

# Sources de paquets logiciels

## 1. Principe des paquetages

La gestion des logiciels sous Ubuntu est centralisée sur des serveurs de dépôts (voir le chapitre Préalable à l'installation) et se gère par un gestionnaire, appelé gestionnaire de paquets. Comme toujours sous Linux, il n'y a pas une mais plusieurs manières de construire, obtenir et gérer ces paquets logiciels.

Ubuntu utilise la philosophie Debian en matière de paquets. Aussi un logiciel aura pour extension `.deb` (Rappel : ne pas utiliser de logiciels Debian pour une distribution Ubuntu, l'extension est la même mais pas son contenu). Son nom renseigne sans ambiguïté sur son numéro de version, sa distribution (parfois) et sa catégorie (processeur i386).

Exemple :

```
dhcp3-common_3.1.3-2ubuntu3_i386.deb
```

### a. Manuellement avec le fichier `sources.list`

#### Dépôts indispensables

Bien régler son fichier de sources de logiciels fait partie des tous premiers réglages de l'administrateur. Deux adresses sont couramment utilisées en France :

- `fr.archive.ubuntu.com` (serveur Oléane).
- `security.ubuntu.com` (serveur canonical à Auckland).

Comme indiqué dans le chapitre Préalable à l'installation, il est parfois utile de changer de sources car il arrive que le serveur Oléane soit surchargé. Quoi qu'il en soit, il est très important de disposer d'un fichier de sources bien réglé : en voici un modèle minimal :

```
# Sources Lucid Lynx 10.04

# Sections principale et à diffusion restreinte
deb http://fr.archive.ubuntu.com/ubuntu/ lucid main restricted
deb http://fr.archive.ubuntu.com/ubuntu/ lucid-updates main restricted
deb http://security.ubuntu.com/ubuntu/ lucid-security main restricted

# Sections principale et à diffusion restreinte
deb http://fr.archive.ubuntu.com/ubuntu/ lucid main restricted
deb http://fr.archive.ubuntu.com/ubuntu/ lucid main restricted
deb http://fr.archive.ubuntu.com/ubuntu/ lucid main restricted
```

Le fichier prend en compte les quatre sections sur deux paragraphes et pour chaque type de sources : **normales, nouvelles, mises à jour de sécurité**. Volontairement compact, certains administrateurs lui préfèrent une version plus détaillée pour chaque section : c'est une affaire de goût !

Voici maintenant les lignes à rajouter (avec les droits administrateurs bien sûr) en cas de besoin des sources de logiciels (on peut les intégrer au fichier et les laisser commentées si on ne les utilise pas) :

```
# Sections principale et à diffusion restreinte
# A activer en cas de besoin en décommentant la ligne
#deb-src http://fr.archive.ubuntu.com/ubuntu/ lucid main
restricted universe multiverse
#deb-src http://fr.archive.ubuntu.com/ubuntu/ lucid-updates main
restricted universe multiverse
#deb-src http://security.ubuntu.com/ubuntu/ lucid-security main
restricted universe multiverse
```

La différence réside dans la dénomination du paquet `deb-src` au lieu de `deb`, avec cette fois-ci la démonstration des quatre sections en même temps. Les sources permettent la compilation d'un logiciel dans un but d'adaptation à la plate-forme (voir le chapitre Maintenance avancée du système).

## Dépôts secondaires

À côté des dépôts indispensables, se trouvent les dépôts des logiciels **rétro-portés** (*backports*) c'est-à-dire des logiciels de la prochaine version Ubuntu adaptés à la version courante :

```
# Logiciels backports
# A activer en cas de besoin en décommentant la ligne
deb http://fr.archive.ubuntu.com/ubuntu/ lucid-backports main
restricted universe multiverse
#deb-src http://fr.archive.ubuntu.com/ubuntu/ lucid-backports main
restricted universe multiverse
```

Il existe aussi les logiciels **commerciaux** payants en partenariat avec la société Canonical :

```
# Logiciels commerciaux
# A activer en cas de besoin en décommentant la ligne
#deb http://archive.canonical.com/ubuntu lucid partner
#deb-src http://archive.ubuntu.com/ubuntu lucid partner
```

