slice
│       ├──user@1000.service ...
│       └─app.slice
│           ├──pulseaudio.service
│           │   └─974 /usr/bin/pulseaudio --daemonize=no --log-target=journal
│           ├──pipewire.service
│           │   ├──973 /usr/bin/pipewire
│           │   └─984 /usr/bin/pipewire-media-session
│           └─dbus.service
│               └─982 /usr/bin/dbus-daemon --session --address=systemd: --nofork --nopidfile --systemd-activation --
└─syslog-only
    ├──init.scope
    │   ├──958 /lib/systemd/systemd --user
    │   └─959 (sd-pam)
    ├──session-3.scope
    │   ├──993 sshd: trainee [priv]
    │   ├──999 sshd: trainee@pts/0
    │   ├──1000 -bash
    │   ├──1003 su -
    │   ├──1004 -bash
    │   ├──1010 systemd-cgls
    │   └─1011 less
    └─session-1.scope
        ├──578 /bin/login -p --
        └─975 -bash
```

```
|   |   |986 su -
|   |   |987 -bash
|---init.scope
|   |1 /sbin/init
|---system.slice
|   |---apache2.service
|   |   |595 /usr/sbin/apache2 -k start
|   |   |597 /usr/sbin/apache2 -k start
|   |   |598 /usr/sbin/apache2 -k start
|   |---systemd-udevd.service
|   |   |317 /lib/systemd/systemd-udevd
|   |---cron.service
|   |   |491 /usr/sbin/cron -f
|   |---polkit.service
|   |   |495 /usr/libexec/polkitd --no-debug
|   |---rtkit-daemon.service
|   |   |979 /usr/libexec/rtkit-daemon
|   |---auditd.service
|   |   |460 /sbin/auditd
|   |---wpa_supplicant.service
|   |   |498 /sbin/wpa_supplicant -u -s -0 /run/wpa_supplicant
|   |---ModemManager.service
|   |   |515 /usr/sbin/ModemManager
|   |---inetd.service
|   |   |694 /usr/sbin/inetd
|   |---systemd-journald.service
|   |   |296 /lib/systemd/systemd-journald
|   |---mdmonitor.service
|   |   |432 /sbin/mdadm --monitor --scan
|   |---ssh.service
|   |   |580 sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups
lines 1-58
```

[q]

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.

Limitation de la Mémoire

Commencez par créer le script **hello-world.sh** qui servira à générer un processus pour travailler avec les CGroups :

```
root@debian11:~# vi hello-world.sh
root@debian11:~# cat hello-world.sh
#!/bin/bash
while [ 1 ]; do
    echo "hello world"
    sleep 360
done
```

Rendez le script exécutable et testez-le :

```
root@debian11:~# chmod u+x hello-world.sh
root@debian11:~# ./hello-world.sh
hello world
^C
```

Créez maintenant un CGroup dans le sous-système **memory** appelé **helloworld** :

```
root@debian11:~# mkdir /sys/fs/cgroup/memory/helloworld
```

Par défaut, ce CGroup héritera de l'ensemble de la mémoire disponible. Pour éviter cela, créez maintenant une limite de **40000000** octets pour ce

CGroup :

```
root@debian11:~# echo 40000000 > /sys/fs/cgroup/memory/helloworld/memory.limit_in_bytes
root@debian11:~# cat /sys/fs/cgroup/memory/helloworld/memory.limit_in_bytes
39997440
```



Important - Notez que les 40 000 000 demandés sont devenus 39 997 440 ce qui correspond à un nombre entier de pages mémoire du noyau de 4Ko. ($39\,997\,440 / 4096 = 9\,765$).

Lancez maintenant le script **helloworld.sh** :

```
root@debian11:~# ./hello-world.sh &
[1] 1073
root@debian11:~# hello world
[Entrée]

root@debian11:~# ps aux | grep hello-world
root      1073  0.0  0.0   6756   3100 pts/0    S   06:33   0:00 /bin/bash ./hello-world.sh
root      1077  0.0  0.0    6180    712 pts/0    R+  06:34   0:00 grep hello-world
```

Notez qu'il n'y a pas de limite de la mémoire, ce qui implique l'héritage par défaut :

```
root@debian11:~# ps -ww -o cgroup 1073
CGROUP
8:devices:/user.slice,7:pids:/user.slice/user-1000.slice/session-3.scope,5:memory:/user.slice/user-1000.slice/session-3.scope,1:name=systemd:/user.slice/user-1000.slice/session-3.scope,0::/user.slice/user-1000.slice/session-3.scope
```

Insérer le PID de notre script dans le CGroup **helloworld** :

```
root@debian11:~# echo 1073 > /sys/fs/cgroup/memory/helloworld/cgroup.procs
```

Notez maintenant l'héritage de la limitation de la mémoire - **5:memory:/helloworld** :

