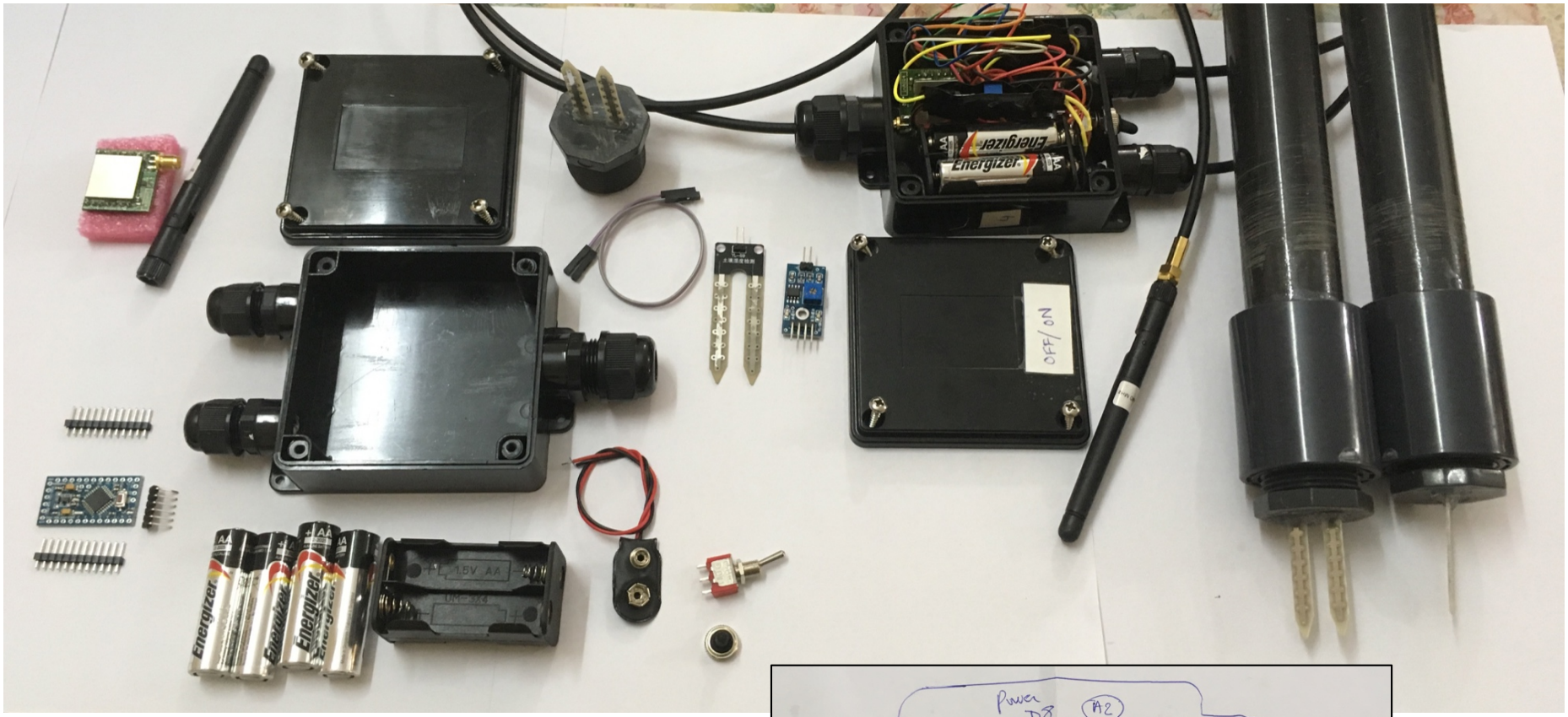
WaterSense Deployment Guidelines and Instructions

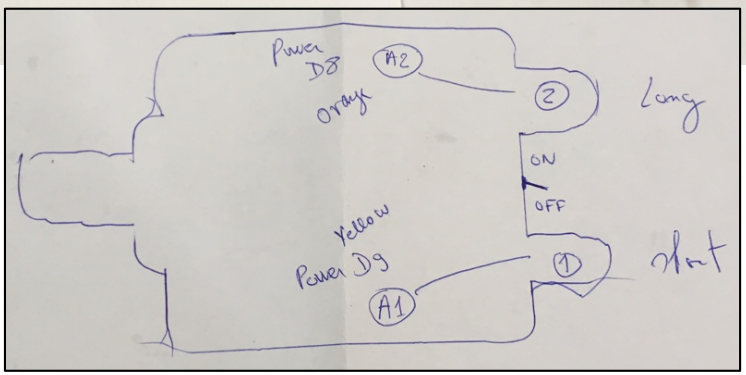
Prof. Congduc Pham
http://www.univ-pau.fr/~cpham
Université de Pau, France

Reading instructions

- Recommended reading:
  - WS-Low-cost-LoRa-IoT-step-by-step.pdf
  - WS-Low-cost-LoRa-IoT-outdoor-step-by-step.pdf
  - WS-Low-cost-LoRa-GW-step-by-step.pdf
  - FAQ-from-WAZIUP.pdf
- This document specifically focuses on deployment issues for WaterSense while the above mentioned documents provide more general and broader information on the long-range hardware platform.

Sensing Devices





Sensing device Important issues (1)

- Don't forget that you should never transmit without an antenna!
- When a device with the soil moisture program has been flashed and each time you switch it on, it is going to transmit, so don't forget the antenna in any case!
- Put a tag to identify which position of the switch is ON and which position is OFF.
- Put a name tag on the case to remember the device's address, see next slide.

Sensing device
Arduino_LoRa_Simple_SoilHum

- For each sensor node that you install at a farm, you have to change the device's address, starting at 2 for instance. Address 1 is reserved for the gateway.

```
//////////////////////////////////////////////////////////////////
// CHANGE HERE THE LORA MODE, NODE ADDRESS
#define LORAMODE  1
// you need to change the node address for each sensor in the same organization/farm
// node address starts at 2 and ends at 255
#define node_addr 2
//////////////////////////////////////////////////////////////////
```

- If needed, change the frequency for measure and transmission, in minutes.

```
//////////////////////////////////////////////////////////////////
// CHANGE HERE THE TIME IN MINUTES BETWEEN 2 READING & TRANSMISSION
unsigned int idlePeriodInMin = 60;
//////////////////////////////////////////////////////////////////
```

Sensing device Deployment (1)

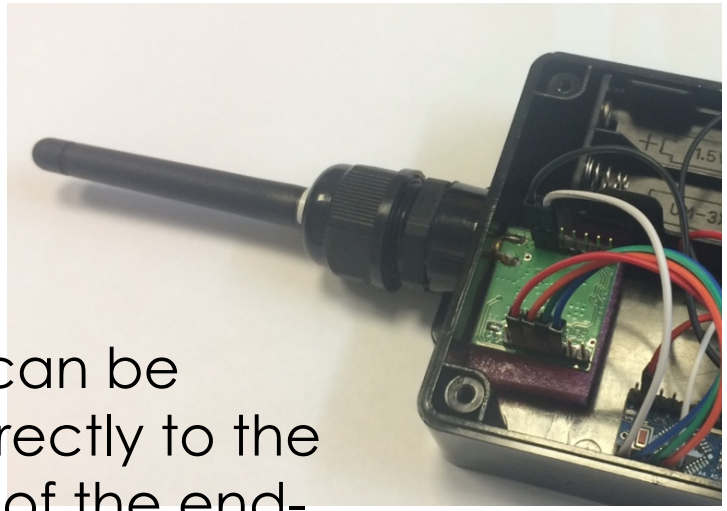
- Strongly tighten all cable glands, especially the one of the antenna cable so that the cable cannot turn and get disconnected from the radio module!
- Remember to put a shade cover to protect from direct sun!

