

PREMIERS PAS AVEC UN RASPBERRY PI 3

Alain MICHEL



RÉSUMÉ :

Ce guide est destiné à présenter de manière assez détaillée l'**installation et la première configuration d'un Raspberry Pi 3**, avec le système d'exploitation **Raspbian**.

Il est destiné à des débutants qui souhaitent se lancer dans la découverte de ces excellents mini-ordinateurs à tout faire. Il présente les tout premiers pas, depuis le **téléchargement et l'installation du système** d'exploitation jusqu'aux **premiers paramètres nécessaires** pour commencer à utiliser son Raspberry. La dernière partie présente quelques **configurations un peu plus « avancées »**, non pas parce qu'elles sont complexes à mettre en œuvre, mais plutôt parce qu'elles sont plus spécifiques à une certaine utilisation qu'on voudra avoir de sa machine.

Les manipulations décrites ci-dessous sont détaillées le plus précisément possible pour être réalisables par des personnes désirant découvrir ces micro-ordinateurs et n'ayant aucune connaissance des systèmes d'exploitation Linux largement utilisés pour les Raspberry Pi.

Vous pouvez [télécharger une version PDF](#) ^[P.] de ce guide.

AUTEURS :

- Alain MICHEL

LICENCES :

Creative Commons - Attribution - Partage dans les Mêmes Conditions

Table des matières

I. TÉLÉCHARGEMENT, INSTALLATION ET PREMIER DÉMARRAGE DU SYSTÈME	5
1. Avant de commencer...	5
2. Téléchargement et installation du système	5
a. Téléchargement du système : Raspberry Pi OS	5
b. Installation sur la carte micro-SD	6
3. Premier démarrage	10
II. PREMIÈRES CONFIGURATIONS	14
1. Les premiers réglages	14
2. Passer le clavier en français	14
3. Configurer la localisation	17
4. Modifier le mot de passe	19
5. Activer le SSH	21
6. Mettre à jour le système	23
7. Utiliser l'interface de configuration Raspi-Config	25
8. Personnaliser le menu principal des applications	26
III. CONFIGURATIONS AVANCÉES	28
1. Pour aller un peu plus loin...	28
2. Empêcher la mise en veille de l'écran après 10 minutes	28
3. Démarrer en mode console	31
4. Connaître l'adresse IP du Raspberry - Attribuer une IP fixe	33
5. Contrôler à distance son Raspberry Pi	36
a. Contrôler à distance son Raspberry Pi avec SSH	36
b. Contrôler à distance son Raspberry Pi avec VNC	41
6. Configurer son Raspberry entièrement à distance avec SSH	48
a. Activer le protocole SSH directement sur la carte micro-SD	49
b. Attribuer une adresse IP fixe directement sur la carte micro-SD	50
c. Premier démarrage et premières configurations entièrement via SSH	53
d. Mise à jour du système via SSH	59
7. Utiliser Pi-hole sur son Raspberry pour bloquer les publicités	61
a. Présentation de Pi-hole	61
b. 1. Télécharger et installer Pi-hole	62
c. 2. Configurer ses machines pour utiliser Pi-hole	68
d. 3. Interface Web d'administration de Pi-hole	83
8. Modifier le dépôt de mises à jour	92
IV. POUR ALLER PLUS LOIN...	94
1. 3 sites à connaître	94
2. Raspberrypi.org	94
3. Raspbian-france.fr	94

4. Framboise314.fr	94
GLOSSAIRE	95

I. Téléchargement, installation et premier démarrage du système

1. Avant de commencer...

MATÉRIEL NÉCESSAIRE :

- Un **Raspberry Pi 3 ou 4** avec son boîtier et son alimentation
- Une carte micro SD (16 Go minimum) et un adaptateur micro-SD / USB ou bien micro-SD / SDCard pour la brancher sur son ordinateur
- Un clavier USB et une souris USB
- Un câble HDMI et un écran (avec entrée HDMI)
- Une connexion (WiFi ou filaire RJ45)
- Un PC (sous Linux de préférence ; tout à fait faisable sous Windows, cependant)

2. Téléchargement et installation du système

a. Téléchargement du système : Raspberry Pi OS

AUTEURS :

- Alain MICHEL

LICENCES :

Creative Commons - Attribution - Partage dans les Mêmes Conditions

Les Raspberry Pi sont livrés par défaut **vierges de tout système d'exploitation** et de tout dispositif de stockage.

C'est à l'utilisateur de choisir et d'installer le système d'exploitation qu'il veut (la plus grande majorité sont des systèmes Linux) et de choisir la carte de stockage de type micro-SD avec la capacité qu'il veut (16 Go minimum conseillés).

i Définition : Quel OS choisir ?

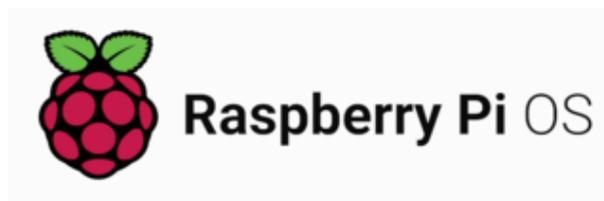
Il existe de nombreux systèmes d'exploitation dédiés au Raspberry Pi.

Certains sont « spécialisés » pour un certain type d'activité, par exemple, RecalBox OS qui est dédié à la transformation de votre Raspberry Pi en une console de jeu dite de « *retrogaming* », ou encore OSMC pour ceux qui désirent transformer leur Raspberry en « *media center* ».

Dans ce petit guide, on se contentera d'utiliser un système d'exploitation « *standard* », c'est à dire sans fonctionnalités orientées pour une utilisation spécifique.

Dans cette optique, **Raspberry Pi OS** (anciennement nommé Raspbian) est particulièrement indiqué. Il s'agit d'un système d'exploitation GNU/Linux, spécialement conçu et optimisé pour les Raspberry Pi.

Sa version actuelle est basée sur la dernière version de la distribution Debian, baptisée Buster (version 10.X).



p Conseil : Raspberry Pi OS

Pour faire fonctionner notre Raspberry, on va donc commencer par **télécharger** son système d'exploitation **Raspberry Pi OS** à partir de cette page du site Raspbian France :

<https://raspberrypi.fr/telechargements/>



Attention :

Le fichier téléchargé est une archive compressée .ZIP

Il est **nécessaire d'extraire son contenu** (glisser - déposer dans le répertoire de son choix sur l'ordinateur) pour obtenir le fichier .img de l'image de Raspbian qu'on installera sur la carte micro-SD.

À la date de réalisation de ce guide, l'image de Raspberry Pi OS obtenue est un fichier nommé **2021-01-11-raspios-buster-armhf-full.img** (selon les mises à jour de cet OS, le nom du fichier peut évidemment changer).

b. Installation sur la carte micro-SD

AUTEURS :

- Alain MICHEL

LICENCES :

Creative Commons - Attribution - Partage dans les Mêmes Conditions



Conseil :

La carte micro-SD doit être vierge et formatée. Si elle n'est pas vierge, elle sera complètement effacée.



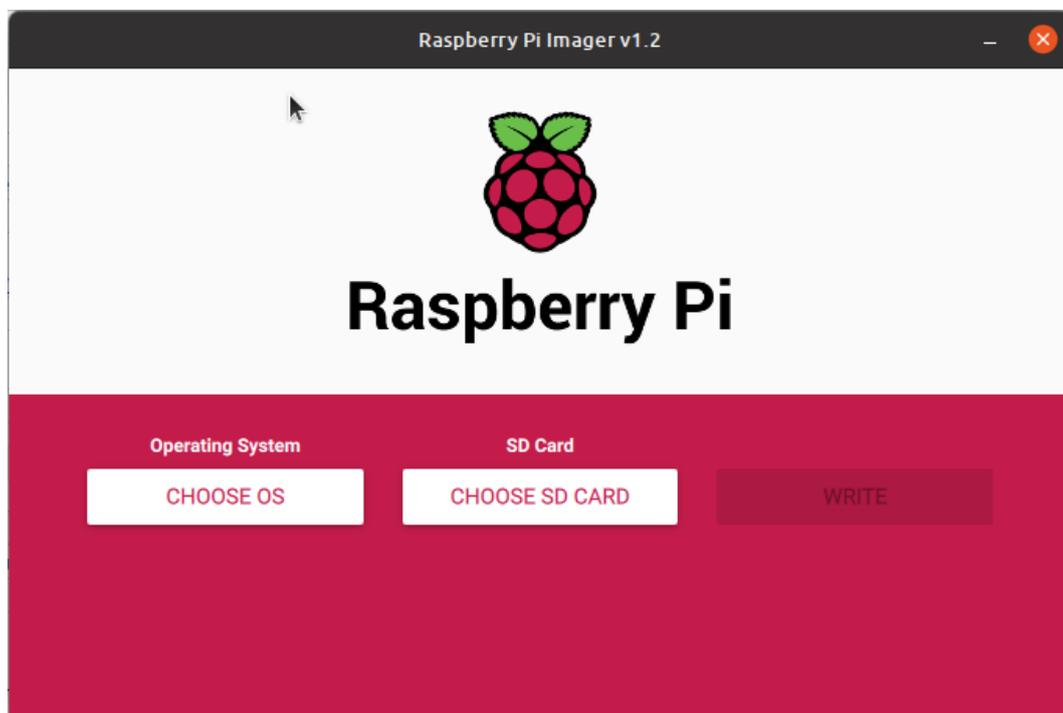
Attention : Nouveau

Depuis début mars 2020, il existe un nouveau logiciel proposé par la fondation Raspberry Pi pour **faciliter la création des cartes micro-SD, depuis n'importe quel système d'exploitation : Raspberry Pi Imager.**

→ Un tutoriel complet est accessible ici :

<https://raspberrypi.fr/creer-carte-sd-windows-mac-linux-raspberry-pi-imager/>

À noter : le logiciel permet même de télécharger directement l'image du système Raspbian. Si vous l'avez déjà téléchargée ou si vous préférez installer un autre système d'exploitation, vous aurez la possibilité de choisir votre propre image.



La fenêtre de l'installateur Raspberry Pi Imager



Complément :

Je laisse tout de même en ligne les contenus des deux chapitres ci-dessous à toutes fins utiles, pour archive.

Ils ne sont plus nécessaires si vous utilisez Raspberry Pi Imager.

Ils décrivent comment installer l'image de Raspberry Pi OS sur la carte micro-SD en passant par une ligne de commande, dans le terminal. Cette méthode fonctionne parfaitement, si vous décidez de ne pas utiliser l'installeur « *Raspberry Pi Imager* ».

Avec un ordinateur sous Linux

C'est le cas de figure idéal, où quelques commandes suffiront pour copier sur la carte micro-SD l'image de Raspberry Pi OS téléchargée à l'étape précédente.

Les explications ci-dessous sont faites pour les systèmes Linux basés sur Debian et ses dérivés (Ubuntu, etc.)

3 étapes successives pour installer le système Raspberry Pi OS sur la carte micro-SD :

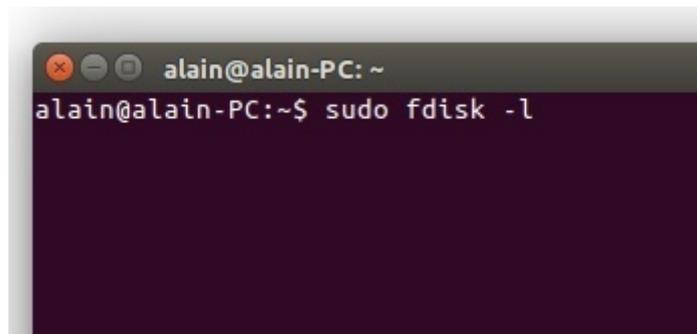


Méthode : 1. Repérer le nom de la carte

Il faut commencer par **brancher sa carte micro-SD sur l'ordinateur**, soit avec un adaptateur micro-SD / USB, soit avec un adaptateur micro-SD / Sdcard (si vous avez un lecteur de carte SD intégré à votre ordinateur).

La première opération va consister à **repérer le nom de la carte** pour l'ordinateur.

Dans le terminal, saisir la commande **fdisk -l** (précédée de l'instruction **sudo** pour avoir les droits administrateur), puis valider par la touche Entrée, ce qui donne :



```
alain@alain-PC: ~  
alain@alain-PC:~$ sudo fdisk -l
```

Il sera nécessaire de saisir le mot de passe administrateur.

L'exécution de cette commande va renvoyer la liste de tous les disques installés sur l'ordinateur (disques durs et disques amovibles USB), avec des informations techniques comme leur capacité en Go, entre autres choses, leurs partitions existantes et leur nom ([plus d'infos sur ces noms...](#))

Pour repérer sa carte micro-SD dans la liste, on s'aidera de la capacité affichée.

ATTENTION : sur mon exemple, et pour les besoins de la capture d'écran ci-dessous, il s'agit d'une **clé USB** de 2 Go (1,9 GiB).

```

alain@alain-PC: ~
Disque /dev/sdc : 2,7 TiB, 3000592982016 octets, 5860533168 secteurs
Unités : sectors of 1 * 512 = 512 octets
Sector size (logical/physical): 512 bytes / 4096 bytes
I/O size (minimum/optimal): 4096 bytes / 4096 bytes
Disklabel type: gpt
Disk identifier: 8FC1136F

Périphérique Start      Fin    Secteurs  Size Type
/dev/sdc1      2048 5860532223 5860530176 2,7T Linux filesystem

Disque /dev/sdd : 1,9 GiB, 1993342976 octets, 3893248 secteurs
Unités : sectors of 1 * 512 = 512 octets
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x00000000

Périphérique Amorçage Start      Fin Secteurs  Size Id Type
/dev/sdd1      *      32 3893247 3893216 1,9G b W95 FAT32
alain@alain-PC: ~$

```

→ Sur la même ligne, à gauche de la capacité du disque (de la carte), on voit son nom : **/dev/sdd**

C'est ce nom qu'il faut noter.

On voit également que ce disque amovible est bien partitionné, qu'il comporte une seule partition appelée **/dev/sdd1** qui occupe tout l'espace disponible (1,9 G) et qu'elle est formatée dans le système de fichiers FAT32.

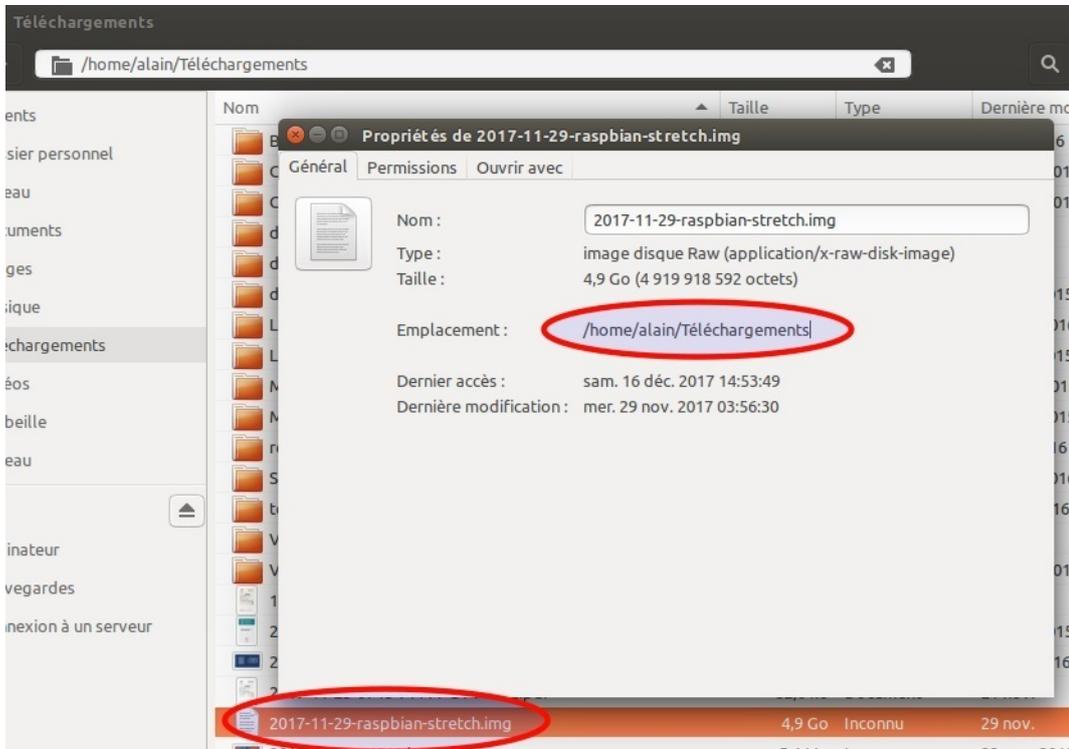
ATTENTION : Ce n'est PAS ce nom de partition **/dev/sdd1** qu'on utilisera tout à l'heure, mais bien celui de la carte (pas de chiffre à la fin du nom).



Méthode : 2. Repérer le chemin vers le fichier image téléchargé

Il faut maintenant connaître le **chemin exact vers le fichier** de Raspbian téléchargé auparavant, dans le répertoire où il se trouve.

Pour cela, il suffit de faire un clic-droit dessus, de choisir « *Propriétés* » et de noter ce chemin indiqué :



Dans mon exemple, le fichier se trouve dans le dossier des téléchargements.

Le **chemin complet vers ce fichier** sera donc le suivant : **/home/alain/Téléchargements/2017-11-29-raspbian-stretch.img**

(il s'agit d'une ancienne capture d'écran, avec l'ancien nom de l'OS : Raspbian)



Méthode : 3. Copie du fichier image sur la carte micro-SD

Dernière étape : il faut maintenant **ouvrir une fenêtre de terminal** pour saisir la commande qui va copier le fichier image de Raspbian sur la carte micro-SD.

La commande se construira avec les noms qu'on a notés précédemment, sur le modèle :

→ `sudo dd bs=1M if=chemin_vers_le_fichier_raspbian of=/dev/nom_de_ma_carte_microSD`

 Dans mon exemple, la commande sera donc :

`sudo dd bs=1M if=/home/alain/Téléchargements/2017-11-29-raspbian-stretch.img of=/dev/sdd`
validée par la touche Entrée.

Pour ceux que ça intéresse, la signification des différents éléments de cette ligne de commande :

- `sudo` : pour lancer la commande en administrateur
- `dd` : la commande qui lance la copie, selon une méthode particulière
- `bs=1M` : pour Block Size (copier les données par « blocs » de 1 Mo)
- `if` : pour Input File (fichier d'entrée = chemin complet vers le fichier à copier)
- `of` : pour Output File (fichier de sortie = chemin vers la destination de la copie).



Attention :

À l'exécution de cette commande, la copie du fichier Raspbian sur la carte micro-SD dans le but d'en faire un disque « bootable » va commencer. Elle va durer plusieurs minutes. **Ne rien toucher tant que le processus n'est pas terminé.**

NB : disque « bootable » = disque contenant un système d'exploitation et sur lequel un ordinateur sera capable de démarrer puis de fonctionner.



Complément :

Si on n'a pas d'ordinateur sous Linux à disposition, on peut aussi choisir une solution « intermédiaire » en faisant démarrer son PC Windows sur une clé Live-USB Linux Ubuntu ou Debian, de façon à pouvoir travailler quand même dans un environnement Linux.

Avec un ordinateur sous Windows

Au préalable

Si on ne dispose pas d'ordinateur sous Linux, il est bien évidemment possible de réaliser la préparation de la carte micro-SD à partir d'un ordinateur Windows.

Pour cela, on aura besoin d'installer tout d'abord un logiciel spécifique, dédié à cette opération : le logiciel libre **Win32DiskImager**.

[Lien direct pour le téléchargement](#)

Lien vers le site : <https://sourceforge.net/projects/win32diskimager/>



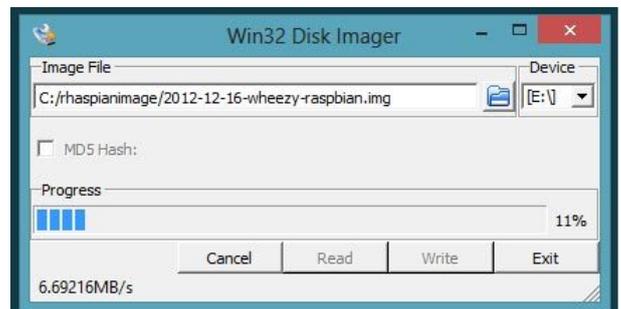
Méthode :

En cliquant sur le lien ci-dessous, vous accéderez à un descriptif de la méthode à utiliser.

Ce guide est **publié par Raspbian France**, sur cette page :

<https://raspbian-france.fr/creez-carte-sd-raspbian-raspberry-pi-windows/>

[Télécharger le contenu en PDF](#)^[p.]



3. Premier démarrage

AUTEURS :

- Alain MICHEL

LICENCES :

Creative Commons - Attribution - Partage dans les Mêmes Conditions



Rappel :

Pour le tout premier démarrage, il sera nécessaire d'avoir la totalité du matériel à sa disposition :

- Un Raspberry Pi 3 avec son boîtier et son alimentation
- La carte micro SD (avec le système Raspberry Pi OS installé à l'étape précédente) insérée dans son logement dans le Raspberry.
- Un clavier USB et une souris USB
- Un câble HDMI et un écran (avec entrée HDMI)
- Une connexion (WiFi ou filaire RJ45)



Attention :

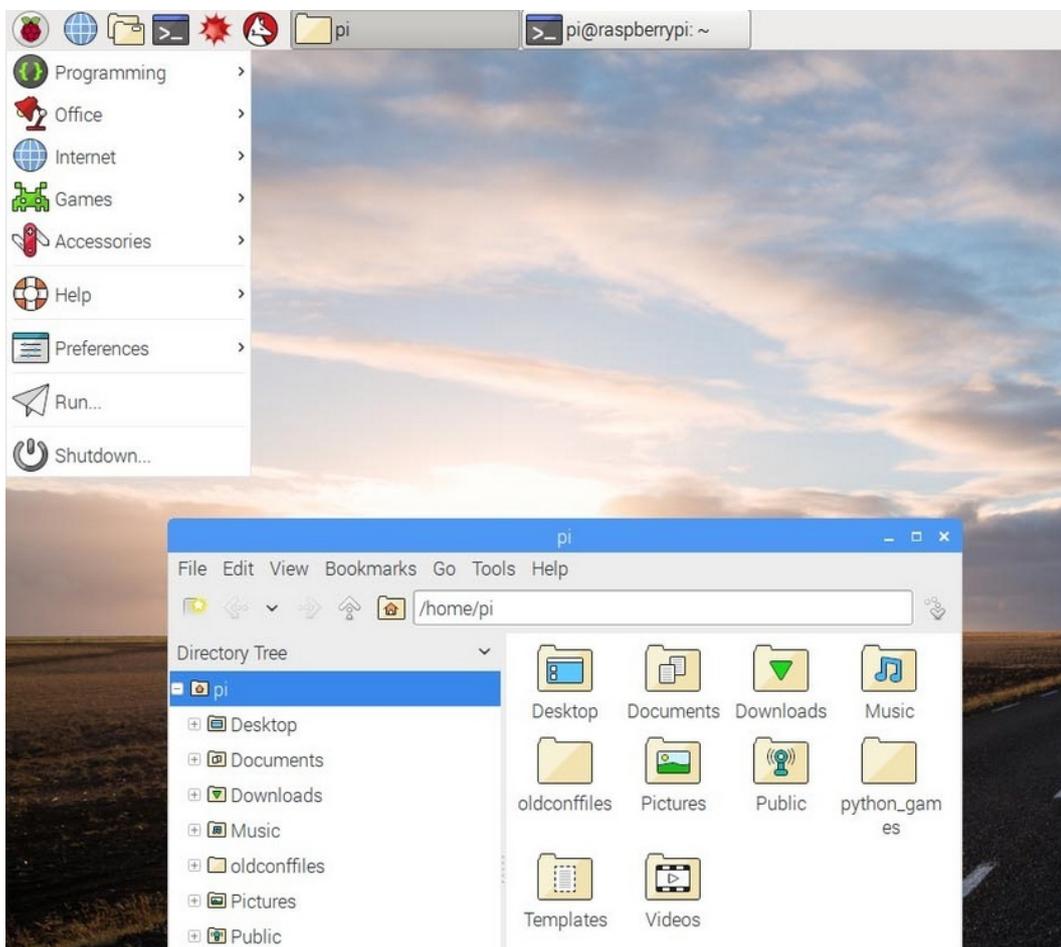
Le Raspberry **ne possède pas** d'interrupteur marche / arrêt :

- il s'allume quand on le branche sur le secteur ;
- il s'éteint quand on le débranche (après avoir arrêté correctement le système d'exploitation, comme sur tous les ordinateurs).



Méthode :

- Brancher le clavier et la souris sur les ports USB du Raspberry Pi,
- Brancher le câble HDMI sur la prise correspondante du Raspberry et derrière votre écran ; brancher l'écran sur le secteur,
- Brancher le cordon d'alimentation du Raspberry dans sa prise dédiée puis le boîtier transformateur sur le secteur.
- Le Raspberry et l'écran s'allument, au bout de quelques secondes vous êtes « sur le bureau » de votre tout nouvel ordinateur...

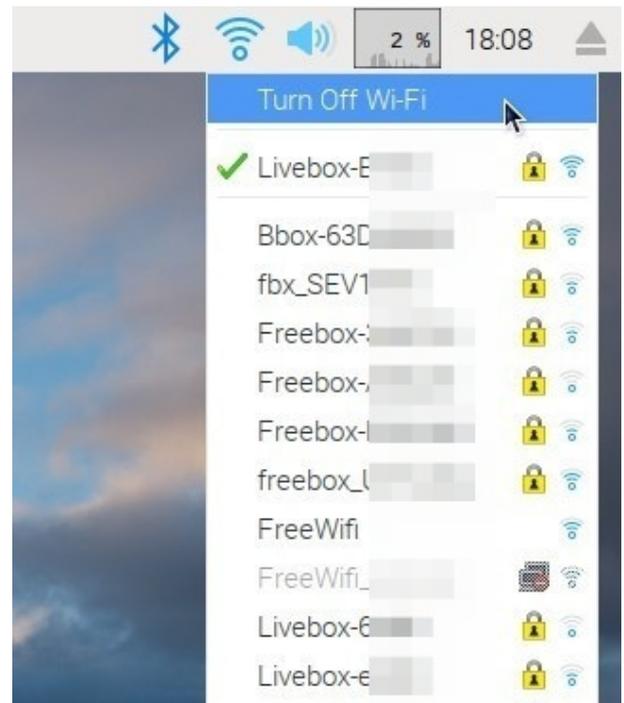


**Conseil :**

Pensez dès à présent à **connecter le Raspberry Pi à internet**, il dispose du **WiFi**, si vous êtes à portée de votre box.

Il suffit de cliquer sur l'icône du WiFi en haut à droite de l'écran, de choisir votre réseau puis de saisir la clé de sécurité.

Sinon, le connecter directement à la box avec un câble Ethernet.





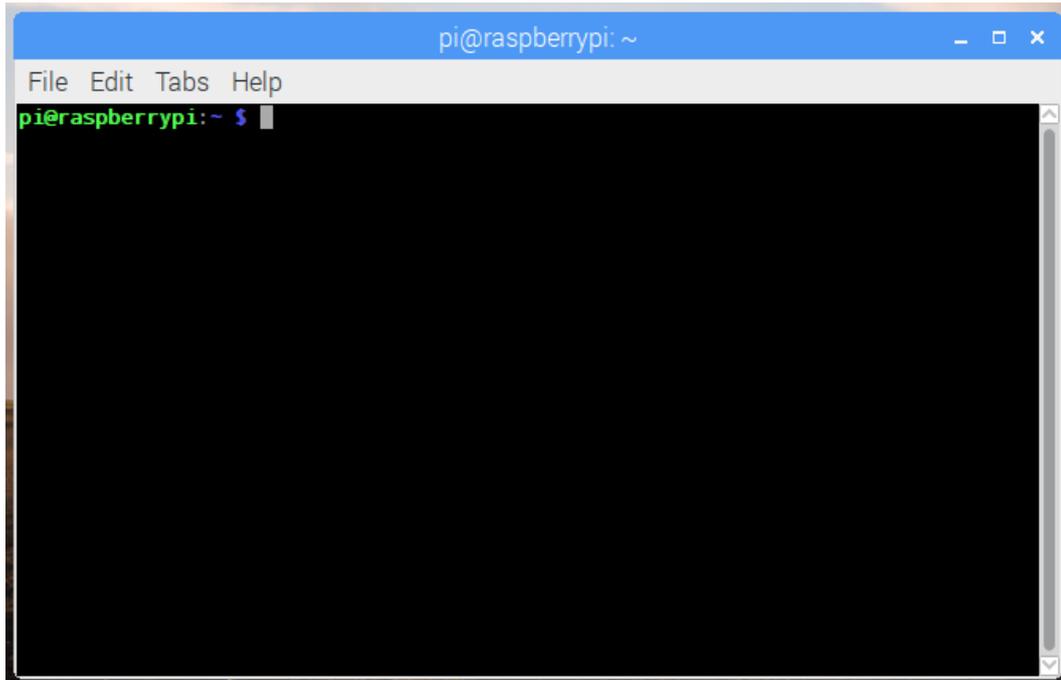
Complément :

Par défaut, le système est en langue anglaise ; on va le changer par la suite pour le mettre en français.

Par défaut également, le système d'exploitation démarre entièrement jusqu'à l'affichage du bureau de l'utilisateur **sans exiger d'identification à l'ouverture de session**. Là aussi, on pourra changer ce comportement par défaut pour l'adapter à son utilisation.

Par défaut toujours, le nom de la machine (le « *hostname* ») est « **raspberrypi** », le nom de l'utilisateur est « **pi** » et son mot de passe est « **raspberry** ».

D'où l'invite de commande qu'on peut voir quand on ouvre le **terminal** : **pi@raspberrypi** (comprendre : c'est l'utilisateur « *pi* » qui a ouvert une session sur la machine « *raspberrypi* »).



Le *Hostname* peut lui aussi être changé par la suite, cela peut être utile dans le cas d'une utilisation de plusieurs Raspberry dans un réseau local.



Rappel :

Le nom d'utilisateur et le mot de passe par défaut évoqués ci-dessus sont valables pour la distribution Raspbian choisie pour ce guide.

D'autres distributions auront d'autres identifiants par défaut ; ils sont toujours précisés sur les sites qui proposent les distributions pour Raspberry Pi.

II. Premières configurations

1. Les premiers réglages

Une fois que votre Raspberry est allumé et a fini de démarrer, il va falloir faire quelques réglages et configurations « de base » avant de pouvoir commencer à l'utiliser vraiment.

2. Passer le clavier en français

AUTEURS :

- Alain MICHEL

LICENCES :

Creative Commons - Attribution - Partage dans les Mêmes Conditions

! Attention :

Il y a **deux réglages différents** à faire si on veut configurer le clavier en français pour tout type d'utilisation :

- la configuration du clavier pour l'utilisation du Raspberry Pi **en « mode console »** (c'est à dire en ligne de commande, sans interface graphique - peu fréquent pour les débutants, mais indispensable dans certains cas)
- la configuration du clavier pour l'utilisation **en mode graphique** (c'est à dire à travers une interface graphique avec le bureau, des menus, des fenêtres, etc. ; celle à laquelle tout le monde est habitué sur son ordinateur).

Ces deux réglages sont indépendants et si, par exemple, on ne configure le clavier en français que en mode graphique, il pourra très bien être toujours en anglais si on redémarre le Raspberry Pi en mode console... De quoi s'arracher les cheveux si on n'est pas prévenu ! Surtout dans le cas de saisie de mot de passe...

📎 Conseil :

Nous allons voir ici comment **passer le clavier en français pour une utilisation en mode console**.

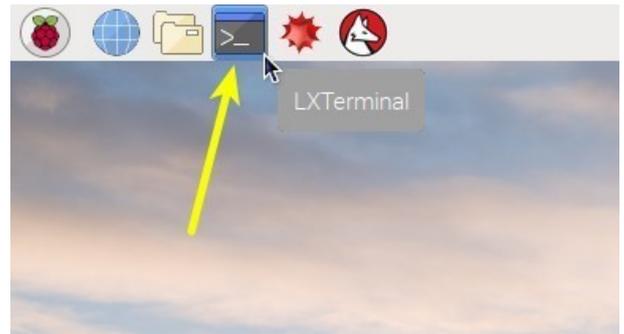
C'est la **toute première action à effectuer** étant donné les conseils de sécurité que j'évoquerai un peu plus bas (empêcher le démarrage et l'ouverture de session « automatique » et au contraire, exiger le login et le mot de passe en mode console, avant de lancer manuellement l'interface graphique).

