Pour plus de contenu, connectez-vous !

Pywebdriver

- Pywebdriver - installation Ubuntu
- Pywebdriver - gestion afficheur client
- Nouveau appareil (imprimante, TPE, afficheur client·e) ou erreur matos
- Pywebdriver - installation sur Raspberry Pi

Pywebdriver - installation Ubuntu

> https://github.com/pywebdriver

1. Sur ordinateur Ubuntu 20.04, 22.04 et 24.04 neuf

> Ekimia s'occupe parfois l'installation : voir s'ils n'ont pas mis Pywebdriver en favori dans le navigateur internet.
> Si oui, voir s'ils l'ont bien installé.

Si KO, utiliser successivement les 3 commandes suivantes (le mot de passe du PC sera nécessaire) :

Pour trouver facilement ces commandes, rendez-vous sur ce pad :
https://pad.cloud117.fr/p/pywebdriver

```
sudo apt install python-is-python3
wget https://raw.githubusercontent.com/akretion/pywebdriver/master/debian/install.sh
sudo chmod +x install.sh
sudo ./install.sh
# S'il propose 2 versions, ne pas prendre celle par défaut, mais l'autre
# À "Handle HTTPS(Y/n)" répondre Y (pour oui)
```

> Si le Firefox a un mot de passe principal, il y aura une étape en plus. Le terminal demandera *Enter Password or Pin for "NSS Certificate DB"* et il faut écrire le mot de passe principal

>

> Un redémarrage peut être nécessaire avant l'étape 2 ↓

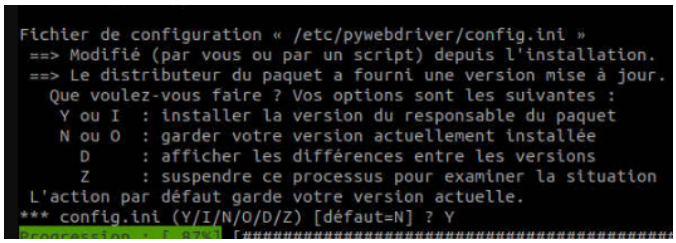
1bis. Sur ordinateur Ubuntu venant d'une version antérieure

- Enlever le dépôt ancien que l'on utilisait pour télécharger pywebdriver
  - Chercher "Logiciels et mises à jour" dans le menu des applications
  - Dans "Autres logiciels", enlever les lignes contenant pywebdriver
  - Actualiser
- **S'assurer qu'on a bien python3.* et désinstaller python2.* (sinon il y aura des erreurs de droits d'accès et rien ne se lancera)**

```
python
# Voir si il y a marqué Python 2.*.* (* étant la version utilisé)
# Si oui, il faut passer en python3 qui doit être sur l'ordi
# Commencer à taper puis →tabulation
pyth →tab
# Il doit y avoir python3

# Sinon mettre à jour : https://www.pythoncentral.io/how-to-update-python/

# Désinstaller python2
sudo apt remove python2.*
# Et installer ce qu'il faut
sudo apt install python-is-python3

#Tester et lancant python
python
# Doit apparaitre une version de python en 3.quelquechose
CTRL+D #pour en sortir
```

- Suivre les commandes suivantes

```
# Désinstaller l'ancienne version de pywebdriver
sudo apt remove pywebdriver
sudo apt purge pywebdriver
sudo rm -r /etc/pywebdriver/

# Installer la nouvelle version
wget https://raw.githubusercontent.com/akretion/pywebdriver/master/debian/install.sh
sudo chmod +x install.sh
sudo ./install.sh
# À "Handle HTTPS(Y/n)" répondre Y (pour oui)

# Il va poser une question sur le fichier de configuration ↓
```



