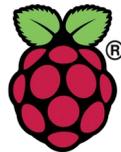




Mise en place d'un serveur RASPBERRY PI



DeltaLab

Espace Maison Milon
2 Place E.Colongin
84600 Grillon

deltalabprototype.fr

Table des matières

I.	Pré-requis	02
II.	Montage du boîtier	03
III.	Installation de la distribution	05
IV.	Se Connecter à la Raspberry pi	
	1. Connexion SSH	07
	2. Connexion USB	10
V.	Changer les paramètres	
	1. Id , mot de passe , hostname	11
	2. Réglages réseaux	12
	3. Pare-Feu	13
	4. Logging	14
VI.	Mise en place du serveur Node-RED	15

I. Pré-requis

Pour pouvoir suivre ce tutoriel, vous aurez besoin des éléments suivants :

◆ Matériel :

- Raspberry Pi 2 ou 3
- Boîtier en aluminium
- Carte micro SD d'au moins 32Go
- Alimentation 5V micro USB
- Câble Ethernet et une connexion à internet
- un PC

◆ Logiciel :

- Un distribution **Raspbian Stretch Lite**
- Le logiciel **balenaEtcher**
- Un client SSH , comme **PuTTY** ou **OpenSSH**

Il est évidemment possible de trouver des raspberry pi déjà montées dans leur support, ou possédant déjà un OS de départ. Ignorez les parties suivantes si vous êtes dans ces cas.

II. Montage du boîtier

Le boîtier en aluminium utilisé est un eleduino.

Il fournit est avec un tournevis et deux « radiateurs ».



Visser la carte sur le support puis déposer les « radiateurs » après avoir enlevé le papier



Emboîter la partie supérieure en faisant attention à l'emplacement des ports



Visser les deux parties métalliques noires en mettant les marquages à l'extérieur



Nota Bene : Le montage en lui-même est semblable , mais les boîtiers de raspberry pi 2 et 3 sont très différents en terme de ports. Ne vous trompez pas.

III. Installation de la distribution

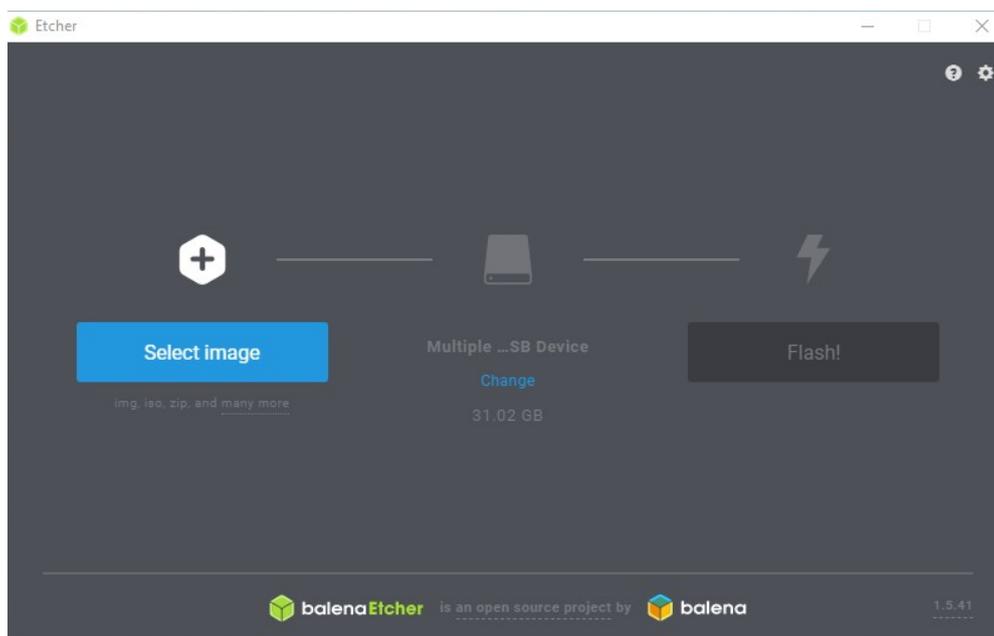
La distribution utilisée est **Raspberry pi OS Lite**. C'est une distribution récente, qui succède à Raspbian. Elle ne propose aucune interface graphique, on utilise des commandes pour naviguer.