La configuration du clavier en mode graphique se fera juste après.

⚙️ Méthode :

La manipulation consiste à **éditer un fichier du système** pour le modifier en déclarant la langue du clavier. L'édition de ce fichier se fera avec les droits administrateur, dans un terminal, au moyen de l'éditeur de texte présent par défaut, appelé **nano**.

1. Ouvrir un terminal, au moyen de l'icône en haut à gauche :



2. Saisir la commande **sudo nano /etc/default/keyboard** et valider par Entrée

ATTENTION : Sur le clavier, il faudra plutôt saisir : **sudo nqno /etc/default/keyboard** puisque quand on saisit ça, le clavier est pour le moment encore en **qwerty** !!!
Donc, appuyer sur la touche Q pour obtenir la lettre A...

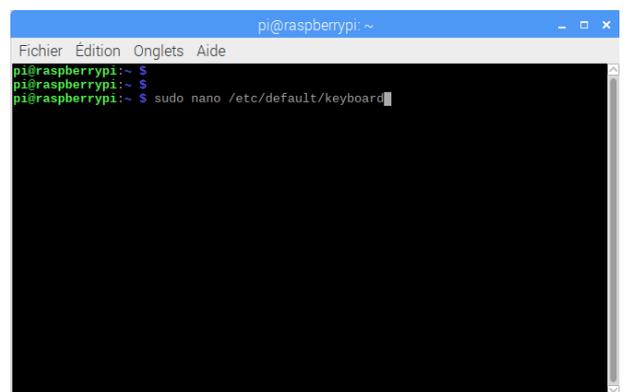
→ Saisir le mot de passe de l'utilisateur, qui est pour l'instant **raspberrypi** (c'est le mot de passe par défaut, on le changera par la suite)

Même remarque qu'à l'étape précédente : au moment de saisir le mot de passe après avoir appuyé sur la touche Entrée, au lieu de saisir « *raspberrypi* » il faudra utiliser les touches **rqspberry**

→ **IMPORTANT** : sur tous les systèmes Linux, **quand on saisit un mot de passe dans le terminal**, pour des raisons de sécurité, **RIEN n'apparaît**, on ne voit pas les caractères saisis, ni même des petites étoiles ou des petits cercles, comme on est habitué à voir sur d'autres systèmes. C'est normal.

Interprétation de la commande, pour ceux que ça intéresse :

- `sudo` : pour exécuter la commande en administrateur
- `nano` : pour lancer l'éditeur de texte
- `/etc/default/keyboard` : chemin vers le fichier à modifier



3. Le fichier *keyboard* s'ouvre alors dans l'éditeur *nano*, directement dans le terminal (vous pouvez redimensionner la fenêtre avec la souris pour y voir plus clair).

→ Repérer la 6e ligne où on peut lire **XKBLAYOUT="gb"**

```

GNU nano 2.7.4

# KEYBOARD CONFIGURATION FILE

# Consult the keyboard(5) manual page.

XKBMODEL="pc105"
XKBLAYOUT="gb"
XKBVARIANT=""
XKBOPTIONS=""

BACKSPACE="guess"

```

→ déplacer le curseur – avec les flèches du clavier, la souris est inopérante dans cette fenêtre – pour aller modifier la valeur **gb** et la remplacer par **fr**.

→ On doit lire maintenant **XKBLAYOUT="fr"**

```

pi@raspberrypi: ~
Fichier  Édition  Onglets  Aide
GNU nano 2.7.4  Fichier : /etc/default/keyboard
# KEYBOARD CONFIGURATION FILE
# Consult the keyboard(5) manual page.
XKBMODEL="pc105"
XKBLAYOUT="fr"
XKBVARIANT=""
XKBOPTIONS=""
BACKSPACE="guess"

```

4. **Enregistrer la modification** avec la combinaison de touches **Ctrl + O** au clavier et **valider par la touche Entrée**.

5. **Quitter l'éditeur nano** avec la combinaison de touches **Ctrl + X**.

On se retrouve dans la fenêtre du terminal, comme au début.

```

pi@raspberrypi: ~
Fichier  Édition  Onglets  Aide
GNU nano 2.7.4  Fichier : /etc/default/keyboard
# KEYBOARD CONFIGURATION FILE
# Consult the keyboard(5) manual page.
XKBMODEL="pc105"
XKBLAYOUT="fr"
XKBVARIANT=""
XKBOPTIONS=""
BACKSPACE="guess"

```

Appuyer sur les touches Ctrl + X pour quitter.

Appuyer sur les touches Ctrl + O pour « écrire » les modifications dans le fichier.

6. Dernière étape : **faire redémarrer le Raspberry Pi** pour prendre en compte les modifications effectuées.

→ Saisir la commande **sudo reboot** et valider par la touche Entrée.

Le Raspberry redémarre.

Autre possibilité : fermer le terminal et faire redémarrer de la manière « *classique* », en passant par le menu des applications (en haut à gauche) → Shutdown → Reboot

```

pi@raspberrypi:~$ sudo nano /etc/default/keyboard
pi@raspberrypi:~$ sudo reboot

```

3. Configurer la localisation

AUTEURS :

- Alain MICHEL

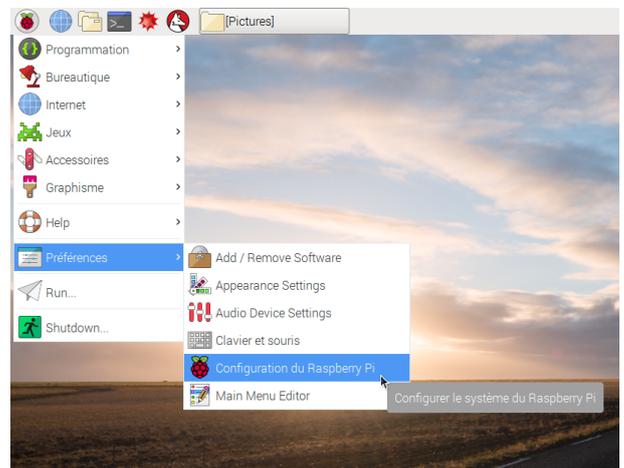
LICENCES :

Creative Commons - Attribution - Partage dans les Mêmes Conditions

Un ensemble de réglages qui vont modifier la langue du système pour la passer en français, ainsi que le fuseau horaire, la configuration du clavier et la région du WiFi.

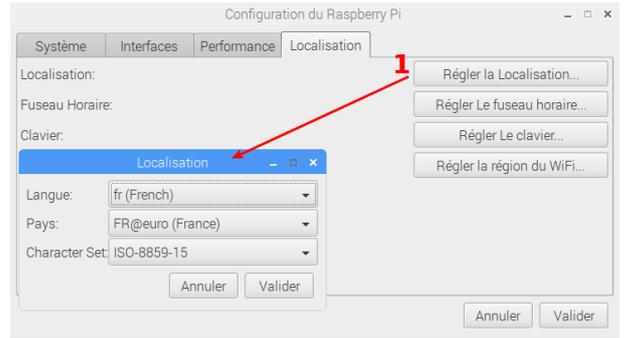
Méthode :

1. Accéder au menu des applications → **Préférences** → **Configuration** du Raspberry Pi.



2. Dans la fenêtre qui s'ouvre (que vous pouvez redimensionner à la souris), accéder au 4e onglet « **Localisation** » :

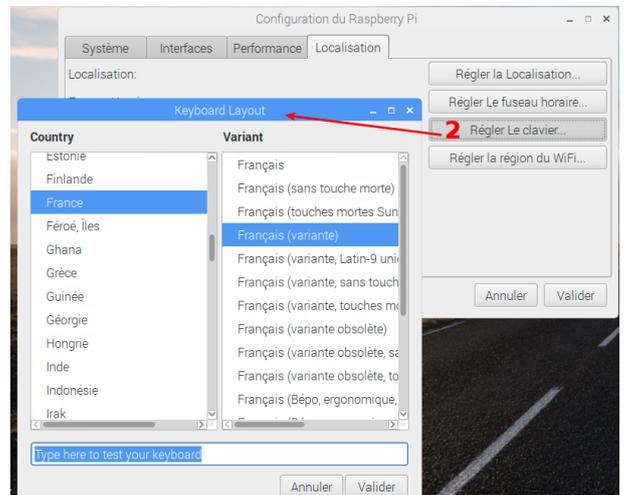
- Cliquer sur le bouton « **Régler la localisation** » puis sélectionner la langue et le pays.
- Valider.



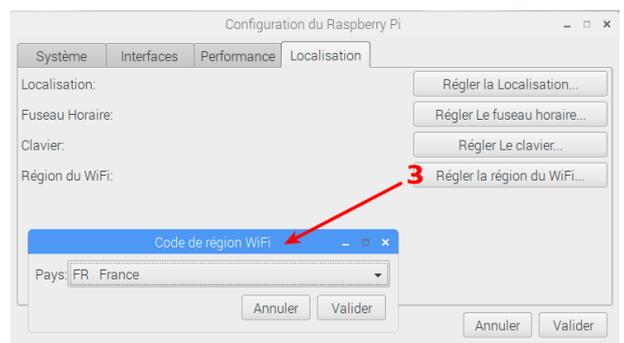
3. Régler de la même manière le **fuseau horaire** : Région → Europe ; Localisation → Paris

4. Cliquer sur « **Régler le clavier** » :

- Pays → France
- Variante → Français (tout court) ou Français (variante). Vous pouvez faire des tests dans le champ de saisie au bas de la fenêtre.
- Valider



5. Cliquer sur « **Régler la région du WiFi** » :
Pays → FR France



6. **Valider l'ensemble des modifications** effectuées avec le bouton Valider sur la fenêtre de configuration. Il va falloir **redémarrer le Raspberry Pi**.

4. Modifier le mot de passe



Attention :

Votre Raspberry Pi, on l'a vu précédemment, est configuré par défaut pour démarrer automatiquement jusqu'à l'affichage du bureau, sans avoir à saisir de mot de passe pour l'ouverture de session. De plus, le **mot de passe par défaut** « *raspberry* » est universellement connu, accessible par une simple recherche sur internet. Or, ce mot de passe d'administrateur est indispensable quand vous exécutez des commandes dans le terminal ou en mode console, par exemple pour modifier votre configuration ou pour effectuer des mises à jour du système.

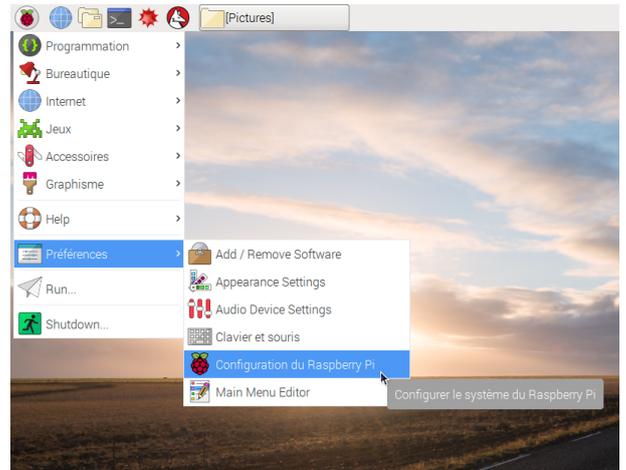


Il est donc **ESSENTIEL pour la sécurité** de votre machine de **modifier ce mot de passe** par défaut, à plus forte raison si vous destinez votre Raspberry Pi à une utilisation en tant qu'objet connecté, accessible depuis internet.



Méthode :

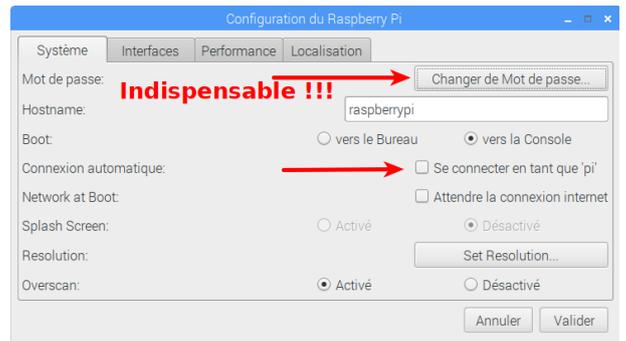
1. Accéder au menu des applications → **Préférences** → **Configuration** du Raspberry Pi.



2. Dans la fenêtre qui s'ouvre (que vous pouvez redimensionner à la souris), accéder au 1^{er} onglet «Système»

- Cliquer sur le bouton « **Changer le mot de passe** » et saisissez votre nouveau mot de passe.

- Valider



3. Sur cette même fenêtre, profitez-en pour **décocher** la case « **Se connecter en tant que 'pi'** » : l'ouverture de session ne se fera plus de manière automatique.

Au cours du démarrage, vous serez invités à saisir :

- votre login : **pi**
- votre mot de passe : celui que vous venez de personnaliser



Complément :

C'est également sur cet onglet que vous pourrez modifier le nom d'hôte, c'est à dire le nom de la machine tel qu'il apparaît sur votre réseau local.

Par défaut, ce nom d'hôte est « *raspberrypi* » .

Vous pouvez le personnaliser et mettre ce que vous voulez. Ce sera plus particulièrement intéressant et nécessaire si vous utilisez deux ou plusieurs Raspberry Pi sur votre réseau local.

5. Activer le SSH

AUTEURS :

- Alain MICHEL

LICENCES :

Creative Commons - Attribution - Partage dans les Mêmes Conditions



Définition :

Le protocole SSH (**Secure Shell**) est un protocole de communication sécurisé, dans lequel toutes les données échangées entre deux machines distantes sont chiffrées.

Pour plus d'informations sur le fonctionnement de SSH, voir ici :

<https://openclassrooms.com/courses/reprenez-le-contrôle-a-l'aide-de-linux/la-connexion-sécurisée-a-distance-avec-ssh>



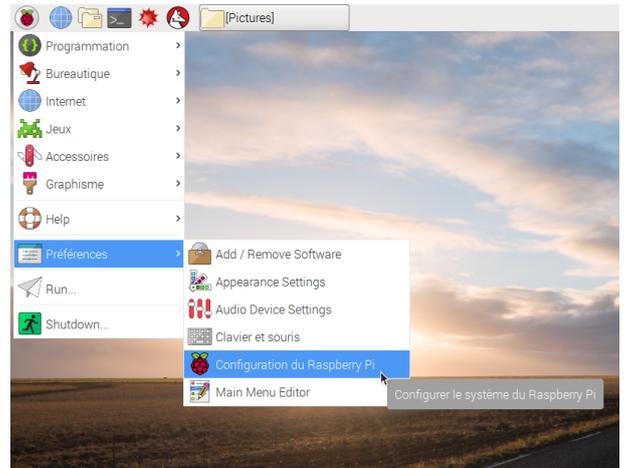
Conseil :

Avec notre Raspberry Pi, il peut être particulièrement intéressant d'utiliser ce protocole SSH pour se connecter au Raspberry depuis une autre machine et pouvoir lui faire exécuter des commandes (par exemple pour faire les mises à jour du système) ou lancer des applications à distance (par exemple lancer un mediacenter pour diffuser des contenus sur un écran public), depuis un autre ordinateur connecté.

Cela nous permet de contrôler le Raspberry Pi à distance, sans nécessiter le branchement d'un clavier et d'une souris ainsi que d'un écran dédiés.

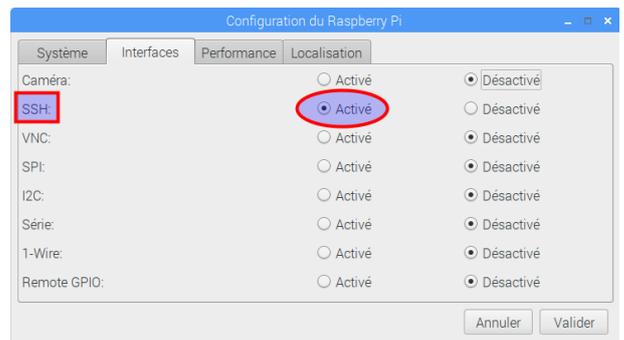
**Méthode :**

1. Accéder au menu des applications → **Préférences** → **Configuration** du Raspberry Pi.



2. Dans la fenêtre qui s'ouvre (que vous pouvez redimensionner à la souris), accéder au 2^e onglet **Interfaces**.

- Cocher le bouton « **Activé** » sur la seconde ligne **SSH**.
- Valider

**Attention :**

Le **serveur SSH** du Raspberry Pi est désormais activé. Il est prêt à fonctionner.

Pour pouvoir s'y connecter depuis un autre ordinateur et le contrôler à distance, il va falloir ensuite installer sur cet ordinateur distant un *client SSH*.

Cela sera décrit plus bas, dans le chapitre « *Configurations avancées* »

**Complément :**

C'est également sur cet onglet « *Interfaces* » qu'on pourra éventuellement activer d'autres fonctionnalités, d'autres interfaces possibles, comme par exemple **VNC**, pour pouvoir contrôler le Raspberry Pi à distance ET exécuter ses programmes dans un environnement graphique, sur l'ordinateur distant (et non pas uniquement en mode console, en ligne de commande, comme avec SSH. Nous verrons cela dans la partie « *Configurations avancées - Contrôler à distance son Raspberry Pi* ».

On pourra aussi activer le port de contrôle de la caméra pour connecter une webcam, ou les différentes possibilités de connexion de cartes micro-contrôleur (cartes GPIO) permettant d'ajouter de très nombreuses fonctionnalités.

6. Mettre à jour le système

AUTEURS :

- Alain MICHEL

LICENCES :

Creative Commons - Attribution - Partage dans les Mêmes Conditions

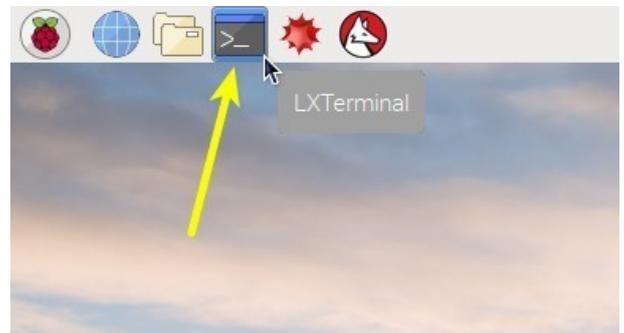
Il est **indispensable de maintenir son système à jour**, ne serait-ce que pour des raisons de sécurité (correction d'éventuelles failles).

Cette opération peut se faire très simplement, en lançant les commandes appropriées, **elle doit être faite régulièrement**.



Méthode :

1. Ouvrir un terminal, au moyen de l'icône en haut à gauche :



2. Saisir la commande **sudo apt update**, Appuyez sur la touche Entrée, saisissez le mot de passe et valider.

Le Raspberry Pi se connecte au *dépôt* (le serveur) dédié hébergeant les *paquets* (fichiers, logiciels) de la distribution Raspbian installée, de manière à y récupérer les dernières versions.

Attendre la fin de l'exécution de la commande.

```
pi@raspberrypi ~
alain@alain-PC:~$ ssh pi@192.168.1.13
pi@192.168.1.13's password:
Linux raspberrypi 4.9.59-v7+ #1047 SMP Sun Oct 29 12:19:23 GMT 2017 an

The programs included with the Debian GNU/Linux system are free softwa
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon Dec 25 17:07:50 2017 from 192.168.1.21
pi@raspberrypi:~$ sudo apt update
Atteint:1 http://raspberrypi.org/debian stretch InRelease
Atteint:2 http://archive.raspberrypi.org/debian stretch InRelease
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances
Lecture des informations d'état... Fait
All packages are up to date.
pi@raspberrypi:~$
```

3. Saisir la commande **sudo apt upgrade** et valider par la touche Entrée.

- Répondre **Oui** aux questions éventuelles (saisir **O** (pour **Oui**) et valider par Entrée).

```
pi@raspberrypi:~$ sudo apt upgrade
Réception de:1 http://raspbian.42.fr/raspbian stretch InRelease [15,0 kB]
Atteint:2 http://archive.raspberrypi.org/debian stretch InRelease
Réception de:3 http://raspbian.42.fr/raspbian stretch/main armhf Packages [11,7 MB]
11,7 Mo réceptionnés en 13s (875 ko/s)

Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances
Lecture des informations d'état... Fait
9 packages can be upgraded. Run 'apt list --upgradable' to see them.
pi@raspberrypi:~$ sudo apt upgrade
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances
Lecture des informations d'état... Fait
Calcul de la mise à jour... Fait
Les paquets suivants seront mis à jour*:
  imagemagick imagemagick-6-common imagemagick-6.q16 libimage-magick-perl
  libimage-magick-q16-perl libmagickcore-6.q16-3 libmagickcore-6.q16-3-extra
  libmagickwand-6.q16-3 perlmagick
0 mis à jour, 0 nouvellement installés, 0 à enlever et 0 non mis à jour.
Il est nécessaire de prendre 3 319 ko dans les archives.
Après cette opération, 0 o d'espace disque supplémentaires seront utilisés.
Souhaitez-vous continuer ? [O/n]
```

4. Attendre la fin de l'exécution de la commande, c'est à dire du téléchargement puis de l'installation des mises à jour éventuelles.

Si vous n'avez jamais fait cette mise à jour, ou du moins pas fait depuis longtemps, et en fonction de la qualité de votre connexion internet, cette opération peut être relativement longue (plusieurs minutes ou dizaines de minutes...).

```
pi@raspberrypi:~$ sudo apt upgrade
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances
Lecture des informations d'état... Fait
Les paquets suivants seront mis à jour*:
  imagemagick imagemagick-6-common imagemagick-6.q16 libimage-magick-perl
  libimage-magick-q16-perl libmagickcore-6.q16-3 libmagickcore-6.q16-3-extra
  libmagickwand-6.q16-3 perlmagick
9 mis à jour, 0 nouvellement installés, 0 à enlever et 0 non mis à jour.
Il est nécessaire de prendre 3 319 ko dans les archives.
Après cette opération, 0 o d'espace disque supplémentaires seront utilisés.
Souhaitez-vous continuer ? [O/n] o
Réception de:1 http://raspbian.42.fr/raspbian stretch/main armhf libimage-magick-q16-
8:6.9.7.4+dfsg-11+deb9u4 [216 kB]
Réception de:2 http://raspbian.42.fr/raspbian stretch/main armhf libimage-magick-perl
:7.4+dfsg-11+deb9u4 [53,3 kB]
Réception de:3 http://raspbian.42.fr/raspbian stretch/main armhf perlmagick all 8:6.9
11+deb9u4 [1 264 B]
Réception de:4 http://raspbian.42.fr/raspbian stretch/main armhf imagemagick-6-common
:7.4+dfsg-11+deb9u4 [184 kB]
Réception de:5 http://raspbian.42.fr/raspbian stretch/main armhf libmagickcore-6.q16-
6:9.7.4+dfsg-11+deb9u4 [1 370 kB]
Réception de:6 http://raspbian.42.fr/raspbian stretch/main armhf libmagickwand-6.q16-
6:9.7.4+dfsg-11+deb9u4 [405 kB]
Réception de:7 http://raspbian.42.fr/raspbian stretch/main armhf imagemagick-6.q16 ar
7.4+dfsg-11+deb9u4 [563 kB]
72% [7 libimage-magick-q16 4 019 B/563 kB 1%]
```

Quand c'est terminé, vous pouvez refermer le terminal.



Conseil :

Les deux commandes décrites ci-dessus peuvent être combinées sur une seule et même ligne de commande, pour qu'elles s'exécutent l'une après l'autre, de manière enchaînée, sans interruption :

sudo apt update && sudo apt upgrade -y

→ On combine les deux commandes au moyen de l'opérateur **&&** et on ajoute l'option **-y** à la fin de la 2^e commande pour forcer la réponse "Oui" lors de l'opération d'installation de mise à jour des paquets.



Attention :

Il se peut que vous soyez confrontés à un problème particulier de connexion au serveur par défaut vers lequel la commande **apt update** tente de se connecter : **http://mirrordirector.raspbian.org/raspbian**.

Depuis quelques mois il est relativement inaccessible et semble causer des problèmes aux utilisateurs : la commande de mise à jour est extrêmement longue et finit par aboutir à des messages d'erreur.

Si c'est le cas, il va falloir **modifier le dépôt utilisé par défaut** pour les mises à jour.

Nous verrons cela dans la dernière partie « *Configurations avancées* »

7. Utiliser l'interface de configuration Rasp-Config

AUTEURS :

- Alain MICHEL

LICENCES :

Creative Commons - Attribution - Partage dans les Mêmes Conditions

Il existe un utilitaire de configuration nommé **raspi-config** qui se lance en ligne de commande, depuis un terminal.

Dans cet utilitaire, on peut configurer la plupart des paramètres du Raspberry Pi, comme ceux que nous venons de détailler un par un (changer le mot de passe, configurer le réseau, la langue du système, les options de démarrage, etc.)

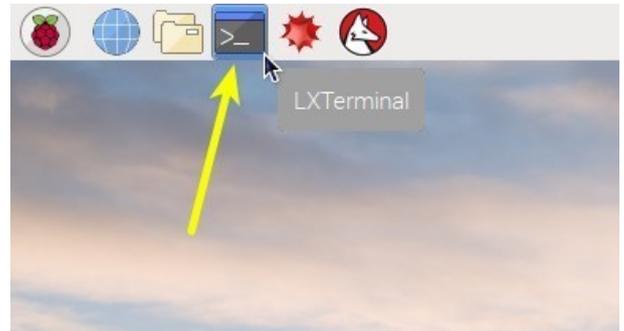
Cet utilitaire de configuration est un peu moins commode que l'application accessible depuis le bureau, dans l'interface graphique du Raspberry : ici, on travaille dans la fenêtre du terminal et la souris est inopérante, il faut tout faire au clavier, avec les flèches de direction et la touche *Entrée* pour valider ou *Échap* pour annuler.

Il faut néanmoins connaître cet utilitaire qui offre plus de possibilités que la fenêtre de configuration classique.

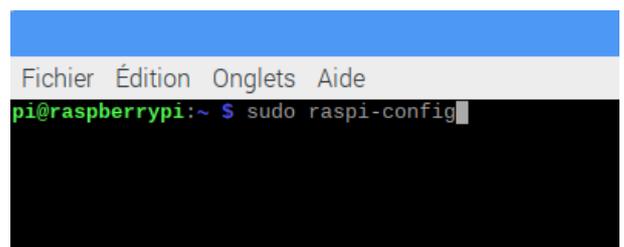


Méthode :

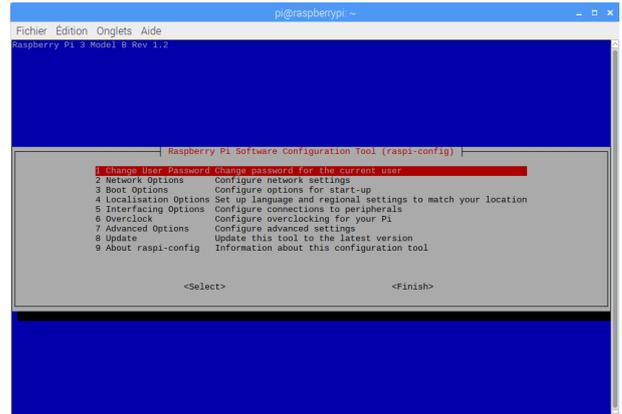
1. Ouvrir un terminal, au moyen de l'icône en haut à gauche.



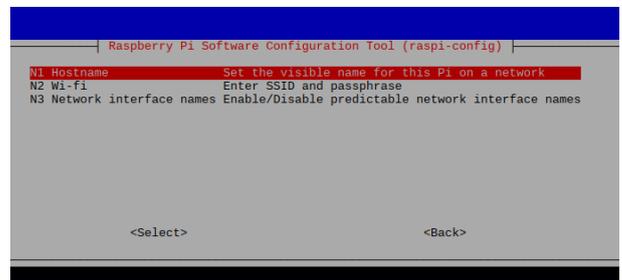
2. Saisir la commande **sudo raspi-config**, valider par la touche Entrée ; saisissez le mot de passe et valider.



3. L'utilitaire de configuration se lance alors dans la fenêtre du terminal (possibilité de la redimensionner à la souris, pour être plus à l'aise).



4. Avec les **flèches de direction** du clavier, la touche **Entrée** et la touche **Échap**, vous pouvez naviguer dans les différentes parties du menu pour activer et configurer certains paramètres.



8. Personnaliser le menu principal des applications

AUTEURS :

- Alain MICHEL

LICENCES :

Creative Commons - Attribution - Partage dans les Mêmes Conditions

L'équivalent du « *Menu Démarrer* » de Windows s'appelle ici le **Menu des applications**. Il est accessible par l'icône en forme de framboise (of course) visible tout en haut à gauche de l'écran.

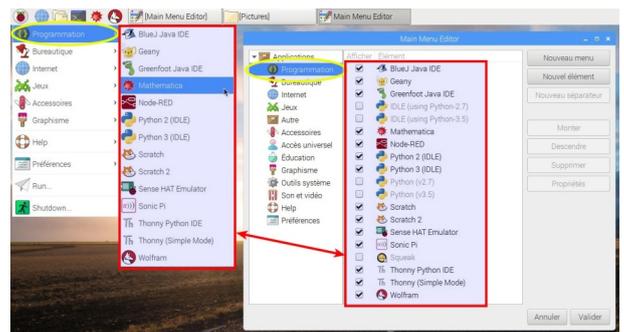


Méthode :

1. Accéder au menu des applications → **Préférences** → **Main Menu Editor** du Raspberry Pi.



2. Dans la fenêtre qui s'ouvre, vous retrouverez toutes les entrées du menu principal du Raspberry Pi et vous pourrez cocher (ou décocher) les cases des logiciels que vous souhaitez voir apparaître (ou pas) dans le menu. Valider pour terminer.



III. Configurations avancées

1. Pour aller un peu plus loin...

Selon l'utilisation qu'on souhaite faire de son Raspberry Pi, il peut être nécessaire de modifier la façon de le lancer ou de l'utiliser, par exemple en le lançant seulement en mode console pour le contrôler à distance depuis un autre ordinateur, ou bien en l'utilisant par exemple comme lecteur multimédia diffusant en continu sur un écran de présentation.

2. Empêcher la mise en veille de l'écran après 10 minutes

AUTEURS :

- Alain MICHEL

LICENCES :

Creative Commons - Attribution - Partage dans les Mêmes Conditions

Rappel :

Par défaut, **le Raspberry Pi est configuré pour éteindre l'écran au bout de 10 minutes d'inactivité** et ne propose pas d'utilitaire d'économiseur d'écran facilement paramétrable, comme sur la plupart des systèmes d'exploitation.

Dans la majorité des cas ce ne sera pas un problème, mais ça peut s'avérer gênant si on souhaite utiliser le Raspberry comme lecteur multimédia, branché sur un écran public, par exemple, pour diffuser en continu des vidéos, des présentations, des diaporamas, etc. C'est un sujet assez important des forums dédiés au Raspberry sur internet, des dizaines de solutions sont proposées et de nombreuses ne fonctionnent pas ; il est très difficile d'y voir clair...

Complément :

On l'a vu plus haut, le Raspberry peut fonctionner en mode « *Desktop* », dans un environnement graphique habituel, ou en mode « *console* », sans environnement graphique, uniquement en ligne de commande. Dans chaque mode de fonctionnement une mise en veille de l'écran après 10 minutes est prévue, et si on a besoin d'empêcher cette mise en veille, **il va falloir le faire pour chacun des deux modes**.

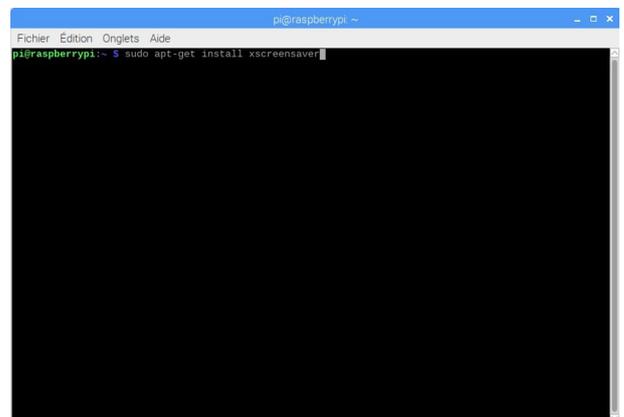
Méthode : En mode classique, « Desktop »

Pour empêcher la mise en veille en mode Desktop, on va tout bêtement **ajouter un économiseur d'écran** puisqu'il n'est pas présent par défaut, et on pourra donc le configurer justement pour empêcher la mise en veille.

