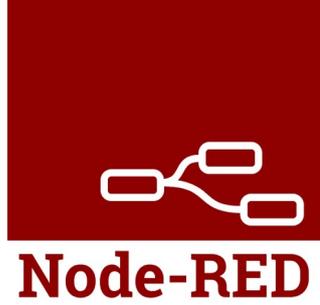
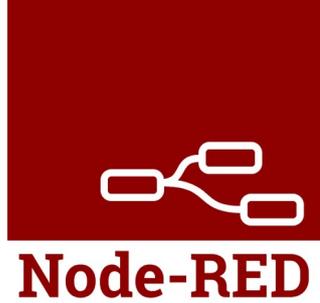


Node-Red



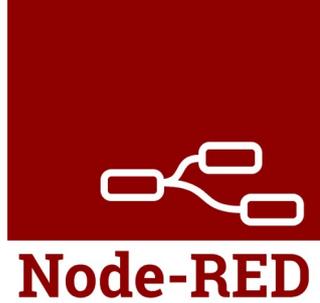
Présentation.

Node-Red



- Node-Red est un langage de développement graphique créé par IBM, au début de l'année 2013.
- Son objectif est de relier des équipements matériels, des services en ligne pour faciliter l'intégration des IoT (Internet of Things).
- Il fait désormais partie de la fondation Java Script depuis 2016
- L'importation et l'exportation de programmes Node-Red se fait au format JSON.
- Il est diffusé en open-source, depuis 2013, et est sous Licence Apache 2.0 (https://fr.wikipedia.org/wiki/Licence_Apache).
- Multi plateformes

Node-Red



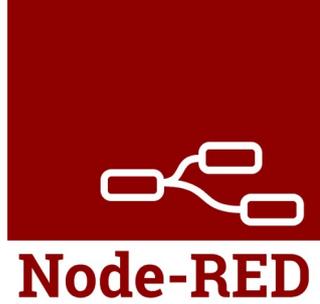
FONCTIONNEMENT

- Node-RED est basé sur le concept de programmation par flux de données. Au lieu de saisir des commandes de programmation dans un éditeur de texte, vous travaillez avec un éditeur de flux visuel comme environnement graphique. Vous y créez un réseau composé de petits blocs de code préétablis, les "nœuds" (Nodes).

DOCUMENTATION

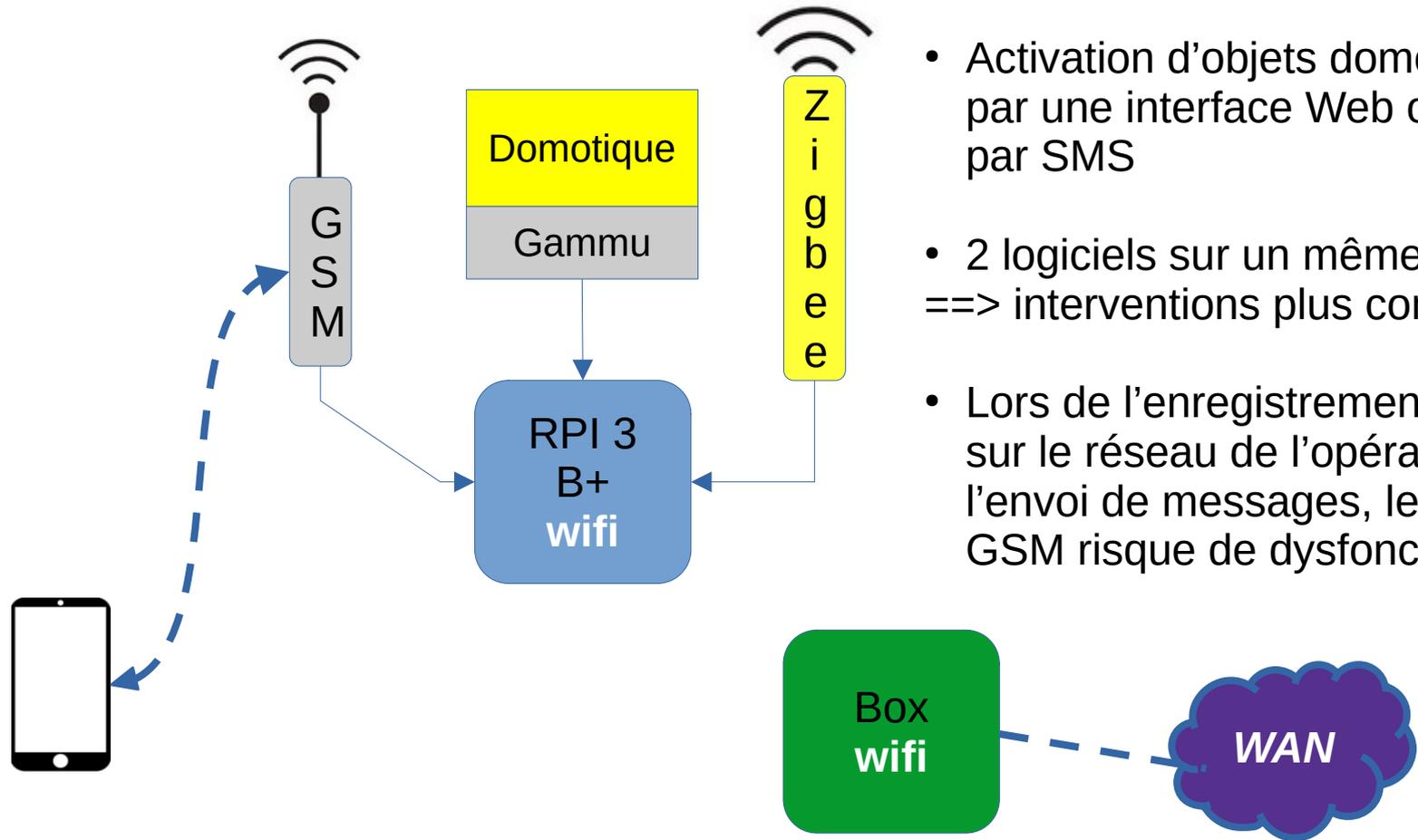
- La documentation est facilement accessible et claire.

