* List of associated Files

1. Thread.py
2. Mosquitto\_1.py
3. mosquitto\_o.py
4. new.py
5. reconnect.py
6. scan.py
7. datascience.py
8. authentication.txt
9. data2.txt
10. data1.txt

* Multi-Threading
  1. Crontab
     + On reboot we will run thread.py file.
  2. Online
     + We will create an “online\_1” thread in which we will call “online1” function which on execution will activate online mqtt paho client.
  3. Offline
     + We will create an “offline\_1” thread in which we will call “offline1” function which on execution will activate mqtt paho client that will be connected to local broker.
  4. Time Check
  5. Route Check
     + We will create “Router\_check” thread in which we will call “route\_c” function. This function is keeps an every on network adaptor, whenever network adaptor goes this function send mosquito broker IP, username and password to the ESP8266 after it is being scanned.
  6. Execution
     + On the execution of this script first a while loop is executed and it wait until network adaptor is successfully up. After that we start and join the threads.
* Online Client
  1. Authentication
     + Connect client to “videoupload.hopto.org” using port “1234”
     + Connect using the authorized username and password
  2. On Connect
     + On connect subscribe to topics present in the “topics.txt” file along with the mac address of the respective ESP8266 inside the switch module.
  3. On Message
     + Whenever a message is received is forward to local mqtt client using the “publish.single” function.
* Offline Client
  1. Authentication
     + Connect client to mosquito broker running on local IP using the port 1883
     + Connect using the authorized username and password
  2. On Connect
     + On connect subscribe to topics mentioned in “topic.txt” file along with mac address. Also subscribe to the topics mentioned inside the “on\_connect” function.
  3. On Disconnect
     + Whenever local client gets disconnect from the server, “on\_disconnect” function is called. This function wait until network adaptor gets up and then it creates a new local mqtt client.
  4. On message
     + If message is received on the topic “/config/”. Data is written into data2.txt file and Data Acquisition function is called which basically locates IP of ESP8266 and send mqtt related data to ESP8266 TCP/IP server. Then client unsubscribe previous topic and subscribe to new topics present in topic.txt file.
     + If message is received on topic “/wifi/scan”, it call up the “Search” function. This function enlist the name of Wi-Fi connections available to Odroid from connection.
     + When a message is received on the topic “/wifi/ssid” it saved as SSID of new wifi connection and it waits for message on topic “/wifi/password”
* Day/Night Switching module
  1. Data\_acq function
  2. Schedular fuction
  3. Sunset / Sunrise Function
* Local functions
  1. New.py
     + my\_ip\_address function
     + ARPScanner function
     + Data1Writing function
     + Data2Reading function
     + FileWriting function
     + Data1\_Reading function
     + tcp\_client
     + Data\_Acquisition function
     + Data\_Forward function
     + tcp\_client\_lwt
  2. Scan.py
     + Search function
  3. Reconnect.py
     + Overview
     + all\_interfaces function
     + get\_ip\_address function
     + router\_c function