

Projet 2 - Jouer avec les E/S du Raspberry Pi (rPi) en Bash

Le Raspberry Pi a un connecteur à 40 *pins* étant principalement des GPIOs. Ce projet vous demande de commencer à travailler sur le rPi, puis ensuite à communiquer à travers ce connecteur.

Préparation de l'environnement

Vous allez recevoir un rPi 3, une clé USB pour ce défi. Vous pouvez les garder le temps du projet. La clé USB doit être en format FAT32. Vous pouvez y transférer vos scripts du projet 1.

Défi 1: Script de préparation d'un rPi

Avant de commencer à travailler sérieusement sur un rPi, il faut bien le configurer. Il faut donc avoir un script de prêt. Pour ce défi, créer un fichier nommé *init_rPi.sh* qui doit exécuter les actions suivantes:

- Création d'un nouvel utilisateur et de son mot de passe
- Désactiver le compte *pi*
- Activation du serveur de temps, modification du serveur pour *time.apple.com*. Vous pouvez juste appelé votre script du projet précédent.
- Mise à jour du système. Il faut mettre à jour ce qui est déjà installé.
- BONUS: sélectionner un point d'accès Wifi. Demande le mot de passe et se connecte.

Ce script peut être interactif (interagit avec l'utilisateur, pour demander le nom d'utilisateur et son mot de passe en autre). ## Défi 2: Et la lumière fût

Pour ce défi, vous devez concevoir un script nommée *lumiere.sh* qui permet d'activer ou désactiver un GPIO. Il est de la structure suivante:

```
lumiere.sh on|off xx
```

Description du script

Lorsque le script est appelé, il faut indiquer *on* ou *off* pour activer ou désactiver le GPIO puis *xx* correspond au niveau de GPIO. Vous pouvez trouver ces numéros sur le site officiel. On parle ici du numéro GPIO et non du numéro de pin physique. Votre script doit faire la conversion vers le bon numéro.

Le script doit initialiser une pin si elle ne l'est pas lorsqu'il est appelé.

BONUS: Ayez une gestion d'erreur qui retourne une erreur si un numéro de GPIO inconnu. L'erreur à retourner pourrait ressembler à: "Numéro de GPIO inexistant."