1. Ouvrir le terminal et saisir la commande :

sudo apt-get install xscreensaver

puis valider par la touche Entrée et saisir le mot de passe administrateur.



```

pi@raspberrypi ~
Fichier Édition Onglets Aide
pi@raspberrypi:~$ sudo apt-get install xscreensaver

```

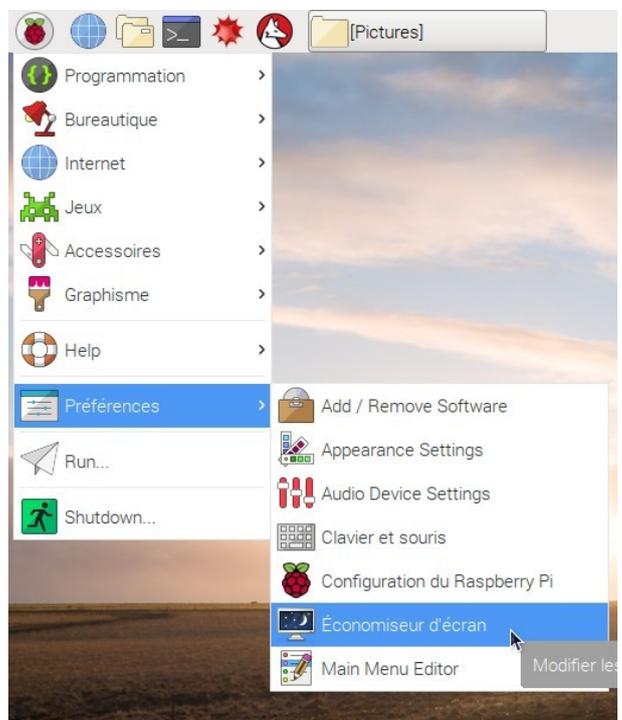
2. Répondre **O** à la question « *Souhaitez-vous continuer ?* » et laisser le téléchargement et l'installation de l'économiseur d'écran se faire.

```

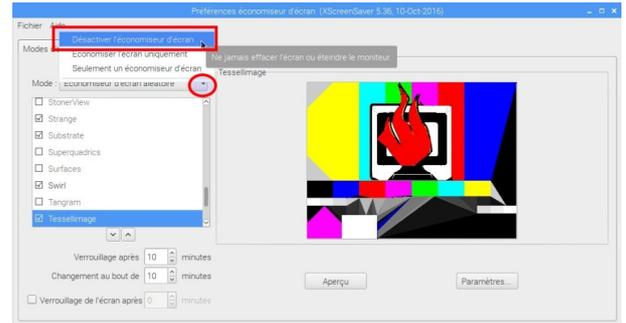
pi@raspberrypi:~$ sudo apt-get install xscreensaver
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances
Lecture des informations d'état... Fait
The following additional packages will be installed:
  libjpeg-turbo-progs libturbojpeg0 miscfiles xscreensaver-data
Paquets suggérés :
  xfbtstank xdeliclock xscreensaver-data-extra xscreensaver-gi xscreensaver-gi-extra fortune qcam
  | streamer gdm3 | kdm-gdmcompat
Paquets recommandés :
  peris
Les NOUVEAUX paquets suivants seront installés :
  libjpeg-turbo-progs libturbojpeg0 miscfiles xscreensaver xscreensaver-data
0 mis à jour, 5 nouvellement installés, 0 à enlever et 0 non mis à jour.
Il est nécessaire de prendre 2 347 ko dans les archives.
Après cette opération, 7 364 ko d'espace disque supplémentaires seront utilisés.
Souhaitez-vous continuer ? [O/n] O
Réception de:1 http://raspbian.42.fr/raspbian stretch/main armhf libturbojpeg0 armhf 1:1.5.1-2 [121 k
B]
Réception de:2 http://raspbian.42.fr/raspbian stretch/main armhf libjpeg-turbo-progs armhf 1:1.5.1-2
[104 kB]
Réception de:3 http://raspbian.42.fr/raspbian stretch/main armhf miscfiles all 1.5+dfsg-2 [1 305 kB]
16% [3 miscfiles 12,0 kB/1 305 kB 1%]

```

3. Dès que c'est terminé, vous pouvez refermer le terminal et aller dans le menu des applications → Préférences → Économiseur d'écran.



Dans la fenêtre qui s'ouvre, il ne reste plus qu'à **choisir de désactiver l'économiseur d'écran** dans la liste déroulante en haut à gauche.



Méthode : En mode console

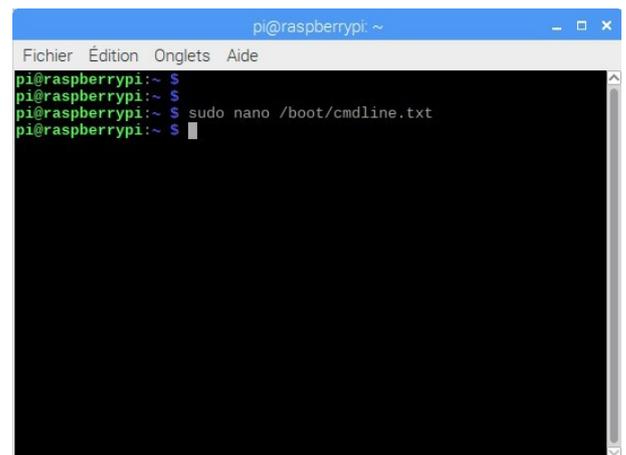
Il faut également empêcher le mode console de désactiver l'écran après 10 minutes.

Pour cela, on va éditer un fichier « sensible » du système d'exploitation : celui qui définit les commandes du démarrage du système.

1. Ouvrir le terminal et saisir la commande :

sudo nano /boot/cmdline.txt

Valider par la touche Entrée puis saisir le mot de passe administrateur.

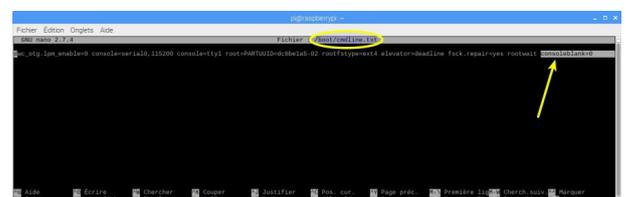


2. Le fichier *cmdline.txt* va s'ouvrir dans l'éditeur intégré, directement dans le terminal.

On va juste **rajouter** l'instruction **consoleblank=0** tout à la fin de la ligne, pour désactiver complètement la mise en veille de la console.

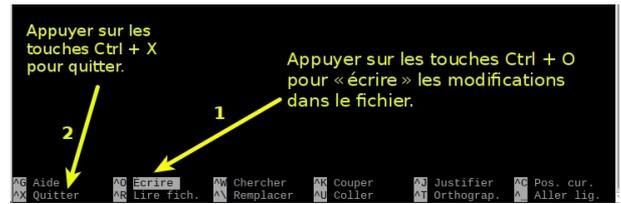
 **Attention :**

- Ce fichier est **ESSENTIEL** pour le démarrage du système, prenez garde à ne rien modifier dans ce qui est déjà en place.
- **NE PAS** ajouter de ligne, la totalité des instructions dans ce fichier **DOIVENT** être écrites **sur une seule ligne**.



4. Enregistrer la modification avec la combinaison de touches **Ctrl + O** au clavier et **valider par la touche Entrée**.
Quitter l'éditeur nano avec la combinaison de touches **Ctrl + X**.

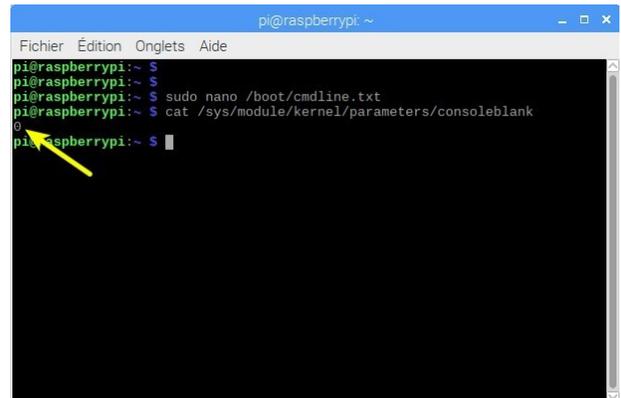
On se retrouve dans la fenêtre du terminal, comme au début.



On peut avoir confirmation que la modification a bien été prise en compte en saisissant dans le terminal la commande suivante :

cat /sys/module/kernel/parameters/consoleblank

Cette commande affiche la valeur attribuée au paramètre *consoleblank* qui détermine la durée en secondes avant la mise en veille : une **valeur égale à 0** signifie la **désactivation de la mise en veille** de la console.



6. Faire redémarrer le Raspberry Pi pour prendre en compte la modification effectuée :
 → saisir dans le terminal la commande **sudo reboot**.

3. Démarrer en mode console

AUTEURS :

- Alain MICHEL

LICENCES :

Creative Commons - Attribution - Partage dans les Mêmes Conditions

Démarrer votre Raspberry Pi en mode console permet de profiter au mieux de ses performances puisqu'elles ne sont pas mobilisées pour faire fonctionner l'interface graphique pour l'utilisateur. La contrepartie est que toutes les actions doivent être faites en ligne de commande, ce qui n'est pas à la portée de tous.

Cependant, ce démarrage en console permet une plus grande sécurité pour votre machine puisque vous serez obligé de saisir votre login et votre mot de passe pour l'ouverture de la session. **Il ne restera plus qu'à lancer manuellement, en fonction des besoins, l'interface graphique.**

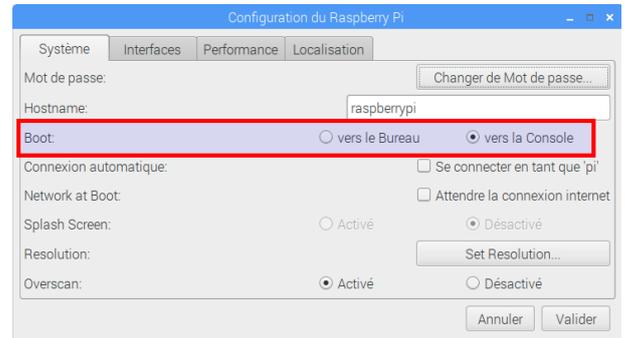
De plus, on l'a déjà évoqué plus haut, ce mode de fonctionnement en console peut suffire si on veut utiliser son Raspberry par exemple en lecteur multimédia branché sur un écran de diffusion, en le commandant depuis un autre ordinateur, via SSH.



Méthode :

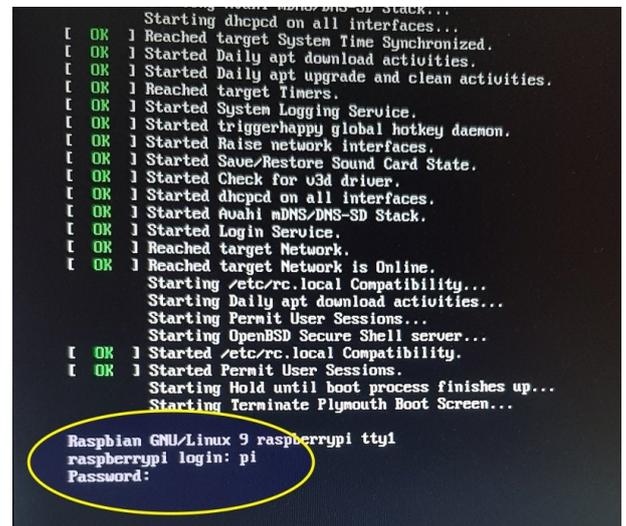
1. Accéder au **menu des applications** → **Préférences** → **Configuration** du Raspberry.

2. Dans la fenêtre qui s'ouvre, sur le 1^{er} onglet « **Système** », choisir l'option **Boot : Vers la console** puis valider.



3. Faire redémarrer le Raspberry Pi : menu des applications → Shutdown → Reboot

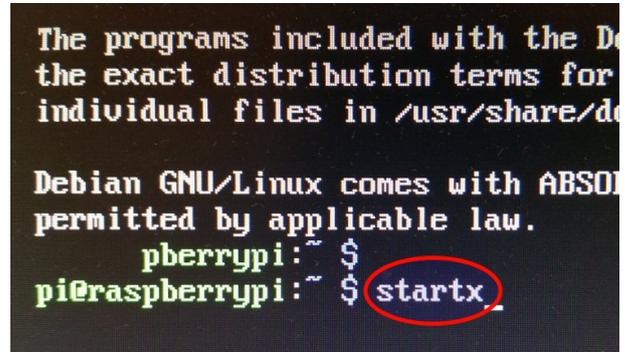
Le démarrage s'interrompt désormais sur une demande login (RAPPEL : le login est **pi**) et de mot de passe (vous avez en principe déjà modifié le mot de passe par défaut *raspberry*).



+ Complément : Lancer manuellement l'interface graphique depuis le mode console

Même si on a démarré en mode console + lignes de commande, on peut à tout moment lancer manuellement l'interface graphique pour se retrouver dans un environnement de bureau habituel.

Pour cela, il suffit de saisir la simple commande **startx** puis de valider par la touche Entrée.



```

The programs included with the De
the exact distribution terms for
individual files in /usr/share/d

Debian GNU/Linux comes with ABSO
permitted by applicable law.
pberrypi:~$
pi@raspberrypi:~$ startx
  
```

On arrive au bout de quelques secondes sur le bureau habituel de l'environnement graphique.

+ Complément : Arrêter le Raspberry depuis le mode console

Si on souhaite arrêter complètement le Raspberry après avoir travaillé en mode console, là aussi cela se fera avec une commande particulière.

→ Il suffit de saisir la commande **sudo halt** et de valider par la touche Entrée.

Le Raspberry va s'éteindre et vous pourrez le débrancher.

4. Connaître l'adresse IP du Raspberry - Attribuer une IP fixe

AUTEURS :

- Alain MICHEL

LICENCES :

Creative Commons - Attribution - Partage dans les Mêmes Conditions



Définition : Connaître l'adresse IP du Raspberry Pi

Pour contrôler le Raspberry Pi à distance via SSH, comme nous le verrons dans le chapitre suivant, il est **nécessaire de connaître son adresse IP** dans notre réseau local.

Rien de plus simple.

Démarrer le Raspberry Pi en mode console ou bien **ouvrir son terminal** si on est dans son interface graphique, et saisir la commande :

ifconfig

(valider par la touche Entrée)

On voit tout de suite l'adresse IP du Raspberry, soit dans la partie *wlan* s'il est connecté en WiFi, soit dans la partie *eth0* s'il est connecté en filaire, par un câble Ethernet RJ45.

```

pi@raspberrypi:~$ ifconfig
eth0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    ether b8:27:eb:7c:77:77 txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1 (Boucle locale)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

wlan0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.13 netmask 255.255.255.0 broadcast 192.168.1.255
    inet6 fe8::c02 prefixlen 64 scopeid 0x20<link>
    ether b8:27:eb:92:77:77 txqueuelen 1000 (Ethernet)
    RX packets 853 bytes 121262 (118.4 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 662 bytes 102943 (100.5 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

pi@raspberrypi:~$

```



Attention :

Cette manipulation devra être exécutée la toute première fois sur le Raspberry Pi , puisqu'on a besoin de son adresse IP pour pouvoir le contrôler via SSH.

Il faudra penser à noter l'adresse IP pour ne pas l'oublier. D'une façon générale, elle ne devrait pas changer par la suite, sauf quelques cas particuliers. En effet, la plupart des box internet des fournisseurs d'accès vont réattribuer la même adresse IP à un périphérique réseau habituel et connu.

MAIS : c'est n'est pas certain et garanti à 100 %. Pour cela, il n'y a qu'un moyen : **attribuer une IP fixe** à votre Raspberry (cf. ci-dessous)



Conseil : Attribuer une IP fixe au Raspberry Pi.

C'est la meilleure solution pour être certain de pouvoir en prendre le contrôle à tout moment en SSH, et pour avoir une meilleure visibilité et compréhension des périphériques connectés de son réseau local.

La **meilleure façon de procéder**, c'est d'utiliser directement les **fonctionnalités proposées par les box des fournisseurs d'accès** à internet qui prévoient cette possibilité d'octroyer une adresse IP fixe à un appareil connecté. Pour cela, il faudra **accéder à l'interface d'administration de votre box**, dans un navigateur internet.

Généralement, il suffit d'ouvrir son navigateur internet et d'accéder à l'adresse **192.168.1.1**



Il faudra éventuellement vous reporter aux guides d'utilisation de votre box pour trouver des explications précises sur la manière de procéder.

D'une façon générale, le principe et le processus sont les mêmes, quelle que soit la box utilisée.

L'exemple ci-dessous est réalisé sur une Livebox v4 de Orange.

1. Dans les paramètres de la box, dans la partie « Réseau », il est possible d'attribuer un bail DHCP statique, c'est à dire une adresse IP fixe, à un de ses équipements.

Je cherche le Raspberry dans la liste des équipements connectés et je lui attribue une adresse IP.

Vous pouvez parfaitement lui attribuer l'adresse IP que vous avez notée au chapitre précédent (*connaître l'adresse IP du Raspberry Pi*).

Baux DHCP statiques

Attribuez vous-même une adresse IP à votre équipement.

raspberrypi 192.168.1.13 B8: : : : :9:92

Équipement	Adresse IP statique	Adresse MAC
alain-PC	192.168.1.21	E: : : : :C0:E0

Baux DHCP dynamiques

Retrouvez tous les équipements dont l'adresse IP est attribuée par le serveur DHCP de votre Livebox.

Équipement	Adresse IP dynamique	Adresse MAC
------------	----------------------	-------------

2. Dès que c'est fait, je vois mon Raspberry apparaître dans la liste de mes équipements réseau ayant une IP fixe.

Baux DHCP statiques

Attribuez vous-même une adresse IP à votre équipement.

France Telecom Liveplug H... BC: : : : :A:D4

Équipement	Adresse IP statique	Adresse MAC
alain-PC	192.168.1.21	E0: : : : :0:E0
raspberrypi	192.168.1.13	B8: : : : :9:92

C'est tout. Votre Raspberry Pi aura désormais toujours la même adresse IP qui lui est réservée dans votre réseau local.



Complément :

Si vous n'avez pas la possibilité d'accéder à l'interface de paramétrage de votre box internet, alors, il faudra modifier certaines choses dans le fichier de configuration du réseau du Raspberry Pi.

Les manipulations à effectuer sont décrites sur cette page :

<https://www.domo-blog.fr/configurer-adresse-ip-static-raspbian-stretch/>

5. Contrôler à distance son Raspberry Pi

AUTEURS :

- Alain MICHEL

LICENCES :

Creative Commons - Attribution - Partage dans les Mêmes Conditions

Le **contrôle à distance de votre Raspberry Pi depuis un autre ordinateur** est particulièrement intéressant dans certaines situations ou utilisations particulières, par exemple si on ne souhaite pas l'utiliser comme un "simple" micro-ordinateur, mais plutôt comme un objet connecté avec des fonctions spécifiques en matière de multimédia ou de domotique.



Attention :

Deux moyens sont à notre disposition pour cette prise de contrôle à distance :

- via **SSH**, uniquement en ligne de commande dans un terminal (le plus « rapide » et efficace dans certains cas, et quand on est un utilisateur assez avancé),
- via **VNC**, avec une interface graphique complète habituelle et prise en charge du clavier et de la souris.

a. Contrôler à distance son Raspberry Pi avec SSH

AUTEURS :

- Alain MICHEL

LICENCES :

Creative Commons - Attribution - Partage dans les Mêmes Conditions



Rappel :

Nous avons vu plus haut comment activer le protocole SSH (donc le **serveur** SSH) sur le Raspberry Pi lors de son premier démarrage. Il faut maintenant, pour pouvoir le contrôler depuis un autre ordinateur installer un **client** SSH sur cet ordinateur. Pour le contrôle via SSH, **nous avons besoin de connaître l'adresse IP** du Raspberry Pi.

Sur un ordinateur fonctionnant sous Linux



Attention :

Un « *client SSH* » est un utilitaire à installer sur son ordinateur qui va permettre de se connecter à un serveur SSH implanté sur un autre ordinateur et d'en prendre le contrôle. Sur un ordinateur sous Linux avec une distribution basée sur Debian, l'installation se fait simplement en une seule ligne, qui combine deux commandes :

sudo apt-get update && sudo apt-get install openssh-client

Cette ligne de commande est à saisir ou copier coller dans le terminal de l'ordinateur destiné à prendre le contrôle à distance du Raspberry,

Il faut valider par la touche Entrée et attendre le téléchargement puis l'installation du client SSH.

(Cette manipulation ne se fait qu'une seule fois, bien entendu).



Méthode : Prendre le contrôle du Raspberry Pi

1. Dès que le client SSH est installé sur votre ordinateur, il suffit d'ouvrir un terminal et de saisir la commande **ssh** suivie des données permettant d'identifier la machine à atteindre, sur votre réseau local, c'est à dire le **nom d'utilisateur** et l'**adresse IP** de votre Raspberry, selon la syntaxe :

ssh <nom_utilisateur>@<adresse_IP>

Dans mon exemple :

ssh pi@192.168.1.13

validé par la touche Entrée et suivi du **mot de passe d'administrateur du Raspberry Pi**.

```
alain@alain-PC: ~
alain@alain-PC:~$ ssh pi@192.168.1.13
```

2. Vous ouvrez alors la **session d'utilisateur sur le Raspberry**, depuis votre ordinateur : vous le contrôlez désormais à distance, depuis le client SSH de votre ordinateur. Toutes les commandes que vous pourrez saisir dans ce terminal seront exécutées par le Raspberry.

```
alain@alain-PC:~$ ssh pi@192.168.1.13
pi@192.168.1.13's password:
Linux raspberrypi 4.9.59-v7+ #1047 SMP Sun Oct 29 12:19:23 GMT 2017 armv7l
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon Dec 25 17:14:52 2017 from 192.168.1.21
pi@raspberrypi:~$
```

3. En fin de session, pour fermer la connexion SSH et arrêter la prise de contrôle à distance, il faut saisir la commande **exit** et valider.

On revient dans le terminal de l'ordinateur.

```
alain@alain-PC:~$ ssh pi@192.168.1.13
pi@192.168.1.13's password:
Linux raspberrypi 4.9.59-v7+ #1047 SMP Sun Oct 29 12:19:23 GMT 2017 armv7l
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon Dec 25 17:14:52 2017 from 192.168.1.21
pi@raspberrypi:~$ exit
^C
Connection to 192.168.1.13 closed.
alain@alain-PC:~$
```

Sur un ordinateur Windows 10



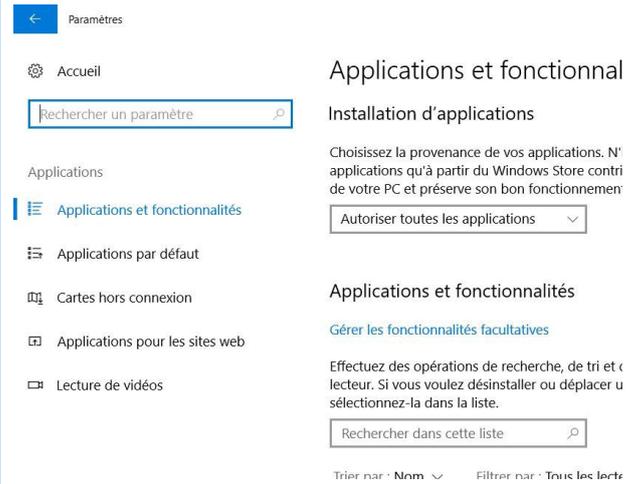
Attention :

Bonne nouvelle pour les ordinateurs fonctionnant sous Windows 10 : ils possèdent désormais un **client OpenSSH intégré**.

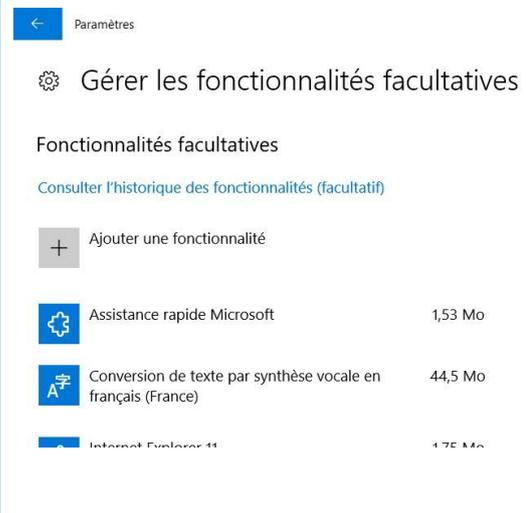


Conseil : Activer le client Open SSH sur Windows 10

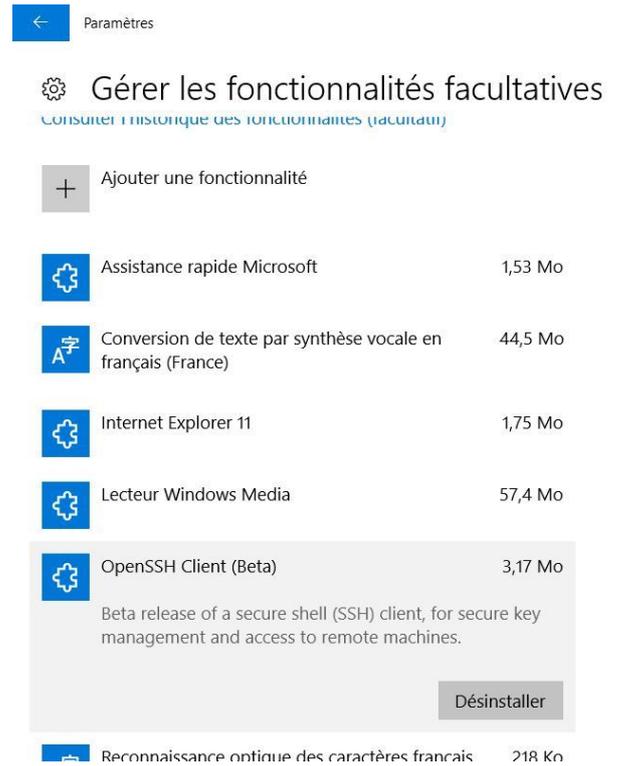
1. Aller dans « Paramètres » → « Applications » → « Applications et fonctionnalités ».



2. Cliquer sur « **Gérer les fonctionnalités facultatives** »

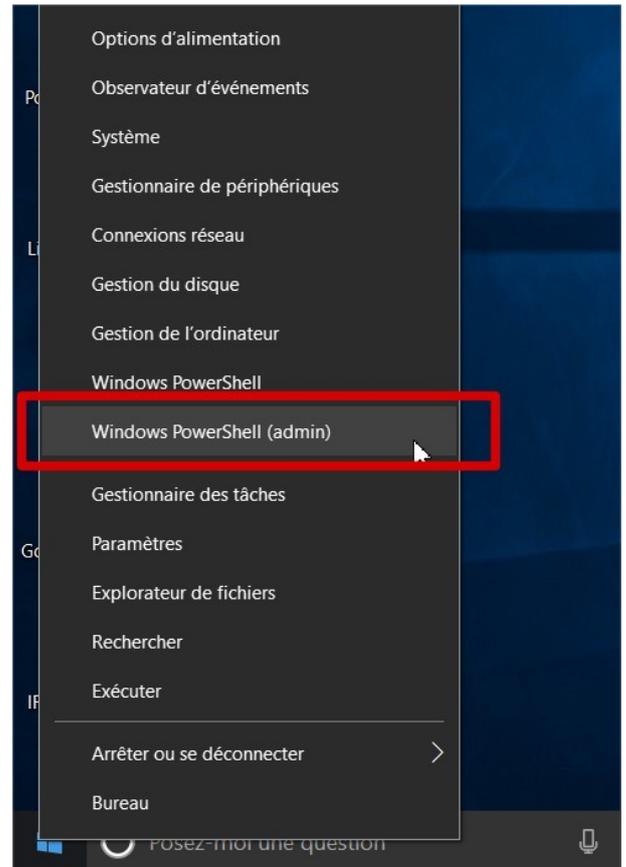


3. Cliquer sur « *Ajouter une fonctionnalité* » et choisir **OpenSSH Client (Beta)** puis l'installer.



Méthode : Contrôler le Raspberry Pi depuis le client OpenSSH de Windows 10

1. Il faut tout d'abord **ouvrir le Windows PowerShell**, c'est dire l'équivalent Windows du terminal Linux. Pour cela, il suffit de faire un clic-droit sur l'icône du menu Démarrer, en bas à droite, et de choisir « *Windows PowerShell (Admin)* » dans la liste.



2. Dans cette fenêtre, nous allons pouvoir saisir la commande qui nous permettra de nous connecter en SSH au Raspberry Pi. C'est la même commande que sous Linux :

ssh <nom_utilisateur>@<adresse_IP>

Dans mon exemple :

ssh pi@192.168.1.13

validé par la touche Entrée et suivi du **mot de passe d'administrateur du Raspberry**.

```

pi@raspberrypi: ~
Windows PowerShell
Copyright (C) Microsoft Corporation. Tous droits réservés.

PS C:\WINDOWS\system32> ssh pi@192.168.1.13
ssh: connect to host 192.168.1.13 port 22: Connection timed out

PS C:\WINDOWS\system32> ssh pi@192.168.1.13
The authenticity of host '192.168.1.13 (192.168.1.13)' can't be established.
ED25519 key fingerprint is SHA256:59pGNN5yn... c0Fc.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.1.13' (ED25519) to the list of known hosts.
pi@192.168.1.13's password:
Linux raspberrypi 4.9.59-v7+ #1047 SMP Sun Oct 29 12:19:23 GMT 2017 armv7l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sun Dec 17 18:47:03 2017 from 192.168.1.21
pi@raspberrypi: ~

```

3. Je suis désormais en train de contrôler le Raspberry depuis mon ordinateur sous Windows ; **toutes les commandes que je peux saisir ici sont des commandes Linux et elles vont s'exécuter sur le Raspberry Pi.**

```

pi@raspberrypi /
Windows PowerShell
Copyright (C) Microsoft Corporation. Tous droits réservés.

PS C:\WINDOWS\system32 ssh pi@192.168.1.13
pi@192.168.1.13's password:
Linux raspberrypi 4.9.59-v7+ #1047 SMP Sun Oct 29 12:19:23 GMT 2017 armv7l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Dec 26 16:37:36 2017 from 192.168.1.15
pi@raspberrypi:~$ sudo apt update
Atteint: http://archive.raspberrypi.org/debian stretch InRelease
Atteint:2 http://raspbian.42.fr/raspbian stretch InRelease
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances
Lecture des informations d'état... fait
All packages are up to date.
pi@raspberrypi:~$ cd /
pi@raspberrypi:~$ ls
bin boot dev etc lib media mnt opt root run sbin src sys tmp
pi@raspberrypi:~$ uname -a
Linux raspberrypi 4.9.59-v7+ #1047 SMP Sun Oct 29 12:19:23 GMT 2017 armv7l GNU/Linux
pi@raspberrypi:~$

```

4. Comme précédemment, quand j'ai terminé ce que j'ai à faire sur le Raspberry, en fin de session, pour **fermer la connexion SSH** et arrêter la prise de contrôle à distance, il faut saisir la commande **exit** et valider. On revient dans le PowerShell de Windows.

SSH autres versions de Windows

Sur toutes les autres Windows, pour pouvoir utiliser SSH, je vais être obligé d'installer un logiciel dédié.

Il en existe plusieurs, le plus connu actuellement est **PuTTY** ; un **logiciel libre** à télécharger depuis le site de son éditeur :

<https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html>

Un guide complet est accessible sur cette page, dans la seconde partie de ce chapitre :

<https://openclassrooms.com/courses/reprenez-le-contrôle-a-l'aide-de-linux/la-connexion-securisée-a-distance-avec-ssh#/id/r-41601>

à partir du titre « *Se connecter via SSH à partir d'une machine Windows* »

b. Contrôler à distance son Raspberry Pi avec VNC

AUTEURS :

- Alain MICHEL

LICENCES :

Creative Commons - Attribution - Partage dans les Mêmes Conditions

Activer VNC sur le Raspberry Pi

De la même manière que pour le protocole SSH qu'on a vu précédemment, **le serveur VNC n'est pas activé par défaut** sur le Raspberry Pi. C'est à l'utilisateur de l'activer, selon ses besoins.

