

Projet 2025

Plateforme de gestion des informations IoT & Cybersécurité

Version 19 mai 2025 (18:33)

1 Cahier des charges

On désire concevoir par nos propres moyens une plateforme de gestion des informations en IoT (*Internet of Things*, c'est-à-dire les objets connectés).

L'architecture générale de la solution qu'il vous est demandé de mettre en place est représentée sur la figure 1. Vous pouvez remarquer que les connexions IoT se feront au travers du protocole MQTT (*Message Queuing Telemetry Transport*). Soit directement depuis le device, soit par l'intermédiaire d'un programme (ex : Python) qui collecte les données et les retransmet à la plateforme via MQTT (*publish*). D'un autre côté vous avez un ou des programmes (ex : Python) qui récupèrent ces données depuis MQTT (*subscribe*) et les stockent (ex : base de données SQL). D'autres programmes pourraient également enregistrer directement des données dans la BdD. Finalement, à partir de cette BdD, différents programmes peuvent présenter les données à l'utilisateur : des scripts PHP qui génèrent des pages HTML, des programmes Python (ou autre) qui génèrent des documents PDF, des images PNG, etc.

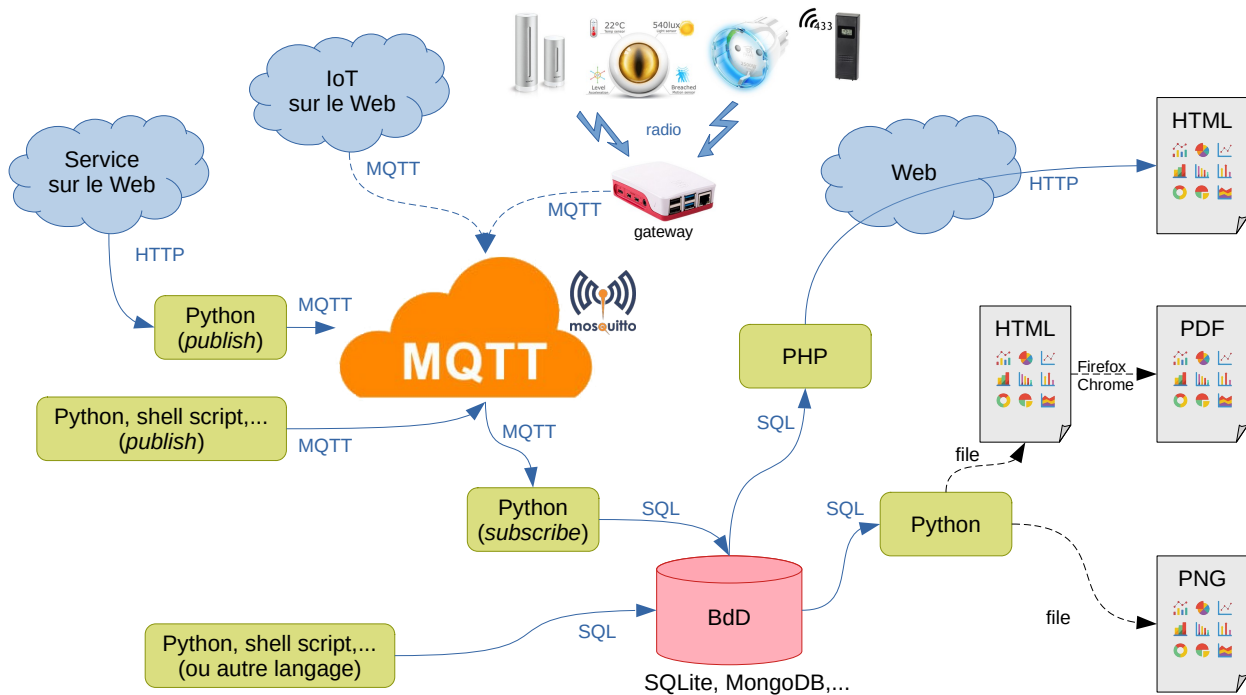


FIGURE 1 – Architecture générale

2 Travail à réaliser

2.1 Plateforme IoT

- **Ce qu'il faut absolument faire :**
 - programme Python qui envoie des données via MQTT
 - programme Python qui reçoit des données via MQTT et les stocke dans une BdD
 - site web dynamique (aka scripts PHP) qui présente les données via une interface web avec la génération de graphiques
- **Ce qui est "fortement" conseillé :**
 - dans les messages MQTT, la "charge utile" (le *payload*) est une chaîne de caractères → pour envoyer des données structurées on peut utiliser du JSON par exemple
 - écrire des programmes qui vont chercher des données sur le web (ex : Open Data) et les envoient sur notre plateforme en MQTT
 - sur la base des données collectées, calculer "d'autres" données ; ex : différence de température entre 2 villes, moyenne des températures d'une zone géographique donnée, etc. ; NB : étant donné que toutes les données n'arriveront pas forcément "au même rythme", à vous de trouver une solution...
- **Ce qui serait un plus :**
 - écrire un programme qui génère un document PDF à partir des données
 - écrire un programme qui génère une image PNG à partir des données ; cette image pourrait d'ailleurs ensuite être incluse dans une page web (NB : moins dynamique que si c'est un script PHP qui génère cette image à la volée)