Sensing device Deployment (2)

- When inserting the soil moisture tube, don't forget to put some soft soil so that the soil moisture sensor's tips can softly be inserted.

Connecting an antenna to the sensing device

- Taking the Modtronix inAir4 (433MHz), the antenna that you can buy with the module is usually a simple ¼ wave whip/monopole antenna (SMA-male)



- The antenna can be connected directly to the radio module of the end-device, using a larger cable gland to connect the antenna through the cable gland.

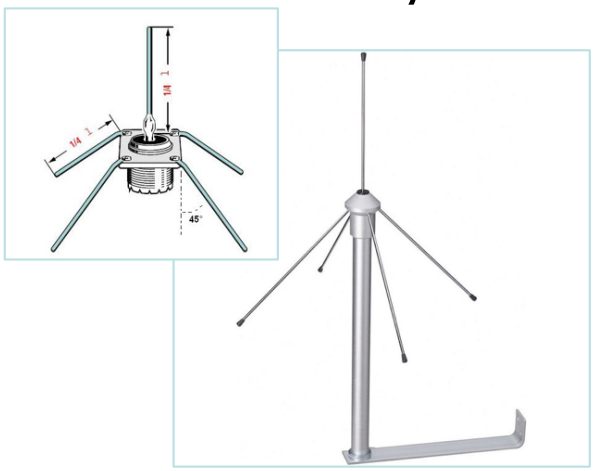
Use a coaxial antenna cable

- However, when the antenna is connected directly to the radio module, placing the device may be difficult as the antenna should be placed at a high location such as on top of a mast.

- Using an extension coaxial cable between the antenna and the radio module greatly ease the deployment of device **but**:
  - The antenna cable should not be too long to avoid high attenuation: 2m-3m
  - A ¼ wave monopole antenna will not provide good performance

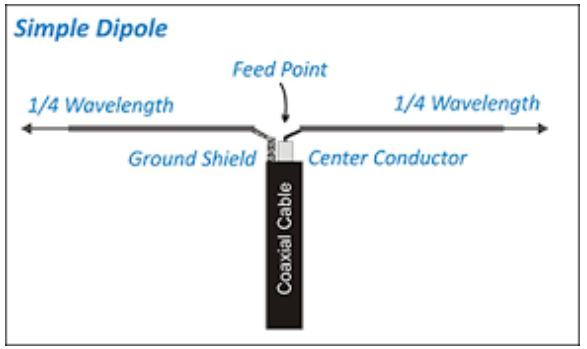
Antenna for device with a coaxial cable

- At the end of a coaxial cable, it is possible to connect a ground plane antenna (usually ¼ wave) or a ½ wave dipole antenna.



Ground plane

Sleeve dipole



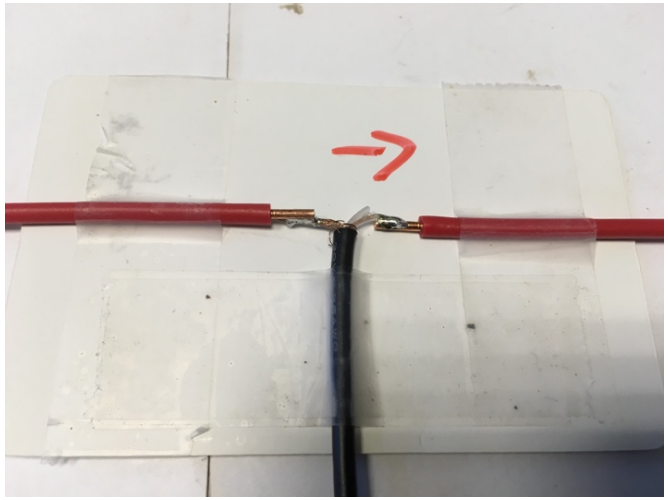
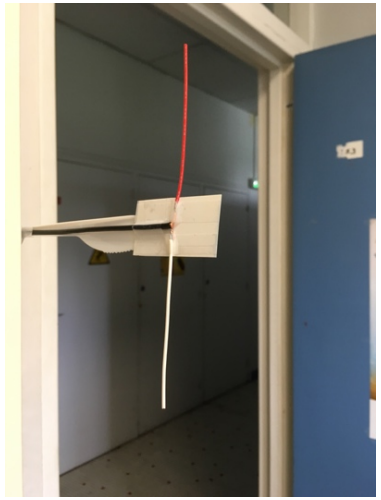
Simple dipole

More complex: collinear, array,…

- Some of them are easy to build (ground plane and simple dipole) and there are many tutorials.

Simple ½ wave dipole antenna

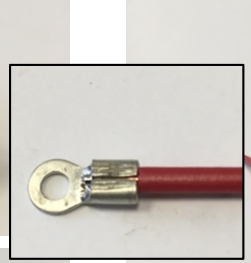
- The very simple dipole can be made with 2 pieces of ¼ wave wires. ¼ wave in 433MHz is about 16.4cm.

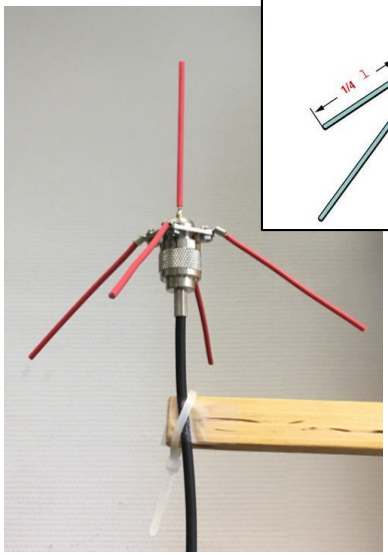



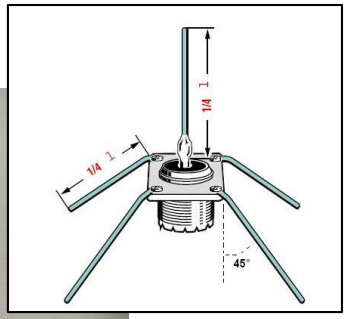

- There is no balun here but it is still better than the ¼ wave monople if a coaxial cable is used.
- You can buy a 3m **RG58** cable (SMA-m to SMA-f for instance), keep the male side, cut the female side and solder the core conductor and the braid as shown.

Simple ¼ wave ground plane antenna

- The ground plane antenna can be made with 5 pieces of ¼ wave wires. ¼ wave in 433MHz is about 16.4cm.





