

# Les fichiers apt\_preferences

- Objet : Modifier la priorité d'installation des paquets
- Niveau requis :  
[avisé](#)

(ou fou furieux)

- Commentaires : *Modifier la priorité d'installation des paquets lorsque différentes versions sont proposées par différents dépôts.*
- **Avertissement** : Attention, ce tuto, c'est le manuel de l'apprenti mauvais-administrateur-système, à utiliser avec modération...
- Suivi :  
[à-compléter](#)
  - Création par [captfnfab](#) 23/05/2013
  - Testé par [lr0nsh007er](#) le 19/07/2015
  - Mise à jour par [smolski](#) le 09-08-2015
- Commentaires sur le forum : [Lien vers le forum concernant ce tuto](#) <sup>1)</sup>

## Introduction

Lorsque l'on dispose de plusieurs versions installables dans les dépôts renseignés dans les fichiers [sources.list](#), il faut définir des priorités pour que APT sache quelle version installer.

Par exemple, si l'on a les dépôts Debian de testing et unstable et que l'on veut rester autant que possible en testing, il faut indiquer à APT que *testing* a une priorité supérieure à *sid*. Inversement, si pour un paquet donné on veut utiliser sa version présente dans *sid*, il faut le spécifier également. La définition de ces priorités s'appelle le **pinning**.

C'est à cela que servent les fichiers **preferences**.

Les cas d'utilisation raisonnable sont les suivants :

1. Vous voulez être en *testing* avec les dépôts *sid* sous la main au cas où ;
2. Vous voulez être en *testing* ou *sid* mais récupérer la version *sid* ou *experimental* d'un paquet en particulier ;
3. Vous utilisez le dépôt multimedia, mais ne voulez pas donner une forte priorité à tous les paquets qu'il propose.

En dehors de ça, si vous voulez mélanger stable et une *testing/unstable/experimental*, alors vous devriez prendre rendez-vous d'urgence chez votre garagiste pour qu'il s'occupe de votre carie.

## Les priorités

Les fichiers preferences permettent de fixer la priorité des paquets suivant leurs dépôt. Voici la signification d'une priorité P.

- $P \geq 1000$  : le paquet sera installé même si APT est forcé d'installer une version plus ancienne<sup>2)</sup>

que la version actuelle. Cette opération n'est pas sans risque.

- $990 \leq P < 1000$  : le paquet sera installé sauf si une version plus récente est déjà installée.
- $500 \leq P < 990$  : le paquet ne sera installé qu'à défaut de version plus récente installée ou disponible dans la distribution cible.
- $100 \leq P < 500$  : le paquet ne sera installé que s'il n'en existe aucune version plus récente, installée ou disponible dans une autre branche.
- $0 < P < 100$  : le paquet ne sera installé que si aucune autre version du paquet n'est installée.
- $P < 0$  : le paquet ne sera jamais installé.

## Configuration initiale

Cette configuration est très bien, franchement, je ne vois pas pourquoi vous voulez la changer 😊

- Par défaut, la priorité de tous les dépôts est de 500, à l'exception du dépôt `backports`, de priorité 100, et du dépôt `experimental` de priorité 1.
- Lors d'une installation, les dépôts correspondant à la branche courante prennent une priorité 990.
- Lors d'une installation avec l'option « `-t votrebranche` », les dépôts correspondant à la branche « `votrebranche` » prennent une priorité 990.
- Lors d'une installation en précisant la branche d'un paquet avec « `/votrebranche` », le paquet concerné prend une priorité 990.

### exemple :

La commande :

```
apt-get -t stretch-backports install libreoffice
```

donne une priorité élevée à **toute la branche stretch-backports** le temps de l'instance d'APT.

L'option `-t` (t pour target) indique le dépôt cible où l'on va chercher le paquet à installer en gérant correctement les dépendances par rapport à ce dépôt et aux dépôts stables.

Merci à chalu de cette précision-ci sur le forum à ce post là :

- <https://debian-facile.org/viewtopic.php?pid=207154#p207154>

La commande :

```
apt-get install libreoffice/stretch-backports
```

pose des problèmes de dépendances parce que la version prioritaire du paquet `libreoffice-common` restera celle des sources habituelles et non pas celles des backports.

Lien sur le forum :

[différence install -t backports & paquet/backports](#)

On peut vérifier les priorités en utilisant la ligne de commande, par exemple, si vous êtes en *stable* :

```
apt-cache policy
```

## Synaptic

Le pinning fonctionne si tu utilises soit [apt](#), [apt-get](#) ou [aptitude](#), mais si tu utilises [synaptic](#), le fichier créé en faisant du pinning va entrer en conflit avec le fichier de configuration de synaptic !

Configuration de synaptic :

1. Tu vas dans :  
configuration → préférences onglet Distribution
2. Tu coches Préférer les version de et tu choisies la version que tu désires.

Normalement tu ne devrais plus y être submergé par des demande de mise à jours.

