A nouvelle année, nouvelles résolutions ! Paru dans GNU/Linux Magazine numéro 68, un premier article dédié à la découverte des outils disponibles pour la récupération des configurations de votre machine promettait une suite. Je vous propose donc de reprendre ce voyage au centre de votre ordinateur.

Cet article sera articulé autour du diagnostic de pannes et des correctifs associés mais aussi de « trucs & astuces » permettant d'optimiser votre configuration.

1 Savez-vous planter les choux ?

Les choux peut-être pas, mais les ordinateurs je pense que c'est arrivé au moins une fois à chaque lecteur qui lit cet article. Rien de plus énervant qu'un écran qui se fige, le pointeur de souris qui ne se déplace plus, le clavier qui ne répond plus non plus... Planté ! Ce constat est assez facile à établir, mais en trouver la cause reste un problème nettement plus complexe tellement le nombre de causes possibles est important. Deux solutions : ignorer le problème et relancer le système en se disant que ça « passera » ou alors chercher la cause du problème en essayant de trouver une méthode reproduisant le « plantage » tant redouté.
La première solution fonctionne assez bien mais si l'on affaire à un « vrai » problème de configuration matérielle ou logicielle, on arrivera assez vite à ne plus pouvoir éviter la seconde, voire la troisième. J'allais oublier cette troisième solution destinée aux plus riches d'entre vous : changer de matériel. Cette solution coûteuse peut parfois se révéler efficace, parfois pas du tout. A quoi bon changer un composant sans savoir si c'était bien celui-ci qui posait problème ?
Quelle que soit la solution retenue, il sera de toute manière nécessaire de localiser la panne pour y apporter la réponse adéquate. Certaines sociétés ou certains particuliers se sont spécialisés dans ce dépannage ou l'utilisateur est dépassé par les événements.
Cet article vous fera, je l'espère, découvrir quelques outils et méthodes pour valider un par un les composants de votre ordinateur. Cela vous permettra, j'espère, d'être un peu moins dépourvu quand un problème surviendra. L'article reproduira les étapes qu'il faut suivre pour isoler le « fautif ».

1.1 Les causes possibles d'un « plantage »

On retrouve deux grandes catégories de pannes provoquant des dysfonctionnements importants sur un système informatique.

- Une panne du matériel.
- Une erreur logicielle au niveau du noyau et des pilotes (drivers).

Il faudra donc être capable de séparer la panne matérielle qui « force » l'ordinateur à entrer dans un mode de fonctionnement non prévu et instable, de l'erreur logicielle qui équivaut à demander à l'ordinateur de faire « n'importe quoi » et qui va donc se « planter ».
Le terme « planter » utilisé ici équivaut à un arrêt du système à cause d'erreurs sévères : instruction illégale ou composants défaillants. Le terme de « plantage » est aussi utilisé pour décrire le fonctionnement d'une application qui se ferme inopinément. Ce cas est plus souvent lié à une erreur de programmation de l'application qu'à une erreur du système. Nous ne parlerons donc pas de ce second cas dans cet article.

1.2 La mémoire qui flanche ?

Une cause « classique » des dysfonctionnements se situe au niveau de la gestion de la mémoire de votre machine. Je ne parle pas d'allocation mémoire réalisée par votre système d'exploitation favori mais bien de la manière avec laquelle la carte mère configure et utilise vos barrettes de mémoire. Les applications chargent en mémoire des données et des instructions destinées au processeur. Voici un scénario qui peut provoquer l'arrêt de la machine.
Votre application favorite est lancée depuis la ligne de commande ou une interface graphique via un clic de souris. Votre système d'exploitation va donc charger ce programme en mémoire puis en exécuter les instructions. Au plus bas niveau de votre système, le processeur exécute des instructions en langage machine (assembleur) présentes en mémoire.
Malheureusement, votre mémoire est défaillante et elle n'arrive pas à maintenir un état stable des informations qui lui sont confiées. Par exemple, la lettre C est représentée par le code 67

ou 1000011 en binaire. Si la zone mémoire change d'état, alors le contenu de la valeur stockée à cet endroit changera : on pourrait se retrouver avec 1000010. Un seul bit a changé, mais on passe de la lettre « C » (67) à la lettre « B »(66). Le résultat ne se fait pas attendre, le processeur se retrouve à exécuter des instructions qui n'ont plus de sens, provoquant l'arrêt de la machine.

Fig. 1 : Memtest en action

Les pannes de mémoires sont généralement dues à des composants endommagés, des barrettes incompatibles ou un « timing d'accès » à la mémoire (nommé CAS (Column Address Strobe) dans la plupart des BIOS) trop agressif permettant des lectures dans des zones mémoire avant que les écritures aient eu le temps de se stabiliser pour en garantir le contenu.
Il est assez simple de valider votre configuration mémoire grâce à l'utilitaire Memtest (http://www.memtest.org). Cet utilitaire de taille très modeste, 96 Kilooctets, est écrit en assembleur ce qui lui permet de s'exécuter indépendamment du système d'exploitation. Ceci garantit un test « proche » du matériel, sans sur-couche pouvant apporter d'autres problèmes.
La version 1.65 actuelle est téléchargeable sous deux formes : une iso que l'on peut graver sur un CDRW de préférence vu la petite taille du programme ou sous la forme d'une amorce système (pre-compiled bootable binary). Il suffira de décompresser l'archive puis d'exécuter la commande suivante : ~~dd if=memtest.bin of=/dev/fda~~. Ceci écrasera le contenu de votre disquette pour y copier Memtest. Rebootez l'ordinateur avec la disquette placée dans le lecteur, Memtest devrait se lancer.
La figure 1, est une capture d'écran de Memtest en fonctionnement. Le barre de progression principale affichée en haut à droite de l'écran (2% dans cet exemple) permet de suivre la progression du test. Les erreurs apparaissent dans la partie inférieure de l'écran. Il suffit de laisser fonctionner Memtest au moins une fois jusqu'à 100%. Si aucune ligne ne s'est affichée alors vous pouvez considérer que la mémoire est fonctionnelle. Dans le cas contraire, il faudra soit procéder à son remplacement s'il s'agit d'une seule barrette mémoire, soit tester les barrettes indépendamment (l'une puis l'autre) avec ce logiciel. Dans le cas de 2 ou 3 barrettes qui ne fonctionnent pas correctement ensemble, mais qui fonctionnent correctement individuellement, il faudra vérifier qu'elles sont bien de même modèle/marque et remettre le BIOS dans une configuration initiale.
Votre mémoire est bonne ? Testons les autres composants de votre machine pour traquer la source de ce plantage.

