[] | *heckert/de/fems/apps/FEMS_App_22_Websocket_JSON_lesend_DE.png*

FEMS App REST/JSON lesend

Version:2023.6.1

Inhaltsverzeichnis

1. FEMS App REST/JSON lesend **2**

1.1. Voraussetzungen . . . 2
1.2. Grundlagen REST/JSON . . . 2
1.3. Lesezugriff . . . 2
1.3.1. /channel Endpunkt . . . 3
1.3.2. Datenpunkte . . . 3
1.3.3. Beispiel 1 — Abfrage des Ladezustands: cURL . . . 4
1.3.4. Beispiel 2 — Abfrage des Ladezustands: Python . . . 5
1.3.5. Beispiel 3 — Abfrage des Ladezustands: Talend API Tester . . . 6

2. Kontakt **8**

3. Verzeichnisse **9**

3.1. Abbildungsverzeichnis. . . . 9
3.2. Tabellenverzeichnis . . . 10

1. FEMS App REST/JSON lesend

Diese Anleitung dient der Beschreibung des Lesezugriffs auf ein Heckert Solar-Stromspeichersystem mittels REST-/JSON-API. Anschließend wird die Funktionsweise der Schnittstelle erklärt.

1.1. Voraussetzungen

Das auf das Stromspeichersystem zugreifende Gerät (z. B. Notebook/PC) muss direkten Zugriff auf die IP-Adresse des EMS haben — also z. B. im selben physischen Netzwerk angeschlossen sein.

1.2. Grundlagen REST/JSON

Die REST-/JSON-Schnittstelle ermöglicht den Zugriff auf das EMS im lokalen Netzwerk über eine an REST angelehnte Schnittstelle.

1.3. Lesezugriff

Diese App stellt eine an REST angelehnte Schnittstelle zur Verfügung, mit der Datenpunkte im System ausgelesen werden können.

Diese App ist im Standard-Lieferumfang des EMS enthalten.

Die Basis-Adresse für die REST-Zugriffe lautet `http://<BENUTZER>:<PASSWORT>@<IP>:80/rest`

- `http` ist das Protokoll
- `<BENUTZER>` ist der Benutzername. Da die Authentifizierung lediglich über das Passwort erfolgt, kann hier ein beliebiger Wert (z. B. "x") angegeben werden
- `<PASSWORT>` ist das Passwort des Benutzers. Der Standard "Gast"-Benutzer im EMS hat das Passwort "**user**"
- `<IP>` ist die IP-Adresse des EMS
- `80` ist der Port für die REST/JSON-Api (optional)

Wenn Ihr EMS also die lokale IP-Adresse '192.168.0.23' hat, lautet die Basis-Adresse für REST-Zugriffe `http://x:user@192.168.0.23:80/rest`

Aus Sicherheitsgründen werden *Simple Authentication Requests* nicht unterstützt, da bei dieser Variante Passwörter über die URL übertragen werden. Für REST-Calls muss explizit Header-Authentication verwendet werden. Eine Abfrage direkt über den Browser (ohne Erweiterung) ist folglich **nicht** möglich. In diesem Fall erscheint eine Fehlermeldung, vgl. Abbildung Fehlermeldung bei *Simple Authentication Requests*. Für eine Abfrage im Browser, nutzen Sie bitte die Erweiterung Talend-API-Tester wie hier beschrieben.