2.2 Cybersécurité

Comme vous pourrez le constater, les technologies utilisées dans ce projet, du moins dans leur utilisation "basique", ne sont pas sécurisées. Dans cette partie il vous est demandé d'étudier les différentes vulnérabilités auxquelles pourrait être exposée une telle plateforme de gestion des informations IoT.

Pour ce travail vous devrez sortir de votre zone de confort. Ce projet est en effet orienté sur la "**gestion des informations**". Outre les problèmes et solutions de sécurité "classiques" au niveau du réseau, il existe également des vulnérabilités au niveau informationnel : capteurs défectueux envoyant des données aberrantes, données manquantes, empoisonnement des données, etc.

- **Ce qu'il faut faire :**
 - identifier les différentes vulnérabilités ; les catégoriser (réseau, information, etc.) ; veiller à citer les sources utilisées !
 - tester l'exploitation de ces vulnérabilités et observer leur(s) impact(s) (ex : sur vos tableaux de bord)
 - trouver et mettre en œuvre des mesures de sécurité adéquates
 - documenter vos travaux → rapport cybersécurité

3 Pour vous aider

Histoire de vous aider un petit peu, j'ai écrit un programme Python qui récupère les informations météo sur OpenWeather (<https://openweathermap.org/>) pour l'emplacement "IUT MdM" et les republie sur

des topics MQTT. J'ai mis ce programme dans la `crontab` d'un Linux afin qu'il soit automatiquement exécuté toutes les 2mn. Voici une trace d'un programme qui écouterait sur ce(s) topic(s) (sur le broker `test.mosquitto.org`).

```
Received `17.74` from `/rtmdm/mm/sae203/g553dtQJS7yk2G/weather/temp` topic
Received `17.4` from `/rtmdm/mm/sae203/g553dtQJS7yk2G/weather/feels_like` topic
Received `16.73` from `/rtmdm/mm/sae203/g553dtQJS7yk2G/weather/temp_min` topic
Received `19.42` from `/rtmdm/mm/sae203/g553dtQJS7yk2G/weather/temp_max` topic
Received `1020` from `/rtmdm/mm/sae203/g553dtQJS7yk2G/weather/pressure` topic
Received `70` from `/rtmdm/mm/sae203/g553dtQJS7yk2G/weather/humidity` topic
Received `{"temp": 17.74, "feels_like": 17.4, "temp_min": 16.73, "temp_max": 19.42,
"pressure": 1020, "humidity": 70}` from `/rtmdm/mm/sae203/g553dtQJS7yk2G/weather/all` topic
```

Edit 10/05/2022 Mon tutoriel "MQTT & Python" spécialement créé pour cette SAÉ disponible à l'URL suivante : https://munier.perso.univ-pau.fr/tutorial/iot/2022/20220510-mqtt_python/

Edit 01/12/2023 Ce projet de SAÉ a fait l'objet d'une publication au *5ème Workshop Pédagogique Réseaux & Télécoms* fin 2023. Lien disponible [ici](#).

