

# Configuration des fonctions réseau & compilation du noyau Linux

Philippe Latu  
philippe.latu(at)inetdoc.net

<https://www.inetdoc.net>

## Résumé

Dans ce support de travaux pratiques, on se propose de préparer un système GNU/Linux pour être utilisé comme équipement d'interconnexion réseau. Après avoir passé en revue les fonctions réseau utiles du noyau Linux et sélectionné les pilotes des périphériques effectivement présents sur la plateforme matérielle, on construit un paquet de noyau Linux à partir de ses sources.

## Table des matières

1. Copyright et Licence .....	1
1.1. Méta-information .....	1
1.2. Conventions typographiques .....	2
2. Le noyau courant et son arborescence .....	2
3. Les sources du noyau Linux .....	4
4. La configuration du noyau Linux .....	7
5. La compilation & l'installation du nouveau noyau Linux .....	9
6. Documents de référence .....	10

## 1. Copyright et Licence

Copyright (c) 2000,2024 Philippe Latu.  
Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

Copyright (c) 2000,2024 Philippe Latu.  
Permission est accordée de copier, distribuer et/ou modifier ce document selon les termes de la Licence de Documentation Libre GNU (GNU Free Documentation License), version 1.3 ou toute version ultérieure publiée par la Free Software Foundation ; sans Sections Invariables ; sans Texte de Première de Couverture, et sans Texte de Quatrième de Couverture. Une copie de la présente Licence est incluse dans la section intitulée « Licence de Documentation Libre GNU ».

### 1.1. Méta-information

Ce document est écrit avec [DocBook XML](#) sur un système [Debian GNU/Linux](#). Il est disponible en version imprimable au format PDF : [interco.kernel.qa.pdf](#).

Toutes les commandes utilisées dans ce document ne sont pas spécifiques à une version particulière des systèmes UNIX ou GNU/Linux. C'est la distribution Debian GNU/Linux qui est utilisée pour les tests présentés. Voici une liste des paquets contenant les commandes :

- coreutils - The GNU core utilities
- procps - Utilitaires pour le système de fichiers /proc
- pciutils - Utilitaires PCI Linux
- kmod - outils pour gérer les modules du noyau Linux
- make - utilitaire pour mener la compilation

- libncurses5 - Bibliothèques partagées pour l'utilisation d'un terminal
- libncurses5-dev - bibliothèques de développement pour ncurses
- kernel-package - A utility for building Linux kernel related Debian packages
- fakeroot - outil de simulation des privilèges superutilisateur

## 1.2. Conventions typographiques

---

Tous les exemples d'exécution des commandes sont précédés d'une invite utilisateur ou prompt spécifique au niveau des droits utilisateurs nécessaires sur le système.

- Toute commande précédée de l'invite \$ ne nécessite aucun privilège particulier et peut être utilisée au niveau utilisateur simple.
- Toute commande précédée de l'invite # nécessite les privilèges du super-utilisateur.

## 2. Le noyau courant et son arborescence

---

Avant d'attaquer la compilation d'un nouveau noyau à partir de ses sources, on doit identifier et localiser les différents composants du noyau en cours d'exécution sur le système.

Le jeu de questions ci-dessous suppose que la configuration système est directement issue de l'installation de la distribution Debian GNU/Linux. Le noyau courant exécuté est fourni via un paquet de la distribution.

Q1. Quelle est la commande UNIX usuelle qui identifie le noyau et sa version ?

Effectuer une recherche dans les pages de manuels des commandes installées sur le système avec une requête du type : `apropos informations, système`.

C'est la commande `uname` qui identifie le noyau courant. Pour interroger les pages de manuels à l'aide de la commande `apropos`, il faut que les paquets correspondant soient installés et que l'index de recherche soit construit.