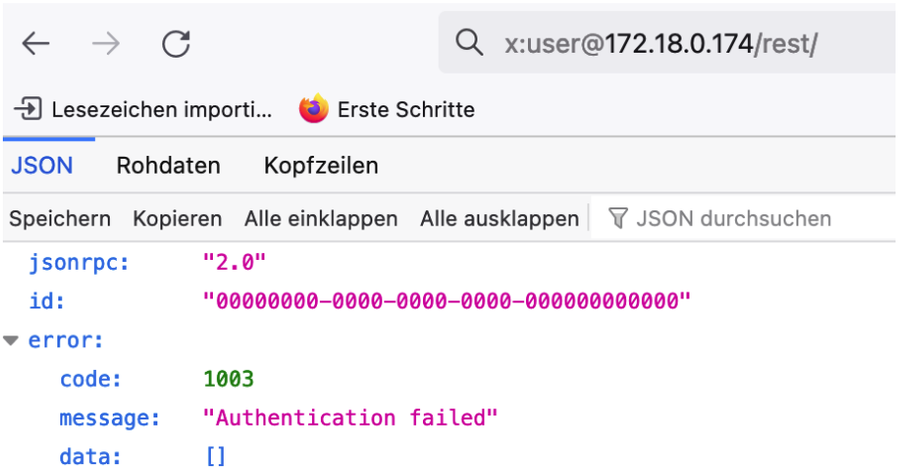


Abbildung 1. Fehlermeldung bei *Simple Authentication Requests*

1.3.1. /channel Endpunkt

Der `/channel` Endpunkt ermöglicht den Zugriff auf einzelne Datenpunkte, sogenannte "Channels", im System.

Die vollständige Adresse des Endpunkts lautet:

http://x:<PASSWORT>@<IP>:80/rest/channel/<KOMPONENTE>/<KANAL>

- `<KOMPONENTE>` ist die ID der Komponente
- `<KANAL>` ist die ID des Kanals

1.3.2. Datenpunkte

Die folgenden Datenpunkte der Komponente *_sum* können ausgelesen werden:

Datenpunkt	Beschreibung	Einheit
State	0: Ok, 1: Info, 2: Warning, 3: Fault	
EssSoc	Ladezustand [0 - 100]	Prozent [%]
EssActivePower	AC-seitige Wirkleistung des Speichers inkl. überschüssiger DC-Erzeugung bei Hybrid-Wechselrichter	Watt [W]
EssReactivePower	AC-seitige Blindleistung des Speichers	Voltampere Reaktiv [var]
GridActivePower	Wirkleistung am Netzanschlusspunkt	Watt [W]
GridMinActivePower	Minimale je gemessene Wirkleistung am Netzanschlusspunkt	Watt [W]
GridMaxActivePower	Maximale je gemessene Wirkleistung am Netzanschlusspunkt	Watt [W]

ProductionActivePower	Wirkleistung des PV-Ertrags und ggf. Ertrag durch externe Wechselrichter	Watt [W]
ProductionMaxActivePower	Maximale je gemessene Wirkleistung der PV-Anlage	Watt [W]
ProductionAcActivePower	Wirkleistung der externen AC-Wechselrichter	Watt [W]
ProductionDcActualPower	Leistung der DC-Erzeugung des Hybridwechselrichters	Watt [W]
ConsumptionActivePower	Wirkleistung des elektrischen Verbrauchs	Watt [W]
ConsumptionMaxActivePower	Maximale je gemessene Wirkleistung des elektrischen Verbrauchs	Watt [W]
EssActiveChargeEnergy	Kumulierte elektrische Energie der AC-seitigen Speicherbeladung inkl. überschüssige PV-Erzeugung beim Hybrid-Wechselrichter	Wattstunden [Wh]
EssActiveDischargeEnergy	Kumulierte elektrische Energie vom Speicher zum Verbrauch über AC-Ausgang des Wechselrichters inkl. PV-Erzeugung	Wattstunden [Wh]
GridBuyActiveEnergy	Kumulierte elektrische Energie des Netzbezuges	Wattstunden [Wh]
GridSellActiveEnergy	Kumulierte elektrische Energie der Einspeisung	Wattstunden [Wh]
ProductionActiveEnergy	Kumulierte elektrische Energie der PV-Erzeugung + Erzeugung externer Wechselrichter	Wattstunden [Wh]
ProductionAcActiveEnergy	Kumulierte elektrische Energie der externen Wechselrichter	Wattstunden [Wh]
ProductionDcActiveEnergy	Kumulierte elektrische Energie der PV-Erzeugung des Wechselrichters	Wattstunden [Wh]
ConsumptionActiveEnergy	Kumulierter elektrischer Verbrauch	Wattstunden [Wh]
EssDcChargeEnergy	Kumulierte DC-elektrische Energie der Speicherbeladung	Wattstunden [Wh]
EssDcDischargeEnergy	Kumulierte DC-elektrische Energie der Speicherentladung	Wattstunden [Wh]
EssDischargePower	Tatsächliche AC-seitige Wirkleistung des Speichers	Watt [W]
GridMode	1: On-Grid, 2: Off-Grid	

Table 1. Datenpunkte der Komponente *_sum*

1.3.3. Beispiel 1 — Abfrage des Ladezustands: cURL

Das Kommandozeilen-Programm cURL ist sowohl unter Windows und Linux vorinstalliert.

