Schalten und Messen von 230 V

Für die Aktivierung von Ressourcen mit Strom bieten sich verschiedene Möglichkeiten an. Entweder kann dies direkt an der Ressource selbst erfolgen, über einen Zwischenstecker oder in der Unterverteilung. Für diese Anwendungsfälle gibt es entsprechende Produkte.

Das Trait "Powerable" eignet sich für die Aktivierung von Ressourcen mit Strom. Damit können die Ressourcen eingeschaltet werden.

Je nach den verwendeten Geräten zur Stromaktivierung lässt sich auch der Stromverbrauch messen.

Wenn Ressourcen mit 230V aktiviert werden, ist es wichtig zu beachten, dass diese Ressourcen auch einen unerwarteten Spannungsverlust verkraften können sollten. Die direkte Schaltung der Eingangsspannung mag zwar die einfachste Methode sein, ist aber nicht immer die beste Option, insbesondere wenn es um Ressourcen geht, die solche Spannungsabfälle nicht vertragen.

- Nous A1T mit Tasmota Firmware
- Shelly 1
- Steckerschutz und Sicherheitsplomben
- Shelly 1 Plus
- Shelly Plug S
- Shelly Plug 1
- Tipps zu Shelly Generationen
- Gosund EP2

Nous A1T mit Tasmota Firmware

Mit dem Nous A1T lässt sich kostengünstig über einen Zwischenstecker die Stromversorgung von Ressourcen einschalten. Durch die Tasmota Firmware kann auch die Software angepasst werden.

Wichtig ist, dass es sich bei dem Nous um die Tasmota-Variante handelt, da nur diese ohne externe Cloud funktioniert.

Wichtige Links

- https://tasmota.github.io/docs/Firmware-Builds
- https://gitlab.com/fabinfra/fabaccess/grafana/-/tree/main?ref_type=heads
- https://tasmota.github.io/docs/Commands

Tasmota Schaltsteckdose ins Netzwerk einbinden - HowTo

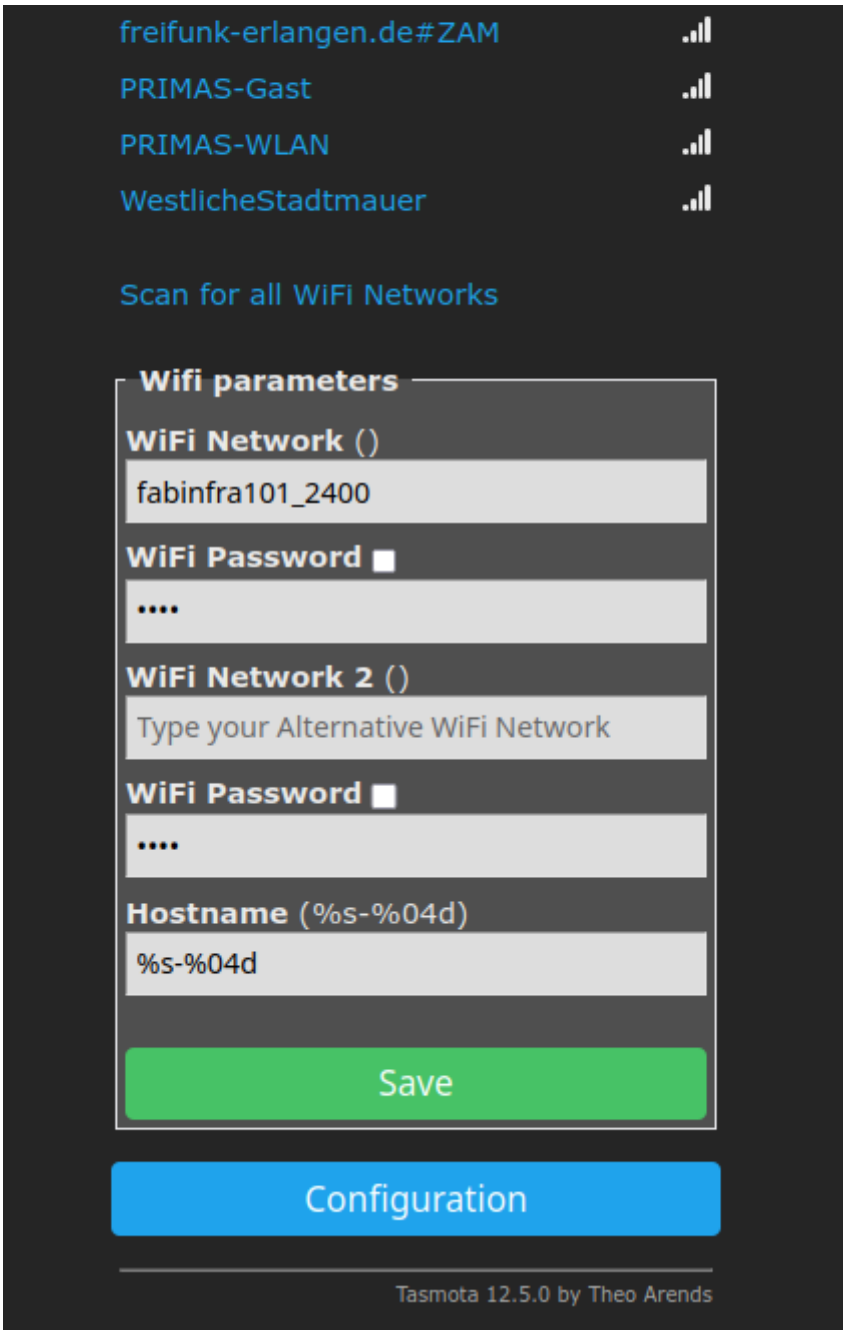
Diese Anleitung basiert auf https://nous.technology/product/a1t/de.html