Merci à **valmy** et **Severian** qui ont initié cette recommandation sur le forum ! 😊

## Précautions

### ATTENTION !

En faisant joujou avec les fichiers preferences, on peut très rapidement faire quelque chose qu'on ne voulait pas, et qui nous oblige à **réinstaller** le système.

Pour éviter cela, il existe une série de tests que vous pouvez faire pour tester votre configuration, et de précautions à prendre.

## Vérifier la configuration

Après avoir **créé ou modifié** votre fichier preferences, la première chose à faire est dans un terminal en root :

```
apt-get update
```

Puis vérifiez que vos modifications ont bien été prises en compte grâce à `apt-cache policy`.

Si les résultats affichés ne vous conviennent pas, vous risquez d'avoir une mauvaise surprise après une mise à jour...

## Les exemples d'utilisation

### Rester en Stable

Ce fichier donne une priorité de 900 à tous les paquets de la release *stable* et une priorité de -10 aux autres (donc ils ne sont pas installés).

Il est parfaitement inutile si vous n'avez que les dépôts *stable* dans votre `sources.list`, il permet en revanche d'éviter à tout prix l'installation de paquets *testing* ou *unstable* si vous êtes en *stable* mais avez (pour une raison que je ne veux pas connaître) tout de même envie de mettre ces dépôts dans votre `sources.list`.

On crée un fichier `/etc/apt/preferences.d/00stable-seulement` contenant le code suivant :

### 00stable-seulement

```
Package: *
Pin: release a=stable
Pin-Priority: 900

Package: *
Pin: release a=stable-updates
Pin-Priority: 900

Package: *
Pin: release a=proposed-updates
Pin-Priority: 900

Package: *
Pin: release o=Debian
Pin-Priority: -10
```

## Stable avec suivi d'un paquet dans les Backports

Par exemple, pour installer la version d'*iceweasel* des backports et la maintenir à jour.

On crée un fichier `/etc/apt/preferences.d/90suivi-backports` contenant le code suivant :

### 90suivi-backports

```
Package: iceweasel
Pin: release a=stretch-backports
Pin-Priority: 900
```

Par défaut le dépôt `stable-backports` a une priorité de 100. Il est alors inutile de préciser la priorité des autres paquets de `stable-backports`

## Stable avec dépôt multimédia à faible priorité

Pour garder le dépôt `multimedia` à disposition, sans faire les mises à jour vers ce dépôt systématiquement.

On crée un fichier `/etc/apt/preferences.d/50multimedia` contenant le code suivant :

## 50multimedia

```
Package: *
Pin: release o=Unofficial Multimedia Packages,a=stable
Pin-Priority: 100
```

### Testing avec Sid non-prioritaire

On suppose que vous avez comme sources quelque chose comme ça :

```
deb http://deb.debian.org/debian testing main contrib non-free
deb http://deb.debian.org/debian sid main contrib non-free
```

mais que vous voulez rester en *testing* autant que possible.

On crée un fichier `/etc/apt/preferences.d/40sid-et-testing` contenant le code suivant :

## 40sid-et-testing

```
Package: *
Pin: release n=sid
Pin-Priority: 100
```

### Testing avec suivi d'un paquet dans Sid

Par exemple, je suis en Testing mais veut installer la version du paquet `firefox` du dépôt de Sid tout en restant à jour.

On crée un fichier `/etc/apt/preferences.d/40firefox-sid` contenant le code suivant :

## 90firefox-sid

```
Package: *
Pin: release n=sid
Pin-priority:100

Package: firefox
Pin: release n=sid
Pin-Priority: 900
```

Cette méthode n'est pas conseillé sur Stable. Dans ce cas, il est préférable de construire le paquet depuis les sources du paquet dans Sid en suivant [le wiki rétroportage](http://wiki.debian.org/Backporting)

## Sid avec suivi d'un paquet dans Experimental

Par exemple, je suis en sid mais veut installer la version experimental de iceweasel tout en restant à jour.

On crée un fichier `/etc/apt/preferences.d/40suivi-experimental` contenant le code suivant :

### 40suivi-experimental

```
Package: iceweasel
Pin: release a=experimental
Pin-Priority: 900
```

Par défaut le dépôt expérimental a une priorité de 1. Il est alors inutile de préciser la priorité des autres paquets

## Sid avec les dépôts multimedia de experimental

La priorité du dépôt deb-multimedia pour experimental est à 500 par défaut. Si vous avez ce dépôt dans votre fichier sources.list, faite donc attention à votre fichier preferences si vous ne désirez pas que tous les paquets issus de deb-multimedia passent en experimental.

On crée un fichier `/etc/apt/preferences.d/40multimedia-experimental` contenant le code suivant :

### 40multimedia-experimental

```
Package: *
Pin: release o=Unofficial Multimedia
Packages,a=experimental,l=Unofficial Multimedia Packages
Pin-Priority: 1
```

## Les paquets particuliers

La forme particulière affecte une priorité (Pin-Priority) à un paquet précis, à une version précise ou à un intervalle spécifiant plusieurs versions.