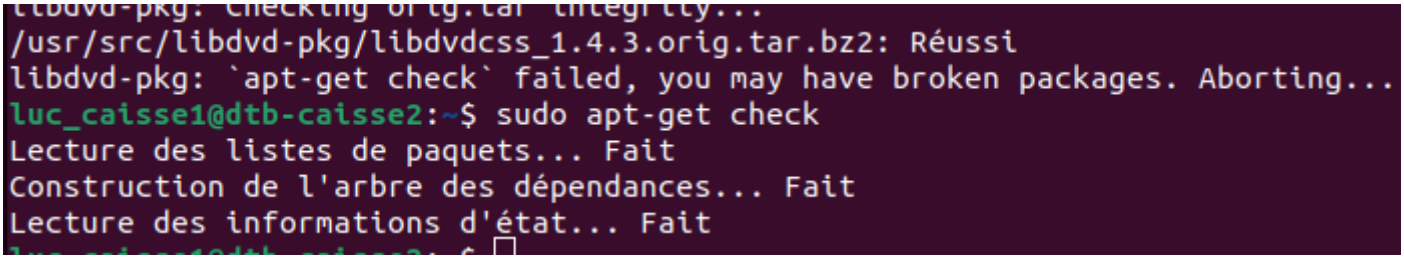
répondre **Y**

1ter. Erreurs à l'installation

Les tests sur l'écran de tests ne fonctionnent pas

- **Avez-vous redémarrer l'ordinateur** ? Sinon faîtes le avant de retester
- Vu à partir du 5 mai sur des Ubuntu 22.04 en python 3.10, le test d'impression bugue → tester sur un Odoo car ça fonctionne quand même peut-être ! [Issue en cours]

Vous avez l'erreur "you may have broken packages"



```
sudo apt-get check
```

Si c'est une erreur de certif et que vous voulez passer en http et pas en https

Pour cela :

- dans le config.ini, mettre host=localhost et commenter ou supprimer les deux lignes de certificats
- relancer pywebdriver
- normalement en http://localhost:8069 ça marche

Vous voulez recommencer l'installation ?

```
sudo apt remove pywebdriver # ça supprime le paquet
sudo apt purge pywebdriver # ça supprime le fichier de configuration
sudo rm -r /etc/pywebdriver/ # supprimer le dossier de données

# et là recommencer l'installation
sudo ./install.sh
```

2. Finalisation et vérification de l'installation

Interface de test

Se connecter à l'adresse https://localhost:8069/

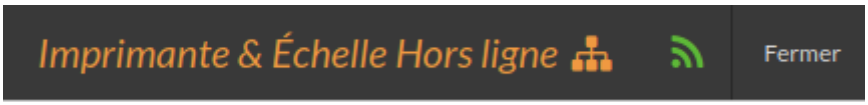
Pour passer l'alerte de sécurité de Firefox, clic sur « Avancé » puis « Accepter le risque et poursuivre »
Cliquer sur "Statut de la Posbox" : on voit ici les appareils bien connectés au point de vente et ceux sans signal.

On peut tester la liaison pour l'imprimante, l'afficheur client et la liaison TPE.

Sur Odoo

Vérifier que l'adresse https://localhost:8069/ est bien renseignée dans Odoo >> Point de vente >> Configuration >> Point de vente >> Matériel Proxy.

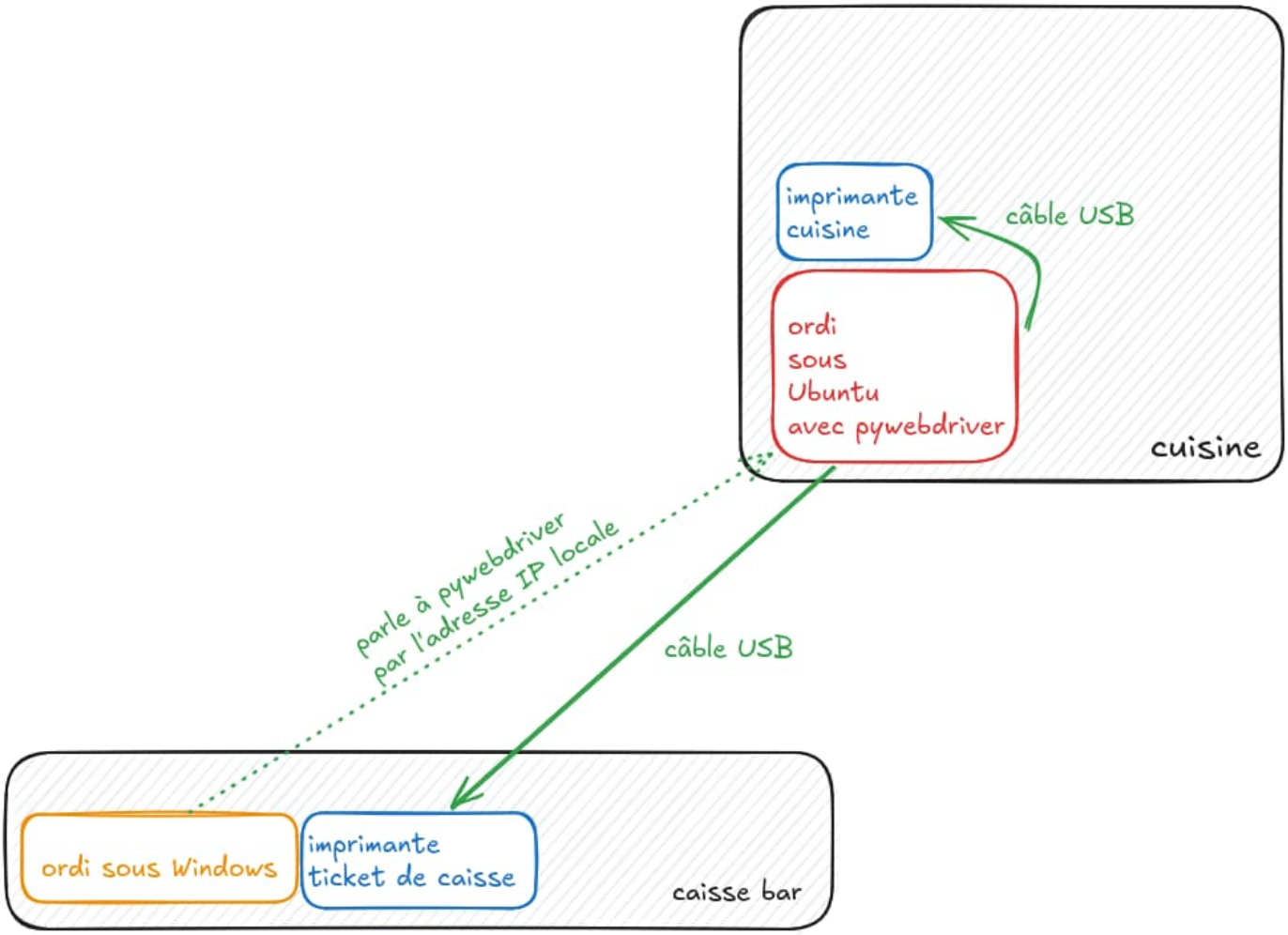
Une fois Pywebdriver installé, et lors de la connexion à Odoo, on voit en haut à droite du point de vente les appareils non reliés :