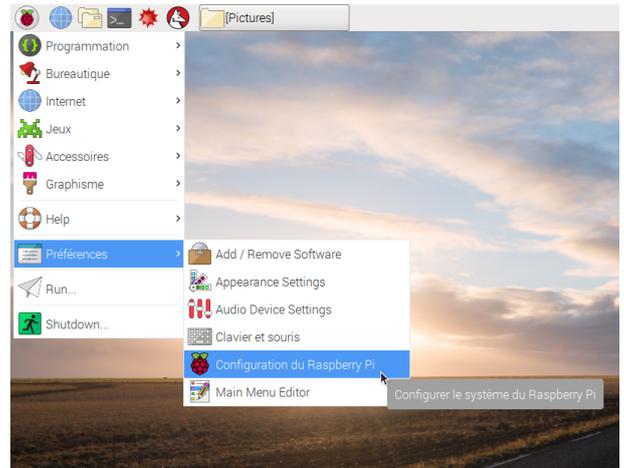
Il y a deux méthodes pour cela :

- soit **depuis la fenêtre de configuration** du Raspberry (en mode Desktop - interface graphique)
- soit avec l'utilitaire **Raspi-Config depuis un terminal** (aussi bien depuis le terminal du Raspberry lui-même, en mode interface graphique, que depuis le terminal d'un ordinateur distant en contrôlant le Raspberry via SSH).



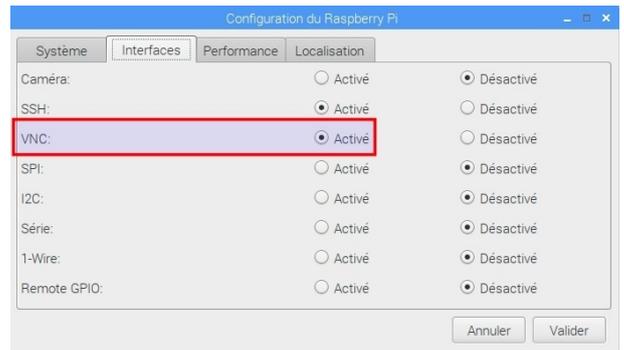
Méthode : Depuis la fenêtre de configuration du raspberry

1. Accéder au menu des applications → **Préférences** → **Configuration** du Raspberry Pi.



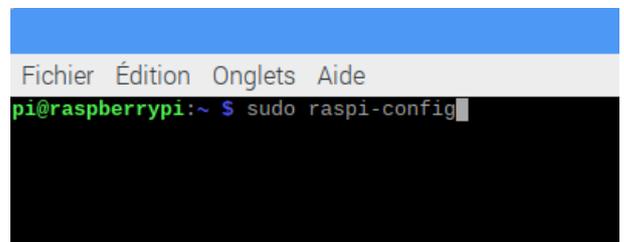
2. Dans la fenêtre qui s'ouvre (que vous pouvez redimensionner à la souris), accéder au 2^e onglet **Interfaces**.

- Cocher le bouton « **Activé** » sur la troisième ligne **VNC**.
- Valider



Méthode : Depuis un terminal, avec l'utilitaire raspi-config (méthode alternative)

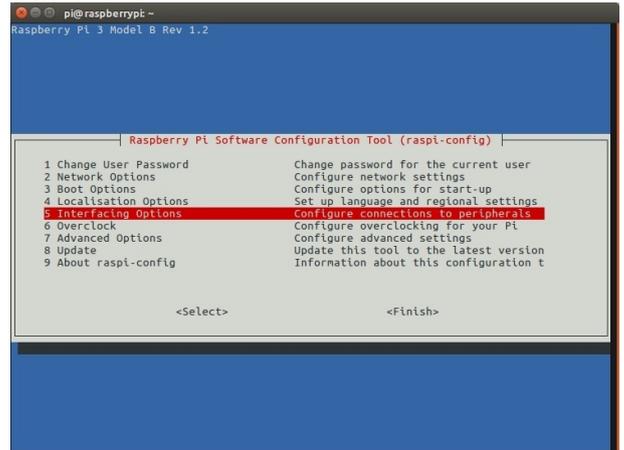
1. Saisir la commande **sudo raspi-config**, valider par la touche Entrée ; saisissez le mot de passe et valider.



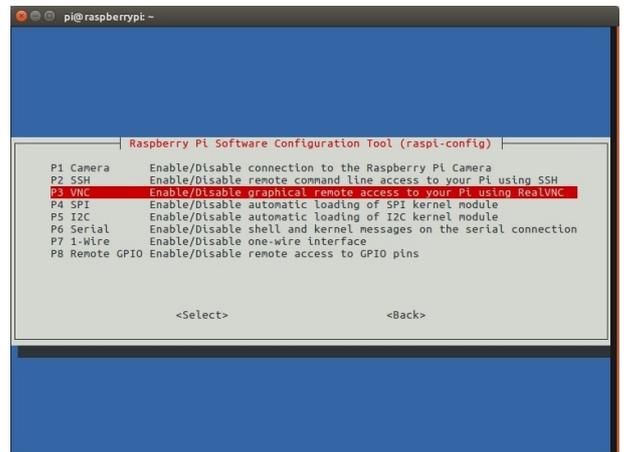
2. L'utilitaire de configuration se lance alors dans la fenêtre du terminal (possibilité de la redimensionner à la souris, pour être plus à l'aise).

Avec les **flèches de direction** du clavier, la touche **Entrée** et la touche **Échap**, vous pouvez naviguer dans les différentes parties du menu pour activer et configurer certains paramètres.

Se positionner sur la 5^e ligne « Interfacing options » et valider par la touche Entrée.

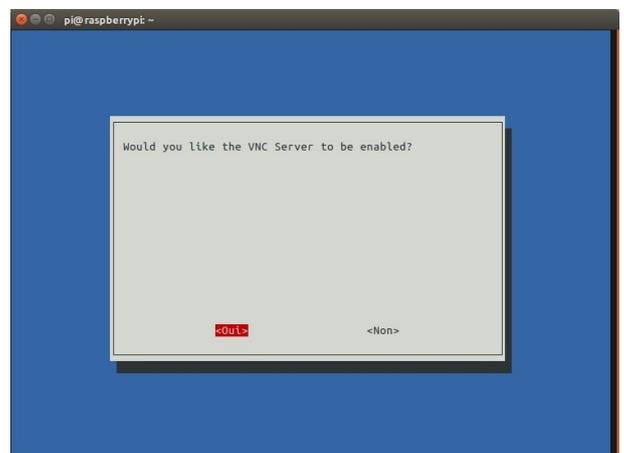


3. Aller sur la 3e ligne « P3 VNC » et valider par Entrée.



4. Avec la touche tabulation ou les flèches de direction du clavier, se positionner sur « **Oui** » et valider avec la touche Entrée.

Le serveur VNC est désormais activé.



5. Avec la touche tabulation ou les flèches de direction du clavier, se positionner sur « **Finish** » et valider avec la touche Entrée. Refermer le terminal.



Attention :

Le **serveur VNC** du Raspberry Pi est désormais activé. Il est prêt à fonctionner.

Pour pouvoir s'y connecter depuis un autre ordinateur et le contrôler à distance, il va falloir ensuite installer sur cet ordinateur distant un *client VNC*. (Cf. plus bas).

→ Il faut auparavant vérifier s'il est bien configuré pour démarrer en mode Bureau.

Configurer le Raspberry Pi pour démarrer en mode Desktop



Attention :

Pour prendre le contrôle du Raspberry Pi depuis un autre poste avec VNC, il faut que ce Raspberry ait démarré de façon « classique », **en mode Desktop**, c'est à dire **avec une interface graphique** (en non pas en mode console, avec seulement la possibilité de saisir des lignes de commande).

On peut **configurer ce type de démarrage de deux façons** :

- soit **depuis la fenêtre de configuration** du Raspberry (en mode Desktop - interface graphique)
- soit avec l'utilitaire **Raspi-Config depuis un terminal** (aussi bien depuis le terminal du Raspberry Pi lui-même, en mode interface graphique, que depuis le terminal d'un ordinateur distant en contrôlant le Raspberry via SSH).



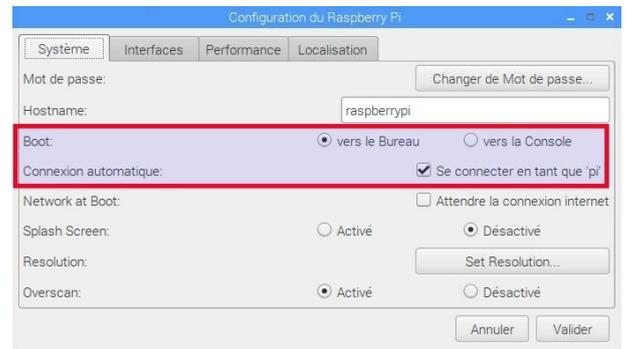
Méthode : Depuis la fenêtre de configuration du Raspberry Pi

1. Accéder au **menu des applications** → **Préférences** → **Configuration** du Raspberry.

2. Dans la fenêtre qui s'ouvre, sur le 1^{er} onglet « *Système* » :

- choisir l'option **Boot : Vers le bureau**
- **cocher la case** « *Connexion automatique : se connecter en tant que 'pi'* »

puis valider.

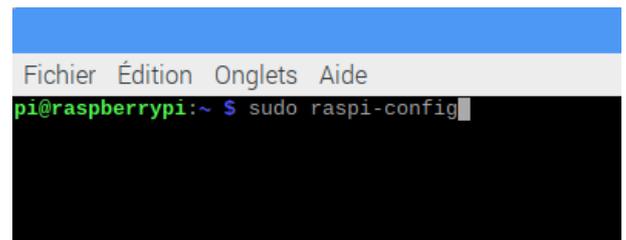


3. Faire redémarrer le Raspberry Pi : menu des applications → Shutdown → Reboot



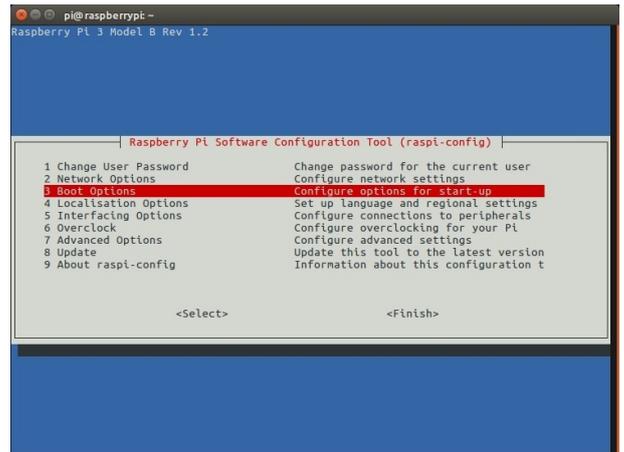
Méthode : Depuis un terminal avec l'utilitaire Raspi-Config (méthode alternative)

1. Saisir la commande **sudo raspi-config**, valider par la touche Entrée ; saisissez le mot de passe et valider.



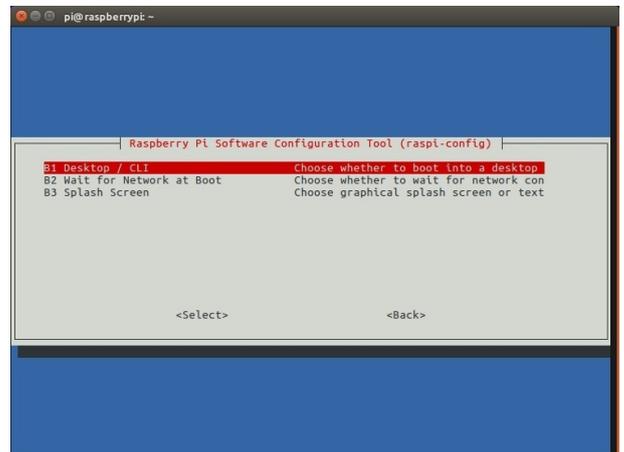
2. L'utilitaire de configuration se lance alors dans la fenêtre du terminal (possibilité de la redimensionner à la souris, pour être plus à l'aise). Avec les **flèches de direction** du clavier, la touche **Entrée** et la touche **Échap**, vous pouvez naviguer dans les différentes parties du menu pour activer et configurer certains paramètres.

Se positionner sur la 3^e ligne « *Boot options* » et valider par la touche **Entrée**.



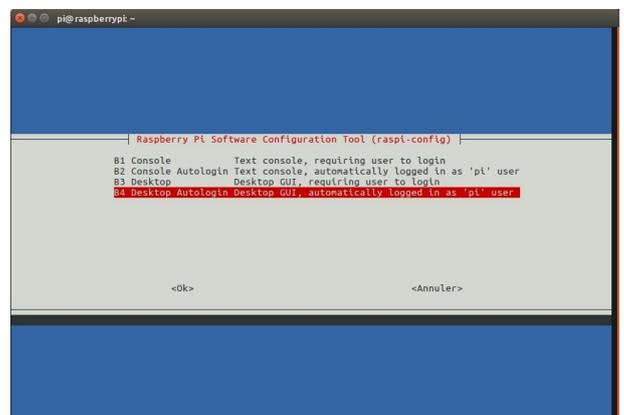
3. Rester sur la 1^{ère} ligne « *B1 Desktop / CLI* » et valider par **Entrée**.

On va ensuite choisir que le Raspberry démarre en mode Bureau (Desktop) plutôt qu'en mode console (en ligne de commande - Command Line Interface).



4. Avec les flèches de direction du clavier, se positionner sur la 4^e ligne « *B4 Desktop - Autologin Desktop GUI, automatically logged in as 'pi' user* » puis sur « **OK** » avec la touche **Tab** du clavier, et valider avec la touche **Entrée**.

Il faut ensuite **redémarrer le Raspberry**.



Il est maintenant configuré pour démarrer en mode « *Desktop GUI* » c'est à dire en **mode bureau avec interface graphique pour l'utilisateur** (Graphical User Interface) et il ouvrira automatiquement une session sans avoir à saisir de mot de passe.

→ Votre Raspberry Pi est maintenant prêt à être commandé à distance via VNC, depuis un client VNC installé sur un autre ordinateur.

Prendre le contrôle du Raspberry Pi avec un client VNC

Définition :

Un « **client VNC** » est un utilitaire à installer sur son ordinateur qui va permettre de se connecter à un « **serveur VNC** » implanté sur un autre ordinateur dont on veut prendre le contrôle.

Attention :

Il existe de nombreux logiciels clients VNC différents. Celui qu'on installera ici sur son ordinateur est choisi pour être pleinement compatible avec le serveur VNC présent sur le Raspberry Pi. Ce client VNC est celui de **RealVNC** ; il s'installe sur tout type de plateformes et il est gratuit pour une utilisation personnelle.

On le téléchargera à partir de cette page :

<https://www.realvnc.com/en/connect/download/viewer/>

Il faut ensuite procéder à son installation sur votre ordinateur, selon un processus classique.



Méthode : Prendre le contrôle du Raspberry avec un client VNC

1. Dès que le client VNC est installé sur votre ordinateur, il suffit de le lancer. Dans la fenêtre qui s'ouvre, il faut **saisir l'adresse IP du Raspberry** dans votre réseau local.

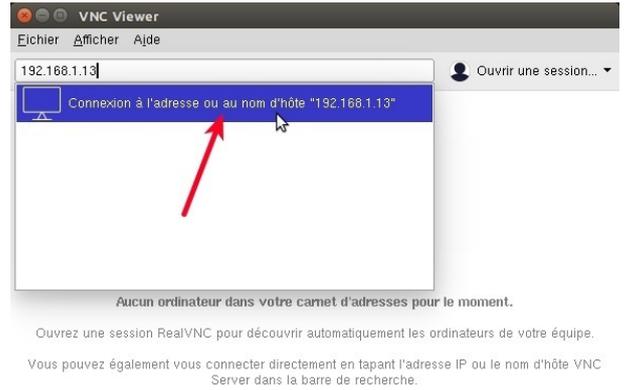
RAPPEL : il s'agira de préférence d'une **adresse IP fixe** qu'on lui aura attribuée auparavant (cf. chapitre précédent : « *Connaître l'adresse IP du Raspberry Pi - Attribuer une IP fixe* » Dans mon exemple : **192.168.1.13**)

NB : Vous pouvez ignorer le bouton « *Ouvrir une session* » destiné à la création d'un compte utilisateur sur le site internet de Real VNC.



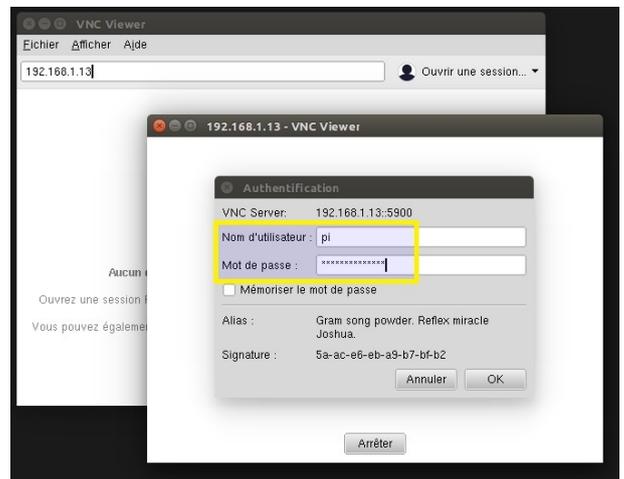
2. Si l'adresse saisie existe dans votre réseau local, le client VNC fait apparaître en surbrillance une ligne « Connexion à l'adresse ou au nom d'hôte "192.168.1.13" ».

Il suffit de cliquer sur cette ligne en surbrillance pour lancer la demande de connexion à distance au Raspberry.

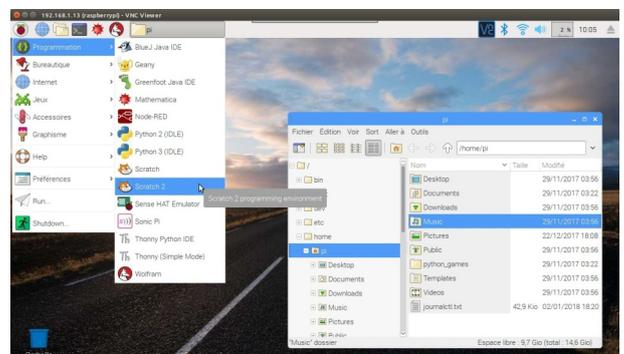


3. Une nouvelle fenêtre s'ouvre. Il va falloir saisir le **nom d'utilisateur** et le **mot de passe d'ouverture de session sur le Raspberry** puis valider par OK.

RAPPEL : le mot de passe par défaut est *raspberry* mais vous l'avez très certainement déjà modifié (fortement conseillé !) - cf. chapitre « *Premières configurations - Modifier le mot de passe* »



4. La connexion avec le Raspberry s'établit et une fenêtre s'ouvre, montrant directement le bureau de votre Raspberry dont vous avez le **contrôle total en mode Desktop - interface graphique**.

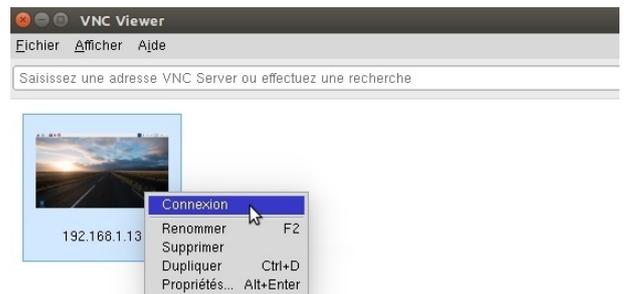


5. À la fin de votre session de travail à distance, vous allez refermer la connexion VNC au moyen de l'icône à droite de la barre d'icônes de RealVNC présente en haut de l'écran.



6. On retrouve la fenêtre initiale du client VNC. On y voit maintenant apparaître une miniature de la connexion qui vient d'être établie, l'adresse IP et le nom d'utilisateur seront mémorisés.

Pour se reconnecter, il n'y aura qu'à faire un clic-droit sur cette miniature et à cliquer sur la ligne « Connexion » et saisir le mot de passe d'utilisateur du Raspberry Pi.



6. Configurer son Raspberry entièrement à distance avec SSH

AUTEURS :

- Alain MICHEL

LICENCES :

Creative Commons - Attribution - Partage dans les Mêmes Conditions

Rappel :

Nous avons vu plus haut [comment contrôler le Raspberry Pi depuis un autre ordinateur](#) au moyen d'un **client** SSH sur cet ordinateur.

Nous allons voir ici comment préparer et configurer son Raspberry **entièrement à distance**, depuis un autre ordinateur, avec l'avantage pratique de ne **même pas avoir besoin d'un clavier ni d'un écran dédié**. Pour cela, deux opérations distinctes seront à réaliser AVANT le tout premier démarrage du Raspberry, directement dans les fichiers sur la carte micro-SD :

- activer le **protocole SSH**
- attribuer une **adresse IP fixe**

Toutes les manipulations décrites ci-dessous ont été faites sur un ordinateur fonctionnant sous Linux Ubuntu 20.04

a. Activer le protocole SSH directement sur la carte micro-SD



Hypothèse :

Situation de départ : nous venons donc de copier l'image du système Raspbian sur notre carte micro-SD.

Elle est prête à être insérée dans notre Raspberry pour son premier démarrage, comme on l'a vu plus haut dans le premier chapitre (premier démarrage).

Tout le problème consiste à pouvoir activer le protocole SSH **avant** d'insérer la carte dans le Raspberry et de l'allumer, si on veut pouvoir le contrôler à distance dès son premier démarrage.

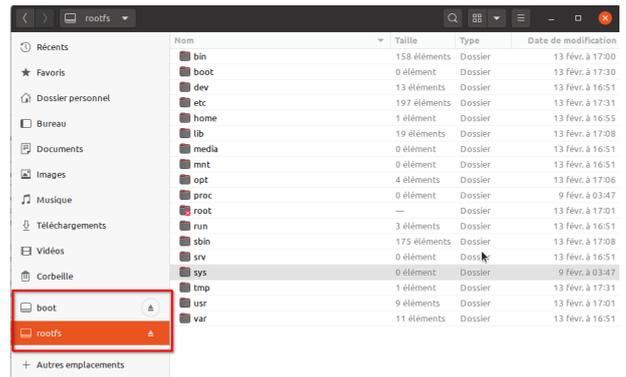


Méthode :

1. Brancher à nouveau la carte micro-SD sur l'ordinateur qui a servi à la préparer, avec *Raspberry Pi Imager*.

Le système va la détecter et afficher deux partitions :

- **boot** qui contient quelques fichiers
- **rootfs** qui contient l'arborescence du système Raspbian Buster installé



Les deux partitions visibles sur la carte micro-SD après installation du système Raspbian



Pour activer le SSH sur votre Raspberry, il suffit de **créer un nouveau fichier** nommé **ssh** dans la partition **boot**.
C'est tout !

2. Nous allons donc ouvrir un éditeur de texte pour créer un nouveau fichier – que nous allons laisser entièrement vide) et qu'on va « *Enregistrer sous...* » le nom **ssh** dans la partition **bootfs**.

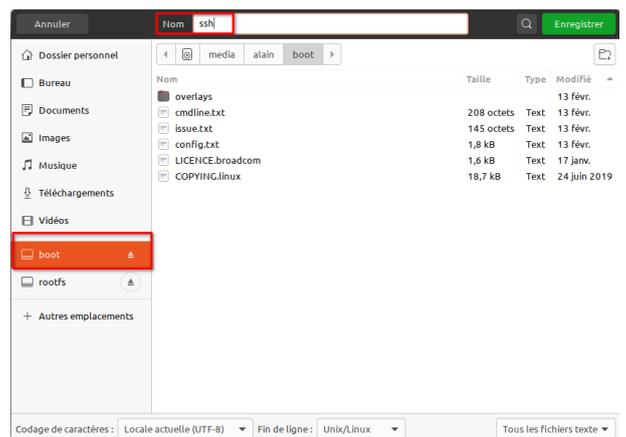
Attention :

- il doit rester entièrement vide
- on ne met **aucune extension de fichier** à son nom : il s'appelle « **ssh** » – tout court – pas de **ssh.txt** ou autre !

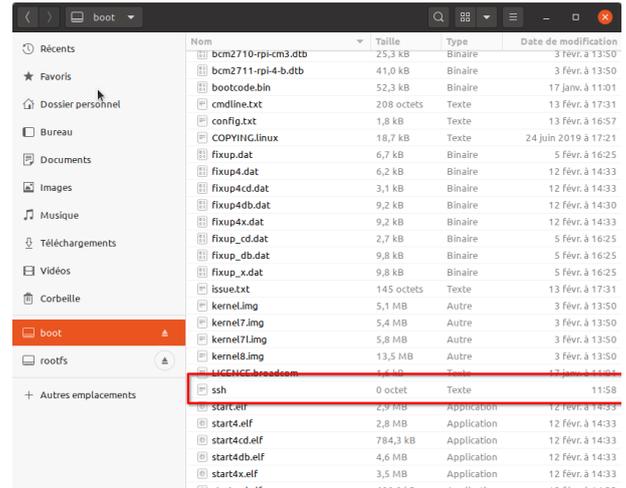


Cette création du fichier **ssh** peut également être faite en ligne de commande, dans un terminal, au moyen de la commande **touch** en saisissant la commande suivante (adaptée à mon exemple) :

touch /home/alain/media/boot/ssh



3. Quand c'est fait, nous voyons ce fichier dans la liste des contenus de la partition boot.



Ça y est, cette première étape est terminée, le protocole SSH sera fonctionnel sur le Raspberry dès son premier démarrage, nous permettant ainsi d'en prendre tout de suite le contrôle depuis un autre ordinateur.

b. Attribuer une adresse IP fixe directement sur la carte micro-SD

Conseil :

Pour pouvoir contrôler le Raspberry à distance via SSH, la manipulation est bien plus aisée s'il a une **adresse IP fixe** dans notre réseau local.

Il est possible de le faire avant même son premier démarrage, en modifiant un des fichiers système, sur la carte micro-SD.

Méthode :

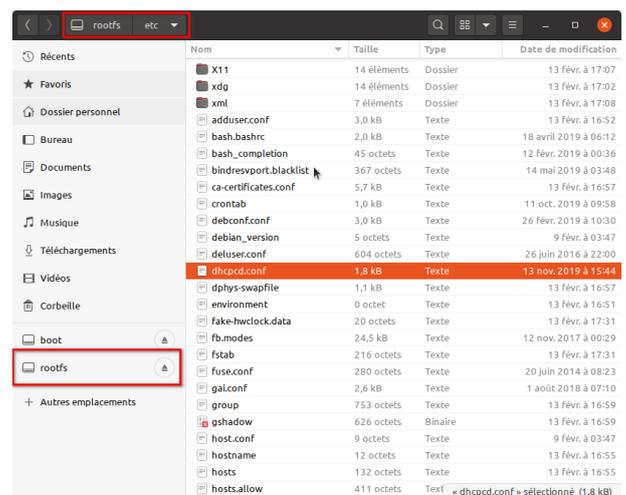
1. Brancher à nouveau la carte micro-SD sur l'ordinateur qui a servi à la préparer.

Il va falloir modifier un fichier qui se trouve dans l'arborescence du système Raspbian, sur la partition rootfs.

Ce fichier s'appelle **dhcpcd.conf** et il se trouve dans le répertoire **/etc**

Il ne peut pas être modifié en l'ouvrant directement dans l'éditeur de texte de votre ordinateur étant donné que l'utilisateur standard de l'ordinateur n'a pas les droits de modification sur ce fichier, on ne l'aura qu'en lecture seule si on procède de la sorte.

Il va donc falloir l'ouvrir via le terminal, avec les droits administrateur.



Le fichier à modifier pour attribuer une adresse IP fixe au Raspberry

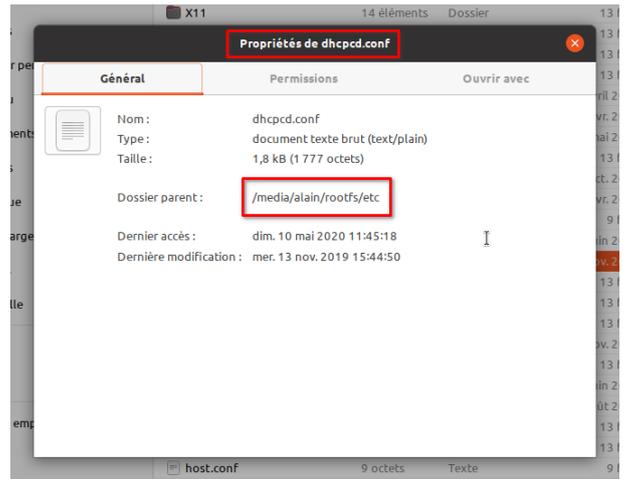
2. Chercher le chemin d'accès au fichier

Pour l'ouvrir avec les droits administrateur, via le terminal, on a besoin de connaître le chemin d'accès à ce fichier : un clic-droit → propriétés nous renseigne à ce sujet.

Le chemin d'accès au fichier à utiliser ensuite dans le terminal sera :

/media/alain/rootfs/etc/dhcpd.conf

NB : ce chemin commence par /media/alain car la partition rootfs de la carte micro-SD est pour le moment ouverte sur l'ordinateur qui a servi à la préparer, elle est donc montée dans le répertoire /media de l'utilisateur, comme une simple clé USB ou un disque externe.



Chemin vers le fichier à modifier

3. Éditer le fichier via le terminal

On peut donc maintenant ouvrir le terminal (raccourci clavier Ctrl + Alt + T) et saisir la commande suivante pour éditer le fichier en question avec gedit qui est l'éditeur de texte par défaut sur le système Linux Ubuntu :

sudo gedit /media/alain/rootfs/etc/dhcpd.conf

Dès qu'on valide par entrée, et après avoir saisi le mot de passe administrateur, le fichier dhcpd.conf va s'ouvrir dans l'éditeur de texte, mais cette fois-ci avec les droits suffisants pour le modifier.

NB : il est possible d'éditer le fichier directement dans le terminal avec l'éditeur **nano** (un peu moins simple à utiliser, la souris n'est pas active, il faut passer uniquement par le clavier).

- pour enregistrer les modifications : combinaison de touches Ctrl + O au clavier et valider par la touche Entrée.
- pour quitter ensuite l'éditeur nano : combinaison de touches Ctrl + X.



4. Trouver la partie à modifier

Rechercher la partie « *# Example static IP configuration* » (exemple de configuration d'adresse IP fixe)

C'est dans cette partie qu'il va falloir apporter quelques modifications.

NB : toutes ces lignes commencent par le caractère # ce qui fait qu'elles sont « *inactives* », elles sont considérées comme des commentaires dans le fichier, à ne pas exécuter.

Il va falloir « dé-commenter » 3 de ces lignes, c'est à dire **enlever le caractère #** pour les rendre actives, et/ou les modifier.

```
42
43 # Example static IP configuration:
44 #interface eth0
45 #static ip_address=192.168.0.10/24
46 #static ip6_address=fd51:42f8:caae:d92e::ff/64
47 #static routers=192.168.0.1
48 #static domain_name_servers=192.168.0.1 8.8.8.8 fd51:42f8:caae:d92e::1
49
50 # To be replaced by the static IP address of the DHCP server
```

Partie à modifier

5. Modifier le fichier

Il faut dé-commenter les 3 lignes en jaune :

- **interface eth0** (pas d'autre modification)
- **static ip_address=192.168.1.60** → je modifie ce qui existe pour renseigner l'adresse IP que je souhaite attribuer au Raspberry, ici 192.168.1.60 pour moi.
- **static routers=192.168.1.1** → je modifie ce qui existe pour renseigner l'adresse IP de mon routeur (passerelle).

```
42
43 # Example static IP configuration:
44 interface eth0
45 static ip_address=192.168.1.60
46 #static ip6_address=fd51:42f8:caae:d92e::ff/64
47 static routers=192.168.1.1
48 #static domain_name_servers=192.168.0.1 8.8.8.8 fd51:42f8:caae:d92e::1
49
50 # It is possible to fall back to a static IP if DHCP fails.
```

Lignes modifiées dans le fichier dhcpd.conf

NB 1 : **eth0** est le nom de l'interface réseau Ethernet pour la **connexion filaire** du Raspberry. Dans un premier temps, c'est la seule active et c'est celle qu'on utilisera lors du premier démarrage du Raspberry ; la configuration initiale est plus simple en filaire, on se préoccupera du WiFi plus tard si nécessaire.

NB 2 : Il faut évidemment que l'adresse IP choisie pour le Raspberry respecte le plan d'adressage des adresses utilisées par mon routeur. Dans mon exemple, j'ai un routeur Orange (Livebox) dont l'adresse (passerelle) est 192.168.1.1 et qui fournit des adresses IP du genre 192.168.1.**XX** pour mon réseau local.