Schaltsteckdose einstecken und Web Interface öffnen

1. Die Dose öffnet kurze Zeit nach Einstecken einen eigenen Access Point. Wir verbinden uns mit dieser SSID und rufen das Web Interface http://192.168.4.1 (HTTPS nicht von Haus aus unterstützt!) vom Nous auf. In der Regel ist die IP-Adresse folglich `192.168.4.1`. Siehe auch https://nous.technology/product/a1t/de.html

Mit Wifi SSID verbinden

Wir suchen unsere lokale Werkstatt-SSID aus und verbinden das Modul:


freifunk-erlangen.de#ZAM
PRIMAS-Gast
PRIMAS-WLAN
WestlicheStadtmauer
Scan for all WiFi Networks
Wifi parameters
WiFi Network ()
fabinfra101_2400
WiFi Password
••••
WiFi Network 2 ()
Type your Alternative WiFi Network
WiFi Password
••••
Hostname (%s-%04d)
%s-%04d
Save
Configuration
Tasmota 12.5.0 by Theo Arends


Tasmota
Successful WiFi Connection
Redirecting to new device's IP address
192.168.188.39
Tasmota 12.5.0 by Theo Arends

Nach der Kopplung findet sich die Schaltsteckdose nun im Netzwerk eingebunden wieder. Wir verwenden die im Beispiel angegebene IP-Adresse `192.168.188.39` und rufen erneut das Web Interface auf: http://192.168.188.39. Damit sind wir in der Lage alle weiteren Einstellungen vorzunehmen.

Tipp: Wir empfehlen das Gerät nach dem Einbinden sinnvoll in der Netzwerkübersicht des Routers/Access Points zu benennen.

Modul konfigurieren

Folgende Configs sind - ausgehend von den Werkseinstellungen - je Steckdose zu ändern. Alle hier nicht angegebenen Einstellungen können beim Standard belassen werden.

Configuration → Configure MQTT

- Host (Die IP-Adresse oder Hostname des FabAccess Servers)
- Port (Standard 1883)
- Benutzername
- Passwort

Configuration → Configure Other

- Passwort für den Web Admin setzen (Der Nutzername ist `admin` und kann nicht verändert werden)
- Device Name (für die eigene Übersicht)
- Friendly Name 1 (für die eigene Übersicht)
- Topic: entweder statisch vergeben (z.B. tasmota_1 oder die Standards nutzen. Die ID ist sehr wichtig, da wir genau diese für die FabAccess Actor Konfiguration in `bffh.dhall` benötigen, allerdings dort ohne führendes `tasmota_`). Der Standardwert ist `tasmota_%06X`.
- HTTP API **deaktivieren**

Spezifische Kommandos in der Konsole

- PowerOnState anpassen. Die Steckdose kann so konfiguriert werden, dass sie immer den letzten gespeicherten Zustand einnimmt (d.h. wenn die Dose vorher aus war, wird sie beim nächsten Einstecken auch aus sein). Das Setzen erfolgt in diesem fall über die Console durch Ausführen von `PowerOnState 3` (Bestätigen mit Enter). Es gibt weitere Modi (0, 1, 2, 3, 4, 5) - siehe Dokumentation.

