

Installation Debian Linux Alpha sur Serveur Compaq DS20

Y. Morère

Avril 2006

Résumé

Le but de cet article est de présenter l'installation d'une distribution Debian Linux pour architecture Alpha. Le serveur utilisé est un Compaq DS20. Il est équipé d'une carte RAID Mylex DAC960PDU 3 canaux 8MB. Le principale problème de l'installation sera de réaliser la configuration du RAID sur les disques présents dans la machine.

Table des matières

1	Introduction	1
2	Configuration du RAID	2
2.1	Allumage de la machine et Console SRM	3
2.2	Accès à la configuration de la carte RAID	4
3	Installation de Debian Linux	5
3.1	Changement de noyau	7
3.2	Problème périphériques souris	10
4	Conclusion	10

1 Introduction

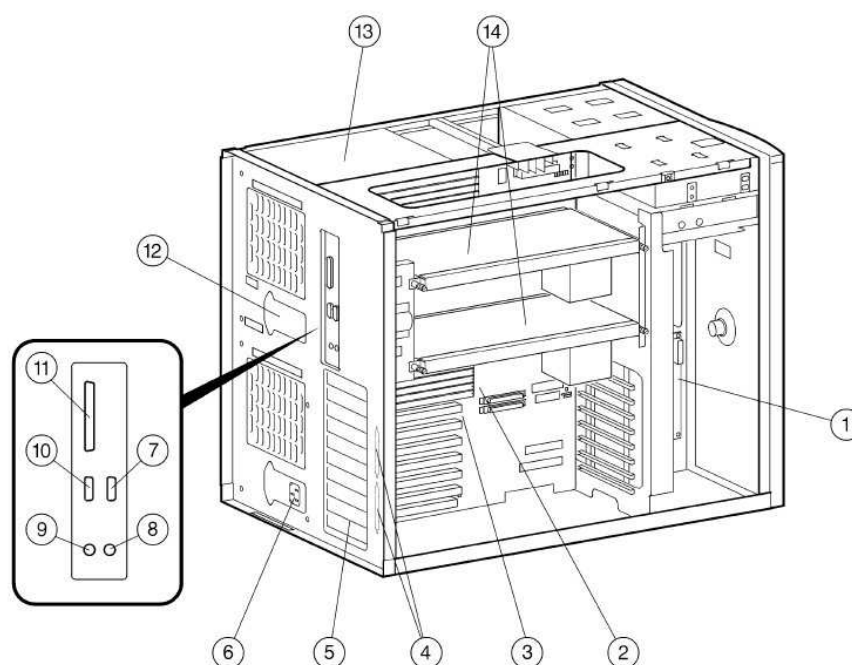
Voici une petite présentation du matériel mis à ma disposition :

- ▷ Compaq Alphaserver DS20 :
 - ▷ 1 CPU ALPHA 500 MHz
 - ▷ 1 GB mémoire SDRAM ECC
 - ▷ lecteur DAT 4mm 8GB SCSI
 - ▷ lecteur cdrom SCSI
 - ▷ carte raid 3 canaux cache 8MB split Mylex DAC960PDU
 - ▷ 1 disque 18GB 7200tr/mn UWSCSI
 - ▷ 2 disques 36.4 GB 10000tr/mn UWSCSI
 - ▷ 4 disques 36.4GB 7200tr/mn UWSCSI

Site officiel <http://h18002.www1.hp.com/alphaserver/archive/ds20/>



FIG. 1 – Photo du Serveur Compaq DS20



- | | |
|---------------------------------|----------------------------------|
| 1. Server feature card | 8. Mouse port |
| 2. Memory DIMMs | 9. Keyboard port |
| 3. PCI slots | 10. Serial port (COM2) |
| 4. SCSI breakouts | 11. Parallel port |
| 5. PCI/ISA bulkhead port | 12. Optional power inlet |
| 6. Power inlet and power supply | 13. Fan or optional power supply |
| 7. Serial port (COM1) | 14. CPU modules |

FIG. 2 – Configuration du Serveur Compaq DS20

2 Configuration du RAID

C'est cette partie qui a pris le plus de temps. N'étant pas trop familier des systèmes RAID matériels (ni logiciels d'ailleurs), et du matériel Compaq Alpha (peu nombreux sont ceux qui

peuvent s'offrir ce genre de matériel), j'ai du faire pas mal de recherche sur Internet pour trouver de la documentation concernant ces matériels.

