NRED2 **Réception de requête http sous Node-RED** Durée 1h30.

GColin

Objectif : Mettre en place un serveur http et récupérer les données envoyées par le client.

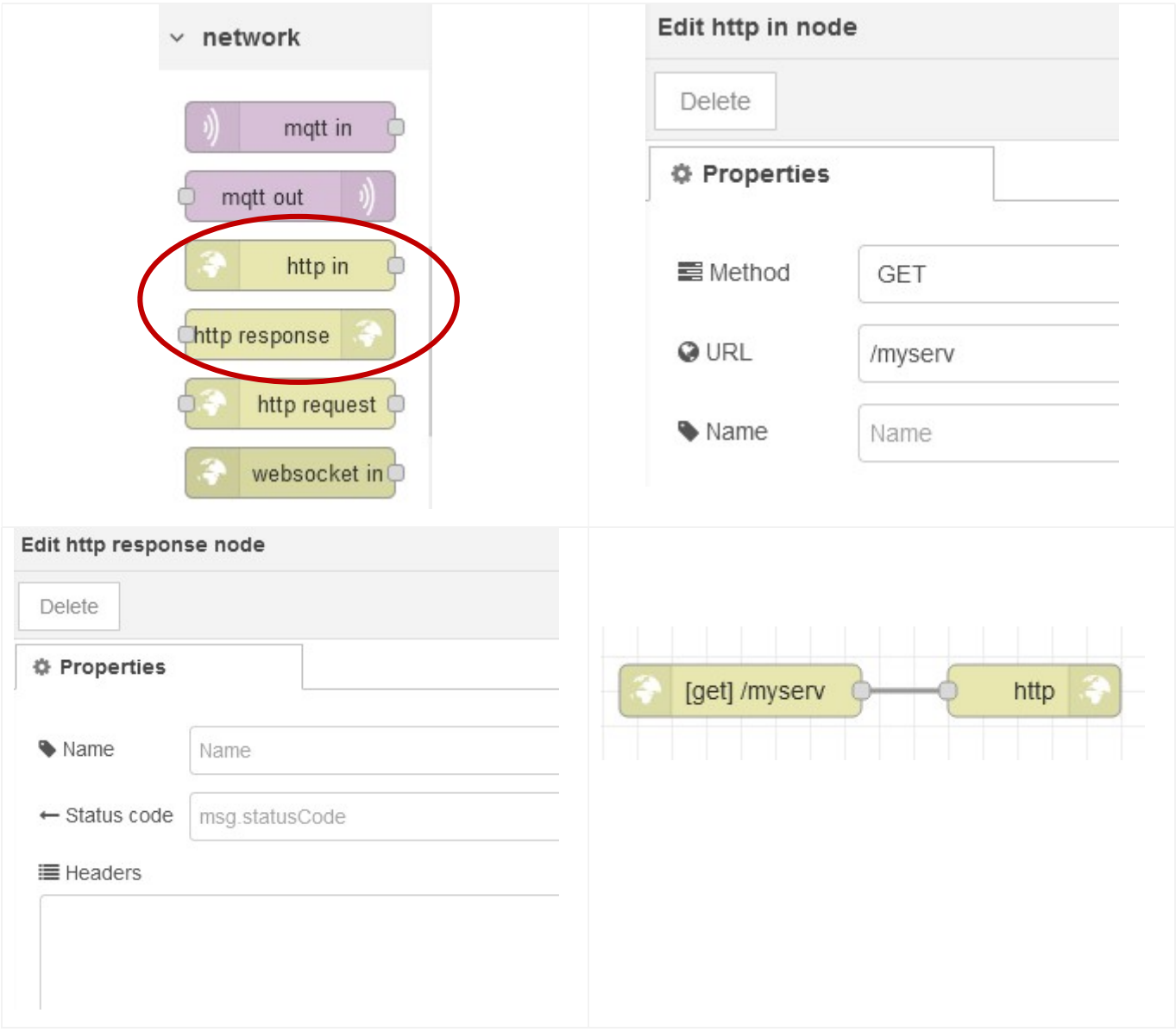
On utilisera le logiciel curl pour faire les requêtes http.