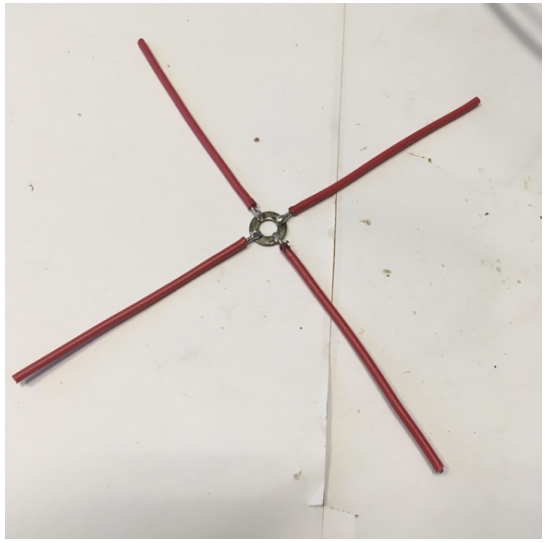
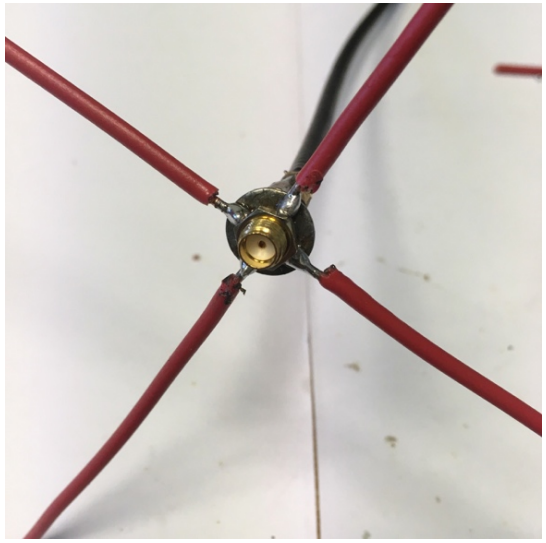
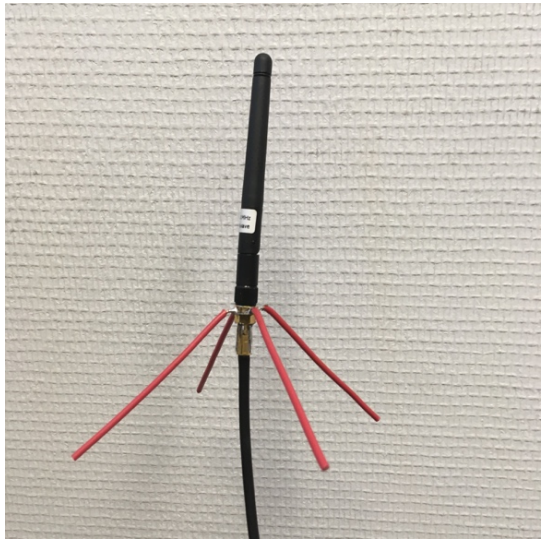


N Female Panel Mount Chassis

- You can buy a 3m RG58 cable with an SMA-male at one end and a male N-connector at the other end. Or build your own cable.

Even simpler ¼ wave ground plane antenna

- With an existing SMA-m/SMA-f cable, you can also build a ground plane antenna by adding 4 radiant wires to the ¼ wave monopole.

- This is a cheaper solution for sensing devices.

Some cable links

2m RG58 N male to SMA male

https://www.aliexpress.com/item-img/RG58-2m-N-Male-Jack-to-SMA-Male-M-M-RF-Coax-Pigtail-WLAN-Adapter-Adaptor/32616929641.html#

https://www.aliexpress.com/item-img/SMA-M-le-SMA-Femelle-Connecteur-Extension-C-ble-RG58-2-M/32543987605.html

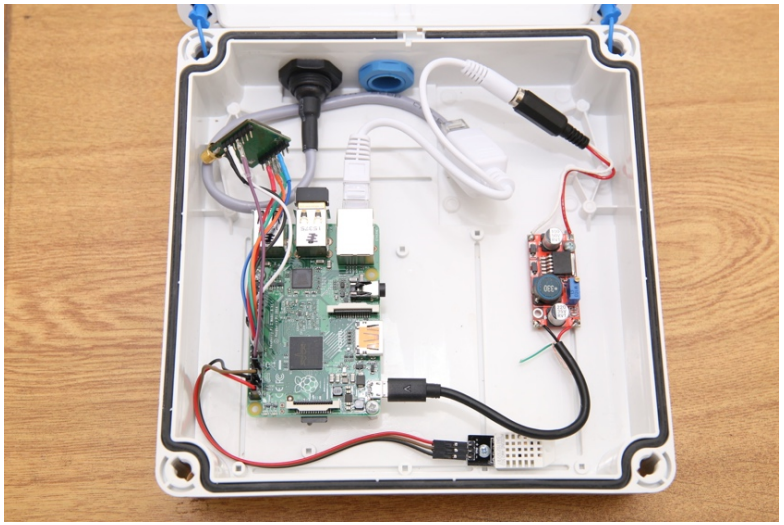
2m RG58 SMA male to SMA female

Summary of antennas for sensing device

- The easiest solution would be to buy a sleeve dipole for the frequency range you are operating.
- If you want to try the DIY approach, try first the simple dipole and then see if the range is acceptable.
- A ground plane antenna can be purchased or also made. You can test both solutions.
- Remember that RF transmissions depend a lot on the antenna location, the environment and many other factors!

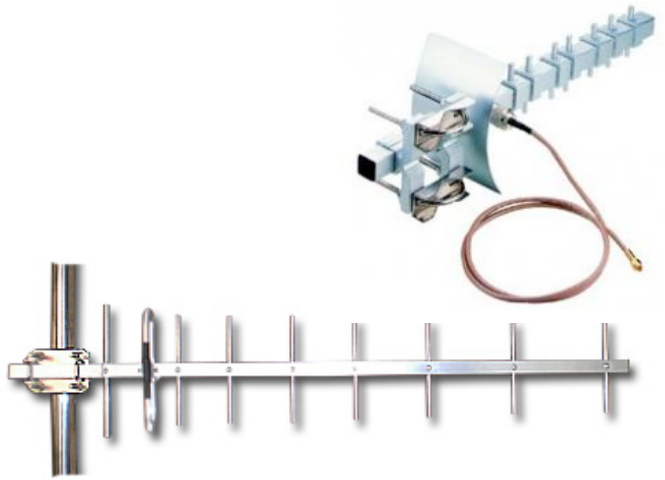
Installing the gateway

- Power the gateway either with PoE or directly with a 5V USB adaptor.
- If possible, it is much better to put the gateway indoor.
- Try avoiding long antenna cable: 2m to 5m.
- If the gateway needs to be put outdoor because of the antenna cable constraints, don't forget to protect it from direct sun!
- Get Internet access by connecting the Ethernet cable to a DSL or 4G router that will assign an IP address with DHCP.