2.1 Allumage de la machine et Console SRM

Dans un premier temps il faut se familiariser avec la console SRM. Il s'agit du programme de boot utilisé par les systèmes Alpha. On peut le comparer au BIOS des PCs, mais comme il s'agit d'une console, l'outil est très puissant. Par ailleurs Tru64 Unix et OpenVMS ont besoin de cette console pour booter. Cette console est reconnaissable par son écran bleu et son prompt composé de 3 > >>>. Pour plus d'informations, je vous conseille la lecture du SRM Firmware Howto dans sa version 0.8 de 2000 <http://www.faqs.org/docs/Linux-HOWTO/SRM-HOWTO.html#AEN11> ou encore la version 0.8.1 de 2004 <http://www.alphalinux.org/faq/SRM-HOWTO/>.

Cette console permet de configurer les périphériques de la machine ainsi que les variables stockées dans la NVRAM. Elle contient des outils de diagnostic, deverminage et configuration. On pourra par exemple taper la commande suivante pour obtenir la liste complète des utilitaires :

```
>>> help | more
```

Il est possible de voir la liste complète des variables d'environnement et leurs valeurs par la commande :

```
>>> show
```

ou mieux

```
>>> show | more
```

On remarque donc que cette console SRM ressemble fortement à une console UNIX. on peut donc taper des commandes du style :

```
show boot*
```

afin de voir toutes le variables qui commencent par `boot`.

Les variables d'environnement sont classées en 3 catégories : *read-only*, *warm non-volatile*, ou *cold non-volatile*. Les variables les plus importantes pour pouvoir booter un linux sont `bootdef_dev`, `boot_file`, `boot_flags`, et `auto_action`, qui sont toutes *cold non-volatile*. Ce qui explique qu'il faille un redémarrage à froid pour la prise en compte des modifications.

Pour configurer les variables d'environnements, on utilise la commande `set` de la manière suivante :

```
>>> set bootdef_def dka0
```

Enfin, deux raccourcis de touches importants sont `CTRL-C` qui permet d'arrêter une commande en cours (comme un boot automatique), et `CTRL-P` qui, utilisée dans le prompt `aboot` permet de revenir dans la console SRM.

2.2 Accès à la configuration de la carte RAID

La carte RAID n'est pas directement configurable à partir de la console SRM. En effet il est nécessaire de lancer le programme de configuration de cette carte. Après quelques recherches, je suis tombé sur l'adresse suivante <http://lists.debian.org/debian-alpha/2003/10/msg00087.html> qui fournit de nombreux liens pour notre carte RAID Mylex DAC960PDU. On trouvera aussi pas mal de chose chez LSI, le racheteur de Mylex <http://www.lsilogic.com/>. plus particulièrement ici :

- ▷ <ftp://ftp.compaq.com/pub/products/alphaserver/storage/controllers/kzpac/>
- ▷ <ftp://ente.limmat.ch/pub/alpha/dac960/>
- ▷ <ftp://ftp.compaq.com/pub/softpaq/alphant/>

Les programmes qui nous intéressent se trouvent dans l'archive <ftp://ftp.compaq.com/pub/softpaq/alphant/m-arcapp.zip> et sont ARCCF.EXE et ARCCF.EXE.

