

Email Alert Instructions Written January 10, 2021

This instruction uses a python program, and PHP script is to send an email Alert on two digital inputs. It is used on EMONCMS base image Version: low-write 10.2.7. Image: emonSD-24Jul20.img. Used WinSCP for file manager from my Windows computer to the RPI, and Putty to SSH into the headless PI.

One Digital input monitors when a Leak is detected using LevelSense:Floor Water Sensor. The second Digital input monitor the Mechanical Room Access using: uxcell Magnetic Reed Switch NC/NO Door Alarm.

Used MQTT connection to read data from EMONCMS inputs. Using this connection does not require Emoncms to log data. Hindsight, using a JSON read, without logging, may use less resources on the PI.

Note: had to change security level on my gmail account to accept email.

1: create directory /home/pi/usefulscripts

2. install swiftmailer: "<https://community.openenergymonitor.org/t/if-watt-feed-goes-to-zero-send-email/13482/8>"

```
git -C /opt/emoncms/modules clone -b 'v5.4.8' --single-branch
https://github.com/swiftmailer/swiftmailer.git
```

3. Create PHP mailer script. Listed in section 9 of this document. I named it "swift.php". CHMOD 755

4. Create empty textfile.txt in /home/pi/usefulscripts

5. Install in-process scheduler.  
"<https://www.raspberrypi.org/forums/viewtopic.php?t=291059>"  
pip3 install schedule (pip3 is for python 3)

6. Create Python program to open MQTT connection. Listed in section 10 of this document. Read values, add date time stamp. Write to file. After 1 hour read file, search for a 1. Close file and delete contents. Again be sure to CHMOD 755.

7. Runs as service.  
/etc/systemd/system/alert.service  
[Unit]  
Description=Alert service  
After=mosquitto.service

```
[Service]
ExecStart=/usr/bin/python3 -u alert.py
WorkingDirectory=/home/pi/usefulscripts
StandardOutput=inherit
StandardError=inherit
Restart=always
User=pi
```

```
[Install]
WantedBy=multi-user.target
```

8. Start automatically on reboot by using this command: `sudo systemctl enable alert.service`

9. `swiftmail.php`

```
<?php
// git -C /opt/emoncms/modules clone -b 'v5.4.8' --single-branch
https://github.com/swiftmailer/swiftmailer.git
require_once '/opt/emoncms/modules/swiftmailer/lib/swift_required.php';
$transport = (new Swift_SmtpTransport('smtp.gmail.com', 465))
->setEncryption('ssl')
->setUsername('xxxxxxx@gmail.com')
->setPassword('xxxxxxx')
;

// Create the Mailer using your created Transport
$mailer = new Swift_Mailer($transport);

// Create a message
$message = (new Swift_Message('Leak or Entry Detected'))
->setFrom(['noreply@emoncms.org' => 'EmonCMS'])
->setTo(['xxxxxxx@gmail.com' => 'name'])
->setBody('Leak Detected or Mechanical Room Access')
->attach(Swift_Attachment::fromPath('/home/pi/usefulscripts/textfile.txt'));
;
// Send the message
$result = $mailer->send($message);
?>
```

10. `alert.py`

```
#Run: python3 alert.py
import time
import schedule
import os
from datetime import datetime
import paho.mqtt.client as mqttClient
import subprocess

def on_connect(client, userdata, flags, rc):
    if rc == 0:
        print("Connected to broker")
        global Connected
        Connected = True
    else:
        print("Connection failed")

Connected = False # for state value

broker_address = "127.0.0.1" # your broker address
port = 1883
user = 'emonpi'
password = 'default mqtt password'
client = mqttClient.Client("Python")
client.username_pw_set(user, password=password)
```

```

client.on_connect = on_connect
client.connect(broker_address, port = port)

logfile = r"/home/pi/usefulscripts/textfile.txt" # name of my log file
my_data = ['1', '2']

def write_log(text, file):
    f = open(file, 'a') # 'a' will append to an existing file if it exists
    for sub_list in text[1:]:
        value1, time1 = text
        line = "{} Time {} \n".format(value1, time1)
        f.write(line)
    return

def on_message(client, userdata, message):
    value = int(message.payload)
    if value == 1:
        dt = datetime.now().strftime('%Y-%m-%d %H:%M:%S.%f')[:-3]
        my_data = [value, dt]
        print("Value = ", value, " Time: ", dt)
        write_log(my_data, logfile)
def read_file():
    f = open(logfile)
    for line in f:
        number = line[0]
        if number == "1":
            print(line[0], "Leak Detected")
            subprocess.call(["php", "/home/pi/usefulscripts/swiftmail.php"])
            break
    f.close()

if os.path.exists("textfile.txt"):
    os.remove("textfile.txt")
    print("File deleted !")
else:
    print("File does not exist !")

schedule.every(60).minutes.do(read_file)

client.loop_start()
while 1:
    client.subscribe("emon/emontx3/DI2",2)
    client.subscribe("emon/emontx3/DI1",2)
    schedule.run_pending()
    client.on_message = on_message

time.sleep(10)
client.loop_stop()

```