Node-Red



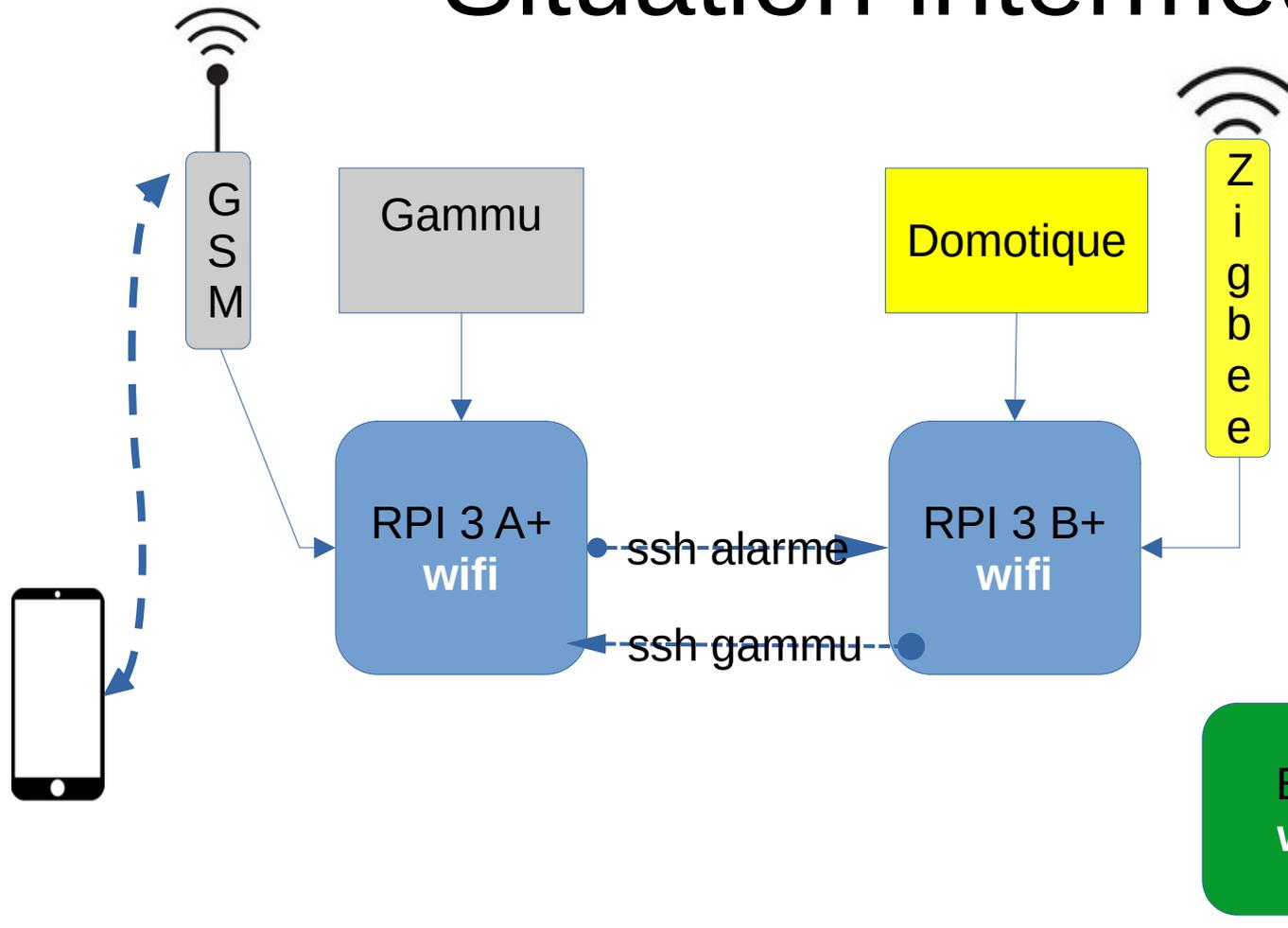
Simplification d'un projet domotique

Situation initiale



- Activation d'objets domotiques par une interface Web ou par SMS
- 2 logiciels sur un même RPI ==> interventions plus complexes
- Lors de l'enregistrement sur le réseau de l'opérateur ou de l'envoi de messages, le module GSM risque de dysfonctionner !!

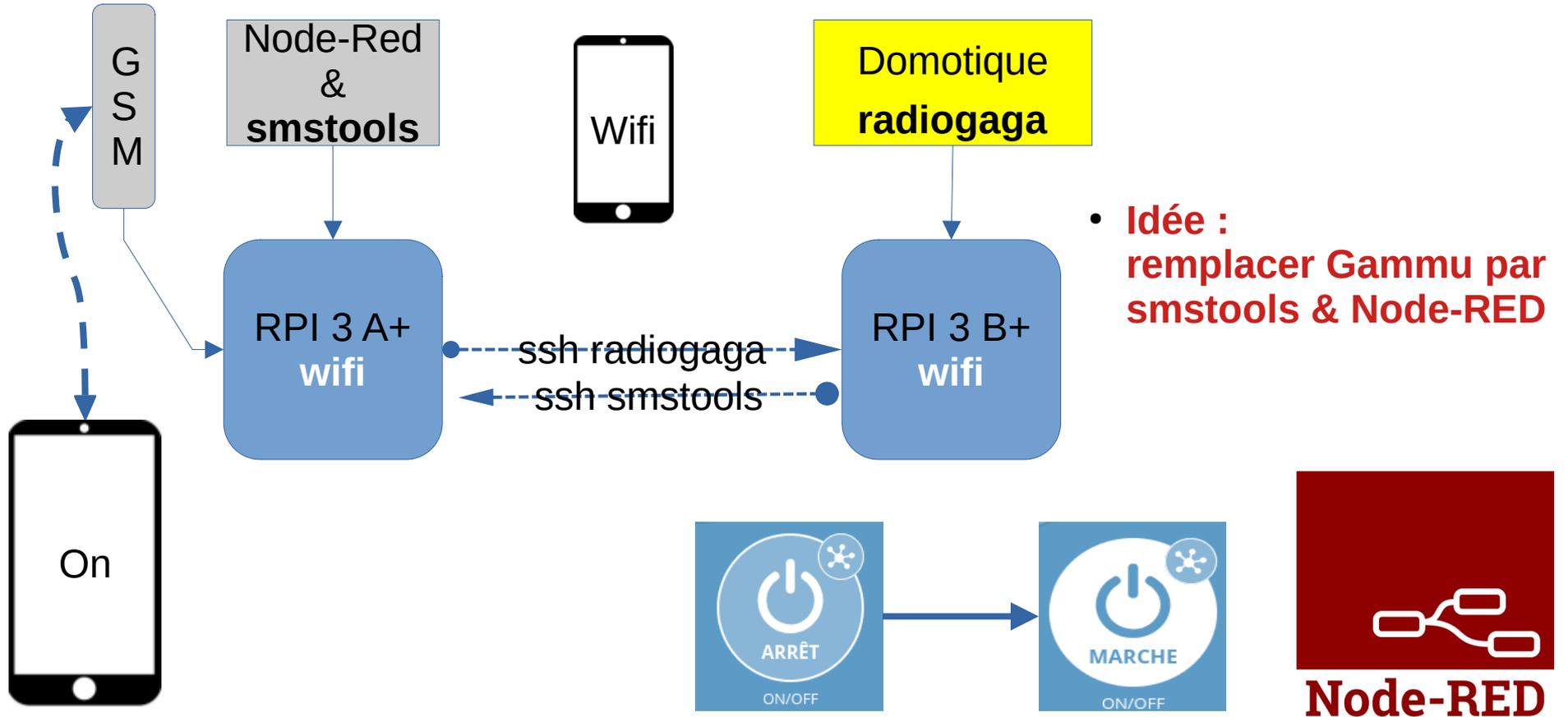
Situation intermédiaire



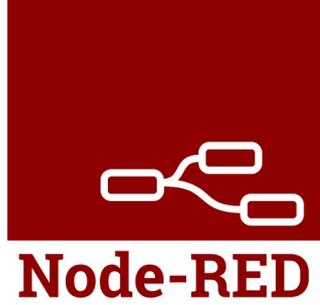
Idée : installer l'application Gammu et l'antenne GSM sur 1 RPI , puis l'application Domotique et son antenne Zigbee sur un autre RPI. Ainsi chaque antenne dispose de plus de puissance. Puis interconnecter les deux RPI en ssh.

- **Problème : droits « root » sur fichiers Gammu lors d'envoi SMS**

Gammu > smstools & Node-RED



Node-Red



Résultats des tests!!

Node-RED



Node-RED

Rechercher le noeud

ex 01 | ex 02 | smstools_ssh

Entrée

- ping
- sms - in

Sortie

Séquence

Analyseur

Stockage

Web Things IO

- Set property
- Get property
- Execute action
- Inject

1 On

2 Check SMS entrants

3 PULSE

Connecté

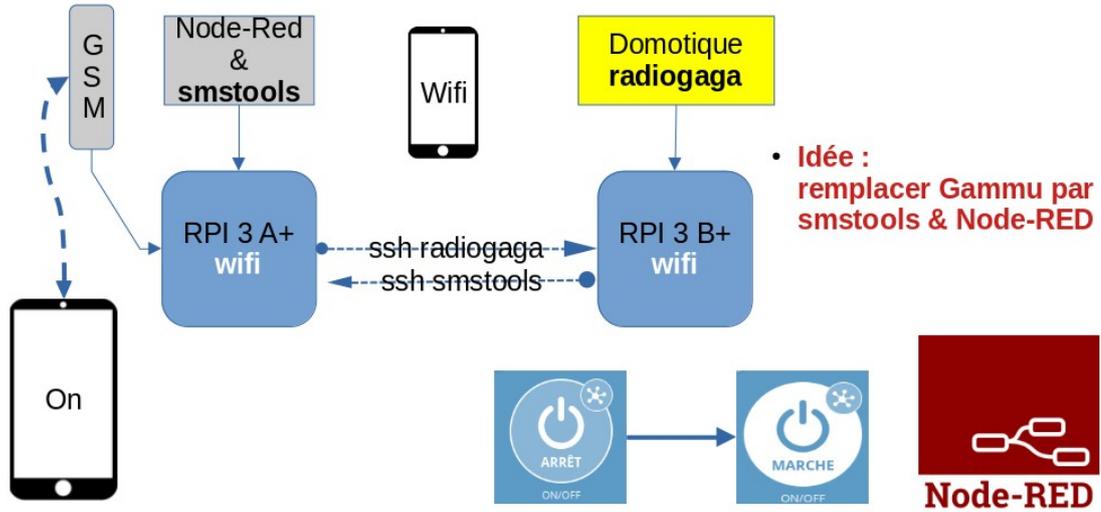
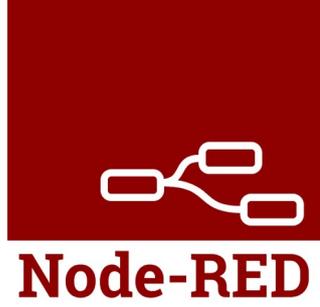
4 MARCHÉ

PULSE

OBJET

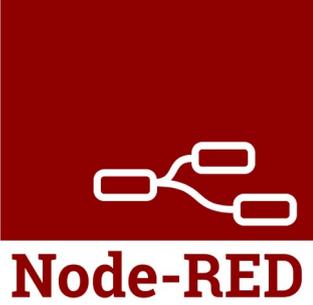
Pulse_On

Node-Red



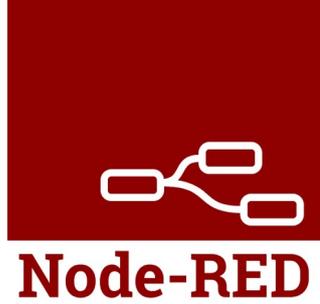
Le détail de l'installation des logiciels sur les RPI est reporté à la fin de l'exposé.