Je n'ai pas trouvé de solution pour lancer la configuration de la carte RAID par l'intermédiaire de la console SRM. En effet dans le manuel du serveur (oui, je suis chanceux, j'ai eu aussi les manuels mais on les trouve aussi ici <ftp://ftp.compaq.com/pub/products/alphaserver/storage/controllers/kzpac/>), toutes les configurations sont faites par l'intermédiaire de la console ARC/AlphaBIOS/ARCSBIOS, basée sur les spécifications "Advanced RISC Computing" (ARC) qui fournissent un environnement opérationnel pour Windows NT. Ceci est détaillé dans les pages d'installation Linux Debian <http://www.debian.org/releases/stable/alpha/ch05s01.html.fr>.

D'après la documentation du serveur, j'ai changé la console de démarrage en fixant la variable d'environnement `os_type` pour obtenir la console AlphaBIOS. Par défaut cette variable était positionnée sur `OpenVMS`, je l'ai fixée à `NT`.

```
>>>set os_type NT
```

Un redémarrage à froid plus tard, j'obtiens une console graphique de configuration (Après appui sur la touche F2) qui va me permettre de lancer un programme de maintenance situé sur une disquette par l'intermédiaire du Menu Utilities -> Run maintenance program.

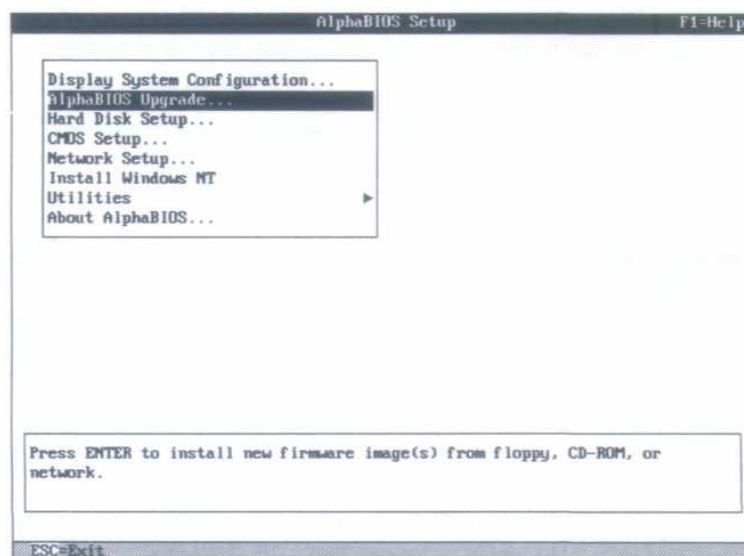


FIG. 3 – Console AlphaBIOS pour la configuration de la carte RAID

Il faut donc copier les fichiers ARCCF.EXE et ARCFL.EXE sur une disquette formatée DOS.

Ensuite dans le menu choisir "exécuter un programme de maintenance" et taper `arccf`, le lecteur de disquette étant déjà sélectionné par la lettre `A`.

Si tout se passe bien vous devriez obtenir la fenêtre du programme de configuration de la carte RAID comme décrit dans la documentation ftp://ftp.compaq.com/pub/products/alphaserver/storage/controllers/kzpac/rcu_ug.pdf

Pour la configuration de votre architecture RAID, je vous renvoie à ce manuel qui est très bien fait. Les principaux points sont les suivants :

- ▷ Configuration des groupes de disques
- ▷ Création des disques logiques RAID
- ▷ Ajout éventuel d'un disque "hot spare"
- ▷ Initialisation des disques logiques RAID
- ▷ Sauvegarde de la configuration de la carte sur disquette

Dans un premier temps je pensais faire un disque logique en RAID 5 avec 5 disques de 34Go + 1 en "hot spare". Malheureusement, une limitation de la carte DAC960DPU, ne permet pas de créer des disques logiques de plus de 32Go, et ceci même après un "flashage" du firmware de la carte RAID. Ce "flashage" est réalisé par la même procédure que pour la configuration en utilisant le fichier `ARCFL.EXE` de la disquette. Il a permis de passer d'un firmware 2.70 à un firmware 2.73.