Pour interroger les pages de manuels, on contrôle la liste des paquets correspondants installés et on lance manuellement la construction de l'index de recherche :

```
$ aptitude search ~imanpages
i  manpages          - Pages de manuel pour le système GNU/Linux
i  A manpages-dev    - Pages de manuel sur l'utilisation de GNU/Linux pour le développement
i  manpages-fr       - Version française des pages de manuel sur l'utilisation de GNU/Linux
i  manpages-fr-extra - Version française des pages de manuel
```

```
<snip/>
# /etc/cron.daily/man-db
```

```
<snip/>
$ apropos -a informations système
dumpe2fs (8)  - Afficher des informations sur le système de fichiers ext2/ext3/ext4
fstab (5)    - Informations statiques sur les systèmes de fichiers
proc (5)     - Pseudosystème de fichiers d'informations sur les processus
uname (1)   - Afficher des informations sur le système
```

Pour obtenir la version courante du noyau exécuté :

```
$ uname -a
Linux vm0 4.12.0-2-686-pae #1 SMP Debian 4.12.12-2 (2017-09-11) i686 GNU/Linux
```

Q2. Où est placée l'image de la partie monolithique du noyau courant ?

Repérer le paquet Debian correspondant au noyau et retrouver l'image dans la liste des fichiers de ce paquet.

Une fois la version courante du noyau identifiée à l'aide de la commande `uname`, on peut faire la correspondance avec les paquets de noyau installés.

```
$ aptitude search ~ilinux-image
i A linux-image-4.12.0-1-686-pae - Linux 4.12 for modern PCs
i   linux-image-4.12.0-2-686-pae - Linux 4.12 for modern PCs
i   linux-image-686-pae          - Linux pour PC modernes - métapaquet
```

Connaissant le nom du paquet de noyau installé on peut lister les fichiers qu'il contient. À partir de cette liste on peut localiser la partie monolithique du noyau ainsi que ses modules dans l'arborescence du système de fichiers.

C'est dans le répertoire `/boot` que sont placées les images des noyaux disponibles sur un système GNU/Linux.

```
$ ls -A1 /boot/ | grep 4.12.0-2
config-4.12.0-2-686-pae ❶
initrd.img-4.12.0-2-686-pae ❷
System.map-4.12.0-2-686-pae ❸
vmlinuz-4.12.0-2-686-pae ❹
```

- ❶ Fichier de configuration du noyau de la distribution. Il contient l'ensemble des options qui ont été sélectionnées par le responsable du paquet. C'est une configuration très complète dans la mesure où un noyau publié dans une distribution doit supporter le maximum de matériel.
- ❷ Image compressée du disque RAM d'initialisation contenant une arborescence racine simplifiée, des outils et l'ensemble des modules du noyau. Cette technique d'initialisation est la seule qui puisse fonctionner sur des systèmes sans disque dur où sur lesquels aucun système GNU/Linux n'a encore été installé.
- ❸ Fichier de cartographie des appels de fonctions du noyau. Cette cartographie est une aide à la mise au point pour les développeurs. On y trouve une identification nominative des fonctions en cas de problème au lieu d'adresses numériques en hexadécimal.
- ❹ Fichier image de la partie monolithique du noyau. C'est ce fichier qui est utilisé par le gestionnaire de démarrage pour lancer le système d'exploitation. Le gestionnaire de démarrage y accède directement à l'aide d'un appel BIOS.

Q3. Où sont placés les fichiers des modules correspondant au noyau courant ?

Comme dans le cas précédent, la liste des fichiers du paquet permet de retrouver l'arborescence de stockage des modules.

On peut parcourir la liste des fichiers contenus dans le paquet de noyau et effectuer des recherches par mots clés en utilisant la commande suivante :

```
$ dpkg -L linux-image-4.12.0-2-686-pae | egrep -e 'kernel$'
/lib/modules/4.12.0-2-686-pae/kernel
/lib/modules/4.12.0-2-686-pae/kernel/arch/x86/kernel
```

La liste ci-dessus montre que les modules du noyau sont placés dans le répertoire `/lib/modules/4.12.0-2-686-pae/kernel`.

Q4. Dans quel cas de figure utilise-t-on l'arborescence ou le disque RAM ?

Il faut bien différencier l'utilisation du disque RAM `initrd-*` de l'arborescence installée sur le disque du système.

Le fichier image du disque RAM d'initialisation **a déjà été identifié** ci-dessus.