Antennas for gateway

- Antennas for gateways can be placed on a building, at a high location.
- As for end-devices, you can easily use ground plane or sleeve dipole antenna. More complex high gain antenna or a directional Yagi antenna can be purchased depending on your budget and whether the device deployment allows it.



Gateway software installation

- An SD card image with a Raspberry Raspbian Jessie version is provided.
- You will need an 8GB SD card. Be careful, some SD cards will not work. This one has been successfully tested. It has to be class 10.

- Look at https://www.raspberrypi.org/documentation/installation/installing-images/ to see the procedure depending on your OS. 7948206080 bytes should be written, otherwise you may have a problem.
- Once flashed, insert the SD card and power-up the Raspberry-based gateway.

Gateway's simple command interface

- Once connected to the gateway's WiFi, use a terminal window to ssh on the gateway
  - `ssh pi@192.168.200.1`
  - Password is `loragateway`
- You should see the simple text interface
- Select Q and RETURN to quit this interface
- You should be in the `lora_gateway` folder

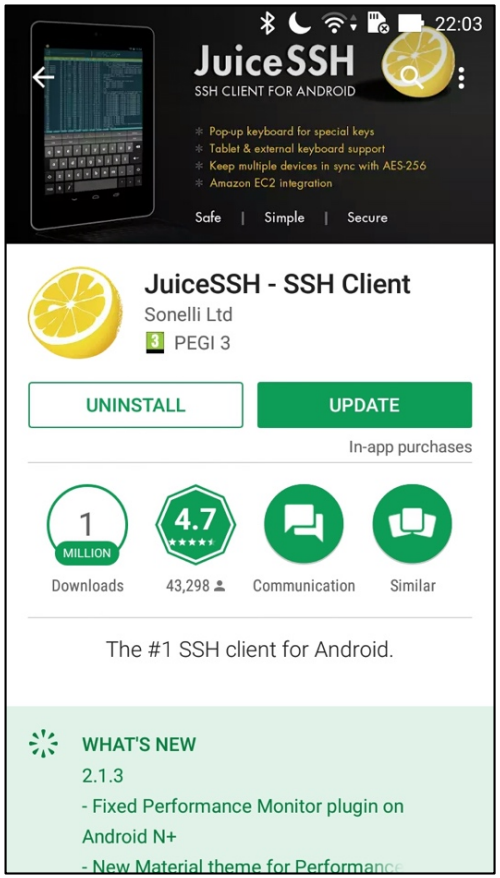
You can use an Android smartphone or tablet to connect to the gateway with the JuiceSSH app! See in next slide.

```
=======================================* Gateway 00000027EB5A71F7 *===
0- sudo python start_gw.py                                           +
1- sudo ./lora_gateway --mode 1                                      +
2- sudo ./lora_gateway --mode 1 | python post_processing_gw.py       +
3- ps aux | grep -e start_gw -e lora_gateway -e post_proc -e log_gw  +
4- tail --line=25 ../Dropbox/LoRa-test/post-processing.log           +
5- tail --line=25 -f ../Dropbox/LoRa-test/post-processing.log        +
6- less ../Dropbox/LoRa-test/post-processing.log                     +
-------------------------------------------------------* Bluetooth *--+
a- run: sudo hciconfig hci0 piscan                                   +
b- run: sudo python rfcomm-server.py                                 +
c- run: nohup sudo python rfcomm-server.py -bg > rfcomm.log &        +
d- run: ps aux | grep rfcomm                                         +
e- run: tail -f rfcomm.log                                           +
-------------------------------------------------------* Connectivity *--+
f- test: ping www.univ-pau.fr                                        +
-------------------------------------------------------* Filtering msg *--+
l- List LoRa reception indications                                   +
m- List radio module reset indications                               +
n- List boot indications                                             +
o- List post-processing status                                       +
p- List low-level gateway status                                     +
-------------------------------------------------------* Configuration *--+
A- show gateway_conf.json                                            +
B- edit gateway_conf.json                                            +
C- show clouds.json                                                  +
D- edit clouds.json                                                  +
-------------------------------------------------------* Update *--+
U- update to latest version on repository                            +
V- download and install a file                                       +
W- run a command                                                     +
-------------------------------------------------------* kill *--+
K- kill all gateway related processes                                +
k- kill rfcomm-server process                                        +
R- reboot gateway                                                    +
S- shutdown gateway                                                  +
---------------------------------------------------------------------+
Q- quit                                                              +
=====================================================================
Enter your choice:
```

```
pi@raspberrypi:~/lora_gateway $
```

Using Android smartphone or tablet

- Use an Android smartphone or tablet with JuiceSSH to connect to the gateway



Gateway update (1)

- The gateway must be updated to the latest version.
- It is better to do the update procedure at the university office so that the gateway can have easy internet access.
- The update procedure can easily be done by connecting to the gateway and run the `update_gw.sh` script.
- To connect to the gateway, connect to the gateway's WiFi WS_PI_GW_XXXXXXXX WiFi
- Password is `loragateway`

GATEWAY UPDATE (2)

- Then type the following commands
  - `cd`
  - `cp lora_gateway/scripts/update_gw.sh .`
  - `rm -rf lora_gateway`
  - `./update_gw.sh`
- Removing any previous `lora_gateway` folder triggers a full update.
- The gateway will obtain the latest distribution from our github repository and will create a new `lora_gateway` folder.
- Next periodic updates without deleting the existing `lora_gateway` folder, i.e. preserving existing configuration files, will be presented later on.

GATEWAY UPDATE (3)

- It is also possible to get the latest version of this script
  - `cd`
  - `wget https://raw.githubusercontent.com/CongducPham/WaterSense/master/WaterSenseGateway/scripts/update_gw.sh`
- Then type the following commands
  - `rm -rf lora_gateway`
  - `./update_gw.sh`
- Again, removing any previous `lora_gateway` folder triggers a full update

Configuring the gateway (1)

- Go into the `scripts` folder in the newly created `lora_gateway` folder
- Run the `basic_config_gw.sh` script
  - `./basic_config_gw.sh`
- The script will get the hardware address of the gateway to define the gateway'id, using the last 5 bytes of the MAC address
  - `ifconfig`

```
pi@raspberrypi:~ $ ifconfig
eth0      Link encap:Ethernet  HWaddr b8:27:eb:79:5c:47
          inet addr:10.0.13.185  Bcast:10.0.13.255  Mask:255.255.255.0
          inet6 addr: fe80::ba27:ebff:fe79:5c47/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:3500 errors:0 dropped:0 overruns:0 frame:0
```

- It will also compile the low-level gateway software, remember to check compilation options of `radio.makefile`
- If you need advanced configuration, use `config_gw` and follow the instructions as shown in the next slides **otherwise you are all done**

Advanced configuration only (1)

```
*******************************************
*** compile lora_gateway executable Y/N  ***
*******************************************
```

Enter Y

```
*********************************************************
*** create log symb link to ~/Dropbox/LoRa-test Y/N  ***
*********************************************************
```

Enter Y

```
***********************************
*** configure hostapd.conf Y/N  ***
***********************************
```

Enter Y

```
*************************************************************
*** configure a newly installed hostapd/dnsmasq package Y/N  ***
*************************************************************
```