Autre possibilité : les dépôts "Proposed" apportant des **logiciels à tester** ou des **mise à jour non encore intégrées** dans les dépôts principaux. On peut comparer ceux-ci à la déclinaison **Sid** de la distribution Debian :

```
# Logiciels proposed
# A activer en cas de besoin en décommentant la ligne
#deb http://fr.archive.ubuntu.com/ubuntu/ lucid-proposed main
restricted universe multiverse
#deb-src http://fr.archive.ubuntu.com/ubuntu/ lucid-proposed main
restricted universe multiverse
```

Tout ces dépôts, outre le fait qu'il n'y a pas à proprement parlé de mises à jour, sont sous la responsabilité de l'administrateur car ils peuvent induire des problèmes sur le système (surtout le premier !). Le risque est important pour un serveur, moindre sur un poste de travail et peut contribuer à résoudre des situations de pilotes non encore adaptés à un matériel récent.

## Dépôts accessoires

Enfin, on trouve des dépôts spécifiques généralement liés à une application (codecs multimédia par exemple) ou pour des raisons légales (dépôt **Médibuntu**). Ces dépôts se structurent de la même façon que les autres. Exemple de **DRBL** (*Diskless Remote Boot in Linux*) pour la fourniture d'un environnement sans disque :

```
# Sources DRBL
deb http://free.nhcc.org.tw/ubuntu lucid main restricted universe multiverse
deb http://free.nhcc.org.tw/drbl-core drbl stable
```

Vous devrez alors importer une clé **GPG** (*GNU Privacy Guard*) chargée de vérifier l'authenticité des transactions car ces logiciels ne viennent pas des dépôts officiels. Deux étapes : la récupération sur le serveur par l'outil `wget` et l'installation par l'utilitaire `apt-key`. Ce qui donne toujours pour l'exemple DRBL :

```
sudo wget http://drbl.nhcc.org.tw/GPG-KEY-DRBL
```

```
sudo apt-key add GPG-KEY-DRBL
```

La première commande récupère la clé, tandis que la deuxième ajoute à la liste la clé déclarée fiable.

Autre exemple : le dépôt Médibuntu qui propose des logiciels ne pouvant être inclus dans la distribution (droits des licences, disparités légales entre pays, etc.) :

```
sudo wget http://fr.packages.medibuntu.org/medibuntu-key.gpg
```

```
sudo apt-key add medibuntu-key.gpg
```

Et l'ajout du dépôt dans le fichier `sources.list` :

```
# Sources Medibuntu
deb http://fr.packages.medibuntu.org/ lucid free non-free
#deb-src http://fr.packages.medibuntu.org/ lucid free non-free
```

Dès lors, une application comme le célèbre Skype (logiciel propriétaire de voix IP) est possible après une mise à jour nécessaire.



Une application graphique gère plus simplement le chiffrement et les "trousseaux" de clefs : `seahorse-preferences`. Elle sera vue dans le chapitre Session de travail en mode graphique.

## b. Sources et interface graphique

Une application graphique (`update-manager`) règle précisément toutes ces options par le menu **Système - Administration - Sources de logiciels**. L'avantage de l'interface graphique est qu'elle permet de choisir facilement parmi une liste un autre serveur que le serveur national. Les onglets présentés dans l'application sont au nombre de cinq :

- **Logiciels Ubuntu** : donne le choix des quatre sections (`main`, `universe`, `restricted`, `multiverse`), des sources et du serveur de téléchargement. Une fonction permet de rechercher le meilleur serveur mais, après plusieurs tests, le doute est permis... (ce qui est vrai à un instant T peut ne plus l'être à T+1). Préférez un serveur en relation avec votre fournisseur d'accès (votre FAI est Free : choisissez `ftp.free.fr`) ou remettez-vous en au serveur national (`fr.archive.ubuntu.com`).
- **Logiciels de tierces parties** : indique l'adresse de serveurs extérieurs pour des logiciels spécifiques. L'adresse d'un serveur miroir local s'écrit aussi ici avec le même format de ligne vu dans le fichier `sources.list` (exemple : `deb http://192.168.3.3/lucid lucid main restricted universe multiverse`).
- **Mises à jour** : concerne les quatre niveaux de mises à jour (`security`, `updates`, `proposed`, `backports`).
- **Authentification** : contient les clés de signatures fiables, soit celle du CD-Rom d'installation et du ou des serveurs utilisés.
- **Statistiques** : laisse le choix de participer à l'étude statistique d'utilisation des paquets : à ne pas utiliser si vous êtes peu concerné par le projet Ubuntu et un peu paranoïaque...

