

Côté informatique

Voici le principe de fonctionnement que je voulais faire initialement :

1. lecteur des informations sur le port série (côté local)
2. envoi sur serveur de base de données SGBD (côté serveur)

On y reviendra plus tard.

Pour l'instant j'installe tout sur le raspberryPi.

Installer PHP

La lecture sur des informations va s'effectuer en PHP. C'est le plus simple pour moi parce que c'est très proche du langage C. Dans un premier temps il faut installer l'interpréteur PHP :

```
sudo apt-get install php
```

On s'aperçoit que l'installation du package PHP intègre d'autres composants.

On retrouve Apache2 et des fonctions associées à SQLite3, LDAP et JSON.

Plusieurs fichiers de configuration sont créés :

- /etc/php/7.0/apache2/php.ini
- /etc/php/7.0/cli/php.ini
- fichiers présents dans /etc/php/7.0/mods-available/

Deux binaires sont accessibles :

- php
- phar

Lecture des listes de paquets...

Fait

Construction de l'arbre des dépendances

Lecture des informations d'état...

Fait

The following additional packages will be installed:

apache2 apache2-bin apache2-data
apache2-utils libapache2-mod-
php7.0 libapr1 libaprutil1
libaprutil1-dbd-sqlite3
libaprutil1-ldap php-common php7.0
php7.0-cli php7.0-common php7.0-
json

php7.0-opcache php7.0-readline
ssl-cert

Paquets suggérés :

apache2-doc apache2-suexec-
pristine | apache2-suexec-custom
php-pear openssl-blacklist

Les NOUVEAUX paquets suivants
seront installés :

apache2 apache2-bin apache2-data
apache2-utils libapache2-mod-
php7.0 libapr1 libaprutil1
libaprutil1-dbd-sqlite3
libaprutil1-ldap php php-common
php7.0 php7.0-cli php7.0-common

```
php7.0-json
php7.0-opcache php7.0-readline
ssl-cert
0 mis à jour, 18 nouvellement
installés, 0 à enlever et 0 non
mis à jour.
Il est nécessaire de prendre 4 582
ko dans les archives.
Après cette opération, 17,0 Mo
d'espace disque supplémentaires
seront utilisés.
Souhaitez-vous continuer ? [0/n]
```

Installer le SGBD PostgreSQL

J'ai installé le SGBD PostgreSQL ainsi que le bibliothèque de connexion PHP.

```
sudo apt-get install php-pgsql
postgresql
```

Une fois installé, les programmes n'occuperont que 30 Mo. Je n'ai pas l'habitude d'avoir une installation prenant aussi peu de place.

```
Résultat :
Lecture des listes de paquets...
Fait
Construction de l'arbre des
dépendances
Lecture des informations d'état...
Fait
The following additional packages
will be installed:
  libpq5 php7.0-pgsql
postgresql-9.6 postgresql-
client-9.6 postgresql-client-
common postgresql-common
  postgresql-contrib-9.6 sysstat
Paquets suggérés :
  postgresql-doc locales-all
postgresql-doc-9.6 libdbd-pg-perl
isag
Les NOUVEAUX paquets suivants
seront installés :
  libpq5 php-pgsql php7.0-pgsql
postgresql postgresql-9.6
postgresql-client-9.6
  postgresql-client-common
postgresql-common postgresql-
contrib-9.6 sysstat
0 mis à jour, 10 nouvellement
```

```
installés, 0 à enlever et 0 non
mis à jour.
Il est nécessaire de prendre 6 281
ko dans les archives.
Après cette opération, 30,2 Mo
d'espace disque supplémentaires
seront utilisés.
Souhaitez-vous continuer ? [0/n]
```

Paramétrer le SGBD PostgreSQL

Voilà comment je fais pour me connecter au moteur de SGBD PostgreSQL. J'effectue ces manipulations depuis le compte pi.

1. Je me connecte avec le compte du SGBD qui se nomme postgres :

```
sudo su postgres
```

2. Je me connecte au SGBD :

```
psql
```

Oh joie ! L'invite postgres=# apparaît.

Créer la base de données

Je vais utiliser les informations suivantes :

- base de données : ampere
- utilisateur de la base de données (rôle) : r_ampere

Il faut créer l'utilisateur de la base de données. Création du rôle r_ampere

De manière schématique, l'utilisateur de connexion à la base de données s'appelle un rôle.

```
CREATE ROLE r_ampere PASSWORD
'ThisIsN0tAnPwd' LOGIN VALID UNTIL
'infinity';
```

Il faut créer la base de données qui va accueillir les données.

```
CREATE DATABASE "ampere"
WITH OWNER "r_ampere"
ENCODING 'UTF8'
```

J'ai choisi d'utiliser l'encodage UTF-8 dans la base de données, et l'utilisation des règles de la langue Française (fr_FR).

```
LC_COLLATE = 'fr_FR.UTF-8'
LC_CTYPE = 'fr_FR.UTF-8';
```

