

Accro



Inscrit:

27/11/2005

16:24

De Québec  
(Charlesbourg)

Groupe:

Utilisateurs  
enregistrés

Post(s): 379

Niveau : 18

HP : 85 / 428



MP : 126 /

1272

EXP : 14

**Gestion des dépendances - SLAX/AliXe**

#1

Bonjour,

Ce guide explique comment gérer les dépendances sous un système SLAX/AliXe.

Sur les liveCD SLAX/AliXe, il n'y a pas de gestionnaire de paquet. Alors comment on fait pour installer un paquet avec toutes les dépendances requises, lequel travail va nous aider ultérieurement à concevoir des modules compressés .MO utiles à la remasterisation des liveCD SLAX et/ou AliXe.

Gnumeric par l'exemple

Pour réaliser ce travail, j'ai 3 distributions d'installées, lesquelles sont les toutes dernières versions de Slackware, ZenWalk et Vector.

De ZenWalk ou Vector, je regarde celle dont la version de Gnumeric est la plus à jour. Supposant que le résultat donne ZenWalk, alors je boot avec cette distro.

Ensuite je cherche où est le binaire "gnumeric"

```
# whereis gnumeric
```

Le plus souvent l'output est "/usr/bin"

Je me rends donc sous ce répertoire

```
# cd /usr/bin
```

La commande "ldd" va m'aider à identifier les dépendances

```
ldd gnumeric
```

L'output va donner quelque chose comme ça:

```
linux-gate.so.1 => (0xffffe000)
libspreadsheet-1.7.0.so => /usr/lib/libspreadsheet-1.7.0.so (0xb7b54000)
libpopt.so.0 => /usr/lib/libpopt.so.0 (0xb7b4b000)
libm.so.6 => /lib/tls/libm.so.6 (0xb7b28000)
libc.so.6 => /lib/tls/libc.so.6 (0xb79f9000)
libgoffice-0.so.3 => /usr/lib/libgoffice-0.so.3 (0xb7908000)
libgtk-x11-2.0.so.0 => /usr/lib/libgtk-x11-2.0.so.0 (0xb7615000)
libgdk-x11-2.0.so.0 => /usr/lib/libgdk-x11-2.0.so.0 (0xb7595000)
libpango-1.0.so.0 => /usr/lib/libpango-1.0.so.0 (0xb755a000)
libgobject-2.0.so.0 => /usr/lib/libgobject-2.0.so.0 (0xb7520000)
libglib-2.0.so.0 => /usr/lib/libglib-2.0.so.0 (0xb749a000)
libgsf-1.so.114 => /usr/lib/libgsf-1.so.114 (0xb746c000)
libglade-2.0.so.0 => /usr/lib/libglade-2.0.so.0 (0xb7454000)
libgnomeprintui-2-2.so.0 => /usr/lib/libgnomeprintui-2-2.so.0 (0xb741e000)
libgnomeprint-2-2.so.0 => /usr/lib/libgnomeprint-2-2.so.0 (0xb73a6000)
libgnomecanvas-2.so.0 => /usr/lib/libgnomecanvas-2.so.0 (0xb7376000)
libxml2.so.2 => /usr/lib/libxml2.so.2 (0xb7259000)
libz.so.1 => /usr/lib/libz.so.1 (0xb7247000)
libpangoft2-1.0.so.0 => /usr/lib/libpangoft2-1.0.so.0 (0xb7221000)
libatk-1.0.so.0 => /usr/lib/libatk-1.0.so.0 (0xb7207000)
libgdk_pixbuf-2.0.so.0 => /usr/lib/libgdk_pixbuf-2.0.so.0 (0xb71f0000)
libpangocairo-1.0.so.0 => /usr/lib/libpangocairo-1.0.so.0 (0xb71e8000)
libcairo.so.2 => /usr/lib/libcairo.so.2 (0xb71a6000)
libgmodule-2.0.so.0 => /usr/lib/libgmodule-2.0.so.0 (0xb71a2000)
libdl.so.2 => /lib/tls/libdl.so.2 (0xb719e000)
libart_lgpl_2.so.2 => /usr/lib/libart_lgpl_2.so.2 (0xb7188000)
/lib/ld-linux.so.2 (0xb7f2c000)
libpcre.so.0 => /usr/lib/libpcre.so.0 (0xb7169000)
libX11.so.6 => /usr/X11R6/lib/libX11.so.6 (0xb709c000)
libXrender.so.1 => /usr/X11R6/lib/libXrender.so.1 (0xb7093000)
libXrandr.so.2 => /usr/X11R6/lib/libXrandr.so.2 (0xb708f000)
libXi.so.6 => /usr/X11R6/lib/libXi.so.6 (0xb7087000)
libXinerama.so.1 => /usr/X11R6/lib/libXinerama.so.1 (0xb7083000)
libfontconfig.so.1 => /usr/X11R6/lib/libfontconfig.so.1 (0xb705c000)
libXcursor.so.1 => /usr/X11R6/lib/libXcursor.so.1 (0xb7053000)
libXext.so.6 => /usr/X11R6/lib/libXext.so.6 (0xb7045000)
libXfixes.so.3 => /usr/X11R6/lib/libXfixes.so.3 (0xb7040000)
libbz2.so.1 => /lib/libbz2.so.1 (0xb702f000)
libFreeType.so.6 => /usr/lib/libFreeType.so.6 (0xb6fc5000)
libpng12.so.0 => /usr/lib/libpng12.so.0 (0xb6f8c000)
libexpat.so.0 => /usr/lib/libexpat.so.0 (0xb6f6c000)
```