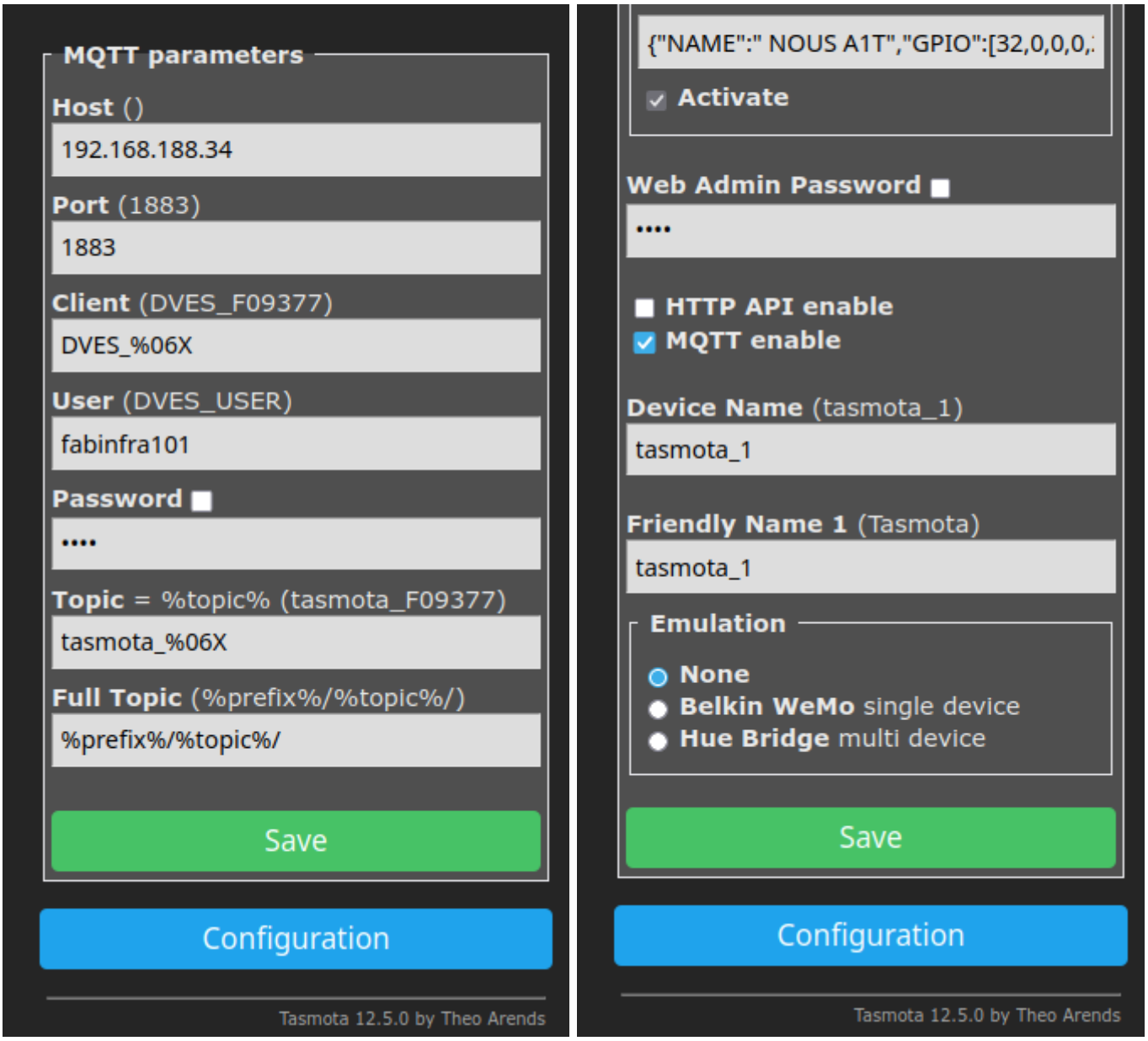
- Werkseinstellungen durch 7x hintereinander die Steckdose schnell ein- und wieder ausstecken. Diese Option ist praktisch, aber bietet minderen Manipulationsschutz. Das kann mit SetOption65 per Console unterbunden werden: `SetOption65 1` (Bestätigen mit Enter)
- Den Button an der Steckdose deaktivieren, damit er nicht per Hand gedrückt werden kann. Standardmäßig lässt sich die Nous A1T einfach umschalten per Druck. Die SetOption73 verhindert das. Das Setzen erfolgt durch Ausführen von `SetOption73 1` (Bestätigen mit Enter)
- Zeitzone setzen. So erhalten wir stichhaltige Informationen über MQTT. Hierzu gibt es ein praktisches Onlinewerkzeug: https://tasmotatimezone.com. Für die angegeben Location und Zeitzone erhalten wir den passenden Konfigurationsbefehl für Tasmota. Beispielbefehl: `Backlog Latitude 50.831742; Longitude 12.94054565505144; TimeDST 0,0,3,1,1,120; TimeSTD 0,0,10,1,1,60; TimeZone 99`. Hinweis: Der Zeitserver (NTP) ist automatisch vorkonfiguriert (Befehl `NtpServer`) und muss nicht angefasst werden.

Kalibrierung der Stromwerte (Netzspannung) für korrektes Monitoring/Reporting

- als kurzes Tutorial in Englisch: https://tasmota.github.io/docs/Power-Monitoring-Calibration/#setup
- als 2-minütiges Praxisvideo auf Youtube: https://www.youtube.com/embed/9M2G2EzEXAk

TLS-Unterstützung für sichere Kommunikation

- Firmware Upgrade / Flash custom TLS enabled Firmware. Diese erlaubt die (fast) automatische Nutzung des Lets Encrypt Root Zertifikats ohne sämtliche weitere Konfiguration auf dem Modul. Lediglich mosquitto und BFFH müssen Lets Encrypt verwenden



Die Screenshots zeigen ein paar der obigen Einstellungen

Bekannte Probleme

Werkseinstellungen als Manipulationsmethode

> Nutzer können die Admins ärgern

Nutzer können FabAccess umgehen, wenn sie die Steckdosen auf Werkseinstellung zurücksetzen und die Tasmota Schaltsteckdose umkonfigurieren. Um die Smart-Steckdose auf die Werkseinstellungen zurückzusetzen, benötigen Sie: Stecken Sie das Gerät 6 mal schnell hintereinander ein und aus und lassen Sie es das siebte Mal eingeschaltet. Die LED sollte zu blinken beginnen. Dies bedeutet, dass die Steckdose wieder angeschlossen werden kann. Per `SetOption65` kann dieses Verhalten geändert werden (siehe oben). Allerdings bewirkt diese Option auch, dass im Falle fehlender Verbindung keine Konfiguration der Dosen mehr erfolgen kann. Der einzige Weg ist dann das Zurücksetzen per OTA-Flash.