Finalement, j'ai opté pour 3 disques logiques de 32Go en RAID 1. L'initialisation des 7 disques durs a duré 4 à 5 heures.

A la fin de l'initialisation, il ne faut pas oublier de rebasculer sur la console SRM. Pour cela il faut choisir dans le menu de configuration de la console AlphaBIOS, le type de système d'exploitation et choisir Unix ou OpenVMS. Dans le menu `CMOS Setup -> Advanced CMOS Setup`, sélectionner `Unix Console (SRM)` ou `OpenVMS Console (SRM)`.

Au prochain redémarrage à froid, on retrouvera notre console SRM, qui nous servira pour l'installation de Linux.

3 Installation de Debian Linux

La première chose à faire est de télécharger une image ISO de la distribution Linux à installer. J'ai choisi Debian Linux, car c'est une distribution que je connais assez bien. J'ai hésité avec Gentoo, mais comme je voulais que le système soit opérationnel très rapidement, j'ai évité une longue phase de recompilation.

De plus l'installateur de la distribution Debian, permet de réaliser de manière automatique le partitionnement des disques. Très pratique, ceci évite le partitionnement un peu compliqué en disque BSD comme expliqué dans la documentation Gentoo <http://www.gentoo.org/doc/fr/handbook/handbook-alpha.xml?part=1&chap=4>.

L'image de la dernière distribution stable pour architecture alpha peut être téléchargée ici : http://cdimage.debian.org/debian-cd/3.1_r1/alpha/iso-cd/debian-31r1a-alpha-netinst.iso

On insère alors le CD fraîchement gravé, dans le lecteur CDROM et on reboot la machine. Si comme moi vous avez de la chance, la machine boote directement sur le cdrom et l'installation démarre.

Dans le cas contraire, il faut préciser dans la console SRM sur quel périphérique démarrer.

La manière de nommer les périphériques est spécifique dans la console SRM.

Les deux premières lettres correspondent à une classe de matériel :

- ▷ DR : RAID set device

- ▷ DV : Floppy Drive
- ▷ EW : Ethernet port (TULIP, DEC 21040)
- ▷ EI : Ethernet port (Intel 82557 or 82559)
- ▷ PK : SCSI port (controller)
- ▷ DK : SCSI disk
- ▷ MK : SCSI tape
- ▷ PU : DSSI port
- ▷ DU : DSSI disk
- ▷ MU : DSSI tape
- ▷ JK : SCSI monitor (or robot)
- ▷ DQ : (E)IDE Device (disk or CD-ROM)

Le reste du nom est composé de lettre/chiffre séparé par des points

- ▷ a,b,c,... adapter ID (one letter adapter designator)
- ▷ 0,1,2,... Device number (SCSI unit numbers are forced to 100x Node ID)
- ▷ 0,1,2,... Bus Node ID
- ▷ 0,1,2,... Channel Number
- ▷ 0,1,2,... Channel Number (used for multi-channel devices)
- ▷ 0,1,2,... Logical Slot number
 - ▷ EISA : they correspond to the physical slot numbers (1-3)
 - ▷ PCI :
 - slot 5= SCSI controller on system backplane (DS3300)
 - slot 6= On board VGA (DS3300)
 - slot 7= PCI to EISA bridge chip (DS3300)
 - slots 11 - 14 = Correspond to Physical PCI option slots : PCI11, PCI12, PCI13 and PCI14 (64bit) (DS3300)
- 6 0,1 Hose number : 0 PCL0 (32bit PCI) ; 1 EISA (DS3300)

Mais il existe une manière plus simple de nommer les périphériques de boot à partir de la console SRM :

- ▷ dva0 - First floppy drive, /dev/fd0 under Linux
- ▷ dqa0 - Primary IDE CD-ROM or hard disk as Master, /dev/hda under Linux
- ▷ dqa1 - Primary IDE CD-ROM or hard disk as Slave, /dev/hdb under Linux
- ▷ dka0 - SCSI disk on first bus, Device 0, /dev/sda under Linux
- ▷ ewa0 - First Ethernet Device, /dev/eth0 under Linux

Par exemple pour booter sur le disque scsi d'adresse 6, entrez :

```
>>> boot dka600
```

Normalement vous devriez avoir la console de démarrage de l'installation. A cet instant tapez ENTER pour un noyau 2.4, et linux26 pour un noyau 2.6.