Ce fichier est utilisé lors du lancement du système d'exploitation. Il est reconnu par le gestionnaire de démarrage de la même façon que la partie monolithique du noyau. Une fois le système complètement initialisé, les opérations de (chargement|déchargement) des modules utilisent l'arborescence du disque dur : `/lib/modules/`uname -r`/`.

Q5. Que contiennent les arborescences `/proc` et `/sys` ?

Consulter les documents ressource [sysfs](#) et [Linux Filesystem Hierarchy](#)

L'arborescence `/sys` est une représentation visible de l'arbre des périphériques physiques vus par le noyau. Elle est construite dynamiquement en fonction des branchements «à chaud»

effectués sur les différents bus de la machine. Les informations répertoriées dans cette arborescence sont du type : nom de périphérique, canal DMA, vecteur d'interruption, tensions d'alimentation, etc.

L'arborescence `/proc` comprend l'ensemble des paramètres du noyau en cours d'exécution. Ces paramètres sont modifiables en cours de fonctionnement. L'exemple emblématique, vis-à-vis de ces travaux pratiques est donné par l'ensemble des «réglages» possibles sur les machines d'états de la pile des protocoles réseau. La liste des paramètres donnée par la commande `ls /proc/sys/net/ipv4/` en donne un aperçu.

Q6. Quelle est la commande qui permet de lister les modules chargés en mémoire ?

À quel paquet appartient elle ?

Rechercher dans la base de données des paquets de la distribution les informations relatives aux manipulations sur les modules.

```
aptitude search '?description("modules du noyau Linux")'  
i A kmod - outils pour gérer les modules du noyau Linux
```

Le paquet `kmod` fournit la commande `lsmod` ainsi que les autres outils de manipulation sur les modules.

```
$ dpkg -L kmod | grep sbin/  
/sbin/depmod  
/sbin/insmod  
/sbin/lsmod  
/sbin/modinfo  
/sbin/modprobe  
/sbin/immod
```

Q7. Quelles sont les commandes qui permettent de charger un module en mémoire «manuellement» ?

Identifier celle qui traite automatiquement les dépendances entre modules.

Rechercher les informations dans la liste des fichiers du paquet ainsi que dans les pages de manuels des commandes.

On dispose de deux commandes : `insmod` et `modprobe`. Seule la commande `modprobe` traite les dépendances au (chargement/déchargement) d'un module. Illustration avec un module de gestion des dispositifs de stockage sur le bus USB.

```
# modprobe -v usb-storage  
insmod /lib/modules/4.12.0-2-686-pae/kernel/drivers/usb/storage/usb-storage.ko
```

Q8. Quelles sont les commandes qui permettent de retirer un module de la mémoire «manuellement» ?

Identifier les options de la commande qui traite automatiquement les dépendances entre modules.

Rechercher les informations dans les pages de manuels des commandes.

Comme dans le cas précédent, c'est la commande `modprobe` qui retire de la mémoire les modules associés au déchargement.

```
# modprobe -rv usb-storage  
rmmod usb_storage
```

### 3. Les sources du noyau Linux

Il faut bien reconnaître que s'attaquer à toutes les options de configuration du noyau Linux en partant de zéro est une tâche particulièrement ardue. Pour rendre la démarche plus aisée, on se propose de partir de la configuration fournie avec le paquet de la distribution. En procédant par modifications élémentaires à partir de cette configuration réputée sûre puisque permettant le fonctionnement du système actuel, on limite ainsi les possibilités d'erreurs.

Q9. Quels sont les principaux canaux de diffusion des sources du noyau Linux ?

Rechercher un site web, un dépôt de code en ligne et le nom du paquet de la distribution.

- Le site principal de publication des sources du noyau Linux est à l'adresse <http://www.kernel.org/>.
- Le développement du système de contrôle de version git a été initié par les développeurs du noyau Linux. Depuis, des services en lignes ont été bâtis à partir de git. Les branches de développement du noyau sont disponibles sur le site [GitHub](https://github.com/torvalds/linux) à l'adresse <https://github.com/torvalds/linux>.
- La distribution Debian GNU/Linux propose des paquets contenant les sources qui on servi à construire les paquets de noyau. Pour identifier ces paquets, on effectue une recherche dans le catalogue de la distribution.