NB 3 : il est préférable de choisir une adresse IP qui soit **en dehors de la plage d'adresses** attribuées automatiquement et de manière dynamique par le serveur DHCP de votre box.

Par exemple pour ma box, cette plage d'adresses contient toutes les adresses entre 192.168.1.**10** et 192.168.1.**50** :

Retour Réseau

DHCP NAT/PAT DNS UPnP DynDNS DMZ NTP IPv6

Le serveur DHCP de votre Livebox attribue automatiquement une adresse IP à chaque équipement de votre réseau local.

Uniquement pour des équipements IPv4.

Paramètres du serveur DHCP

Activer le serveur DHCP

Adresse IP de votre Livebox: 192.168.1.1

Masque de sous-réseau du LAN: 255.255.255.0

Adresse IP de début: 192.168.1.10

Adresse IP de fin: 192.168.1.50

Annuler Enregistrer

Plage d'adresses IP potentiellement octroyées par la box.

En choisissant l'adresse 192.168.1.**60** pour le Raspberry, je suis donc bien en dehors de cette plage d'adresses et je ne risque pas un éventuel conflit d'adresse IP avec une nouvelle machine qui se connecterait sur mon réseau local.

6. Enregistrer les modifications et quitter

- Je m'assure bien d'avoir enlevé le signe # au début de ces 3 lignes, je ne touche pas aux autres lignes.
- J'enregistre les modifications : Ctrl + S (ou bien Menu → Enregistrer)
- Je referme la fenêtre de l'éditeur de texte gedit
- Je peux également maintenant refermer le terminal

c. Premier démarrage et premières configurations entièrement via SSH



Hypothèse :

Situation de départ : nous venons d'activer SSH et d'attribuer une adresse IP fixe au Raspberry en intervenant directement sur les fichiers, sur la carte micro-SD.

Elle est prête à être insérée dans notre Raspberry pour son premier démarrage que nous contrôlerons entièrement à distance, via SSH.



Conseil : Branchements préalables nécessaires

- La carte micro-SD est maintenant insérée dans son logement sur le Raspberry Pi.
- un câble Ethernet est branché sur la prise correspondante sur le Raspberry et le relie au routeur (Box d'accès à internet).
- Le cordon d'alimentation est branché sur le Raspberry et l'adaptateur secteur est prêt à être branché sur une prise électrique.

Il n'y a **aucun clavier, aucune souris, aucun écran** relié au Raspberry.



Rappel :

Dans mon exemple, toutes les manipulations décrites sont faites depuis un ordinateur fonctionnant sous Linux Ubuntu 20.04. Elles peuvent également être faites depuis un ordinateur fonctionnant sous Windows 10, via le « *Windows Power Shell* ». À ce sujet, on se reportera au [chapitre correspondant](#).

Première connexion SSH



Méthode :

1. Branchement

Brancher l'adaptateur secteur pour alimenter le Raspberry et lancer son premier démarrage. Patienter quelques secondes.

2. Établissement d'une connexion SSH avec le Raspberry

Ouvrir le terminal sur l'ordinateur qui servira à contrôler le Raspberry.

Saisir la commande :

ssh pi@192.168.1.60

Alerte de sécurité obtenu à la toute première tentative de connexion SSH vers le Raspberry

- ssh... pour utiliser le protocole ssh
- pi : c'est le nom de l'utilisateur par défaut sur le Raspberry
- @192.168.1.60 : pour indiquer sur quelle machine, à quelle adresse IP la connexion SSH doit être établie.

On obtient une alerte de sécurité nous prévenant que ce périphérique auquel on tente de se connecter n'est pas encore connu et authentifié sur notre machine. On va l'ajouter en répondant « **yes** » et en validant par la touche **Entrée**.

3. Connexion avec le Raspberry Pi

Après avoir validé la commande à l'étape précédente, le mot de passe d'utilisateur du Raspberry est demandé pour établir la connexion SSH.

ATTENTION : à ce stade, c'est le mot de passe par défaut de l'utilisateur par défaut du Raspberry qu'il faudra saisir.

→ Ce mot de passe est : **raspberry**

Il faudra bien évidemment le modifier dès qu'on sera connecté avec le Raspberry.

Connexion SSH avec le Raspberry Pi

→ **IMPORTANT** : sur tous les systèmes Linux, **quand on saisit un mot de passe dans le terminal**, pour des raisons de sécurité, **RIEN n'apparaît**, on ne voit pas les caractères saisis, ni même des petites étoiles ou des petits cercles, comme on est habitué à voir sur d'autres systèmes. **C'est normal**.

La dernière ligne visible sur le terminal nous montre bien que nous sommes maintenant aux commandes du Raspberry : **pi@raspberrypi : ~ \$** (et non plus aux commandes de l'ordinateur qui exécute la fenêtre du terminal par lequel on a pris le contrôle du Raspberry).

Nous sommes désormais l'utilisateur « *pi* » sur la machine « *raspberrypi* ».

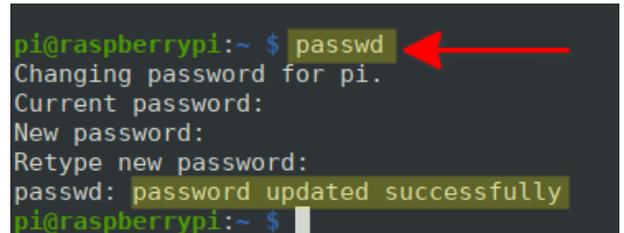
Changement du mot de passe par défaut

On va tout suite modifier le mot de passe par défaut pour accéder au Raspberry en tant qu'utilisateur « pi » :

- Dans le terminal, saisir la commande : **passwd**

- Saisir le mot de passe actuel (par défaut) : raspberry
- Saisir le nouveau mot de passe
- Confirmer le nouveau mot de passe

Le mot de passe est alors modifié.



```
pi@raspberrypi:~ $ passwd
Changing password for pi.
Current password:
New password:
Retype new password:
passwd: password updated successfully
pi@raspberrypi:~ $
```

Changement du mot de passe par défaut, via SSH

C'est ce nouveau mot de passe qu'on utilisera désormais pour toutes les autres actions à effectuer par la suite.

Exécuter l'utilitaire raspi-config

Tout un ensemble de paramètres, notamment la langue, la région, l'heure, etc. peuvent être configurés depuis un utilitaire dédié qui s'exécute dans le terminal.

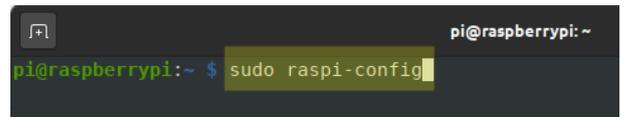
Cet utilitaire s'appelle **raspi-config**, on va l'exécuter en tant qu'administrateur pour effectuer les réglages indispensables.

Méthode : Configuration avec raspi-config

1. Lancer raspi-config

On va lancer dans le terminal l'interface de configuration raspi-config en tant qu'administrateur avec la commande :

sudo raspi-config

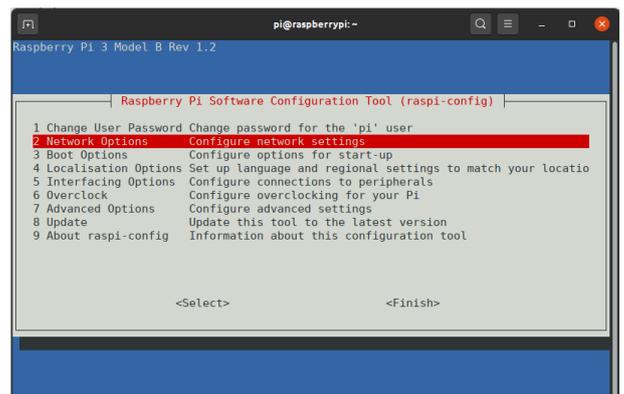


```
pi@raspberrypi:~ $ sudo raspi-config
```

2. Navigation et validation des choix

On navigue entre les différentes options des listes avec les flèches haut et bas du clavier.

On accède aux choix « *Select* » ou « *Finish* » ou « *OK* » ou « *Cancel* » avec les flèches gauche et droite du clavier, ou bien avec la touche Tabulation



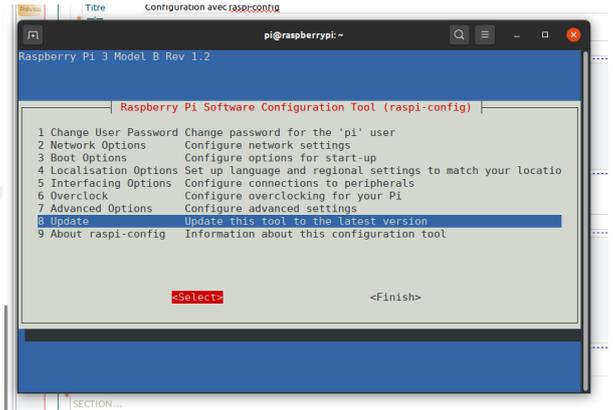
```
Raspberry Pi Software Configuration Tool (raspi-config)
1 Change User Password Change password for the 'pi' user
2 Network Options Configure network settings
3 Boot Options Configure options for start-up
4 Localisation Options Set up language and regional settings to match your locatio
5 Interfacing Options Configure connections to peripherals
6 Overclock Configure overclocking for your Pi
7 Advanced Options Configure advanced settings
8 Update Update this tool to the latest version
9 About raspi-config Information about this configuration tool

<Select> <Finish>
```

Interface de l'utilitaire de configuration raspi-config

3. Mise à jour de l'utilitaire raspi-config

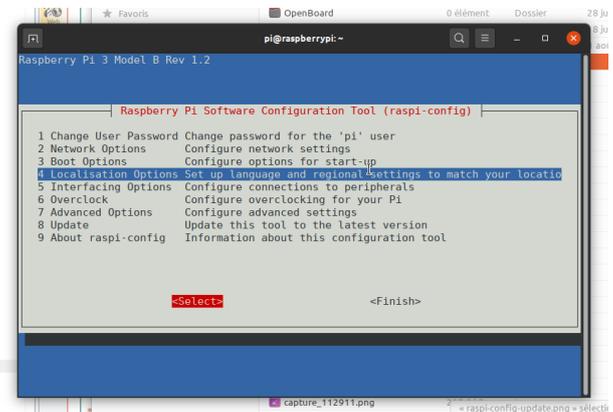
Première des choses à faire : **mettre à jour l'utilitaire raspi-config lui-même** pour être certains d'utiliser la dernière version. Pour ça, on se dirigera vers l'option 8 du menu principal. On valide, on laisse s'exécuter le processus, puis raspi-config est relancé.



Mise à jour de l'utilitaire de configuration

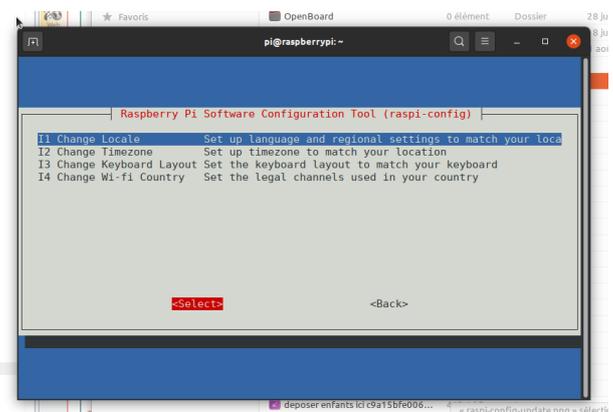
4. Paramétrage de la localisation

On se rend maintenant sur l'option 4 du menu principal pour paramétrer la localisation, les options de langue, d'heure, de clavier, etc.



Paramétrage de la localisation

Après avoir validé on tombe sur un sous-menu avec 4 entrées : on les configurera toutes, l'une après l'autre, dans l'ordre. Premier choix à valider : la langue, avec la première entrée « *Change locale* » qu'on choisit, puis on valide.

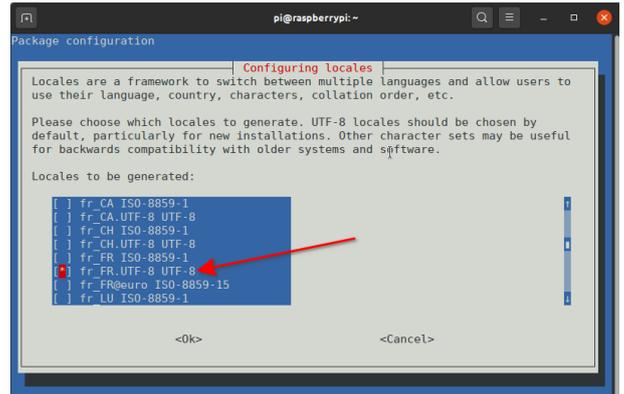


Configuration des paramètres régionaux et de langue

Sur cet écran, il va falloir faire défiler la longue liste des langues avec la flèche bas ↓ du clavier pour choisir la ligne **fr_FR.UTF-8 UTF-8**

 **ATTENTION** : pour choisir une ligne, il faut appuyer sur la barre d'espace. Cela met une petite étoile entre les crochets à gauche de la ligne.

On va ensuite se positionner sur OK avec la touche Tabulation puis on valide.



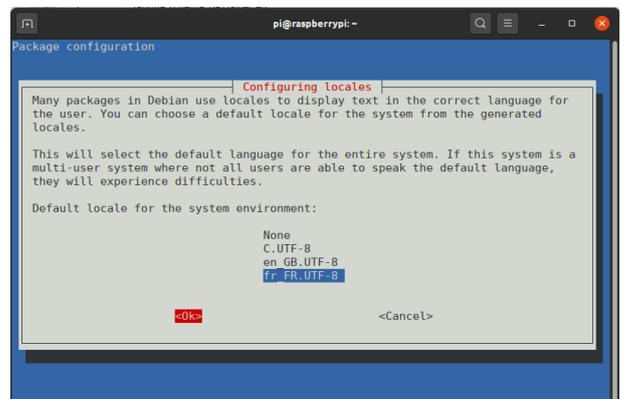
Choix de la langue française dans raspi-config

On arrive sur un écran de confirmation où il faut **choisir la langue par défaut**.

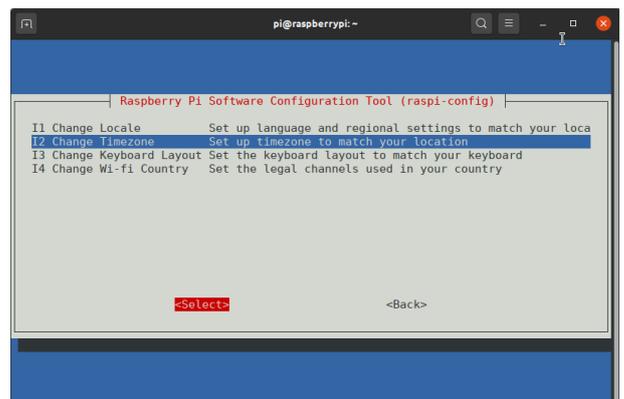
Avec la flèche du clavier, on choisit donc la langue française puis on valide avec OK.

Il faut alors patienter quelques secondes (ou dizaines de secondes, en fonction de votre connexion internet) pour laisser le temps de télécharger les fichiers pour la langue française et de faire les modifications nécessaires.

Pendant ces quelques secondes, on retrouve le terminal où on peut voir l'avancement des opérations, puis on se retrouve à l'écran d'accueil de raspi-config.



On va donc devoir refaire le choix du menu 4 « Localisation options » pour choisir ensuite la 2^e entrée : « Change timezone ».

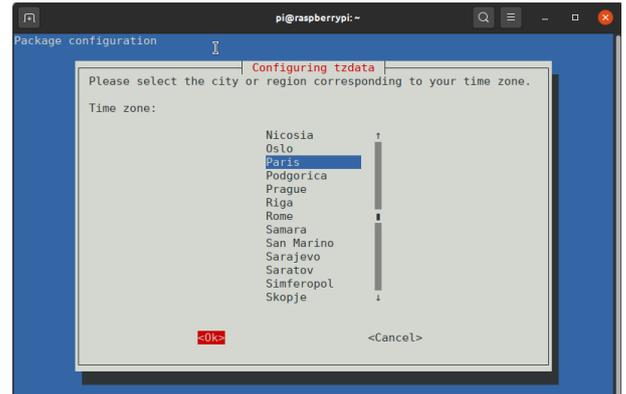


Paramétrer le bon fuseau horaire

Après un écran intermédiaire où on pourra confirmer qu'on est bien sur une zone « *Europe* » on arrivera sur la liste des différentes « *Time zone* » où le réglage par défaut est positionné sur « *London* ».

On fera défiler la liste pour s'arrêter sur « *Paris* » avant de valider.

La paramétrage est pris en compte et on se retrouve à l'écran d'accueil de raspi-config.

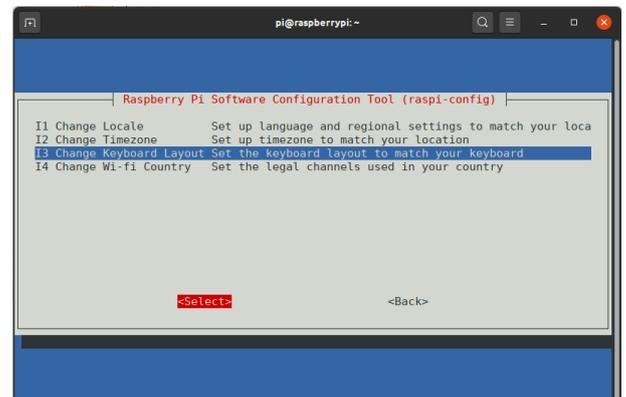


Choix du fuseau horaire Paris

→ option 4 : Localisation options

→ choix 3 : Change keyboard layout : pour adapter les réglages de votre clavier à la configuration de langue choisie.

Là encore, après quelques secondes sur le terminal, on se retrouvera à l'écran d'accueil de raspi-config.



Configuration du clavier

On va revenir une dernière fois sur l'option 4 « *Localisation options* » pour aller ensuite entrer sur le 4^e choix : « *Change Wifi country* » pour adapter la configuration WiFi au pays où nous sommes.

Faire défiler la liste pour arriver jusqu'à la ligne **FR France** puis valider.



Adaptation des paramètres WiFi au pays

On sera alors ramené encore une fois sur l'écran d'accueil de raspi-config.

Les paramètres essentiels ont maintenant été faits, on peut **refermer l'utilitaire de configuration** en sélectionnant « *Finish* » puis en validant.

On se retrouve dans le terminal.

! Attention :

Il ne reste plus qu'à **faire redémarrer le Raspberry** pour que toutes ces modifications soient effectivement prises en compte.

→ Pour cela, on utilise la commande : **sudo reboot**

Comme le Raspberry redémarre, vous allez constater dans le terminal que la connexion SSH avec lui est interrompue... forcément.

On va lui laisser une trentaine de secondes pour faire son redémarrage, puis on va de nouveau établir la connexion SSH pour passer à la suite des opérations (mises à jour du système, etc.) :

Rappel : pour cela, on va utiliser la commande **ssh pi@192.168.1.60** où 192.168.1.60 est l'adresse IP fixe de mon Raspberry, bien évidemment, cette adresse sera à modifier en fonction de votre configuration.

d. Mise à jour du système via SSH

💡 Hypothèse :

Maintenant que la configuration initiale est faite, grâce à l'utilitaire de configuration raspi-config, après l'avoir fait redémarrer, on peut procéder à la mise à jour du système.

Nous sommes toujours connectés au Raspberry via SSH, dans le terminal.

⚙️ Méthode : Mise à jour du système

1. Vérification de l'existence de mises à jour

Dans le terminal connecté au Raspberry en SSH, on saisit la commande :

sudo apt update

On entre le mot de passe administrateur du Raspberry et la recherche de mises à jour est lancée, on voit ici qu'il y a 152 paquets logiciels qui peuvent être mis à jour.

ATTENTION : le mot de passe administrateur pour le Raspberry a été modifié dans les étapes précédentes. Il ne s'agit donc plus ici du mot de passe par défaut « *raspberrypi* » mais bien de celui que vous aurez défini vous-mêmes.

```

pi@raspberrypi:~$ sudo apt update
Atteint :1 http://archive.raspberrypi.org/debian buster InRelease
Atteint :2 http://raspbian.raspberrypi.org/raspbian buster InRelease
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances
Lecture des informations d'état... Fait
152 paquets peuvent être mis à jour. Exécutez « apt list --upgradable » pour les voir.
pi@raspberrypi:~$
  
```

Recherche de mises à jour

2. Téléchargement et installation des mises à jour.

Nous allons maintenant lister et confirmer l'installation des mises à jour disponibles.

Pour cela, on entre la commande :

sudo apt upgrade

La liste des paquets à mettre à jour est alors affichée, on est informés de l'espace que ça prendra sur la mémoire disponible, et on est invité à confirmer l'installation : il suffit de saisir la lettre « o » pour Oui, puis de valider par Entrée.

NB : il y a autant de mises à jour car c'est la toute première vérification suite à l'installation du système. Cela va donc prendre un certain temps pour les télécharger, en fonction de votre connexion internet et surtout pour les installer (plusieurs minutes), étant donné que le Raspberry Pi n'est pas un monstre de puissance processeur. C'est normal.

7. Utiliser Pi-hole sur son Raspberry pour bloquer les publicités

RÉSUMÉ :

Publié en juin 2020

AUTEURS :

- Alain MICHEL

LICENCES :

Creative Commons - Attribution - Partage dans les Mêmes Conditions

Nous savons maintenant [configurer et contrôler le Raspberry Pi entièrement à distance](#) depuis un autre ordinateur, avec le protocole SSH.

Nous allons voir ici une application concrète de ces compétences avec une utilisation particulière du Raspberry en le transformant en **résolveur DNS** de manière à **bloquer toutes les pubs** sur nos périphériques connectés **au sein de notre réseau local** (ordinateurs, téléphones, tablettes).

On utilisera pour cela un logiciel très connu, entièrement dédié à cela : **Pi-hole**

<https://pi-hole.net/>

Toutes les manipulations décrites ci-dessous ont été faites sur un ordinateur fonctionnant sous Linux Ubuntu 20.04

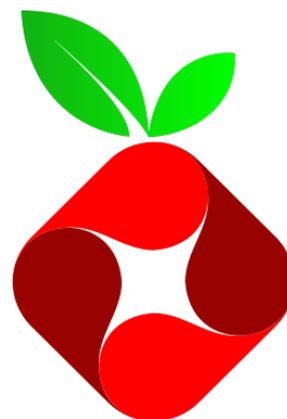
a. Présentation de Pi-hole

Principe de fonctionnement

Pi-hole est un logiciel libre qui s'installe idéalement sur un Raspberry Pi qui sera toujours allumé et connecté à votre box. Il a pour rôle de **bloquer l'affichage des publicités** sur toutes les machines connectées au niveau d'un réseau local. Il joue le rôle de **serveur DNS local** pour bloquer toutes les requêtes faites vers des domaines servant à afficher de la pub sur les sites que vous visitez.

Pi-hole fonctionne en se basant sur des **listes de domaines** malveillants et/ou connus pour servir à l'affichage de publicités. Concrètement, il faut configurer ses machines connectées avec l'adresse IP du Raspberry comme serveur DNS : au cours de notre navigation internet, toutes les requêtes sont envoyées vers Pi-hole, toutes celles qui pointent vers un site figurant dans une de ses listes de blocage seront bloquées, toutes les requêtes légitimes pourront passer et être soumises au serveur DNS public que vous choisirez d'utiliser dans les paramètres de Pi-hole.

Par conséquent, les sites web visités s'afficheront normalement mais les publicités éventuelles qu'ils contiennent ne s'afficheront plus.



Le logo de Pi-hole

Intérêt par rapport à un bloqueur de pubs « classique » dans le navigateur

L'intérêt de Pi-hole par rapport à un bloqueur de publicités classique comme uBlockOrigin (excellent, par ailleurs) est qu'**il n'agit pas seulement dans le navigateur utilisé** à un moment donné sur une machine donnée, **mais sur l'ensemble de toutes les requêtes DNS effectuées par toutes les machines connectées** dans mon réseau local (et qui seront configurées pour l'utiliser, évidemment) y compris les requêtes faites hors navigateur, par exemple les requêtes de télémétrie de certains logiciels, ou les requêtes d'affichage de publicités à l'intérieur de certaines applications gratuites sur mobile.

Il faut savoir que dès l'instant où un smartphone ou une tablette sont connectés à internet, une quantité impressionnante de requêtes DNS sont envoyées très régulièrement, à notre insu, vers des domaines soit publicitaires, soit traqueurs d'activité (soit les deux).

Pi-hole va tout simplement envoyer toutes ces requêtes indésirables et non respectueuses de votre vie privée vers une sorte de « *trou noir* » (d'où son nom) où elles seront perdues (elles sont en fait redirigées vers une adresse IP locale 127.0.0.1, elles n'aboutissent donc pas).

+ Complément : Documentation officielle

Toute la documentation officielle de Pi-hole est accessible ici :

<https://docs.pi-hole.net/>

b. 1. Télécharger et installer Pi-hole

i Définition :

Pi-hole est un **logiciel libre**, il peut être téléchargé (avec toutes ses sources, pour ceux qui ont les compétences pour les étudier et éventuellement les modifier) depuis sa page sur le site Github :

<https://github.com/pi-hole/pi-hole>

-  Tout ce qui suit a été effectué sur un ordinateur fonctionnant sous Linux.
Il est cependant tout à fait possible de le faire depuis un ordinateur sous Windows 10 en utilisant le terminal PowerShell : clic-droit sur le menu Démarrer → Windows PowerShell (admin).

Méthode : Installation sur le Raspberry via SSH

Nous allons installer Pi-hole sur le Raspberry en nous connectant au Raspberry en SSH, comme on l'a vu dans le chapitre précédent.

1. Connexion SSH au Raspberry

- Ouvrir un terminal
- Entrer la commande : **ssh pi@192.168.1.60**
- Entrer le mot de passe administrateur du Raspberry
- Valider par la touche Entrée

Nous sommes maintenant connecté au Raspberry, via SSH, en tant qu'administrateur

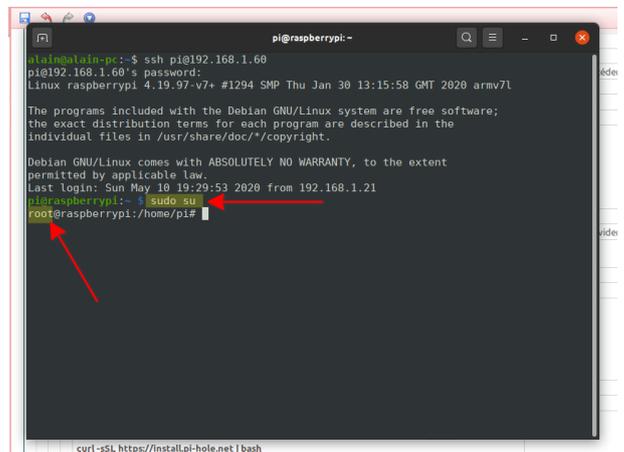
-  → **ATTENTION** : Pour rappel, 192.168.1.60 est l'adresse IP fixe que j'ai attribuée à mon Raspberry dans les chapitres précédents. **Vous utiliserez évidemment dans la commande l'adresse IP que vous aurez choisie** et configurée.

2. Passer en mode root (super utilisateur)

Pour exécuter la commande suivante qui installera Pi-hole, il est nécessaire d'être connecté au Raspberry non pas en tant qu'utilisateur standard « pi » mais en tant que « root », c'est à dire Super Utilisateur, avec tous les droits.

→ Pour cela on saisira dans le terminal la commande suivante :

sudo su



```

pi@raspberrypi:~
pi@raspberrypi:~$ ssh pi@192.168.1.60
pi@192.168.1.60's password:
Linux raspberrypi 4.19.97-v7+ #1294 SMP Thu Jan 30 13:15:58 GMT 2020 armv7l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sun May 10 19:29:53 2020 from 192.168.1.21
pi@raspberrypi:~$ sudo su
root@raspberrypi:~/home/pi#
  
```

Passer en mode Super utilisateur (root)

3. Préparation du téléchargement de Pi-hole

Dans le terminal, on saisit maintenant la commande donnée par les développeurs, sur la page Github :

curl -sSL https://install.pi-hole.net | bash

et on valide par la touche Entrée.

Cela va exécuter un script qui permettra la configuration préalable des principaux paramètres de pi-hole, puis son téléchargement, puis son installation, le tout de manière automatisée.

En savoir plus sur la commande `curl` ^[p.95].

```

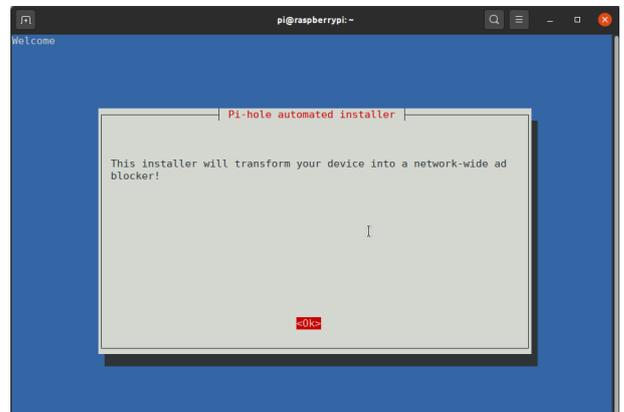
pi@raspberrypi:~$ sudo su
root@raspberrypi:/home/pi# curl -sSL https://install.pi-hole.net | bash
[ ] Root user check

```

Lancement de l'installation de Pi-hole

Après quelques secondes sur l'écran précédent, on arrive sur une interface d'installation qui rappelle celle de raspi-config vue précédemment.

On va avoir quelques écrans informatifs à valider les uns après les autres avant d'arriver à l'installation proprement dite.

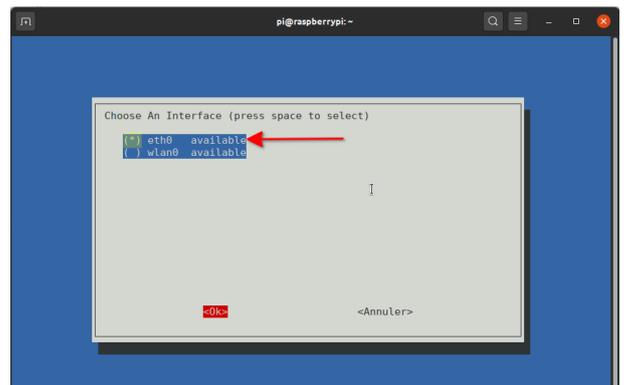


Première étape de l'installation de Pi-hole

4. Choix de l'interface réseau à utiliser pour Pi-hole

On arrive sur un premier choix à faire, celui de l'interface réseau qui sera utilisée par Pi-hole : réseau **filaire** (interface **eth0**) ou réseau WiFi (interface wlan0).

C'est l'interface filaire eth0 qui est choisie par défaut, il est conseillé de ne pas changer ce choix, plus fiable pour le fonctionnement futur.



Choix de l'interface réseau pour le fonctionnement de Pi-hole

5. Choix du serveur DNS extérieur

À cette étape, on choisit un fournisseur DNS public parmi la liste qui est proposée.

Il est possible d'ajouter un autre serveur DNS en choisissant « *custom* » ; il faudra alors le configurer dans un écran suivant.

→ Pour protéger sa vie privée, on NE choisira évidemment PAS le DNS de Google... !

On pourra conseiller le **serveur DNS de Quad9** (filtered, DNSSEC) qui est reconnu comme très fiable et protecteur de la vie privée des utilisateurs, géré par une organisation à but non-lucratif et non pas par une entreprise, possédant de très nombreux serveurs à travers le monde, dont 3 en France (Paris, Lyon et Marseille), cette proximité étant gage de réactivité et de rapidité de traitement des requêtes.

→ Site : <https://www.quad9.net/>

→ « *Quad9 : un résolveur DNS ouvert qui veut vous protéger en respectant votre vie privée* » :

<https://www.nextinpact.com/news/105638-quad9-resolveur-dns-ouvert-qui-veut-vous-protger-en-respectant-votre-vie-privee.htm>

De plus ces serveurs DNS de Quad9 supportent le **chiffrement des requêtes** avec **DoH**^[p.95] (DNS over HTTPS).

6. Choix des listes de blocage des domaines publicitaires

Pi-hole fonctionne sur des listes de référence recensant tous les domaines connus comme étant des domaines publicitaires.