Vous pouvez trouver l'image de la distribution sur le site officiel Raspberrypi
(<https://raspberrypi.fr/telechargements/>)

Maintenant il vous faut flasher la carte micro SD pour installer la distribution, il faudra installer le logiciel **balenaEtcher**. C'est un logiciel qui permet de graver des images sur des disques, clé USB, carte SD et micro SD.

Utiliser un lecteur de carte micro SD branchée en USB avec l'ordinateur

Une fois le logiciel installé, démarrez le et sélectionnez **Raspberry pi OS** et la micro SD puis appuyez sur « Flash ! ».



Une fois terminé, vous devrez ouvrir le boot puis ajouter un fichier “**ssh.txt**” et supprimer l’extension **.txt**. Ce fichier permet d’activer le serveur SSH de la carte. Pour créer un fichier **.txt** faites un clic droit dans le dossier dans une zone vide puis cliquez sur **nouveau** et **document texte**. Puis renommer le “**ssh**”.

Pour afficher l’extension des fichier cliquez sur **affichage** en haut de la fenêtre puis cocher la case **extensions de noms de fichiers**.

Si vous ne pouvez pas supprimer l’extension, cliquer sur la flèche en haut à droite puis sur **propriétés** > **avancés** et cocher la case **autoriser l’indexation**.

Un fois l’extension **.txt** supprimée dans le nom du fichier, validé le changement. Il ne reste plus qu’à insérer la micro SD dans votre carte puis brancher le câble d’alimentation et le câble internet.

Pour savoir si votre carte est bien alimentée, regarder sur le côté si la led rouge est allumée.



IV. Se connecter à la Raspberry pi

1. Connexion en SSH

SSH(Secure Shell) est un protocole de communication qui permet de chiffrer les segments TCP et rendre la communication sécurisée.

Pour pouvoir communiquer en SSH il faut un serveur (la carte) et un client (le PC). Le serveur est déjà installé sur la carte mais il faut installer un client sur le PC. Pour le client vous pouvez utiliser deux logiciels : **PuTTY** ou **OpenSSH**

OpenSSH est un client disponible directement sur Windows 10 et qui utilise l'invite de commande ou Powershell.

PuTTY est un émulateur de terminal et un client disponible sur internet qui utilise sa propre interface.

Si vous êtes sous Windows 10, je vous recommande d'utiliser OpenSSH mais si vous êtes sur une version antérieure de Windows ou sur une autre distribution, installez plutôt PuTTY.

Avant de commencer vous devez trouver l'adresse IP de votre carte pour pouvoir vous connecter en SSH.

Pour ce faire, ouvrez le menu démarrer, tapez **cmd** et cliquez sur **invite de commandes**. Ensuite tapez : **ping raspberrypi**.

La commande devrait vous retourner un adresse comme ci-dessous.

```
cs Invite de commandes
Microsoft Windows [version 10.0.17134.829]
(c) 2018 Microsoft Corporation. Tous droits réservés.

C:\Users\user>ping raspberrypi

Envoi d'une requête 'ping' sur raspberrypi.localdomain [192.168.1.141] avec 32 octets de données :
Réponse de 192.168.1.141 : octets=32 temps<1ms TTL=64

Statistiques Ping pour 192.168.1.141:
    Paquets : envoyés = 4, reçus = 4, perdus = 0 (perte 0%),
Durée approximative des boucles en millisecondes :
    Minimum = 0ms, Maximum = 0ms, Moyenne = 0ms

C:\Users\user>
```

Installer et se connecter avec OpenSSH :

Pour installer OpenSSH il faut cliquer sur le menu démarrer puis sur **paramètres**. Ensuite allez dans **Applications** et ensuite sur **Gérer les fonctionnalités facultatives**.

Puis cliquer sur **Ajouter une fonctionnalité**, chercher **client OpenSSH**, puis appuyez sur **installer**. Redémarrer le pc et ouvrez l'invite de commandes ou Powershell et entrez **ssh** pour vérifier l'installation.

