

Système écran hall

Configuration du Raspberry

Systeme

Dans le menu > Préférence > Configuration de Raspberry Pi:

- Systeme
 - Changer le hostname
- Interface
 - Activer SSH
 - Activer VNC

VNC

Modifier dans les options:

- Sécurité
 - Authentification: Mot de passe VNC
- Utilisateurs et autorisations
 - Utilisateur standard (user) > changer le mot de passe

Désactiver la veille automatique

RPI 4

Éditez /etc/xdg/lxsession/LXDE/autostart :

```
sudo nano /etc/xdg/lxsession/LXDE/autostart
```

Supprimez ou commentez avec un # la ligne suivante :

```
@xscreensaver -no-splash
```

Ajoutez cette ligne en fin de fichier :

```
@xset s noblank @xset s off @xset -dpms
```

Editez le fichier /etc/lightdm/lightdm.conf :

```
sudo nano /etc/lightdm/lightdm.conf
```

Ajoutez la ligne suivante dans la section [Seat:*] :

```
xserver-command=X -s 0 -dpms
```

Après un redémarrage, le Raspberry Pi ne passera plus en veille.

Lancer Firefox en mode kiosk

Installer Firefox

```
sudo apt install firefox-esr
```

Créer un fichier sur le bureau "start_firefox.sh"

[start_firefox.sh](#)

```
#!/bin/bash

firefox -kiosk https://gip-
cei.ymag.cloud/index.php/planning/afficheur/3/
```

Rendre le script exécutable

```
sudo chmod +x /home/esli/Desktop/start_firefox.sh
```

Masquer la souris

```
sudo apt install unclutter
```

```
sudo nano /etc/xdg/lxsession/LXDE/autostart
```

```
@unclutter
```

Autostart

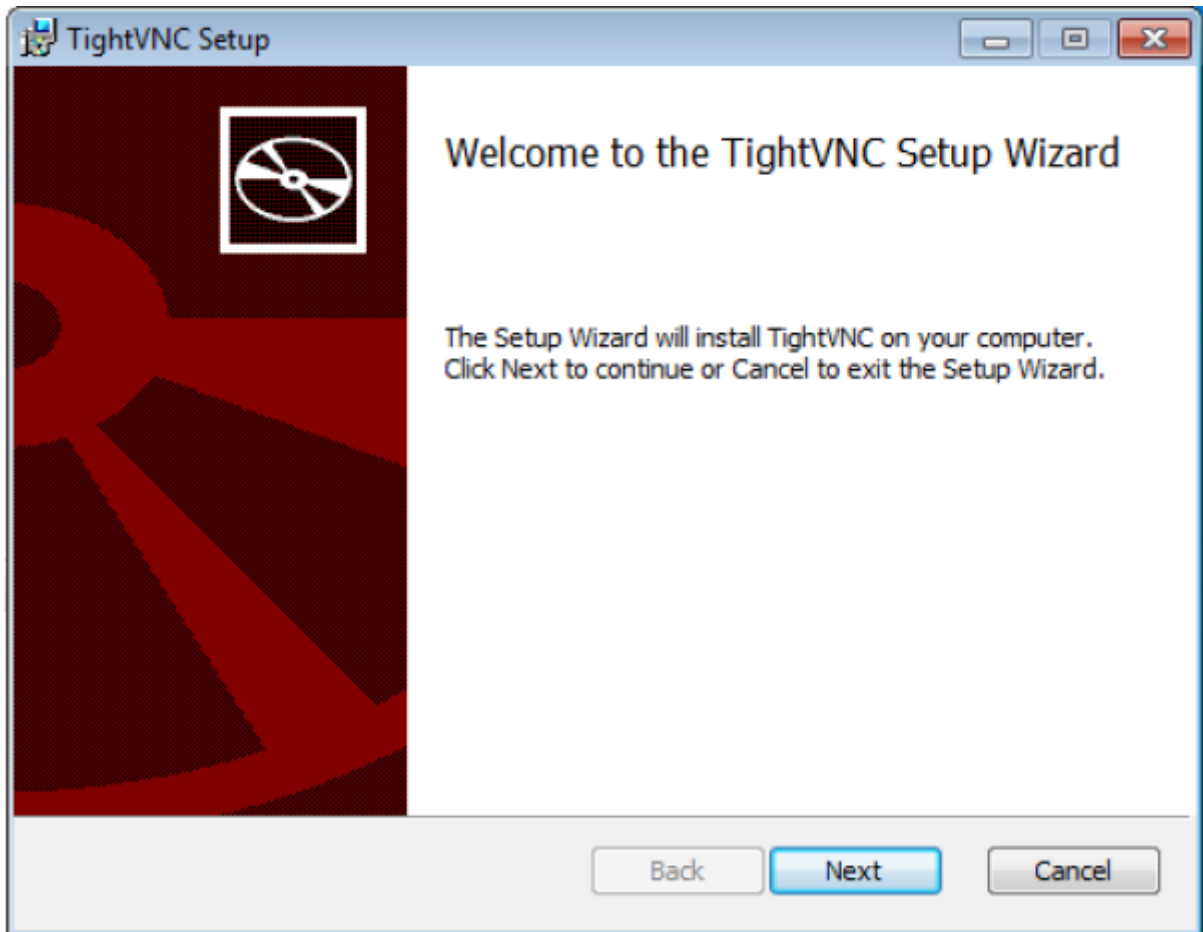
```
sudo nano /etc/xdg/lxsession/LXDE-pi/autostart
```

```
@/home/esli/Desktop/start_firefox.sh
```