```
root@debian11:~# ps -ww -o cgroup 1073
CGROUP
8:devices:/user.slice,7:pids:/user.slice/user-1000.slice/session-3.scope,5:memory:/helloworld,1:name=systemd:/user.slice/user-1000.slice/session-3.scope,0::/user.slice/user-1000.slice/session-3.scope
```

Constatez ensuite l'occupation mémoire réelle :

```
root@debian11:~# cat /sys/fs/cgroup/memory/helloworld/memory.usage_in_bytes
274432
```

Tuez le script **hello-world.sh** :

```
root@debian11:~# kill 1073
root@debian11:~# ps aux | grep hello-world
root      1086  0.0  0.0   6180   716 pts/0    S+   06:37   0:00 grep hello-world
[1]+  Terminated                  ./hello-world.sh
```

Créez un second CGroup beaucoup plus restrictif :

```
root@debian11:~# mkdir /sys/fs/cgroup/memory/helloworld1
root@debian11:~# echo 6000 > /sys/fs/cgroup/memory/helloworld1/memory.limit_in_bytes
root@debian11:~# cat /sys/fs/cgroup/memory/helloworld1/memory.limit_in_bytes
4096
```

Relancez le script **hello-world.sh** et insérez-le dans le nouveau CGroup :

```
root@debian11:~# ./hello-world.sh &
[1] 1089
```

```
root@debian11:~# hello world
[Entrée]
```

```
root@debian11:~# echo 1089 > /sys/fs/cgroup/memory/helloworld1/cgroup.procs
```

Attendez la prochaine sortie de **hello world** sur le canal standard puis constatez que le script s'arrête :

```
root@debian11:~# ps aux | grep hello-world
root      1100  0.0  0.0   6180   720 pts/0    S+   06:45   0:00 grep hello-world
[1]+  Killed                  ./hello-world.sh
```

Notez la trace dans le fichier **/var/log/messages** :

```
root@debian11:~# tail /var/log/messages
May  4 06:44:43 debian11 kernel: [ 994.012423] workingset_nodereclaim 0
May  4 06:44:43 debian11 kernel: [ 994.012423] pgfault 0
May  4 06:44:43 debian11 kernel: [ 994.012423] pgmajfault 0
May  4 06:44:43 debian11 kernel: [ 994.012423] pgrefill 0
May  4 06:44:43 debian11 kernel: [ 994.012423] pgscan 0
May  4 06:44:43 debian11 kernel: [ 994.012423] pgsteal 0
May  4 06:44:43 debian11 kernel: [ 994.012425] Tasks state (memory values in pages):
May  4 06:44:43 debian11 kernel: [ 994.012426] [  pid ]   uid  tgid total_vm      rss pgtables_bytes swapents
oom_score_adj name
May  4 06:44:43 debian11 kernel: [ 994.012428] [  1089]     0  1089    1689     780    53248         0
0 hello-world.sh
May  4 06:44:43 debian11 kernel: [ 994.012430] oom-
kill:constraint=CONSTRAINT_MEMCG,nodemask=(null),cpuset=/,mems_allowed=0,oom_memcg=/helloworld1,task_memcg=/hello
world1,task=hello-world.sh,pid=1089,uid=0
```

La Commande cgcreate

Cette commande permet la création d'un CGroup :