J'ai opté dans un premier temps pour noyau 2.4. Je ne rentrerai pas dans les détails de l'installation de Linux Debian. Pour plus d'informations je vous revoie à l'adresse suivante <http://www.debian.org/releases/stable/alpha/index.html.fr>.

J'ai donc installé le système sur le premier disque de 18Go, et chaque disque logique RAID 1, est monté dans les répertoires /home/users[123].

L'installation terminée, il ne nous reste plus qu'à configurer le premier disque comme disque de démarrage. L'installateur Debian a installé **aboot**, le second bootloader, nécessaire au lancement du système d'exploitation.

Dans mon cas le premier disque de ma baie est référencé par l'adresse **sra0.0.0.8.1**. Je configure donc la variable **boot_def_dev** comme suit :

```
set bootdef_dev sra0.0.0.8.1
```

Un reboot à froid sera nécessaire pour la prise en compte. Normalement, votre serveur doit démarrer directement sur la distribution fraîchement installée qu'il faut finir de configurer.

3.1 Changement de noyau

Un fois le serveur opérationnel, j'ai décidé de passer au noyau de version supérieure. J'ai donc installé les paquets nécessaires à l'aide d'`apt-get`.

Un petit coup d'oeil dans le fichier `aboot.conf`, confirme que le noyau sera bien pris en compte au démarrage. De plus les liens (`vmlinuz`, `vmlinuz.old`, `initrd.img` et `initrd.img.old`) à la racine `/` pointent bien vers les bons fichiers.

```
yann@kawalpha:~$ more /boot/etc/aboot.conf
#
# aboot default configurations
#
0:2/vmlinuz ro initrd=/initrd.img root=/dev/rd/c0d0p4
1:2/vmlinuz.old ro initrd=/initrd.img.old root=/dev/rd/c0d0p4
2:3/vmlinux.new.gz ro root=/dev/sda2
3:3/vmlinux ro root=/dev/sda2
8:- ro root=/dev/sda2          # fs less boot of raw kernel
9:0/- ro root=/dev/sda2       # fs less boot of (compressed) ECOFF kernel
-
yann@kawalpha:~$ more /etc/aboot.conf
#
# aboot default configurations
#
0:2/vmlinuz ro initrd=/initrd.img root=/dev/rd/c0d0p4
1:2/vmlinuz.old ro initrd=/initrd.img.old root=/dev/rd/c0d0p4
2:3/vmlinux.new.gz ro root=/dev/sda2
3:3/vmlinux ro root=/dev/sda2
8:- ro root=/dev/sda2          # fs less boot of raw kernel
9:0/- ro root=/dev/sda2       # fs less boot of (compressed) ECOFF kernel
-
yann@kawalpha:~$
```

On peut noter que les fichiers `/etc/aboot.conf` et `/boot/etc/aboot.conf` sont identiques.

Je redémarre la machine, et à ma grande surprise le noyau actif est toujours le 2.4.27 de l'installation.

