

La carte Raspberry

K. Boudjelaba

BTS SN – EC-IR, 1ère année



Table des matières

Définitions

Caractéristiques du Raspberry Pi 3-Modèle B+

Installation du système d'exploitation

Précautions à prendre

Les ports GPIO

La bibliothèque Wiring Pi (C++)

Connection en SSH au Raspberry Pi pour le contrôler depuis un ordinateur

Connection à distance via VNC (Virtual Network Computing)

Création d'un NAS

Connection au serveur NAS

Définitions

Système embarqué

Un système embarqué est un système complexe qui intègre du logiciel et du matériel conçus ensemble afin de fournir des fonctionnalités données. Il contient généralement un ou plusieurs microprocesseurs.

Les systèmes embarqués fonctionnent généralement en Temps Réel : les opérations de calcul sont alors faites en réponse à un événement extérieur (interruption matérielle) ou à un programme.

Les ordinateurs embarqués sous le système d'exploitation Linux sont massivement présents dans les technologies modernes (transports, multimédia, téléphonie mobile ...).

Définitions

Raspberry Pi

Le Raspberry Pi est un nano-ordinateur équipé d'un processeur ARM. Il permet l'exécution de plusieurs variantes du système d'exploitation libre GNU/Linux (Ex. Debian) sur carte SD destiné à des applications d'informatique embarquée.

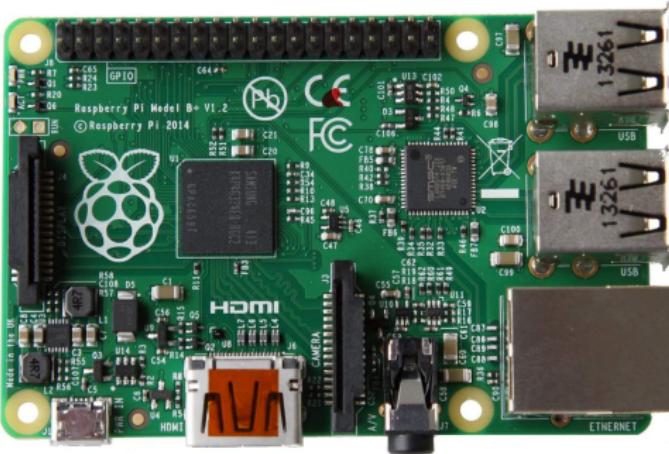


Figure 1: Carte Raspberry Pi

Attention: il s'agit de matériel fragile, à manipuler avec précaution.
Le Raspberry Pi n'est pas une carte Arduino.

Connectivité

- Sans fil : Une puce réseau combinant du Bluetooth 4.2 BLE et du Wi-Fi Dual Band b/g/n/ac pour plus de rapidité et de moins de latences.
- Filaire : Un port Gigabit Ethernet.

Caractéristiques

- Processeur ARM Cortex A53
- Lecteur de carte micro-SD
- Header GPIO 40 broches
- Support du boot PXE
- Port caméra CSI pour connecter la caméra Raspberry Pi
- Port d'affichage DSI pour connecter l'écran tactile Raspberry Pi
- 4 ports USB 2.0
- 1 prise jack
- Alimentation 5V / 2.5A (transformateur non fourni).

Installation du système d'exploitation

Installation du système d'exploitation

Raspbian

Raspbian (actuellement appelé Raspberry Pi OS) est un système d'exploitation libre basé sur la distribution GNU/Linux Debian, et optimisé pour la carte Raspberry Pi.

