

Sommaire

Découvrez les ressources de votre machine.....	1
Introduction.....	1
Décrire sa configuration matérielle.....	1
Le Bus PCI.....	2
DMI est votre ami.....	4
Hdparm, l'outil indispensable.....	6
LSHW, un outil complet.....	8
Le petit jeu du processeur.....	9
Analyser sa configuration.....	11
SMART.....	11
Ventilation et température.....	11
La méthode ACPI.....	11
La méthode sensors.....	13
Les autres informations utiles et inutiles.....	14
Conclusion.....	15
Liens utiles.....	15
Copyright.....	16

Découvrez les ressources de votre machine

Découvrez les ressources de votre machine
Par Erwan Velu

Note : Cet article a fait l'objet d'une parution dans Linux Magazine. Sur proposition de l'auteur et avec l'accord de l'éditeur, nous le publions ici de manière à alimenter la base documentaire de Léa.

Introduction

Les ordinateurs sont de magnifiques machines, toujours plus performantes, perfectionnées, permettant à leurs heureux possesseurs d'exécuter les derniers jeux à la mode ou des applications toujours plus complexes. Qu'il soit simple utilisateur ou administrateur chevronné, le propriétaire de cette machine utilise un système d'exploitation pour s'affranchir de l'accès et de la configuration de son matériel. Cette suite logicielle permet une abstraction du fonctionnement complexe d'un ordinateur.

Combien connaissent la procédure complète de démarrage d'un ordinateur récent, combien savent à quoi servent l'ACPI, les APICs, la DSDT, le SMART, le DMI ?

Ces acronymes qui peuvent sembler barbares ou méconnus, font références à des technologies que vous utilisez tous les jours sans même le savoir. Le système d'exploitation les configure et tout se passe pour le mieux dans le meilleur des mondes.

Cependant, de temps en temps, un petit grain de sable vient gripper cette belle mécanique, vous empêchant d'utiliser votre fabuleux ordinateur. Il faut alors trouver une solution, comprendre ce qui se passe pour pouvoir corriger le problème. Il faut alors se plonger dans ces technologies, les comprendre pour enfin trouver la bonne solution.

Cette série d'articles sera une introduction à quelques technologies réputées problématiques ou mystérieuses ainsi qu'un résumé des problèmes rencontrés ces dernières années avec leurs solutions. Cette série d'articles n'a pas pour objectif d'être exhaustif dans les sujets abordés, ni dans leur traitement mais plutôt une explication simplifiée mais pas simpliste des technologies et de leur utilisation. Cette première partie sera donc dédiée d'une part à la description de sa configuration matérielle ainsi qu'à l'analyse de son état.

Décrire sa configuration matérielle

Lorsque l'on est confronté à un problème que l'on ne comprend pas, des forums d'entraide, des canaux de discussion (IRC) et des listes de diffusions sont à votre disposition pour vous aider au dépannage. Cependant, lorsque l'on parle de problèmes souvent liés au matériel, il est indispensable d'indiquer la configuration matérielle sur laquelle ces problèmes se posent.

Certains problèmes sont spécifiques à un modèle ou une marque de carte mère et même parfois à la version du BIOS de celle-ci. Plus votre description sera précise, plus vous obtiendrez une réponse rapide de quelqu'un qui reconnaîtra sa configuration et qui aura déjà eu à résoudre ce problème.

Votre ordinateur est composé au minimum d'une carte mère, d'un processeur, de mémoire et d'une carte vidéo auxquels on peut ajouter des cartes additionnelles (cartes son, cartes d'acquisition diverses, cartes modem/adsl, cartes réseau). Les cartes additionnelles sont soit sur des ports d'extension (slot), soit déjà intégrées sur la carte mère mais toutes sont connectées via un bus que l'on nomme PCI (Peripheral Component

Interconnect). Il existe également un port AGP (Accelerated Graphics Port) qui ne peut recevoir qu'une carte graphique tandis que son homologue PCI peut recevoir tout type de carte.

Le Bus PCI

Pour connaître la liste des cartes additionnelles disponibles sur votre machine, utilisez la commande « lspci ».

```
[root@R1 root]# lspci
00:00.0 Host bridge: Intel Corp. 82855PM Processor to I/O Controller (rev 03)
00:01.0 PCI bridge: Intel Corp. 82855PM Processor to AGP Controller (rev 03)
00:1d.0 USB Controller: Intel Corp. 82801DB/DBL/DBM (ICH4/ICH4-L/ICH4-M) USB UHCI Controller #1 (rev 01)
00:1d.7 USB Controller: Intel Corp. 82801DB/DBM (ICH4/ICH4-M) USB 2.0 EHCI Controller (rev 01)
00:1f.0 ISA bridge: Intel Corp. 82801DBM LPC Interface Controller (rev 01)
00:1f.1 IDE interface: Intel Corp. 82801DBM (ICH4) Ultra ATA Storage Controller (rev 01)
00:1f.3 SMBus: Intel Corp. 82801DB/DBL/DBM (ICH4/ICH4-L/ICH4-M) SMBus Controller (rev 01)
00:1f.5 Multimedia audio controller: Intel Corp. 82801DB/DBL/DBM (ICH4/ICH4-L/ICH4-M) AC'97 Audio Controller (rev 01)
00:1f.6 Modem: Intel Corp. 82801DB/DBL/DBM (ICH4/ICH4-L/ICH4-M) AC'97 Modem Controller (rev 01)
01:00.0 VGA compatible controller: nVidia Corporation NV34M [GeForce FX Go 5200] (rev a1)
02:03.0 CardBus bridge: Ricoh Co Ltd RL5c476 II (rev ac)
02:03.2 FireWire (IEEE 1394): Ricoh Co Ltd R5C552 IEEE 1394 Controller (rev 04)
02:05.0 Ethernet controller: Broadcom Corporation BCM4401 100Base-T (rev 01)
02:07.0 Network controller: Intel Corp. PRO/Wireless LAN 2100 3B Mini PCI Adapter (rev 04)
Analysons une ligne :
```

```
01:00.0 VGA compatible controller: nVidia Corporation NV34M [GeForce FX Go 5200] (rev a1)
Chaque carte est connectée sur un bus PCI (le bus numéro 01 dans cet exemple), possède une position sur celui-ci (00.0), une classe (« VGA compatible controller »), une marque, un modèle et une révision de celui-ci.
```