À cette étape on est invité à choisir la ou les listes de référence sur lesquelles va s'appuyer le logiciel pour filtrer les requêtes DNS. Toutes les listes présentées sont cochées par défaut.

→ vous pouvez **décocher la dernière** de la liste : « **HostsFile** » qui n'est apparemment plus maintenue et qui sera en erreur par la suite si on la laisse cochée.

Pour la désélectionner, il suffit d'appuyer sur la barre d'espace quand la ligne est sélectionnée.

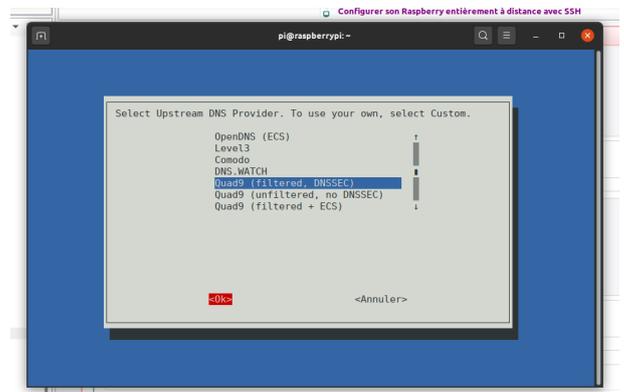
→ On aura plus tard la possibilité d'**ajouter sa propre liste** de blocage, depuis l'interface d'administration de pi-hole.

7. Choix des protocoles IP à utiliser pour Pi-hole

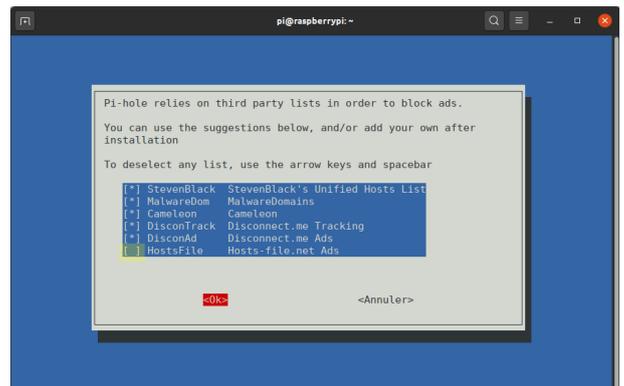
Il faut maintenant déterminer sur quels protocoles IP le logiciel Pi-hole va agir.

Par défaut les deux protocoles **IPv4** et **IPv6** sont sélectionnés.

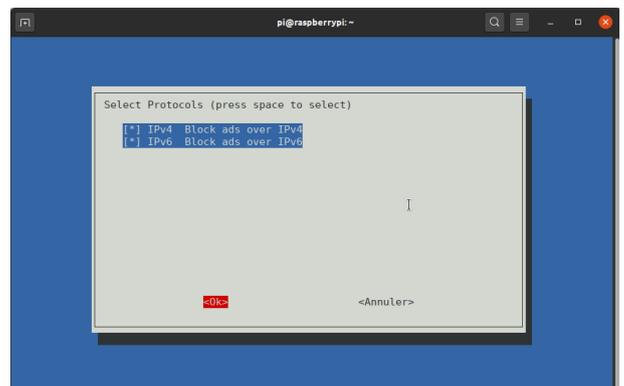
→ on maintient ce choix et on valide.



Choix du fournisseur DNS



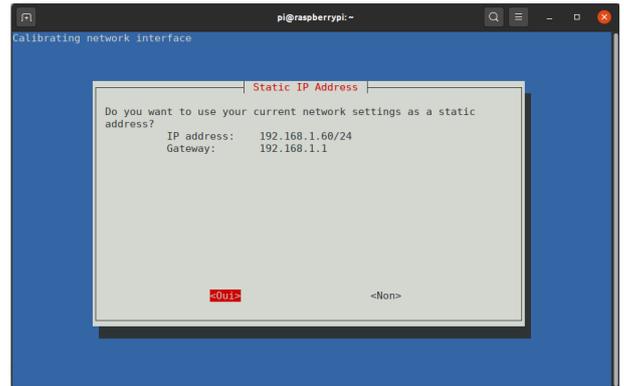
Choix des listes de blocage des domaines publicitaires



Choix des protocoles IP à utiliser

8. Configuration réseau à utiliser

On doit maintenant la configuration IP du réseau local à utiliser. Pi-hole a déjà détecté cette configuration et demande juste confirmation si on veut l'utiliser : on voit sur l'image que l'adresse IP fixe du Raspberry ainsi que l'adresse du routeur (Gateway / passerelle) ont été détectées. Il suffit de valider.



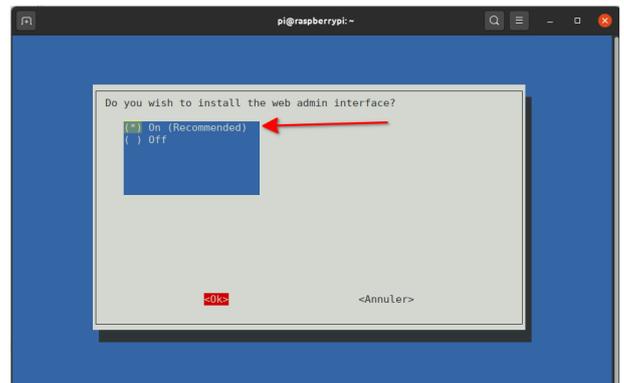
Configuration réseau à utiliser pour Pi-Hole

Viennent ensuite deux écrans purement informatifs qu'il suffira de valider par la touche Entrée.

9. Installation de l'interface d'administration

On va maintenant accepter le choix par défaut qui installera également l'**interface web d'administration** du logiciel Pi-hole. Même s'il est possible de tout administrer en ligne de commande, ce sera quand même bien plus facile et rapide depuis cette interface qui s'affichera dans le navigateur de votre ordinateur.

On valide donc ce choix par OK.

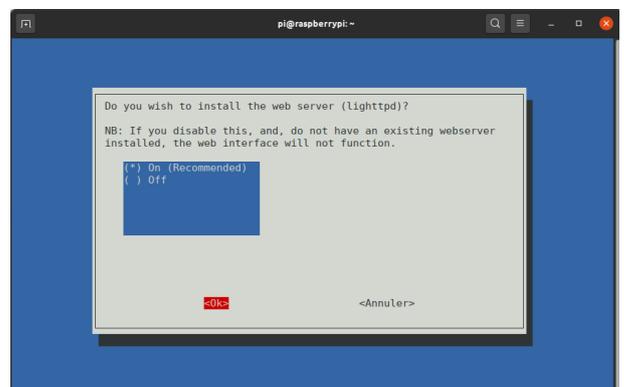


Installation de l'interface web d'administration de Pi-hole

10. Installation du serveur web pour l'interface d'administration

Pour être accessible et fonctionner sur notre ordinateur, l'interface web d'administration du logiciel Pi-hole installé sur le Raspberry a besoin, comme n'importe quel site internet, d'un **serveur Web**.

On va à cette étape valider son installation (le choix est fait pas défaut, il suffit de valider par la touche Entrée).



Installation du serveur web permettant l'utilisation de l'interface d'administration

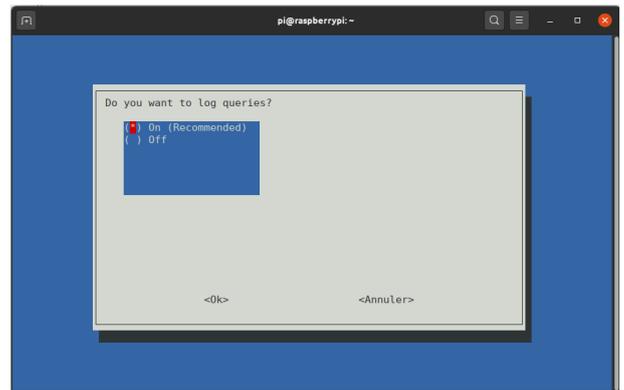
11. Activation de la journalisation des requêtes DNS

Là aussi, on va laisser le choix par défaut, recommandé, pour autoriser l'enregistrement des journaux (logs) de toutes les requêtes DNS qui passeront par le logiciel.

On valide simplement cette étape.

On peut revenir sur ce choix à tout moment pas la suite.

Attention : cet enregistrement d'un journal de toutes les requêtes peut poser un problème de confidentialité et de respect de la vie privée au sein de votre foyer. En effet, en activant ce choix, le Raspberry devient une sorte de « *boîte noire* » dans laquelle seront enregistrées **toutes** les requêtes vers **tous** les sites visités par les ordinateurs ou machines connectées du foyer.

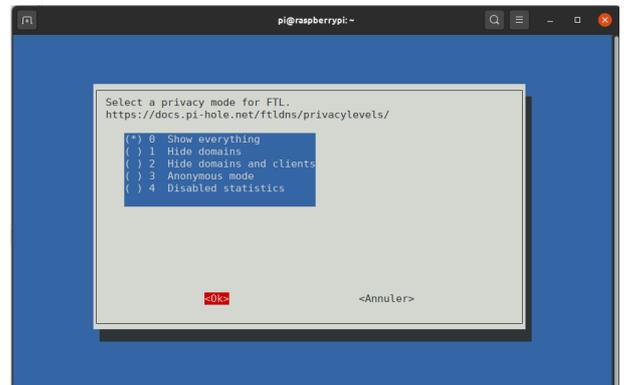


Activation de la journalisation des requêtes

12. Niveau de confidentialité de l'enregistrement des requêtes

À cette étape on a la possibilité de modifier pour l'augmenter le niveau de confidentialité des enregistrements de requêtes à des fins statistiques, en choisissant de masquer (ou pas) certaines informations. Ces enregistrements serviront à construire le « *dashboard* » (tableau de bord) de l'activité de pi-hole visible sur l'interface web du logiciel.

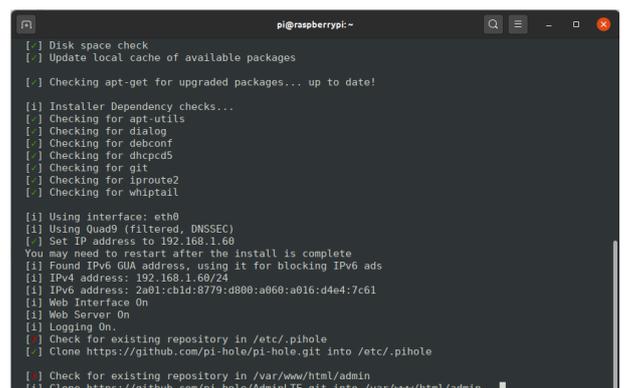
→ on conservera le choix par défaut d'enregistrer les requêtes dans leur intégralité en sachant qu'on pourra à tout moment le modifier par la suite.



Niveau de confidentialité des journaux des requêtes

13. Téléchargement de pi-hole

Après la fin de cette série d'étapes visant à configurer les paramètres de pi-hole, le logiciel est maintenant effectivement téléchargé.



Téléchargement de pi-hole

14. Installation de Pi-hole

Le logiciel est maintenant installé avec tous les paramètres définis au préalable dans les étapes précédentes.

→ Il faut patienter jusqu'au bout du processus, NE PAS refermer le terminal !

15. Fin de l'installation

À la fin de l'installation, un écran récapitulatif des infos utiles pour la suite nous est affiché :

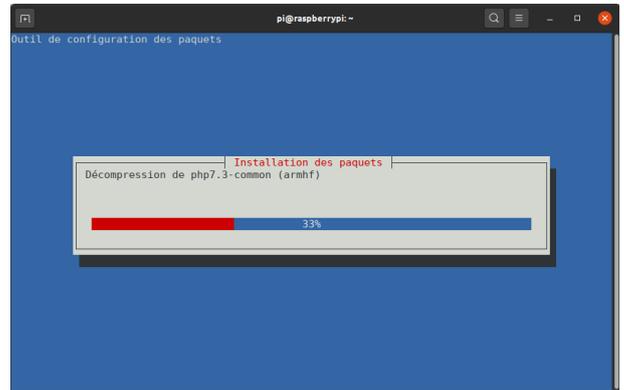
- l'adresse IP à utiliser désormais comme serveur DNS sur nos appareils connectés,
- l'adresse URL de l'interface d'administration de pi-hole,
- le mot de passe par défaut pour s'identifier comme administrateur sur cette interface web.



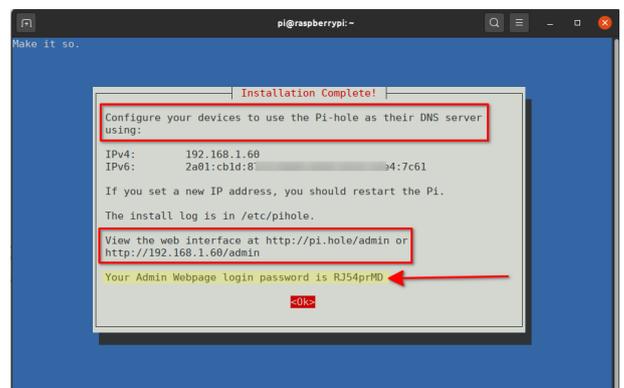
ATTENTION !

Il faut à ce moment-là **noter précieusement ces informations**, en particulier le mot de passe d'accès à l'administration de l'interface Web. Vous pouvez aussi faire une capture d'écran, pour les retrouver lors du tout premier accès à cette interface web d'administration.

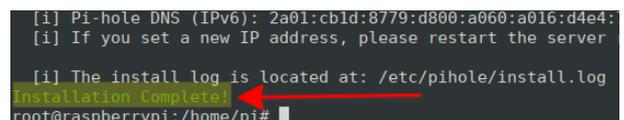
Après avoir validé l'écran précédent, on revient dans le terminal où on voit un message de confirmation d'installation réussie.



Installation de pi-hole



Récapitulatif des infos utiles à la fin de l'installation de Pi-hole



Confirmation d'installation réussie

Pi-hole est maintenant installé et opérationnel sur votre Raspberry.

On peut désormais quitter le contrôle du Raspberry, déconnecter la connexion SSH (avec la commande **exit** dans le terminal) qui nous servait à le commander à distance, et nous diriger vers notre ordinateur à partir duquel on va pouvoir tout d'abord configurer la connexion internet pour utiliser le Raspberry comme serveur DNS et d'autre part surveiller et administrer le fonctionnement de Pi-hole à travers la fameuse interface d'administration Web.

+ Complément : À propos des serveurs DNS et des listes de blocage

Pour en savoir plus sur les **serveurs DNS alternatifs**, sur le blocage des publicités au moyen de **liste de blocage DNS**, on se reportera avantagusement à ces pages, de Sebsauvage :

- Listes de blocage DNS : <https://sebsauvage.net/wiki/doku.php?id=dns-blocklist>
- Liste et comparatif de serveurs DNS alternatifs : <https://sebsauvage.net/wiki/doku.php?id=dns-alternatifs>
- Filtrage des pubs et traqueurs sur Android avec DNSFilter : <https://sebsauvage.net/wiki/doku.php?id=dnsfilter>

c. 2. Configurer ses machines pour utiliser Pi-hole

Configuration « globale » – peu utilisable en pratique

Le plus simple pour définir notre Raspberry Pi comme serveur DNS pour toutes les machines connectées dans notre réseau local serait de **définir l'adresse du serveur DNS directement dans les paramètres de notre routeur** (notre box). Mais cette modification est souvent impossible, elle est **souvent interdite par les FAI** (fournisseurs d'accès internet, comme Orange, SFR, Bouygues, etc.).

Orange par exemple ne le permet pas sur ses Livebox, ce paramétrage n'est pas accessible dans l'interface de configuration de la Livebox.

En effet, les FAI tiennent beaucoup à ce que nous utilisions leurs serveurs DNS, cela leur permet ainsi de traquer toutes les requêtes faites, d'en établir des statistiques, de les exploiter à des fins statistiques (ou publicitaires), voire de les modifier pour nous empêcher d'accéder à certains sites (DNS menteurs).

 Ce dernier point précis (DNS menteurs) n'est pas mauvais en soi dans la mesure où nous pouvons utiliser ce principe à des fins de protection de l'utilisateur et de ses données personnelles, **c'est précisément ce que l'on fait avec Pi-hole !** Malheureusement, il peut aussi nous être imposé par les FAI pour censurer l'accès à certains sites et on va à ce moment-là à l'encontre d'un principe fondateur de l'Internet : la [Neutralité du Net](#).

Cette configuration « globale » reste possible sur les Freebox, et sur tous les routeurs que l'on peut acheter individuellement sur des sites ou magasins spécialisés.

 **ATTENTION !** Ce type de configuration qui se fait au niveau du routeur soulève tout de même des questions au sujet de sa **radicalité** et de son **côté intrusif** dans le cas d'utilisateurs multiples au sein d'un même foyer : n'oublions pas que si elle est mise en place, **toutes les machines connectées au réseau local**, de chaque personne du foyer, passeront forcément à travers Pi-hole pour accéder à internet. Et donc, **toutes les requêtes** faites par toutes ces machines **pourront être enregistrées** dans les journaux du Pi-hole (si les options de confidentialité le permettent, différents niveaux sont paramétrables), ce qui signifie que la personne qui gère Pi-hole pourrait avoir connaissance de tous les sites visités par les personnes du foyer. Dans ce cas, il y a un aspect très intrusif pour la vie privée de ces personnes. L'utilisation de cette solution nécessiterait une information complète à ce sujet de toutes les personnes concernées et/ou un paramétrage bien plus restrictif au niveau des « *Privacy settings* » dans l'interface web de Pi-hole.

Méthode : → Configuration alternative – à conseiller, la plus souvent utilisée

On vient de le voir, dans la plupart des cas il nous sera impossible de définir le Raspberry avec Pi-hole comme serveur DNS utilisé pour nos connexions directement au niveau de notre routeur (et donc une seule fois, pour toutes), **on va donc être obligé de faire cette configuration sur chaque machine**, l'une après l'autre. Par exemple, sur mon ordinateur fixe, puis sur un ordinateur portable, puis sur un téléphone ou tablette.

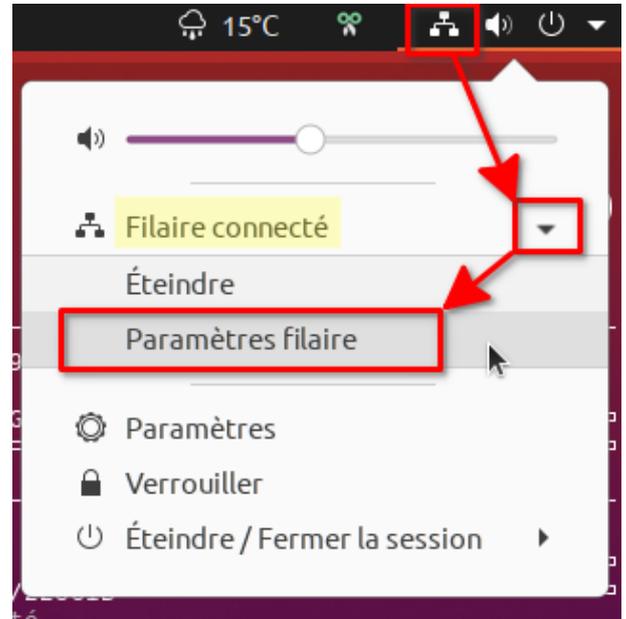
Cette seconde méthode, bien qu'un peu moins rapide aura cependant le mérite d'être moins radicale que la première dans la mesure où on pourra ici – si besoin – décider quelle machine utilisera Pi-hole et quelle autre ne l'utilisera pas.

 Malgré les descriptions qui sont un peu longues à faire (cf. ci-dessous), pas d'affleurement, ces manipulations s'effectuent en quelques secondes sur chaque machine concernée.

Configuration sur un ordinateur sous Linux (Debian, Ubuntu et dérivées)

1. Ouvrir le gestionnaire des connexions réseau

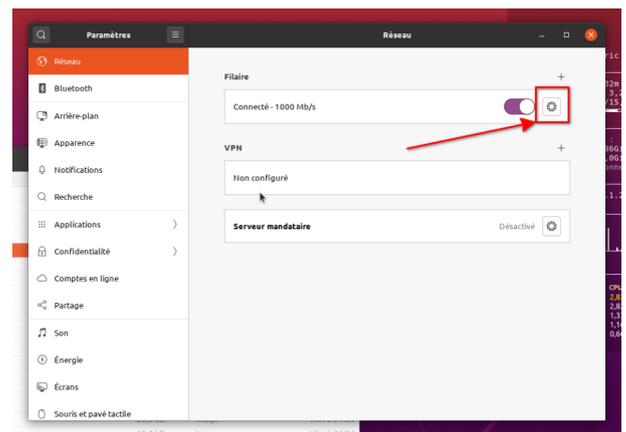
- On commence par cliquer sur l'icône de la connexion réseau, dans la barre tout en haut à droite de l'écran,
- Cliquer sur le petit triangle noir à droite de la ligne « *Filaire connecté* »
- Cliquer ensuite sur « *Paramètres filaire* »



Ouvrir le gestionnaire des connexions réseau

2. Accéder aux paramètres de la connexion filaire

Un clic sur la roue dentée en haut à droite de la fenêtre va ouvrir une autre fenêtre avec tous les paramètres de la connexion filaire.



Accéder aux paramètres de la connexion filaire

3. Fenêtre de paramétrage de la connexion filaire pour IPv4

Sur l'onglet **IPv4**, on va pouvoir indiquer les paramètres suivants :

- désactiver l'attribution automatique d'un serveur DNS en basculant le curseur à droite sur sa position désactivée.
- saisir dans le champ « DNS » l'adresse IP du Raspberry Pi faisant fonctionner Pi-hole qui deviendra notre nouveau serveur DNS.

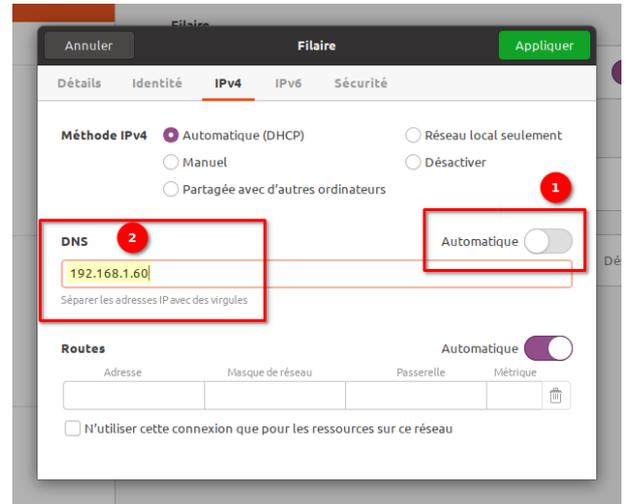


NB : Dans mon exemple, je saisis l'adresse IP fixe attribuée à mon Raspberry : 192.168.1.60
Bien évidemment, vous devrez adapter pour indiquer l'adresse IPv4 de votre Raspberry.

NB : il peut être judicieux de **rajouter un serveur DNS secondaire**, qui sera utilisé si le premier (celui de Pi-hole) ne répond pas pour

une raison ou pour une autre (par exemple, le Raspberry Pi ne fonctionne plus, rencontre un problème, a été débranché, etc.) Dans ce cas, on rajoutera son adresse IP à la suite de la première, simplement séparée par une virgule, comme indiqué. On peut choisir celui de Quad9 qui est utilisé aussi par Pi-Hole : 9.9.9.9 On pourrait donc avoir dans le champ dédié les informations suivantes :

192.168.1.60,9.9.9.9



Fenêtre de paramétrage de la connexion filaire

→ Cela signifie que mon ordinateur utilisera désormais comme serveur DNS celui de Pi-hole (192.168.1.60), mais que si ce dernier devient indisponible, il utilisera en secours le 2^e serveur renseigné (9.9.9.9), pour éviter que la navigation web devienne impossible.

4. Fenêtre de paramétrage de la connexion filaire pour IPv6

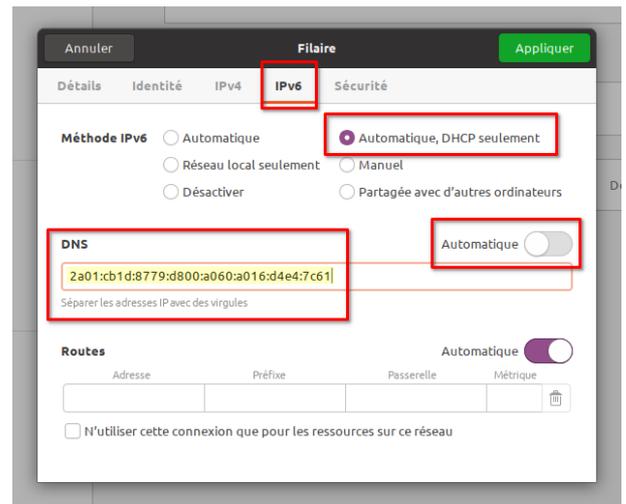
Sur l'onglet **IPv6**, on va pouvoir indiquer les paramètres suivants :

- cocher la case « Méthode IPv6 : **Automatique, DHCP seulement** »
- désactiver l'attribution automatique d'un serveur DNS en basculant le curseur à droite sur sa position désactivée.
- saisir dans le champ « DNS » l'adresse IPv6 du Raspberry Pi faisant fonctionner Pi-hole qui deviendra notre nouveau serveur DNS.



Cette adresse nous a été fournie à la dernière étape de l'installation de Pi-Hole.

Si vous ne l'avez pas notée, ou si vous ne la retrouvez pas, on pourra toujours la récupérer plus tard depuis l'interface web d'administration de Pi-hole. Dans ce cas, laisser cette étape de configuration du DNS en IPv6 en suspens et revenir plus tard pour la finaliser.



Fenêtre de paramétrage de la connexion filaire IPv6

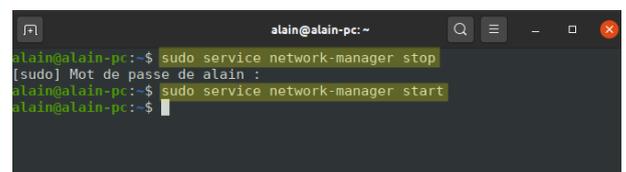
5. Valider et refermer les deux fenêtres de configuration

On peut maintenant cliquer sur le bouton vert « **Appliquer** » en haut à droite de la fenêtre des paramètres, elle va se refermer. On peut ensuite refermer la fenêtre générale des paramètres qui s'était ouverte tout à l'heure.

6. Rendre ces modifications effectives au niveau de la connexion

Nous avons validé les nouveaux paramètres, mais ils ne sont pas encore effectifs et utilisés par la connexion en cours. **Pour que ces paramètres soient effectifs** et qu'enfin notre machine utilise Pi-hole, **il va falloir faire redémarrer de gestionnaire de connexions.**

Pour cela, on va utiliser successivement deux commandes dans le terminal.



Redémarrer le gestionnaire des connexions

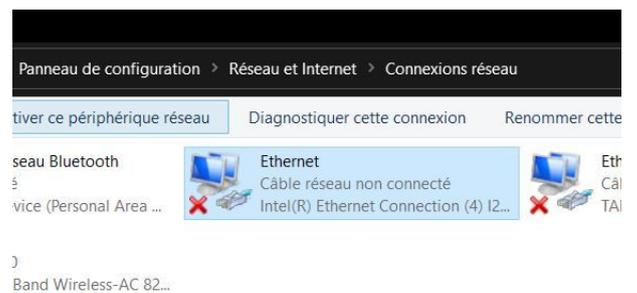
- ouvrir le terminal (Ctrl + Alt + T)
- saisir et valider la commande : **sudo service network-manager stop** pour arrêter le gestionnaire de connexions réseau (le connexion internet est coupée)
- saisir et valider la commande : **sudo service network-manager start** pour démarrer le gestionnaire de connexions (la connexion internet est rétablie et elle utilise nos nouveaux paramètres).

Configuration sur un ordinateur sous Windows 10

Pour accéder aux paramètres de la connexion :

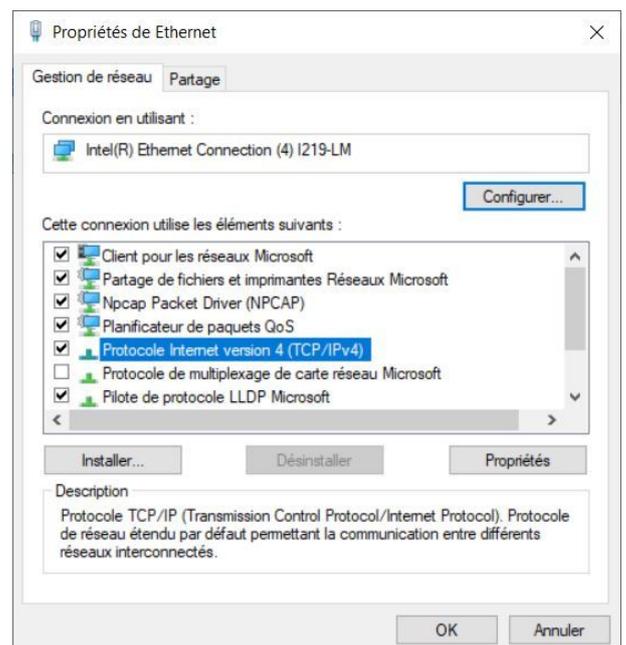
- Cliquer sur le **menu démarrer**
- Cliquer sur la roue dentée « **Paramètres** »
- Cliquer sur la catégorie « **Réseau et internet** »
- Dans la zone à gauche de l'écran, cliquer sur **Ethernet**
- Cliquer sur « **Modifier les options d'adaptateur** »

Une nouvelle fenêtre va s'ouvrir, elle présente les différentes cartes réseau présentes sur l'ordinateur (carte réseau filaire Ethernet ou carte réseau sans fil WiFi, ou Bluetooth). Sur cette fenêtre, **double-cliquer sur l'icône de la carte réseau Ethernet** utilisée par l'ordinateur.



Une troisième petite fenêtre va alors s'ouvrir, permettant d'accéder à tous les paramètres de la carte réseau Ethernet.

Dans la liste des éléments visible au centre de cette fenêtre, on va **double-cliquer sur la ligne Protocole Internet version 4 (TCP-IPv4)**



Une nouvelle fenêtre va (encore !!!) s'ouvrir.

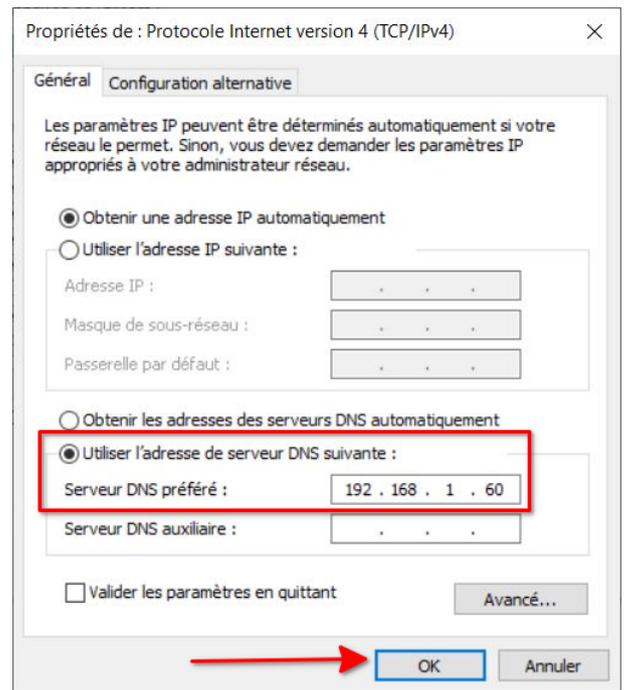
Dans cette fenêtre on va :

- Cocher la case « **Utiliser l'adresse de serveur DNS suivante :** »
- Saisir l'adresse IPv4 de notre Raspberry : 192.168.1.60 (pour l'exemple) dans le champ « *Serveur DNS préféré* ».
- Valider par OK

NB : il peut être judicieux de **rajouter un serveur DNS secondaire**, qui sera utilisé si le premier (celui de Pi-hole) ne répond pas pour une raison ou pour une autre (par exemple, le Raspberry Pi ne fonctionne plus, rencontre un problème, a été débranché, etc.)

Dans ce cas, on rajoutera son adresse IP dans le second champ « *Serveur DNS auxiliaire* », comme indiqué. On peut choisir celui de Quad9 qui est utilisé aussi par Pi-Hole : 9.9.9.9

→ Cela signifie que mon ordinateur utilisera désormais comme serveur DNS celui de Pi-hole (192.168.1.60), mais que si ce dernier devient indisponible, il utilisera en secours le 2^e serveur renseigné (9.9.9.9), pour éviter que la navigation web devienne impossible.