Schaltsteckdose schaltet per MQTT-Befehl nicht oder blinkt

> Die Nous Steckdose blinkt, aber sie ist im Netzwerk erreichbar (nicht im Access Point Modus)

Wenn die Schaltsteckdose blinkt, sich aber nicht bedienen lässt: Nachprüfen, ob der MQTT Server für Geräte erreichbar ist. Die Nous Steckdosen blinken in der Regel dann, wenn keine Verbindung zum Server besteht oder weil ggf. die IP-Adresse falsch konfiguriert ist. In diesem Fall die Tasmota Settings und die des Mosquitto Servers prüfen. Übrigens blinken die Schaltsteckdosen bei fehlender MQTT-Verbindung nicht, wenn sie eingeschalten sind - dann leuchtet die LED durchgängig grün.

https://videos.stadtfabrikanten.org/videos/embed/2023f007-a544-4f05-82c6-d437ffb6c17b

Tasmota Schaltsteckdose auf MQTT-Funktionalität testen

Nach dem Einbinden und Konfigurieren unserer Schaltsteckdose in unser Netzwerk können wir sie von einem beliebigen Linux Client überprüfen, ohne dabei FabAccess anzufassen. Somit schließen wir von Anfang an Probleme aus. Wir nutzen dazu das Kommando `mosquitto_pub`. Für das Schalten sprechen wir nicht direkt die Nous A1T an (denn wir haben der Nous Dose die Server-Informationen bereits mitgegeben), sondern

den MQTT Server mit seiner IP-Adresse und Begleitinformationen Port, Benutzer und Passwort. Dabei übergeben wir mit `-t` außerdem das Topic und mit `-m` den Wert. In folgenden Beispiel schalten wir die Dose zunächst aus und dann an. Das Topic setzt sich aus dem Präfix `cmnd`, dem Gerätename `tasmota_1` und dem Kommando `POWER` zusammen.

```
#mosquitto_pub verfügbar machen, falls nicht aufrufbar
sudo apt install mosquitto-clients
```

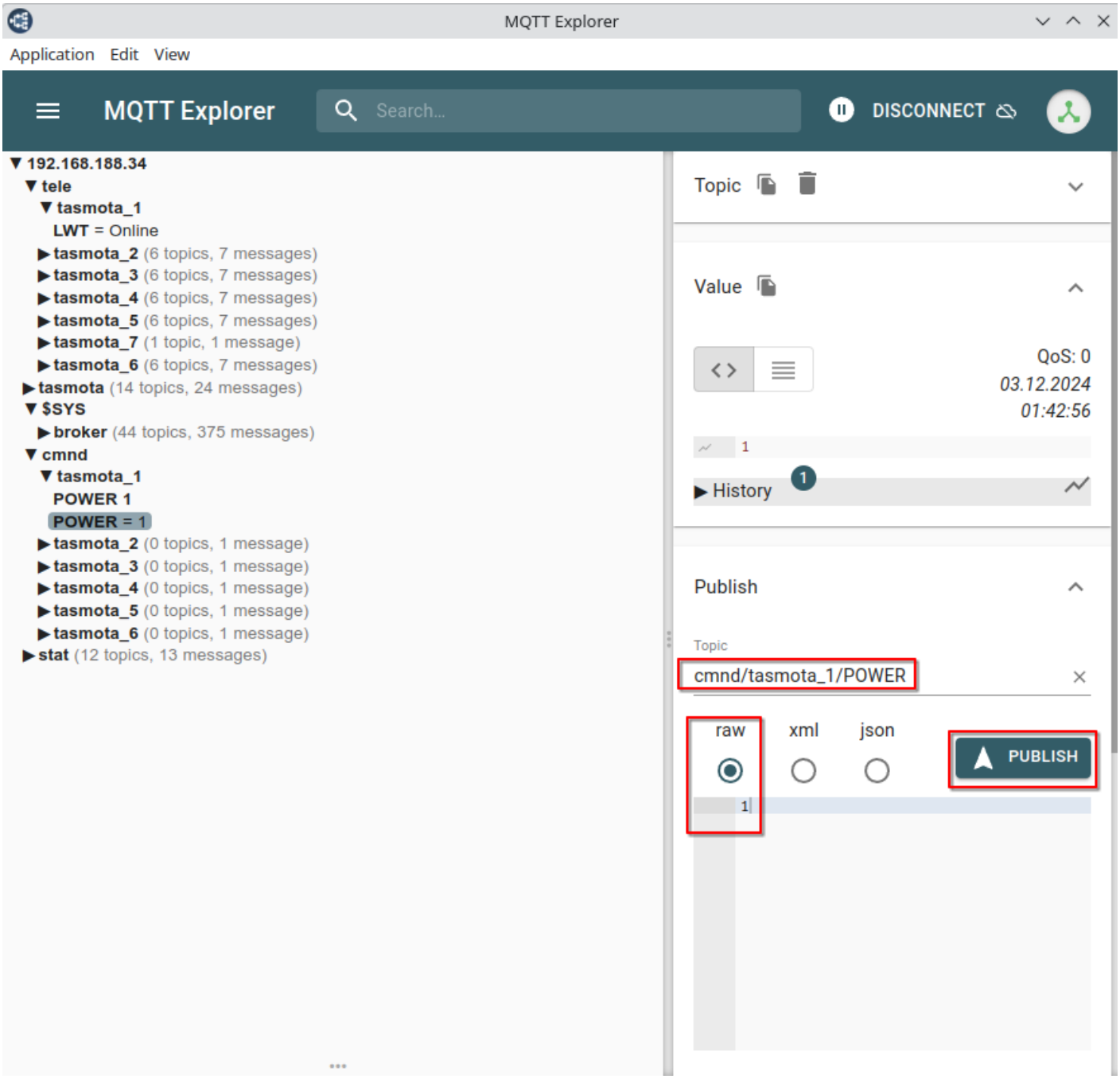
Ausschalten:

```
mosquitto_pub -d -h 192.168.188.34 -u fabinfra101 -P fablocal -p 1883 -t "cmnd/tasmota_1/POWER" -m '0'
```

Einschalten:

```
mosquitto_pub -d -h 192.168.188.34 -u fabinfra101 -P fablocal -p 1883 -t "cmnd/tasmota_1/POWER" -m '1'
```

Wer nicht auf Kommandozeile operieren will oder etwas Debug-Übersicht benötigt, der kann den MQTT Explorer verwenden. Im Feld `Topic` kann der obige Befehl eingegeben werden. der Wert kann als `raw` Wert eingetippt werden. Dann klicken wir auf `PUBLISH` und führen die Aktion aus.