Pour les plus curieux, les modèles, les marques et les classes ne sont en fait qu'une valeur numérique interprétée de manière textuelle par lspci via le fichier « /usr/share/pci.ids ». La commande « lspci -n », permet de connaître la valeur numérique des cartes PCI.

```
01:00.0 Class 0300: 10de:0328 (rev a1)
La valeur 10de est le code international associé à cette marque, 0328 le modèle de puce électronique. La liste quasi complète des identifiants est disponible sur le site http://www.pcidatabase.com/.
```

Par exemple, ces valeurs sont utilisées par un pilote (driver, module) pour savoir si une carte pour lequel il est conçu est présente ou non sur le système. Lorsqu'un constructeur produit une nouvelle carte, il y associe un nouvel identifiant : il faut donc que le nouveau driver contienne cet identifiant pour qu'il puisse être associé à cette carte.

Il est également possible de connaître la fréquence du bus associé à une carte, le numéro de son numéro d'interruption (IRQ), les zones mémoire quelle occupe ainsi que les options qui lui sont associées. La commande « lspci -v » permet de connaître ces informations.

```
01:00.0 VGA compatible controller: nVidia Corporation NV34M [GeForce FX Go 5200] (rev a1) (prog-if 00
```

[VGA])

```
Subsystem: Samsung Electronics Co Ltd: Unknown device c00f
Flags: bus master, 66Mhz, medium devsel, latency 248, IRQ 11
Memory at c8000000 (32-bit, non-prefetchable) [size=16M]
Memory at d8000000 (32-bit, prefetchable) [size=128M]
Expansion ROM at <unassigned> [disabled] [size=128K]
Capabilities: [60] Power Management version 2
```

Capabilities: [44] AGP version 3.0

Dans cet exemple, la fréquence du bus AGP 3.0 (Capabilities [44]) associée à cette carte est de 66Mhz, elle utilise l'interruption numéro 11. On peut noter également qu'une carte possède également une section « Subsystem » qui contient les valeurs « Subvendor » et « Subproduct » qui permettent à l'intégrateur du composant d'inscrire sa marque et son modèle. Cela permet de différencier les cartes en fonction des options commerciales choisies (quantité et type de mémoire, etc..) lors de l'intégration de ce composant.

Ces informations permettent de mieux comprendre le nombre de cartes, leurs marques et leurs modèles, leurs types ainsi que leur configuration. Le programme « lspcidrake » disponible sur les distributions Mandrakelinux permet de connaître le pilote (driver) associé à une carte, il utilise le fichier « /usr/share/ldetect-1st/pcitable » pour effectuer cette association.

```
[root@R1 root]# lspcidrake | grep nVidia
```

```
Card:NVIDIA GeForce FX (generic): nVidia|GeForce FX Go 5200 [DISPLAY_VGA]
```

Remarque : les nouvelles versions du bus PCI connues sous le nom de PCI-X et PCI-Express ne modifient pas ces outils. Seuls les débits, les fréquences et la connectique pour le PCI-Express évoluent.

Nota : Le bus PCI primaire (00) est généralement équipé de ponts (bridges) vers d'autres bus PCI ou le bus AGP. Généralement le bus 01 correspond au bus AGP et 02 aux cartes connectées sur les bus PCI secondaires. La commande « lspci -t » permet de mieux comprendre l'architecture des bus PCI.

```
[root@R1 root]# lspci -t
```

```
-[00]--00.0
  +-01.0-[01]----00.0
  +-1d.0
  +-1d.1
  +-1d.2
  +-1d.7
  +-1e.0-[02]---+03.0
  |           +-03.1
  |           +-03.2
  |           +-05.0
  |           \-07.0
  +-1f.0
  +-1f.1
  +-1f.3
  +-1f.5
```

```
\-1f.6
```

On remarque que le bus [02:] est le prolongement du pont PCI identifié « 1e.0 » correspondant à « 00:1e.0 PCI bridge: Intel Corp. 82801 PCI Bridge (rev 81) » dans la capture précédente.

Maintenant nous connaissons la liste des cartes disponibles sur notre machine. Mais quelle est la marque de ma carte mère, combien de barrettes de mémoire sont installées et sur quel emplacement ? Combien de slots PCI sont libres ? Quelle est la version de mon BIOS ?

A toutes ces questions, on a tendance soit à redémarrer la machine pour consulter les messages du BIOS, soit à ouvrir sa machine pour y récupérer ces informations.

DMI est votre ami

L'organisation DMTF (Distributed Management Task Force) conçoit des technologies pour faciliter la collecte d'informations. DMI (Desktop Management Interface) est un standard décrivant les ressources matérielles d'une machine, il est intégré au BIOS de votre machine. La version 2.3 du standard définit 39 catégories d'informations ("type" en anglais), allant de la marque de la carte mère jusqu'au mode de fonctionnement de votre alimentation. Cette technologie qui fait le lien entre les informations purement matérielles et l'environnement purement logiciel est cependant très mal implémenté par les fabricants informatiques. Il n'est pas rare de trouver des informations non ou mal remplies. Il n'en reste pas moins un outil indispensable pour mieux connaître son matériel.

Un utilitaire nommé « dmidecode » permet d'accéder à ces informations. Cet outil est assez verbeux, il ne sera montré que certaines parties de son affichage afin d'alléger cet article. Une version de dmidecode supérieure à 2.4 est nécessaire pour avoir un affichage de bonne qualité. Les versions précédentes n'affichaient pas toutes les options.