Enter N

```
*******************************************
*** configure bluetooth network name Y/N  ***
*******************************************
```

Enter N

```
***********************************
*** install DHT22 support Y/N  ***
***********************************
```

Enter Y

Advanced configuration only (2)

```
****************************************
*** edit gateway_conf.json now? Y/N ***
****************************************
```

Enter N

```
************************************
*** activate DHT22 MongoDB Y/N/Q  ***
************************************
```

Enter Q

```
************************************************************
*** edit LoRa data MongoDB local storage option? Y/N  ***
************************************************************
```

Enter N

```
******************************
*** run gateway at boot Y/N ***
******************************
```

Enter Y

```
**********************************************
*** check configuration (recommended) Y/N  ***
**********************************************
```

Enter N

```
********************
*** reboot Y/N  ***
********************
```

Enter N

Start the command interface

```
> ./cmd.sh
```

As you can see, the gateway id shown by the command interface is now correct

```
pi@raspberrypi:~/lora_gateway $ ./cmd.sh
=======================================* Gateway 00000027EB5A71F7 *===
0- sudo python start_gw.py                                            +
1- sudo ./lora_gateway --mode 1                                       +
2- sudo ./lora_gateway --mode 1 | python post_processing_gw.py        +
3- ps aux | grep -e start_gw -e lora_gateway -e post_proc -e log_gw   +
4- tail --line=25 ../Dropbox/LoRa-test/post-processing.log            +
5- tail --line=25 -f ../Dropbox/LoRa-test/post-processing.log         +
6- less ../Dropbox/LoRa-test/post-processing.log                      +
-------------------------------------------------* Bluetooth *--+
a- run: sudo hciconfig hci0 piscan                                    +
b- run: sudo python rfcomm-server.py                                  +
c- run: nohup sudo python rfcomm-server.py -bg > rfcomm.log &         +
d- run: ps aux | grep rfcomm                                          +
e- run: tail -f rfcomm.log                                            +
-------------------------------------------------* Connectivity *--+
f- test: ping www.univ-pau.fr                                         +
-------------------------------------------------* Filtering msg *--+
l- List LoRa reception indications                                    +
m- List radio module reset indications                                +
n- List boot indications                                              +
o- List post-processing status                                        +
p- List low-level gateway status                                      +
-------------------------------------------------* Configuration *--+
A- show gateway_conf.json                                             +
B- edit gateway_conf.json                                             +
C- show clouds.json                                                   +
D- edit clouds.json                                                   +
-----------------------------------------------------* Update *--+
U- update to latest version on repository                             +
V- download and install a file                                        +
W- run a command                                                      +
-----------------------------------------------------* kill *--+
K- kill all gateway related processes                                 +
k- kill rfcomm-server process                                         +
R- reboot gateway                                                     +
S- shutdown gateway                                                   +
-----------------------------------------------------------------------+
Q- quit                                                               +
=======================================================================
Enter your choice:
```

Configuring data management

- For WaterSense, received data from devices will be uploaded to the WAZIUP Orion data platform. Therefore `clouds.json` file should be set as follows:

```
{
    "name":"WAZIUP Orion cloud",
    "script":"python CloudOrion.py",
    "type":"iotcloud",
    "write_key":"",
    "enabled":true
},
```

- Modify `clouds.json` accordingly
- `CloudOrion.py` script will use information from `key_Orion.py` to configure data management for each organization
- Therefore you need to configure this file for each farm

KEY_ORION.PY

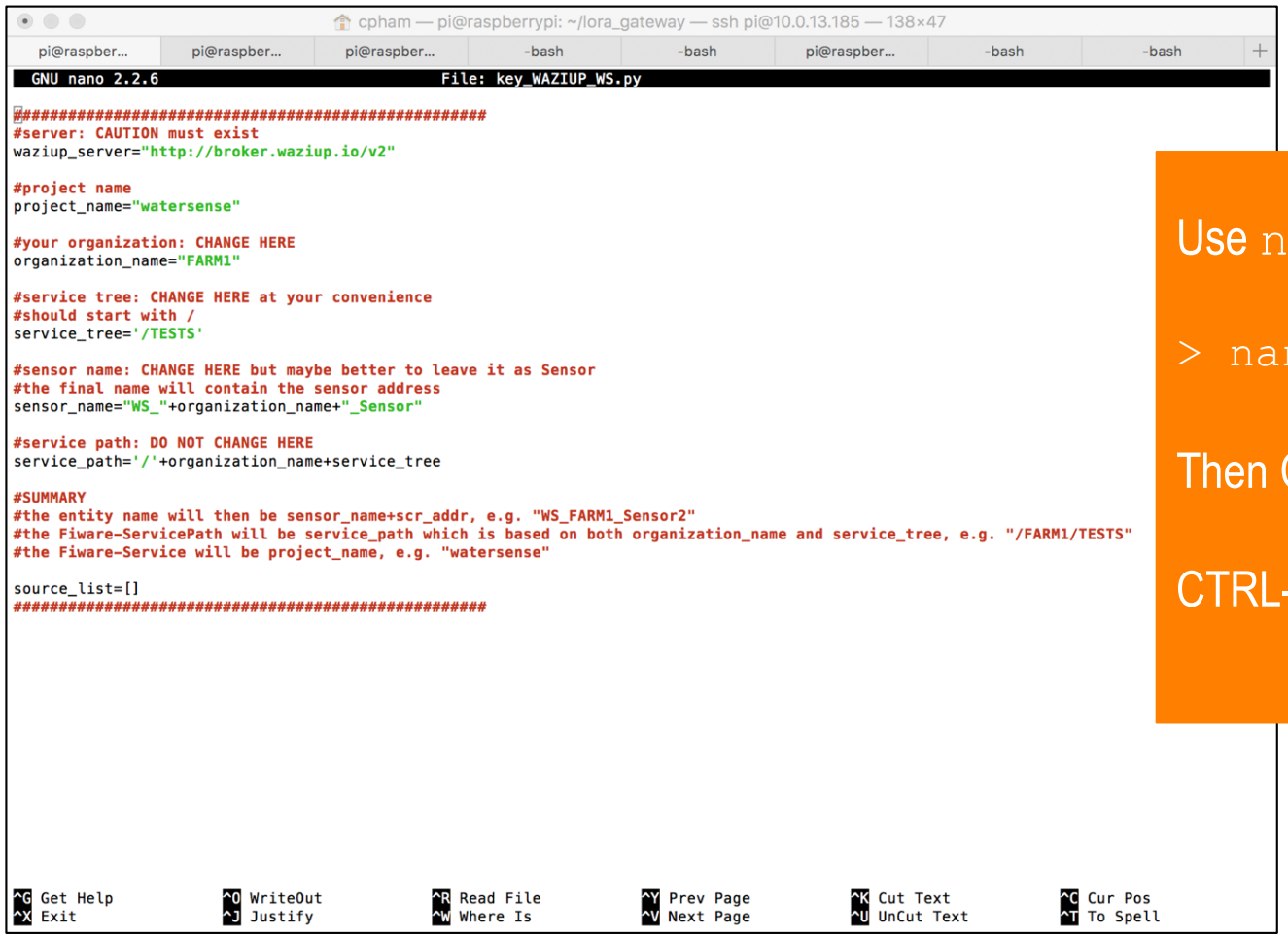
```
#######################################################
#server: CAUTION must exist
orion_server="http://broker.waziup.io/v2"

#project name
project_name="watersense"

#your organization: CHANGE HERE
organization_name="FARM1"

#service tree: CHANGE HERE at your convenience
#should start with /
service_tree='/TESTS'

#sensor name: CHANGE HERE but maybe better to leave it as Sensor
#the final name will contain the sensor address
sensor_name="WS_"+organization_name+"_Sensor"

#service path: DO NOT CHANGE HERE
service_path='/'+organization_name+service_tree

#SUMMARY
#the entity name will then be sensor_name+scr_addr, e.g. "WS_FARM1_Sensor2"
#the Fiware-ServicePath will be service_path which is based on both
organization_name and service_tree, e.g. "/FARM1/TESTS"
#the Fiware-Service will be project_name, e.g. "watersense"

source_list=[]
#######################################################
```