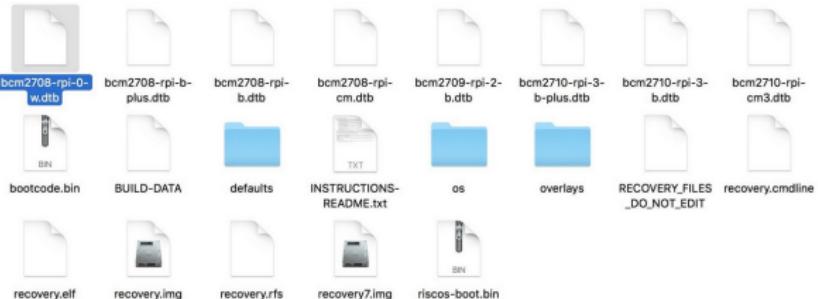
Raspbian ne fournit pas simplement un système d'exploitation basique, il est aussi livré avec plus de 35 000 paquets, c'est-à-dire des logiciels pré-compilés livrés dans un format optimisé, pour une installation facile sur la carte Raspberry Pi via les gestionnaires de paquets.

Par ailleurs, en tant que distribution dérivée de Debian, il répond à la majeure partie de la très vaste documentation de Debian.

Installation du système d'exploitation

Étapes à suivre

- ▶ Carte microSD : la carte doit être formatée en FAT32 et de capacité minimale de 8 Go (cette carte SD sera le disque de démarrage de la carte Raspberry Pi)
- ▶ Télécharger [NOOBS \(NOOBS_v3_8_0.zip\)](#) sur le site officiel de Raspberry Pi - rubrique Download ou [NOOBS \(choisir la dernière version\)](#)
- ▶ Extraire les fichiers de l'archive .zip de Noobs. Ouvrir le fichier .zip de NOOBS téléchargé.
- ▶ Copier le contenu de l'archive de Noobs sur la carte SD formatée.



Copier tous ces fichiers à la racine de la carte SD.

Installation du système d'exploitation

Étapes à suivre (suite)

- ▶ Installer Raspbian grâce à Noobs.
 1. Insérer la carte SD contenant Noobs dans le Raspberry Pi.
 2. Connecter un clavier, une souris, un écran aux ports USB du Raspberry.
 3. Brancher l'alimentation électrique micro USB. Cette action met sous tension et démarre automatiquement le Raspberry Pi.
 4. Raspberry Pi devrait booter sur l'utilitaire NOOBS et afficher une liste d'OS disponibles à installer.
 5. Sélectionner l'OS à installer : Choisir Raspbian Full [RECOMMENDED] et cliquer sur "Install" ou appuyer sur le raccourci "i".
- ▶ Changer la langue du clavier si nécessaire. Menu → Préférences → clavier et souris
- ▶ On peut connecter la carte à internet en filaire ou en wifi si nécessaire ...

Important : Précautions à prendre

Précautions à prendre

Important

1. Connecter un clavier, une souris, un écran aux ports USB du Raspberry.
2. Pour allumer le Raspberry, il suffit de brancher son alimentation (munie d'un connecteur micro USB) au secteur.
3. Une fois le Raspberry branché au secteur, ne pas le débrancher sous aucun prétexte. Il faut attendre le démarrage complet du Raspberry pour l'éteindre.
4. Pour éteindre le Raspberry, soit :
 - ▶ En cliquant sur l'icône Raspberry (framboise) dans le coin supérieur gauche de l'écran et en choisissant Déconnexion (Logout). Dans le menu suivant, on peut choisir Arrêter (Shutdown), Redémarrer (Reboot) ou Déconnexion (Logout). Voir la figure 2. Le Raspberry est éteint lorsque la LED rouge est éteinte.

Précautions à prendre

Important



Figure 2: Etapes à suivre pour éteindre le Raspberry Pi

- ▶ Ou en ouvrant le terminal et en tapant : `sudo shutdown -h now`
5. Quand le Raspberry est complètement éteint, on peut débrancher l'alimentation du secteur.

Les ports GPIO

Les ports GPIO

La Raspberry Pi possède des ports appelés GPIO (General Purpose Input/Output, Entrées/Sorties pour un usage générique). Ces ports mettent à notre disposition de nombreuses entrées-sorties, une alimentation, un port SPI, un port I2C, et un port série.