Node-Red



Démarrage.

Node-Red



Au départ d'un PC se connecter en ssh à smstools, puis démarrer node-red !!

```
pi@smstools:~ $ node-red-start
```

Start Node-RED

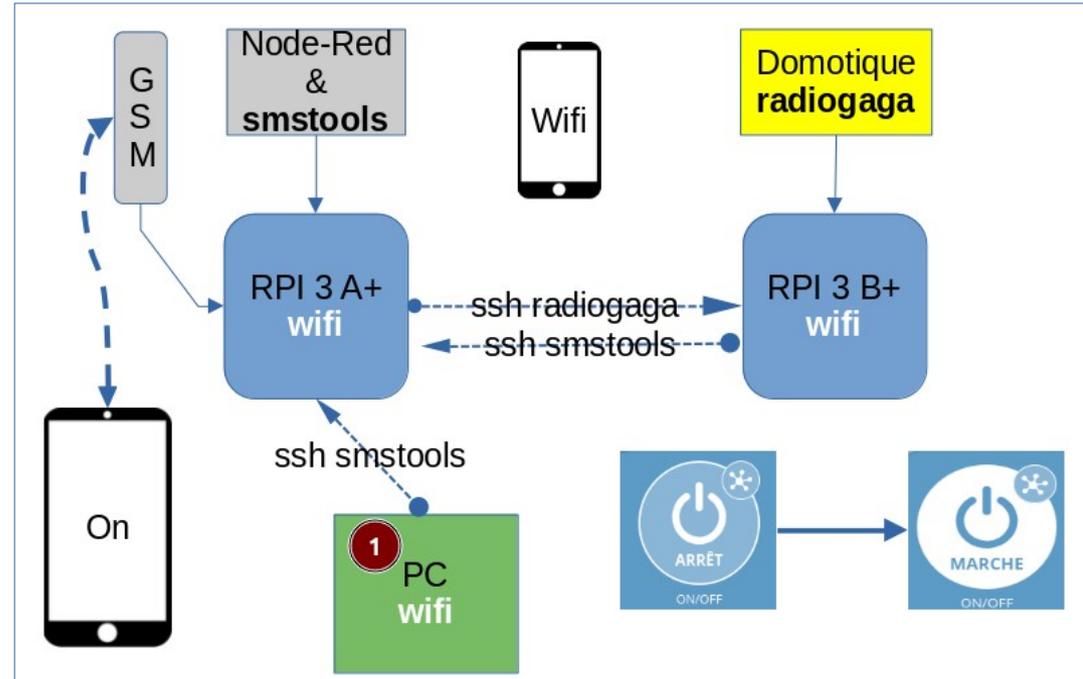
Once Node-RED has started, point a browser at <http://192.168.89.149:1880>

On Pi Node-RED works better with the Firefox or Chrome browser

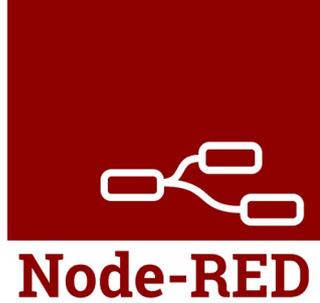
- Use **node-red-stop** to stop Node-RED
- Use **node-red-start** to start Node-RED again
- Use **node-red-log** to view the recent log output
- Use **sudo systemctl enable nodered.service**
« to autostart Node-RED at every boot »

Use `sudo systemctl disable nodered.service` to disable autostart on boot

To find more nodes and example flows - go to <http://flows.nodered.org>



Node-Red



Starting as a systemd service.

25 Nov 18:40:50 - [info]

Welcome to Node-RED

=====

25 Nov 18:40:50 - [info] Node-RED version: v4.0.5

25 Nov 18:40:50 - [info] Node.js version: v20.18.0

25 Nov 18:40:50 - [info] Linux 6.6.51+rpt-rpi-v8 arm64 LE

25 Nov 18:40:51 - [info] Loading palette nodes

25 Nov 18:40:55 - [info] **Settings file : /home/pi/.node-red/settings.js**

25 Nov 18:40:55 - [info] Context store : 'default' [module=memory]

25 Nov 18:40:55 - [info] **User directory : /home/pi/.node-red**

25 Nov 18:40:55 - [warn] Projects disabled : editorTheme.projects.enabled=false

25 Nov 18:40:55 - [info] **Flows file : /home/pi/.node-red/flows.json**

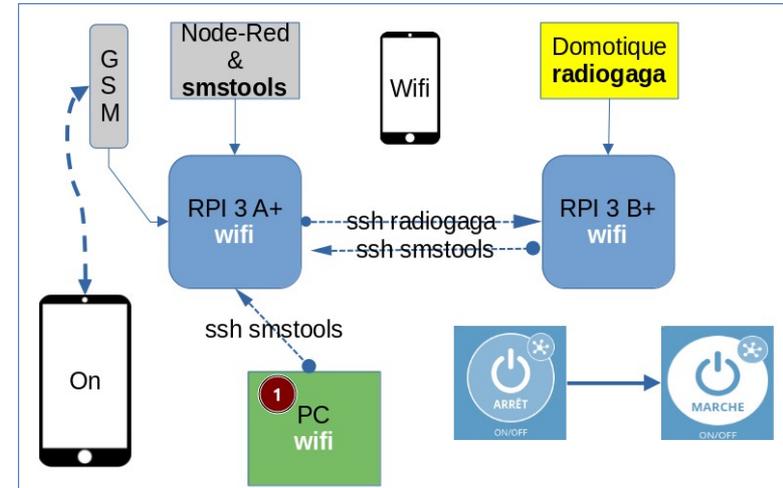
25 Nov 18:40:55 - [info] **Server now running at http://127.0.0.1:1880/**

25 Nov 18:40:55 - [warn] Using unencrypted credentials

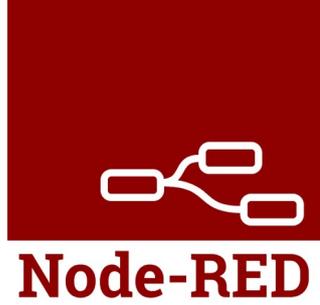
25 Nov 18:40:55 - [info] Starting flows

25 Nov 18:40:55 - [info] [python-function:02e32e3e1497e915] Python function " running on PID 893

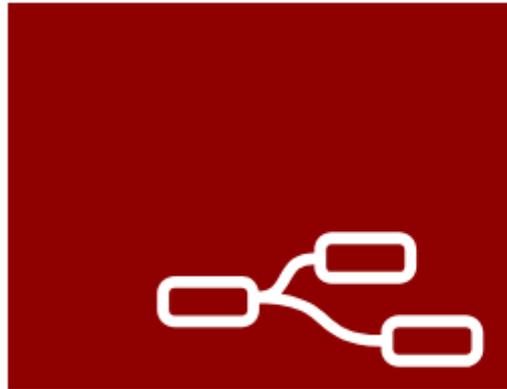
25 Nov 18:40:55 - [info] Started flows



Node-Red



Pour accéder à l'interface « web » de Node-RED, au départ du navigateur de votre PC taper `http://192.168.34.149:1880`



Node-RED

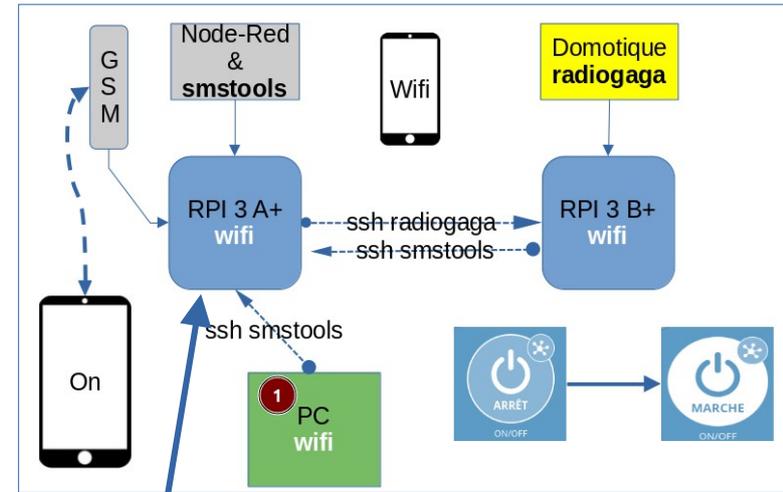
Nom d'utilisateur:

labojurbise

Mot de passe:

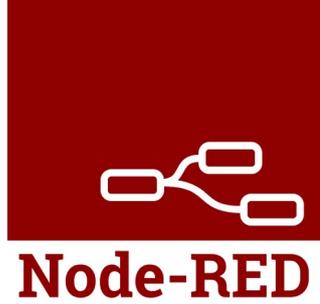
.....