Tasmota Actor für BFFH installieren

Diese Anleitung findest du unter Aktor: Tasmota.

Tasmota Web UI in FabAccess-Farben

Zum "hübsch" machen können wir alle Farben in Tasmota ändern. Das hat einerseits eine kosmetische Wirkung, andererseits eine warnende bzw. informierende! Sobald wir sehen,

dass unsere Nous Steckdose andere Farben als der Standard hat wissen wir, dass wir sie schonmal konfiguriert haben und dass sie zu unserem FabAccess Setup gehört. In Umgebungen, wo vielleicht noch andere Geräte im Netzwerk ihr (Un)wesen treiben, hilft das:

```
mosquitto_pub -d -h MQTT_SERVER -p MQTT_SERVER_PORT -u MQTT_USER -P MQTT_PASSWORD -t "cmnd/tasmota_1/WebColor" -m '{"WebColor":["#00d4aa","#3c474d","#3c474d","#000000","#dddddd","#00d4aa","#3c474d","#ff5661","#008000","#faffff","#00d4aa","#009275","#d43535","#931f1f","#47c266","#5aaf6f","#faffff","#999999","#00d4aa"]}'
```

Der Standard kann wie folgt zurückgesetzt werden:

```
mosquitto_pub -d -h MQTT_SERVER -p MQTT_SERVER_PORT -u MQTT_USER -P MQTT_PASSWORD -t "cmnd/tasmota_1/WebColor" -m '{"WebColor":["#eaeaea","#252525","#4f4f4f","#000000","#dddddd","#65c115","#1f1f1f","#ff5661","#008000","#faffff","#1fa3ec","#0e70a4","#d43535","#931f1f","#47c266","#5aaf6f","#faffff","#999999","#eaeaea","#08405e"]}'
```

> Die exakte Doku der Farben findet sich in https://github.com/arendst/Tasmota/blob/development/tasmota/my_user_config.h

Erweitertes Setup - Custom Tasmota Firmware mit TLS-Support

Neben der unverschlüsselten Standardvariante mit MQTT lässt sich auch eine sichere MQTTS-Verbindung herstellen, sofern Tasmota entsprechend dafür ausgestattet ist. Hierzu ist das Kompilieren der Firmware notwendig, weil TLS aus Platzgründen standardmäßig nicht eingebaut ist. Wir beziehen uns auf https://tasmota.github.io/docs/Create-your-own-Firmware-Build-without-IDE/#build-the-firmware

Umgebung aufsetzen

```
dnf install python python-virtualenv
pip install --upgrade pip
```

```
cd /home/tomate/FabInfra

virtualenv platformio-core
cd platformio-core
. bin/activate
pip install -U platformio
pip install --upgrade pip
```

```
cd /home/tomate/FabInfra
git clone https://github.com/arendst/Tasmota.git
```

Anpassungen vornehmen

```
vim platform.ini
```

```
; uncomment the following to enable TLS with 4096 RSA certificates
-DUSE_4K_RSA
```

```
#ganz oben
lib_extra_dirs              =
                        ${common.lib_extra_dirs}
                        lib/lib_ssl
```

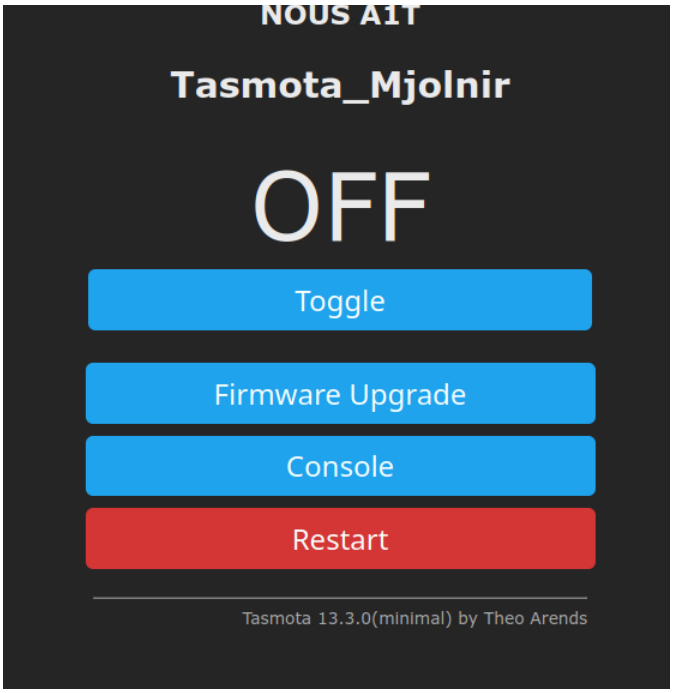
```
vim tasmota/user_config_override.h
```