Les figures 3 et 4 montrent le port GPIO de la carte.

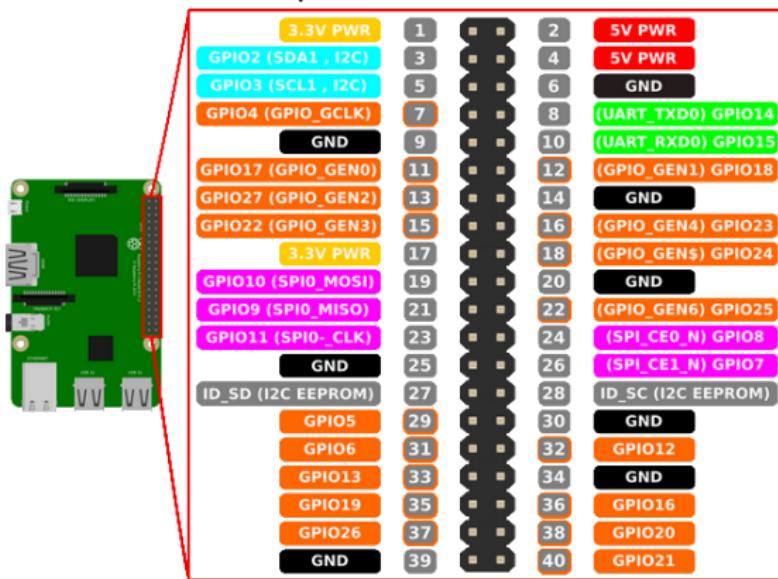


Figure 3: Les ports GPIO

Les ports GPIO

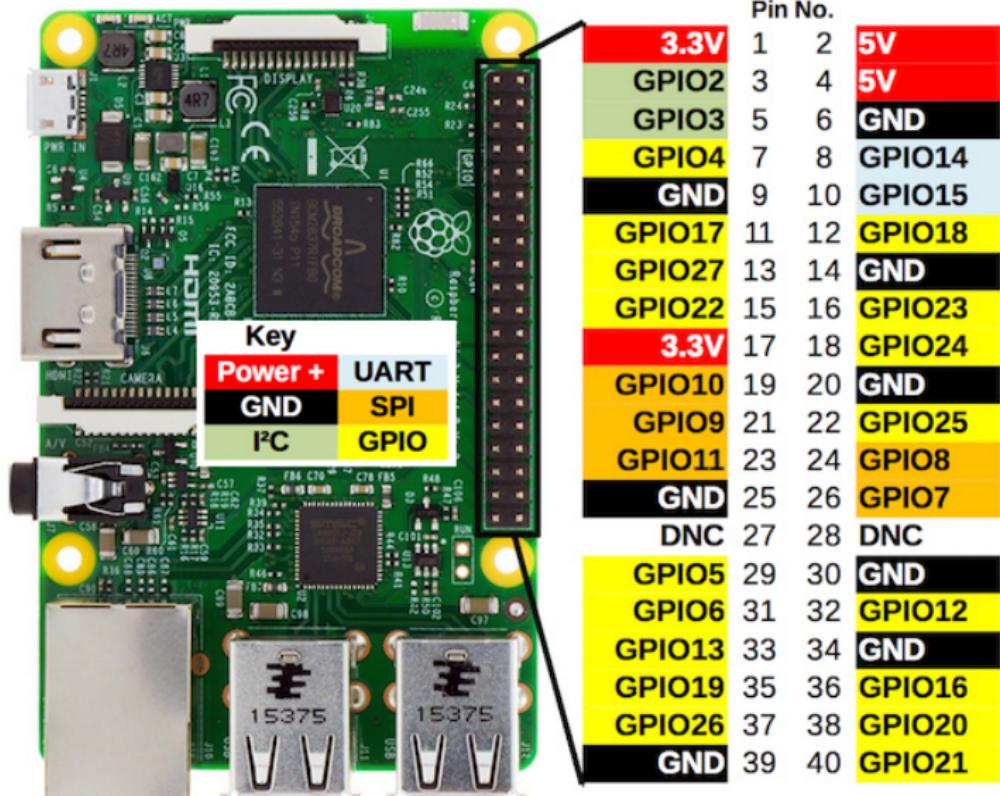


Figure 4: Les ports GPIO

Les ports GPIO

Les cartes Raspberry Pi 3 - modèle B+ disposent de 40 connectiques GPIO, qui se divisent en différentes catégories avec des usages spécifiques.

- ▶ Power + : Alimentation électrique continue
- ▶ GND : Ground (masse, retour à la terre)
- ▶ I2C : Protocole série synchrone
- ▶ SPI : Autre protocole série synchrone
- ▶ UART : Protocole asynchrone + conversion communication série vers parallèle
- ▶ GPIO : Port général qui peut éventuellement être transformé en I2C ou SPI.

Les ports GPIO sont numérotés de 1 à 40, en partant du haut à gauche quand on tient la Raspberry pi ports GPIO à droite. C'est le mode de numérotation dit "BOARD". Un autre mode de numérotation existe, qui repose sur l'adressage processeur, appelé mode "BCM".

La bibliothèque Wiring Pi (C++)

La bibliothèque Wiring Pi (C++)