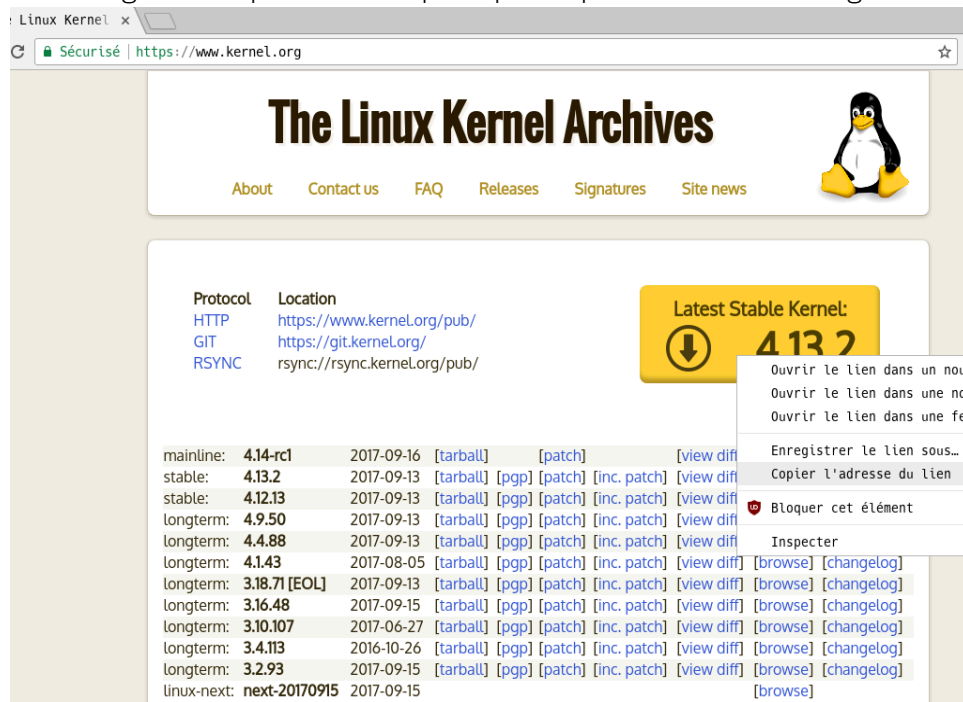
```
$ aptitude search linux-source
p linux-source - Linux kernel source (meta-package)
p linux-source-4.12 - Linux kernel source for version 4.12 with Debian patches
```

Q10. Donner un exemple de téléchargement des sources du noyau sans passer par une interface graphique ?

Rechercher un outil permettant de lancer un téléchargement HTTP(s).

Lorsque l'on utilise des serveurs qui ne possèdent ni écran ni clavier, il est nécessaire d'effectuer les opérations sans recours à une interface graphique. Les outils les plus courants dans ce contexte sont `url` et `wget`.

- Téléchargement à partir du site principal de publication kernel.org.



### Téléchargement des sources du noyau Linux - vue complète

```
$ wget https://cdn.kernel.org/pub/linux/kernel/v4.x/linux-4.13.2.tar.xz
-- https://cdn.kernel.org/pub/linux/kernel/v4.x/linux-4.13.2.tar.xz
Résolution de cdn.kernel.org (cdn.kernel.org)... 2a04:4e42:1d::432, 151.101.121.176
Connexion à cdn.kernel.org (cdn.kernel.org)|2a04:4e42:1d::432|:443... connecté.
requête HTTP transmise, en attente de la réponse... 200 OK
Taille : 100574388 (96M) [application/x-xz]
Sauvegarde en : « linux-4.13.2.tar.xz »
<snip/>
```

- Téléchargement à partir du gestionnaire de paquets de la distribution.

```
# aptitude install linux-source
Les NOUVEAUX paquets suivants vont être installés :
  linux-source linux-source-4.12{a} make{a}
0 paquets mis à jour, 3 nouvellement installés, 0 à enlever et 0 non mis à jour.
Il est nécessaire de télécharger 102 Mo d'archives. Après dépaquetage, 103 Mo
seront utilisés.
Voulez-vous continuer ? [Y/n/?]
```

Q11. Quel est le groupe système qui permet de compiler un noyau ou des modules ?