Edit 14/05/2024 Suite à quelques adaptations (pour réutiliser ces informations dans le cadre d'un projet de recherche), j'ai modifié les messages publiés sur le topic `[...]/all` : au lieu d'envoyer un unique message JSON avec toutes les mesures, j'envoie un message JSON pour chaque mesure mais en y ajoutant différentes métadonnées.

Si vous avez installé Eclipse Mosquitto sur votre machine, les 2 commandes côté client `mosquitto_pub` et `mosquitto_sub` devraient également avoir été installées. Elles permettent respectivement de publier et de souscrire sur un broker MQTT directement depuis la ligne de commandes. Dans le cas présent, vous pouvez exécuter la commande suivante :

```
mosquitto_sub -v -h test.mosquitto.org -t /rtmdm/mm/sae203/g553dtQJS7yk2G/weather/#
```

L'option `-v` permet de préfixer chaque message par le topic sur lequel il a été reçu. Voici un exemple de sortie dans le terminal :

```
/rtmdm/mm/sae203/g553dtQJS7yk2G/weather/temp 14.89
/rtmdm/mm/sae203/g553dtQJS7yk2G/weather/all {"name": "foo.owm.1715937603.0767155", "results": [{"label":
→ "temp", "value": 14.89, "unit": "Celsius"}], "collections": ["IUT_MdM_weather"], "featureOfInterest":
→ "IUT_MdM_temp"}
/rtmdm/mm/sae203/g553dtQJS7yk2G/weather/feels_like 13.61
/rtmdm/mm/sae203/g553dtQJS7yk2G/weather/all {"name": "foo.owm.1715937603.0771918", "results": [{"label":
→ "feels_like", "value": 13.61, "unit": "Celsius"}], "collections": ["IUT_MdM_weather"],
→ "featureOfInterest": "IUT_MdM_feels_like"}
/rtmdm/mm/sae203/g553dtQJS7yk2G/weather/temp_min 11.94
/rtmdm/mm/sae203/g553dtQJS7yk2G/weather/all {"name": "foo.owm.1715937603.0774415", "results": [{"label":
→ "temp_min", "value": 11.94, "unit": "Celsius"}], "collections": ["IUT_MdM_weather"],
→ "featureOfInterest": "IUT_MdM_temp_min"}
/rtmdm/mm/sae203/g553dtQJS7yk2G/weather/temp_max 16.1
/rtmdm/mm/sae203/g553dtQJS7yk2G/weather/all {"name": "foo.owm.1715937603.0776632", "results": [{"label":
→ "temp_max", "value": 16.1, "unit": "Celsius"}], "collections": ["IUT_MdM_weather"], "featureOfInterest":
→ "IUT_MdM_temp_max"}
/rtmdm/mm/sae203/g553dtQJS7yk2G/weather/pressure 1009
/rtmdm/mm/sae203/g553dtQJS7yk2G/weather/all {"name": "foo.owm.1715937603.0780761", "results": [{"label":
→ "pressure", "value": 1009, "unit": "hPa"}], "collections": ["IUT_MdM_weather"], "featureOfInterest":
→ "IUT_MdM_pressure"}
```

```
/rtmdm/mm/sae203/g553dtQJS7yk2G/weather/humidity 45
/rtmdm/mm/sae203/g553dtQJS7yk2G/weather/all {"name": "foo.owm.1715937603.0783267", "results": [{"label":
→ "humidity", "value": 45, "unit": "%"}], "collections": ["IUT_MdM_weather"], "featureOfInterest":
→ "IUT_MdM_humidity"}
```

Évaluation

Pour ce projet vous travaillerez par binôme, ce qui évitera certains "problèmes" entre séances de TD et séances de TP. Votre évaluation pour ce projet en SAE203 se fera en trois parties :

1. une évaluation finale du projet terminé (lors de la dernière séance)
2. un suivi individualisé de l'avancement de votre projet au fur et à mesure des séances
3. un rapport sur la partie cybersécurité

Infos complémentaires

- Si par hasard le broker `test.mosquitto.org` était indisponible, vous pouvez le remplacer par un des brokers MQTT suivants :
 - `broker.hivemq.com` (cf. <https://broker.hivemq.com>)
 - `broker.emqx.io` (cf. <https://www.emqx.com/en/mqtt/public-mqtt5-broker>)