

Gateway output

```
{  
  "gw_id": "eui-b827ebfffea85a2b",  
  "payload": "AMecANB+1bNwpHGplUnVs3A3GoU3OWo=",  
  "dev_eui": "70B3D54995A971A4",  
  "lora": {  
    "spreading_factor": 7,  
    "bandwidth": 125,  
    "air_time": 61696000  
  },  
  "coding_rate": "4/5",  
  "timestamp": "2018-02-02T14:21:30.488Z",  
  "rssi": -89,  
  "snr": 7.5,  
  "app_eui": "70B3D57ED0009CC7",  
  "frequency": 868300000  
}
```

Atom upload

Uploading project (main folder)...

Reading file status

Upload done, resetting board...

No files to upload

ets Jun 8 2016 00:22:57

rst:0x7 (TG0WDT_SYS_RESET),boot:0x17 (SPI_FAST_FLASH_BOOT)

configsip: 0, SPIWP:0xee

clk_drv:0x00,q_drv:0x00,d_drv:0x00,cs0_drv:0x00,hd_drv:0x00,wp_drv:0x00

mode:DIO, clock div:1

load:0x3fff9028,len:8

load:0x3fff9030,len:1060

load:0x4009fa00,len:0

ho 12 tail 0 room 4

load:0x4009fa00,len:14084

entry 0x400a05d4

Not yet joined...

Not yet joined...

Not yet joined...

Not yet joined...

Not yet joined...

Not yet joined...

Not yet joined...

Not yet joined...

Not yet joined...

Not yet joined...

Not yet joined...

Not yet joined...

Not yet joined...

Not yet joined...

Lopy4 code

```
from network import LoRa
```

```
import socket
```

```
import time
```

```
import binascii
```

```
# Initialize LoRa in LORAWAN mode.
```

```
lora = LoRa(mode=LoRa.LORAWAN)
```

```
# create an OTAA authentication parameters
```

```
app_eui = binascii.unhexlify('70B3D57ED0009CC7')
```

```
app_key = binascii.unhexlify('BFC390775421DF12D5C12ED976EDB910')
```

```
# join a network using OTAA (Over the Air Activation)
```

```
lora.join(activation=LoRa.OTAA, auth=(app_eui, app_key), timeout=0)
```

```
# wait until the module has joined the network
```

```
while not lora.has_joined():
```

```
    time.sleep(2.5)
```

```
    print('Not yet joined...')
```

```
# create a LoRa socket
```

```
s = socket.socket(socket.AF_LORA, socket.SOCK_RAW)
# set the LoRaWAN data rate
s.setsockopt(socket.SOL_LORA, socket.SO_DR, 5)
# make the socket blocking
# (waits for the data to be sent and for the 2 receive windows to expire)
s.setblocking(True)
# send some data
s.send(bytes([0x01, 0x02, 0x03]))
# make the socket non-blocking
# (because if there's no data received it will block forever...)
s.setblocking(False)
# get any data received (if any...)
data = s.recv(64)
print(data)
```