Se connecter



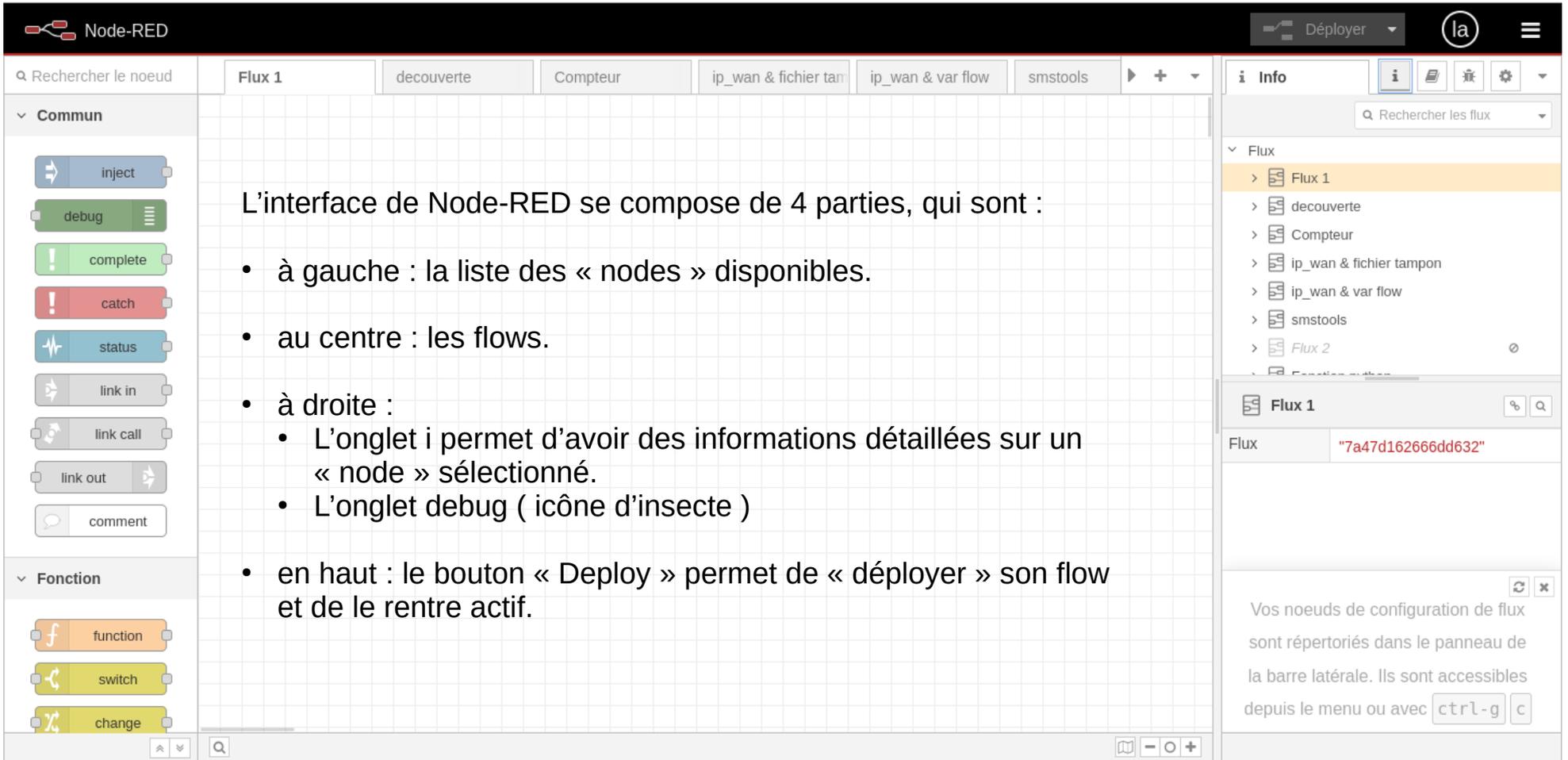
192.168.34.149

Node-Red



Présentation de l'interface « web ».

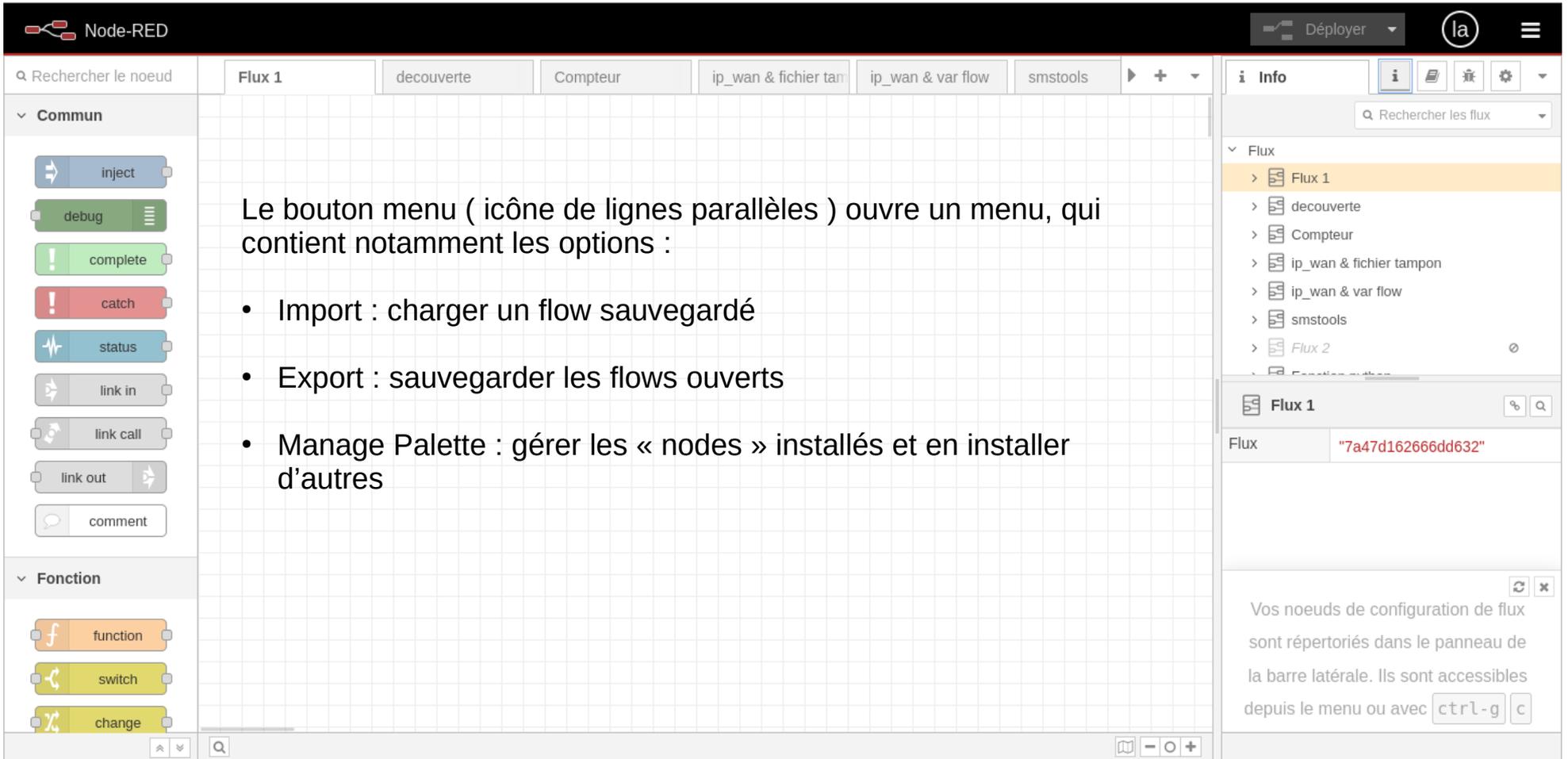
Node-Red



L'interface de Node-RED se compose de 4 parties, qui sont :

- à gauche : la liste des « nodes » disponibles.
- au centre : les flows.
- à droite :
 - L'onglet i permet d'avoir des informations détaillées sur un « node » sélectionné.
 - L'onglet debug (icône d'insecte)
- en haut : le bouton « Deploy » permet de « déployer » son flow et de le rendre actif.

Node-Red

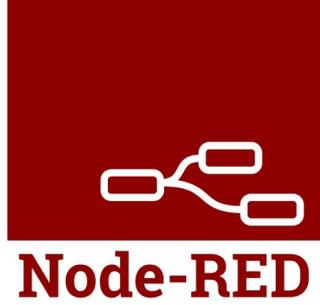


The screenshot shows the Node-RED web interface. The top bar includes the Node-RED logo, a search bar for nodes, and a 'Déployer' button. The main workspace is a grid with a central text area. On the left, there are two palettes: 'Commun' (Inject, Debug, Complete, Catch, Status, Link In, Link Call, Link Out, Comment) and 'Fonction' (Function, Switch, Change). On the right, the 'Info' panel is open, displaying a search bar and a list of flows under the 'Flux' category. The selected flow is 'Flux 1' with ID '7a47d162666dd632'. Below the list, there is a section for 'Flux 1' with a search bar and a description: 'Vos noeuds de configuration de flux sont répertoriés dans le panneau de la barre latérale. Ils sont accessibles depuis le menu ou avec `ctrl-g` `c`'.