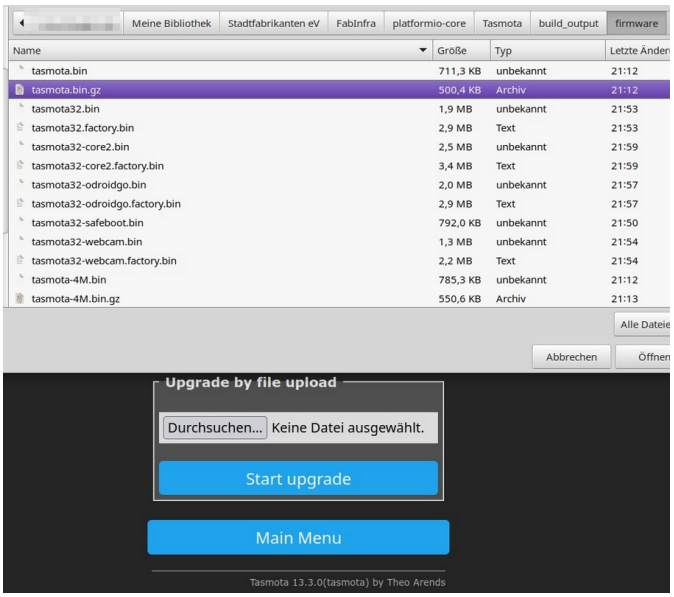
```
#ifndef _USER_CONFIG_OVERRIDE_H_
#define _USER_CONFIG_OVERRIDE_H_

#ifndef USE_MQTT_TLS
#define USE_MQTT_TLS                        // Use TLS for MQTT connection (+34.5k code, +7.0k mem and +4.8k
additional during connection handshake)
#define MQTT_TLS_ENABLED       true             // [SetOption103] Enable TLS mode (requires TLS version)
#define USE_MQTT_TLS_CA_CERT                // Force full CA validation instead of fingerprints, slower, but
simpler to use.  (+2.2k code, +1.9k mem during connection handshake)
                                    // This includes the LetsEncrypt CA in tasmota_ca.ino for verifying server
certificates
//  #define USE_MQTT_TLS_FORCE_EC_CIPHER          // Force Elliptic Curve cipher (higher security) required by
```

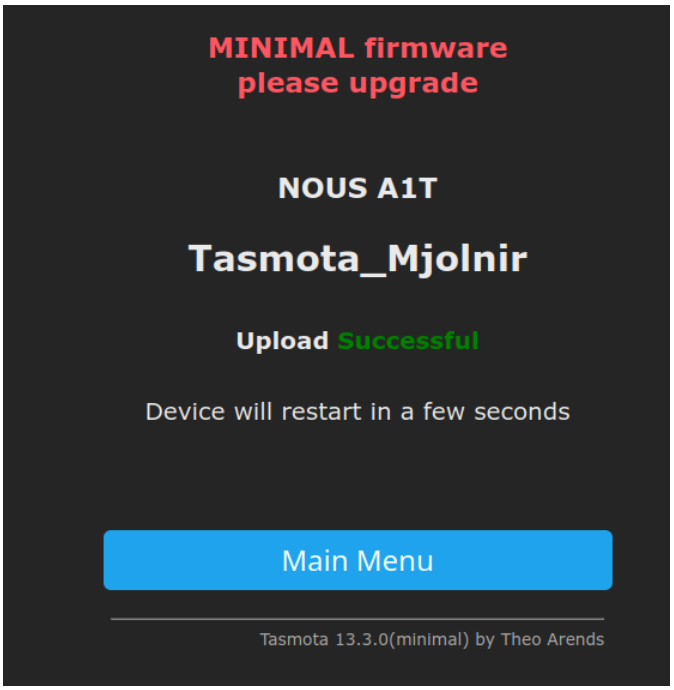
```
some servers (automatically enabled with USE_MQTT_AWS_IOT) (+11.4k code, +0.4k mem)
#endif


#endif  // _USER_CONFIG_OVERRIDE_H_
```

Flashen

Für das Flashen zuerst http://ota.tasmota.com/tasmota/tasmota-minimal.bin.gz aufspielen, dann das custom kompilierte **tasmota.bin.gz**