En regardant de plus près la documentation d'`aboot`, je m'aperçois que ce dernier va chercher les liens dans `/boot` et non à la racine `/`.

```
yann@kawalpha:~$ ls -al /
[...]
lrwxrwxrwx  1 root root    28 2006-04-07 11:00 vmlinuz -> boot/vmlinuz-2.6.8-3-generic
lrwxrwxrwx  1 root root    29 2006-04-07 09:30 vmlinuz.old -> boot/vmlinuz-2.4.27-2-g
lrwxrwxrwx  1 root root    31 2006-04-07 11:00 initrd.img -> boot/initrd.img-2.6.8-3-g
lrwxrwxrwx  1 root root    32 2006-04-07 09:30 initrd.img.old -> boot/initrd.img-2.4.2
[...]

yann@kawalpha:~$ ls -al /boot/
```



```
total 15115
drwxr-xr-x  4 root root    1024 2006-04-07 12:51 .
drwxr-xr-x 21 root root    4096 2006-04-07 11:00 ..
-rw-r--r--  1 root root   88064 2004-09-06 07:42 bootlx
-rw-r--r--  1 root root   40310 2005-11-22 18:09 config-2.4.27-2-generic
-rw-r--r--  1 root root   49832 2006-02-09 12:06 config-2.6.8-3-generic
drwxr-xr-x  2 root root    1024 2006-04-07 09:30 etc
lrwxrwxrwx  1 root root     26 2006-04-07 12:51 initrd.img -> initrd.img-2.6.8-3-gener
-rw-r--r--  1 root root 5308416 2006-04-07 09:30 initrd.img-2.4.27-2-generic
-rw-r--r--  1 root root 5619712 2006-04-07 11:00 initrd.img-2.6.8-3-generic
lrwxrwxrwx  1 root root     27 2006-04-07 09:30 initrd.img.old -> initrd.img-2.4.27-2-
drwxr-xr-x  2 root root   12288 2006-04-07 09:14 lost+found
-rw-r--r--  1 root root   32256 2004-09-06 07:42 net_about.nh
-rw-r--r--  1 root root    512 2004-09-06 07:42 net_pad
-rw-r--r--  1 root root  629027 2005-11-23 03:11 System.map-2.4.27-2-generic
-rw-r--r--  1 root root  989034 2006-02-09 16:06 System.map-2.6.8-3-generic
lrwxrwxrwx  1 root root     23 2006-04-07 12:51 vmlinuz -> vmlinuz-2.6.8-3-generic
-rw-r--r--  1 root root 1276038 2005-11-23 03:10 vmlinuz-2.4.27-2-generic
-rw-r--r--  1 root root 1349440 2006-02-09 16:06 vmlinuz-2.6.8-3-generic
lrwxrwxrwx  1 root root     24 2006-04-07 09:30 vmlinuz.old -> vmlinuz-2.4.27-2-generi
yann@kawalpha:~$
```

Il est d'ailleurs possible de choisir de manière interactive au démarrage, par l'intermédiaire du chargeur `about`, le noyau désiré.

La commande suivante, permet, à partir du premier disque SCSI, de choisir le noyau à charger.

```
>>> boot dka0 -fl i
about> ?
h, ?          Display this message
q            Halt the system and return to SRM
p 1-8        Look in partition <num> for configuration/kernel
l            List pre-configured kernels
d <dir>      List directory <dir> in current filesystem
b <file> <args> Boot kernel in <file> (- for raw boot)
              with arguments <args>
0-9          Boot pre-configuration 0-9 (list with 'l')
```

Dans un premier temps, il faut choisir la partition sur laquelle sont stockés les fichiers noyau et `initrd`. Dans mon cas, ils sont situés sur la seconde partition. Voici les commandes :

```
>>> boot sra0.0.0.8.1 -fl i
about> ?
h, ?          Display this message
q            Halt the system and return to SRM
p 1-8        Look in partition <num> for configuration/kernel
l            List pre-configured kernels
d <dir>      List directory <dir> in current filesystem
b <file> <args> Boot kernel in <file> (- for raw boot)
              with arguments <args>
0-9          Boot pre-configuration 0-9 (list with 'l')
about>p 2 #selectionne la seconde partition
```



```
about>l #permet de lister les noyaux disponibles
about>0 #charge la première configuration d'about.conf
```