Rechercher le groupe consacré aux manipulations des sources dans la liste des groupes système.

On cherche la chaîne `src` dans le fichier `/etc/group` et on ajoute l'utilisateur normal dans ce groupe.

```
# grep src /etc/group
src:x:40:

# adduser etu src
Ajout de l'utilisateur « etu » au groupe « src »...
Ajout de l'utilisateur etu au groupe src
Fait.

# id etu
uid=1000(etu) gid=1000(etu) groupes=1000(etu),24(cdrom),25(floppy),
29(audio),30(dip),40(src),44(video),46(plugdev)
```

Q12. Quel est le répertoire du système dédié au stockage des sources du noyau Linux ?

Faire une recherche dans le document [Linux Filesystem Hierarchy](#).

Vérifier que ce répertoire appartient bien au groupe `src`.

C'est le répertoire `/usr/src` qui doit accueillir les sources du noyau.

On vérifie que les membres du groupe système `src` ont bien accès en écriture à ce répertoire.

```
# chgrp -R src /usr/src

# chmod 2775 /usr/src
```

Q13. Quelles sont les commandes «rituelles» d'installation des sources du noyau Linux ?

Pour chaque commande, expliquer les opérations réalisées et justifier le choix des options.

Il faut consulter les ressources suivantes : [Debian Linux Kernel Handbook](#) et [Manuel de référence Debian - Chapitre 9](#).

Pour traiter cette question, on utilise le fichier source obtenu à l'aide du gestionnaire de paquets. D'après les documents de référence on doit utiliser la séquence de commandes suivante.

```
$ cd /usr/src/
$ tar xf linux-source-4.12.tar.xz❶
$ ln -s linux-source-4.12 linux❷
$ cd linux
$ cp /boot/config-4.12.0-2-686-pae .config❸
$ make menuconfig❹
```

- ❶ Extraction de l'arborescence des sources du noyau.
- ❷ Création d'un lien symbolique sur l'arborescence de travail. L'utilisation de ce lien permet de conserver plusieurs arborescences de sources. De cette façon, on peut travailler sur plusieurs versions de noyau.
- ❸ Copie du fichier de configuration fourni avec le paquet de noyau. Ce fichier est réputé fiable puisqu'il correspond au noyau en cours d'exécution et que le système est opérationnel. Cette opération est optionnelle. En l'absence du fichier `.config` dans l'arborescence des sources du noyau, la commande suivante procède à la copie de la configuration du noyau courant.

- ④ Lancement de l'interface des menus de configuration des options du noyau Linux. C'est à ce niveau que les «choses sérieuses» commencent.

La dernière commande n'est utilisable que si le paquet de bibliothèques de développement `ncurses` est installé. `aptitude install libncurses-dev`.

## 4. La configuration du noyau Linux

---

On se propose de configurer un système d'interconnexion. Le noyau correspondant doit donc comprendre les éléments suivants.

- Un cœur système monolithique : microprocesseur, périphériques non réseau et système de fichiers
- Le support des fonctions réseau nécessaires au routage
- Le support du filtrage netfilter sous forme modulaire
- Un pilote d'interface réseau Ethernet sous forme modulaire
- Les fonctions de l'ancien sous-système RNIS sous forme modulaire
- Un pilote d'interface RNIS sous forme modulaire

Q14. Quelle est la commande utilisée pour les opérations de configuration et de compilation ?

Toutes les opérations de compilation du noyau étant basées sur des `Makefiles`, c'est la commande `make` qui sert aussi pour la configuration.

Q15. Comment obtenir la liste des options de cette commande ?

La commande `make help` donne la liste des options disponibles.

Q16. Quelles sont les 3 options de configuration du noyau ?

Préciser les différences entre ces 3 options.

Les 3 commandes sont `make config`, `make menuconfig` et `make xconfig`.

Il est préférable d'utiliser la commande `make menuconfig`. C'est le meilleur compromis entre facilité de navigation et administration distante. Les bibliothèques de développement `ncurses` ne consomment que très peu de ressources CPU et l'utilisation d'une interface graphique sur un serveur est à proscrire.