On se retrouve à nouveau sur la fenêtre de l'étape précédente, avec l'accès aux paramètres de la carte réseau Ethernet.

On va maintenant **double-cliquer sur la ligne *Protocole Internet version 6 (TCP-IPv6)***

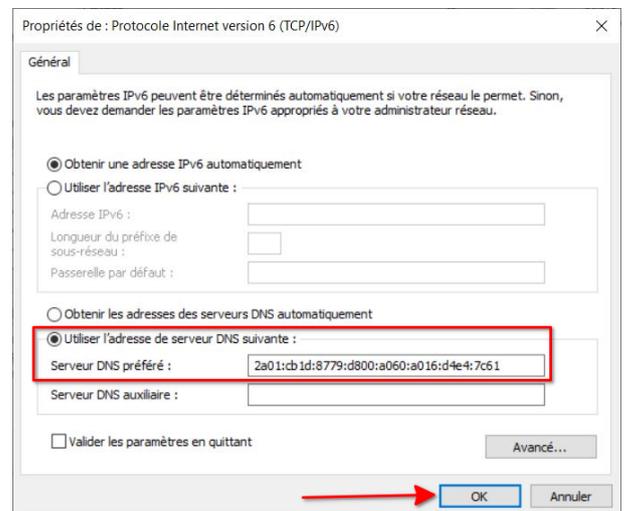
Même principe que pour IPv4.

Dans cette fenêtre on va :

- Cocher la case « **Utiliser l'adresse de serveur DNS suivante :** »
- Saisir l'adresse IPv6 de notre Raspberry 2a01:cb1d:8779:d800:a060:a016:d4e4:7c61 (pour l'exemple) dans le champ « *Serveur DNS préféré* ».
- Valider par OK

 **Cette adresse nous a été fournie à la dernière étape de l'installation de Pi-Hole.**

Si vous ne l'avez pas notée, ou si vous ne la retrouvez pas, on pourra toujours la récupérer plus tard depuis l'interface web d'administration de Pi-hole. Dans ce cas, laisser cette étape de configuration du DNS en IPv6 en suspens et revenir plus tard pour la finaliser.



Valider à nouveau par OK pour refermer la fenêtre des paramètres de la carte.

Sur la fenêtre des différentes connexions réseau :

- faire un **clic-droit Désactiver** sur l'icône de la carte réseau Ethernet dont on vient de modifier les paramètres de connexion.
- après quelques secondes, refaire un **clic-droit Activer** sur cette même icône de la carte réseau Ethernet.

Configuration sous Android

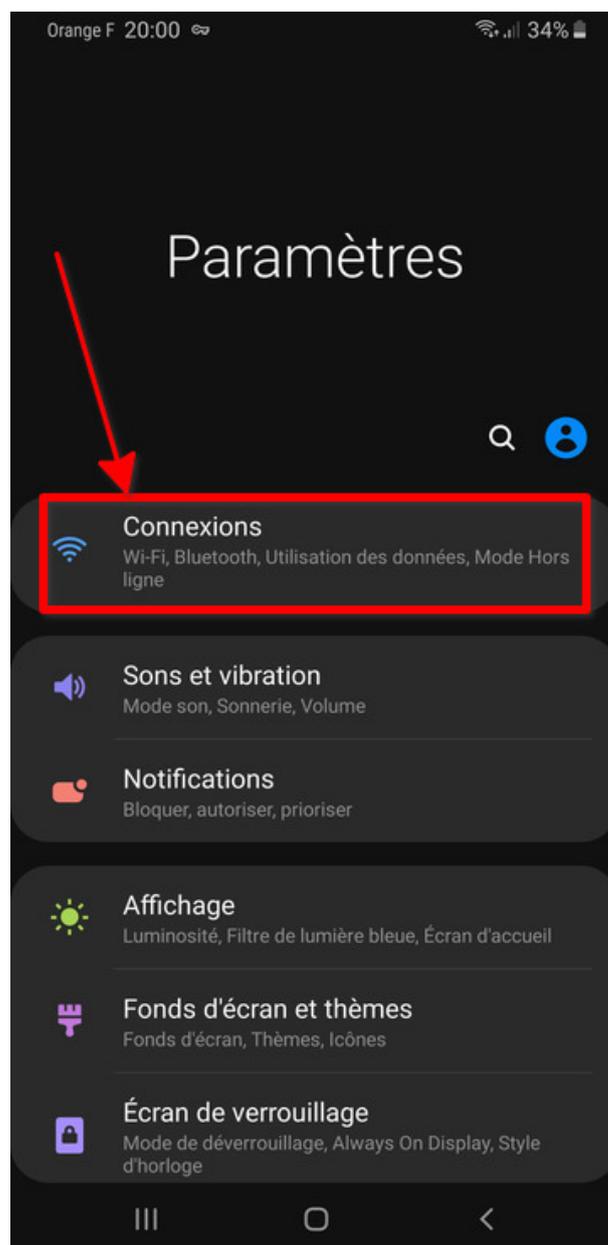
Il est bien sûr possible de modifier le serveur DNS pour votre appareil Android connecté en WiFi à votre routeur, pour lui indiquer d'utiliser Pi-hole.

Voici comment procéder.

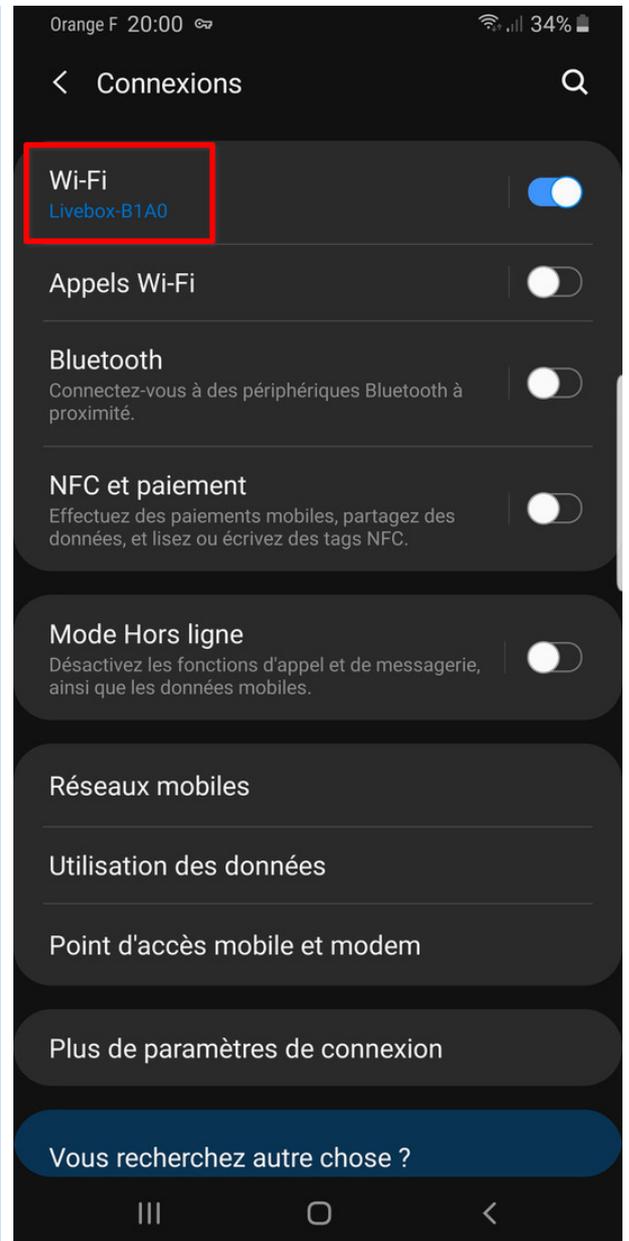
Méthode : Modifier le serveur DNS sur un appareil Android

 Cette description est faite sur un téléphone fonctionnant sous Android 9.

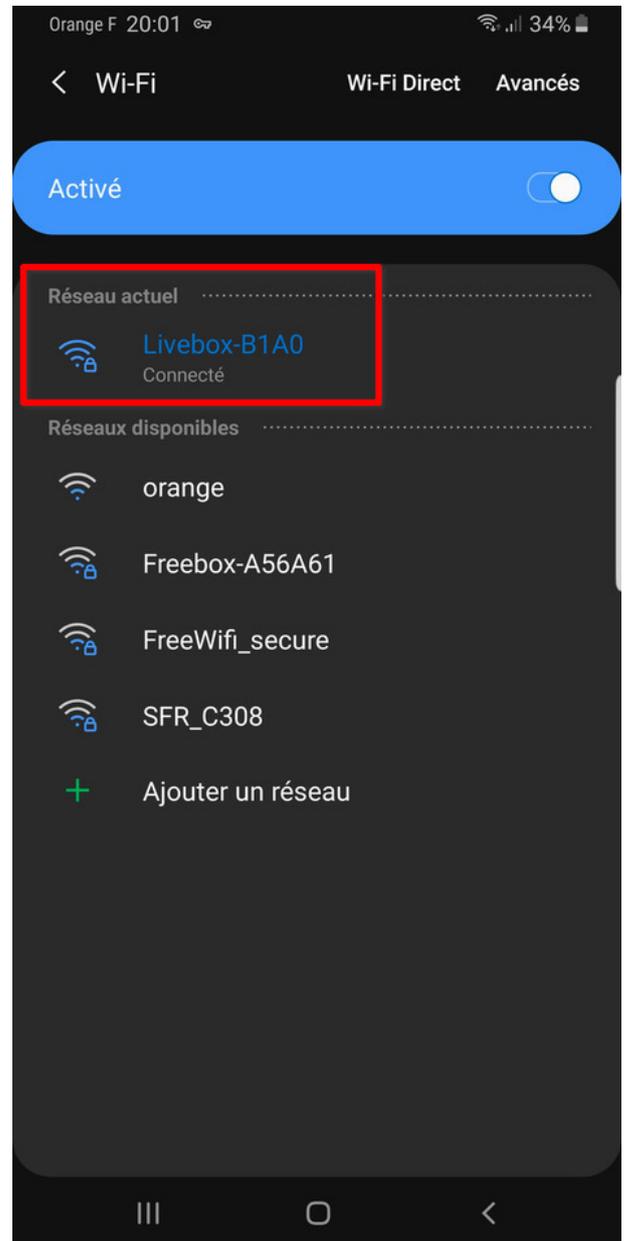
1. aller dans les paramètres, sur « *Connexions* »



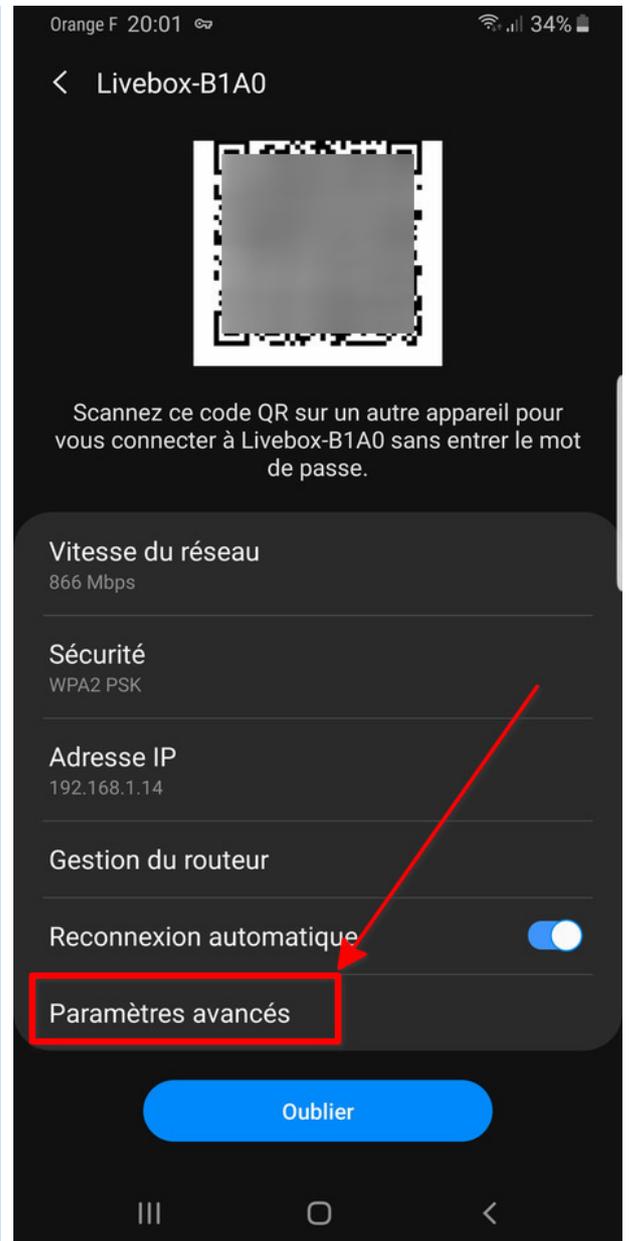
2. Cliquer sur WiFi



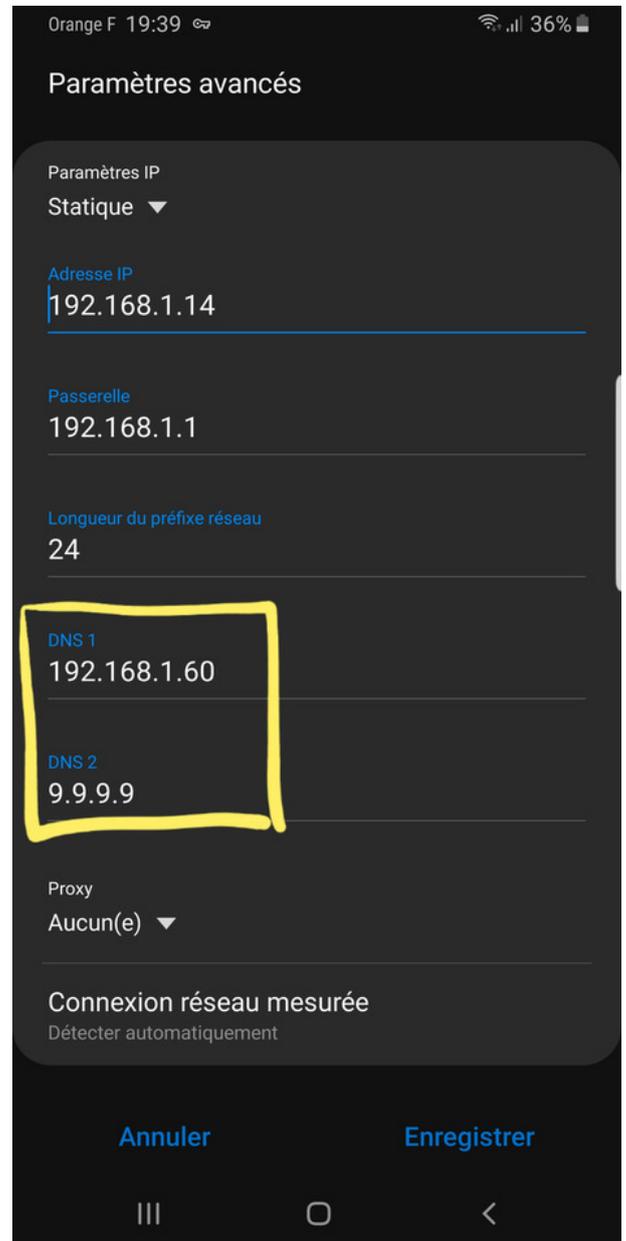
3. Cliquer sur le nom de la connexion WiFi.



4. Cliquer sur « Paramètres avancés »



5. Dans le champ DNS1 on saisira l'adresse IP du Pi-hole
Dans le champ DNS2, on pourra saisir l'adresse d'un serveur DNS alternatif (ici, celui de Quad9), que l'appareil utilisera si jamais un incident a rendu le Pi-hole non fonctionnel.



Configuration sous iOS

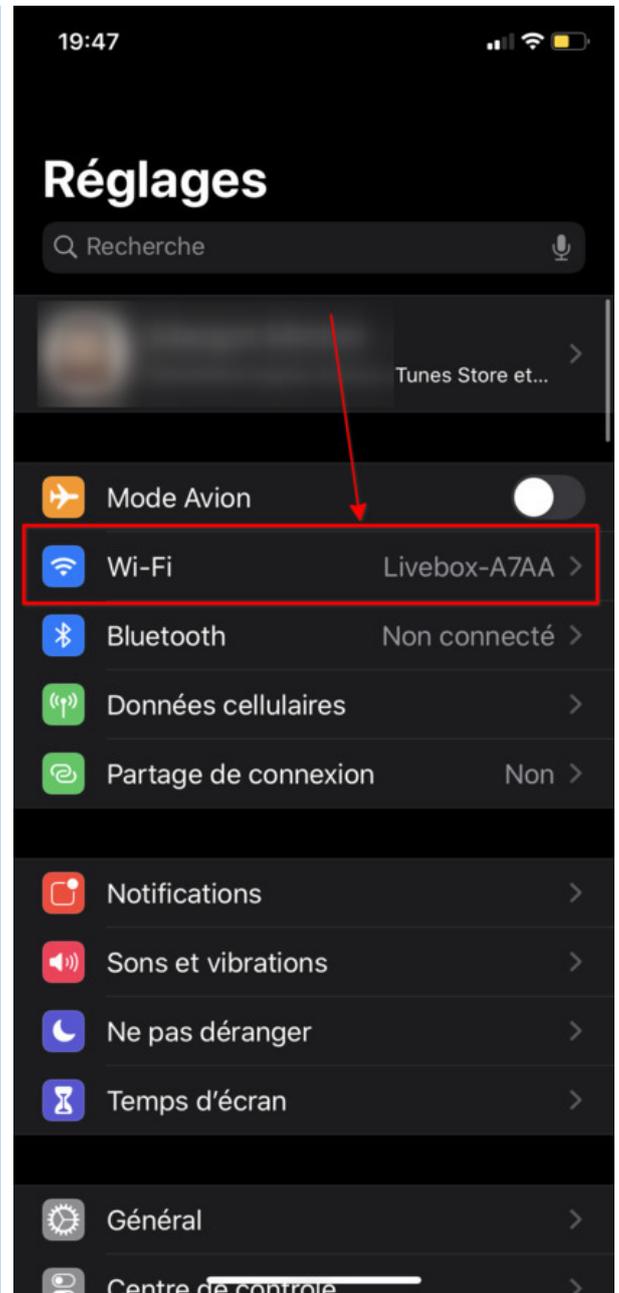
Il est également possible de modifier le serveur DNS pour votre appareil iOS connecté en WiFi à votre routeur, pour lui indiquer d'utiliser Pi-hole.

Voici comment procéder.

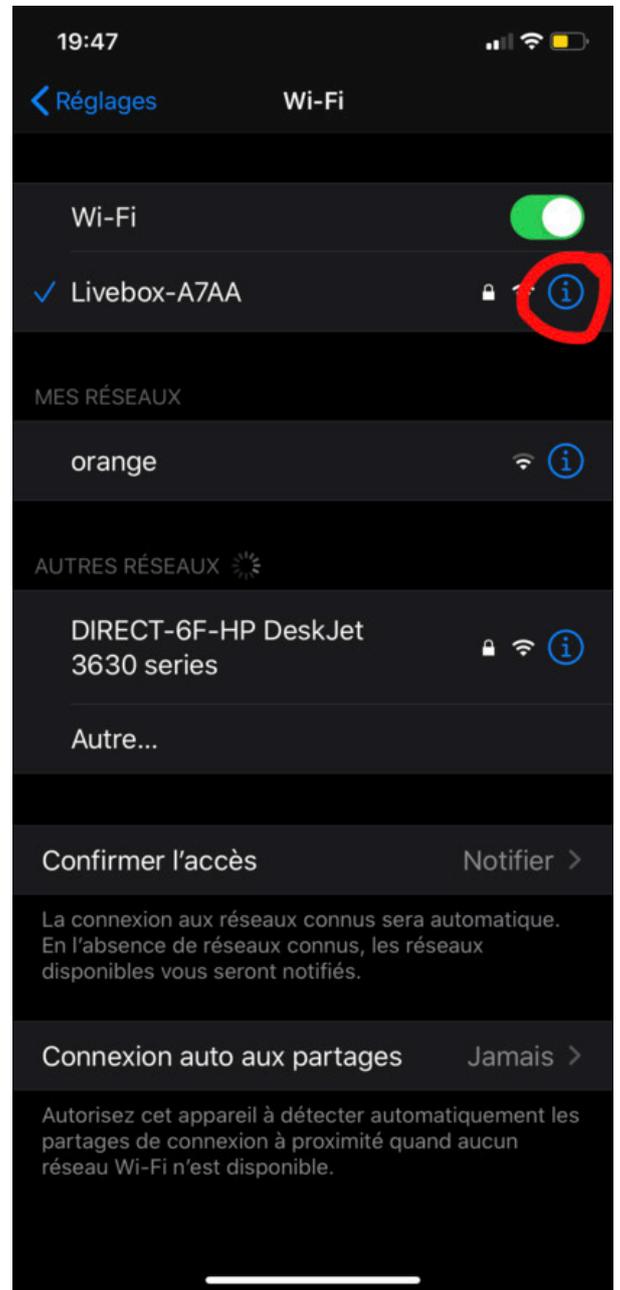
Méthode : Modifier le serveur DNS sur un appareil iOS

 Cette description et les captures d'écran sont faites sur un iPhone fonctionnant sous iOS 13.

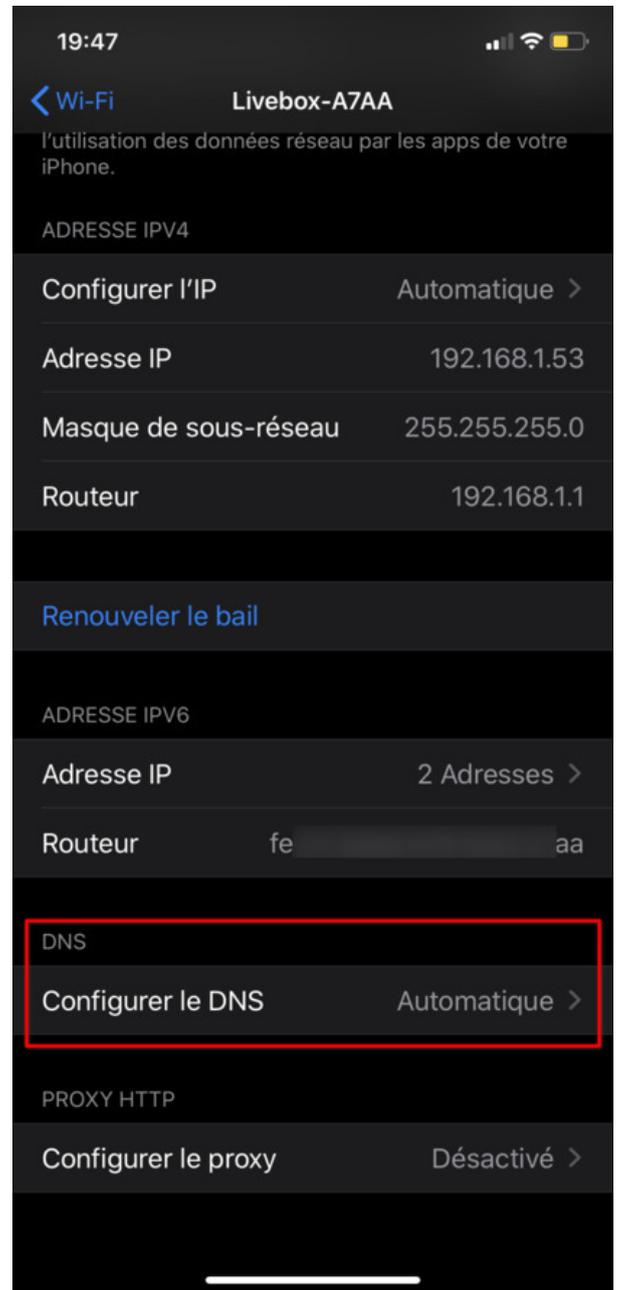
1. Aller dans « Réglages » → WiFi



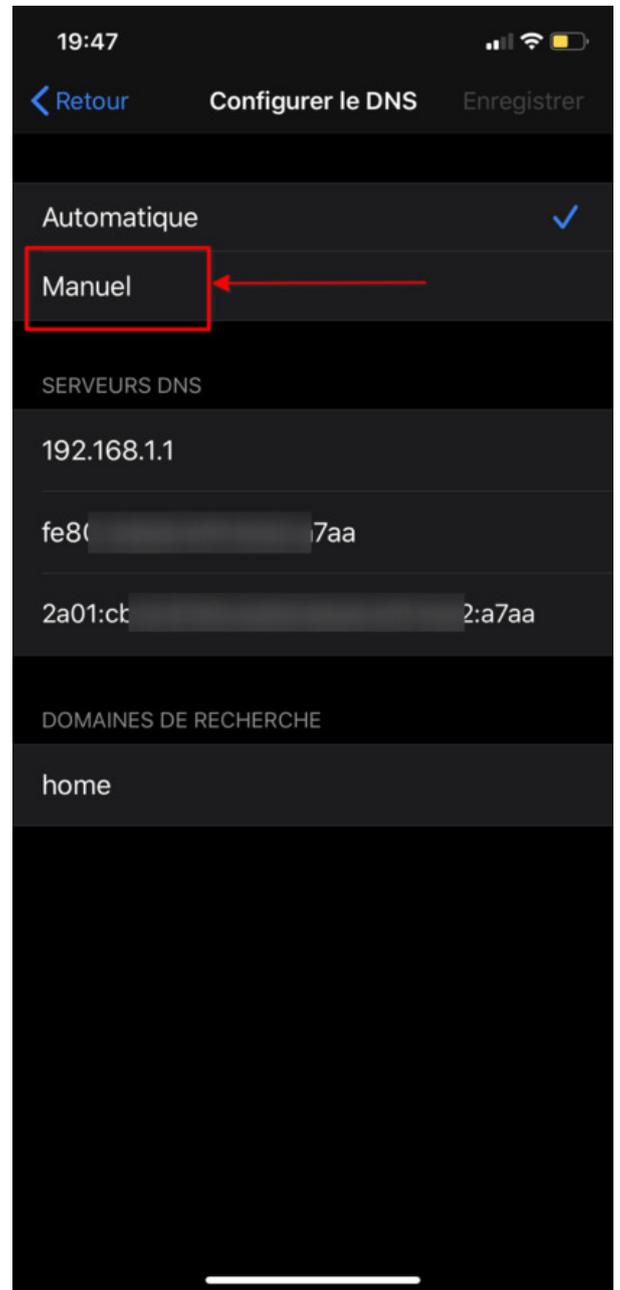
2. Cliquer sur le réseau WiFi auquel on est connecté (il faut cliquer sur « i » bleu à droite de la ligne.



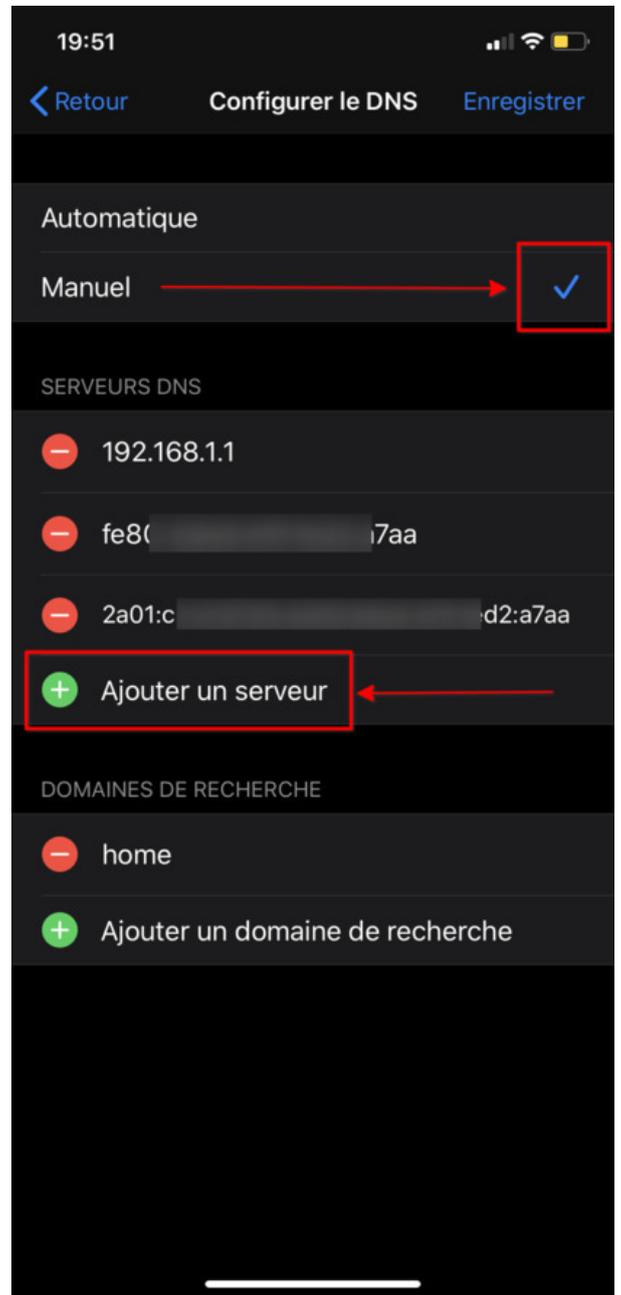
3. Cliquer sur « Configurer le DNS »



4. Cliquer sur « Manuel »



5. Cliquer sur « Ajouter un serveur »



6. Saisir l'adresse IP du Pi-Hole qu'on utilisera comme serveur DNS.



d. 3. Interface Web d'administration de Pi-hole

Maintenant que notre Pi-hole est installé et que son utilisation est configurée sur nos machines au sein de notre réseau local (ordinateur fixe, portable, téléphones, tablette), il ne nous reste plus qu'à accéder à son interface web d'administration à partir de laquelle on pourra, entre autres choses :

- observer un **graphique de l'activité**, avec le nombre de requêtes bloquées,
- consulter la **liste détaillée de requêtes DNS**, bloquées ou pas
- créer et gérer une « **liste blanche** » de sites qu'on souhaite toujours autoriser, et une « **liste noire** » de sites qu'on souhaite toujours interdire
- gérer les listes de domaines à bloquer sur lesquelles s'appuie Pi-hole pour fonctionner : on peut en rajouter ou en supprimer.
- désactiver Pi-hole de manière temporaire, sur la durée qu'on souhaite (quelques secondes, quelques minutes, quelques heures)
- vérifier quels sont les appareils de notre réseau local qui utilisent effectivement Pi-hole (ou pas)

- **modifier le serveur DNS** public utilisé par Pi-hole, en choisissant parmi la liste proposée par défaut ou bien en ajoutant un serveur DNS de son choix.

Méthode : Accéder à l'interface Web en consultation

On peut accéder à l'interface web d'administration de Pi-hole simplement en consultation, pour observer les graphiques d'activité, sans être authentifié.

L'accès se fait dans son navigateur web, en utilisant une de ces deux possibilités :

- `http://<ADRESSE-IP-DE-VOTRE-PIHOLE>/admin/` (dans mon exemple, ce serait donc `http://192.168.1.60/admin/`)
- `http://pi.hole/admin`

Conseil : Modifier le mot de passe par défaut de l'interface web d'administration

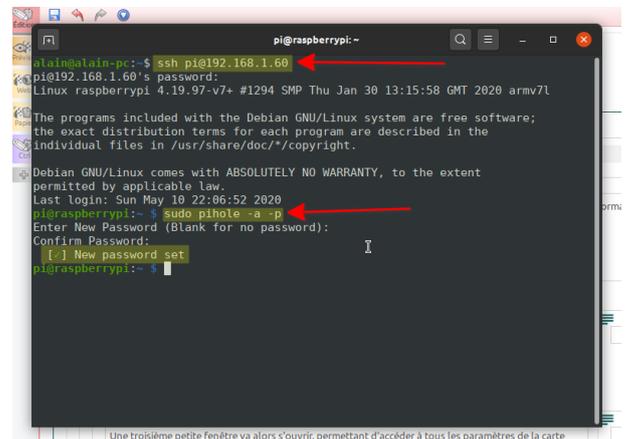
Vous vous rappelez qu'à la fin de l'installation de Pi-hole on avait obtenu un mot de passe provisoire pour accéder à l'interface web en tant qu'administrateur (cf. chapitre « *Télécharger et installer Pi-hole* », un peu plus haut).