Le bouton menu (icône de lignes parallèles) ouvre un menu, qui contient notamment les options :

- Import : charger un flow sauvegardé
- Export : sauvegarder les flows ouverts
- Manage Palette : gérer les « nodes » installés et en installer d'autres

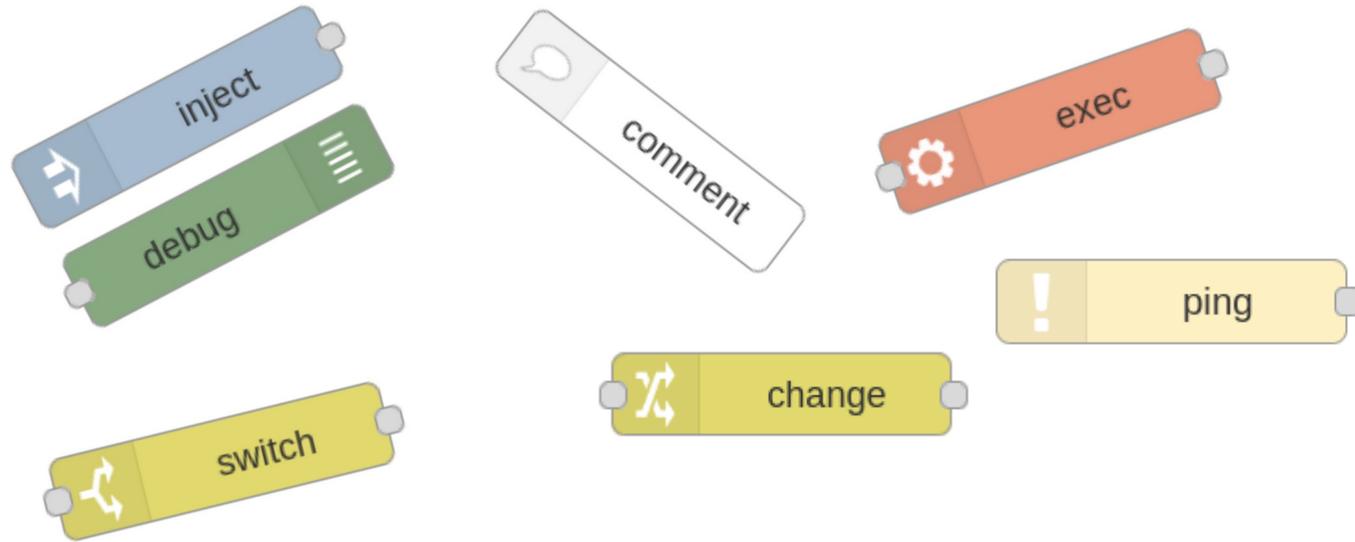
Node-Red



Utilisation.

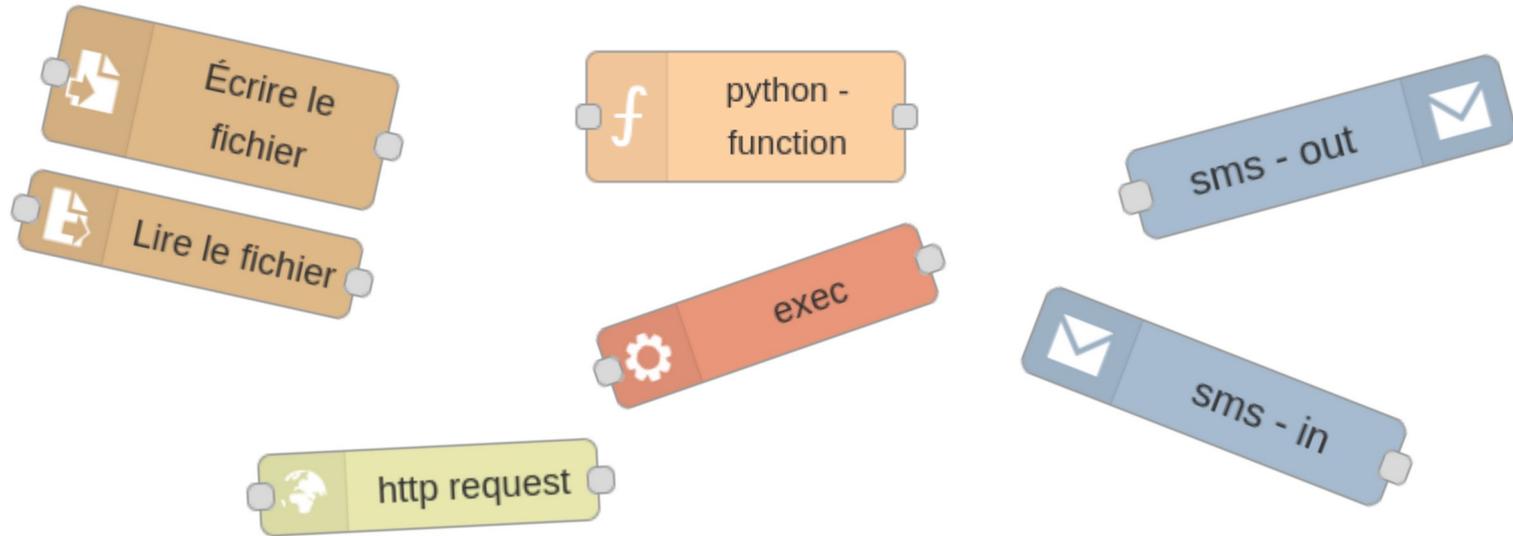
Node-Red

Utilisation de « nodes » simples

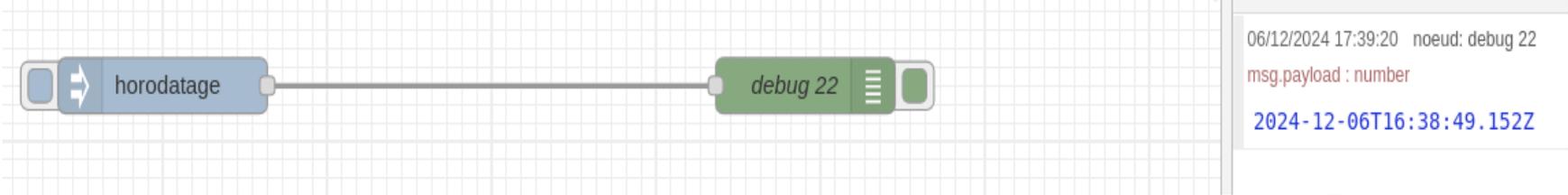


Node-Red

Utilisation de « nodes » plus spécialisés

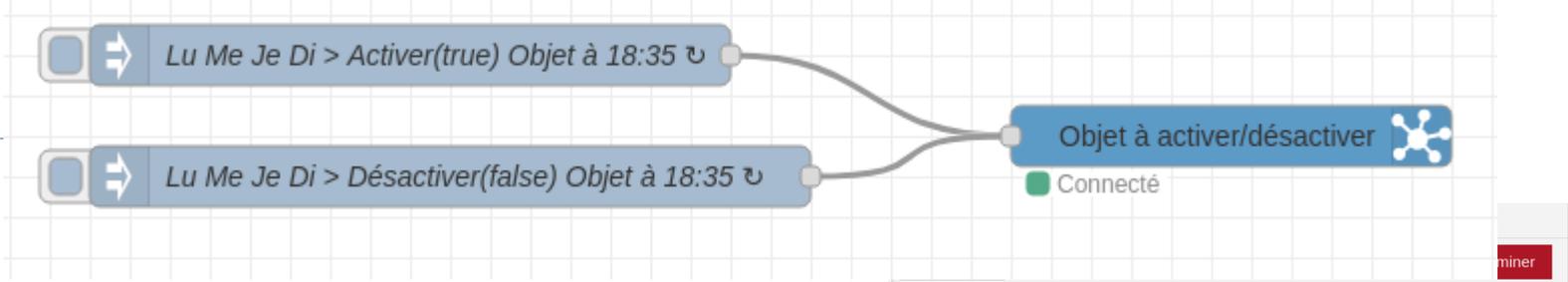


Node-Red



Node-Red

nœud « inject »



Doublic sur
nœud « inject »

The screenshot shows the "Propriétés" (Properties) panel for an inject node. The "Nom" (Name) field contains "Lu Me Je Di > Activer(true) Objet à 18:35". The "Payload" field is set to "msg. payload" and the "Type" dropdown is set to "true". Below the payload field, there are options for "Ajouter" (Add) and "Injecter maintenant" (Inject now). The "Répéter" (Repeat) checkbox is checked, and the "à une heure précise" (at a specific time) dropdown is selected. The time is set to "à 18:35". The "les" (on) days are checked for "Lundi" (Monday), "Jeudi" (Thursday), and "Dimanche" (Sunday), while "Mardi" (Tuesday), "Vendredi" (Friday), and "Mercredi" (Wednesday) are unchecked. At the bottom, there is a status indicator "Active" with a radio button.