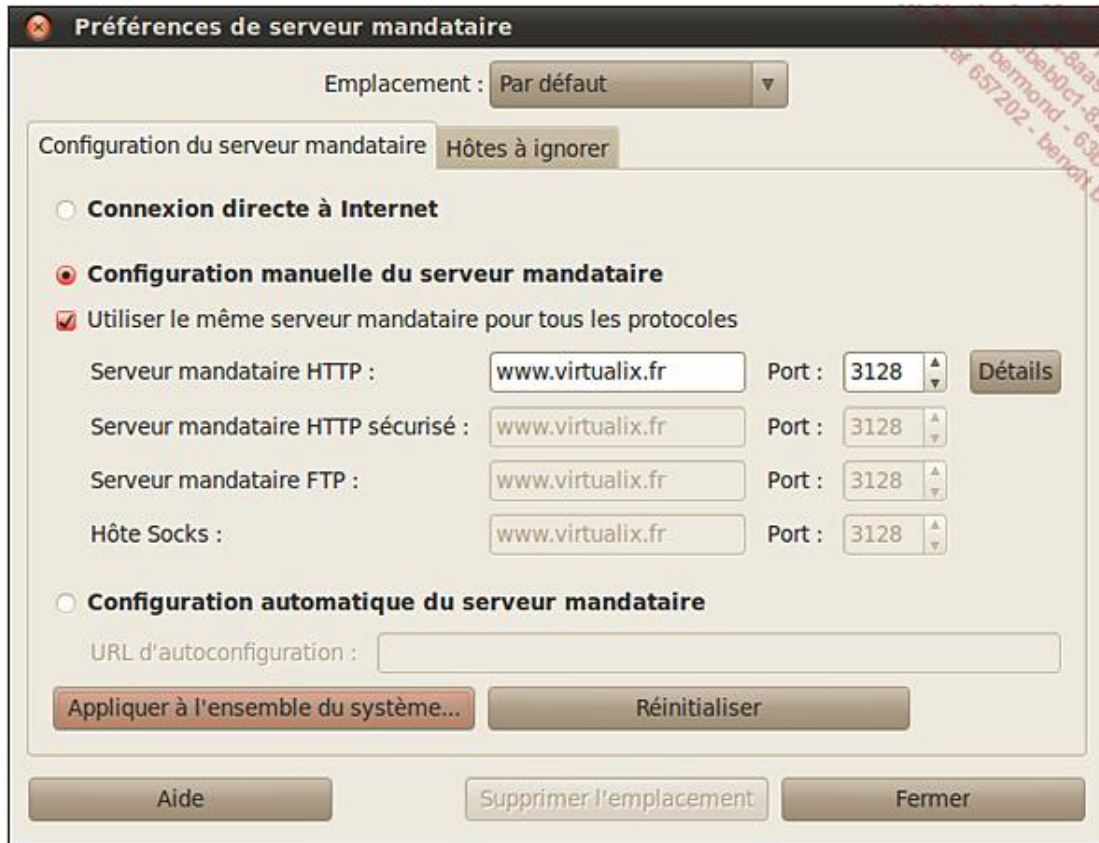


## c. Cas d'un serveur mandataire

S'il existe sur le réseau un serveur mandataire (`proxy`), un réglage supplémentaire est à mettre dans un fichier (à créer) nommé `/etc/apt/apt.conf`. En voici un exemple où le serveur proxy se nomme `www.virtualix.fr`, port 3128 (pas de demande d'authentification utilisateur) :

Acquire::http::Proxy "http://www.virtualix.fr:3128";

Cette démarche s'applique sur un serveur. Dans le cas d'une Ubuntu en mode graphique (et seulement dans ce cas), la gestion du serveur mandataire passe par le menu **Système - Préférences - Serveur mandataire** :



Cette configuration ne fonctionne plus si vous sortez de l'environnement graphique, auquel cas il faut écrire le fichier `/etc/apt/apt.conf`. L'environnement pris en compte est celui de GNOME et sa suite **Evolution**. Attention : un navigateur comme **Firefox** se paramètre individuellement et indépendamment dans l'application par le menu **Edition - Préférences**, onglets **Avancé/Réseau**, bouton **Paramètres**.