1 – Requête GET

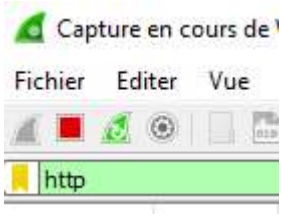
⇨ Lancer Node-RED sur la Raspberry

⇨ Placer les ***nodes*** "http in" et "http response" sur le plan, les configurer et les relier comme ci-dessous (pas de modification sur le node http response). **Ne pas oublier de déployer**.

⇨ indiquer /myserv pour l'url



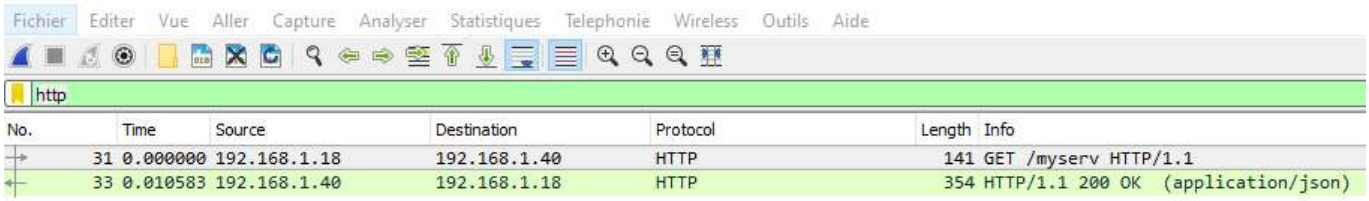
⇨ Lancer WireShark sur le PC et placer un filtre http



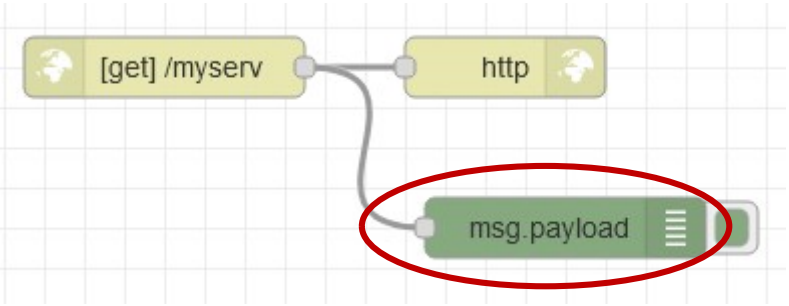
⇨ Sous l'invite de commande du PC, taper la commande suivante en remplaçant l'adresse IP de la raspberry :

curl "192.168.1.40:1880/myserv"

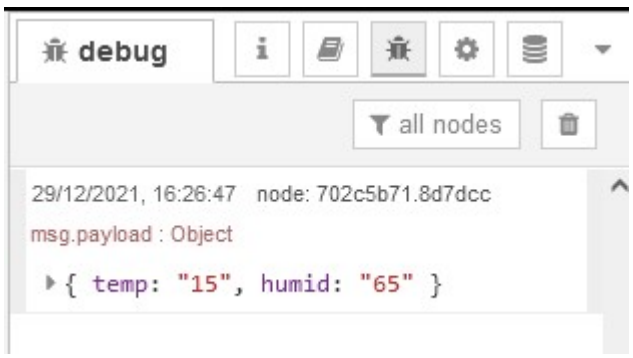
⇨ Vérifier les échanges sous WireShark comme ci-dessous :