Node-Red

The image shows a Node-Red workflow on a grid background. A yellow node labeled "192.168.89.150" with a warning icon is connected to three green nodes: "objet complet du message", "msg.payload (mS)", and "msg.topic".

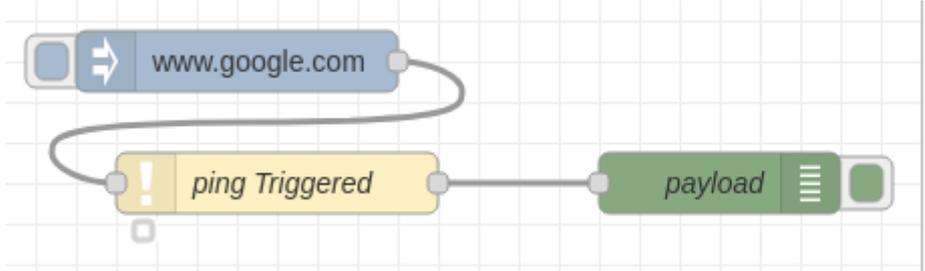
The console log on the right shows the following messages:

```
16/12/2024 05:21:52 noeud: objet complet du message
192.168.89.150 : msg : Object
  { payload: 215, topic: "192.168.89.150", _msgid: "d586e59fc91cf8a1" }
16/12/2024 05:21:52 noeud: msg.payload (mS)
192.168.89.150 : msg.payload : number
215
16/12/2024 05:21:52 noeud: msg.topic
192.168.89.150 : msg.topic : string[14]
"192.168.89.150"
```

Two configuration panels are shown below the flow:

- Modifier le noeud ping**: Target: 192.168.89.150, Protocol: IPv4, Mode: Timed, Ping (S): 60.
- Modifier le noeud debug**: Sortie: msg. topic, Vers: fenêtre de débogage, console du système: , état du noeud (32 caractères): , Nom: topic.

Node-Red



www.google.com : msg.payload : number
162



Modifier le noeud change

Supprimer Annuler Terminer

Propriétés

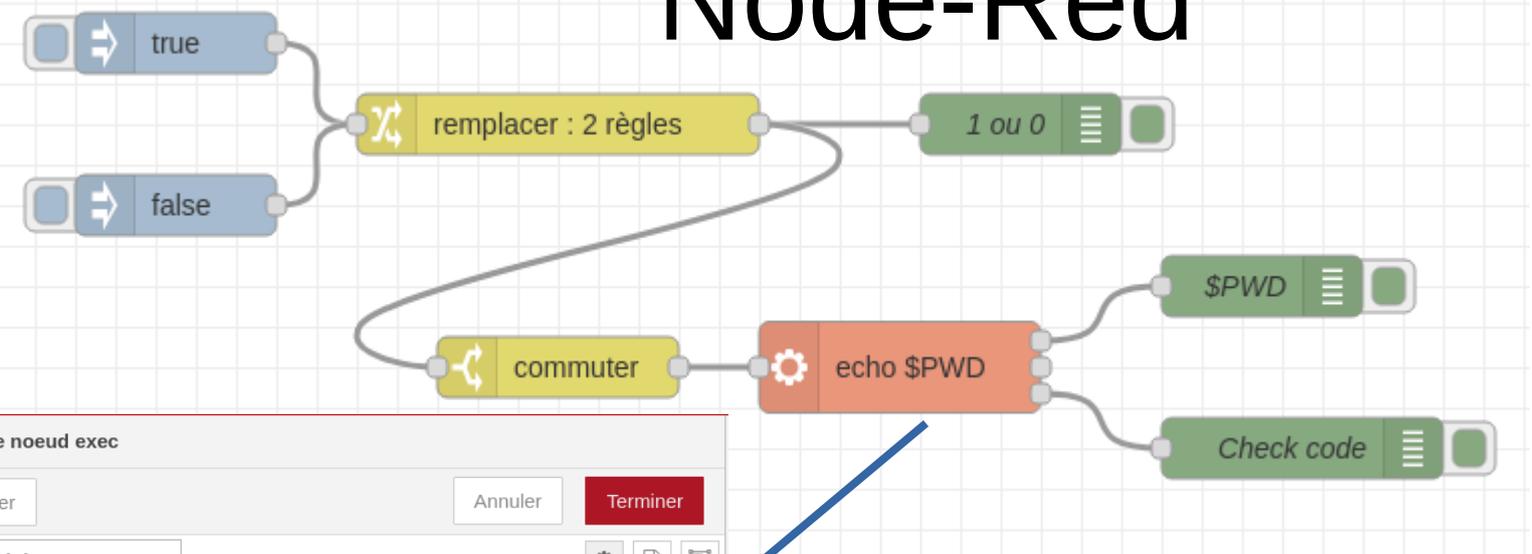
Nom

Règles

- Remplacer msg. payload
- Rechercher true
- Remplacer par 1
- Remplacer msg. payload
- Rechercher false
- Remplacer par 0

+ Ajouter

Node-Red



```
06/12/2024 18:35:11 noeud: check code  
msg.payload : Object  
▼ object  
  code: 0
```

Modifier le noeud exec

Supprimer Annuler Terminer

Propriétés

Commande: echo \$PWD

Joindre: msg. payload

Paramètres d'entrée supplémentaires

Sortie: lorsque la commande est terminée - mode exec

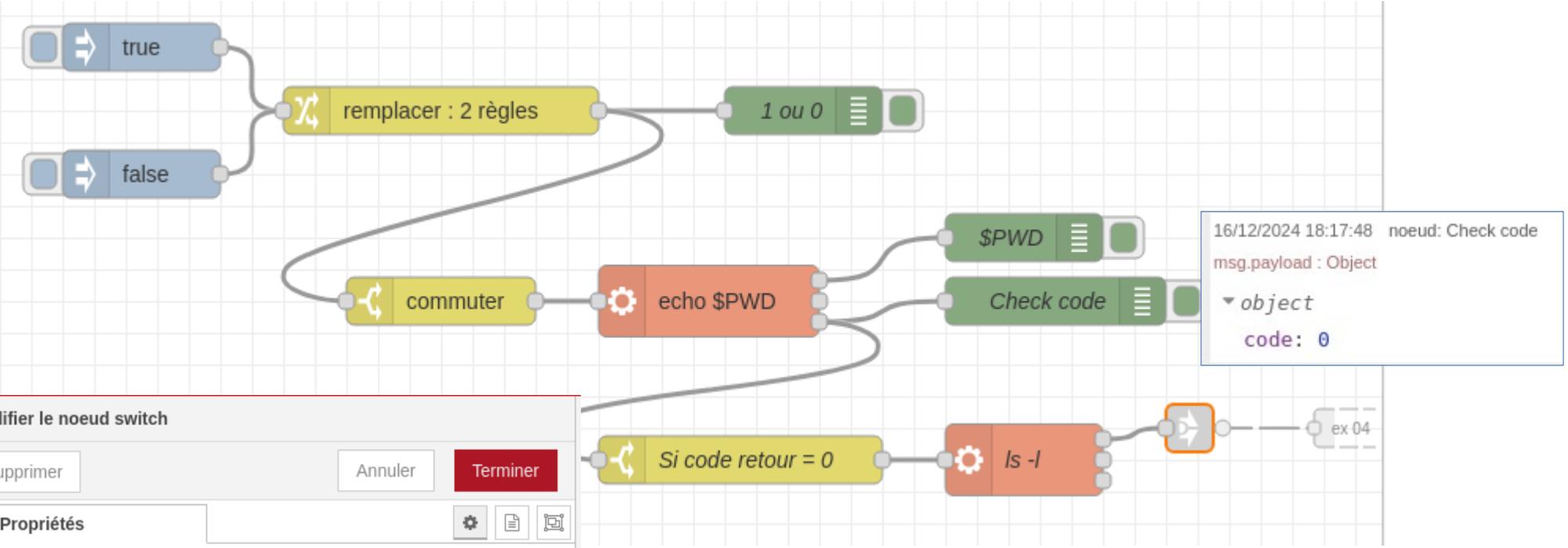
Temps mort: Facultati secondes

Masquer la console:

