# Transformer un Raspberry Pi en enceinte Bluetooth avec PipeWire

Configuration d'un Raspberry Pi pour qu'il fonctionne comme une enceinte Bluetooth, permettant de diffuser de la musique depuis un téléphone, une tablette ou un ordinateur vers des enceintes connectées au Pi.

## Prérequis matériels

- Raspberry Pi (modèle 3 ou ultérieur avec Bluetooth intégré)
- Carte SD avec Raspberry Pi OS Lite déjà installé : voir "Installer un Raspberry Pi sans écran et en Wi-Fi"
- Câble audio jack 3.5mm vers RCA
- Enceintes ou système audio avec entrée analogique RCA.

## Configuration initiale du système

### Configuration de la connexion automatique

Pour permettre une connexion automatique sans interface utilisateur :

```
sudo raspi-config
```

Naviguez vers :

- Option 1 : System Options
- Option S5 : Boot / Auto Login
- Sélectionnez Console Autologin

Raspberry Pi OS se connectera automatiquement à l'utilisateur créé lors de l'installation, ce qui lancera automatiquement PipeWire et WirePlumber.

## Installation des paquets nécessaires

Mettez à jour les dépôts et installez PipeWire, WirePlumber avec les dépendances Bluetooth et Python :

```
sudo apt update
sudo apt install pipewire wireplumber libspa-0.2-bluetooth python3-dbus
```

# Configuration Bluetooth

## Téléchargement des scripts de gestion

Téléchargez le script de gestion Bluetooth et son fichier de service systemd directement depuis le dépôt GitHub :

```
cd ~
wget https://raw.githubusercontent.com/fdanis-oss/pw_wp_bluetooth_rpi_speaker/master/speaker-agent.py
wget https://raw.githubusercontent.com/fdanis-oss/pw_wp_bluetooth_rpi_speaker/master/.config/systemd/user/speaker-agent.service
```

Rendre le script exécutable :

```
chmod +x speaker-agent.py
```

## Configuration du script Bluetooth

Pour autoriser plusieurs connexions simultanées (mixage audio), utilisez la configuration par défaut qui permet à plusieurs appareils de se connecter et de mixer leur audio.

Pour n'autoriser qu'une seule connexion à la fois, modifiez le fichier de service pour ajouter l'option --single-connection :

```
sed -i 's|ExecStart=python speaker-agent.py|ExecStart=python speaker-agent.py --single-connection|' speaker-agent.service
```

## Installation du service systemd

Créez le répertoire utilisateur pour systemd si nécessaire :

```
mkdir -p ~/.config/systemd/user
```

Déplacez le fichier de service au bon emplacement :

```
mv speaker-agent.service ~/.config/systemd/user/
```

Activez et démarrez le service :

```
systemctl --user daemon-reload
systemctl --user enable speaker-agent.service
systemctl --user start speaker-agent.service
```

## Configuration de BlueZ pour appairage automatique

Modifiez la configuration Bluetooth pour permettre l'appairage sans interaction :

```
sudo sed -i 's/#JustWorksRepairing.*/JustWorksRepairing = always/' /etc/bluetooth/main.conf
```

Redémarrez le service Bluetooth :

```
sudo systemctl restart bluetooth
```

# Utilisation

1. Connectez les enceintes : branchez les enceintes ou système Hi-Fi sur la prise jack 3.5mm du Raspberry Pi
2. Redémarrez le système :

```
sudo reboot
```

3. Appairez un appareil :

- Sur le téléphone/tablette/ordinateur, activez le Bluetooth
- Recherchez les nouveaux appareils
- Sélectionnez raspberrypi (ou un nom similaire)
- L'appairage se fera normalement automatiquement sans code PIN requis

4. Diffusez de l'audio : une fois connecté, l'audio devrait être automatiquement redirigé vers les enceintes. Le volume peut être contrôlé depuis l'appareil source

# Dépannage

## Vérifier l'état des services

```
systemctl --user status speaker-agent.service
systemctl --user status pipewire
systemctl --user status wireplumber
```

# Vérifier les connexions Bluetooth

```
bluetoothctl
# Dans l'interface bluetoothctl :
devices
info <adresse_MAC>
```

# Journal des services

Pour voir les logs de l'agent Bluetooth :

```
journalctl --user -u speaker-agent.service -f
```

# Réinitialiser les appareils appairés

Si vous rencontrez des problèmes de connexion :

```
bluetoothctl
# Dans l'interface bluetoothctl :
remove <adresse_MAC>
# Puis réappairez l'appareil
```

# Résoudre l'erreur "appairage non accepté"

L'erreur "appairage non accepté" lors de la tentative de connexion d'un appareil au Raspberry Pi est un problème courant lié aux profils et services Bluetooth. Voici comment le résoudre, étape par étape.

1. Diagnostic initial

Vérification de l'état de l'interface Bluetooth et du service. Ouvrez un terminal sur le Pi et exécutez ces commandes :

Vérifiez que l'adaptateur est listé et qu'il est UP RUNNING.

```
hciconfig -a
```

S'il est marqué DOWN, il faut le réactiver.

Vérifiez que le service Bluetooth est bien actif (active (running)).

```
sudo systemctl status bluetooth
```

Si l'adaptateur est désactivé (DOWN), vous pouvez essayer de le réactiver :

```
sudo rfkill unblock bluetooth
```

2. Configuration du profil A2DP

L'erreur "Pas de services utilisables" ou "appairage non accepté" indique souvent que le profil A2DP (Audio Sink) n'est pas correctement publié.

Le gestionnaire audio PulseAudio gère souvent les connexions Bluetooth. Vérifiez si le module nécessaire est installé :

```
sudo apt install pulseaudio-module-bluetooth
```

Après l'installation, rechargez PulseAudio :

```
pulseaudio -k
pulseaudio --start
```

Parfois, il faut configurer l'agent Bluetooth (bluetoothctl) pour accepter les connexions sur le bon profil. Ouvrez l'outil en ligne de commande :

```
sudo bluetoothctl
```

Dans l'interface bluetoothctl, exécutez les commandes suivantes dans l'ordre :

`agent on` : Active l'agent de gestion des associations.

`default-agent` : Définit l'agent par défaut.

`discoverable on` : Rend le Pi visible.

`pairable on` : Autorise l'appairage.

`scan on` (temporairement) : Pour voir les appareils à proximité.

Ensuite, depuis le téléphone, tentez à nouveau l'appairage. Si une clé est demandée, confirmez-la à la fois sur le téléphone et dans le terminal bluetoothctl.

Une fois l'appairage réussi, assurez-vous de faire confiance (trust) à l'appareil depuis bluetoothctl.

Vérifier les services exposés Après l'appairage, dans bluetoothctl, utilisez la commande `info <adresse MAC du telephone>` pour lister les services du Pi vus par le téléphone.

Vous devriez voir des UUIDs de services audio comme 0000110a-0000-1000-8000-00805f9b34fb (Audio Source) ou 0000110b-0000-1000-8000-00805f9b34fb (Audio Sink). Leur absence confirme le problème de profil.

# Crédits et licence

Ce tutoriel est basé sur le travail de Frédéric Danis :

- Article original : pw_wp_bluetooth_rpi_speaker
- Licence : LGPL-2.1-or-later

Les scripts fournis sont sous licence LGPL-2.1-or-later. Vous êtes libre de les utiliser, modifier et distribuer selon les termes de cette licence.

---

Revision #5
Created 2026-01-08 00:29:17 UTC by Choukajohn
Updated 2026-01-08 17:37:52 UTC by Choukajohn