Um den Ladezustand des Stromspeichers auszulesen, senden Sie einen `GET`-Request an die Adresse: http://x:user@192.168.0.23:80/rest/channel/_sum/EssSoc

Sie erhalten eine Antwort im JSON-Format:

Windows

Der folgende Befehl speichert die Antwort im JSON-Format in die Datei *out.json*

```
>curl -o out.json http://x:user@192.168.0.23:80/rest/channel/_sum/EssSoc
```

Um den Inhalt der Datei auszugeben, nutzen Sie:

```
>type out.json
```

Ausgabe:

```
{"address":"_sum/EssSoc","type":"INTEGER","accessMode":"RO","text":"","unit":"%","value":
99}
```

Den Wert des Ladezustands finden Sie unter *value*. Im Beispiel oben beträgt er 99 %.

Linux

Der folgende Befehl speichert die Antwort im JSON-Format in die Datei *out.json*

```
$curl -o out.json http://x:user@192.168.0.23:80/rest/channel/_sum/EssSoc
```

Um den Inhalt der Datei auszugeben, nutzen Sie:

```
>cat out.json
```

Ausgabe:

```
{"address":"_sum/EssSoc","type":"INTEGER","accessMode":"RO","text":"","unit":"%","value":
99}
```

Den Wert des Ladezustands finden Sie unter *value*. Im Beispiel oben beträgt er 99 %.

1.3.4. Beispiel 2 — Abfrage des Ladezustands: Python

Python Versionen für Windows und Linux erhalten Sie hier: https://www.python.org/downloads/

Um den Ladezustand des Stromspeichers auszulesen, muss ebenfalls ein `GET`-Request an die Adresse: `http://x:user@192.168.0.23:80/rest/channel/_sum/EssSoc` gesendet werden.

Hierfür kann die *requests*-Bibliothek genutzt werden, die zu Beginn importiert werden muss:

```
import requests

url = 'http://192.168.0.23:80/rest/channel/_sum/EssSoc'

user = 'x'
password = 'user'

session = requests.Session()
session.auth = (user, password)

response = session.get(url)
response.raise_for_status()
```

Der Befehl liefert eine Antwort im JSON-Format. Diese kann mit dem folgenden Befehl ausgegeben werden:

```
print(response.text)
```

Ausgabe:

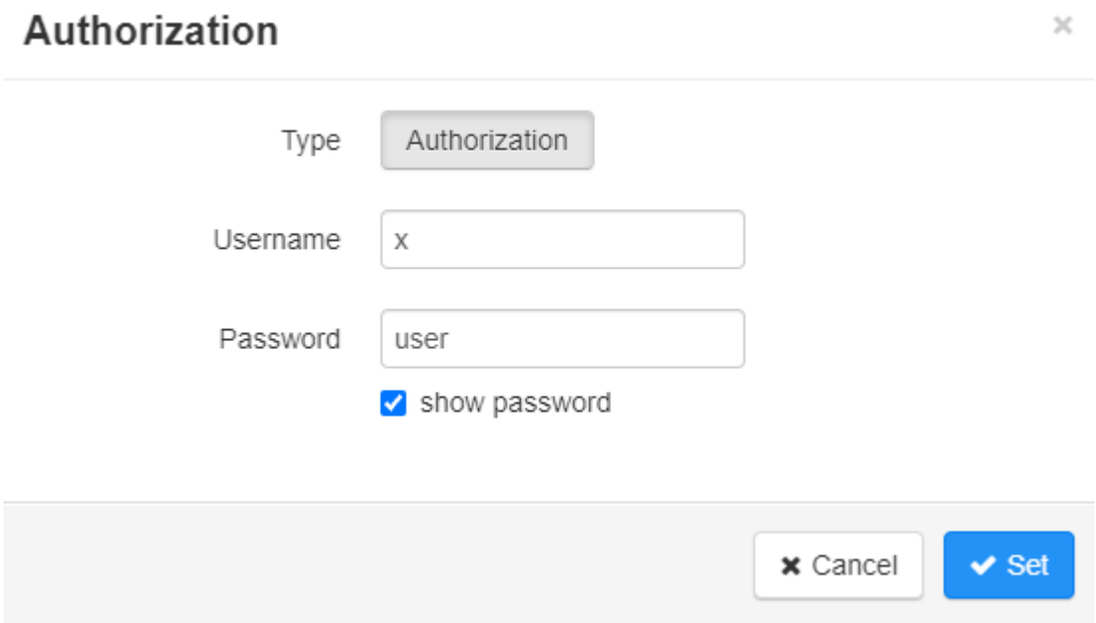
```
{"address":"_sum/EssSoc","type":"INTEGER","accessMode":"RO","text":"","unit":"%","value":
99}
```