1.3 Il ne fait pas un peu chaud là ?

Ces dix dernières années la puissance consommée et dissipée par nos ordinateurs a littéralement explosée ! Du 486 qui ne nécessitait quasiment pas de ventilateur, on est arrivé à des stations de travail/jeu avec des alimentations pouvant atteindre 1KiloWatt pour les dernières et 500watt pour des modèles plus grand public.
Le processeur est un des composants gourmands de la machine. Certains modèles peuvent consommer plus de 130W ! Autant dire que si le système est mal ventilé, la montée en charge du processeur peut provoquer un échauffement important provoquant une surchauffe de celui-ci. S'il dépasse ces caractéristiques de fonctionnement, il va alors exécuter du code incorrect provoquant un plantage du système.
Un bon test pour vérifier que votre processeur n'est pas la cause de votre instabilité : commencez par relever la température de votre machine au repos à l'aide des informations

commencez par relever la température de votre machine au repos à l'aide des informations fournies soit par l'acpi, soit par ~~lm_sensors~~ (cf. le premier article : numéro 68).

```
[erwan@R1 ~]$ cat /proc/acpi/thermal_zone\
/THRM/temperature
temperature:      42 C

[root@konilope ~]# sensors | grep Temp
M/B Temp: +39°C (low=+15°C, high=+40°C) sensor = thermistor
CPU Temp:+127°C (low=+15°C, high=+45°C) sensor = thermistor
Temp3:    +44°C (low=+15°C, high=+45°C) sensor = thermistor
```

La méthode utilisant ~~lm_sensors~~ nécessite une interprétation de la part de l'utilisateur. Dans cet exemple, il apparaît évident que le processeur ne peut avoir une température de fonctionnement aussi élevée (127°C). L'erreur est due à un mauvais calcul de la part de ~~lm_sensors~~. Il se base sur le type de capteurs thermiques utilisé par la carte mère pour calculer la température des différents composants.
Malheureusement, le montage électrique autour du capteur de température peut varier d'un modèle de carte à un autre, ce qui fausse les calculs.
On considéra donc, que pour cette machine (konilope), sa température se situe aux alentours de 42° : valeur moyenne entre les deux valeurs « cohérentes » relevées par ~~lm_sensors~~. On pourra cependant vérifier cette approximation en relevant la température du disque dur à l'aide de smartmontools.

```
[root@konilope ~]# smartctl -A /dev/hda | grep Temperature
194 Temperature_Celsius 0x0022 042 052 000 Old_age Always - 42
```

De manière générale, il est préférable de maintenir sa machine en dessous des 55° et celle de son processeur en dessous de 70°C. Au-delà, le système risque d'être instable. Si ce n'est pas le cas, effectuez les modifications nécessaires pour y parvenir : meilleure ventilation du processeur ou du boîtier par exemple.
Ensuite, téléchargez l'utilitaire cpuburn depuis un miroir (http://pages.sbcglobal.net/redelm/cpuburn_1_4_tar.gz) ou depuis votre distribution Linux favorite (oui, oui, ~~urpmi cpuburn~~ ça fonctionne [NDLR : ~~apt-get install cpuburn~~ aussi]).
Vous y trouverez plusieurs programmes nommés burnXX où XX représente votre processeur. La version P6 sera à utiliser sur les processeurs Intel Pentium 4, la version K7 pour les processeurs Athlon x86. Pour les processeurs plus récents, utilisez la version P6.
Ce logiciel réalise des boucles en langage assembleur ayant pour but de stresser votre processeur de manière importante et donc le faire chauffer. Lancez cpuburn autant de fois que vous avez de processeur sur votre machine, puis observez la température de votre machine. Si lors de la montée en température votre système se fige, alors vous avez un problème de thermie.
Si cpuburn ne fait pas planter votre ordinateur au bout de 10mn, vous pouvez considérer que votre système est stable de ce coté-là. Vous pouvez aussi utiliser ce test pour vérifier que vos ventilateurs d'ordinateur portable se déclenchent bien en cas de surchauffe.
Sur un portable récent, on obtient le résultat suivant au bout de quelques minutes de cpuburn.

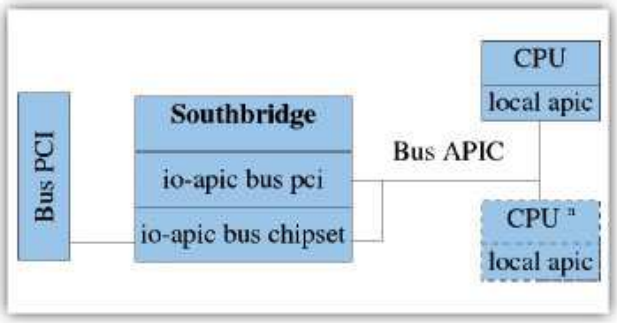
```
[root@R1 ~]# cat /proc/acpi/thermal_zone/THRM/temperature
temperature:    80 C
```

Malgré la température très élevée du processeur, le système est resté stable ce qui permet de valider qu'il est capable d'encaisser une montée en température élevée. Bien entendu, plus votre machine sera « chaude », plus elle s'usera rapidement.
Il ne faut pas oublier que les autres composants peuvent souffrir de la chaleur dégagée par le processeur par exemple : les disques durs en sont un bon exemple car les plateaux ne supportent pas la chaleur.
Ce test est bon aussi ? A priori, les composants de base de votre système sont en bon état, vous pouvez commencer à regarder du coté configuration logicielle.