> Certaines actis ont des paramétrages spécifiques, par exemple plusieurs points de vente rattachés à une seule imprimante ticket de caisse (cf. CHE) => dans ce cas la caisse principale est rattachée à Pywebdriver via https://localhost:8069/ , mais les caisses secondaires ont une autre adresse IP de renseignée dans Configuration >> Point de vente

3. Cas d'un ordi qui communique avec une IP locale

Exemple chez BAJ, voici le fonctionnement au 1er octobre 2024

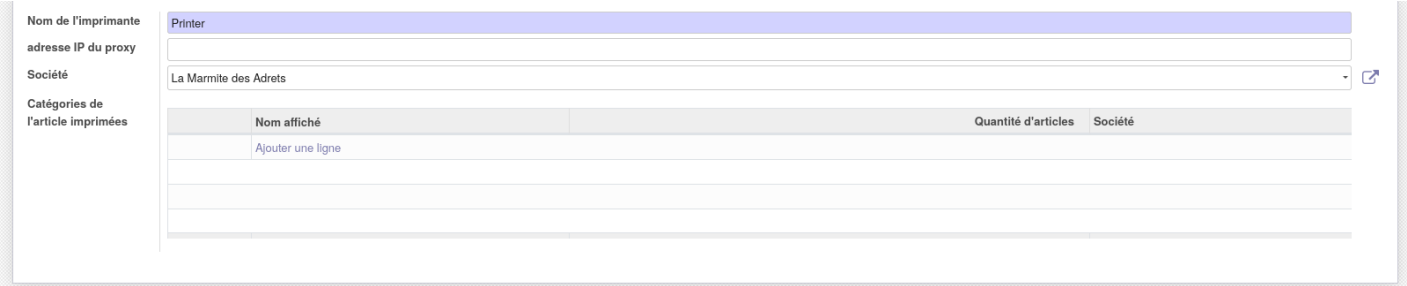


> Pour que cela fonctionne, il faut changer le **config.ini** sur le pc qui est en cuisine.

```
# Accéder au fichier de config
cd /etc/pywebdriver/
# Le modifier avec les droits admin
sudo nano config.ini
# Commenter le host
# Avant : host=127.0.0.1
# Après : ;host=127.0.0.1
# Enregistrer en faisant CTRL + X (puis O pour enregistrer et Entrée)
```

Il faut ensuite ajouter une imprimante dans la configuration du point de vente :





Il faut ensuite récupérer l'adresse ip du PC relié à l'imprimante cuisine. (J'ai utilisé ip route et pris la dernière adresse ip mentionnée et ça a fonctionné). Il faut ajouter ensuite l'host (Normalement :8069). On peut tester le lien entre les deux PC en accédant à l'interface de Pywebdriver depuis le PC qui doit servir aux commandes.