## 2. Interfaces de gestion de paquets

L'administrateur ne choisit pas une seule interface de gestion de paquets mais plusieurs en fonction de son besoin. Classée de la plus spartiate à la plus conviviale, chaque interface comporte des avantages. Il n'est pas question ici de fournir un mode opératoire complet de chaque interface mais d'en montrer tactiquement les principales utilisations. L'approche dans cet ouvrage se veut plus didactique que théorique, aussi vous verrez que :

- la commande détermine le choix de l'interface la mieux adaptée.
- seules les meilleures interfaces sont utilisées.

Suivant ce principe, l'utilitaire `dselect` "front-end" trop orienté Debian ne sera pas évoqué ou des bizarreries comme **Gdebi**, interface graphique de **dpkg** qu'il vaut mieux oublier...



Ces commandes, essentiellement d'ordre administratif pour un serveur, se tapent en mode console sous le compte du `root` ou, à défaut, en précédant la commande par le mot-clé `sudo`. La gestion graphique, réservée à un poste de travail, utilise l'outil **Synaptic**.

### a. Utilitaire **dpkg**

Premier utilitaire dit de bas niveau de gestion de paquets **dpkg** (*debian package*) s'utilise pour installer un paquet le plus souvent "solitaire" c'est-à-dire sans dépendances à gérer. Auquel cas une interface plus évoluée est nécessaire. Ensuite, l'utilité de **dpkg** tourne autour de sa capacité à donner des informations.

## Commandes usuelles

Rechercher un ou plusieurs paquets :

```
dpkg -l debconf
```

Variante avec un motif (tous les paquets commençant par deb).

```
dpkg -l deb*
```

Une description de tous les paquets se trouve dans `/var/lib/dpkg/available`.

Installer un paquet (exemple avec le logiciel de virtualisation **VirtualBox** dans sa version non OSE c'est-à-dire non OpenSource) :

```
dpkg -i virtualbox_3.2-3.2.6-63112_Ubuntu_lucid_i386.deb
```

Les options `-r` et `-P` suppriment et purgent respectivement le paquet avec la commande `dpkg`.

Connaître l'appartenance d'un fichier à un paquetage installé :

```
dpkg -S /etc/apache2/apache2.conf
```

Cette commande retourne `apache2.2-common`, paquet du serveur Web Apache.

Connaître la liste complète des paquetages installés :

```
dpkg --get-selections
```

Si l'on effectue une redirection de la commande dans un fichier, cette commande réinstalle une distribution avec pour copie conforme l'ensemble des éléments :

```
dpkg --get-selections > liste_paquetages.txt
```

La reprise :

```
dpkg --get-selections < liste_paquetages.txt
```

```
apt-get dselect-upgrade
```

## b. Utilitaire Apt

Utilitaire phare et incontournable de la gestion de paquets même si l'équipe Debian demande à utiliser **Aptitude** plutôt que **Apt** (*Advanced Packaging Tool*) pour la raison d'une meilleure gestion des dépendances. Mais les habitudes sont là et beaucoup d'administrateurs soutiennent encore l'utilisation d'Apt et de sa commande principale : `apt-get`.

Comme je soutiens l'utilisation d'Aptitude, certaines commandes ne seront pas volontairement présentées (exemple : installation de paquets). Attention : le choix de l'outil importe peu mais il est impératif, sous peine de problème, dans la base des paquets, **de ne pas mélanger les deux méthodes**. Autrement dit si vous utilisez `apt-get` : gardez cette méthode !