1.4 APIC es-tu là ?

Le manque de stabilité peut aussi s'expliquer par la gestion des interruptions. L'architecture PIC (Programmable Interrupt Controllers) représentée par la famille de composants Intel 8259, permettait une gestion de 8 interruptions pour un unique processeur. On les associe par 2 pour obtenir les 16 niveaux d'interruptions nécessaires depuis nos vieux 80286 (milieu des années 80).
Avec l'arrivée des systèmes multiprocesseurs, l'architecture APIC (Advanced Programmable Interrupt Controllers) (Intel 82093) a été ajoutée pour permettre une gestion plus fine des

interruptions. En effet, dans un contexte multiprocesseur, il est nécessaire de pouvoir choisir quel processeur traitera quelle interruption pour ne pas rester figé dans l'ancien modèle (PIC), où le processeur de boot est assigné à un ensemble d'interruptions. De plus, l'architecture APIC apporte une gestion de 24 interruptions, une gestion de priorité des interruptions et le routage de l'interruption sur le bon processeur, contre un simple routage de 2*8 interruptions dans le modèle PIC sur un seul processeur.
Depuis une dizaine d'années, Intel propose le modèle APIC comme nouvelle architecture, y compris pour les systèmes mono-processeur. Les ordinateurs récents, y compris les portables, sont maintenant équipés uniquement de composants APIC, même si l'on peut obtenir un comportement de type PIC avec ceux-ci. L'architecture APIC est constituée d'un ou plusieurs composants situés physiquement dans le southbridge (IO-APIC), généralement un par bus, et un composant situé dans le processeur : le Local APIC (LAPIC). Le LAPIC va permettre la gestion des interruptions au niveau du processeur local. Il permet aussi de générer des interruptions par exemple pour transmettre des interruptions à des processeurs voisins dans le contexte d'un système multiprocesseur. Ces messages sont appelés « IPI » (Inter Processor Interrupts). De plus, le Local APIC (LAPIC) inclut un timer de précision cadencé à la fréquence du bus, généralement il peut être utilisé par l'ordonnanceur de votre système d'exploitation favori ou d'autres outils de haut niveau (profiler kernel : ~~oprofile, vtune~~). Les LAPIC et l'IO-APIC sont reliés par un bus spécifique pour permettre un échange rapide des informations.
Schéma logique d'implémentation :



Les IO-APIC vont donc se charger de collecter les interruptions provenant des périphériques systèmes et d'en assurer le transfert vers le ou les processeur(s) suivant une route préalablement programmée par le BIOS et/ou le système d'exploitation.
Cependant, l'implémentation des APIC dans les BIOS peut provoquer certaines instabilités sur votre système Linux. Par expérience, il n'y a pas de règle pour trouver la configuration qui peut aider votre machine à se stabiliser car cela dépend de chaque machine. Des différences peuvent même exister entre deux versions de BIOS d'un même ordinateur. Cependant, dans la série 2.6 du noyau Linux et les versions 2.4 récentes, nous disposons d'outils permettant de re-paramétrer cette configuration en activant/désactivant les IO-APIC ou les LAPIC.
Chaque composant possède un activateur ou un inhibiteur, on aura donc d'un côté ~~apic~~ et ~~noapic~~ pour la gestion des IO-APIC, ~~nolapic~~ et ~~lapic~~ de l'autre pour la gestion des LAPIC. Ces 4 options sont à utiliser sur la ligne de commande du noyau, c'est-à-dire au moment où le bootloader (lilo ou grub par exemple) va charger le noyau en mémoire. L'utilisation de ~~apic~~ et ~~lapic~~ permet d'activer ces composants qui ne l'ont pas été par le BIOS. L'activateur ~~apic~~ n'est disponible que pour les architectures de type x86_64 car celui-ci est déjà activé sur les systèmes de type x86. On désactivera les APIC dans le cas où le système affiche des messages d'erreurs sur les APIC comme :

```
APIC error on CPU0: 40(40)
```

ou

```
MP-BIOS bug: 8254 timer not connected to IO-APIC
failed.
timer doesn't work through the IO-APIC - disabling NMI Watchdog!
```

La désactivation des IO-APIC permet par exemple de corriger le comportement parfois récalcitrant de certaines cartes vidéo qui ne parviennent pas à afficher correctement en mode 2D/3D (écran noir par exemple).
La non activation des LAPIC peut avoir comme conséquence un dysfonctionnement de l'ACPI (gestion d'énergie avancée). Ce cas est traité par le noyau. Vous retrouverez dans les messages de démarrage la ligne suivante : ~~Local APIC disabled by BIOS you can enable it with~~

messages de démarrage la ligne suivante : ~~Local APIC disabled by BIOS -- you can enable it with lapic~~
Pour résumer, si votre machine, en général à architecture mono-processeur, a tendance à se bloquer lors de l'utilisation de périphériques tels que le port USB, la carte graphique ou la carte son, essayez de désactiver le LAPIC puis l'IO-APIC puis les 2. Pour exemple, les cartes mère A7N8X de chez Asus ont largement tendance à se bloquer si l'on ne spécifie pas ~~nolapic~~ lors du boot. Sur certaines machines, il faudra faire le contraire, à savoir forcer l'activation des LAPIC. Ceci dépendra de votre machine et de sa conception.

