



REEL Demo – Romande Energie ELectric network in local balance Demonstrator

Deliverable: 3a1 Communication specification GridEye
and DSM

Demo site: Rolle

Developed by
Peter Gysel, FHNW
Michael Roth, FHNW
Dominik Link, FHNW
in collaboration with
Fabrizio Sossan, EPFL

[Lausanne, 28.07.2018]

1. Description of deliverable and goal

1.1. Executive summary

The focus for this task was defining and building the central server infrastructure for the REEL demonstrator. This infrastructure is needed to store and distribute data from various project members such as DepSys and InnoSense as it can be seen in Figure 1. The data can be used for analyzing electrical grid parameters. In Addition, commands should be sent to enable or disable high load devices such as boilers to stabilize the grid. It is necessary to build security functions from the beginning. Otherwise unauthorized people could access measured data from households and gain insights on personal behavior. They could even control devices such as boilers over the internet and cause high loads in a short time.

REEL project partners decided to use the time series database influxdb as they already used it in previous research projects. Our main task was to build a proxy for the Application Programmable Interface (API) over which data can be stored and retrieved to the central database 'influxdb'. The severely limited authentication and authorization mechanisms of influxdb can be extended with this approach.

1.2. Research question

How could we secure the central database access and validate data before it is saved?

1.3. Novelty of the proposed solutions compared to the state-of-art

Using the JavaScript proxy express gateway (<https://www.express-gateway.io>), we built a proxy, which is able to protect the central database. Using a proxy to protect an API is a common concept. However, we did not find a similar opensource solution which is tailored to influxdb API. There is a paid version of influxdb which contains proprietary authentication and authorization. However, data validation is not a part of this product. Therefore, a custom solution was needed which we implemented by configuring and extending express gateway.

1.4. Description

Express gateway is a JavaScript application that can be configured and extended. The Express gateway acts as an API proxy, which receives requests from various clients and

redirects them to the database after successful user authentication, authorization and data validation.

It can be configured to act as a TLS server to secure the communication between clients and itself, therefore only one certificate is sufficient to protect multiple backends.

It uses a local installed redis database to store user data such as users and their credentials.

Figure 1 shows the architecture we built for securing the central database. Parties such as InnoSense can put data into the central database through accessing the express gateway proxy over an encrypted channel. The proxy checks authentication and access privileges through the configuration saved to the local redis database. On success, it redirects the request to the influxdb database.

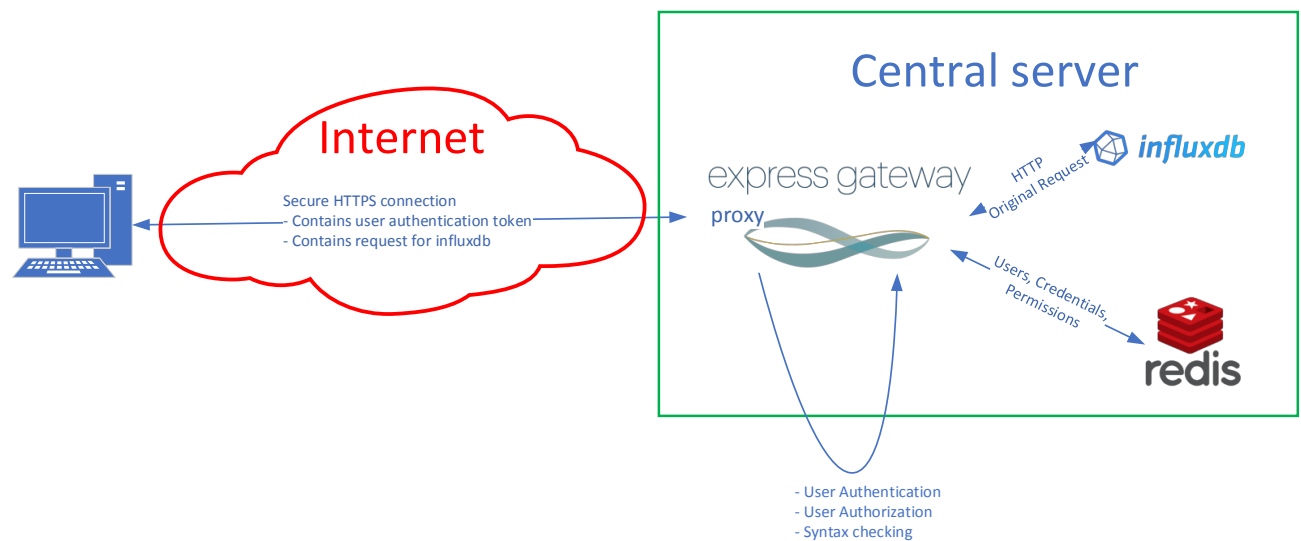


Figure 1: Architecture of the central database

Authentication and authorization can be configured using policies. Detailed access privileges on certain time series can be achieved by extending influxdb with a JavaScript plugin. We demonstrated this successfully in our lab.

The gateway can also be configured and used as a proxy for relaying configuring commands to the GridSense server. This scenario results in less configuration tasks for InnoSense AG. For example, when a new party needs to send commands to a set of Grid Sense Units (GSU), the central server administrators could create users and privileges

centrally. Commands could then be relayed to the target server (e.g. GridSense DMS) using one set of credentials.

2. Achievement of deliverable:

2.1. Date

28. May 2018

2.2. Demonstration of the deliverable

We have built a local lab with virtualization technologies (virtualbox). We set up a influxdb instance and configured an express gateway instance which was used to proxy API requests from a client to the database.

Besides user management, we also built prototypes for data validation.

After successful tests, we deployed the gateway with 3 users (read, write, read+write privileges) on the central server which will eventually be located in the Romande Energie datacenter.

3. Impact

Authorized parties of the REEL demonstrator may get access to the central database to push and retrieve data. Unauthorized people are not able to retrieve or store data to the central database. As soon as we have detailed data schemes, we will be able to deploy detailed privileges and data validation.