Shelly Pro 3EM Emulator for HomeWizard P1 Meter

This Docker container emulates a Shelly Pro 3EM energy meter by reading data from your HomeWizard P1 meter. This provides a fast, reliable proxy for devices like the Marstek Venus V3 that may have connection issues with the HomeWizard directly.

Features

- ✅ Polls HomeWizard P1 meter every second for fast updates
- ✅ Caches data in memory for instant responses
- ✅ Emulates both Gen1 and Gen2 Shelly API endpoints
- ✅ Single-phase data mapped to Phase A (suitable for most residential setups)
- ✅ No external dependencies or cloud services required

Quick Start

1. Create Project Directory

```bash
mkdir shelly-emulator
cd shelly-emulator
```

2. Create Files

Create these 4 files in the directory:

- `emulator.py` (the Python application code)
- `Dockerfile` (Docker image configuration)
- `docker-compose.yml` (Docker Compose configuration)
- `requirements.txt` (Python dependencies)

Copy the content from the artifacts provided.

3. Configure Your Setup

Edit `docker-compose.yml` and change the `HOMEWIZARD_HOST` to your HomeWizard P1 meter's IP address:

```yaml
environment:
  - HOMEWIZARD_HOST=192.168.1.50  # Change this to your HomeWizard IP
  - PORT=8080
```

If you want to use a different port, change the port mapping:

```yaml
ports:
  - "8080:8080"  # Change first number to desired external port
```

4. Enable HomeWizard Local API

Make sure you've enabled the Local API on your HomeWizard P1 meter:

1. Open the HomeWizard Energy app
2. Go to **Settings** → **Meters** → Select your meter
3. Enable **Local API**

5. Build and Run

```bash
# Build and start the container
docker-compose up -d

# View logs
docker-compose logs -f

# Stop the container
docker-compose down
```

6. Test the Emulator

Once running, test it with:

```bash
# Get device info
curl http://localhost:8080/shelly

# Get current status
curl http://localhost:8080/status

# Get emeter data
curl http://localhost:8080/emeter/0
```

You should see JSON responses with energy data.

Configure Your Marstek Venus V3

Now configure your Marstek to use the emulated Shelly meter:

1. Access your Marstek Venus V3 configuration interface
2. Look for the energy meter settings
3. Select **Shelly Pro 3EM** as the meter type
4. Enter the IP address of the machine running this Docker container
5. Set the port to `8080` (or whatever you configured)
6. Save and test the connection

API Endpoints

The emulator provides these endpoints compatible with Shelly Pro 3EM:

Gen1 API (most common)

- `GET /shelly` - Device information
- `GET /status` - Full device status
- `GET /settings` - Device settings
- `GET /emeter/0` - Energy meter data

Gen2 RPC API

- `GET /rpc/EM.GetStatus` - Energy meter status

Data Mapping

Since HomeWizard P1 meters typically measure single-phase or combined three-phase data, the emulator maps data as follows:

- **Phase A**: Gets all the active power, current, and voltage from HomeWizard
- **Phase B & C**: Set to zero (suitable for single-phase installations)
- **Total values**: Match Phase A values

If you have a true three-phase setup, you may need to modify the `convert_hw_to_shelly()` method in `emulator.py`.

Troubleshooting

Container won't start

```bash
```

```
# Check logs for errors
docker-compose logs

# Ensure HomeWizard IP is correct
# Ensure HomeWizard Local API is enabled
```

No data appearing

```bash
# Check if HomeWizard is reachable
curl http://YOUR_HOMEWIZARD_IP/api/v1/data

# Check emulator logs
docker-compose logs -f
```

Marstek can't connect

- Ensure the Marstek and Docker host are on the same network
- Check firewall rules on the Docker host
- Verify the port is correctly mapped in docker-compose.yml
- Try accessing the emulator from the Marstek's network using curl

Advanced Configuration

Change Polling Interval

Edit `emulator.py` and modify this line:

```python
await asyncio.sleep(1)  # Poll every second
```

Change `1` to your desired interval in seconds.

Enable Debug Logging

Edit `emulator.py` and change the logging level:

```python
logging.basicConfig(
    level=logging.DEBUG,  # Changed from INFO to DEBUG
    format='%(asctime)s - %(name)s - %(levelname)s - %(message)s'
)
```

Then rebuild:

```bash
docker-compose up -d --build
```

System Requirements

- Docker and Docker Compose installed
- Network access to HomeWizard P1 meter
- Minimal resources: ~50MB RAM, negligible CPU

Support

If you encounter issues:

1. Check the logs: `docker-compose logs -f`
2. Verify HomeWizard connectivity: `curl http://HOMEWIZARD_IP/api/v1/data`
3. Test emulator endpoints: `curl http://localhost:8080/status`

License

This is a custom solution for bridging HomeWizard P1 meters with devices expecting Shelly Pro 3EM API.