Accès à distance (pour Windows)

Installation de Tightvnc

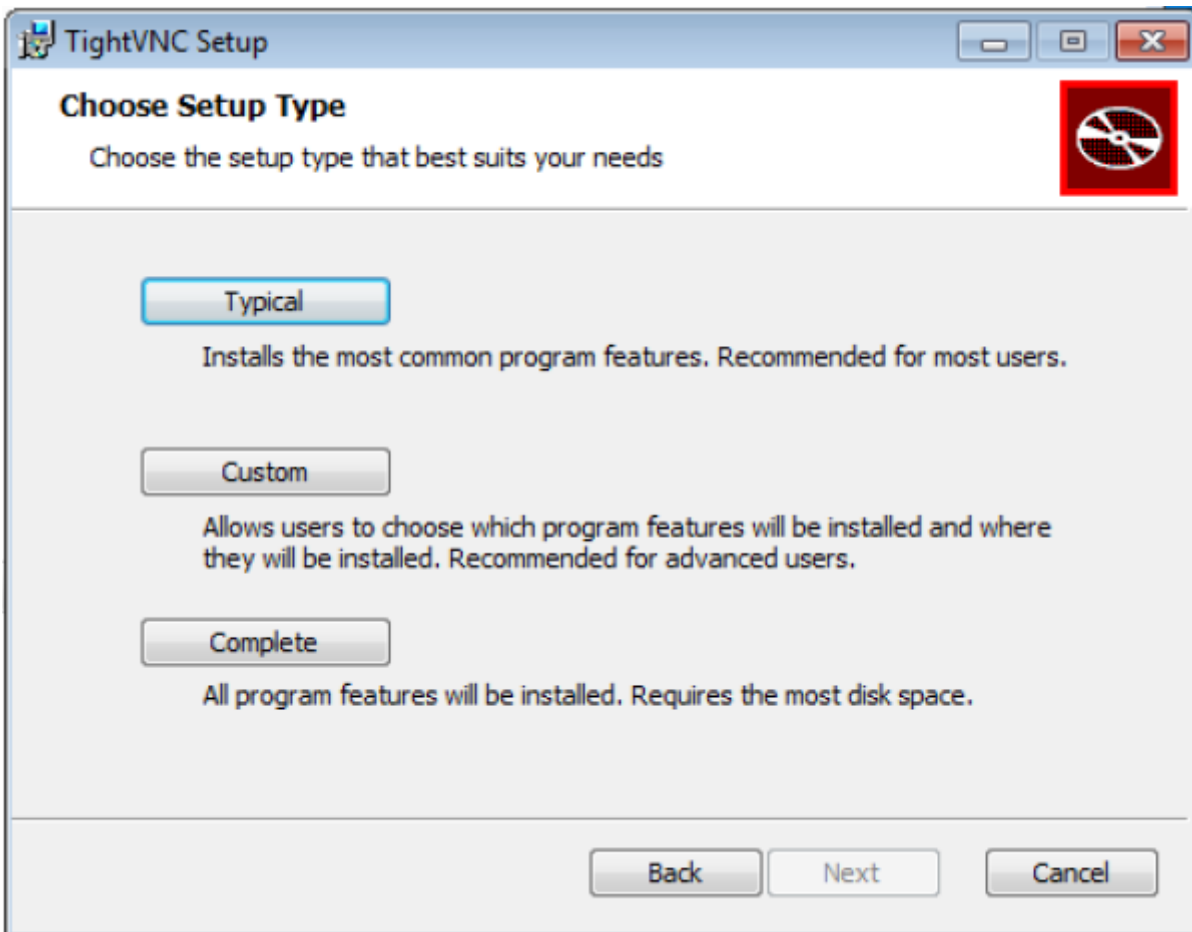
1. Téléchargez et exécutez le logiciel d'installation de Tightvnc:
<https://www.tightvnc.com/download.php> prendre "Installer for Windows (64-bit)"