Le langage le plus largement utilisé sur Raspberry PI est Python mais, étant un micro-ordinateur, il est possible de l'utiliser avec d'autres langages comme le langage C/C++ et de piloter le Raspberry Pi en C/C++ (Comme Arduino par exemple).

Lors de l'utilisation de GPIO en Python, l'utilisateur importe la bibliothèque GPIO, crée certaines variables, configure les broches, lance le programme et le GPIO commence à fonctionner.

Pour C++, la tâche n'est pas aussi simple et repose sur des flux de fichiers complexes. Heureusement, il existe une bibliothèque appelée "Wiring Pi" qui rend l'interfaçage avec GPIO beaucoup plus facile (cette bibliothèque, par défaut, est fournie avec Raspbian).

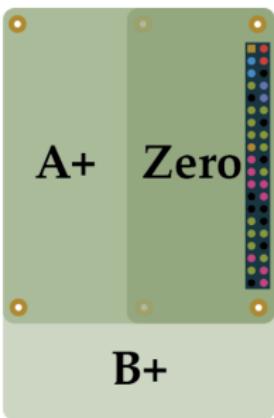
Numérotation des pin (broches) dans Wiring Pi

Le numéro de broche dans Wiring Pi n'est pas simple car le concepteur de la bibliothèque a utilisé un schéma de numérotation de câblage qui ne fait référence ni au numéro GPIO ni au numéro de broche. Car les premières Raspberry ont 8 broches GPIO utilisables numérotées de 0 à 8 et les versions ultérieures ont beaucoup plus de GPIO avec leurs noms et leur numérotation modifiés.

Les figures 5 et 6 donnent les numéros de broches des ports GPIO.

La bibliothèque Wiring Pi (C++)