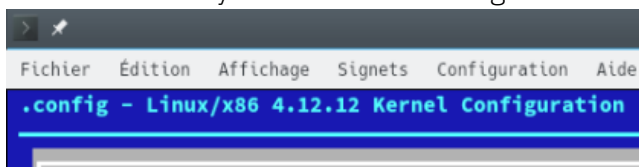
Q17. Sans opération préalable, quel est le fichier contenant les options de configuration du noyau utilisé ?

C'est le fichier texte `.config` qui contient l'ensemble des options de configuration du noyau Linux courant. Il est placé à la racine de l'arborescence des sources du noyau ; soit le répertoire `/usr/src/linux` dans notre cas.

Le fichier «patron» de configuration pour ces travaux pratiques doit donc être copié dans le répertoire `/usr/src/linux` et renommé `.config`. L'opération a déjà été effectuée à la [Q : Q13](#).

Q18. Une fois la commande de configuration exécutée, comment identifier la version du noyau à compiler ?

La version du noyau en cours de configuration est indiquée en haut à gauche de l'écran.



### Identification version noyau Linux - vue complète

Q19. Quelles sont les options utiles des rubriques Networking Support et Networking options ?

On accède aux différents types de réseaux supportés par le noyau Linux via l'item Networking Support.

```
[ ] 64-bit kernel
  General setup --->
[*] Enable loadable module support --->
[*] Enable the block layer --->
  Processor type and features --->
  Power management and ACPI options --->
  Bus options (PCI etc.) --->
  Executable file formats / Emulations --->
[*] Networking support --->
  Device Drivers --->
  Firmware Drivers --->
  File systems --->
  Kernel hacking --->
  Security options --->
--*-- Cryptographic API --->
[*] Virtualization --->
  Library routines --->
```

### Accès aux fonctions réseau - vue complète

On accède aux fonctions réseau du noyau Linux via l'item Networking options.

```
-- Networking support
| Networking options --->
[ ] Amateur Radio support --->
< > CAN bus subsystem support --->
< > IrDA (infrared) subsystem support --->
< > Bluetooth subsystem support --->
< > RxRPC session sockets
Wireless --->
{M} RF switch subsystem support --->
< > Plan 9 Resource Sharing Support (9P2000) (Experimental)
```

### Accès aux fonctions réseau du noyau Linux - vue complète

À partir du support [Fonctions réseau du noyau Linux](#) et de l'organisation des menus, on distingue les options génériques, telles que le support des sockets, des options spécifiques telles que celles relatives au filtrage.

Q20. Quelles sont les options utiles des rubriques Device Drivers puis Network device support ?

Voir le support [Fonctions réseau du noyau Linux](#) pour s'orienter dans les options à sélectionner.

Pour accéder au catalogue des interfaces réseau supportées par le noyau il faut passer par la catégorie des pilotes de périphériques ou Device Drivers pour accéder à l'item Network device support.

Q21. Quelles sont les options utiles de la rubrique ISDN subsystem ?

À partir de la liste des pilotes de périphériques du noyau, on accède aux paramétrage du sous-système RNIS/ISDN.

```
Fichier  Édition  Affichage  Signets  Configuration  Aide
.config - Linux/x86 4.12.12 Kernel Configuration
> Device Drivers > Network device support > ISDN support > Search (isdn)
Symbol: ISDN [=y]
Type : boolean
Prompt: ISDN support
Location:
  -> Device Drivers
(1) -> Network device support (NETDEVICES [=y])
Defined at drivers/isdn/Kconfig:5
Depends on: NET [=y] && NETDEVICES [=y] && !S390 && !UML
```

### Accès au sous-système RNIS/ISDN - vue complète



Il existe trois types d'utilisation des connexions RNIS/ISDN dans le noyau Linux.

