

[debian](#), [jessie](#), [stretch](#), [todo](#)



Cet article **n'est pas une procédure de mise à jour** et concerne simplement les opérations qui ont été nécessaires pour retrouver les fonctionnalités de l'ancien système à l'occasion de la mise à jour ainsi que quelques notes sur ce qui a changé.

Une mise à jour sans douleur



Sans douleur mais avec quelques sueurs froides

Pour cette mise à jour, j'ai suivi [ce guide simple et direct](#) et disons qu'en 2 heures, téléchargements inclus, c'était réglé. Donc pour moi, un excellent guide... Par contre il ne donne que ce qu'il annonce, c'est à dire la procédure de mise à jour du système *et pis c'est tout !* Debian étant ce qu'il est, en dehors de quelques choix cornéliens du genre "je garde l'ancien fichier de conf de tel service ou je prends le nouveau ?", c'est vraiment passé comme un lettre à la poste et après le reboot final on obtiens bien un système Debian 9 Stretch (9.3 dans mon cas vu que j'étais bien occupé ailleurs et que

j'ai traîné) 😎

Par contre, contrairement à ce que j'avais naïvement cru et espéré (et c'est bien évidemment volontaire), cette mise à jour ne touche pas les éléments les plus sensibles comme MySQL ou PHP.

Et c'est là que ça deviens plus *touchy*... Mais tout est relatif : après 3heures de plus (donc 5 heure après la première commande), le système était bon à 95% avec MariaDB à la place de MySQL et PHP 7 au lieu de la version 5.6 (le seul problème non résolu à ce moment là était l'*URL rewriting* qui ne fonctionnait plus dans DokuWiki, donc une fonction purement cosmétique).

MySQL

Debian passe du moteur de base de données MySQL à MariaDB. Il y a sans doute plein de raisons pour ce choix et probablement des pelletées d'articles, *posts* et autres 🐹 [trolls](#) qui étudient cela de long en large mais voici [un début d'explications](#) bien suffisant pour moi.

Je n'ai qu'un usage purement personnel et basique de MySQL (principalement une base de données dédiée à [Piwigo](#) et une autre (enfin 2 dans les faits) pour [Kodi](#)) et donc aucune raison de ne pas passer à MariaDB pour rester accroché à MySQL comme un bernacle à son rocher...



Même si les bases de données MySQL sont totalement compatibles avec MariaDB, il n'en est pas moins **primordial** de les sauvegarder avant de passer à la suite.

Pendant la mise à jour vers *Stretch*, j'avais eu quelques alertes concernant *PhpMyAdmin* que je n'ai pas réussi à résoudre avec les options intégrées pour recommencer le processus donc j'ai fini par laisser le problème de côté pour y revenir par la suite et m'apercevoir qu'au final, *PhpMyAdmin* était fonctionnel mais affichait quelques alertes aussi claires et explicites qu'un caillou volcanique. 

Donc j'ai choisit de désinstaller et réinstaller le paquet mais c'est à la fin de cette procédure que j'ai remarqué qu'à chaque utilisation, le commande `apt` indiquait une longue liste de paquets devenus inutiles, dont MySQL...

Et j'ai donc supposé que MariaDB avait déjà pris le relais...
Je me souviens avoir pensé sur le coup que le changement de moteur de base de donnée était

vraiment **incroyablement** simple. 

Tellement **incroyablement** simple que je n'aurais pas dû y croire. 

Ayant une sauvegarde de toutes les bases de la veille et aucune modification par la suite, j'ai désinstallé...

Vous connaissez "Et paf les bases de données !" ? 

Premier réflexe : j'ai cherché s'il est possible de réinstaller MySQL. Comme Internet est un véritable vivier de bernacles qui s'accrochent à leurs rochers, il y a plein de tutoriels pour ça. Mais au final j'ai préféré installer MariaDB et j'ai bien vite retrouvé les bases de données et les utilisateurs hérités de

MySQL, mais seulement en local 

Rectifier les privilèges n'a toutefois pris que quelques minutes...

Apache et PHP

Comme dit plus haut, le passage à Stretch a laissé la version de *PHP* telle quelle.

L'installation en elle-même n'a pas été particulièrement compliquée, par contre je me suis gratté le crâne un moment avant de réaliser qu'il fallait désactiver la version 5.6 et activer la nouvelle :

```
<cli>root@muffin:~# a2dismod php5.6 root@muffin:~# a2enmod php7.0 root@muffin:~# service apache2 restart </cli>
```

J'ai depuis découvert [ici](#) qu'il aurait aussi probablement fallu faire ceci : `<cli>root@muffin:~# update-alternatives -set php /usr/bin/php7.0</cli>`

URL rewriting d'une ferme DokuWiki

C'est la fonctionnalité (évoquée [ici](#) et détaillée [là](#)) qui m'a pris le plus de temps à remettre en route...

J'ai d'abord cherché du côté du module `mod_rewrite` mais c'est en fait dans le fichier de configuration

de chaque hôte virtuel qu'il a fallu copier des éléments qui n'était pas nécessaires avant. J'ai commencé par copier la quasi-totalité d'un fichier de configuration complet puis j'ai affiné ensuite ([voir les détails](#)).

Je suppose simplement qu'Apache est plus sensible qu'il ne l'était...

Ou une ré-installation

48h après le début de la mise à jour vers Stretch, le disque-dur principal du serveur a lâché 🤖 et je suis reparti d'une installation toute fraîche...

From:

<http://wiki.geekitude.fr/> - **Geekitude**

Permanent link:

http://wiki.geekitude.fr/archives/info/os/debian/de_jessie_a_stretch

Last update: **2024/05/17 08:23**