⇨ Ajouter le node "debug" comme ci-dessous



⇨ Se placer en mode debug pour voir les messages, comme ci-dessous

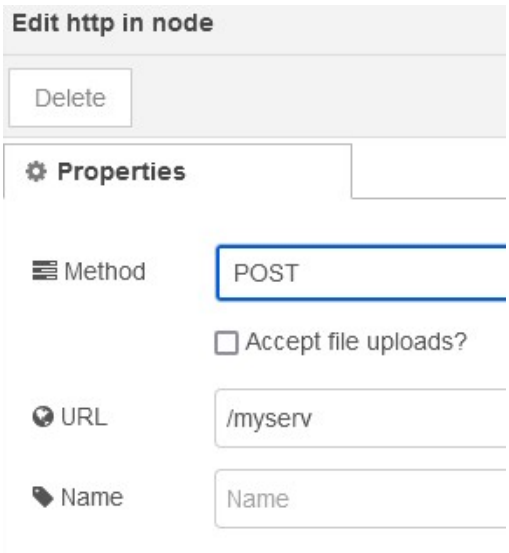


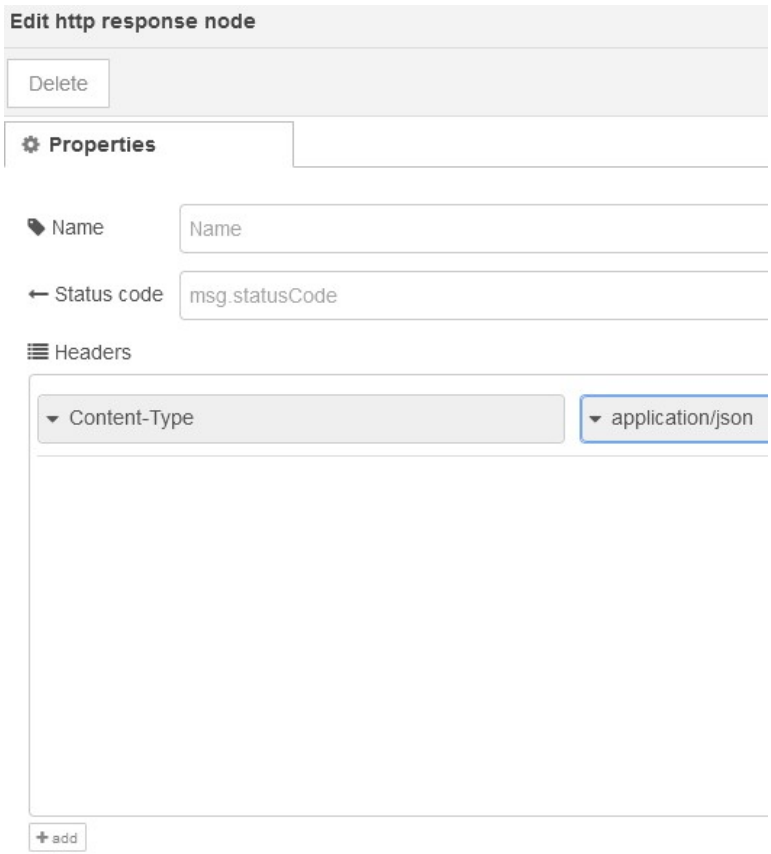
⇨ Lancer la requête http ci-dessous (modifier l'IP Raspberry) et observer le message dans la fenêtre debug.

curl "192.168.1.40:1880/myserv?temp=15&humid=65

2 – Requête POST

⇨ Modifier les 2 nodes précédents comme ci-dessous (méthode POST et ajout d'une entête type json en appuyant sur +add).





⇨ Sous l'invite de commande du PC, lancer notepad pour éditer et enregistrer le fichier capteur.json ci-dessous :

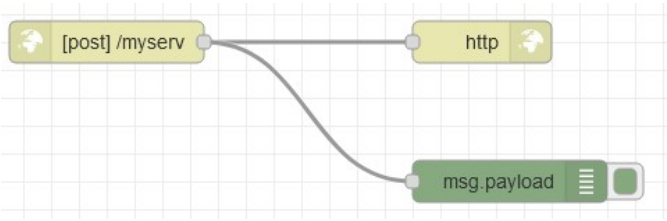
c:\users\dupont> notepad capteur.json

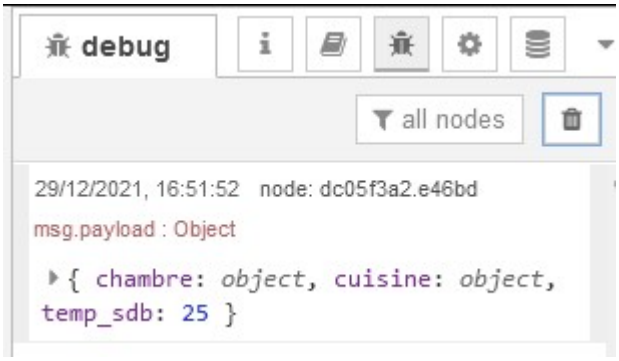
```
{
    "chambre":{
        "temperature":22,
        "id":"1234"
    },
    "cuisine":{
        "temperature":19,
        "id":"5678",
        "humidite":78
    },
    "temp_sdb":25
}
```