On peut ensuite sélectionner les catégories de produits qui seront liés à l'imprimante en question.

Différents soucis

Nouvel appareil (imprimante, TPE, afficheur client·e)

Voir tuto dédié : https://librairie.grap.coop/books/r%C3%A9serv%C3%A9-au-service-info/page/rajouter-un-nouveau-appareil-imprimante-tpe-afficheur-client

Erreur de droits d'accès ?

> Redémarre après une belle installation clean !

Vérifier qu'il y a bien la règle de droits d'accès :

```
cd /lib/udev/rules.d
cat 99-pywebdriver.rules
SUBSYSTEM=="usb", GROUP="usbusers", MODE="0660"
```

Vérifier que le compte utilisateur est bien dans les bons groupes

```
groups <lecompteutilisateur>
<lecompteutilisateur> adm cdrom sudo dip plugdev lpadmin sambashare
```

Pywebdriver - gestion afficheur client

Branchement par USB

- Normalement message d’erreur “ATE1 E0” qui dit que la communication passe pas avec l’ordi -> Puis “x-” “x-”
- Dans le terminal : `sudo gedit /etc/pywebdriver/config.ini`
- Dans le texte, Au niveau de [display_driver], changer le device_name => device_name=/dev/ttyACM0
- quand il y a aussi un lecteur de code barre (paragraphe du dessous), changer ACM0 en ACM1
- `sudo service pywebdriver restart`
- Dans Pywebdriver, tester en cliquant sur "Afficher le statut"

> Si ça ne marche pas, inverser ACM1 et ACM0 sur l'un et l'autre des items.

Configuration Point de vente

- Cocher Afficheur client
- Configurer message d’accueil et de fin

Nouveau appareil (imprimante, TPE, afficheur client·e) ou erreur matos

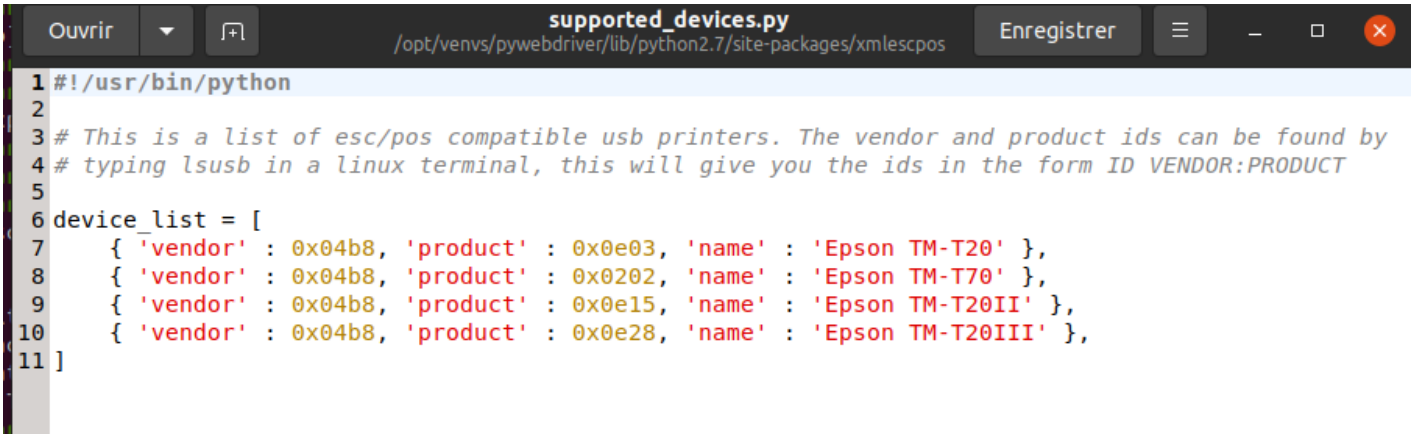
Repérer le matériel connecté à la caisse

> Repérer le matériel connecté à la caisse : `sudo dmesg | grep tty`

Ajouter une imprimante

Ou sinon

```
# Lister les trucs en usb
lsusb

# Brancher et débrancher pour chopper les infos différentes
# On cherche le product ID et le vendor ID

Exemple sur
Bus 003 Device 008: ID 0cf3:0036 Atheros Communications, Inc.

product ID : 0cf3
et vendor ID : 0036

# et on le rajoute dans le fichier qui va bien
sudo gedit /opt/venvs/pywebdriver/lib/python3.8/site-
packages/pywebdriver/plugins/escpos_driver.py
```



