

# Raspberry Pi OS

## Mettre à jour son système

Tenir son système à jour est très important, notamment en termes de sécurité. Il est important d'effectuer régulièrement les mises à jour, mais comment faut-il faire exactement ? A quoi correspondent les commandes que beaucoup de tutoriels nous font saisir et quelles sont leurs différences. C'est quoi cette histoire de paquets et de dépendances dont me parle tous les sites ?

Nous allons essayer de répondre à toutes ces questions dans ce tutoriel qui sera un peu plus théorique que les deux précédents.

### 1. Introduction à Advanced Packaging Tool (APT)

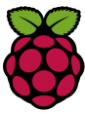
Pour expliquer cette partie, il va falloir entrer dans le vif du sujet.

Pour rappel, le système d'exploitation Raspberry Pi OS (anciennement nommé Raspbian) est une distribution Linux basée sur Debian, comme beaucoup d'autres distributions très connues telles qu'Ubuntu ou Linux Mint par exemple. Debian est souvent citée comme une des plus robustes, et son système de packaging ne déroge pas à la règle. En effet, chaque application et composant est intégré dans un « package » ou « paquet » afin d'être installé sur le système d'exploitation. Debian utilise un ensemble d'outils appelés « Advanced Packaging Tool » ou « APT » pour gérer ce système de packages (à ne pas confondre avec la commande « `apt` »).

#### a. Mais c'est quoi ce « `apt` » que je n'arrête pas de taper ?

Afin de pouvoir interagir avec ce fameux APT, de nombreux outils ont été développés, notamment « `apt-get` » ou encore « `apt-cache` » qui permettent de communiquer avec APT en lignes de commandes. Avec le temps « `apt-get` » et « `apt-cache` » se sont enrichis avec de très nombreuses commandes parfois très complexes et très spécifiques et beaucoup d'entre elles ne sont utilisées que par des utilisateurs « avancés ».

Afin de faciliter la gestion et l'apprentissage de ces commandes et fournir à l'utilisateur moyen des méthodes simples pour gérer les packages, « `apt` » a été créé



## Tutoriel N°3 : Mettre à jour son Raspberry Pi et Raspberry Pi OS

de façon plus structurée. Ainsi l'utilisateur n'a plus à « jongler » entre les commandes « `apt-get` » et les commandes « `apt-cache` ».

### b. La plupart des commandes que je vois sont identiques entre « `apt` » et « `apt-get` », mais lequel dois-je utiliser ?

Répondre à cette question demande de bien saisir la différence entre « `apt` » et « `apt-get` » et le but de ce tutoriel n'est pas de vous noyer sous une tonne d'informations. Disons que pour résumer, « `apt` » se trouve au-dessus d'« `apt-get` » et permet, en une simple commande « `apt` », d'exécuter plusieurs commandes « `apt-get` » ou encore des commandes avec des options particulières définie par défaut, toujours dans le but de faciliter la vie à l'utilisateur.

Afin de ne pas non plus perturber les utilisateurs les plus avertis, les syntaxes données aux requêtes « `apt` » sont volontairement très proches d'« `apt-get` », au point où la plupart des commandes « `apt-get` » que nous pouvons voir dans des tutoriaux sur internet peuvent être, tout simplement, remplacée par « `apt` ».

C'est à vous de choisir celles que vous voulez utiliser, les deux sont amenés à cohabiter et à évoluer ensemble, même si je vous recommande plutôt d'utiliser les commandes « `apt` » si vous êtes débutant.

## 2. Les commandes « `apt` » importantes

Alors, je vous préviens, après avoir lu ce chapitre, vous ne voudrez plus des mises à jour de Windows ! Le système de paquets et de dépendances est, de loin ce qui fait la force de Linux à mon sens.

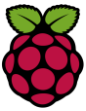
Mettre à jour son système, c'est avant tout mettre à jour des paquets au moyen des dépôts conçus exclusivement dans ce but. Ok, mais c'est quoi un dépôt ?

Pour faire simple, un dépôt est un espace de stockage en ligne permettant de facilement mettre à jour les logiciels qui le composent.

Si vous ouvrez un terminal et que vous saisissez :

```
sudo apt update
```

Cette commande va interroger les différents dépôts auquel vous êtes abonnés pour vérifier s'il existe des mises à jour pour les logiciels installés sur votre système. Attention, cette commande n'installe pas les mises à jour, elle permet seulement de voir le nombre de mises à jour disponible et de les préparer pour une éventuelle installation.



## Tutoriel N°3 : Mettre à jour son Raspberry Pi et Raspberry Pi OS

```
sudo apt list --upgradable
```

Cette commande permet de voir en détails la liste des applications qui peuvent être mises à jour ainsi que le détail des versions de chaque application (oui, tout ça en une commande).

Pour procéder au téléchargement et à l'installation de ces fameuses mises à jour, il suffit de taper la commande :

```
sudo apt upgrade
```

Cette commande va récupérer la même liste que précédemment, va calculer le poids de ces mises à jour et va vous demander si vous souhaitez bien les installer.

Confirmez en tapant « Y ».

Il est également possible de lui dire dès le début que vous êtes d'accord pour les installer en tapant :

```
sudo apt upgrade -y
```

Le « -y » placé à la fin remplace le « Y » que vous auriez dû taper en confirmation.