2 A vos marques, rebootez !

Certaines machines peuvent présenter quelques soucis pour redémarrer. Encore une fois, on retrouve au cœur de ce problème l'interaction entre le noyau et le matériel notamment l'ACPI. Les ordinateurs portables dont le modèle HP nc6120 ont par exemple le bon goût de rester bloqués sur le message ~~No reboot fixup found for your hardware~~. Ceci indique que le noyau ne connaît pas de contournement pour ce matériel (ce qui est le cas de la quasi-totalité des ordinateurs), mais puisque vous voyez ce message c'est que votre machine n'a pas réussi à rebooter.
Heureusement, il existe des contournements pour permettre un redémarrage plus « bas niveau ». Le noyau définit les modes suivants : « b », « c », « h », « s » et « w ».
Le mode de reboot peut être défini sur la ligne de commande du noyau suivant la forme ~~reboot=<mode>~~. Le mode « s » est réservé aux systèmes multiprocesseurs, il va réinitialiser chacun des processeurs.
Le mode « b » permet quant à lui de rebooter en faisant un appel direct au BIOS qui se chargera de faire redémarrer la machine. Ce mode fonctionne plutôt bien car dans le cas d'un BIOS posant problème sur l'ACPI, on peut espérer avoir un reboot correct. C'est le cas sur de notre portable cité précédemment.
Le mode « h » se chargera de faire redémarrer la machine en réinitialisant le processeur. C'est l'équivalent du mode « s » pour les environnements monoprocesseurs.
Pour finir, les modes « c » et « w » correspondent aux modes « cold » et « warm » : ces modes sont plus connus sous le nom de « reboot à froid » et « reboot à chaud ». Le premier effectuera les tests de mémoire tandis que le second exécutera un reboot plus court.
De manière générale, on utilisera plutôt le mode « b » ou « h » pour les systèmes monoprocesseurs récalcitrant et le mode « s » pour les systèmes multiprocesseurs.

3 Toujours plus de performance

Votre configuration est bien stable depuis un certain temps. Toutefois, vous trouvez qu'elle est un peu lente, elle ne vous donne pas satisfaction par rapport à ce qu'elle devrait fournir. Cette partie de l'article va explorer comment vérifier et corriger les problèmes de performance des différents composants.

Note:
Certaines configurations monoprocesseurs nécessitent de désactiver LAPIC (~~nolapic~~)

3.1 Le disque dur

Il arrive parfois que le disque dur semble un peu lent et pénalise le système dans son ensemble. ~~hdparm~~ est là pour vous aider à y voir plus clair. ~~hdparm~~ est à l'origine conçu pour les systèmes de disques IDE de type PATA. Le nouveau système de disque SATA, ne permet pas toujours en fonction de l'implémentation, l'appel à toutes ces astuces. Cependant les nouvelles séries de kernel à partir du 2.6.15 devrait permettre une gestion plus unifiée des systèmes de disques PATA et SATA.
Le test ~~hdparm -t /dev/<mon_device>~~ permet de tester la vitesse de lecture depuis le disque dur jusqu'en mémoire. Une valeur de 25Mo/sec est un minimum pour des machines récentes. Plusieurs paramètres peuvent influencer les performances d'un disque IDE.

```
[root@R1 ~]# hdparm -cd /dev/hda
/dev/hda:
 IO_support   =  0 (default 16-bit)
 using_dma    =  1 (on)
[root@R1 ~]#
```

La commande ~~hdparm -cd~~ permet de vérifier que les entrées/sorties sont réalisées en 16-bit

La commande hdparm -u permet de vérifier que les entrées/sorties sont réalisées en 16 bit (le mode 32-bit n'offre pas de performance supplémentaire) alors que le DMA (Direct Memory Access) est activé. Ceci permet la minimisation de l'utilisation du processeur pendant les accès au disque dur.
La commande hdparm -c0d1 /dev/<mon_device> permet de configurer le périphérique dans ce mode. Le noyau de votre distribution Linux devrait le réaliser de manière automatique.
L'option -I permet d'obtenir plus d'informations sur la configuration de votre disque dur.

```
[root@R1 ~]# hdparm -I /dev/hda
/dev/hda:
ATA device, with non-removable media
        Model Number:       FUJITSU MHT2040AT
        Serial Number:      NN77T3C13KB9
        Firmware Revision:  0022
Standards:
        Used: ATA/ATAPI-6 T13 1410D revision 3a
        Supported: 6 5 4 3
Configuration:
        Logical         max     current
        cylinders       16383   16383
        heads           16      16
        sectors/track   63      63
        --
        CHS current addressable sectors:   16514064
        LBA    user addressable sectors:   78140160
        device size with M = 1024*1024:       38154 MBytes
        device size with M = 1000*1000:       40007 MBytes (40 GB)
Capabilities:
        LBA, IORDY(cannot be disabled)
        bytes avail on r/w long: 4      Queue depth: 1
        Standby timer values: spec'd by Standard, no device specific minimum
        R/W multiple sector transfer: Max = 16  Current = 16
        Advanced power management level: 128 (0x80)
        Recommended acoustic management value: 254, current value: 254
        DMA: mdma0 mdma1 mdma2 udma0 udma1 udma2 udma3 udma4 *udma5
             Cycle time: min=120ns recommended=120ns
        PIO: pio0 pio1 pio2 pio3 pio4
             Cycle time: no flow control=240ns  IORDY flow control=120ns
Commands/features:
        Enabled Supported:
           *    READ BUFFER cmd
           *    WRITE BUFFER cmd
           *    Host Protected Area feature set
           *    Look-ahead
           *    Write cache
           *    Power Management feature set
                Security Mode feature set
           *    SMART feature set
           *    Mandatory FLUSH CACHE command
           *    Device Configuration Overlay feature set
           *    Automatic Acoustic Management feature set
           *    SET MAX security extension
                Power-Up In Standby feature set
           *    Advanced Power Management feature set
           *    DOWNLOAD MICROCODE cmd
           *    SMART self-test
           *    SMART error logging
Security:
                supported
        not     enabled
        not     locked
                frozen
        not     expired: security count
        not     supported: enhanced erase
        40min for SECURITY ERASE UNIT.
HW reset results:
        CBLID- above Vih
        Device num = 0 determined by the jumper
Checksum: correct
[root@R1 ~]#
```