Ce résultat donne toutes les bibliothèques (ainsi que son emplacement) dont dépend l'application gnumeric.

L'étape suivante est de trouver le paquet qui contient chacune des bibliothèques. Pour ce faire j'utilise mc (Midnight Commander). Sous mc je me positionne au répertoire "/var/log/packages".

J'ouvre le menu "Commande" puis "Rechercher de Fichier". La boîte de dialogue se présente ainsi:

```
Recherche de fichiers
Commencer à : .
Nom du fichier : *
Contenu : libpopt.so.0
[ ] Respect de la casse
[<Valider>] faites ENTER
```

L'output va vous donner le nom du paquet qui contient cette bibliothèque. Des fois il y a plus d'un paquet, il suffit de noter le nom d'un paquet

plus d'un paquet, il suffit de noter le nom d'un paquet.

#### la commande grep

Au lieu d'utiliser mc, vous pouvez aussi lancer la recherche à partir de la commande grep, comme suit:

```
# grep libpopt.so.0 /var/log/packages/*
```

L'output va en être le suivant:

```
aaa_elflibs-11.0.0-i486-9:usr/lib/libpopt.so.0.0.0
popt-1.7-1386-1:usr/lib/libpopt.so.0.0.0
```

Donc les 2 paquets aaa\_elflibs et popt contiennent la librairie libpopt.so.0 .

ET VOUS FAÎTES ÇA POUR TOUTES LIBRAIRIES LISTÉES PAR LA COMMANDE ldd sauf pour les librairies suivantes:

```
linux-gate.so.1
/lib/ld-linux.so.2
```

Donc le but c'est de noter le nom des paquets requis en plus du paquet Gnumeric.

#### Le répertoire "/var/log/packages"

Sous une Slackware (et ses dérivés), ce répertoire renseigne sur les paquets TGZ installés.

Donc quand vous aurez listé tous les paquets nécessaires au bon fonctionnement de Gnumeric, allez consulter le rép. "/var/log/packages" sous le liveCD SLAX, pour identifier ce qui est déjà présent et ce qui vous manque par rapport à Gnumeric.

Et vous allez répéter ce processus pour tous les paquets/modules à être créés. UN TRAVAIL DE MOINE...

#### Le mode de compression TGZ

C'est le mode de compression choisi par la Slackware, ZenWalk et autres dérivés Slackware. C'est tout simplement un paquet compressé avec TAR puis recompressé avec GZIP.

#### Le mode de compression TLZ

C'est le nouveau mode de compression choisi par Vector. Ça utilise le nouveau mode de compression LZMA (<http://tukaani.org/>)

Donc un paquet TLZ c'est un paquet compressé avec TAR puis recompressé avec LZMA.

Les 2 outils suivants provenant du site <http://tukaani.org> vont vous permettre de travailler avec des paquets TLZ de Vector:

```
lzma
pkgtools-tukaani
```

Le "pkgtools-tukaani" va faire en sorte que les commandes installpkg, removepkg, upgradepkg et explodepkg vont maintenant travailler avec les TLZ de Vector en plus de travailler avec les TGZ.

#### Les fichiers "slack-desc" et "slack-required"

Le 1er est normalement présent dans le fichier TGZ/TLZ. Il donne une description du paquet et des fois il renseigne sur les dépendances requises (mais ça n'est pas son rôle). C'est le paquet "slack-required" qui renseigne sur les dépendances requises mais il n'est pas toujours présent dans le paquet TGZ/TLZ.

#### Autres sources d'info pour les dépendances

Le site du concepteur du paquet Gnumeric donne des infos sur les dépendances requises, alors il faudrait prendre l'habitude de consulter les sites des concepteurs.

#### Les dépôts de paquets TGZ/TLZ

Toutes distributions Slackware ou dérivées a un dépôt de paquets. Consultez-les. En

plus de ça, il y a les dépôts suivants:

<http://slacky.it> (section download)

<http://linuxpackages.net>

#### En conclusion

Bon maintenant avec ce guide vous savez ce qui précède la création de modules. 🙌

a+, alisou, 😊

N.B. Je vais verrouiller ce post qui est un tutoriel. Si vous avez des questions à ce propos, ouvrez une nouvelle discussion, 😊

Contribution le : Aujourd'hui 07:55:4