Comment connaître la marque et le modèle de ma carte mère ?

Le « type 2 » de DMI donne la marque de la carte mère ainsi que son modèle.

```
[root@R1] dmidecode
```

```
[...]
DMI type 2, 8 bytes.
Board Information Block
  Vendor: ASUSTeK Computer INC.
  Product: P4S8L
  Version: REV 1.xx
  Serial Number: xxxxxxxxxxxx
```

```
[...]
```

Notez que l'on peut également voir la révision de sa carte mère : par exemple, sur le modèle A7N8X d'ASUSTeK, la version 1 souffrait d'un défaut de conception et pouvait poser problème lors de l'installation / configuration. Grâce à DMI, on peut connaître simplement cette information sans ouvrir la machine.

Comment connaître la version du BIOS de ma carte mère ?

Le « type 0 » contient les informations du BIOS. On y retrouve la marque du constructeur du BIOS, sa version, sa date et sa taille.

```
DMI type 0, 20 bytes.
```

```
BIOS Information Block
  Vendor: Award Software, Inc.
  Version: ASUS P4S8L ACPI BIOS Revision 1005
  Release: 06/11/2003
  BIOS base: 0xF0000
  ROM size: 256K
  Capabilities:
```

Flags: 0x00000007FCBDF80

Cette information peut sembler peu importante pour l'utilisateur mais il faut reconnaître que les BIOS ne sont pas d'une qualité irréprochable et il est souvent nécessaire de les remettre à jour pour corriger des bugs. Il vous sera donc possible de fournir facilement cette information lors de vos discussions au sujet de vos problèmes. Les copies d'écran sont relativement faciles à comprendre, il ne semble pas nécessaire de les commenter plus.

Combien de barrettes mémoires sont installées sur ma machine ?

Le « type 17 » de DMI, nous donne le nombre de slots mémoire disponibles et leur configuration. Le « type 6 » donne le même type d'information mais il est obsolète depuis la version de 2.0 de DMI et n'est pas disponible sur tous les systèmes.

Handle 0x001F

DMI type 17, 23 bytes.

```
Memory Device
  Array Handle: 0x001E
  Error Information Handle: No Error
  Total Width: Unknown
  Data Width: Unknown
  Size: No Module Installed
  Form Factor: DIMM
  Set: 1
  Locator: DIMM 1
  Bank Locator: Not Specified
  Type: DRAM
  Type Detail: Synchronous
  Speed: Unknown
```

Handle 0x0020

DMI type 17, 23 bytes.

```
Memory Device
  Array Handle: 0x001E
  Error Information Handle: No Error
  Total Width: 64 bits
  Data Width: 64 bits
  Size: 256 MB
  Form Factor: DIMM
  Set: 2
  Locator: DIMM 2
  Bank Locator: Not Specified
  Type: DRAM
  Type Detail: Synchronous
```

Speed: Unknown

Dans cet exemple, la machine possède deux slots de mémoire, dont un seul est utilisé. Le type de mémoire utilisé est « SDRAM » avec une taille de 256 Méga Octets.

Combien de slots PCI sont utilisés ?

Il peut être pratique de connaître le nombre de slots PCI utilisés et quels sont leurs caractéristiques. La commande « lspci » ne permet de voir que les cartes PCI connectées mais pas les slots associés. Dans cet exemple, la machine possède 2 slots PCI dont un seul est utilisé. Le type « short » indique qu'il s'agit qu'un slot PCI 32-bit.

Handle 0x001B

Hardware—hard plus—matos

```
DMI type 9, 13 bytes.
System Slot Information
  Designation: PCI 1
  Type: 32-bit PCI
  Current Usage: Available
  Length: Short
  ID: 0
  Characteristics:
    5.0 V is provided
    3.3 V is provided
    PME signal is supported
Handle 0x001C
DMI type 9, 13 bytes.
System Slot Information
  Designation: PCI 2
  Type: 32-bit PCI
  Current Usage: In Use
  Length: Short
  ID: 0
  Characteristics:
    5.0 V is provided
    3.3 V is provided
```

PME signal is supported

DMI nous permet également de connaître le nombre de ports PCI libres ainsi que leur configuration. Cela peut se révéler très pratique sur un serveur distant en vue d'une intervention : ajouter une carte réseau par exemple.

Cette présentation reste succincte mais DMI vous permet de récolter encore quelques indices sur la configuration de votre machine : la description du processeur, des ports séries, parallèles, la dernière méthode de boot utilisée (réseau ou local).

Hdparm, l'outil indispensable

Les disques IDE sont capables de donner une description très précise de leur configuration. L'outil « hdparm » plus souvent connu pour permettre l'optimisation des disques durs (nous y reviendrons dans un prochain article) permet également de consulter les informations d'un disque. L'option « -i » suivie du nom du périphérique (hda pour le premier disque IDE du premier contrôleur IDE, hdb pour le second, hdc pour le premier disque IDE du second contrôleur IDE etc..).

```
[root@R1 root]# hdparm -i /dev/hda
```

```
/dev/hda:
Model=FUJITSU MHT2040AT, FwRev=0022, SerialNo=NN77T3C13KB9
Config={ HardSect NotMFM HdSw>15uSec Fixed DTR>10Mbs }
RawCHS=16383/16/63, TrkSize=0, SectSize=0, ECCbytes=4
BuffType=DualPortCache, BuffSize=2048kB, MaxMultSect=16, MultSect=16
CurCHS=16383/16/63, CurSects=16514064, LBA=yes, LBASects=78140160
IORDY=yes, tPIO={min:240,w/IORDY:120}, tDMA={min:120,rec:120}
PIO modes: pio0 pio1 pio2 pio3 pio4
DMA modes: mdma0 mdma1 mdma2
UDMA modes: udma0 udma1 udma2 udma3 udma4 *udma5
AdvancedPM=yes: mode=0x80 (128) WriteCache=enabled
Drive conforms to: ATA/ATAPI-6 T13 1410D revision 3a:
```

* signifie the current active mode

La première ligne vous indiquera la marque, le modèle et le numéro de série de votre disque dur.