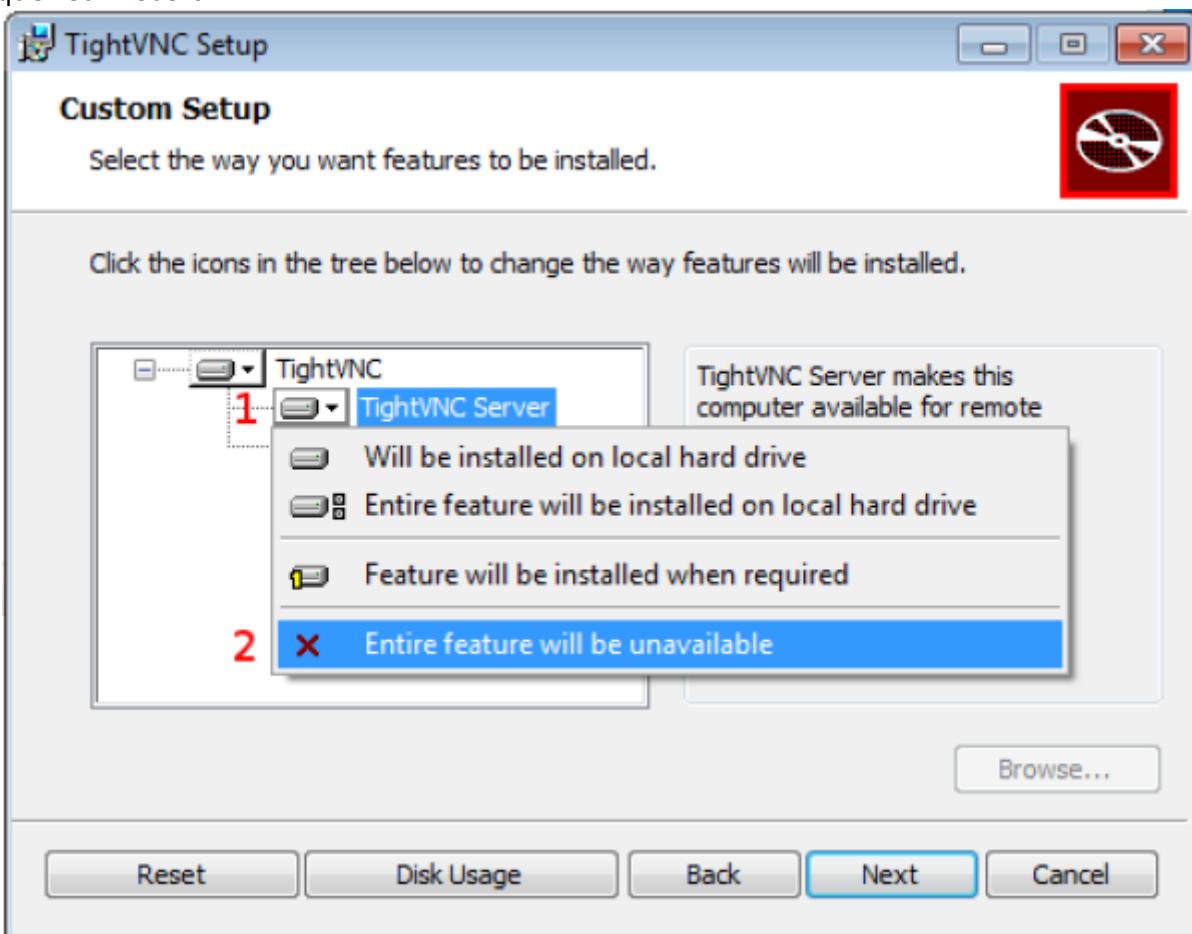
2. Cliquez sur Next



3. Cochez "I accept the terms..." et cliquez Next

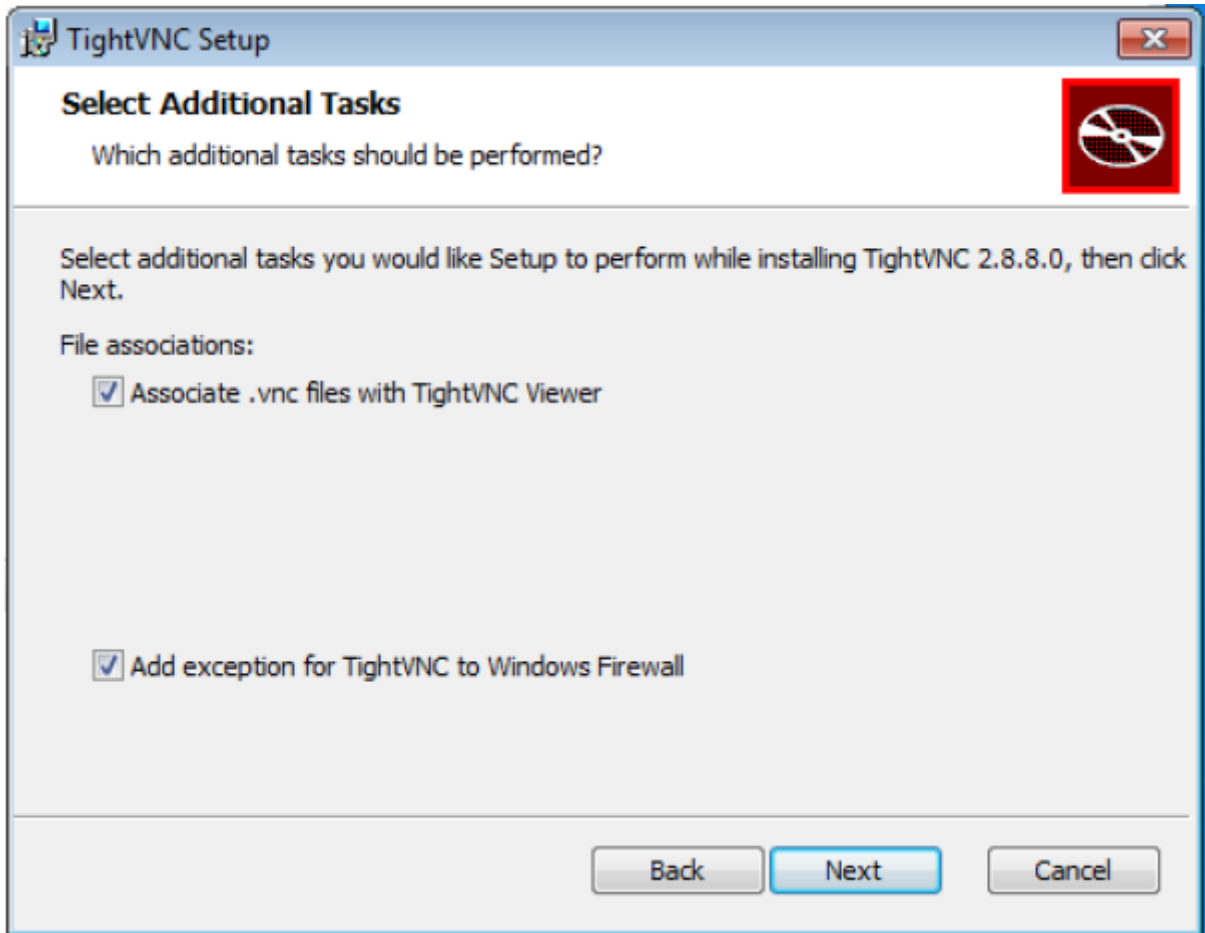