Recherche d'un paquetage ou d'un fichier appartenant à un paquetage non installé :

```
apt-cache search dos2unix
```

La recherche par `apt-cache` donne de meilleurs résultats que par Aptitude. L'exemple de `dos2unix` le montre. Cet utilitaire utilisé pour adapter les fichiers texte du format Windows au format Unix (principalement pour les caractères de fin de ligne comme les retours chariots) appartient au paquetage `tofrodo`. La commande `apt-cache` retourne deux paquets alors qu'Aptitude ne trouve rien !



Le nom passé dans la commande peut être tronqué, dans l'exemple `dos2` retournera la même chose ; cela est utile si l'on ne connaît pas exactement la syntaxe de ce que l'on cherche.

Installation d'une nouvelle clé GPG :

```
apt-key add medibuntu-key.gpg
```

Ajoute la clé à la liste des clés fiables. Le paramètre `list`, seul, affiche la liste du trousseau (fichier `/etc/apt/trusted.gpg`) alors que le paramètre `del` supprime une clé de cette liste.

### c. Utilitaire Aptitude

Promu utilitaire principal de gestion de paquets, **Aptitude** dans sa version en mode commande (l'interface semi-graphique n'apporte pas grand chose de plus) remplace `apt-get` et ne change pas beaucoup les usages. Il apporte par contre une meilleure gestion des dépendances, serpent de mer de toute interface de gestion de paquets.

Installe un paquet logiciel en gérant les dépendances liées :

```
aptitude install paquet
```

À l'inverse, cette option désinstalle un paquet logiciel :

```
aptitude remove paquet
```

Cette variante désinstalle un paquet logiciel et tente de supprimer en plus les fichiers de configuration et les répertoires créés sauf s'ils ne sont pas vides :

```
aptitude purge paquet
```

Cette commande fondamentale met à jour la liste des paquets logiciels disponibles par rapport à la liste contenue dans `/etc/apt/sources.list` :

```
aptitude update
```

Mettre à jour le système de façon très propre avec uniquement les paquets logiciels installés ; de plus, seuls les paquets non utilisés seront désinstallés :

```
aptitude safe-upgrade
```

Cette autre option met à jour le système en enlevant les anciennes versions des paquets installés à la différence de la commande précédente. Réputée plus "abrasive" pour le système, elle est quand même à employer par rapport à `safe-upgrade` car elle rend le système plus cohérent :

```
aptitude full-upgrade
```

Donner des informations sur un paquet :

```
aptitude show paquet
```

```
root@ubuntu:~# aptitude show squid
Package: squid
State: not installed
Version: 2.6.18-1ubuntu3
Priority: optionnel
Section: web
Maintainer: Ubuntu Core Developers <ubuntu-devel-discuss@lists.ubuntu.com>
Uncompressed Size: 1622k
Depends: adduser, libc6 (>= 2.4), libdb4.6, libldap-2.4-2 (>= 2.4.7), libpam0g (>= 0.99.9.8), logrotate (>= 3.5.4-1), lsb-base, netbase, squid-common (>= 2.6.18-1ubuntu3)
PreDepends: debconf (>= 1.2.9) | debconf-2.0
Suggests: logcheck-database, resolvconf (>= 0.40), smbclient, squid-cgi, squidclient,
Conflicts: sarg (< 1.1.1-2), squid-novm
Replaces: squid-novm
Description: Internet object cache (WWW proxy cache)
 This package provides the Squid Internet Object Cache developed by the National Labo
 Applied Networking Research (NLNR) and Internet volunteers.
Homepage: http://www.squid-cache.org/
```



L'exemple montre des renseignements sur le paquet **Squid** (logiciel pour la création d'un serveur mandataire). Elle affiche aussi d'autres renseignements intéressants : l'état, les dépendances, les paquets suggérés et les conflits. Enfin, elle indique éventuellement la page Web de l'application.

Cette option, moins connue, télécharge simplement le paquet logiciel dans le répertoire courant, cela est utile en cas de traitement non immédiat :

```
aptitude download paquet
```

Supprimer le cache pour libérer de la place sur le disque dur :

```
aptitude clean
```

L'exécution de cette dernière option (voir le manuel en ligne pour toutes les options possibles) vide le répertoire `/var/cache/apt/archives`, c'est-à-dire là où les paquets logiciels ont été chargés.