```
root@debian11:~# cgcreate -g memory:helloworld2
```

```
root@debian11:~# ls -l /sys/fs/cgroup/memory/helloworld2/
```

```
total 0
```

```
-rw-r--r-- 1 root root 0 May  4 06:47 cgroup.clone_children
--w--w--w- 1 root root 0 May  4 06:47 cgroup.event_control
-rw-r--r-- 1 root root 0 May  4 06:47 cgroup.procs
-rw-r--r-- 1 root root 0 May  4 06:47 memory.failcnt
--w----- 1 root root 0 May  4 06:47 memory.force_empty
-rw-r--r-- 1 root root 0 May  4 06:47 memory.kmem.failcnt
-rw-r--r-- 1 root root 0 May  4 06:47 memory.kmem.limit_in_bytes
-rw-r--r-- 1 root root 0 May  4 06:47 memory.kmem.max_usage_in_bytes
-r--r--r-- 1 root root 0 May  4 06:47 memory.kmem.slabinfo
-rw-r--r-- 1 root root 0 May  4 06:47 memory.kmem.tcp.failcnt
-rw-r--r-- 1 root root 0 May  4 06:47 memory.kmem.tcp.limit_in_bytes
-rw-r--r-- 1 root root 0 May  4 06:47 memory.kmem.tcp.max_usage_in_bytes
-r--r--r-- 1 root root 0 May  4 06:47 memory.kmem.tcp.usage_in_bytes
-r--r--r-- 1 root root 0 May  4 06:47 memory.kmem.usage_in_bytes
-rw-r--r-- 1 root root 0 May  4 06:47 memory.limit_in_bytes
-rw-r--r-- 1 root root 0 May  4 06:47 memory.max_usage_in_bytes
-rw-r--r-- 1 root root 0 May  4 06:47 memory.memsw.failcnt
-rw-r--r-- 1 root root 0 May  4 06:47 memory.memsw.limit_in_bytes
-rw-r--r-- 1 root root 0 May  4 06:47 memory.memsw.max_usage_in_bytes
-r--r--r-- 1 root root 0 May  4 06:47 memory.memsw.usage_in_bytes
-rw-r--r-- 1 root root 0 May  4 06:47 memory.move_charge_at_immigrate
-r--r--r-- 1 root root 0 May  4 06:47 memory.numa_stat
-rw-r--r-- 1 root root 0 May  4 06:47 memory.oom_control
----- 1 root root 0 May  4 06:47 memory.pressure_level
-rw-r--r-- 1 root root 0 May  4 06:47 memory.soft_limit_in_bytes
-r--r--r-- 1 root root 0 May  4 06:47 memory.stat
-rw-r--r-- 1 root root 0 May  4 06:47 memory.swappiness
-r--r--r-- 1 root root 0 May  4 06:47 memory.usage_in_bytes
-rw-r--r-- 1 root root 0 May  4 06:47 memory.use_hierarchy
-rw-r--r-- 1 root root 0 May  4 06:47 notify_on_release
```

```
-rw-r--r-- 1 root root 0 May  4 06:47 tasks
```

Il n'existe cependant pas de commande pour affecter une limitation de la mémoire :

```
root@debian11:~# echo 40000000 > /sys/fs/cgroup/memory/helloworld2/memory.limit_in_bytes
```

La Commande cgexec

Cette commande permet d'insérer la limitation dans le CGroup **et** de lancer le script en une seule ligne :

```
root@debian11:~# cgexec -g memory:helloworld2 ./hello-world.sh &
[1] 1106

root@debian11:~# hello world
[Entrée]

root@debian11:~# cat /sys/fs/cgroup/memory/helloworld2/cgroup.procs
1106
1107
root@debian11:~# ps aux | grep 110
root      1106  0.0  0.0   6756  3060 pts/0    S   06:48   0:00 /bin/bash ./hello-world.sh
root      1107  0.0  0.0   5304   508 pts/0    S   06:48   0:00 sleep 360
root      1108  0.0  0.0      0      0 ?        I   06:49   0:00 [kworker/1:0-events_freezable]
root      1113  0.0  0.0   6180   652 pts/0    S+  06:50   0:00 grep 110
```

La Commande cgdelete

Une fois le script terminé, cette commande permet de supprimer le cgroup :

```
root@debian11:~# kill 1106
root@debian11:~# ps aux | grep 110
```

```
root      1107  0.0  0.0   5304   508 pts/0    S   06:48   0:00 sleep 360
root      1108  0.0  0.0      0      0 ?        I   06:49   0:00 [kworker/1:0-mm_percpu_wq]
root      1115  0.0  0.0   6180   716 pts/0    R+  06:51   0:00 grep 110
[1]+  Terminated                  cgexec -g memory:helloworld2 ./hello-world.sh

root@debian11:~# cgdelete memory:helloworld2

root@debian11:~# ls -l /sys/fs/cgroup/memory/helloworld2/
ls: cannot access '/sys/fs/cgroup/memory/helloworld2/': No such file or directory
```

Le Fichier `/etc/cgconfig.conf`

Afin de les rendre persistants, il convient d'éditer le fichier `/etc/cgconfig.conf` :

```
root@debian11:~# vi /etc/cgconfig.conf
root@debian11:~# cat /etc/cgconfig.conf
group helloworld2 {
    cpu {
        cpu.shares = 100;
    }
    memory {
        memory.limit_in_bytes = 40000;
    }
}
```



Important - Notez la création de **deux** limitations, une de 40 000 octets de mémoire et l'autre de **100 cpu.shares**. Cette dernière est une valeur exprimée sur 1 024, où 1 024 représente 100% du temps CPU. La limite fixée est donc équivalente à 9,77% du temps CPU.

Créez donc les deux CGroups concernés :