La première des choses qu'on va pouvoir faire maintenant, c'est de modifier ce mot de passe fourni pour en choisir un plus personnel.

Pour cela, on va :

- ouvrir un terminal (Ctrl + Alt + T)
- établir une connexion SSH avec le Raspberry : **ssh pi@192.168.1.60** (adresse IP selon mon exemple, à adapter à votre cas)
- entrer le mot de passe administrateur du Raspberry
- une fois la connexion établie, saisir la commande **sudo pihole -a -p** et valider
- **Choisir un nouveau mot de passe** pour l'accès administrateur à l'interface web de Pi-hole et valider (Entrée)
- Confirmer ce nouveau mot de passe et valider.

Vous obtenez alors la confirmation que le nouveau mot de passe est bien pris en compte.



```

pi@raspberrypi:~$ ssh pi@192.168.1.60
pi@192.168.1.60's password:
Linux raspberrypi 4.19.97-v7+ #1294 SMP Thu Jan 30 13:15:58 GMT 2020 armv7l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sun May 10 22:06:52 2020
pi@raspberrypi:~$ sudo pihole -a -p
Enter New Password (Blank for no password):
Confirm Password:
[ ] New password set
pi@raspberrypi:~$

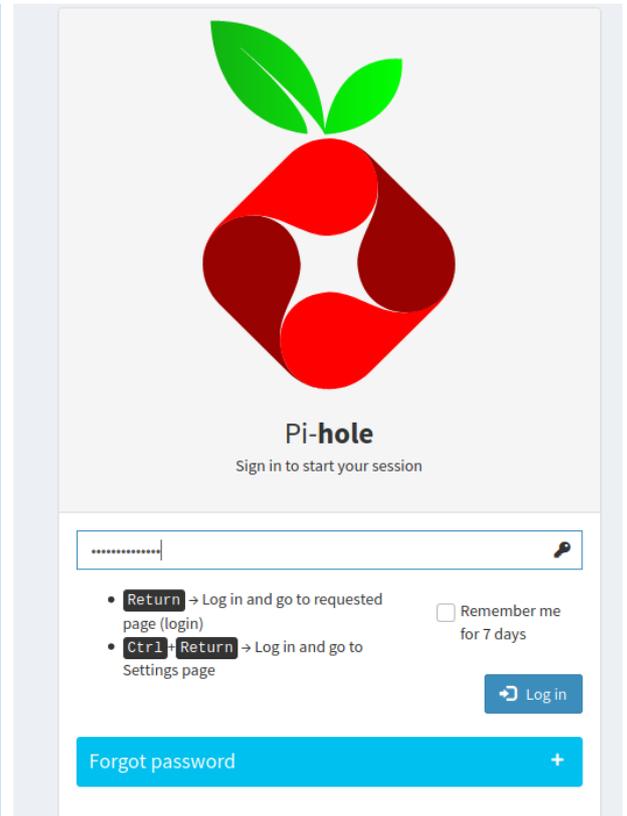
```

Modification du mot de passe d'accès à l'interface web d'administration de Pi-hole

Méthode : Accéder à l'interface Web d'administration et s'authentifier

Même principe, on accède par exemple à l'adresse **http://pi.hole/admin** et on clique sur le lien « Login » dans le menu à droite.

On valide en appuyant sur la touche Entrée ou en cliquant sur le bouton « Log in ».



On est maintenant authentifié en tant qu'administrateur du Pi-hole.

On voit apparaître plusieurs entrées de menu qui n'étaient pas visibles auparavant dans la colonne de gauche.



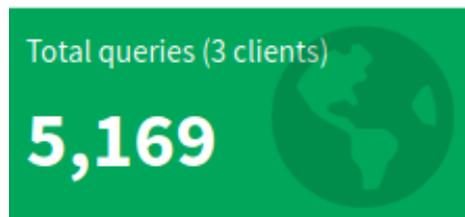
Définition : Comprendre la vue principale de l'interface d'administration.

Vue principale de l'interface administrateur de Pi-hole

Cette page est celle où on arrive après s'être authentifié sur l'interface web d'administration de Pi-hole.



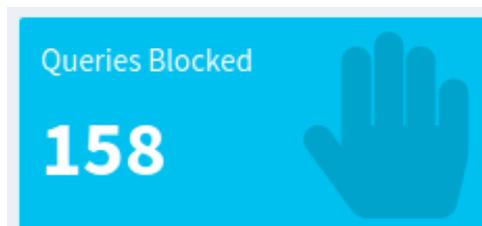
1 Nombre de requêtes DNS traitées



Il s'agit du nombre de requêtes DNS qui ont été traitées par le Pi-hole, en provenance des différentes machines du mon réseau local qui l'utilisent, depuis la mise en fonctionnement de Pi-hole.

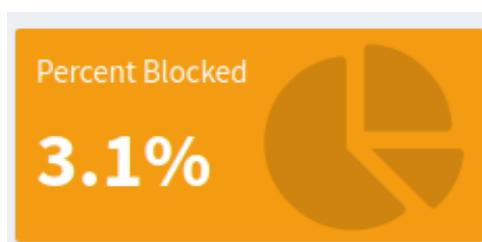
Ce nombre est actualisé régulièrement, il est cumulatif.

2 Nombre de requêtes bloquées



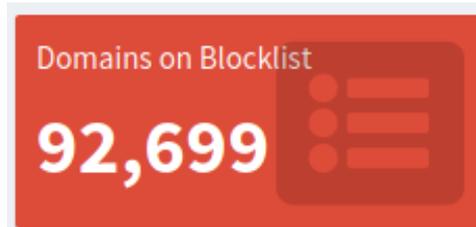
Parmi toutes les requêtes traitées, on voit ici combien ont été bloquées par les filtres de Pi-hole.

3 Pourcentage de requêtes bloquées



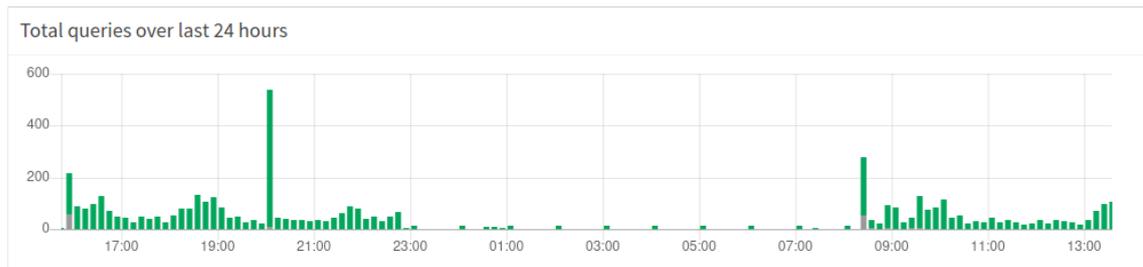
Pourcentage total de requêtes bloquées depuis la mise en service de Pi-hole.

4 Nombre de domaines de blocage



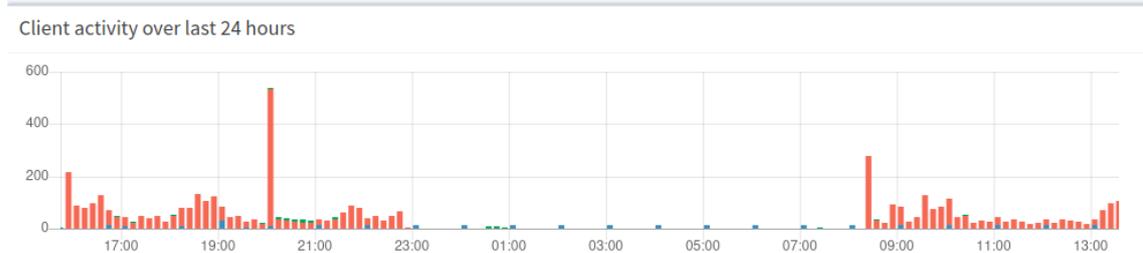
Il s'agit ici du nombre total de domaines qui peuvent être bloqués par Pi-hole, en fonction des listes de blocages choisies et actives.

5 Graphique horodaté des requêtes



Représentation sur un graphique horodaté de toutes les requêtes traitées au cours des dernières 24 h.

6 Graphique horodaté des requêtes par client connecté

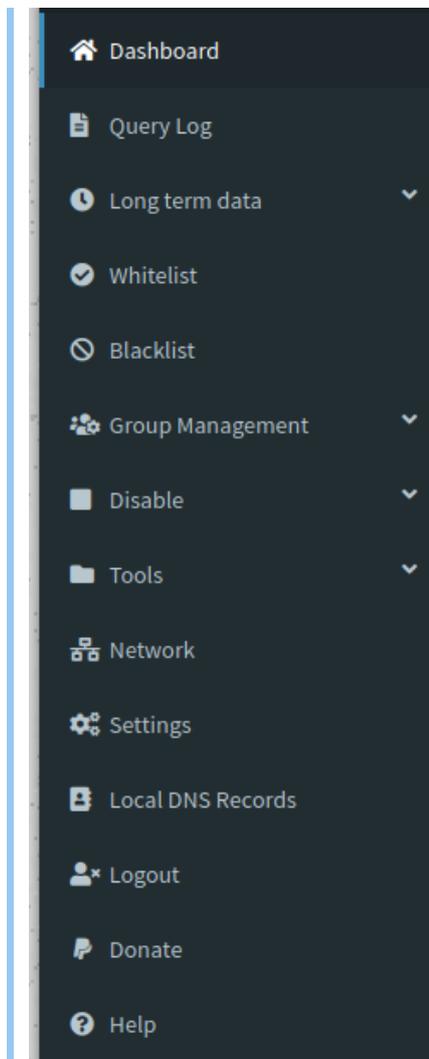


On voit sur ce graphique la totalité des requêtes traitées eu cours des dernières 24 h, réparties selon les machines connectée (même si ce n'est pas trop visible sur la capture d'écran, il y a une couleur différente par machine).

7 Menu

Menu d'accès aux différentes parties et paramètres de l'interface d'administration de Pi-hole.

On pourra en particulier :



- consulter le journal détaillé des requêtes traitées → **Query log**
- créer une liste blanche de domaines toujours autorisés → **Whitelist**
- créer une liste noire de domaines toujours bloqués → **Blacklist**
- modifier ou ajouter des listes de blocage : dans **Group Mangement** → **Adlists**
- désactiver temporairement Pi-hole en cas de besoin → **Disable**
- vérifier quelles sont les machines du réseau local qui utilisent Pi-hole → **Network**
- choisir quels serveurs DNS externes publics (upstream DNS servers) Pi-hole doit utiliser → **Settings** (onglet DNS)
- Se déconnecter de l'interface administrateur → **Logout**

Conseil : Ajouter une liste de blocage DNS

Il peut être intéressant d'ajouter une nouvelles liste de blocage DNS aux listes existantes, pour amener encore un peu plus de possibilités à Pi-hole de bloquer des domaines malveillants et/ou publicitaires.

Une **liste toute prête, efficace, régulièrement actualisée** est disponible depuis cette page :

<https://sebsauvage.net/wiki/doku.php?id=dns-blocklist>

Elle est proposée et est maintenue par une personne particulièrement impliquée dans de domaine, une référence pour tout ce qui concerne Linux, les logiciels libres, la protection de la vie privée en particulier, ou l'informatique en général : « *Sebsauvage* », qui entretien un Wiki qui est une mine de ressources : <https://sebsauvage.net/wiki/doku.php?id=accueil>



Vous consulterez sur la page donnée ci-dessus toutes les explication et les caractéristiques de cette liste de blocage.

Pour la rajouter dans Pi-Hole, on utilisera donc l'URL de cette liste : <https://sebsauvage.net/hosts/hosts>

Par curiosité, au lieu de copier ce lien pour l'ajouter à Pi-hole (comme on va le voir ci-dessous), vous pouvez aussi cliquer dessus pour simplement afficher l'intégralité de cette liste si vous avez envie d'aller voir comment elle est faite et d'éplucher les 248 000 domaines différents qu'elle contient !

On note dans les en-têtes qu'elle est régulièrement mise à jour (dernière version en date du 12 mai, c'est à dire du jour même où j'écris ces contenus). Il y a une mise à jour à peu près hebdomadaire de cette liste.

On note aussi qu'elle contient exactement pour cette version **248 134 domaines**.

```

# Host-block file
# URL: https://sebsauvage.net/hosts/hosts
# https://sebsauvage.net/wiki/doku.php?id=dns-blocklist
# Generated on: 2020-05-12 08:46:58
# BLOCKS: ADVERTISING (WEB and MOBILE apps), analytics (web and mobile apps), tracking, malware.
# This hosts blockfile is built by aggregating and consolidating several reputable sources (see below).
# Installation:
# Windows: C:\Windows\System32\drivers\etc\hosts (/!\ DISABLE DNS CLIENT SERVICE !/!)
# Linux: /etc/hosts (You may need to add "127.0.0.1 %yourhostname%" at the beginning of file)
# Android: Use DNSFilter, Blokada or DNS66 (on https://f-droid.org)
# Sources:
# https://adaway.org/hosts.txt (12182 hosts)
# https://someonewhocares.org/hosts/hosts (18455 hosts)
# https://raw.githubusercontent.com/nottracking/hosts-blocklists/master/dnscrypt-proxy/dnscrypt-proxy.blacklist.txt (197877 hosts)
# https://mirror.cedia.org.ec/malwaredomains/justdomains (20856 hosts)
# https://mirror.cedia.org.ec/malwaredomains/immortal_domains.txt (3195 hosts)
# https://www.malwaredomainlist.com/hostlist/hosts.txt (1184 hosts)
# http://www.hello2002.mps.org/hosts.txt (18356 hosts)
# https://pgl.yoyo.org/adserver/serverlist.php?hostformat=hosts&showintro=1&minetype=plaintext (3352 hosts)
# https://sebsauvage.net/fofoge.fr/firstparty-only-trackers-hosts.txt (13996 hosts)
# Total: 248134 hosts
-----
# Localhost:
127.0.0.1 localhost
127.0.0.1 localhost.localdomain
127.0.0.1 local
255.255.255.255 broadcasthost
::1 localhost
::1 ip6-localhost
::1 ip6-loopback
fe80::1%eth0 localhost
ff00::0 ip6-localnet

```



Méthode : Comment ajouter une liste de blocage dans Pi-hole ?

1. Dans l'interface d'administration, accédez au menu **Group Management** → **Adlists**

2. Dans le champ prévu à cet effet en haut de la page, ajouter l'adresse de la liste voulue. Dans mon exemple, j'ajoute donc l'URL de la liste de Sebsauvage : **https://sebsauvage.net/hosts/hosts**

Je peux éventuellement ajouter un commentaire, une indication dans le 2^e champ

Je clique ensuite sur le bouton bleu **Add**.

Address	Status	Comment	Group assignment	Action
https://raw.githubusercontent.com/StevenBlack/hosts/master/hosts	Enabled	Migrated from jetc/pihole	Default	[X]
https://mirror1.malwaredomains.com/files/justdomains	Enabled	Migrated from jetc/pihole	Default	[X]
http://systcl.org/cameleon/hosts	Enabled	Migrated from jetc/pihole	Default	[X]
https://s3.amazonaws.com/lists.disconnect.me/single_tracking.txt	Enabled	Migrated from jetc/pihole	Default	[X]
https://s3.amazonaws.com/lists.disconnect.me/single_ad.txt	Enabled	Migrated from jetc/pihole	Default	[X]
https://sebsauvage.net/hosts/hosts	Enabled	Liste de blocage Sebsauvage	Default	[X]

3. Après quelques secondes d'attente, je dois avoir une notification surgir en haut de l'écran me disant que la liste a bien été rajoutée dans Pi-hole.

Je peux le constater en voyant qu'elle a été rajoutée à toutes les autres qui étaient déjà présentes par défaut.

Il ne me reste plus maintenant qu'à actualiser les listes pour qu'elle sûr que Pi-hole utilisera désormais la toute dernière version à jour de toutes ces listes.



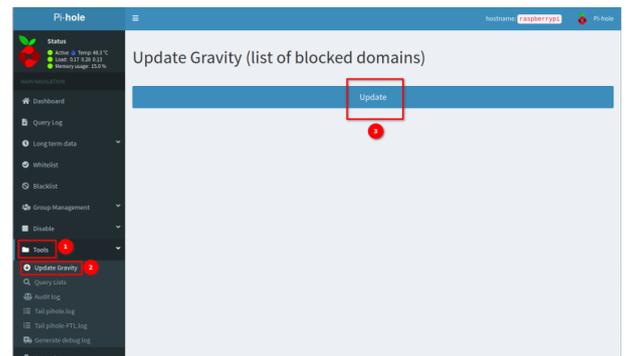
Méthode : Actualiser les listes de blocage

On vient de rajouter une liste de blocage.

Pour être sûr de posséder une version parfaitement à jour de ces listes et assurer que Pi-hole sera toujours efficace, il faut régulièrement procéder à une actualisation des listes.

1. Pour cela, je vais tout simplement :

- accéder au menu **Tools** → **Update Gravity**
- cliquer sur le bouton bleu **Update**

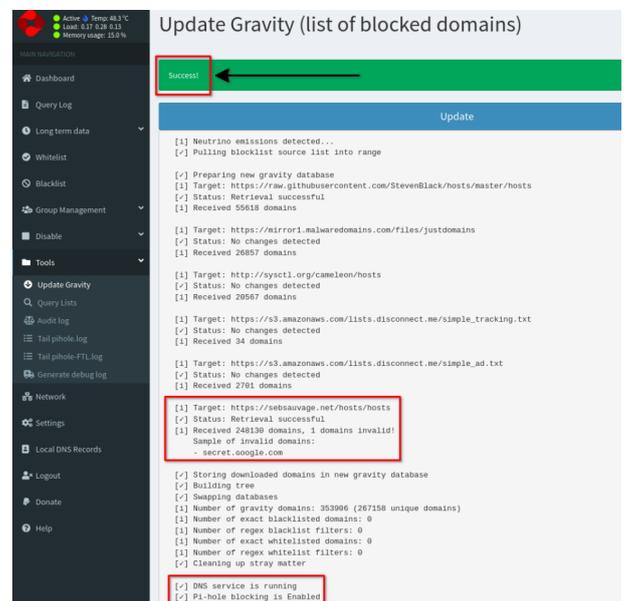


2. La mise à jour de toutes les listes va alors être lancée.

Il faudra patienter un peu, quelques secondes ou dizaines de secondes, le temps de télécharger les nouvelles versions de chaque liste.



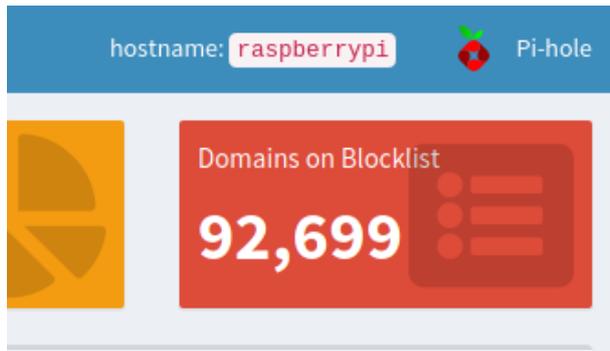
3. Après cela, on aura confirmation qu'on a bien les listes à jour et qu'elles sont effectivement utilisées par Pi-hole.



+ Complément : Nombre de domaines bloqués

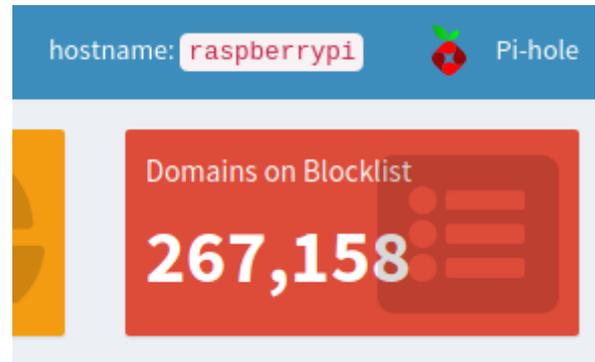
On observera également qu'une liste a été ajoutée en revenant sur le tableau de bord principal (menu **Dashboard**) et en regardant le nombre de domaines bloqués maintenant affiché.

Avant d'ajouter la liste, j'avais **92 699** domaine bloqués



Après avoir ajouté la liste de Sebsauvage, j'ai maintenant **267 158** domaines bloqués.

Mon Pi-hole vient de gagner en efficacité, ma vie privée et mes données personnelles sont un peu mieux protégées.



+ Complément : Liste des machines utilisant Pi-hole dans le réseau local

Pour **vérifier quels sont les appareils qui utilisent Pi-hole** dans votre réseau local (et s'assurer ainsi que leurs requêtes DNS sont bien filtrées), on peut se rendre sur l'entrée de menu **Network**.

On voit alors la liste des différentes machines connectées, avec dans la dernière colonne une confirmation qu'elles utilisent bien Pi-hole. Si c'est le cas, leur ligne est verte dans le tableau.

Sinon, la ligne est rouge.

IP address	Hardware address	Interface	Hostname	First seen	Last Query	Number of queries	Uses Pi-hole
192.168.1.21	e0:3b:1b:00:00:00	eth0	unknown	2020-05-10 22:40:00	2020-05-12 18:48:11	6 583	✓
127.0.0.1	N/A	N/A	localhost	2020-05-10 22:07:00	2020-05-12 18:00:04	467	✓
192.168.1.14	04:00:00:00:00:00	eth0	unknown	2020-05-11 18:13:00	2020-05-12 19:36:38	123	✓
192.168.1.1	24:00:00:00:00:00	eth0	unknown	2020-05-10 22:40:00	Never	0	✗

Showing 1 to 4 of 4 entries

Background color: Last query from this device seen ...

just now ... 24 hours ago ... > 24 hours ago Device does not use Pi-hole

Liste des machines connectées dans le réseau local et utilisant Pi-hole.

+ Complément : Paramètres de Pi-hole : choix des DNS publics

On a déjà choisi un serveur DNS public que Pi-hole pourra utiliser pour fonctionner, à l'étape 5 de l'installation de Pi-hole sur notre Raspberry.

👤 Pour rappel on a choisi, pour diverses raisons, les serveurs DNS de **Quad9** dans leur version « *Quad9 (filtered, DNSSEC)* »

Ce choix n'est pas figé, c'est celui que j'ai fait, et chacun pourra évidemment choisir d'utiliser les serveurs DNS qu'il souhaite. Pour modifier ces choix, on se rendra dans le menu **Settings** puis sur l'onglet **DNS**.

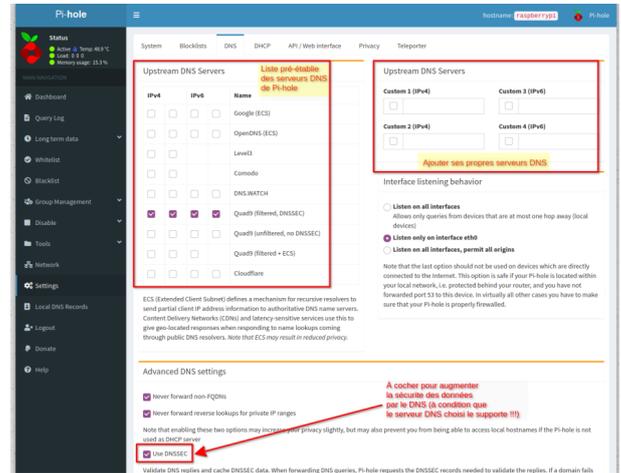
Sur cette page, **dans le premier groupe**, on pourra si besoin modifier les serveurs DNS utilisés par Pi-hole en choisissant parmi ceux qui sont présents par défaut.

Dans mon exemple, j'ai choisi dès l'installation de Pi-hole les serveurs DNS de Quad9.

→ Vérifier que les 4 cases sont bien cochées pour le serveur choisi : les deux premières cases pour le serveur principal et secondaire en IPv4, les deux autres pour le serveur principal et secondaire en IPv6.

Dans le second groupe, sur la droite de la page, on pourra cocher les cases et saisir les adresses IP de serveurs DNS autres que ceux de la première liste si on le souhaite.

→ **Si le serveur DNS choisi le supporte** (c'est le cas pour ceux de Quad9), on pourra également cocher la case « **Use DNSSEC** » un peu plus bas, pour augmenter la sécurité des données envoyées par le DNS.



Paramétrer le serveur DNS public utilisé par Pi-Hole

8. Modifier le dépôt de mises à jour

Les mises à jour du système Raspbian se font en se connectant à un serveur de mises à jour sur lequel sont stockés les paquets logiciels dans leur dernière version en cours. Ce serveur n'est pas unique, il existe de nombreux serveurs « miroirs » de façon à garantir une disponibilité d'accès permanente.

Votre Raspberry avec son système Raspbian utilise par défaut le serveur <http://mirrordirector.raspbian.org/raspbian>

! Attention :

Il se peut que vous soyez confrontés à un problème particulier de connexion à ce serveur par défaut vers lequel la commande `apt update` tente de se connecter. Depuis quelques mois il est relativement inaccessible et semble causer des problèmes aux utilisateurs : la commande de mise à jour est extrêmement longue et finit presque toujours par aboutir à des messages d'erreur.

Si c'est le cas, il va falloir **modifier le dépôt utilisé par défaut** pour les mises à jour.

⚙ Méthode : Modifier le serveur de mises à jour

Le principe de cette manipulation consiste à éditer le fichier **sources.list** (qui contient, comme son nom l'indique, la liste des sources logicielles) pour remplacer le miroir par défaut par un miroir si possible plus proche de nous.

On choisira ici, dans la liste des miroirs existants : <https://www.raspbian.org/RaspbianMirrors>

→ Par exemple, on pourra remplacer `mirrordirector.raspbian.org` par <http://raspbian.42.fr/raspbian/> hébergé par l'école de programmation 42.fr

1. Ouvrir un terminal et saisir la commande :

sudo nano /etc/apt/sources.list

Appuyer sur la touche Entrée, saisir le mot de passe utilisateur et valider.

Le fichier `sources.list` s'ouvre dans l'éditeur `nano`, intégré au terminal

On voit que la première ligne de ce fichier indique l'adresse du serveur `http://mirrordirector.raspbian.org/raspbian`

RAPPEL : la souris est inopérante dans cet éditeur, on se déplace avec les flèches du clavier pour se positionner à l'endroit voulu pour le modifier.



2. Pour empêcher l'utilisation de ce serveur sans pour autant perdre cette ligne d'instructions, on va simplement la « commenter », c'est à dire la **transformer en ligne de commentaire**, qui ne sera **pas prise en compte par le système** quand il utilisera ce fichier.

→ Pour cela, il suffit de rajouter le signe # au début de la ligne.

La première ligne devient donc :

```
#deb http://mirrordirector.raspbian.org/raspbian/ stretch main contrib non-free rpi
```

(et elle change de couleur, elle est maintenant considérée comme un commentaire).

```
pi@raspberrypi ~
GNU nano 2.7.4 Fichier: /etc/apt/sources.list
#deb http://mirrordirector.raspbian.org/raspbian/ stretch main contrib non-free rpi
# Uncomment line below then 'apt-get update' to enable 'apt-get source'
#deb-src http://archive.raspbian.org/raspbian/ stretch main contrib non-free rpi
Aide  ecrire  Chercher  Couper  Justifier  Pos. cur.
Quitter  Lire fich.  Remplacer  Coller  Orthograp.  Aller lig.
```

3. En laissant le curseur positionné sur le # qu'on vient de rajouter, on va appuyer sur la touche *Entrée* du clavier de façon à **insérer une nouvelle ligne au début de ce fichier**, au dessus de la précédente.

→ Saisir sur cette nouvelle ligne les informations suivantes :

```
deb http://raspbian.42.fr/raspbian stretch main contrib non-free rpi
```

C'est cette ligne d'instructions qui sera désormais prise en compte par le système. Elle est identique à la précédente SAUF pour l'adresse du serveur.

```
pi@raspberrypi ~
GNU nano 2.7.4 Fichier: /etc/apt/sources.list
deb http://raspbian.42.fr/raspbian stretch main contrib non-free rpi
#deb http://mirrordirector.raspbian.org/raspbian/ stretch main contrib non-free rpi
# Uncomment line below then 'apt-get update' to enable 'apt-get source'
#deb-src http://archive.raspbian.org/raspbian/ stretch main contrib non-free rpi
Aide  ecrire  Chercher  Couper  Justifier  Pos. cur.
Quitter  Lire fich.  Remplacer  Coller  Orthograp.  Aller lig.
```

4. **Enregistrer la modification** avec la combinaison de touches **Ctrl + O** au clavier et **valider par la touche Entrée**.

Quitter l'éditeur nano avec la combinaison de touches **Ctrl + X**.

On se retrouve dans la fenêtre du terminal, comme au début.

Appuyer sur les touches Ctrl + X pour quitter.

Appuyer sur les touches Ctrl + O pour « écrire » les modifications dans le fichier.

```
Aide  ecrire  Chercher  Couper  Justifier  Pos. cur.
Quitter  Lire fich.  Remplacer  Coller  Orthograp.  Aller lig.
```

5. Dans ce terminal, on va saisir la commande permettant de lancer la recherche de mises à jour :

```
sudo apt update
```

On constate bien que cette recherche de mises à jour se fait maintenant en se connectant au site *http://raspbian.42.fr/raspbian/* à la place du serveur par défaut.

```
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent permitted by applicable law.
Last login: Tue Dec 26 16:47:30 2017 from 192.168.1.15
pi@raspberrypi:~$ sudo nano /etc/apt/sources.list
pi@raspberrypi:~$ sudo apt update
Reception de:1 http://raspbian.42.fr/raspbian stretch InRelease [15,0 kB]
Reception de:2 http://archive.raspberrypi.org/debian stretch InRelease
15,0 ko Reception en 1s (12,2 ko/s)
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances
Lecture des informations d'état... Fait
All packages are up to date.
pi@raspberrypi:~$
```

IV. Pour aller plus loin...

1. 3 sites à connaître

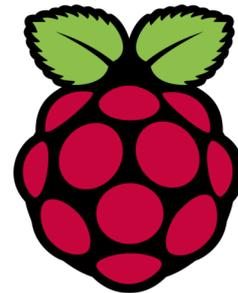
Pour ceux qui voudront explorer, tester, bidouiller, créer, apprendre davantage de choses avec ces Raspberry Pi, voici une sélection de **3 sites à peu près incontournables**.

→ Chacun de ces trois sites propose un **forum pour les utilisateurs**, où vous pourrez poser des questions qui se poseront inévitablement et obtenir des réponses précises et adaptées.

2. Raspberrypi.org

<https://www.raspberrypi.org/>

« We provide outreach and education to help more people access computing and digital making. We develop free resources to help people learn about computing and how to make things with computers, and train educators who can guide other people to learn. »



3. Raspbian-france.fr

<https://raspbian-france.fr/>

« Raspbian France est un site visant à la création de tutoriels en français autour de Raspbian, et de la Raspberry Pi de façon plus générale.

Notre but est de vous fournir des tutoriels de qualité, et surtout toujours simples, afin de répondre aux questions que nous nous sommes posées, et que vous aussi vous poserez donc peut-être. »



4. Framboise314.fr

<http://www.framboise314.fr>

Framboise 314, le Raspberry Pi à la sauce française...



GLOSSAIRE

DoH : DNS over HTTPS

DNS over HTTPS (DoH) est un protocole permettant d'effectuer une résolution DNS (Domain Name System) à distance via le protocole HTTPS. L'un des objectifs de la méthode est d'accroître la confidentialité et la sécurité des utilisateurs en empêchant les écoutes clandestines et la manipulation des données DNS par des attaques de type man-in-the-middle : toutes les requêtes DNS sont chiffrées et ne peuvent donc pas être interceptées lorsqu'elles circulent sur Internet.

Source : https://fr.wikipedia.org/wiki/DNS_over_HTTPS

En savoir plus : <https://www.nextinpact.com/news/108780-quest-ce-que-dns-over-https-doh-quest-ce-que-cela-peut-vous-apporter.htm>

La commande Curl

La commande **curl**, pour **cURL** (abréviation de « *client URL request library* » : « bibliothèque de requêtes aux URL pour les clients » ou see URL : « voir URL ») est un outil Open Source, c'est une interface en ligne de commande, destinée à récupérer le contenu d'une ressource accessible par un réseau informatique.

C'est le couteau suisse de tous les téléchargements de contenus en ligne de commande. Curl fonctionne sous Linux, Windows et MacOS.

Source : <https://fr.wikipedia.org/wiki/CURL>