Il faut autoriser les connexions. Pour cela, le fichier de configuration des clients authentifié doit être adapté. Lors de l'installation du SGBD, il a été créé dans

```
/etc/postgresql/9.6/main/pg_hba.conf
```

On s'identifie avec l'utilisateur postgres :

```
pi: $ sudo su - postgres
```

On ajoute dans le fichier

```
/etc/postgresql/9.6/main/pg_hba.conf :
```

Il faut l'adapter suivant le schéma suivant :

host	database	user	address	local	ampere	r_ampere
auth-method				trust		

On sort de l'environnement postgres :

```
postgres: $ exit
```

Une fois le paramétrage terminé, il faut redémarrer le SGBD :

```
pi: $ sudo service postgresql
reload
```

On teste la connexion :

```
pi: $ psql -U r_ampere -W -d
ampere
```

Principe de fonctionnement

Un programme doit sans cesse boucler pour effectuer cette action :

1. lire une trame sur /dev/ttyAMA0
readTrame
2. ajouter un timestamp dans la trame
3. mémoriser la trame brute avec timestamp dans une mémoire tampon -
saveTrameTampon

saveTrameBdd

saveTrameCsv

Insérer un fichier CSV dans une table MySQL

```
<?php
$databasehost = "localhost";
$databasename = "telereleve_elec";
$databasetable = "tr_journalier";
$databaseusername="Utilisateur";
$databasepassword = "motDePasse";
$fieldseparator = ",";
$lineseparator = "\n";
$csvfile = "releves/teleinfo_20181219.csv";

if(!file_exists($csvfile)) {
    die("File not found. Make sure you specified the correct path.\n");
}

try {
    $pdo = new PDO("mysql:host=$databasehost;dbname=$databasename",
        $databaseusername, $databasepassword,
        array(
            PDO::MYSQL_ATTR_LOCAL_INFILE => true,
            PDO::ATTR_ERRMODE => PDO::ERRMODE_EXCEPTION
        )
    );
} catch (PDOException $e) {
    die("database connection failed: ".$e->getMessage());
}

$affectedRows = $pdo->exec("
    LOAD DATA LOCAL INFILE ".$pdo->quote($csvfile)." INTO TABLE
    `".$databasetable.`
    FIELDS TERMINATED BY ".$pdo->quote($fieldseparator).`
    LINES TERMINATED BY ".$pdo->quote($lineseparator));

echo "Loaded a total of $affectedRows records from this csv file.\n";

?>
```

Transformer une donnée timestamp unix en format date et heure

```
SELECT FROM_UNIXTIME(`TIMESTAMP`) FROM `tr_journalier`
```

Bibliothèque PHP

Le programme sera écrit en PHP. On va au maximum utiliser des méthodes afin d'obtenir des informations sur la consommation :

- quelle est la consommation Intensité instantannée ? - getINST

- quelle est la valeur du compteur HC ? - getHCHC
- quelle est la valeur du compteur HP ? - getHCHP
- quelle est la période tarifaire en cours ? - getPTEC
- quelle est la puissance apparente ? - getPAPP

et obtenir des informations sur l'abonnement :

- quelle est le N° d'identification du compteur ? - getADCO
- quelle est l'Option tarifaire ? - getOPTARIF
- quelle est l'Intensité souscrite ? - getISOUSC

Je pense également à des méthodes d'analyse :

- quelle est la consommation HC/HP de la période tarifaire actuelle ? - getConsoPeriodeNow(tarif)
- quelle est la consommation HC/HP de la période précédente ? - getConsoPeriodePrev(tarif)
- quelle est la consommation HC/HP des 24 dernières heures ? - getConsoPeriode24(tarif)
- quelle est la consommation HC/HP d'une période de 7 jours X ? - getConsoPeriode7jours(tarif, dateHeureDebut)
- quelle est la consommation HC/HP de la minute X ? - getConsoMinute(tarif, dateHeure)
- quelle est la consommation HC/HP de l'heure X ? - getConsoHeure(tarif, dateHeure)
- quelle est la consommation HC/HP du jour X ? - getConsoJour(tarif, dateHeure)
- quelle est la consommation HC/HP de la semaine X ? - getConsoSemaine(tarif, date)
- quelle est la consommation HC/HP du mois X ? - getConsoMois(tarif, date)
- quelle est la consommation HC/HP de l'année X ? - getConsoAnnee(tarif, date)

Et ensuite on pourra partir sur des statistiques :

- moyenne
- tendance
- ...

Projets

http://vesta.homelinux.free.fr/wiki/demodulateur_teleinformation_edf.html

<http://lhuet.github.io/blog/2014/01/montage-teleinfo.html>

Projet de <http://www.magdiblog.fr/gpio/teleinfo-edf-suivi-conso-de-votre-compteur-electrique/>

Le relevé d'info compatible <http://hallard.me/teleinfo-emoncms/>

From:
<https://www.abonnel.fr/> - **notes informatique & technologie**

Permanent link:
https://www.abonnel.fr/electronique/teleinformation-compteur-electricite/cote_informatique

Last update: **2021/01/01 15:30**