Hardware—hard plus—matos

L'option « `bufferType=` » renvoie d'une part le type de mémoire cache utilisée par votre disque dur mais également sa taille « `buffSize=` ». Cette information se révèle très intéressante car généralement une taille de mémoire cache élevée sur un disque dur permet d'en augmenter les performances. Lorsque le kernel demande au disque dur de lire une certaine quantité de données sur celui-ci, le disque dur en profite pour lire quelques secteurs en avance et les stocke dans sa mémoire cache. Ainsi, lors de la prochaine demande d'accès au disque, si celui-ci possède déjà les données dans sa mémoire cache, il pourra directement les transmettre au noyau sans lire sur le disque : le gain de performance est dans ce cas énorme car il n'est pas nécessaire de déplacer la tête du disque dur pour accéder à l'information. Cet exemple montre une taille classique de mémoire cache de 2 MO. Cependant certains disques IDE proposent une taille de 8 MO.

Enfin, le disque dur va lister les modes de fonctionnement dont il dispose, généralement de PIO0 jusqu'à UDMA6. Originellement, le mode PIO (Programmed Input/Output) réalisait le transfert de données en utilisant les ressources processeur (environ 40%). Le débit était donc fonction de la performance et du taux d'utilisation de votre processeur. Le mode DMA (Direct Memory Access) a donc logiquement supplanté cette méthode vieillissante. Celui-ci réalise les opérations d'entrées/sorties IDE (I/O) sans utiliser le processeur. Les performances ne sont pas meilleures mais elles sont beaucoup plus stables en cas d'utilisation intensive du processeur. Ces deux modes fonctionnent entre 2MO et 16MO par seconde en fonction des normes.

Sur du matériel récent (moins de 5 ans), votre disque devrait se comporter en mode Ultra DMA (UDMA) car les modes PIO & DMA sont aujourd'hui obsolètes. L'étoile située devant un mode permet d'indiquer le mode actuel (udma5 dans cet exemple). Le tableau ci—contre indique les correspondance de vitesse maximale par norme.

Type de fonctionnement	Vitesse Maximale en MO/sec
UltraDMA 0	16,6
UltraDMA 1	25
UltraDMA 2	33,3
UltraDMA 3	44,4
UltraDMA 4	66,6
UltraDMA 5	100
UltraDMA 6	133,33

Pour les plus curieux, on notera que la géométrie du disque annoncée par le disque dur est disponible. Celle-ci représente le nombre virtuel de cylindres/têtes/secteurs disponibles sur ce disque : le nombre réel n'est plus accessible depuis quelques années, le firmware de chaque disque fait ensuite la correspondance entre la configuration virtuelle et la configuration réelle. Chaque plateau d'un disque dur est composé de pistes sur lesquelles sont disposés les secteurs (512 octets chacun). Les pistes extérieures contiennent plus de secteurs que les pistes intérieures du disque. Le nombre de secteurs par piste varie donc en permanence entre le début et la fin du disque. Pour faciliter l'accès aux données, le disque virtualise une géométrie dans laquelle le nombre de secteurs par piste est fixe. Lorsque le système demande l'accès à un endroit précis du disque, le firmware va transposer cette demande pour atteindre la bonne piste et le bon secteur où est stockée l'information.

Il sera également possible de connaître la configuration IDE d'un lecteur/graveur de CD/DV D avec cette même commande. Attention, certains contrôleurs SATA utilisent une émulation SCSI ce qui empêchera l'utilisation de l'option « `-i` » sur ceux-ci. Le système de fichier virtuel « `/proc` » contient un répertoire `ide/` dans lequel vous retrouverez ces informations. Le répertoire « `/proc/ide/ide0/hda/` » contiendra les fichiers `cache`, `capacity`, `geometry`, `media`, `model` qui correspondent respectivement à la taille de la mémoire cache du

disque, sa capacité en nombre de secteurs, sa géométrie, le type de media (disque ou cédérom) et enfin son modèle.

Tester les performances d'un disque dur :

vous trouvez de votre disque n'est pas très performant, vous aimeriez savoir quelle est la performance de votre disque dur, la commande « `-t` » permet de tester la vitesse de lecture depuis les premiers secteurs du disque : ce sont les secteurs les plus rapides. Ce test réalise une lecture séquentielle pendant 3 secondes; cela veut dire que l'on ne déplace quasiment pas la tête de lecture du disque dur, on obtient ainsi les performances maximales en lecture. Dans la « vraie vie », le disque dur ne travaille que rarement en mode séquentiel car les données demandées par le système d'exploitation sont rarement placées de manière contiguë sur le disque dur donc la tête de lecture doit se déplacer ce qui dégrade les performances. Cette mesure ne doit pas donc pas être prise comme valeur moyenne de performance mais bien comme une valeur maximale.

```
[root@R1 root]# hdparm -t /dev/hda
/dev/hda:
```

```
Timing buffered disk reads: 84 MB in 3.05 seconds = 27.54 MB/sec
```

Dans cet exemple, la vitesse maximale est de 27Mo par seconde ce qui reste respectable pour un disque dur d'ordinateur portable. Une station de travail ou un poste bureautique récent devrait atteindre les 45MO par seconde sans trop de problèmes.

LSHW, un outil complet

Pour faire un inventaire complet de sa machine, il faut mélanger les outils tels que `dmidecode`, `lspci`, `dmesg`, `hdparm` ou bien le contenu du répertoire `/proc` voire `/sys` sur les kernels de la série 2.6.x. Un outil intègre intelligemment toutes ces informations et les affiche soit sous forme texte soit sous forme d'arbres xml/html.