Den Wert des Ladezustands finden Sie unter *value*. Im Beispiel oben beträgt er 99 %.

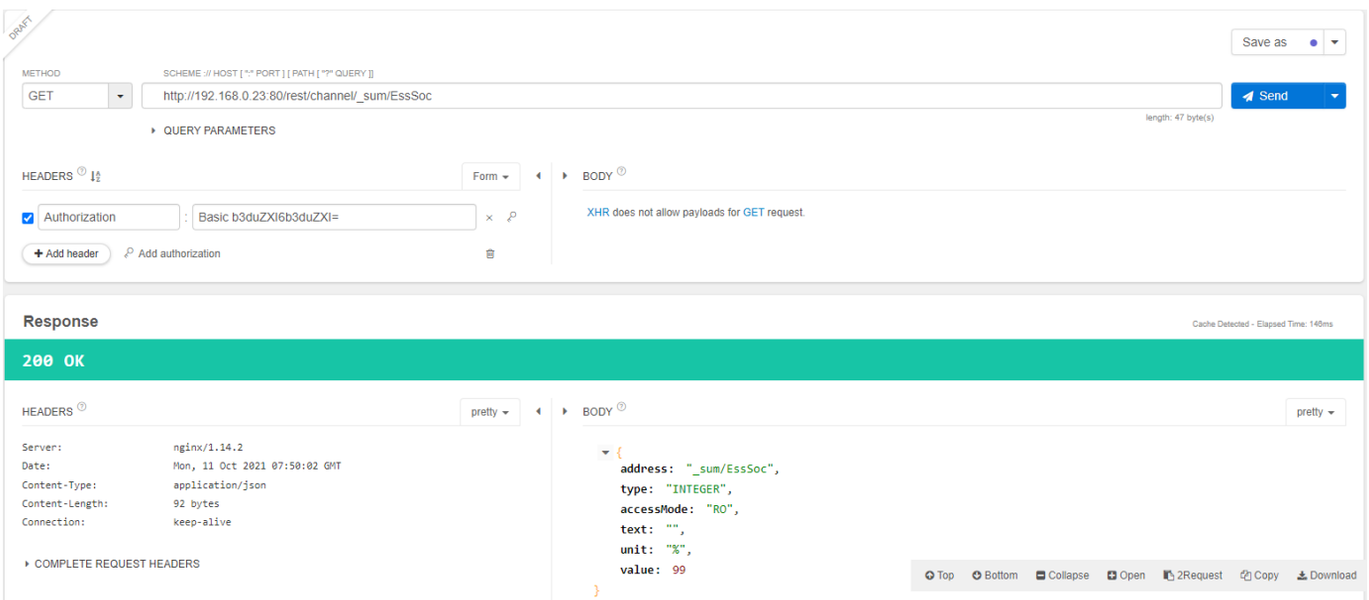
1.3.5. Beispiel 3 — Abfrage des Ladezustands: Talend API Tester

Talend-API-Tester ist eine Erweiterung für Google Chrome, die es ermöglicht, REST-API's zu testen.

Zunächst muss ein *Authorization*-Header hinzugefügt werden:



Anschließend kann der GET-Request ausgeführt werden.



Den Wert des Ladezustands finden Sie unter *value*. Im Beispiel oben beträgt er 99 %.

2. Kontakt

Für Unterstützung wenden Sie sich bitte an:

Symphon-E Service

Telefon Service: +49 (0) 371 45 85 68 - 100

E-Mail Service: symphon-e@heckert-solar.com

3. Verzeichnisse

3.1. Abbildungsverzeichnis

Abbildung 1. Fehlermeldung bei Simple Authentication Requests

3.2. Tabellenverzeichnis

Table 1. Datenpunkte der Komponente _sum