### **Résumé de séquence de mise à jour d'un système**

Pour mettre à jour un système Ubuntu, voici la séquence des trois commandes à exécuter les unes à la suite des autres :

```
aptitude update
```

```
aptitude full-upgrade# ou safe-upgrade
```

```
aptitude clean
```



Aptitude reste par excellence l'outil de l'administrateur, aussi la pratique courante consiste à s'arrêter là.

---

## **d. Utilitaire Wajig**

Parallèlement à l'utilisation d'Aptitude, existe un autre utilitaire que l'on pourrait qualifier "d'exotique" s'il ne disposait pas de qualités certaines. Basé sur le langage Python, l'installation a comme dépendance le paquetage `dselect` et beaucoup de paquets suggérés qu'il est nécessaire de prendre (exemples : `wget`, `fping`) :

```
aptitude install wajig wget fping
```

Wajig comme Aptitude effectue la mise à jour des sources :

```
wajig update
```

Il donne un renseignement double : le nombre de logiciels à mettre à jour et les nouveaux. On voit les nouveautés par un simple :

```
wajig new
```

```

root@jeos:~# wajig new
Package      Description
=====
fglrx-amdcccle-envy  Dummy package for easy transition
fglrx-control-envy   Control panel for the ati graphics accelerators
fglrx-kernel-source-envy  Ati binary kernel module source
nvidia-glx-dev-envy  Nvidia binary xfree86 4.x/x.org driver development files
nvidia-glx-envy      Nvidia binary xfree86 4.x/x.org driver
nvidia-glx-legacy-dev-envy  Nvidia binary xfree86 4.x/x.org 'legacy' driver development files
nvidia-glx-legacy-envy  Nvidia binary xfree86 4.x/x.org 'legacy' driver
nvidia-glx-new-dev-envy  Nvidia binary xfree86 4.x/x.org 'new' driver development files
nvidia-glx-new-envy    Nvidia binary xfree86 4.x/x.org 'new' driver
nvidia-kernel-source-envy  Nvidia binary kernel module source
nvidia-legacy-kernel-source-envy  Nvidia binary 'legacy' kernel module source
nvidia-new-kernel-source-envy  Nvidia binary 'new' kernel module source
openssh-blacklist    List of blacklisted openssh rsa and dsa keys
openssl-blacklist     List of blacklisted openssl rsa keys
openvpn-blacklist     List of blacklisted openvpn rsa shared keys
xorg-driver-fglrx-dev-envy  Video driver for ati graphics accelerators (devel files)
xorg-driver-fglrx-envy   Video driver for ati graphics accelerators
root@jeos:~# _

```

Les paquetages à mettre à jour sont montrés avec les numéros de version : ce qui est intéressant pour l'administrateur :

wajig newupgrade

```

root@jeos:~# wajig newupgrade
Package      Available      Installed
=====
libssl0.9.8  0.9.8g-4ubuntu3.1  0.9.8g-4ubuntu3
openssh-client  1:4.7p1-8ubuntu1.2  1:4.7p1-8ubuntu1
openssh-server  1:4.7p1-8ubuntu1.2  1:4.7p1-8ubuntu1
root@jeos:~# _

```

Wajig utilise bien sûr les commandes `install`, `remove`, `purge` et `upgrade` pour la gestion des paquets. La liste complète des commandes se trouve par un simple `wajig list-commands` ou à cette adresse Internet : <http://www.togaware.com/wajig>

## e. Gestion graphique

Dernier utilitaire en situation graphique : **Synaptic**. Il se lance par le menu **Système - Administration - Gestionnaire de paquets Synaptic**. Même dans le cas d'un poste de travail, l'utilisation de Synaptic est à déconseiller car peu intuitif. On perd beaucoup moins de temps à gérer les paquets avec **Aptitude** en ligne de commande.

### Mise à jour du système

Les mises à jour de sécurité s'installent :

- sans confirmation
- en téléchargement en arrière-plan
- avec un simple avertissement

Le choix de ces trois options se trouvent dans le menu **Système - Administration - Sources de logiciels**, onglet **Mises à jour**.

Pour le reste, l'application `update-manager` (menu **Gestionnaire de mises à jour**) détecte par défaut automatiquement les mises à jour et les installe après fourniture du mot de passe de l'utilisateur privilégié.