`Lshw` (pour HardWare LiSter) est à ce jour l'outil le plus complet sous Linux pour l'inventaire exhaustif d'un ordinateur. Le mélange des sources d'informations permet un niveau d'information très détaillé. Présenté sous la forme d'un arbre, il affiche aussi bien les informations DMI que les paramètres de votre réseau ou bien encore le numéro de série et la configuration de votre disque dur.

L'exemple de la carte réseau est typique de ce mélange d'informations. On y retrouve la marque et le modèle aussi bien que son adresse MAC, son débit maximal ou bien encore son mode de fonctionnement.

La carte réseau Intel Ethernet Pro 100 utilise la version 1.09 du driver `eepro100`, l'interruption numéro 12, son adresse MAC est « `00:d0:b7:23:f0:70` » et utilise l'autonégociation pour obtenir la vitesse de 100Mbps en full duplex. Bien entendu, l'adresse IP est également affichée (« `192.168.1.253` ») : l'option « `link` » indique que le câble est correctement branché. Pour les plus pointilleux, on sait également qu'elle utilise le port PCI `00:13.0` : on peut donc en déduire grâce au « `00:` » qu'il s'agit d'une carte directement branchée sur le PCI primaire de la carte mère.

```
*-network:1 description: Ethernet controller product: 82557/8/9 [Ethernet Pro 100] vendor: Intel Corp.
physical id: 13 bus info: pci@00:13.0 logical name: eth1 version: 08 serial: 00:d0:b7:23:f0:70 size: 100Mbps
clock: 33MHz capabilities: bus_master cap_list mii autonegotiation 100bt-fd 100bt 10bt-fd 10bt ethernet
physical configuration: autonegotiated=100bt broadcast=yes driver=eepro100
driver version=eepro100.c:v1.09j-t 9/29/99 Don duplex=full ip=192.168.1.253 link =yes multicast=yes
resources: iomemory:e1800000-e1800fff ioport:8400-843f iomemory:e1000000-e10ffff irq:12
```

Le deuxième effet `lshw` réside dans le format d'exportation permettant ainsi une réutilisation aisée de cette

Hardware—hard plus—matos

sortie. Le mode par défaut, en texte, permet une consultation depuis n'importe quelle console qu'elle soit locale ou distante (via ssh par exemple) et permet également de scripter facilement la récupération d'informations. La sortie au format HTML (option `-html`), permet directement de consulter le résultat via un serveur Web : cela peut se révéler très pratique pour décrire une machine dans une page personnelle par exemple. Enfin, la sortie XML (option `-xml`) permet à d'autres applications de parcourir son contenu pour l'intégrer plus efficacement.

Nota: depuis la version 2.0, lshw propose également une interface graphique nommée « `gtk-lshw` » et conçue en GTK pour rendre cet affichage plus attractif.

Un grand merci à Lyonel Vincent pour sa réactivité et ce travail remarquable mais encore trop méconnu à mon goût.

Le petit jeu du processeur

Ce cas est tiré d'un cas réel qui m'a été posé mais toute ressemblance avec des faits ayant existés serait bien entendu fortuite :-b. Les processeurs de la game Pentium IV/Xeon d'Intel possèdent la technologie Hyper Threading (HT) qui permet d'avoir l'équivalent de deux processeurs avec un seul processeur physique. Lorsque l'on regarde les informations processeur sous Linux avec la commande « `cat /proc/cpuinfo` », il est affiché deux processeurs.

La question qui se pose : est-ce un mono-processeur avec la technologie HT activée ou bien est-ce un système bi-processeur sans la technologie HT ? Cette question peut paraître bien futile mais le défi qu'elle propose mérite d'être relevé, c'est un petit pas de plus dans la compréhension de son système.

Voici le résultat de la commande « `cat /proc/cpuinfo` » d'une machine.

```
[erwan@n4 ~]$ cat /proc/cpuinfo
```

```
processor       : 0
vendor_id     : GenuineIntel
cpu family    : 15
model        : 2
model name    : Intel(R) Xeon(TM) CPU 2.80GHz
stepping     : 7
cpu MHz      : 2801.184
cache size   : 512 KB
physical id  : 0
siblings     : 2
fdiv_bug     : no
hlt_bug      : no
f00f_bug     : no
coma_bug     : no
fpu          : yes
fpu_exception : yes
cpuid level  : 2
wp           : yes
flags        : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov
pat pse36 clflush dts acpi mmx fxsr sse sse2 ss ht tm pbe cid lm
bogomips    : 5537.79

processor       : 1
vendor_id     : GenuineIntel
cpu family    : 15
model        : 2
```

Hardware—hard plus—matos

```
model name      : Intel(R) Xeon(TM) CPU 2.80GHz
stepping       : 7
cpu MHz        : 2801.184
cache size     : 512 KB
physical id    : 0
siblings       : 2
fdiv_bug      : no
hlt_bug       : no
f00f_bug      : no
coma_bug      : no
fpu           : yes
fpu_exception  : yes
cpuid level    : 2
wp            : yes
flags         : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov
pat pse36 clflush dts acpi mmx fxsr sse sse2 ss ht tm pbe cid lm
```

bogomips : 5586.94

On peut noter que le champ « flags » contient la valeur HT ce qui signifie que le processeur associé supporte cette fonctionnalité mais cela ne veut pas dire qu'elle est activée. Le champ « siblings » nous indique le nombre total de processeurs logiques disponibles sur un processeur physique. Ainsi, le processeur 0 et le processeur 1 sont dans un environnement avec 2 processeurs logiques par processeur physique : cela nous indique donc que l'hyperthreading est activé.

