Guide : Création d'une image ISO personnalisée de Raspberry Pi OS

# Introduction

Ce guide vous permettra de créer une image ISO personnalisée de Raspberry Pi OS. L'image inclura un script automatique qui détecte les clés USB, trouve un fichier Python spécifique, et l'exécute une fois la clé USB connectée.

# Étape 1 : Préparer l’image de base de Raspberry Pi OS

1. Téléchargez l'image officielle de Raspberry Pi OS Lite ou Desktop depuis le site officiel :  
 - [Télécharger Raspberry Pi OS](https://www.raspberrypi.org/software/operating-systems/)  
2. Sélectionnez la version Lite si vous n'avez pas besoin d'une interface graphique complète.  
3. Flashez l'image sur une carte SD pour la Raspberry Pi en utilisant un outil comme Balena Etcher.

# Étape 2 : Installer les paquets et outils nécessaires

Une fois l'image installée et que la Raspberry Pi est démarrée :  
1. Connectez-vous au terminal (SSH ou directement sur la Raspberry Pi).  
2. Mettez à jour votre système et installez les paquets nécessaires avec les commandes suivantes :

sudo apt update && sudo apt upgrade -y  
sudo apt install udev python3 python3-pip

# Étape 3 : Créer un script de détection et d'exécution automatique

Créez un script Python qui détecte les périphériques USB et exécute le fichier Python si celui-ci est trouvé :  
1. Créez un fichier avec la commande suivante :  
 sudo nano /usr/local/bin/usb\_script\_runner.py  
2. Collez le script suivant dans ce fichier :

import os  
import subprocess  
  
SCRIPT\_NAME = "hotspot\_sans\_dmz.py"  
  
def find\_script\_in\_usb():  
 mounted\_devices = subprocess.check\_output("ls /media/pi", shell=True).decode().splitlines()  
 for device in mounted\_devices:  
 script\_path = f"/media/pi/{device}/{SCRIPT\_NAME}"  
 if os.path.exists(script\_path):  
 print(f"Script trouvé : {script\_path}")  
 os.system(f"python3 {script\_path}")  
 break  
 else:  
 print(f"Aucun script trouvé dans {device}")  
  
if \_\_name\_\_ == "\_\_main\_\_":  
 find\_script\_in\_usb()

# Étape 4 : Configurer udev pour surveiller les périphériques USB

1. Modifiez les règles de udev pour que le script soit exécuté automatiquement lorsque vous connectez une clé USB :  
 sudo nano /etc/udev/rules.d/99-usb-script.rules  
2. Ajoutez la règle suivante dans le fichier :

ACTION=="add", KERNEL=="sd[a-z][0-9]", RUN+="/usr/local/bin/usb\_script\_runner.py"

# Étape 5 : Créer l'image ISO personnalisée

1. Une fois que tout est configuré et testé, éteignez la Raspberry Pi.  
2. Retirez la carte SD et connectez-la à votre ordinateur via un lecteur de carte.  
3. Utilisez un outil comme Win32 Disk Imager (Windows) ou la commande `dd` sous Linux pour créer une image de la carte SD.  
 Exemple sous Linux :

sudo dd if=/dev/sdX of=raspberry\_pi\_custom\_os.iso bs=4M

Remplacez `/dev/sdX` par le chemin correct de votre carte SD.  
4. Vous pouvez maintenant utiliser cette image ISO pour flasher d'autres cartes SD avec le script intégré.

# Conclusion

Ce guide vous a permis de configurer un Raspberry Pi pour détecter les clés USB et exécuter automatiquement un script Python. Vous avez également appris à créer une image ISO personnalisée prête à être utilisée sur plusieurs systèmes.