Par exemple, l'entrée suivante affecte une priorité haute à toutes les versions du paquet perl dont le numéro de version commence par **5.8** :

```
Package: perl
Pin: version 5.8*
Pin-Priority: 1001
```

Merci à **caly** sur le chan d'avoir suscité cet ajout. 😊

## Conseils et remarques

Le fait d'avoir des priorités qui ne sont pas égales pour toutes les différentes branches Debian a pour inconvénient que les mises à jours de sécurité et jessie-updates des paquets communs aux branches Unstable et Stable sont moins réactives, qu'elles prennent plus de temps à arriver.

**Nommer les branches par leur nom **\*\*commun\*\*** ou leur nom **\*\*release\*\*****

```
deb http://deb.debian.org/debian/ stretch main contrib non-free
```

apt-cache policy donne une option ( $n=stretch$ )

```
900 http://deb.debian.org/debian/ stretch/main Packages
    release v=6.0.2.1,o=Debian,a=stable,n=stretch,l=Debian,c=main
    origin deb.debian.org
```

donc on peut rajouter dans le fichier **preferences** ce style d'interprétation :

```
Package: *
Pin: release a=stable
Pin-priority: 900
```

```
Package: *
Pin: release n=stretch
Pin-priority: 900
```

**Utiliser **\*\*origin\*\*** au lieu de **\*\*release\*\*****

Une autre façon d'avoir le dépôt multimédia à faible priorité

Pour garder le dépôt multimédia à disposition, sans faire les mises à jour vers ce dépôt systématiquement.

```
deb http://www.deb-multimedia.org jessie main non-free
```

```
Package: *
Pin: origin "www.deb-multimedia.org"
Pin-Priority: 100
```

La dernière version de iceweasel avec buster (testing)

```
deb http://cdn.debian.net/debian unstable main
deb http://cdn.debian.net/debian experimental main
```

```
Package: iceweasel iceweasel-l10n-fr
Pin: origin "cdn.debian.net"
Pin-Priority: 500
```

```
Package: *  
Pin: release a=experimental,n=experimental  
Pin-Priority: 1
```

```
Package: *  
Pin: origin "cdn.debian.net"  
Pin-Priority: 100
```

attention à l'ordre des commandes

## Garder des priorités identiques pour les dépôts d'une même branche

C'est le comportement par défaut quand on n'a que les dépôts de la branche suivie, sans fichier **preferences**.

Tout manquement à cette règle casse le comportement par défaut et peut générer des résultats très dommageables car non prévus par les développeurs Debian.

(Excepté les dépôts **backports**)

## Attribuer une priorité comprise entre 500 et 989

pour la branche suivie et la/les branche(s) comportant des paquets aux versions égales ou inférieures à la branche suivie.

Pourquoi une valeur plus petite que 990 ? Parce que lorsque l'on utilise l'option **-t** ainsi :

```
apt-get install -t <branche> <le_nom_du_paquet>
```

pour installer des paquets d'une branche autre que celle suivie, celle-ci devient la branche par défaut (APT::Default-Release "branche") avec une priorité temporaire de **990**.

Donc, pour *la branche prioritaire* du fichier **preferences**

avoir une priorité égale ou supérieure à 990 perturbe l'option "-t"

## D'un point de vue pratique,

Il est préférable d'utiliser des chiffres ronds, comme 900, 800, 90. Par exemple, si l'on a une priorité 620 et une priorité 630, il sera facile d'intercaler une priorité 625.

L'installation d'un paquet d'une branche **supérieure** peut nécessiter :

- La mise à jour de lib récentes **incompatibles** avec d'autres paquets plus anciens, qui devront également être upgradés (mis à jour...) à leur tour !

Bref, installer ou mettre à jour un paquet d'une branche supérieure peut n'être possible qu'en migrant vers la branche supérieure.



## libc6

Si vous êtes sous stable et que vous voulez installer un paquet de la branche testing, ou même unstable, qui impliquerait des mises à jour aussi importantes que **libc6**, alors abandonnez, le pinning ne sera pas une solution pour vous, puisqu'il romprait la stabilité du système et effectuerait une mise à jour partielle vers testing, ce qui est la pire des situations possibles.

## À consulter :

La documentation de référence sur ce fichier de configuration est disponible dans la page de manuel apt\_preferences, accessible par la commande :

```
man apt_preferences
```

1)

N'hésitez pas à y faire part de vos remarques, succès, améliorations ou échecs !

2)

On appelle cela le *downgrade* par opposition à l'*upgrade*.

From:

<http://debian-facile.org/> - **Documentation - Wiki**

Permanent link:

<http://debian-facile.org/doc:systeme:apt:pinning>

Last update: **23/06/2019 14:46**