Le dernier élément qui nous donnera une solution fiable est le champ « physical id » qui identifie avec un numéro unique chaque processeur physique. Les deux processeurs possèdent le même numéro de processeur physique avec un nombre de processeurs virtuels égal à 2. Nous sommes donc en présence d'un système mono-processeur avec l'Hyperthreading activé.

Si un deuxième processeur « Hyper-Threadé » était présent nous aurions des processeurs numérotés 2 et 3 (champ processor) avec des numéros de processeurs physiques identiques. Le nombre de "siblings" sera quant à lui toujours égal à 2 puisque ce processeur physique possède deux processeurs logiques.

Le cas de processeurs x86-64 : voici également une petite astuce qui peut s'avérer pratique.

La société Intel vient d'ajouter à sa gamme Xeon un processeur à technologie X86-64 pour concurrencer les processeurs de la société rivale AMD. Appelée Nocona ou Xeon EM64T d'un côté et Opteron ou Athlon 64 de l'autre, cette technologie permet d'adresser plus de mémoire et d'accélérer les calculs d'applications gourmandes. Dans le cas d'AMD, il est facile de savoir si on utilise un tel processeur car le nom de celui-ci apparaît dans le fichier virtuel « /proc/cpuinfo ». Cependant, il est très difficile de différencier un processeur de type XEON normal d'un processeur XEON EM64T.

Vous venez de recevoir deux machines identiques mais on vous a dit qu'une des deux est un processeur EM64T. Pour identifier à coup sûr les deux machines, l'astuce consiste à chercher dans la liste du champ « flags » l'acronyme « lm » pour « Long Mode ». Sa présence mettra en évidence un processeur à base de technologie x86-64. Comme vous pouvez le constater, le processeur donné en exemple précédemment possède cet attribut, nous sommes donc en présence d'un processeur XEON EM64T.

Vous savez donc maintenant comment décrire votre configuration préférée et en découvrir les informations les plus intimes. Le nouveau défi qui se présente à vous c'est d'être capable d'en analyser l'état.

Analyser sa configuration

Quelques outils vous permettent d'analyser la configuration matérielle dans laquelle votre système se trouve. Les quelques outils présentés ici vous permettront de connaître l'état de votre disque dur, la température de votre machine ou bien encore la vitesse de vos ventilateurs.

SMART

La technologie SMART (Self-Monitoring, Analysis and Reporting Technology systems) permet de connaître et d'analyser l'état physique de vos disques durs. Elle est disponible sur les disques dur depuis la fin des années 90, et permet de les interroger afin d'obtenir une liste d'attributs (une caractéristique physique) et les valeurs associées. On y retrouve la température du disque, le taux d'erreurs de lecture/écriture, le nombre de cycles d'allumage effectués etc...

L'outil smartmontools permet de suivre l'état de ces attributs afin de déceler une défaillance à venir du disque.

La commande « smartctl -A /dev/hdx » (où x représente votre disque dur) permet de consulter les valeurs courantes du disque. La commande « smartctl -t short /dev/hdx » permet d'effectuer un test rapide du disque pour y déceler des erreurs : on pourra remplacer la valeur « short » par « long » pour un test plus complet.

Enfin, le fichier de configuration « /etc/smartd.conf » permet de configurer le comportement par défaut du démon « smartd » qui aura pour rôle de tester régulièrement l'état de vos disques et de vous avertir en cas de problème.

```
DEVICESCAN -H -l error -l selftest -t -m a@a.com -M exec /bin/mail -s (S/../../02|L/../../6/03)
```

L'ajout de cette ligne dans le fichier de configuration enverra un mail à l'adresse a@a.com si l'un des tests réalisés lors d'un test court (S/) tous les jours à 2 heures du matin, ou d'un test long tous les samedi à 3 heures du matin révèle un problème.

Pour plus d'informations à propos de SMART, consultez l'article présent dans le numéro 68 de Linux Magazine France : il détaille la configuration complète ainsi que le mode de fonctionnement de cette technologie. Bien entendu, les pages de man ainsi que les sites Web et les forums sont à votre disposition pour y trouver une aide complémentaire.

Ventilation et température

Votre machine est instable, elle se bloque complètement de temps en temps ? Elle est peut-être mal ventilée ou bien votre alimentation est un peu sous-dimensionnée (il faut reconnaître que les dernières carte accélératrice 3D sont très gourmande en énergie). Une seule solution pour le vérifier, utiliser les informations fournies par votre machine pour observer les vitesses de rotation de vos ventilateurs, la valeur des tensions de votre alimentation ou bien encore la température de votre processeur et de votre carte mère.

Il existe deux catégories d'outils, ceux qui utilisent les informations fournies par l'ACPI et ceux qui utilisent directement les composants électroniques en charge de la surveillance thermique généralement appelés « sensors ».

La méthode ACPI

Pour rappel, l'ACPI (Advanced Configuration and Power Interface) est un standard qui définit l'interaction

Hardware—hard plus—matos

entre le système d'exploitation et le BIOS pour la gestion de l'énergie et la configuration matérielle. Il remplace le vieillissant APM (Advanced Power Management) et permet ainsi de consulter ou de piloter des événements matériels. L'ACPI est généralement très utile sur les ordinateurs portables mais on peut également en trouver l'utilité sur une machine personnelle ou même un serveur.

La gestion de l'ACPI est très dépendante de la manière dont le BIOS a été programmé et par expérience leur qualité est très souvent plus que limite : non respect des normes, bugs etc.. Il faut donc parfois utiliser des contournements pour faire fonctionner correctement l'ACPI. Le kernel Linux permet quatre options de contrôle sur l'ACPI : on, off, force, ht.

A l'aide de l'option de boot « acpi= ». Vous pouvez changer ce mode en éditant la configuration de votre « bootloader » favori (/etc/lilo.conf pour lilo, /boot/grub/menu.lst pour grub) en ajoutant cette option sur la ligne « append= ».