4. Cliquez sur "Custom"

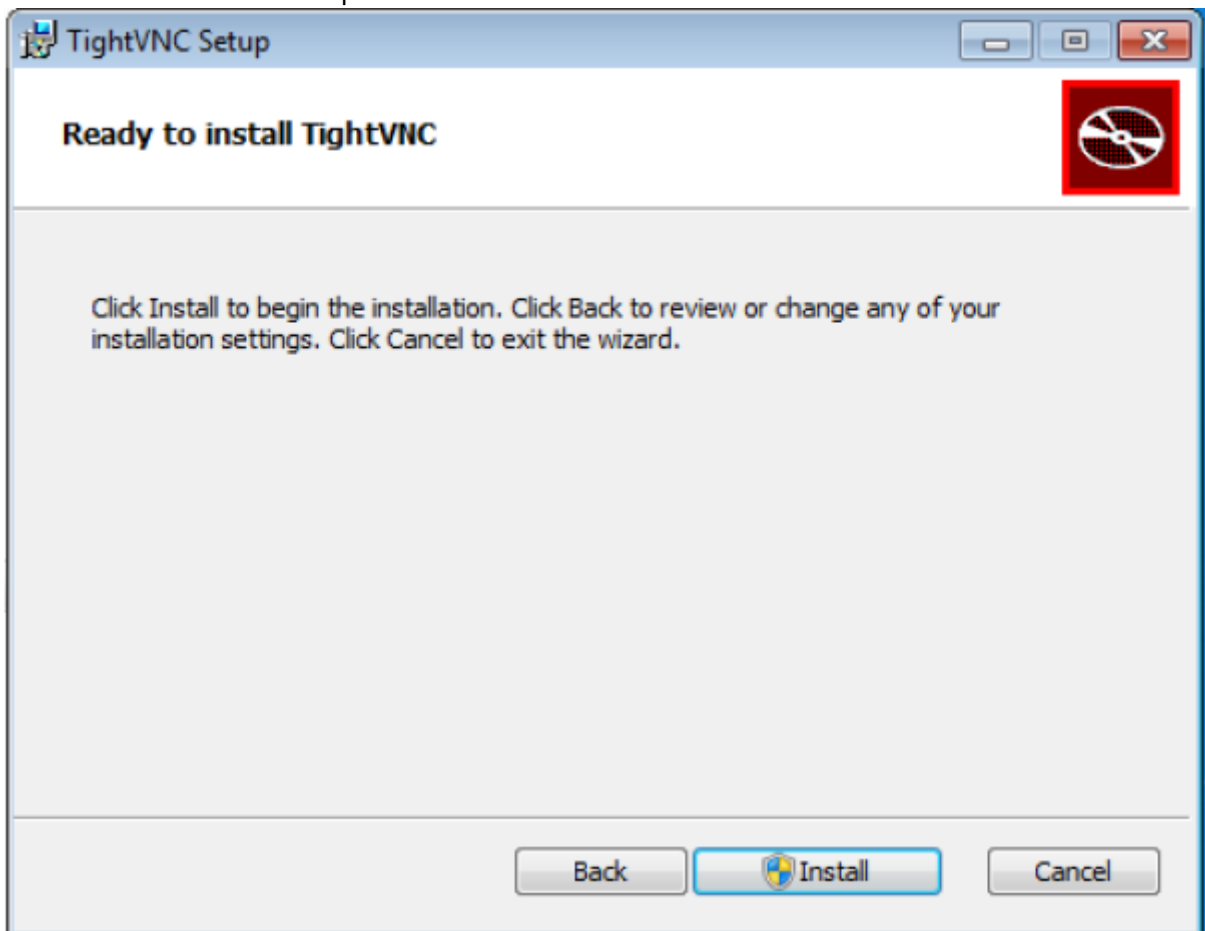


5. Cliquez à côté de "Tightvnc Server" sur la petite flèche (1) et cliquez "Entire feature..." (2)

