



Guide: How to Customize the Daly BMS Python Script

Since the script runs as a **systemd service** in the background, you follow a simple three-step process: **Edit**, **Test**, and **Restart**.

1. Open the Script for Editing

Use the terminal to open the file with the `nano` editor:

```
nano /home/pi/daly.py
```

2. Common Configuration Changes

At the very top of the script, you will see the `CONFIGURATION` section. This is where most changes happen:

- **Battery Capacity (Ah):** Update the `MAX_AH` variable (e.g., `MAX_AH = 280.0` for a 280Ah bank).
- **Polling Interval:** Change `INTERVALL = 3` if you want it to update faster or slower (in seconds).
- **MQTT Settings:** Update `MQTT_SERVER` or `MQTT_PORT` if your Broker IP changes.

3. Adjusting for More Cells (e.g., 8S to 16S)

If you switch from a 24V (8 cells) to a 48V (16 cells) system, you need to tell the script to look for more cell voltages. Find the `range` loops in the code:

- **For 8 Cells:** Use `range(1, 9)`
- **For 16 Cells:** Use `range(1, 17)`

(Note: In Python, the second number is "up to but not including," so 17 means it reads cells 1 through 16.)

4. Save and Test

1. Press **Ctrl + O**, then **Enter** to save.
2. Press **Ctrl + X** to exit the editor.
3. **Crucial Step:** Test the code manually before letting the service take over to ensure there are no syntax errors:

```
python3 /home/pi/daly.py
```

*If it prints data and no errors appear, press **Ctrl + C** to stop it.*

5. Restart the Background Service

⚙️ How to Create the Background Service

A **service** ensures that the script starts automatically when the Pi boots and restarts if it ever crashes.

1. Create the Service File

Run this command to create a new service configuration: `sudo nano /etc/systemd/system/dalybms.service`

2. Paste the Configuration

Copy and paste this block into the editor:

```
[Unit]
Description=Daly BMS MQTT Service
After=network.target

[Service]
ExecStart=/usr/bin/python3 /home/pi/daly.py
WorkingDirectory=/home/pi
StandardOutput=inherit
StandardError=inherit
Restart=always
RestartSec=10
User=root

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

Press **Ctrl+O, Enter**, then **Ctrl+X** to save and exit.

3. Activate the Service

Run these three commands to tell the system about the new service and start it:

- **Reload systemd:** `sudo systemctl daemon-reload`
- **Enable Autostart:** `sudo systemctl enable dalybms.service`
- **Start it now:** `sudo systemctl start dalybms.service`

4. Checking the Status

To see if the script is running correctly, use: `sudo systemctl status dalybms.service`

It should say: Active: active (running)

Troubleshooting (English)

If your friend needs to see what the script is "saying" (for debugging), he can use the **Live Log** command: `sudo journalctl -u dalybms.service -f`

```
sudo systemctl restart dalybms.service
```

Cheat Sheet: Useful Commands

Task	Command
Edit Script	<code>nano /home/pi/daly.py</code>
Restart Service	<code>sudo systemctl restart dalybms.service</code>
Stop Service	<code>sudo systemctl stop dalybms.service</code>
Check Live Logs	<code>sudo journalctl -u dalybms.service -f</code>