| | | | |
|-----------------------|----|----|-----------------------|
| 3v3 Power | 1 | 2 | 5v Power |
| GPIO 2 (WiringPi 8) | 3 | 4 | 5v Power |
| GPIO 3 (WiringPi 9) | 5 | 6 | Ground |
| GPIO 4 (WiringPi 7) | 7 | 8 | GPIO 14 (WiringPi 15) |
| Ground | 9 | 10 | GPIO 15 (WiringPi 16) |
| GPIO 17 (WiringPi 0) | 11 | 12 | GPIO 18 (WiringPi 1) |
| GPIO 27 (WiringPi 2) | 13 | 14 | Ground |
| GPIO 22 (WiringPi 3) | 15 | 16 | GPIO 23 (WiringPi 4) |
| 3v3 Power | 17 | 18 | GPIO 24 (WiringPi 5) |
| GPIO 10 (WiringPi 12) | 19 | 20 | Ground |
| GPIO 9 (WiringPi 13) | 21 | 22 | GPIO 25 (WiringPi 6) |
| GPIO 11 (WiringPi 14) | 23 | 24 | GPIO 8 (WiringPi 10) |
| Ground | 25 | 26 | GPIO 7 (WiringPi 11) |
| GPIO 0 (WiringPi 30) | 27 | 28 | GPIO 1 (WiringPi 31) |
| GPIO 5 (WiringPi 21) | 29 | 30 | Ground |
| GPIO 6 (WiringPi 22) | 31 | 32 | GPIO 12 (WiringPi 26) |
| GPIO 13 (WiringPi 23) | 33 | 34 | Ground |
| GPIO 19 (WiringPi 24) | 35 | 36 | GPIO 16 (WiringPi 27) |
| GPIO 26 (WiringPi 25) | 37 | 38 | GPIO 20 (WiringPi 28) |
| Ground | 39 | 40 | GPIO 21 (WiringPi 29) |



- **GPIO** (General Purpose IO)
- **SPI** (Serial Peripheral Interface)
- **I²C** (Inter-integrated Circuit)
- **UART** (Universal Asynchronous Receiver/Transmitter)
- **PCM** (Pulse Code Modulation)
- **Ground**
- **5V** (Power)
- **3.3V** (Power)

Figure 5: Numérotation des pin (broches) dans Wiring Pi

La bibliothèque Wiring Pi (C++)

| BCM | WiringPi | Name | Physical | Name | WiringPi | BCM |
|-----|----------|---------|----------|------|----------|-----|
| | | 3.3v | 1 | 2 | 5v | |
| 2 | 8 | SDA.1 | 3 | 4 | 5V | |
| 3 | 9 | SCL.1 | 5 | 6 | 0v | |
| 4 | 7 | 1-Wire | 7 | 8 | TxD | 15 |
| | | 0v | 9 | 10 | RxD | 16 |
| 17 | 0 | GPIO.0 | 11 | 12 | GPIO.1 | 1 |
| 27 | 2 | GPIO.2 | 13 | 14 | 0v | 18 |
| 22 | 3 | GPIO.3 | 15 | 16 | GPIO.4 | 4 |
| | | 3.3v | 17 | 18 | GPIO.5 | 5 |
| 10 | 12 | MOSI | 19 | 20 | 0v | 24 |
| 9 | 13 | MISO | 21 | 22 | GPIO.6 | 6 |
| 11 | 14 | SCLK | 23 | 24 | CEO | 10 |
| | | 0v | 25 | 26 | CE1 | 8 |
| 0 | 30 | SDA.0 | 27 | 28 | SCL.0 | 11 |
| 5 | 21 | GPIO.21 | 29 | 30 | 0v | 7 |
| 6 | 22 | GPIO.22 | 31 | 32 | GPIO.26 | 31 |
| 13 | 23 | GPIO.23 | 33 | 34 | 0v | 12 |
| 19 | 24 | GPIO.24 | 35 | 36 | GPIO.27 | 27 |
| 26 | 25 | GPIO.25 | 37 | 38 | GPIO.28 | 16 |
| | | 0v | 39 | 40 | GPIO.29 | 28 |
| | | 0v | | | | 20 |
| | | 0v | | | | 21 |
| BCM | WiringPi | Name | Physical | Name | WiringPi | BCM |

Figure 6: Numérotation des pin (broches) dans Wiring Pi

Connection en SSH au Raspberry Pi pour le contrôler depuis un ordinateur

Connection en SSH au Raspberry Pi pour le contrôler depuis un ordinateur

SSH

SSH (Secure SHell) désigne à la fois un logiciel et un protocole de communication informatique. Ce protocole possède par ailleurs la particularité d'être entièrement chiffré (toutes les commandes exécutées via SSH sont totalement secrètes).