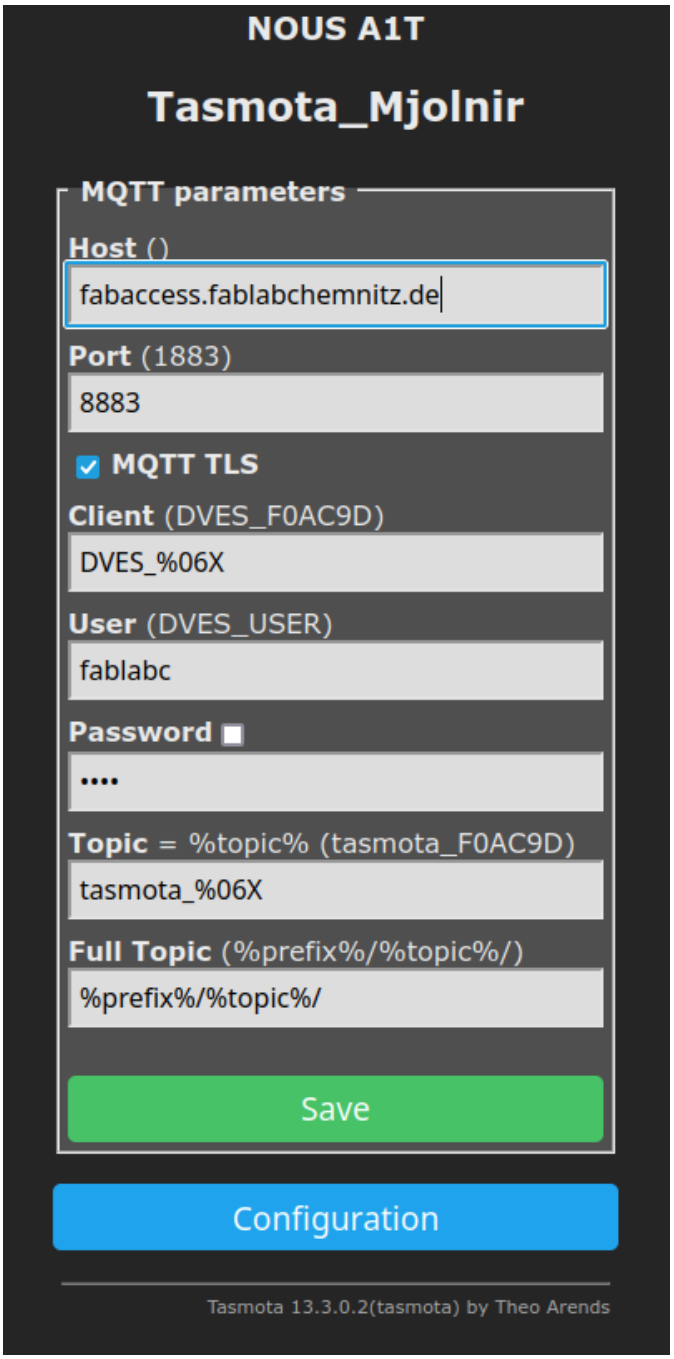
TLS-Verschlüsselung verwenden

Standard ist:

- Host: 192.168.1.192
- Port: 1883

Angepasst ist:

- Host: fabaccess.fablabchemnitz.de
- Port: 8883



Prüfen, ob der DNS-Eintrag klappt und der Port offen ist. Sonst kann keine MQTTS Verbindung aufgebaut werden:

Der DNS-Eintrag ist aktuell im public DNS eingetragen (neycerha) und zeigt auf 192.168.1.192 → das funktioniert nicht, weil wir das DNS der Fritzbox nicht überschreiben können

Für ein LAN wird ein eigener DNS-Server gebraucht, z.B. Unifi DreamMachine oder ein pi-hole.

```
dig fabaccess.fablabchemnitz.de +short
dig @ns2.fablabchemnitz.de fabaccess.fablabchemnitz.de +short
dig @8.8.8.8 fabaccess.fablabchemnitz.de +short

#private DNS Server zuhause
dig @76.76.2.2 fabaccess.fablabchemnitz.de +short
dig @76.76.10.2 fabaccess.fablabchemnitz.de +short
dig @2606:1a40::2 fabaccess.fablabchemnitz.de +short
dig @2606:1a40:1::2 fabaccess.fablabchemnitz.de +short
dig @192.168.1.22 -> empty. Warum? Weil die FritzBox mit einem eigenen Eintrag bereits 192.168.1.192 inne hält und sich nicht überschreiben lässt #fritzbox

telnet fabaccess.fablabchemnitz.de 8883
```

In Tasmota Console:

https://tasmota.github.io/docs/Commands/#setoptions

```
#setzen
SetOption103 1
SetOption132 0

#output prüfen:
SetOption103
SetOption132

#DNS Server 1 "IPAddress4" verändern (testweise)
IPAddress4 192.168.1.22 #default
IPAddress4 8.8.8.8 restart 1

#DNS Server 2 "IPAddress5" verändern (testweise)
IPAddress5 0.0.0.0 #default
IPAddress5 8.8.4.4
IPAddress5 45.136.31.74 restart 1

#mDNS enablen - nur testweise
SetOption55 1
```

Deprecated Fingerprint Methode

Fingerprint erzeugen: https://github.com/issacg/tasmota-fingerprint/releases

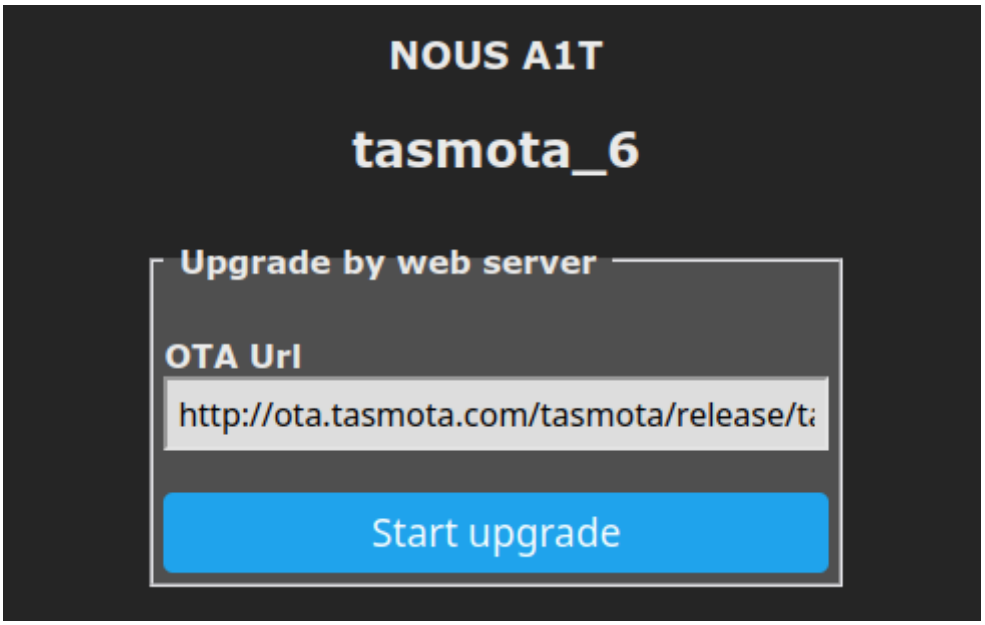
```
cd /opt/
wget https://github.com/issacg/tasmota-fingerprint/releases/download/v1.0.0/tasmota-fingerprint_1.0.0_Linux_armv7.gz
gunzip tasmota-fingerprint_1.0.0_Linux_armv7.gz
chmod +x tasmota-fingerprint_1.0.0_Linux_armv7
./tasmota-fingerprint_1.0.0_Linux_armv7 /etc/ssl/certs/ISRG_Root_X1.pem
```