We need to change the organization_name for each farm. Use FARM1, FARM2, FARM3, etc.

We can define new names latter on when the software platform is stable.

FARM1 has been used for the first deployment

Editing key_Orion.py



Use `nano` to edit the file:

```
> nano key_Orion.py
```

Then CTRL-O + RETURN to save

CTRL-X to quit

Rebooting the gateway

- Your gateway is now updated and configured
- You can now reboot the gateway. To do so, just type:
  - `sudo shutdown -r now`
- Or run `./cmd.sh` and choose option **R.**
- Once the gateway has rebooted, check the WiFi SSID which now should meet your gateway's id.
- Try to avoid unplugging power cable to shutdown your gateway. Log into the gateway and select option **S** instead.
- Your gateway is now ready to be deployed.

Testing the connectivity between devices and gateway

- When deploying the gateway and the devices, the first step is to check connectivity and adjust the gateway/antenna location.
- It is recommended to use an device programmed to send a message every 1 minute for instance.
- Place the device at the planned location in the field, with the mast, as for a definitive setting.
- Then at the gateway side, connect to the gateway WiFi, login on the gateway and use option ***5*** to follow the log file and see whether messages are received.

Testing the connectivity view the log file

```
pi@raspberrypi:~/lora_gateway $ ./cmd.sh
=======================================* Gateway 00000027EB5A71F7 *===
0- sudo python start_gw.py                                            +
1- sudo ./lora_gateway --mode 1                                       +
2- sudo ./lora_gateway --mode 1 | python post_processing_gw.py        +
3- ps aux | grep -e start_gw -e lora_gateway -e post_proc -e log_gw   +
4- tail --line=25 ../Dropbox/LoRa-test/post-processing.log            +
5- tail --line=25 -f ../Dropbox/LoRa-test/post-processing.log         +
6- less ../Dropbox/LoRa-test/post-processing.log                      +
-------------------------------------------------------* Bluetooth *--+
a- run: sudo hciconfig hci0 piscan                                    +
b- run: sudo python rfcomm-server.py                                  +
c- run: nohup sudo python rfcomm-server.py -bg > rfcomm.log &         +
d- run: ps aux | grep rfcomm                                          +
e- run: tail -f rfcomm.log                                            +
-------------------------------------------------* Connectivity *--+
f- test: ping www.univ-pau.fr                                         +
-------------------------------------------------* Filtering msg *--+
l- List LoRa reception indications                                    +
m- List radio module reset indications                                +
n- List boot indications                                              +
o- List post-processing status                                        +
p- List low-level gateway status                                      +
-------------
A- show gatewa
B- edit gatewa
C- show clouds
D- edit clouds
-------------
U- update to
V- download ar
W- run a comma
-------------
K- kill all ga
k- kill rfcomm
R- reboot gate
S- shutdown ga
-------------
Q- quit
==============
Enter your cho
```

```
2017-05-09T21:20:55.169793> --- rxlora. dst=1 type=0x12 src=2 seq=2 len=22 SNR=7 RSSIpkt=-52 BW=125 CR=4/5 SF=12
2017-05-09T21:20:55.170961> 2017-05-09T21:20:55.169591
2017-05-09T21:20:55.171848> rcv ctrl pkt info (^p): 1,18,2,2,22,7,-52
2017-05-09T21:20:55.172549> splitted in:  [1, 18, 2, 2, 22, 7, -52]
2017-05-09T21:20:55.173175> (dst=1 type=0x12(DATA WAPPKEY) src=2 seq=2 len=22 SNR=7 RSSI=-52)
2017-05-09T21:20:55.173853> rcv ctrl radio info (^r): 125,5,12
2017-05-09T21:20:55.174482> splitted in:  [125, 5, 12]
2017-05-09T21:20:55.175149> (BW=125 CR=5 SF=12)
2017-05-09T21:20:55.175820> rcv timestamp (^t): 2017-05-09T21:20:55.168
2017-05-09T21:20:55.176430>
2017-05-09T21:20:55.176983> got first framing byte
2017-05-09T21:20:55.177580> --> got data prefix
2017-05-09T21:20:55.178239> --> DATA with_appkey: read app key sequence
2017-05-09T21:20:55.178967> app key is  0x05 0x06 0x07 0x08
2017-05-09T21:20:55.179583> in app key list
2017-05-09T21:20:55.180277> /!SM1/567/SM2/678
```

You don't even need Internet access for the gateway to perform the connectivity tests.

You can have more freedom by powering the gateway with a battery pack.

Periodic update procedure

- The latest command interface has built-in update procedure
- You can use option **U** to update from repository and still keep all your configuration files: `gateway_conf.json`, `clouds.json` and `key*`
- You can also install a single file with option **V** that will prompt for a URL
- You can enter a URL that has been provided by some administrator
- Example in the next slide

```
pi@raspberrypi:~/lora_gateway $ ./cmd.sh
=====================================* Gateway 00000027EB5A71F7 *===
0- sudo python start_gw.py                                          +
1- sudo ./lora_gateway --mode 1                                     +
2- sudo ./lora_gateway --mode 1 | python post_processing_gw.py      +
3- ps aux | grep -e start_gw -e lora_gateway -e post_proc -e log_gw +
4- tail --line=25 ../Dropbox/LoRa-test/post-processing.log          +
5- tail --line=25 -f ../Dropbox/LoRa-test/post-processing.log       +
6- less ../Dropbox/LoRa-test/post-processing.log                    +
-----------------------------------------------------* Bluetooth *--+
a- run: sudo hciconfig hci0 piscan                                  +
b- run: sudo python rfcomm-server.py                                +
c- run: nohup sudo python rfcomm-server.py -bg > rfcomm.log &       +
d- run: ps aux | grep rfcomm                                        +
e- run: tail -f rfcomm.log                                          +
-----------------------------------------------------* Connectivity *--+
f- test: ping www.univ-pau.fr                                       +
-----------------------------------------------------* Filtering msg *--+
l- List LoRa reception indications                                  +
m- List radio module reset indications                              +
n- List boot indications                                            +
o- List post-processing status                                      +
p- List low-level gateway status                                    +
-----------------------------------------------------* Configuration *--+
A- show gateway_conf.json                                           +
B- edit gateway_conf.json                                           +
C- show clouds.json                                                 +
D- edit clouds.json                                                 +
-----------------------------------------------------------* Update *--+
U- update to latest version on repository                           +
V- download and install a file                                      +
W- run a command                                                    +
-----------------------------------------------------------* kill *--+
K- kill all gateway related processes                               +
k- kill rfcomm-server process                                       +
R- reboot gateway                                                   +
S- shutdown gateway                                                 +
---------------------------------------------------------------------+
Q- quit                                                             +
=====================================================================
Enter your choice:
```