SSH a été créé pour permettre la prise de contrôle à distance d'une machine à travers une interface en lignes de commande.

Installation

L'installation de SSH se découpe en deux parties. En effet, on a besoin d'un côté d'un serveur SSH sur le Raspberry Pi et d'un autre d'un client SSH sur l'ordinateur. Le premier recevra les commandes à lancer tandis que le second les enverra.

Connection en SSH au Raspberry Pi pour le contrôler depuis un ordinateur

Activer le serveur SSH sur le Raspberry Pi : méthode 1

- Activer SSH depuis le Raspberry Pi, dans le cas où on dispose d'un écran et d'un clavier.

Ouvrir un terminal et taper la commande suivante :
`sudo raspi-config`



Figure 7: raspi-config

Aller dans la partie "Interfacing Options" et choisir la ligne SSH, puis "Yes".

Connection en SSH au Raspberry Pi pour le contrôler depuis un ordinateur

Activer le serveur SSH sur la Raspberry Pi : méthode 2 (V1)

- ▶ Ou activer SSH depuis l'ordinateur en créant un fichier sur la carte SD. Dans ce cas, on a besoin d'un ordinateur et d'un lecteur de carte SD.
 1. Insérer la carte SD de votre Raspberry Pi dans un ordinateur et accéder à la partition boot, qui est d'ailleurs la seule accessible depuis Windows.
 2. Crée un fichier nommé ssh dans la partition boot de la carte (sans extension, pas de contenu, juste un fichier vide nommé ssh).
 3. Maintenant, il faut retirer la carte de l'ordinateur, remettre la carte dans le Raspberry Pi, allumer celui-ci.

Connection en SSH au Raspberry Pi pour le contrôler depuis un ordinateur

Activer le serveur SSH sur la Raspberry Pi : méthode 2 (V2)

C > boot (D:) >

Name

- overlays
- bcm2708-rpi-b.dtb
- bcm2708-rpi-b-plus.dtb
- bcm2708-rpi-cm.dtb
- bcm2708-rpi-zero.dtb
- bcm2708-rpi-zero-w.dtb
- bcm2709-rpi-2-b.dtb
- bcm2710-rpi-2-b.dtb
- bcm2710-rpi-3-b.dtb
- bcm2710-rpi-3-b-plus.dtb
- bcm2710-rpi-cm3.dtb
- bcm2711-rpi-4-b.dtb
- bootcode.bin
- cmdline.txt
- config.txt

(a)

cmd

Name

- overlays
- bcm2708-rpi-b.dtb
- bcm2708-rpi-b-plus.dtb
- bcm2708-rpi-cm.dtb
- bcm2708-rpi-zero.dtb
- bcm2708-rpi-zero-w.dtb
- bcm2709-rpi-2-b.dtb
- bcm2710-rpi-2-b.dtb
- bcm2710-rpi-3-b.dtb
- bcm2710-rpi-3-b-plus.dtb
- bcm2710-rpi-cm3.dtb
- bcm2711-rpi-4-b.dtb
- bootcode.bin
- cmdline.txt
- config.txt

(b)

D:\>echo>\ssh

(c)

- start_db.elf
- start_x.elf
- start4.elf
- start4cd.elf
- start4db.elf
- start4x.elf
- ssh

(d)