On y retrouve le modèle, le numéro de série et la version du firmware qui peuvent être utiles en cas de contact avec le Service après-vente ou de mise à jour pour une série de firmwares défaillante. La ligne commençant par DMA dans la partie « Capabilities » permet d'afficher la vitesse de transfert négociée entre le disque dur et le driver IDE de Linux. L'étoile précède la valeur actuelle. Dans cet exemple, le mode udma5 est sélectionné : ceci est conforme aux machines disponibles sur le marché depuis environ 3 ans. Pour rappel, voici le tableau 1 de conversion entre le mode DMA et la vitesse maximale associée.
Une configuration actuelle ne devrait pas être en dessous de UDMA5. Si ce n'était pas le cas, on peut essayer de renégocier la vitesse de transfert à l'aide de la commande :

```
hdparm -X udma5
```

Celle-ci permet de forcer le mode UltraDMA5. Il faut utiliser cette commande avec précaution

Celle-ci permet de forcer le mode UltraDMA5. Il faut utiliser cette commande avec précaution car elle pourrait générer des pertes ou de la corruption de données en cas de transfert pendant ce changement de mode. Vous pouvez aussi impacter les performances de votre disque dur en modifiant sa vitesse de rotation. Cette fonctionnalité connue sous le nom de Automatic Acoustic Management (AAM) est utilisable avec l'option -M de hdparm. La valeur 128 représentera la vitesse la plus faible avec un disque silencieux, la valeur 254 la vitesse la plus élevée, mais avec comme contrepartie un disque plus bruyant. On pourra noter des différences de quelques Mo/sec selon les configurations. Il est aussi à noter que l'utilisation de logiciels permettant une meilleure gestion d'énergie (cpufreq, laptop-mode) peut dégrader les performances d'un disque dur de manière notable.

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

Tableau 1

3.2 Configuration bas niveau du réseau

Le réseau local filaire peut parfois être un peu lent. Voici quelques tests pour vérifier si la lenteur provient du logiciel, du noyau ou du matériel. Tout d'abord, explorons la trousse à outils indispensable pour analyser votre carte réseau : ethtool. Digne successeur de mii-tool, il permet de s'informer sur la configuration de votre carte réseau. Il suffit de lancer ethtool <votre_interface> pour obtenir la configuration matérielle de votre carte. On y retrouve les modes supportés par votre carte (Supported Link Modes ainsi que la vitesse actuelle (Speed), obtenue par auto-négociation dans cet exemple.

```
[root@konilope ~]# ethtool  eth1
Settings for eth1:
        Supported ports: [ TP MII ]
        Supported link modes:   10baseT/Half 10baseT/Full
                                100baseT/Half 100baseT/Full
        Supports auto-negotiation: Yes
        Advertised link modes:  10baseT/Half 10baseT/Full
                                100baseT/Half 100baseT/Full
        Advertised auto-negotiation: Yes
        Speed: 100Mb/s
        Duplex: Full
        Port: MII
        PHYAD: 1
        Transceiver: internal
        Auto-negotiation: on
        Current message level: 0x000020c1 (8385)
        Link detected: yes
[root@konilope ~]#
```

On peut également savoir si le câble réseau est connecté à la carte grâce à la ligne Link Detected. Le type de « duplex » utilisé permet également de savoir si la négociation avec l'élément actif au bout de votre câble (commutateur, concentrateur ou une autre machine) a correctement fonctionné. Le duplex définit le mode de transmission : le mode half-duplex indique que l'on émet puis reçoit des données, le mode full-duplex indique que l'on peut émettre et recevoir en même temps des données. Le mode full-duplex est donc plus que recommandé pour vos équipements réseau.

3.3 Mesurer les {contre}performances de votre réseau

Une fois vérifiée la bonne configuration de votre matériel, nous pouvons passer à la vérification des performances entre deux machines.

L'utilitaire netpipe (http://www.scl.ameslab.gov/netpipe/) permet de réaliser des tests de performance, mais aussi d'intégrité entre deux machines. Il permet de tester les performances du réseau au travers la couche IP et non pas depuis un mode plus bas niveau comme pourrait l'être un test utilisant directement le driver Ethernet. Ce test permet donc de valider que la configuration matérielle, mais aussi logicielle, est correcte.
Sur la première machine, exécutez NPtcp : netpipe se positionne en mode « serveur ».
Sur la seconde machine, exécutez NPtcp -h <machine1> : netpipe se connecte au serveur

netpipe de la machine nommée « machine1 » pour effectuer le test de performance.
Le test par défaut teste les communications allant de 1 octet jusqu'à 8 Mo. Il peut prendre plusieurs minutes si le réseau est « lent ». Le test va montrer pour chaque taille de paquet la performance obtenue. Il est normal d'obtenir des performances faibles pour des paquets d'une taille inférieure à 8 Ko. Transporter des petits paquets nécessite de réaliser beaucoup d'opérations pour le processeur pour une quantité d'information minime. Les performances seront meilleures à partir d'environ 16 Ko et maximales à partir de 128 Ko. Sur un système utilisant du gigabit ethernet, on constatera une performance de 39 Mbps à 256 octets, 287 Mbps à 4 Ko, 614 Mbps à 16 Ko, 845 Mbps à 128 Ko, 895 Mbps à 8 Mo.
Vous pouvez ainsi vérifier et comparer facilement les performances de votre configuration réseau. Netpipe permet également de tester la vitesse du réseau pour une taille de paquets définie entre la taille indiquée par l'option -l et la taille indiquée par l'option -u. Ainsi, NPtcp -l 16384 -u 16384 permet de tester la bande passante réseau pour une taille de paquet égale à 16 Ko.