6. Puis cliquez Next



7. Laissez la case cocher et cliquer Next



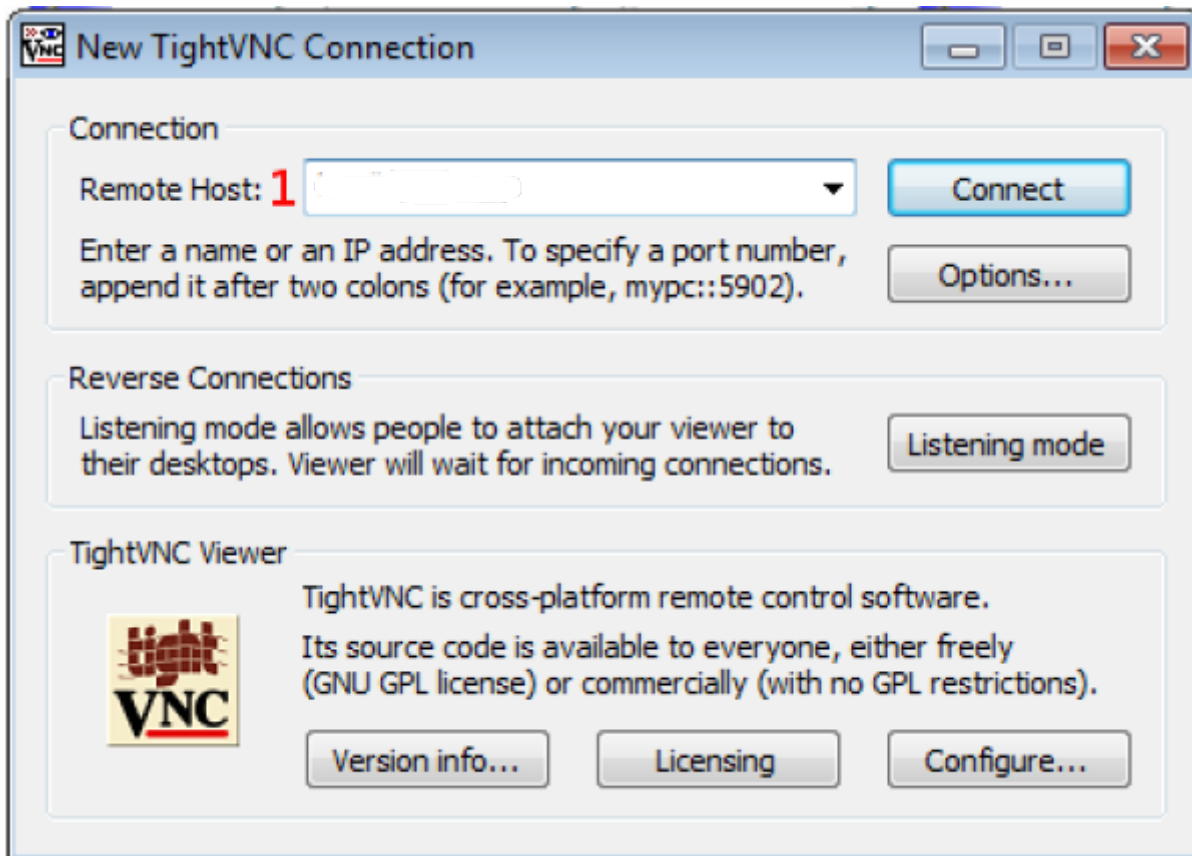
8. Cliquez Install



9. Cliquez Finish

Utilisation de Tightvnc

1. Lancez Tight Viewer



- 2. Inscrivez dans "Remote Host" (1): RPITV-ESLI.local:5900 Puis cliquez sur "Connect"
- 3. MDP: le mot de passe adminlocal des PC