⇨ Lancer une requête http avec curl et envoi d'un fichier json (IP à modifier)

```
curl --data @capteur.json -H "Content-Type: application/json" "192.168.1.40:1880/myserv"
```

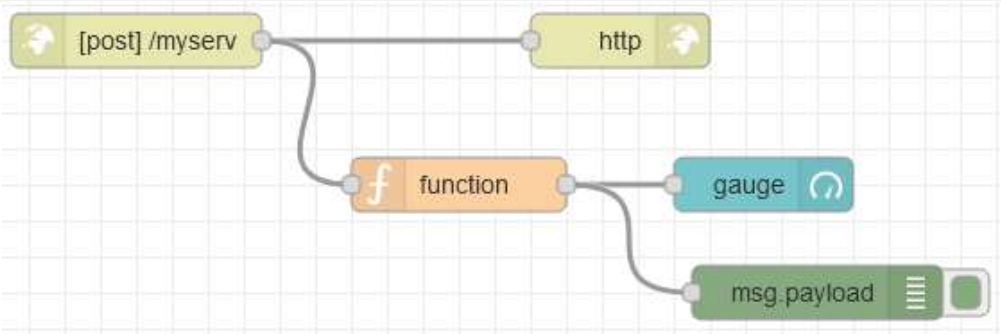
⇨ Observer les données dans la fenêtre debug

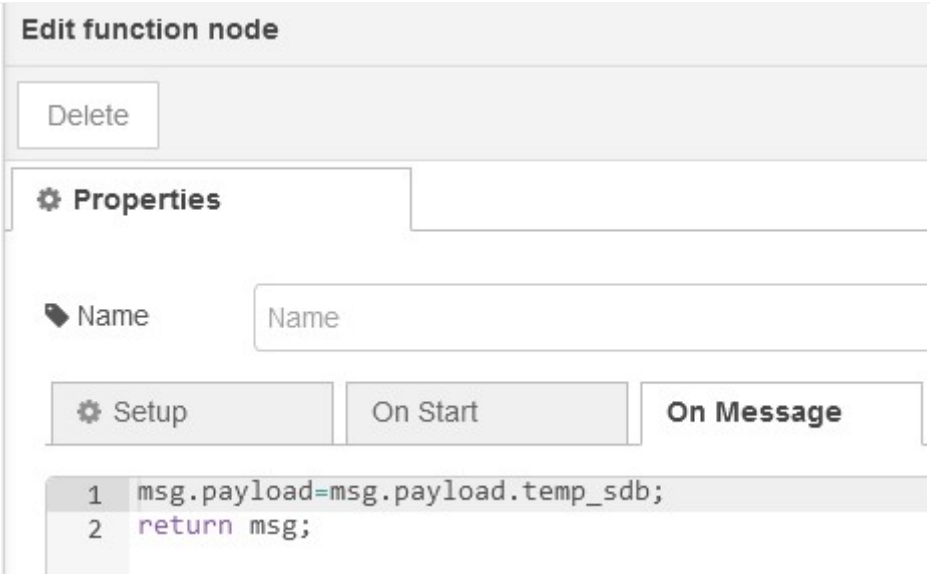


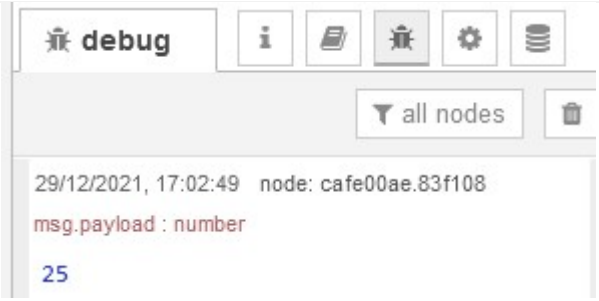


⇨ Ajouter une fonction et une jauge (échelle de 0 à 40) comme ci-dessous.

⇨ Relancer la même requête que précédemment et observer la donnée dans la fenêtre debug et dans l'interface utilisateur (192.168.1.40:1880/ui).







⇨ Modifier la fonction qui filtre temp_sdb par :

```
msg.payload=msg.payload.chambre.temperature;
return msg;
```

⇨ Vérifier le nouvel affichage.

⇨ Ajouter une jauge qui affiche en plus l'humidité de la cuisine et faire vérifier au prof.