Le mode « on » active l'ACPI si le kernel trouve un BIOS compatible ACPI, le mode « off » désactive l'utilisation de l'ACPI même si le BIOS est compatible. Le mode « ht » équivaut au mode « off » excepté pour la partie HyperThreading du processeur. Ainsi, un mode « off » ne permet plus d'utiliser la fonctionnalité HyperThreading de votre processeur alors que les options « on » et « ht » le permettent. Le dernier mode, « force », permet quant à lui d'ignorer que votre BIOS indique qu'il ne supporte pas l'ACPI (BIOS buggé par exemple) mais que l'on souhaite quand même utiliser les ressources disponibles. Ce mode reste à utiliser avec parcimonie et en dernier recours.

Ce petit rappel sur ACPI étant fait, assurez-vous que l'ACPI est bien activé sur votre kernel en vérifiant le résultat de la commande « cat /proc/acpi/info ». La version du code ACPI intégré dans le kernel apparaît.

```
[root@localhost ~]# cat /proc/acpi/info
```

```
version:                20041105
```

```
[root@localhost ~]#
```

L'ACPI est donc activé, il faut donc ensuite insérer quelques modules dans votre kernel pour accéder aux informations matérielles. Le service « acpi » de votre distribution favorite devrait se charger de ce travail sinon vous pouvez ajouter les modules suivants à l'aide de la commande modprobe : thermal, fan, ac, battery, button, processor.

Chaque module se charge de lire la configuration de la table ACPI de votre bios pour les afficher de manière plus « humaine ». Le répertoire /proc/acpi contient maintenant des sous-répertoires nommés respectivement thermal_zone, fan, ac_adapter, battery, button, processor. Chacun contient des fichiers correspondant à la configuration matérielle lue depuis la table ACPI.

Attention, comme dit précédemment la qualité des bios étant plus qu'approximative, sur certaines cartes mères, même une fois les modules insérés dans le kernel, les répertoires restent vides. Cela signifie que le kernel n'arrive pas à parcourir correctement la table ACPI de votre machine. Votre BIOS semble donc non conforme à la norme ACPI. Dans certains cas, la table DSDT qui décrit les ressources ACPI est non conforme et il faudra la recompiler pour avoir une table conforme. Cette étape sera détaillée dans le prochain article qui abordera les différentes manières de dépanner un système. Dans d'autres cas, la table est incomplète, l'utilisation de l'ACPI pour l'accès aux ressources matérielles ne sera pas possible. L'implémentation de l'ACPI reste cependant de meilleure qualité sur les ordinateurs portables que sur les ordinateurs grand public.

Vous faites partie des chanceux : vous avez un bios de bonne qualité, votre répertoire /proc/acpi contient donc quelques informations que vous allez pouvoir exploiter pour analyser l'état de votre machine.

Hardware—hard plus—matos

La température de votre machine peut être lue dans le répertoire « /proc/acpi/thermal_zone/THRM ». Le fichier « temperature » indique la température actuelle en degrés celsius, le fichier « trip_points » les valeurs limites.

```
[root@R1 root]# cat /proc/acpi/thermal_zone/THRM/temperature /proc/acpi/thermal_zone/THRM/trip_points
```

```
temperature:          43 C
critical (S5):        120 C
active[0]:            65 C: devices=0xdff23700
```

```
[root@R1 root]#
```

Vous avez déjà sûrement rencontré les sigles S1,S2,S3,S4,S5 : ils définissent les états d'économie d'énergie. Dans cet exemple, l'arrêt de la machine (état S5) sera déclenché à 120° ! Cette valeur semble démesurée quand on sait qu'un processeur plantera votre système à partir de 90°C. Les chemins indiqués peuvent changer en fonction du matériel. Certains modules ACPI, pour les portables notamment, utilisent des répertoires spécifiques à une marque (Toshiba ou Asus par exemple).

Nous reviendrons sur ces aspects de configuration et l'optimisation de l'ACPI de manière plus détaillée dans le prochain article.

La méthode sensors

Si la méthode ACPI ne vous donne pas entière satisfaction, la méthode sensors devrait vous permettre d'arriver à vos fins. Les cartes mère modernes implémentent des composants électroniques permettant de connaître la température, la vitesse de rotation des ventilateurs ainsi que les tensions de l'alimentation. Le programme lm_sensors associé aux modules noyau vont en permettre l'accès depuis le système d'exploitation. Installez lm_sensors avec votre installateur de paquets favori (quoi j'ai dit urpmi ?).

Le programme « sensors—detect » va permettre l'auto—détection des composants disponibles sur votre carte mère. Assurez—vous au préalable que le module « i2c—dev » est bien inséré dans votre kernel.

La détection va essayer de lister des composants via les bus I2C, SMBus et enfin l'ISA (oui, oui vous avez bien lu « ISA », il y en a encore sur vos cartes mères mais seulement en interne).

En fonction de votre matériel, le script « sensors—detect » vous proposera de tester des configurations différentes difficiles à généraliser dans un article. Le script essaiera tous les modules disponibles et vous proposera à la fin de générer un fichier « /etc/rc.d/init.d/lm_sensors » qui sera utilisé par le script d'initialisation de lm_sensors. Une fois la détection terminée, lancez le script d'initialisation.

```
[root@localhost ~]# service lm_sensors start
```

```
Chargement des modules pour le moniteur matériel :
Lancement du moniteur matériel sensord [ OK ]
```

```
[root@localhost ~]#
```

La commande « sensors » affichera les valeurs « lues » sur les différents composants trouvés sur la carte mère.

```
asb100—i2c—2—2d
```

```
Adapter: SMBus nForce2 adapter at 5500
VCore 1:   +1.68 V (min = +1.57 V, max = +1.73 V)
+3.3V:     +3.26 V (min = +3.14 V, max = +3.47 V)
```