La commande devrait vous retourner ce message.

```
Windows PowerShell
Copyright (C) Microsoft Corporation. Tous droits réservés.

PS C:\Users\Delta Lab 2> ssh
usage: ssh [-46AaCfGgKkMnqsTtVvXxYy] [-b bind_address] [-c cipher_spec]
          [-D [bind_address:]port] [-E log_file] [-e escape_char]
          [-F configfile] [-I pkcs11] [-i identity_file]
          [-J [user@]host[:port]] [-L address] [-l login_name] [-m mac_spec]
          [-O ctl_cmd] [-o option] [-p port] [-Q query_option] [-R address]
          [-S ctl_path] [-W host:port] [-w local_tun[:remote_tun]]
          destination [command]
PS C:\Users\Delta Lab 2> *
```

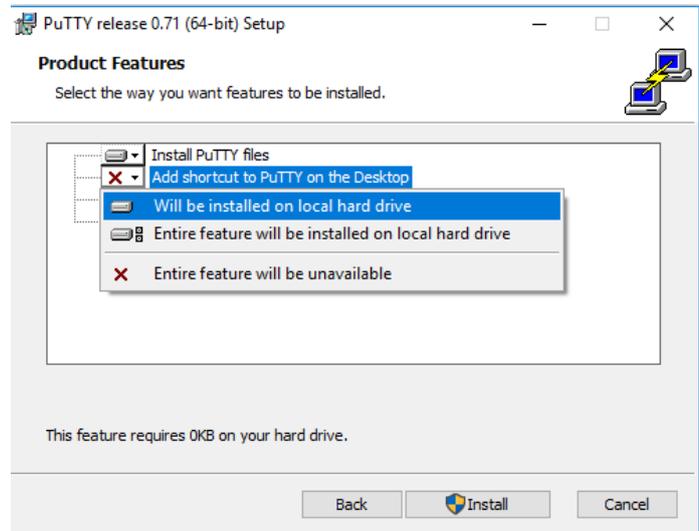
Ensuite tapez : **ssh <nom d'utilisateur>@<l'adresse IP>** pour vous connecter

Le nom d'utilisateur par défaut est **pi** , et son mot de passe est **raspberrypi**

Installer et se connecter avec Putty :

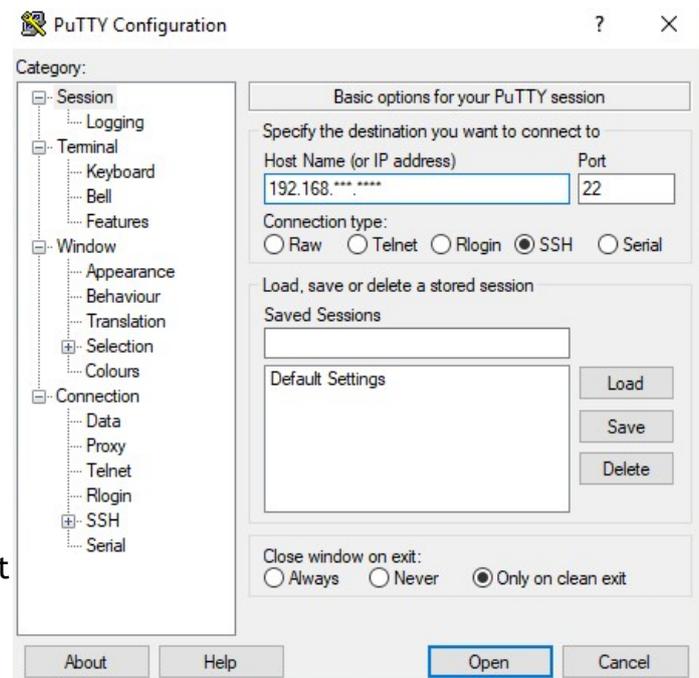
Vous devez télécharger l'installateur de PuTTY sur le site : <https://www.putty.org/>

Lors de l'installation, quand vous arrivez dans "Product Features", cliquez sur "Add shortcut" et ensuite sur "Will be installed on local hard drive".



Une fois le logiciel installé, lancez le. vous arriverez sur cette fenêtre comme ci-dessous. Dans la partie "Host name" tapez l'adresse de la carte. Vérifiez que la case SSH soit bien cocher et que le numéro de "Port" soit bien 22.