L'option -n <nb-cycle> permet de définir le nombre de boucles à réaliser pour le test (50 par défaut). Enfin, pour réaliser des tests « full-duplex », on utilisera l'option -2. Ce test permet de s'assurer du bon fonctionnement du mode de communication permettant l'écriture et la lecture simultanées. Attention, même si ce test donne de bons résultats, cela ne veut pas dire que la configuration est exempte d'erreur de configuration. Par exemple, les performances réseau peuvent être très bonnes en point à point, mais mauvaises lorsque l'on aura une charge réseau avec plusieurs clients. Dans cette situation, il faut vérifier que le contrôle de flux est bien activé sur toutes les machines. L'option ethtool -a permet de vérifier que les champs rx et tx sont placés sur on. Si ce n'était pas le cas activez-les avec la commande ethtool -A eth0 rx on et ethtool -A eth0 tx on.
Les cartes réseau, des serveurs essentiellement, permettent maintenant de réaliser le « tcp segment offload ». Cette technique permet de décharger une partie du traitement TCP directement dans l'électronique de la carte réseau. Ceci a pour effet de décharger le processeur central d'un traitement fastidieux. L'option -k d'ethtool permet de consulter l'état de cette fonctionnalité (off/on). On pourra l'activer sur des serveurs générant un gros trafic réseau par exemple : ethtool -K eth0 tso on. Vous voilà maintenant équipé de nouveaux outils pour traquer la perte de performances de votre installation réseau. Ceci couplé aux habituels tcpdump/ethereal, voilà une belle panoplie du Zorro du réseau.

3.4 Quand la latence s'en mêle

La latence représente le temps nécessaire à un système pour prendre en compte un événement.
Il arrive que certains périphériques de votre machine nécessitant des débits importants ou une latence assez faible sur le bus PCI soient « poussifs ». On retrouve dans cette catégorie les cartes vidéo accélératrices 3D, les cartes son et les cartes d'acquisition firewire ou DVB (récepteur de télévision numérique terrestre : TNT).
Un périphérique branché sur le bus PCI doit attendre que le bus soit libre pour émettre les données vers la mémoire centrale. Le bus PCI qui est le cœur de communication des périphériques est donc soumis en permanence à un arbitrage qui permet de décider qui doit émettre des données et qui doit attendre. Cet arbitrage est en partie réalisé grâce au latency_timer.
Chaque périphérique possède une valeur de latency_timer comprise en 0 et 248 cycles d'horloge du bus. Plus la valeur sera importante plus le périphérique disposera de temps pour transmettre des informations. Une latence de 248 cycles permet d'obtenir une priorité forte sur le bus, ce qui permet un débit important. A contrario, une latence de 0 indiquera au périphérique qu'il doit cesser ses transmissions si un autre composant a besoin du bus : ceci se traduira par une perte de bande passante importante. La configuration réalisée par défaut sur votre machine doit être correcte, mais il peut arriver qu'une carte 3D ou DVB soit configurée avec une latence trop faible, ce qui aurait pour conséquence d'offrir une image saccadée et peu fluide. Une carte son offrirait également le même type de symptômes avec un son haché ou excessivement métallique.
Heureusement, une série d'outils est là pour vous aider à diagnostiquer le problème, mais aussi à le résoudre. Le projet pciutils (http://atrey.karlin.mff.cuni.cz/~mj/linux.shtml#pciutils) contient deux outils indispensables : lspci et setpci. Bien entendu, votre distribution favorite devrait posséder ce logiciel, il vous suffira de l'installer avec votre gestionnaire de paquet favori (urpmi, apt, smart). La commande lspci vous permet de lister les périphériques PCI disponibles sur votre ordinateur (cf. article paru dans le numéro 68). Cet exemple se basera sur une carte vidéo ayant pour adresse sur le bus PCI 01:00.0.

```
[root@lagirafe] lspci -v -s 01:00.0
01:00.0 VGA compatible controller: nVidia Corporation NV34M [GeForce FX Go5200 32M/64M] (rev a1) (prog-if 00 [V
        Subsystem: Samsung Electronics Co Ltd: Unknown device c00f
        Flags: bus master, 66Mhz, medium devsel, latency 16, 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
[root@lagirafe]
```

On retiendra dans cette ligne, la valeur latency 16. Ceci risque de poser problème lorsque celle-ci aura besoin d'un fort débit pendant un jeu par exemple. Augmenter la latence va donc se révéler nécessaire pour ce périphérique. La commande setpci permet de modifier les paramètres PCI d'un périphérique ou d'un bus complet avec tous ses périphériques.
La commande setpci -v -d 01:* latency_timer=20 permet de positionner tous les périphériques du bus 01 à une valeur de 32 cycles. Il est à noter que la commande setpci utilise des valeurs en hexadécimal, mais lspci affichera les valeurs en décimal. Pour en revenir à nos moutons, la correction de la latence de cette carte graphique se fera avec la commande setpci -v -d 01:00.0 latency_timer=f8. Ceci positionne donc la carte sur la plus grande valeur de latence possible, ce qui lui garantira la plus grande bande passante possible. Il est possible d'utiliser la valeur ff comme paramètre pour latency_timer, mais celui-ci sera converti à la volée par la commande setpci en 0xf8.
Attention, certains composants de votre machine doivent rester en latency_timer=0 pour garantir un bon fonctionnement de votre système : il ne faudra donc pas modifier les latences des contrôleurs mémoire et des ponts PCI connectés généralement sur le bus 0:*.