Puis ajouter une ligne et redémarrer le service

```
sudo service pywebdriver restart
# pas besoin normalement mais sait-on jamais : sudo service pywebdriver reload
```

Et enfin recharger la page Pywebdriver

> Puis rajouter ça sur notre ordi et proposer une petite PR, on demande à Pierrick de merger

Rajouter un afficheur ou un TPE

Si on veut rajouter des devices pour un TPE ou un afficheur client·e, on rajoute des noms sur ces lignes là :

Pour l'afficheur :
https://github.com/akretion/pyposdisplay/blob/master/pyposdisplay/pyposdisplay.py#L175
Pour le TPE :
https://github.com/akretion/pypostelium/blob/master/pypostelium/pypostelium.py#L52

Pour trouver le nom de l'appareil, lancer un terminal puis

```
/opt/venvs/pywebdriver/python -m serial.tools.list_ports -v

ou
python3.8 -m serial.tools.list_ports -v

ou
```

```
python3.8
>>> import serial.tools.list_ports
>>> serial.tools.list_ports.comports()
```

Si besoin, débrancher, rebrancher pour voir la diff et trouver le truc qui apparaît.

> Puis rajouter ça sur notre ordi et proposer une petite PR, on demande à Pierrick de merger

Puis faudra faire un rebuild de pywebdriver pour rebuilder un .deb

Pywebdriver - installation sur Raspberry Pi

Trouver l'adresse IP sans y avoir accès directement

> **Contexte** : on a envoyé le raspberry Pi à une acti qui l'a connecté à son réseau (par câble ethernet) et qui l'a branché éléctriquement (led vertes).
> **Le but du jeu** : trouver l'adresse IP du raspberry qui fait tourner Pywebdriver

Plusieurs possibilités :

A. [Windows/Linux] Se connecter à l'interface administration de la box pour trouver les appareils connectés au réseau

1. **Orange** : Livebox : https://livebox
2. **Bouygues** : Bbox : https://mabbox.bytel.fr ou 192.168.1.254.
3. **Free** : Freebox : mafreebox.freebox.fr ou 192.168.0.254

B. [Linux] Depuis un poste Linux, on peut "scanner" le réseau pour trouver l'adresse IP

```
# Installer le logiciel nécessaire pour scanner le réseau
sudo apt install nmap
# Scanner le réseau
sudo nmap -sP 192.168.1.0/24
```

Le résultat sera comme ceci :

```
Nmap scan report for pi (192.168.1.90)
Host is up (0.0029s latency).
MAC Address: B8:27:EB:05:72:2C (Raspberry Pi Foundation)
```

> Bien prendre l'adresse IP qui est au dessus de la ligne Raspberry

source : https://raspberrytips.fr/comment-trouver-ip-raspberry-pi/

Paramétrage Odoo

- Se connecter à l'adresse IP trouver : ouvrir son navigateur et aller sur https://<adresseIP>
- Passer les exceptions de sécurité
- Configurer le point de vente sur Odoo avec l'adresse IP https://<adresseIP>

Installation raspberry Pi chez ECU

> Cas spécial du RaspberryPi chez ECU

Il s'agit de lancer pywebdriver en tâche de fond.

Normalement le programme se lance au démarrage du système.

Pour vérifier cela, lancer le navigateur internet et http://localhost:8069 puis Statut de la PosBox

> Par défaut, le lancement à distance avec vision de l'écran nécessite l'accord de la personne.
> Or sur un Raspberry, pas d'écran, pas de souris par défaut donc il ne peut pas dire oui !
> Donc soit changer les droits de prise à distance (en allant dans le compte admin), soit demander à la personne de connecter un écran (avec miniprise HDMI) et une souris

Paramétrage initial pour lancer au démarrage

- Modifier le fichier rc.local :

```
sudo nano /etc/rc.local


# Rajouter ces lignes avant exit 0


cd /home/pi/Documents/pywebdriver
. /home/pi/Documents/pywebdriver/env/bin/activate
/home/pi/Documents/pywebdriver/pywebdriverd &
```

- Puis redémarrer : `reboot`

Si lancement au démarrage marche pas

Prendre la main à distance par terminal

```
cd /home/pi/Documents/pywebdriver
. /home/pi/Documents/pywebdriver/env/bin/activate
nohup /home/pi/Documents/pywebdriver/pywebdriverd &
```

Accéder à pywebdriver

- Chopper l'IP à distance ou directement sur raspberry avec `ifconfig`
- Puis go checkez sur un navigateur si on accède bien à l'interface pywebdriver : http://192.168.1.71:8069
- en remplaçant 192.168.1.71 par l'adresse IP trouvé avec `ifconfig`