Open vous permettra de vous connecter. L'utilisateur par défaut est pi et son mot de passe est raspberry

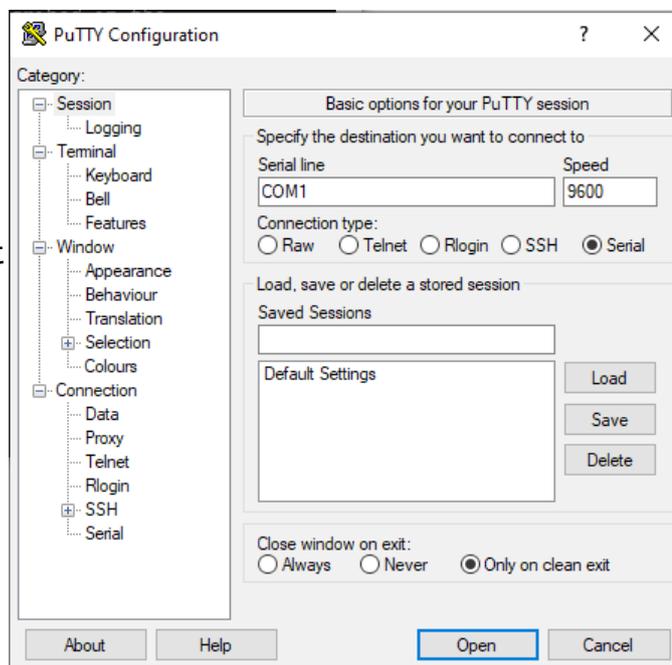


2. Connexion via USB

Vous pouvez également vous connecter à votre raspberry pi via USB. Pour ce faire, reliez votre raspberry pi et votre PC via USB, et utilisez PuTTY.

Choisissez l'option **Serial**, entrez le **port** sur lequel la carte est branchée et réglez la **vitesse**. (par défaut, la vitesse recommandée est de **115200**).

Pour savoir quel port est utilisé, ouvrez le gestionnaire de périphériques de votre PC et regardez dans «ports série»



L'utilisateur et le mot de passe par défaut sont les mêmes qu'avec SSH : **pi** et **raspberry**.

V. Changer les paramètres

1. Nom d'utilisateur , mot de passe et nom d'hôte

Pour changer votre nom d'utilisateur, vous devez vous connecter en root.

- Activez le compte root : `sudo passwd root`. Entrez le mdp de votre choix.
- Déconnectez-vous de votre compte : `logout`
- Connectez-vous en tant que root : id = root , mdp = celui que vous avez mis.
- Changez le nom de votre compte : `usermod -l nouveau_nom ancien_nom`
- Déconnectez-vous de root et reconnectez vous à votre compte

Pour changer votre mot de passe , entrez la commande : `passwd`

Pour changer le nom d'hôte du serveur , entrez : `sudo nano /etc/hostname` et modifiez le fichier en entrant le nom d'hôte que vous voulez.

2. Réglages réseaux

➔ Changer votre IP Ethernet

Entrez la commande `sudo nano /etc/dhcpd.conf`

Cherchez la ligne “Example static IP configuration” puis rentrez vos adresses et enlevez les hashtags à “interface eth0” et aux adresses.

```
# Example static IP configuration:
interface eth0
static ip_address=192.168.1.45/24
#static ip6_address=fd51:42f8:caae:d92e::ff/64
static routers=192.168.1.1
static domain_name_servers=192.168.1.254
```

Utilisez la commande `ip address` pour voir la liste de vos @ip.

Vous pouvez tenter de modifier vos @ip avec la commande `ip address <add/replace/change> @ip/mask dev <interface>` mais elle ne marche pas avec toutes les distributions.

➔ Mettre en place une connexion WiFi

Pour changer les paramètres tapez : **sudo raspi-config**

Le menu s'affiche comme ci-dessous.

```
Raspberry Pi Software Configuration Tool (raspi-config)
1 Change User Password Change password for the current 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 location
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>
```

Selectionnez Network Options , puis suivez les instructions :

- selectionnez le pays dans lequel vous êtes (à priori France)
- entrez le ssid (le nom) du réseau WiFi auquel vous voulez accéder
- entrez le mot de passe / la clé de sécurité du réseau WiFi

Dans certaines distributions, le WiFi se mettra en veille de lui-même. Pour éviter ça ,ouvrez le fichier `/etc/rc.local` ([nano /etc/rc.local](#)) et ajoutez-y la ligne :

```
/sbin/iw wlan0 set power_save off
```

Pour changer l'@ip utilisé par le WiFi, vous pouvez utiliser les mêmes possibilités qu'avec l'@ip ethernet , en spécifiant bien **wlan0** comme interface.