Figure 8: Méthode 2 : à partir du terminal de l'ordinateur

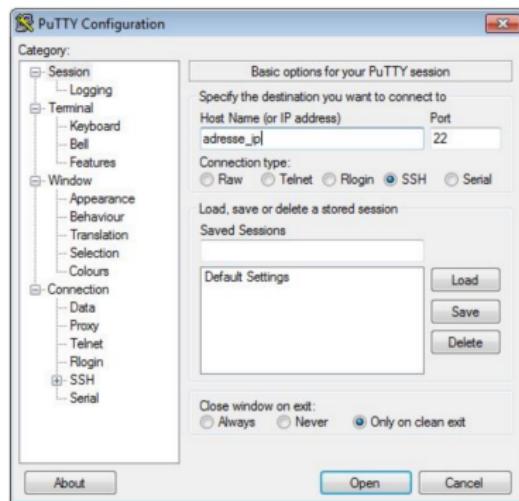
Connection en SSH au Raspberry Pi pour le contrôler depuis un ordinateur

Installer le client SSH sur un ordinateur

Installer le [logiciel Putty](#) qui est un client SSH.

Utiliser SSH pour se connecter à la Raspberry Pi

Lancer Putty et remplir les informations comme sur la figure 9.



Remplacer "adresse_ip" par l'adresse IP ou l'URL du serveur.
Le numéro du port (22) est le port par défaut de SSH.
Cliquer sur le bouton "Open".
Une fenêtre de console s'ouvre en demandant le nom d'utilisateur puis son mot de passe. Une fois ceci fait, la connection est établie avec le Raspberry Pi, il suffit de taper "exit" pour quitter la connexion.

Figure 9: Interface de connection SSH avec Putty

Connection à distance via VNC (Virtual Network Computing)

Connection à distance via VNC (Virtual Network Computing)

Configuration du Raspberry Pi

Dans le terminal, entrer la commande `$ sudo raspi-config`

Dans le [Menu](#), sélectionner [Interfacing Options](#)

Puis [VNC](#), sélectionner [Oui](#) pour permettre la connexion via le serveur VNC.

L'icône de VNC apparaît alors dans la barre de tâche en haut à droite.

Cliquer dessus pour ouvrir le serveur VNC et retrouver l'adresse IP, l'état de la connexion et les informations de sécurité.

Installation du logiciel VNC Viewer

Sur l'ordinateur, télécharger et installer [le logiciel VNC Viewer](#).

Une fois le logiciel installé, ouvrez-le. Il suffit alors de renseigner l'adresse IP du Raspberry Pi puis valider. Entrer le login et le mot de passe pour se connecter au Raspberry Pi.

Connection à distance via VNC (Virtual Network Computing)

Transfert de fichiers

Un outil intéressant dans VNC est le transfert de fichiers qui permet de récupérer ou de charger des fichiers sur le Raspberry Pi sans avoir à faire des allers-retours entre le PC et le Raspberry avec une carte SD ou une clé USB.

Pour transférer un fichier du PC au Raspberry, cliquer sur l'icône VNC dans la fenêtre du PC en haut à gauche. Sélectionner

Transférer des fichiers

Sélectionner ensuite le(s) fichier(s) désiré(s).

Pour transférer des fichiers du Raspberry au PC, ouvrir la fenêtre du serveur VNC en cliquant sur l'icône VNC sur la barre de tâche du Raspberry en haut à droite. Puis cliquer sur l'icône Menu en haut à droite et sélectionner Transfert de fichier

Création d'un NAS

Création d'un NAS

Avec le Raspberry Pi, on va créer un système de stockage multimédia local (NAS : Network Attached Storage, ou Serveur de Stockage en Réseau). En d'autres termes, on va créer un disque dur sur le réseau local avec le Raspberry Pi, sur lequel tous les appareils multimédia (ordinateur, smartphone, télévision...) pourront accéder sans avoir à être branchés dessus.

Pourquoi Raspberry ?

Il existe de nombreuses bonnes raisons de configurer le Raspberry Pi en un serveur NAS : le serveur basé sur le nano-ordinateur est bien moins cher qu'un système NAS prêt à l'emploi (les coûts énergétiques sont considérablement moins élevés en raison des faibles besoins du Raspberry Pi ...).