4 DSDT : le cauchemar continue !

Cette partie de l'article a pour but d'expliquer le problème que peut représenter la table DSDT pour un certain nombre de machines et surtout les ordinateurs portables. La norme ACPI 3.0 de septembre 2004 disponible gratuitement à l'adresse http://www.acpi.info/DOWNLOADS/ACPIspec30.pdf définit la DSDT comme une table remplie par un OEM permettant au système d'exploitation de découvrir les ressources pour la gestion de l'énergie avancée, de la gestion thermique ainsi que la gestion du plug'nplay des ressources définies par l'ACPI. Voici une description certes un peu théorique mais nécessaire pour bien comprendre la problématique et ses conséquences. Le constructeur de l'ordinateur doit donc définir dans le BIOS une table contenant une liste de ressources et leur configuration, et ce, pour que le système d'exploitation puisse s'en servir. Noble cause n'est-il pas ? Tout au long des 618 pages de ce document sont spécifiés les éléments du langage à utiliser ainsi que sa syntaxe. L'ACPI Source Language permettra donc aux développeurs d'écrire des DSDT puis de les compiler dans un format nommé ACPI Machine Language en utilisant un compilateur adéquat. Le système d'exploitation utilisera la table compilée (AML) pour accéder aux ressources ACPI.
Sur la plupart des ordinateurs portables, la table DSDT ne respecte pas la norme ACPI dans sa syntaxe. Ceci est très gênant car cela bloque le code de l'interpréteur ACPI du noyau Linux qui essaie de la lire. Le résultat est sans appel, pas d'ACPI, pas de gestion d'énergie avancée, plus de ventilation automatique sur certains modèles d'ordinateurs, plus de touche de gestion de contraste/luminosité sur d'autres, etc. Ceci est très gênant sur un ordinateur portable.
Alors que faire ? Dans un premier temps, prévenir le constructeur que son BIOS est incorrect et ne respecte pas la norme ACPI. La réponse se fait toujours attendre pour tous ceux qui « osent » poser la question au support technique : à savoir, les constructeurs ignorent tout simplement les mails de ce type.
La seconde étape, puisque vous êtes devant le fait accompli d'une machine non conforme, il va falloir corriger cette table. Il suffit de récupérer la table actuelle via la commande cat /proc/acpi/dsdt > dsdt.aml si vous avez la chance d'avoir un répertoire /proc/acpi malgré une table DSDT endommagée, sinon utilisez la commande acpidump -b -t dsdt -o dsdt.aml. La commande acpidump est fournie dans le projet pmtools (ftp://ftp.kernel.org/pub/linux/kernel/people/lenb/acpi/utils/pmtools-20050926.tar.bz2). Vous avez donc extrait la table DSDT de votre système. Il ne reste plus qu'à essayer de la recompiler. Pour cela, le compilateur IASL de la société Intel (http://developer.intel.com/technology/iapc/acpi/downloads.htm) offre sous Linux et Windows une plate-forme de compilation pour le langage ASL. Suivez les instructions du site pour télécharger et compiler la version unix d'IASL. Ensuite, décompilez la table avec la commande iasl -d dsdt.aml. Le fichier dsdt.dsl vient d'être créé par le décompilateur. Il suffit de le recompiler avec la commande iasl -tc dsdt.dsl pour constater les dégâts ! La DSDT que vous avez dans votre BIOS est généralement compilée par le compilateur de Microsoft qui semble

avez dans votre BIOS est généralement compilée par le compilateur de Microsoft qui semble autoriser une syntaxe plus qu'approximative. Si vous avez de la chance, votre DSDT s'est recompilée sans erreur. Vous obtenez donc un fichier dsdt.hex et un fichier dsdt.aml. Il ne reste plus qu'à injecter cette DSDT dans le noyau Linux et le tour sera joué.
Si vous avez moins de chance, votre DSDT est syntaxiquement incorrecte et génère des erreurs. Les expliquer plus en détail nécessiterait un article complet. Il ne sera cité que les plus « habituelles » (sic).

```
dsdt.dsl  2626:     Field (ECR, DWordAcc, Lock, Preserve)
Error    1048 -      ^ Host Operation Region requires ByteAcc access
```

Cette erreur présente à la ligne 2626 du fichier dsdt.asl indique une erreur de type dans la déclaration du champ ECR. Il suffit dans ce cas de changer le mot DWordAcc par ByteAcc. Celle-ci reste assez simple à corriger. On se demande donc pourquoi les constructeurs ne prennent pas le temps de vérifier et de corriger leur BIOS.

```
dsdt.dsl  2672:    Method (_GLK, 1, NotSerialized)
Warning  2024 -    ^ Reserved method has too many arguments ( _GLK requires 0)
```