Download and install a file (1)

- With option **V**, you can enter an URL that points to a file. The file will be downloaded and installed in the `lora_gateway` folder.

```
Enter your choice:
V
---------------------------------------------------------------------
BEGIN OUTPUT
Download and install a file
Enter the URL of the file:
https://www.dropbox.com/s/mcmg4yeksr340c2/example-install-file.txt
```

```
Download and install a file
Enter the URL of the file:
https://www.dropbox.com/s/mcmg4yeksr340c2/example-install-file.txt
--2017-05-09 22:16:53--  https://www.dropbox.com/s/mcmg4yeksr340c2/example-install-file.txt
Resolving www.dropbox.com (www.dropbox.com)... 162.125.65.1
Connecting to www.dropbox.com (www.dropbox.com)|162.125.65.1|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://dl.dropboxusercontent.com/content_link/Veb5Tx1XY65zpGTJ9ZUYQAuAwhDY9GiEmw9HUxcQXuMh62IneXy7BUp1EF450L0l/file [following]
--2017-05-09 22:16:54--  https://dl.dropboxusercontent.com/content_link/Veb5Tx1XY65zpGTJ9ZUYQAuAwhDY9GiEmw9HUxcQXuMh62IneXy7BUp1EF450L0l/file
Resolving dl.dropboxusercontent.com (dl.dropboxusercontent.com)... 162.125.65.6
Connecting to dl.dropboxusercontent.com (dl.dropboxusercontent.com)|162.125.65.6|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 167 [text/plain]
Saving to: 'example-install-file.txt'

example-install-file.txt            100%[=======================================================================>]     167  --.-KB/s   in 0s

2017-05-09 22:16:55 (17.2 MB/s) - 'example-install-file.txt' saved [167/167]

Done
END OUTPUT
Press RETURN/ENTER...
```

Download and install a file (2)

- This feature is very usefull for end-users to simply update some files on the gateway.
- An administrator can write appropriate configuration files for the end-user and generate an URL to this file (with Dropbox for instance).
- The URL can be either be sent by mail or SMS to the end-user.
- The end-user has to simply log into the gateway (using an Android smartphone or tablet connecting to the gateway's WiFi) and select option V to enter the URL.
- The end-user will then just reboot the gateway with option **R** for the new configuration to run.

Download and install a file (3)

- System files can also be installed with option **W** that will prompt for a command

```
Enter your choice:
W
---------------------------------------------------------------
BEGIN OUTPUT
Run a command
Enter the command to run:
sudo wget -O /etc/test.txt https://www.dropbox.com/s/mcmg4yeksr340c2/example-install-file.txt
```

- Here, the previous example file will be installed in `/etc` under the name `test.txt`
- Like previously, the exact command can be sent to the end-user

Securing with Application Key (1)

- End-device can use application key (app key) on 4 bytes to allow filtering mechanisms at the gateway side.
- The app key is defined in the end-device sketch (`Arduino_LoRa_Simple_SoilHum`) and the feature is activated by uncommenting `#define WITH_APPKEY`

```
#ifdef WITH_APPKEY
////////////////////////////////////////////////////////////////
// CHANGE HERE THE APPKEY, BUT IF GW CHECKS FOR APPKEY, MUST BE
// IN THE APPKEY LIST MAINTAINED BY GW.
uint8_t my_appKey[4]={5, 6, 7, 8};
////////////////////////////////////////////////////////////////
#endif
```

- At the gateway side, post_processing_gw.py has a list of allowed app key

```
app_key_list = [
          #change/add here your application keys
          '\x01\x02\x03\x04',
          '\x05\x06\x07\x08' ]
```

Securing with Application Key (2)

- With app key enforcement at gateway, all LoRa data to be uploaded on clouds will need a valid app key, otherwise the data will be discarded as shown below:

```
--- rxlora. dst=1 type=0x12 src=6 seq=136 len=17 SNR=9 RSSIpkt=-56
rcv ctrl pkt info (^p): 1, 18, 6,136,17,9,-56
splitted in:  [1, 18, 6, 136, 17, 9, -56]
(dst=1 type=0x12 src=6 seq=136 len=17 SNR=9 RSSI=-56)
got first framing byte
--> got app key sequence
app key is: [9, 10, 11, 12]
not in app key list
invalid app key: discard data
```

- This is configured in the `gateway_conf.json` file. Set to true

```
    "freq": 433.3
  },
  "gateway_conf": {
    "gateway_ID": "000000XXXXXXXXXX",
    "ref_latitude": "my_lat",
    "ref_longitude": "my_long",
    "wappkey": false,
    "raw": false,
    "aes": false,
    "log_post_processing": true,
```

How to use app key

- App key can be used to differenciate data from one organization to another
  - Sensing devices of a given organization will use the same app key
  - The gateway is configured to only accept this app key
- App key can be used to distribute the gateway task in case several gateways in the same organization are deployed
  - Sensing devices will be caterorized with 2 app key
  - Each gateway will allow only one of these 2 app key
  - In this way, data that can be received by 2 gateways will be processed by only 1 gateway

Securing by encryption (1)

- `Arduino_LoRa_SoilHum` is an extended version of `Arduino_LoRa_Simple_SoilHum` with data encryption feature.
- Data will be encrypted using 128-bit AES algorithm following the LoRaWAN encryption method.
- Uncomment `#define WITH_AES`

```
//////////////////////////////////////////////////////////////////
// COMMENT OR UNCOMMENT TO CHANGE FEATURES.
// ONLY IF YOU KNOW WHAT YOU ARE DOING!!! OTHERWISE LEAVE AS IT IS
#if not defined _VARIANT_ARDUINO_DUE_X_ && not defined __SAMD21G18A__
#define WITH_EEPROM
#endif
#define WITH_APPKEY
#define LOW_POWER
#define LOW_POWER_HIBERNATE
#define WITH_AES
```

Securing by encryption (2)

- Encryption ensures confidentiality. The two 16-byte encryption keys are defined in the end-device sketch (`Arduino_LoRa_SoilHum`)