Hardware—hard plus—matos

```
+5V:          +4.92 V (min = +4.76 V, max = +5.24 V)
+12V:         +11.43 V (min = +10.82 V, max = +13.19 V)
-12V (reserved):
              -11.94 V (min = -0.00 V, max = -0.00 V)
-5V (reserved):
              -5.01 V (min = -0.00 V, max = -0.00 V)
CPU Fan:      4560 RPM (min = 1997 RPM, div = 4)
Chassis Fan:
              2973 RPM (min = 3994 RPM, div = 2)
Power Fan:    3013 RPM (min = 3994 RPM, div = 2)
M/B Temp:     +46°C (high = +45°C, hyst = +40°C)
CPU Temp (Intel):
              +24°C (high = +60°C, hyst = +50°C)
Power Temp:
              +126°C (high = +45°C, hyst = +40°C)
CPU Temp (AMD):
              +25°C (high = +60°C, hyst = +50°C)
```

vid: +1.650 V (VRM Version 9.0)

Les valeurs affichées par `lm_sensors` sont généralement assez difficiles à bien interpréter. Le montage électronique du composant peut impacter le calcul de « `lm_sensors` ». Dans cet exemple, la température de 126° dans l'alimentation semble fausse. Le reste des informations semble crédible. Ce programme utilise les modules noyau de chaque composant dont la configuration sera réinitialisée. Dans certains cas, la configuration va donc être modifiée et rendre les valeurs totalement absurdes voire empêcher la thermorégulation automatique de certains ventilateurs.

Le programme `mbmon` (paquet `xmbmon`) permet de lire les informations des bus SMBus, ISA ou le composant VIA686 sans le réinitialiser. Les valeurs obtenues sont souvent assez proches de la réalité physique. Son utilisation reste cependant assez spartiate.

```
[root@localhost ~]# mbmon -c 1
```

```
Temp.= 47.0, 25.0, 26.0; Rot.= 4500, 3000, 3013
Vcore = 1.68, 1.68; Volt. = 3.26, 4.92, 11.43, -10.46, -4.51
```

```
[root@localhost ~]#
```

Un parsing simple à base de `perl`, `sed` ou `awk` vous permettra rapidement d'isoler la valeur qui vous intéresse. Il est ensuite possible d'utiliser des outils tels que `rrdtools` ou même `ganglia` pour en observer l'évolution dans le temps. L'utilitaire « `gkrellm` » permet d'afficher en permanence et de manière graphique les valeurs des différents capteurs disponibles via `lm_sensors`.

Les autres informations utiles et inutiles

Le répertoire « `/proc/acpi` » permet également de lire quelques informations intéressantes.

Le fichier « `/proc/acpi/battery/BAT1/info` » contient les informations détaillées de votre batterie d'ordinateur portable.

```
[guibo@guibux guibo]$ cat /proc/acpi/battery/BAT1/info
```

```
present:          yes
design capacity:   4800 mAh
last full capacity: 4600 mAh
battery technology: rechargeable
design voltage:    11100 mV
```

Hardware—hard plus—matos

```
design capacity warning: 0 mAh
design capacity low:     138 mAh
capacity granularity 1: 1 mAh
capacity granularity 2: 1 mAh
model number:
serial number:
battery type:           LION
```

OEM info: SAMSUNG Electronics

On y retrouve donc la capacité/tension maximale (design capacity/voltage), la capacité atteinte lors de la dernière charge complète (last full capacity) ainsi que les valeurs limites qui déclencheront les messages d'alertes indiquant que la machine va bientôt s'arrêter faute d'énergie. Il est également possible de savoir si l'écran de votre ordinateur portable est ouvert ou non via la fichier « `proc/acpi/button/lid/LID0/state` ».

Conclusion

Il existe toujours plusieurs méthodes et sources pour accéder aux informations matérielles de votre machine. Vous voici maintenant équipé de quelques éléments de plus dans votre boîte à outils pour décortiquer votre configuration. Cela permet souvent de mieux la comprendre et d'appréhender les problèmes quelle peut générer.

Dans le prochain article de cette série, il sera question du diagnostic et de la correction des problèmes matériels les plus courants ainsi que des éléments d'optimisation des performances. Nous y retrouverons donc comment corriger ou contourner les problèmes liés aux technologies ACPI & APIC, l'utilisation d'outils pour valider votre configuration (memtest, cpuburn, tiobench, mii-tool, ...) et enfin les techniques d'optimisation des performances (hdparm, setpci, cpufreq, options kernel).

Liens utiles

- Le projet dmidecode : <http://www.nongnu.org/dmidecode/>
- Le projet lshw : <http://ezix.sourceforge.net/software/lshw.html>
- Le projet Smartmontools : <http://smartmontools.sourceforge.net/>
- Le projet xmbmon : <http://www.nt.phys.kyushu-u.ac.jp/shimizu/download/download.html>
- Le projet lm_sensors : <http://secure.netroedge.com/~lm78/>
- Tout savoir sur l'ACPI : <http://acpi.sourceforge.net/>

Je remercie Nicolas Planel pour son expérience et ses remarques toujours précieuses, Antoine Giniès pour quelques captures, la « girafe » pour ses corrections et remarques judicieuses, les habitués du canal #mandrakefr ainsi que ceux qui m'ont permis de découvrir les méandres de l'architecture PC et plus particulièrement mes parents qui m'ont offert mes premiers ordinateurs.

Cette page est issue de la documentation 'pré-wiki' de Léa a été convertie avec HTML::WikiConverter. Elle fut créée par Erwan Velu le 06/08/2005.

Copyright

Copyright © 06/08/2005, Erwan Velu



*Ce document est publié sous licence Creative Commons
Attribution, Partage à l'identique, Contexte non commercial 2.0 :*
<http://creativecommons.org/licenses/by-nc-sa/2.0/fr/>