Supports de stockage :

- ▶ Une clé USB ordinaire qui peut être alimentée directement par le Raspberry Pi.
- ▶ ou un disque dur USB alimenté via le hub USB ou un disque dur avec sa propre alimentation électrique (solution idéale).

Configuration du Raspberry Pi

Mise à jour du Raspberry Pi :

```
sudo apt-get update
sudo apt-get upgrade
```

Créer des dossiers publics et privés qui seront accessibles sur le NAS :

```
sudo mkdir /home/shares
sudo mkdir /home/shares/public
sudo chown -R root:users /home/shares/public
sudo chmod -R ug=rwx,o=rx /home/shares/public
```

Création d'un NAS

Création du serveur NAS avec Samba

On va installer le NAS à proprement parler. Pour cela on va utiliser le logiciel Samba capable de gérer la mise en réseau d'un disque dur pour pouvoir y accéder depuis n'importe quel système d'exploitation ou ordinateur connecté sur le réseau.

Installer Samba sur le Raspberry Pi à l'aide de la commande suivante :

```
sudo apt-get install samba samba-common-bin
```

Édition du fichier de configuration :

```
sudo nano /etc/samba/smb.conf
```

Pour gérer les espaces de stockage privés, il faut se rendre dans la partie "homes". Si on souhaite écrire (envoyer des fichiers) sur le NAS, il faut rechercher les lignes suivantes et les modifier :

```
read only = no (remplacer yes par no)  
security = user (supprimer le #)
```

Création d'un NAS

Création du serveur NAS avec Samba

Enfin, tout en bas du fichier, on va rajouter des paramètres relatifs à l'accès à la partie public du NAS :

```
[public]
comment= Public Storage
path = /home/shares/public
valid users = @users
force group = users
create mask = 0660
directory mask = 0771
read only = no
```

Fermer le fichier en le sauvegardant et redémarrer Samba.

```
sudo /etc/init.d/smbd restart
```

Ajout d'un utilisateur à Samba. Dans cet exemple nous ajouterons l'utilisateur pi.

```
sudo smbpasswd -a pi
New SMB password: raspberry
Retype new smb password: raspberry
```

Création d'un NAS

Ajouter un volume (clé usb, disque) au Raspberry

Connecter une clé USB (et/ou un disque dur) au Raspberry.

```
fdisk -l
```

Noter les noms attribués aux périphériques de stockage branchés.

Exemple : sda1 : nom du disque dur, sdb1 : nom de la clé USB.

```
sudo mkdir /home/shares/public/disk1 (pour le disque dur par exp)
sudo mkdir /home/shares/public/usb1 (pour la clé USB par exp)
```

Relier le disque dur (clé usb) à ce chemin :

```
sudo mount /dev/sda1 /home/sahres/public/disk1
sudo mount /dev/sdb1 /home/shares/public/usb1
```

Permissions :

```
sudo chown -R root:users /home/shares/public/disk1
sudo chown -R root:users /home/shares/public/usb
sudo chmod -R ug=rwx,o=rx /home/shares/public/disk1
sudo chmod -R ug=rwx,o=rx /home/shares/public/usb
```

Création d'un NAS

Connection au serveur NAS

Le NAS est maintenant configuré et il ne reste plus qu'à s'y connecter.

Pour les smartphones :

- ▶ On peut se connecter avec une application :
 - ▶ Android : File Expert, LAN DRIVE ...
 - ▶ IOS : File Explorer, Explorateur de fichier → ... → Connecter serveur ...

Pour les PC Windows :

- ▶ Il faut se rendre dans "Ce PC" et cliquer sur l'onglet "Ordinateur" puis cliquer sur "Connecter un lecteur réseau".
- ▶ Choisir le répertoire public (Si on n'a jamais changé le nom du Raspberry Pi) en renseignant \\raspberrypi\public et au répertoire privé avec le nom d'utilisateur (dans notre exemple pi) en renseignant \\raspberrypi\pi