3. Configurer le pare-feu

Nous allons utiliser **UFW** et **Fail2ban**.

UFW est une application qui simplifie la configuration du pare-feu.

Fail2ban est une application qui bannit les IP des machines qui essaient une attaque sur le système.

Pour l'installer les deux applications, tapez :

```
sudo apt install ufw -y && sudo apt install fail2ban -y
```

Pour configurer le pare-feu, il faudra utiliser les commandes suivantes :

Pour autoriser une connexion en entrée tapez : **sudo ufw allow [protocole]**

Pour interdire une connexion en entrée tapez : **sudo ufw deny [protocole]**

Pour autoriser une connexion en sortie tapez : **sudo ufw allow out [protocole]**

Pour interdire une connexion en sortie tapez : **sudo ufw deny out [protocole]**

Pour le moment vous allez ajouter trois protocoles :

- **22** (SSH)
- **1880** (HTTP)
- **1883** (Broker MQTT)

Ils seront autorisés à se connecter en entrée et en sortie.

Une fois paramétré, démarrez le pare-feu en tapant : **sudo ufw enable**

Pour voir les règles et l'état du pare-feu tapez : **sudo ufw status verbose**

```
pi@raspberrypi:~ $ sudo ufw status verbose
Status: active
Logging: on (low)
Default: deny (incoming), deny (outgoing), disabled (routed)
New profiles: skip

To Action From
--
22 ALLOW IN Anywhere
1880 ALLOW IN Anywhere
1883 ALLOW IN Anywhere
22 (v6) ALLOW IN Anywhere (v6)
1880 (v6) ALLOW IN Anywhere (v6)
1883 (v6) ALLOW IN Anywhere (v6)

pi@raspberrypi:~ $
```

4. Paramétrer le logging (fichiers de journaux)

Vous pouvez accéder au log du système grâce à la commande **journalctl** .

- **journalctl -xe** : pour le voir depuis la fin
- **journalctl status <service>** pour voir le log d 'un service en particulier.

Vous pouvez paramétrer le logging en accédant au fichier de configuration.
Entrez **sudo nano /etc/systemd/journald.conf** pour ouvrir le fichier.
Retirez les # des lignes que vous voulez activer.

Lignes utiles :

- **SystemMaxFileSize** : taille max des fichiers de log. Vous devez renseigner l'unité après la valeur (k - kiloOctet , M - MégaOctet , G - GigaOctet)
- **SystemMaxFiles** : nombre de fichiers de log stockés
- **MaxRetentionSec** : durée pendant laquelle le log est conservé. Vous devez renseigner l'unité après la valeur (month , day , week , year)
- **MaxLevelStore** : niveau maximal des messages stockés (debug, warning, error, notice, ...)

VI. Mise en place du serveur Node-RED

1. Installer Node-RED

Pour installer Node-RED, tapez : **sudo apt-get install build-essential**
puis copier-coller la commande d'installation :

```
bash <(curl -sL  
https://raw.githubusercontent.com/node-red/linux-installers/master/deb/  
update-nodejs-and-nodered)
```

Une fois installé, tapez : **node-red-start**

Une fois lancé, appuyez sur **ctrl+c** pour revenir sur le terminal.

Si vous voulez arrêter Node-RED, tapez: **node-red-stop**

Pour que Node-RED se lance automatiquement au démarrage de la carte,
tapez: **sudo systemctl enable nodered.service**

2. Installer un broker MQTT

Un broker MQTT permet une communication optimisée communication entre les objets connectés tout en économisant la batterie des appareils autonomes.
La base de données permet de stocker des données dans des tableaux qui peuvent être filtrer.

Pour installer le broker MQTT **Mosquitto** tapez : **sudo apt-get install mosquitto -y**

Pour le démarrer : **mosquitto**

MQTT utilise par défaut le port **1883**. Pour en changer, arrêtez MQTT et relancez le avec un port précis : **mosquitto -p <port>**