Accessing Ecoflow Cloud MQTT Server

# Install Power Shell 7

<https://learn.microsoft.com/en-us/powershell/scripting/install/installing-powershell>

# Get MQTT Credentials

1. Copy/paste the following into a script file such as ‘mycreds.ps1’

# Script courtasy of @lwsrbrts (Lewis Roberts)

# https://github.com/v1ckxy/ecoflow-withoutflow/issues/1

# Edit these

$password = 'YourPassword'

$useremail = 'you@emaildomain.com'

# The normal use is Unicode encoding but must specify UTF8 instead.

$base64password = [Convert]::ToBase64String([Text.Encoding]::UTF8.GetBytes($password))

$authObject = [PSCustomObject]@{

os = 'android'

scene = 'IOT\_APP'

appVersion = '4.0.0.53'

osVersion = '13'

password = $base64password

oauth = @{

bundleId = 'com.ef.EcoFlow'

}

email = $useremail

userType = 'ECOFLOW'

}

$Result = Invoke-RestMethod -Method Post -Uri 'https://api.ecoflow.com/auth/login/' -ContentType 'application/json' -Body $($authObject | ConvertTo-Json) -ResponseHeadersVariable 'Headers' -StatusCodeVariable 'StatusCode'

$TokenString = ConvertTo-SecureString -AsPlainText -Force -String $Result.data.token

$certificateData = Invoke-RestMethod -Method Get -Uri 'https://api.ecoflow.com/iot-auth/app/certification' -ContentType 'application/json' -Authentication Bearer -Token $TokenString

Write-Host "Ecoflow User ID: $($Result.data.user.userId)"

Write-Host "Certificate (MQTT) User: $($certificateData.data.certificateAccount)"

Write-Host "Certificate (MQTT) Password: $($certificateData.data.certificatePassword)"

Write-Host "URL: $($certificateData.data.url)"

Write-Host "Port: $($certificateData.data.port)"

Write-Host "Protocol: $($certificateData.data.protocol)"

1. Edit the script to include your Ecoflow App login (email/password)
2. Run the script to obtain the UserID, and certificate user/password for use with MQTT
3. Obtain the serial number of your device from the Ecoflow App by going to settings>Specifications

# Install MQTT Explorer

<https://mqtt-explorer.com/>

# Configure MQTT Explorer

1. Create new connection…

|  |  |  |
| --- | --- | --- |
| **Name** | **Validate Certificate** | **Encryption (tls)** |
| Ecoflow MQTT Server | ENABLED | ENABLED |
| **Protocol** | **Host** | **Port** |
| Mqtt:// | mqtt.ecoflow.com | 8883 |
| **Username** | **Password** | **<- Obtain from script above** |
| *<certificate MQTT user>* | *<certificate MQTT password>* |

1. Configure topics in Advanced settings…
   1. *Add each of the following (QoS = 0)… Refer to script above for Ecoflow User ID…*

/app/device/property/<Device Serial # in UPPERCASE>

/app/<Ecoflow User ID>/<Device Serial # in UPPERCASE>/thing/property/set

/app/<Ecoflow User ID>/<Device Serial # in UPPERCASE>/thing/property/get

/app/<Ecoflow User ID>/<Device Serial # in UPPERCASE>/thing/property/set\_reply

/app/<Ecoflow User ID>/<Device Serial # in UPPERCASE>/thing/property/get\_reply

* 1. Paste your Ecoflow User ID into the MQTT Client ID field

1. Click “BACK” then click “SAVE”

# Observe Data Flow

*Click CONNECT in MQTT Explorer to access Ecoflow public MQTT server*

## /app/device/property/<SN>

JSON payloads containing the state of various data points

These are constantly being updated by the device…

## /app/<USER>/<SN>/thing/property/set

Changes to settings in Ecoflow App will result in JSON payloads published here

These can be pasted into “publish” and modified (if desired) to change device settings

## /app/<USER>/<SN>/thing/property/set\_reply

These are ACK payloads published by the device on receipt of settings messages

## /app/<USER>/<SN>/thing/property/get

When the app is opened, for a given device, a payload like this is published to the ../get topic:

{"from":"IOS","id":"999999999","moduleType":0,"operateType":"latestQuotas","params":{},"version":"1.0"}

## /app/<USER>/<SN>/thing/property/get\_reply

In response to ‘latestQuotas’ (above) device will publish a payload with ‘state of everything’

This can be quite large with key/value pairs (and even nested array of key/values) for ‘everything’

# Notes

* Each device has a different schema of data points and a different process for when/how data points are updated under /app/device/property/<SN>
* There are MQTT libraries available for several scripting languages (Power Shell, Perl, Python, etc.)
* Home Assistant (home-assistant.io) supports MQTT integration including the MQTT triggers for automations, MQTT based sensor entities, etc. by using a local MQTT broker (Mosquitto) to bridge between HA and the public MQTT server…
* Other smart home platforms may support MQTT integration in various ways as well…