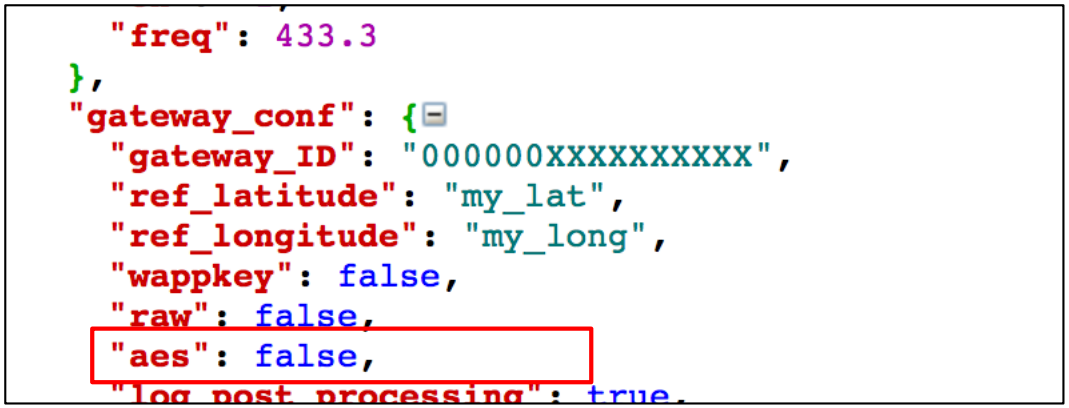
```
unsigned char AppSkey[16] = {
  0x2B, 0x7E, 0x15, 0x16, 0x28, 0xAE, 0xD2, 0xA6,
  0xAB, 0xF7, 0x15, 0x88, 0x09, 0xCF, 0x4F, 0x3C
};

unsigned char NwkSkey[16] = {
  0x2B, 0x7E, 0x15, 0x16, 0x28, 0xAE, 0xD2, 0xA6,
  0xAB, 0xF7, 0x15, 0x88, 0x09, 0xCF, 0x4F, 0x3C
};
```

- And should also be declared in the `loraWAN.py` script on the gateway

```
AppSKey = '2B7E151628AED2A6ABF7158809CF4F3C'
NwkSKey = '2B7E151628AED2A6ABF7158809CF4F3C'
```

Securing by encryption (3)

- With encryption at device and decryption at gateway, there is more robust integrity check of the messages.
- Note that app key can still be used with AES, even if different gateways may have different encryption keys.
- To enable decryption at gateway, AES feature should be activated (set to true) in the `gateway_conf.json` file.



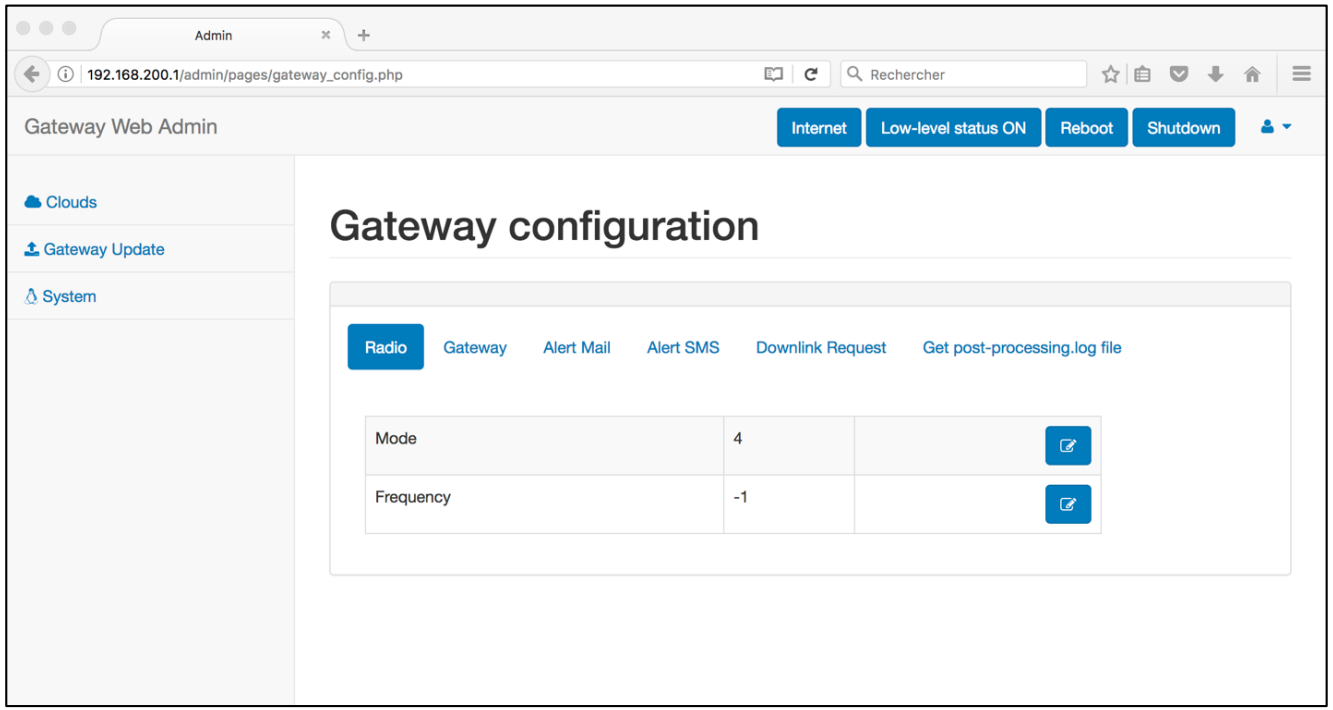
- Otherwise, the gateway will not be able to decrypt and therefore will not be able to push meaningfull data to clouds

Gateway web admin interface (1)

- A gateway web admin interface has been added to the latest version
- To install the web admin interface, check if you have the `gw_web_admin` folder in your `lora_gateway` folder
- If you don't, then update to the latest version
- Then, go into `gw_web_admin` and run the `install.sh` script
  - `cd gw_web_admin`
  - `sudo ./install.sh`
- Refer to the web admin interface tutorial for more information

Gateway web admin interface (2)

- http://192.168.200.1/admin
  - Login: admin
  - Password: loragateway



Web admin features

- Currently, you can use the web admin to:
  - Easily reboot and shutdown your gateway
    - Be carefull, if you shut down the gateway, you need to physically access the gateway to power it it on again
  - Change LoRa mode and frequency
  - Update your gateway
  - Change the WiFi SSID and password
  - Enable/Disable local AES decryption
  - Enable/Disable ThingSpeak and WAZIUP Orion cloud
  - For ThingSpeak, you can specify a new write key
  - For WAZIUP Orion, you can specify the project name, the organization name and the service tree
    - Fiware-service=project_name
    - sensor_name="WS_"+organization_name+"_Sensor"
    - Fiware-servicePath='/'+organization_name+service_tree
    - See slide 28

Further readings

- The WaterSense github resources
  - https://github.com/CongducPham/WaterSense
- A web page explaining our low-cost gateway
  - http://cpham.perso.univ-pau.fr/LORA/RPIgateway.html
- Specific README files on the github, especially those on cloud management and encryption
  - https://github.com/CongducPham/WaterSense/blob/master/WaterSenseGateway/README-NewCloud.md
  - https://github.com/CongducPham/WaterSense/blob/master/WaterSenseGateway/README-aes_lorawan.md