- 4. Vous pouvez cliquer sur la loupe auto (1) et agrandir la fenêtre
- 5. La mise en plein écran est possible (2) ATTENTION pour sortir faire Ctrl+Alt+Maj+F (ne pas oublier)

Fermer Firefox

Firefox en démarré en mode Kiosk, toutes les actions souris ou clavier sont bloqué. Pour sortir faire ATL + F4

Relancer Firefox

Double cliquer sur le fichier start_firefox.sh puis cliquer lancer

Imeon

```
<file python imeon.py> #!/usr/bin/python
```

```
import requests import requests.exceptions import json import time
```

```
LOGIN_URL = "http://192.168.1.15/login" DATA_URL1 = "http://192.168.1.15/battery-status"
DATA_URL2 = "http://192.168.1.15/imeon-status" DATA_URL3 = "http://192.168.1.15/data-lithium"
```

```
EMAIL = "installer@local" # default lmeon login
PASSWORD = "Installer_P4SS" # default lmeon pass
debug = False # True will display the output on your tty
payload = {}
```

```
def poll_lmeon_data():
```

```
    global payload
    session = requests.Session()
```

```
    try:
        resp = session.post(LOGIN_URL, data={"email" : EMAIL , "passwd":
        PASSWORD}, timeout=2)
        data1 = session.get(DATA_URL1, timeout=2)
        data2 = session.get(DATA_URL2, timeout=2)
        data3 = session.get(DATA_URL3, timeout=2)
```

```
        resp.raise_for_status()
        data1.raise_for_status()
        data2.raise_for_status()
        data3.raise_for_status()
```

```
        values1 = data1.json()
        values2 = data2.json()
        values3 = data3.json()
        payload["Timestamp"] = str(time.ctime())
        payload["Battery_status"] = str(values1['bat_status'])
        #payload["Battery_state"] =
        str(values2['state_battery']['message'])
        payload["Battery_activity"] = str(values2['state_battery']['class'])
        payload["Battery_current"] = str(values3['battery-current'])
        payload["Battery_tension"] = str(values3['battery-tension'])
        payload["Battery_temperature"] = str(values3['tmp-bat'])
        payload["Battery_charge"] = str(values3['bms-soc'])
        payload["Battery_error"] = str(values3['error'])
        payload["Battery_warning"] = str(values3['warning'])
```

```
        A = str(values2['state_grid']['message'])
        B = str(values2['state_meter']['message'])
        C = str(values2['state_pv']['message'])
```

```
        payload["Grid_state"] = A.rstrip(A[-1])
        payload["Meter_state"] = B.rstrip(A[-1])
        payload["PV_state"] = C.rstrip(A[-1])
        payload["Meter_status"] = str(values2['state_meter']['class'])
        payload["Inverter_state"] =
        str(values2['state_inverter']['message'])
        payload["State_timeline"] =
        str(values2['state_timeline']['detail'])
        payload["Error_history"] = str(values1['error_history'][0])
```

```
    if debug:
        for x in payload:
```

```
print(x, " : ", payload[x])
```

```
except Exception as err:  
    print("Imeon is eating glue again... (" + str(err)+")")  
    #time.sleep(10)  
    #continue  
    raise SystemExit  
#break
```

```
def run():
```

```
    global payload
```

```
    poll_imeon_data()
```

```
if name == 'main':
```

```
    run()
```

From:
<https://wiki.depantek.net/> - **Dépan'Tek**

Permanent link:
<https://wiki.depantek.net/man:campus:raspberry:start?rev=1724155258>

Last update: **2024/08/20 12:00**