La méthode _GLK n'a besoin que de 0 paramètre tandis que ce code en utilise 2 ! La correction est assez simple : il suffit de remplacer la ligne 2672 par Method(_GLK).
Pour plus de détails sur la correction des erreurs de compilation de votre table DSDT, je vous recommande les tutoriaux disponibles depuis la page http://acpi.sourceforge.net/dsdt/index.php du projet ACPI. Votre DSDT est compilée avec succès, il ne vous reste plus qu'à l'ajouter pour que votre noyau puisse l'utiliser à la place de celle fournie par le BIOS (modifiable uniquement par le constructeur, via une mise à jour). Il existe trois possibilités pour inclure une table DSDT : la modification du noyau Linux (noyau < 2.6.9), l'injection de la DSDT dans la configuration du noyau (noyau ≥ 2.6.9), l'intégration de la table DSDT dans l'initrd qui nécessite un patch kernel disponible à l'adresse http://gaugusch.at/kernel.shtml.
Pour les noyaux 2.6.9 et supérieurs, copiez votre fichier dsdt.hex sous le nom /usr/src/linux/include/acpi/dsdt_table.h, puis reconfigurez votre noyau. Insérer dsdt_table.h dans l'option Include you custom DSDT disponible dans le menu Power Management/ACPI/. Recompilez votre noyau et redémarrez avec votre nouveau noyau.
Pour les noyaux inférieurs à 2.6.9, il est possible de modifier le code source de votre noyau pour intégrer votre DSDT, mais il est préférable d'utiliser le patch permettant l'insertion de la table DSDT dans l'initrd. Recompilez votre noyau avec ce patch, puis suivez la procédure décrite ci-dessous.

Certaines distributions Linux comme Mandriva Linux possèdent le patch qui permet de n'avoir qu'à reconstruire l'initrd avec la commande mkinitrd --dsdt=/boot/dsdt.aml /boot/initrd-`uname -r`.img `uname -r` puis de relancer LILO avant de redémarrer votre machine. Lors du démarrage de celle-ci, vous devriez voir le message suivant :

```
ACPI: Looking for DSDT in initrd... found (at offset 0x4f8d9)
ACPI-0294: *** Info: Table [DSDT] replaced by host OS
```

Vous voilà maintenant avec une table DSDT correcte et un ACPI fonctionnel. La question qui reste ouverte est pourquoi les constructeurs n'utilisent pas un compilateur digne de ce nom pour permettre à leurs clients de pouvoir utiliser d'autres systèmes d'exploitation que Windows de Microsoft.
Pour la petite histoire, si vous cherchez dans votre DSDT vous risquez de tomber ce genre de lignes :

```
If (LEqual (SCMP (\_OS, «Microsoft Windows NT»), Zero))
If (LEqual (SCMP (\_OS, «Microsoft Windows»), Zero))
If (LEqual (SCMP (\_OS, «Microsoft WindowsME: Millennium Edition»), Zero))
If (LEqual (SizeOf (\_OS), 0x27))
```

Lorsqu'un système d'exploitation souhaite utiliser les ressources ACPI du BIOS, il utilise son nom de système d'exploitation (acpi_os_name).

On retrouve donc dans les BIOS des parties qui ne seront exécutées que pour certains systèmes d'exploitation de Microsoft. On retrouve même certains tests où c'est la longueur du nom qui compte...

Ceci prouve bien que les constructeurs d'ordinateurs savent corriger des tables DSDT pour les systèmes d'exploitation, mais pas forcément pour le nôtre malheureusement. « Depuis le noyau 2.6.9, le noyau Linux utilise 'Windows NT' comme acpi_os_name pour assurer une

meilleure compatibilité matérielle », dixit Len Bron d'Intel dans le changelog de cette version.

5 Comment trouver le pilote qui va avec votre matériel ?

Cette question, on se la pose souvent mais uniquement lorsque l'on essaye de brancher un nouveau périphérique sur sa machine.
Les distributions récentes permettent de détecter automatiquement votre matériel et de lui associer un pilote avec sa configuration.
Ce mécanisme fonctionne plutôt bien, mais cela n'évite pas les dysfonctionnements, voire l'absence de fonctionnement.
Pour vérifier si votre noyau Linux est capable de supporter un nouveau périphérique PCI par exemple, il suffit d'en récupérer les identifiants PCI (constructeur et produit), puis de les rechercher dans le fichier ~~/lib/modules/`uname -r`/modules.pcimap~~ qui est auto-généré lors de la commande ~~depmod~~.
L'utilitaire ~~lspcidrake -v~~ vous permet de trouver les identifiants PCI de votre carte, la commande ~~lspci~~ vous permet de trouver le nom de votre périphérique. Puis faites un ~~lspci -n~~ pour en trouver la correspondance numérique. On utilisera la position du périphérique sur le bus pour retrouver la correspondance numérique ~~02:05.0~~ dans cet exemple.

```
[root@pumphome] lspci
[....]
02:05.0 Ethernet controller: Broadcom Corporation BCM4401 100Base-T (rev 01)
[root@pumphome] lspci -n
[...]
02:05.0 Class 0200: 14e4:4401 (rev 01)
[root@pumphome]
```

Ainsi cette carte réseau possède les identifiants ~~14e4~~ (constructeur) et ~~4401~~ (produit). Il suffit de rechercher cette valeur dans le ~~modules.pcimap~~, ce qui nous donnera la correspondance en module (pilote).

```
[root@pumphome]  grep «0x000014e4 0x00004401» \
/lib/modules/2.6.12-12mdk-i686-up-4GB/modules.pcimap
b44          0x000014e4 0x00004401 0xffffffff 0xffffffff 0x00000000 0x00000000 0x0
bcm4400      0x000014e4 0x00004401 0xffffffff 0xffffffff 0x00000000 0x00000000 0x0
```

On voit ici que cette carte peut utiliser le driver ~~b44~~ ou le driver ~~bcm4400~~. On ne trouve pas toujours les identifiants PCI de produit sous leur forme directe, on peut également trouver ~~0Xffffffff~~ qui indique que toutes les cartes du constructeur sont gérées par un module.
Il ne vous restera plus qu'à effectuer la configuration associée à ce périphérique pour qu'il fonctionne correctement.

Conclusion

La fin de cet article conclut une toute petite série de deux articles ayant pour but de vulgariser l'architecture de nos machines, de découvrir les outils disponibles pour mieux comprendre leur configuration et, au besoin, régler les problèmes de configuration matérielle qui ont souvent le don de nous pourrir la vie.