- Le plus récent utilise un mécanisme de sockets adapté aux fonctions réseau actuelles du noyau.
- Le plus ancien hérite des noyaux de la série 2.2.xx. Il comprend une machine d'état logicielle autonome de gestion de l'établissement du maintien et de la libération des connexions. C'est ce type de connexion que l'on utilise dans la suite des travaux pratiques de la série.
- Il existe un troisième type qui utilise le standard CAPI. Il s'agit d'une interface logicielle normalisée entre le noyau et le périphérique matériel.

```
--- ISDN support
<M> Old ISDN4Linux (deprecated) --->
<M> CAPI 2.0 subsystem --->
< > Siemens Gigaset support ----
< > Hypercope HYSN cards (Champ, Ergo, Metro) support (module only)
< > Modular ISDN driver ----
```

### Types de connexions RNIS/ISDN - vue complète

Le catalogue des paramètres utilisables avec le protocole PPP associé au sous-système RNIS/ISDN historique du noyau Linux est donné ci-dessous.

```
--- Old ISDN4Linux (deprecated)
[*] Support synchronous PPP
[*] Use VJ-compression with synchronous PPP
[*] Support generic MP (RFC 1717)
[*] Filtering for synchronous PPP
<M> Support BSD compression
[ ] Support audio via ISDN (NEW)
ISDN feature submodules --->
*** ISDN4Linux hardware drivers ***
Passive cards --->
```

### Paramètres PPP du sous-système RNIS/ISDN - vue complète

Le modèle des cartes implantées dans les postes de travaux pratiques est de type AVM Fritz/PCI 2.0.

## 5. La compilation & l'installation du nouveau noyau Linux

Q22. Quelle est l'option à utiliser avec les sources du noyau pour construire des paquets de la distribution Debian GNU/Linux ?

Rechercher la clé `deb` dans la liste des options du `Makefile` des sources du noyau.

La recherche dans les options permet d'identifier la directive de construction des paquets binaires de la distribution : `bindeb-pkg`.

```
$ make help | less
```

Suivant l'état antérieur de l'installation système, la liste des dépendances est plus ou moins importante lors du lancement de la compilation du noyau.

```
# sudo aptitude install -R fakeroot bison flex libelf-dev libssl-dev
```

Q23. Comment lancer la compilation du noyau ?

Pour faciliter les opérations de (dé|ré)installation du noyau, on se propose de construire un paquet Debian de noyau Linux. L'utilisation d'un paquet permet de s'assurer que tous les fichiers nécessaires ont bien été (copiés|supprimés) dans l'arborescence du système.

```
$ pwd
/usr/src/linux
$ make -j$(grep -c '^processor' /proc/cpuinfo) bindeb-pkg
```

Q24. Quelles sont les étapes d'installation du noyau compilé ?

Quel outil faut-il utiliser pour gérer les paquets localement sur le système ?

Une fois les paquets de noyau construits, il ne reste plus qu'à procéder à l'installation de ces paquets locaux. Cette étape fait appel à l'outil de gestion de bas niveau des paquets Debian : dpkg. Cette opération nécessite les droits du super-utilisateur.

```
# pwd
/usr/src
# dpkg -i linux-image*.deb linux-libc*.deb
```

Après cette installation de paquet de noyau on peut valider la liste des paquets correspondant installés.

```
$ aptitude search ~ilinux-
```

Q25. Comment vérifier que le nouveau noyau sera disponible lors de l'initialisation du système ?

Identifier le gestionnaire d'amorce installé sur le système.

L'opération d'installation du paquet de noyau intègre l'ajout d'une nouvelle entrée dans le gestionnaire de démarrage.

On peut valider la liste des noyaux disponibles au niveau du gestionnaire d'amorce en faisant appel à la commande update-grub.

```
$$ sudo update-grub
Generating grub configuration file ...
Found linux image: /boot/vmlinuz-5.2.14
Found initrd image: /boot/initrd.img-5.2.14
Found linux image: /boot/vmlinuz-5.2.0-2-amd64
Found initrd image: /boot/initrd.img-5.2.0-2-amd64
done
```

Une fois toutes ces étapes franchies, il ne reste plus qu'à relancer le système et vérifier que le noyau exécuté est bien celui qui a été recompilé à partir des sources.

## 6. Documents de référence

---

[Debian Linux Kernel Handbook](#)

**Debian Linux Kernel Handbook** : guide sur les techniques de construction d'un paquet Debian de noyau Linux.

[Manuel de référence Debian](#)

**Manuel de référence Debian - Chapitre 9** : La section 9.7 traite des opérations de configuration et de compilation d'un noyau Linux.