Comme about va chercher les noyaux sur la seconde partition, il faut que les liens soient dans le répertoire boot et non à la racine /. En effet dans ma configuration, le répertoire /boot possède une partition séparée.

```
yann@kawalpha:~$ mount
/dev/rd/c0d0p4 on / type ext3 (rw,errors=remount-ro)
proc on /proc type proc (rw)
sysfs on /sys type sysfs (rw)
devpts on /dev/pts type devpts (rw,gid=5,mode=620)
tmpfs on /dev/shm type tmpfs (rw)
/dev/rd/c0d0p2 on /boot type ext2 (rw)
/dev/rd/c0d1p1 on /home/users1 type ext3 (rw)
/dev/rd/c0d2p1 on /home/users2 type ext3 (rw)
/dev/rd/c0d3p1 on /home/users3 type ext3 (rw)
tmpfs on /dev type tmpfs (rw,size=10M,mode=0755)
yann@kawalpha:~$
```

Une fois les liens recréés dans le répertoire /boot, un redémarrage confirme le chargement du bon noyau.

```
yann@kawalpha:~$ uname -a
Linux kawalpha 2.6.8-3-generic #1 Thu Feb 9 11:33:55 UTC 2006 alpha GNU/Linux
yann@kawalpha:~$ more /proc/cpuinfo
cpu                : Alpha
cpu model          : EV6
cpu variation      : 7
cpu revision       : 0
cpu serial number  :
system type        : Tsunami
system variation   : Goldrush
system revision    : 0
system serial number :
cycle frequency [Hz] : 500000000
timer frequency [Hz] : 1024.00
page size [bytes]   : 8192
phys. address bits : 44
max. addr. space # : 255
BogoMIPS           : 971.28
kernel unaligned acc : 0 (pc=0,va=0)
user unaligned acc  : 0 (pc=0,va=0)
platform string     : AlphaServer DS20 500 MHz
cpus detected       : 1
L1 Icache           : 64K, 2-way, 64b line
L1 Dcache           : 64K, 2-way, 64b line
L2 cache            : 4096K, 1-way, 64b line
L3 cache            : n/a
yann@kawalpha:~$
```

3.2 Problème périphériques souris

La démarrage sur le noyau 2.6.8 se passe bien, jusqu'au démarrage de gdm. Ce dernier démarre bien mais la souris est bloquée. Tout ceci fonctionnait pourtant bien avec la noyau 2.4. Un petit tour sur <http://www.debian.org/releases/stable/alpha/release-notes.fr.txt> nous en apprend un peu plus.

5.2.2. Configuration de la souris

```
À nouveau, à cause de changement dans la couche d'entrée du noyau, il se peut que vous deviez reconfigurer le système de fenêtrage X et 'gpm' si votre souris ne fonctionne pas après avoir fait une mise à jour vers un noyau 2.6. La cause la plus probable est que le périphérique recevant les données de la souris a changé. Il se peut également que vous deviez charger différents modules.
```

Une petite modification du fichier `/etc/modules` permet de réactiver la souris. Dans un premier temps on peut faire :

```
# modprobe psmouse
# startx
```

afin de voir si tout fonctionne. Puis pour que ceci soit définitif, on ajoute la ligne suivante dans le fichier `/etc/modules` :

```
psmouse
```

```
yann@kawalpha:~$ more /etc/modules
# /etc/modules: kernel modules to load at boot time.
#
# This file should contain the names of kernel modules that are
# to be loaded at boot time, one per line. Comments begin with
# a "#", and everything on the line after them are ignored.
```

```
sr_mod
psmouse
yann@kawalpha:~$
```

Un dernier redémarrage à froid, confirmera la bonne marche du serveur.

4 Conclusion

C'en est fini de cet article, toutes remarques et corrections sont les bienvenues à l'adresse morere@univ-metz.fr