Nom: Nom



Node-Red



Modifier le noeud switch

Supprimer Annuler Terminer

Propriétés

Nom: Si code retour = 0

Propriété: msg.payload.code

0

Node-Red

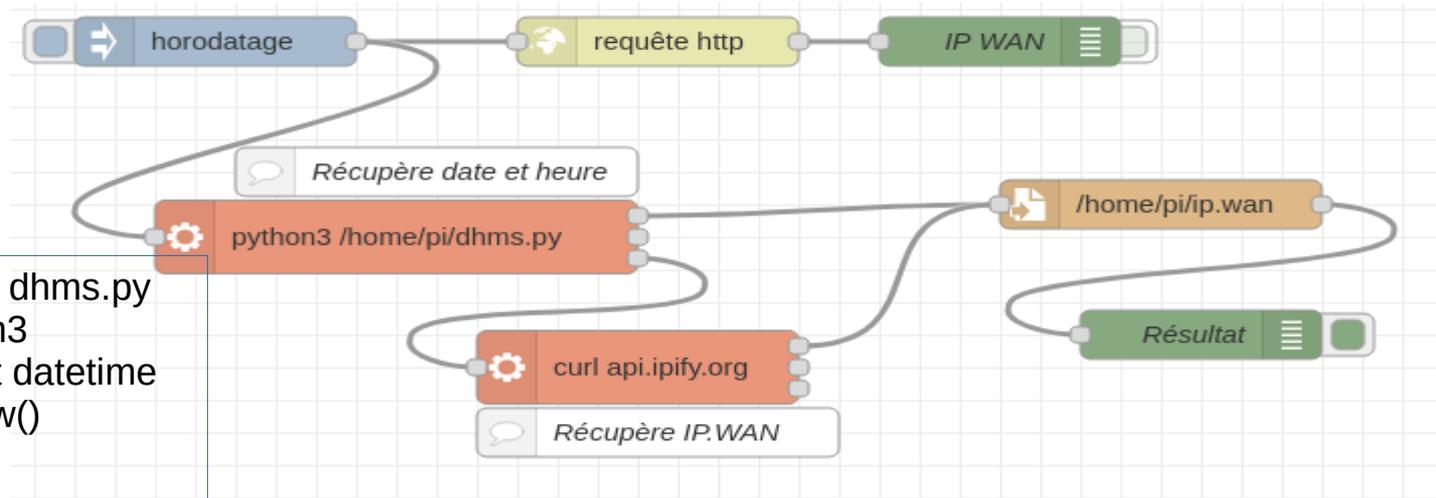
The diagram illustrates a Node-Red workflow. It starts with a 'horodatage' (timestamp) node, followed by an 'requête http' (HTTP request) node, and finally an 'IP WAN' node. A configuration panel for the 'requête http' node is open, showing the following settings:

- Méthode: GET
- URL: http://api.ipify.org
- Charge utile: Ignorer
- Activer la connexion sécurisée (SSL/TLS):
- Utiliser l'authentification:
- Activer le maintien de la connexion:
- Utiliser un proxy:
- N'envoyer que des réponses non-2xx au noeud Catch:
- Désactiver l'analyse HTTP stricte:
- Retourne: une chaîne UTF-8

The message inspector on the right shows the output of the 'IP WAN' node:

```
msg : Object
  object
    _msgid: "3b61157cf17a57f1"
    payload: "193.121.190.147"
    topic: ""
    statusCode: 200
  headers: object
    responseUrl: "http://api.ipify.org/"
    redirectList: array[0]
    retry: 0
```

Node-Red



```
pi@smstools:~ $ cat dhms.py
#!/usr/bin/env python3
from datetime import datetime
today = datetime.now()
print(today)
```

Modifier le noeud exec

Supprimer

Propriétés

Commande

Joindre

Modifier le noeud exec

Supprimer

Propriétés

Commande

Joindre

Modifier le noeud Écrire le fichier

Supprimer

Annuler

Terminer

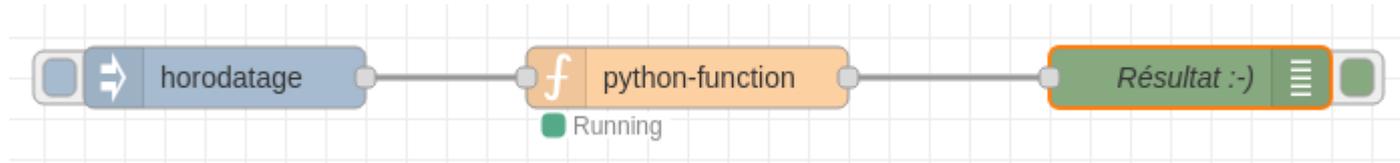
Propriétés

Nom du fichier

Action

Ajouter une nouvelle ligne (n) à chaque charge ?

Node-Red



Paramètres de l'utilisateur

Affichage: **Installer**

Palette:

Clavier: [node-red-contrib-python-function](#)

Environment: Define a function with Python instead of Javascript

Modifier le noeud python-function

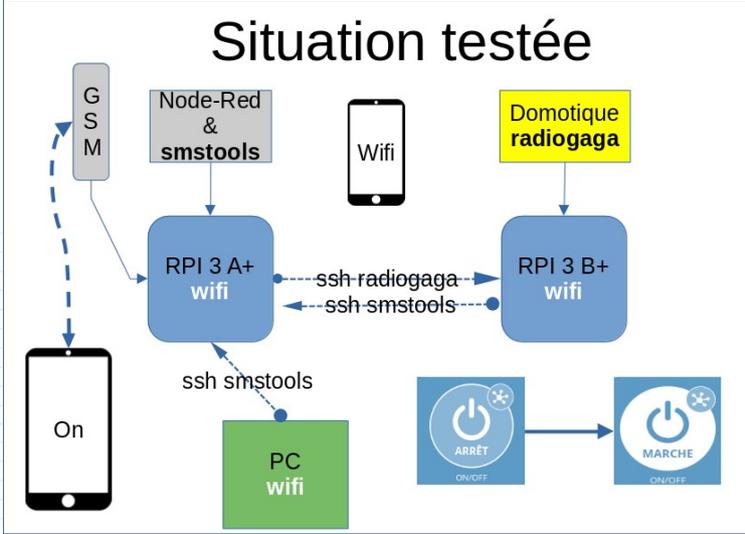
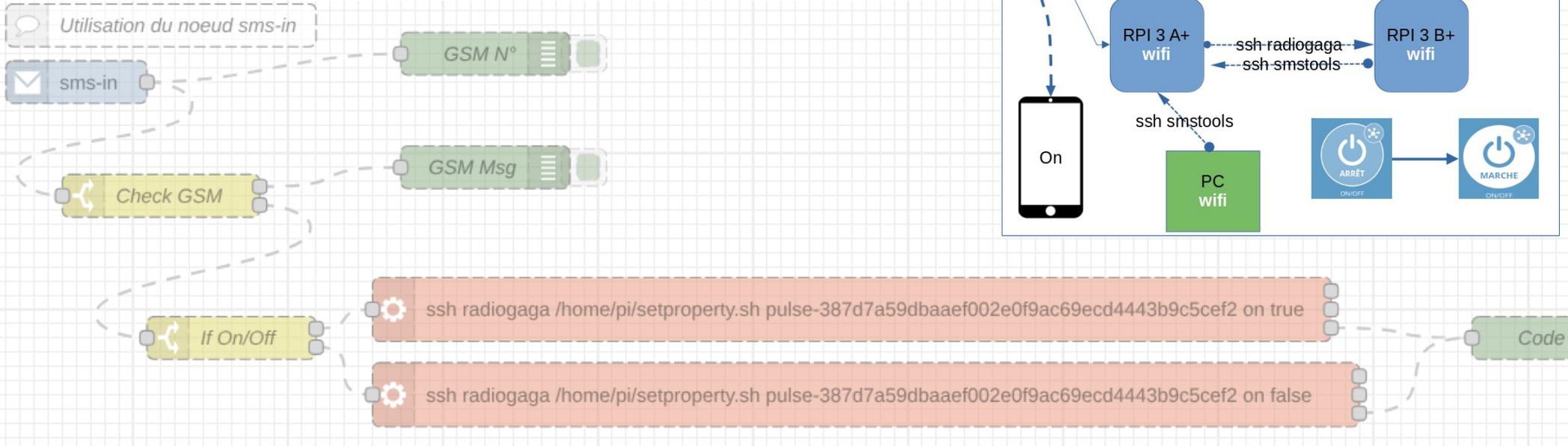
Propriétés

Name:

Function

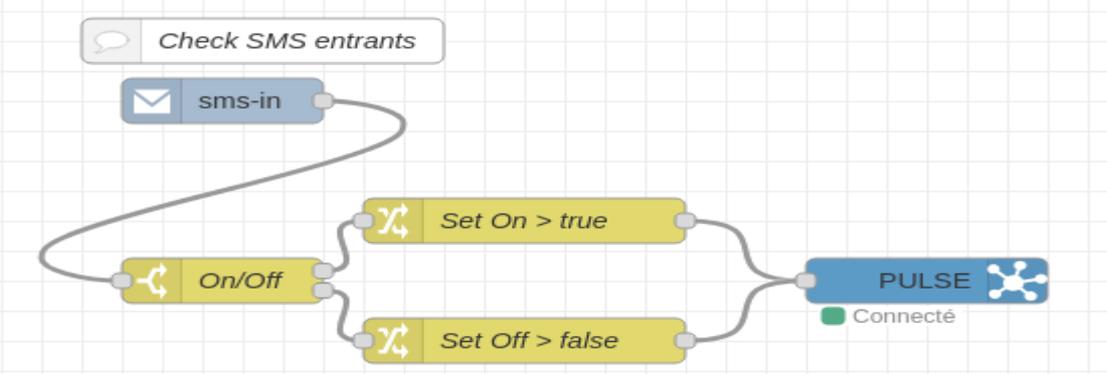
```
1 from datetime import datetime
2 today = print(datetime.now())
3 return msg
```