```
root@debian11:~# cgcreate -g memory:helloworld2

root@debian11:~# ls -l /sys/fs/cgroup/memory/helloworld2/
total 0
-rw-r--r-- 1 root root 0 May  4 06:53 cgroup.clone_children
--w--w--w- 1 root root 0 May  4 06:53 cgroup.event_control
-rw-r--r-- 1 root root 0 May  4 06:53 cgroup.procs
-rw-r--r-- 1 root root 0 May  4 06:53 memory.failcnt
--w----- 1 root root 0 May  4 06:53 memory.force_empty
-rw-r--r-- 1 root root 0 May  4 06:53 memory.kmem.failcnt
-rw-r--r-- 1 root root 0 May  4 06:53 memory.kmem.limit_in_bytes
-rw-r--r-- 1 root root 0 May  4 06:53 memory.kmem.max_usage_in_bytes
-r--r--r-- 1 root root 0 May  4 06:53 memory.kmem.slabinfo
-rw-r--r-- 1 root root 0 May  4 06:53 memory.kmem.tcp.failcnt
-rw-r--r-- 1 root root 0 May  4 06:53 memory.kmem.tcp.limit_in_bytes
-rw-r--r-- 1 root root 0 May  4 06:53 memory.kmem.tcp.max_usage_in_bytes
-r--r--r-- 1 root root 0 May  4 06:53 memory.kmem.tcp.usage_in_bytes
-r--r--r-- 1 root root 0 May  4 06:53 memory.kmem.usage_in_bytes
-rw-r--r-- 1 root root 0 May  4 06:53 memory.limit_in_bytes
-rw-r--r-- 1 root root 0 May  4 06:53 memory.max_usage_in_bytes
-rw-r--r-- 1 root root 0 May  4 06:53 memory.memsw.failcnt
-rw-r--r-- 1 root root 0 May  4 06:53 memory.memsw.limit_in_bytes
-rw-r--r-- 1 root root 0 May  4 06:53 memory.memsw.max_usage_in_bytes
-r--r--r-- 1 root root 0 May  4 06:53 memory.memsw.usage_in_bytes
-rw-r--r-- 1 root root 0 May  4 06:53 memory.move_charge_at_immigrate
-r--r--r-- 1 root root 0 May  4 06:53 memory.numa_stat
-rw-r--r-- 1 root root 0 May  4 06:53 memory.oom_control
----- 1 root root 0 May  4 06:53 memory.pressure_level
-rw-r--r-- 1 root root 0 May  4 06:53 memory.soft_limit_in_bytes
-r--r--r-- 1 root root 0 May  4 06:53 memory.stat
-rw-r--r-- 1 root root 0 May  4 06:53 memory.swappiness
-r--r--r-- 1 root root 0 May  4 06:53 memory.usage_in_bytes
```

```
-rw-r--r-- 1 root root 0 May 4 06:53 memory.use_hierarchy
-rw-r--r-- 1 root root 0 May 4 06:53 notify_on_release
-rw-r--r-- 1 root root 0 May 4 06:53 tasks
```

```
root@debian11:~# cgcreate -g cpu:helloworld2
```

```
root@debian11:~# ls -l /sys/fs/cgroup/cpu/helloworld2/
```

```
total 0
```

```
-rw-r--r-- 1 root root 0 May 4 06:54 cgroup.clone_children
-rw-r--r-- 1 root root 0 May 4 06:54 cgroup.procs
-r--r--r-- 1 root root 0 May 4 06:54 cpuacct.stat
-rw-r--r-- 1 root root 0 May 4 06:54 cpuacct.usage
-r--r--r-- 1 root root 0 May 4 06:54 cpuacct.usage_all
-r--r--r-- 1 root root 0 May 4 06:54 cpuacct.usage_percpu
-r--r--r-- 1 root root 0 May 4 06:54 cpuacct.usage_percpu_sys
-r--r--r-- 1 root root 0 May 4 06:54 cpuacct.usage_percpu_user
-r--r--r-- 1 root root 0 May 4 06:54 cpuacct.usage_sys
-r--r--r-- 1 root root 0 May 4 06:54 cpuacct.usage_user
-rw-r--r-- 1 root root 0 May 4 06:54 cpu.cfs_period_us
-rw-r--r-- 1 root root 0 May 4 06:54 cpu.cfs_quota_us
-rw-r--r-- 1 root root 0 May 4 06:54 cpu.shares
-r--r--r-- 1 root root 0 May 4 06:54 cpu.stat
-rw-r--r-- 1 root root 0 May 4 06:54 notify_on_release
-rw-r--r-- 1 root root 0 May 4 06:54 tasks
```

La Commande cgconfigparser

Appliquez le contenu du fichier **/etc/cgconfig.conf** grâce à l'utilisation de la commande **cgconfigparser** :

```
root@debian11:~# cgconfigparser -l /etc/cgconfig.conf
```

```
root@debian11:~# cat /sys/fs/cgroup/memory/helloworld2/memory.limit_in_bytes
36864
```

```
root@debian11:~# cat /sys/fs/cgroup/cpu/helloworld2/cpu.shares  
100
```

Copyright © 2022 Hugh Norris.
