

Sommaire

Les packages nécessaires pour compiler.....	1
Packages nécessaires pour compiler le noyau.....	1
Packages nécessaires pour compiler une application.....	2
Compilateurs.....	2
Librairies standard.....	2
Make.....	2
Utilitaires.....	3
 Copyright.....	 5

Les packages nécessaires pour compiler

Les packages nécessaires pour compiler

par Jean-Christophe

Que faut-il installer pour pouvoir compiler un programme.

Afin de compiler le noyau (voir rubrique noyau) ou une autre application, vous avez besoin d'un certain nombre de packages, sans quoi vous aurez tout un tas d'erreurs qui conduiront vos compilations à l'échec. Nous allons voir ci-dessous quels packages sont nécessaires dans quel cas (bien sûr, les versions des packages ne sont pas indiquées ici, utilisez les packages livrés avec votre distribution).

Pour voir si un tel package est installé sur votre ordinateur, utilisez les utilitaires graphiques comme kackage ou gnorpm, ou utilisez rpm en ligne de commande. Pour rechercher un package avec rpm, tapez :

```
[jice@taz jice]$ rpm -qa | grep make
```

Cette commande vous renverra tous les packages dont le nom contient "make".

Packages nécessaires pour compiler le noyau

Pour les distributions basées sur les packages RPM (Red Hat, Mandrake...), la commande ci-dessous donne la liste des packages et des fichiers dont dépend le package des sources du noyau (`kernel-source`).

Voyons parmi l'information que retourne cette commande les packages nécessaires à la compilation du noyau.

```
[jice@taz jice]$ rpm -qR kernel-source
```

```
kernel-headers = 2.2.13-7mdk -> pkg kernel-headers
                                -> pkg glibc-devel
                                -> pkg ncurses-devel
glibc-devel
ncurses-devel
/bin/sh
ld-linux.so.2
libc.so.6
libm.so.6
/usr/bin/perl
libc.so.6(GLIBC_2.0)
libc.so.6(GLIBC_2.1)
libm.so.6(GLIBC_2.0)
libm.so.6(GLIBC_2.1)
```

- Pour le package `kernel-headers`, la version est spécifiée (ici `2.2.13-7mdk`).
- Les deux lignes suivantes spécifient les packages `glibc-devel` et `ncurses-devel`.
- Les lignes suivantes spécifient d'autres fichiers nécessaires pour ce package.
 - `/bin/sh` est le shell et provient par exemple du package `bash`.
 - `/usr/bin/perl` est l'interpréteur du langage PERL, qu'on trouve dans le package `perl`.
 - `ld-linux.so.2`, `libc.so.6` et `libm.so.6` se trouvent dans `/lib` et le package `glibc`, dont la version est précisée ici : `2.0` ou `2.1`.

Afin de compiler le noyau, vous aurez besoin d'autres packages :

- le compilateur C, nommé `gcc` ou `pgcc` ou `egcs`. Par exemple les packages (sur une Mandrake) :

- pgcc
- colorgcc (optionnel)
- le programme qui pilote la compilation : make
- make
- Les utilitaires binaires, texte et shell :
binutils, textutils & sh-utils
- Si vous compilez sous X (avec la commande `make xconfig`), vous aurez besoin de Tcl/Tk :
tcl, tclx, tix, tk
(ces packages sont généralement installés par défaut)

Résumé des packages pour compiler le noyau :

(si j'en ai oublié, chez altern.org dites-le moi :)

- kernel-headers
- glibc-devel
- ncurses-devel
- tcl, tclx, tix, tk
- bash (ou sh)
- perl
- pgcc
- make
- binutils, textutils & sh-utils

Packages nécessaires pour compiler une application

Dans ce cas la réponse est moins simple, car les packages nécessaires dépendent de chaque logiciel qu'on compile (suivant le langage utilisé, les bibliothèques utilisées, etc.).

Lorsque vous installez votre nouvelle distribution, choisissez tout ce qui ressemble à "C development", "Development libraries", "C++ development", etc.

Dans quasiment tous les cas, il faut au moins un compilateur, les utilitaire standard (make et binutils) et les librairies standard, soit :

Compilateurs

- pgcc
- pgcc-c++ (les programmes KDE utilisent C++)
- colorgcc
- cpp (le préprocesseur C)

Librairies standard

- glibc-devel
- libstdc++-devel

Make

- make
- pmake

- automake

Utilitaires

- autoconf
- binutils
- textutils
- sh-utils
- bison
- flex
- gawk (ou awk)
- gettext (dont msgfmt...)
- libtool
- patch
- perl
- sed

Ca fait déjà pas mal, je vous l'accorde, mais ce n'est pas fini !

En effet, chaque programme particulier utilisant une ou plusieurs bibliothèques, vous avez besoin de la version de développement de la librairie (nommée en général `librairie-devel`) afin de compiler votre programme. Si vous voulez être tranquilles, je vous conseille d'installer tous les packages de votre distribution qui se terminent par "devel". Cela suffit dans la plupart des cas, mais vous verrez parfois qu'il faut encore installer une librairie particulière.

De toute façon, tout ceci est normalement expliqué dans le fichier README livré avec l'archive du programme à compiler. Comme disent les anglophones : RTFM (Read The Fucking Manual !).

Bon, je ne résiste pas à l'envie de placer ici toutes les librairies que j'ai installées chez moi (Mandrake 6.1 avec ajouts perso) :

```
[jice@taz jice]$ rpm -qa |grep -i "-devel"
faces-devel-1.6.1-11
libtermcap-devel-2.0.8-15mdk
libjpeg-devel-6b-10mdk
libpng-devel-1.0.3-4mdk
gdbm-devel-1.8.0-2mdk
xpm-devel-3.4k-6mdk
Mesa-devel-3.0.99-4mdk
zlib-devel-1.1.3-7mdk
newt-devel-0.40-10mdk
readline-devel-4.0-3mdk
rpm-devel-3.0.3-31mdk
libgr-devel-2.0.13-9mdk
glibc-devel-2.1.1-16mdk
libungif-devel-4.1.0-4mdk
libstdc++-devel-1.1.3-3mdk
ClanLib-devel-0.1.15-1
gtk+-devel-1.2.3-15mdk
imlib-devel-1.9.5-6mdk
kmp3-devel-19990829-1mdk
libmikmod-devel-3.1.6-2mdk
libtiff-devel-3.4-8mdk
ncurses-devel-5.0_990703-4mdk
qt-devel-1.44-12mdk
slang-devel-1.3.8-1mdk
```

Kernel-compil

svgalib-devel-1.4.0-10mdk
XFree86-devel-3.3.5-3mdk

Plus...

Glide2x_SDK-2.1-3
Hermes-1.2.5-1
lesstif-0.87.0-1
OpenPTC-x11-1.0.0-2

Cette page est issue de la documentation 'pré-wiki' de Léa a été convertie avec HTML::WikiConverter. Elle fut créée par Jean-Christophe Cardot le 18/01/2000.

Copyright

Copyright © 18/01/2000, Jean-Christophe Cardot



*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/>