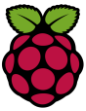
*Attention, il faut, au préalable, saisir la commande « update » pour préparer les fichiers qui seront téléchargés avec la commande « upgrade ». C'est la raison pour laquelle ces deux commandes sont souvent saisies l'une à la suite de l'autre dans les nombreux tutoriaux présents sur internet.*

Avec l'opération que nous venons de faire, nous savons que tous les logiciels installés sur l'OS sont à jour, mais, qu'en est-il de la distribution elle-même ? Et bien pas vraiment... La commande « upgrade » a mis à jour les logiciels installés mais, dans le cas, par exemple, où certains logiciels possèdent des nouvelles dépendances, elles ne seront pas pour autant installées.

La commande que nous allons voir maintenant va au-delà de l'« upgrade » car elle permet justement de gérer les dépendances et si, un logiciel qui doit être mis à jour, a besoin d'une nouvelle dépendance, celle-ci sera automatiquement installée. Mais cette commande va encore au-delà, car, si des paquets ou dépendances ne sont plus nécessaires, ils seront également désinstallés.

Cependant, si vous souhaitez que votre système reste stable, la commande « upgrade » réduit le risque d'instabilité car elle ne met pas à jour le système ou le noyau.

Pour effectuer cette opération, il faut saisir la commande :



## Tutoriel N°3 : Mettre à jour son Raspberry Pi et Raspberry Pi OS

```
sudo apt dist-upgrade
```

ou

```
sudo apt full-upgrade
```

Cette commande va mettre à jour le système, le noyau, les dépendances et tout ce qui est relié de près ou de loin aux logiciels installés. Vu la quantité de données qui peuvent être modifiées, cette commande augmente également le risque d'instabilité système.

*L'avantage de cette commande est qu'elle permet également de changer la version de l'OS installé ! Si vous étiez par exemple sur Raspbian 9 (Stretch), vous passerez directement en Raspbian 10 (Buster). Pratique !*

Avec les commandes que nous venons de voir, votre système sera à jour.

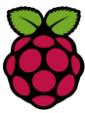
Cependant, le Raspberry Pi possède une commande de plus, car un autre élément peut avoir besoin d'être mis à jour : le firmware.

```
sudo rpi-update
```

Lorsque vous allez exécuter cette commande, un message va vous alerter que cette commande ne fait pas partie du « process » de mise à jour, et qu'il faut une raison bien précise pour exécuter cette commande. Vérifiez bien que rien ne risque de perturber l'installation (coupure électrique ou réseau) le temps de la mise à jour.

Vous verrez dans un prochain tutoriel qu'il est important de connaître cette commande, car il arrive que la *Fondation Raspberry* rajoute certaines fonctionnalités, uniquement disponibles après avoir mis à jour le firmware. Dans tous les cas, s'assurer d'avoir la dernière version est toujours un plus, car avec le temps, de nombreuses mises à jour sont déployées et permettent, par exemple, de réduire la température du processeur ou encore la consommation électrique.

Voici le message que vous verrez :



## Tutoriel N°3 : Mettre à jour son Raspberry Pi et Raspberry Pi OS

```
pi@ARTANIS:~ $ sudo rpi-update
*** Raspberry Pi firmware updater by Hexxeh, enhanced by AndrewS and Dom
*** Performing self-update
*** Relaunching after update
*** Raspberry Pi firmware updater by Hexxeh, enhanced by AndrewS and Dom
*** We're running for the first time
*** Backing up files (this will take a few minutes)
*** Backing up firmware
*** Backing up modules 5.10.17-v7l+
#####
WARNING: This update bumps to rpi-5.10.y linux tree
See: https://www.raspberrypi.org/forums/viewtopic.php?f=29&t=288234
'rpi-update' should only be used if there is a specific
reason to do so - for example, a request by a Raspberry Pi
engineer or if you want to help the testing effort
and are comfortable with restoring if there are regressions.

DO NOT use 'rpi-update' as part of a regular update process.

#####
Would you like to proceed? (y/N)
```

Lorsque vous serez sûr (ou presque) que rien ne viendra perturber votre mise à jour, tapez « Y » pour lancer la procédure.

Nous avons vu toutes les commandes qui permettent de tenir son système à jour ainsi que leur description complète. Il reste cependant quelques commandes supplémentaires très utiles à connaître.

Chaque paquet installé sur le système via APT est conservé en archive à l'emplacement « `/var/cache/apt/archives` ». La commande « `clean` » permet de les supprimer sans risque afin de récupérer un peu d'espace disque.

Pour effectuer cette opération, saisissez :

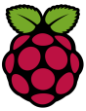
```
sudo apt clean
```

Il arrive également qu'une application soit mise à jour à plusieurs reprises. La commande « `clean` » supprime tout, alors que la commande que nous allons voir, est un peu plus « fine ». Là où la commande précédente supprime la totalité des archives de paquet, cette commande conservera toujours la dernière version d'un paquet.

Pour effectuer cette opération :

```
sudo apt autoclean
```

Cette commande efface moins d'éléments que la précédente mais permet tout de même de gagner beaucoup d'espace de stockage.



## Tutoriel N°3 : Mettre à jour son Raspberry Pi et Raspberry Pi OS

Lorsqu'un paquet est installé, il peut posséder des dépendances mais également des fichiers de configuration. Les commandes que nous avons vues ne permettent pas d'effacer la totalité de ces fichiers et éléments. Heureusement, une commande existe afin de purger notre système de ces éléments inutiles :

```
sudo apt autoremove --purge
```

Pour finir, il peut arriver que nous ne souhaitons plus conserver un paquet en particulier. La dernière commande du tutoriel que nous allons voir, permet de supprimer un paquet ainsi que ses dépendances et ses fichiers de configuration :

```
sudo apt autoremove --purge nom_du_paquet
```

Vous savez maintenant comment gérer les paquets et les mises à jour de votre Raspberry Pi.