Node-Red



sms-in
 Listen for incoming SMS
 node-red-contrib-smstools : sms-in

Node-Red



Modifier le noeud Set property > Modifier le noeud webthingsio-gateway

Supprimer Annuler Sauver

Propriétés

Name:

Host:

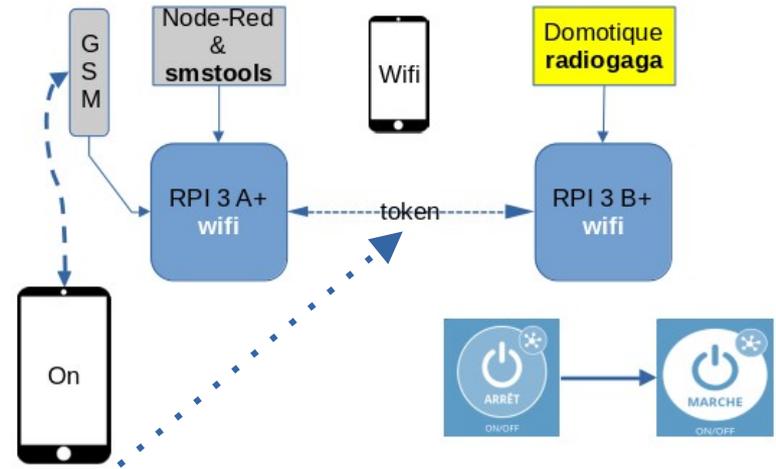
Port:

HTTPS?

Access token (aka local authorization):

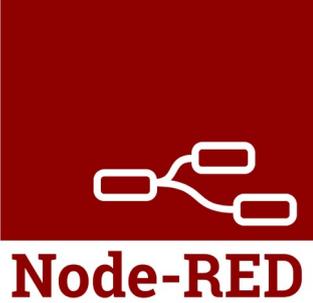
Skip validation?

Situation testée



Dans l'application de domotique, un jeton peut être créé afin d'accéder à la passerelle

Node-Red



Sites « node-red » en francais :

<https://silanus.fr/sin/?p=984>

<https://www.automation-sense.com/blog/automatisme/node-red-francais.html>

<https://www.conrad.ch/fr/guides/domotique/node-red.html>

<https://www.rgot.org/initiation-node-red/>

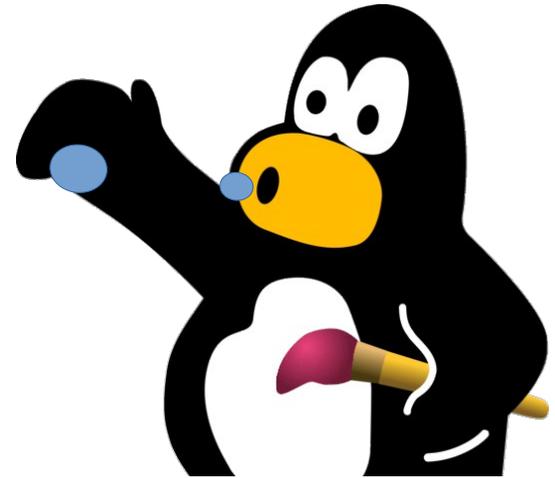
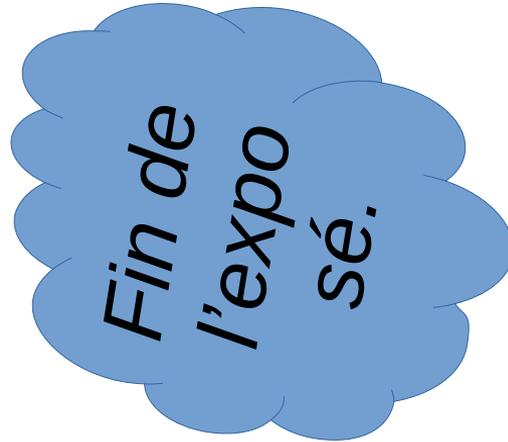
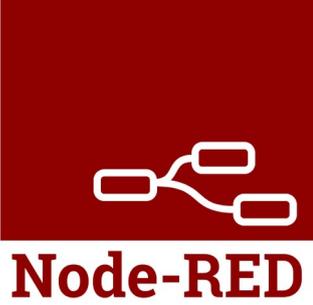
<https://deltalabprototype.fr/wp-content/uploads/2019/10/Serveur-NodeRED.pdf>

<https://blog.burelli.fr/domotique/nodered-first-start/>

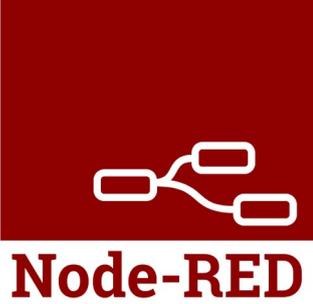
<https://www.stackhero.io/fr-fr/services/Node-RED/documentations>

.....

Node-Red

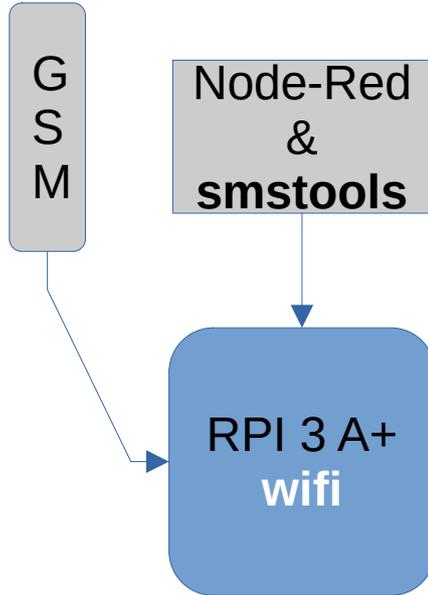


Node-Red

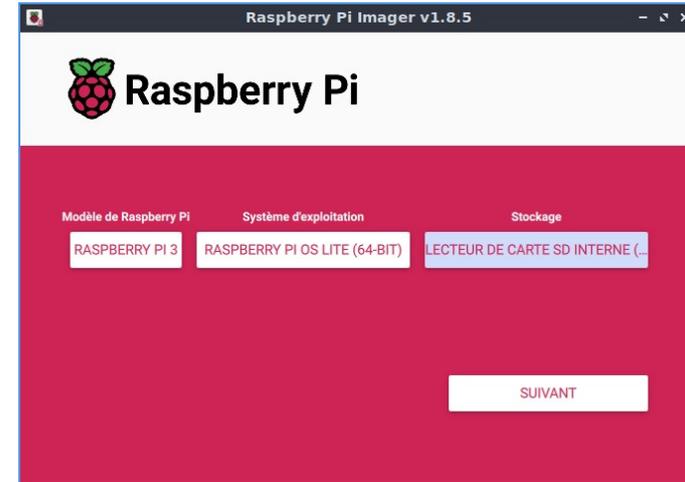
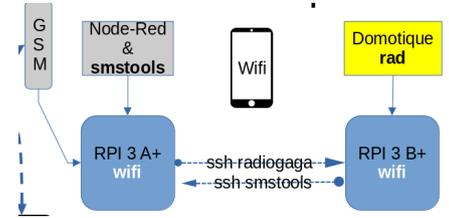


Détail de l'installation des logiciels
sur les RPI.

Etape 01



- Installer raspberry pi OS Lite (imager)
Préconfigurer wifi et activer ssh
- `sudo apt update && sudo apt upgrade`
- Installer smstools
sudo apt install smstools
- Installer et configurer la clef Huawei
Configurer /etc/smsd.conf
- Lancer le serveur smstools
au démarrage du RPI !
- Préparer une liaison ssh par clefs
- Installer Node-Red
<https://nodered.org/docs/getting-started/raspberrypi> (**lien**)
node-red-start(stop)
sudo systemctl enable nodered.service



Etape 02

- Installer raspberry pi OS Lite
sudo apt update && sudo apt upgrade
- Installer application domotique
<https://webthings.io/gateway/#download>
Placer un objet « pulse »
- Préparer une liaison ssh par clefs
- Établir une liaison ssh entre smstools et radiogaga
Au départ d'un rpi, envoyer la clef publique au rpi distant
Créer un fichier « config » dans .ssh
Host smstools (radiogaga)
HostName 192.168.112.149 (150)
User pi
IdentityFile ~/.ssh/smstools (radiogaga)