TLS Fingerprint eintragen und TLS aktivieren (In Tasmota Console):

```
#setzen
SetOption103 1
SetOption132 1
MqttFingerprint F4 EA FC 42 1A 8B 2D 2D 2E 1F 65 21 58 BF D7 3B 35 3F 90 4E

#output prüfen:
SetOption103
SetOption132
MqttFingerprint

#reset Fingerprint:
MqttFingerprint 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

#allow all (Warnung: nicht epfohlen):
MqttFingerprint FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
```

Firmware Upgrade

Das Upgrade kann direkt vom Web Interface aus erledigt werden und funktioniert grundlegend einfach. Das Update dauert ca. 3-5 Minuten und wird bei zwischenzeitlicher Seitenaktualisierung u.U. verschiedene Fehler anzeigen. Diese sind jedoch normal. Zuletzt getestet am 21.11.2024 mit Nous A1T mit Tasmota 12.5.0 (minimal) auf 14.3.0 (minimal).



Weitere Infos finden sich unter https://tasmota.github.io/docs/Upgrading/#decode-config-tool

3D-Druck Einhausung vom MakerSpace Gütersloh

gitlab.com/igami/nous-a1-safebox

Ein kurzes Nous A1T Erklärvideo

Falko Richter von 35 Services e.V. Berlin erklärt grundlegende Schritte und demonstriert, wie man Nous A1T Schaltsteckdosen mit Tasmota Firmware verwendet.

https://www.youtube.com/embed/S6n5mJKozBU

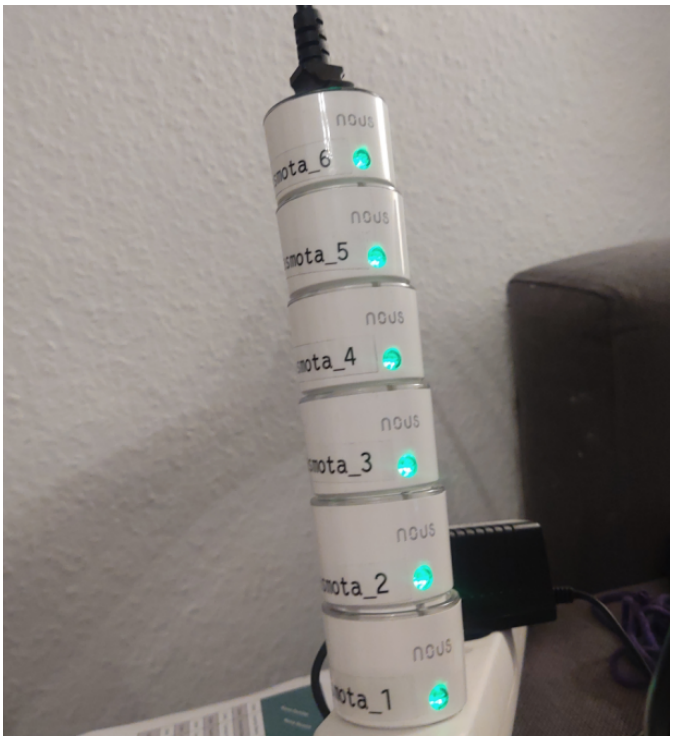
Do's und Dont's

Ein paar Hinweise zur Elektrik.

Nicht Stapeln

Das Stapeln mehrerer Nous Schaltsteckdosen funktioniert in der Praxis, aber ergibt technisch wenig Sinn und ist elektrisch bedenklich, zumal alle Dosen aus gehen, die hinter

der jeweils vorigen Schaltsteckdose klemmen.



Keine Verteilersteckdosenleisten

An die Schaltsteckdosen sollten im Idealfall keine Verteilerleisten angeschlossen werden. Eine Nous A1T kann bis zu 3680 Watt abgeben. Die sind u.U. schnell erreicht. Konzeptionell gesehen ist eine Schaltsteckdose diesen Typs für genau einen Verbraucher ausgelegt. Wenn Du mit einer solchen Schaltsteckdose mehrere Geräte zusammen schalten möchtest, dann empfehlen wir dir einen anderen Typ von Schaltsteckdose - zum Beispiel die Nous A5T. Diese kann bis zu drei Schuko-Geräte aufnehmen und hat zusätzlich USB-Ports.

Shelly 1

Mit dem Shelly 1 können Ressourcen extern über die Unterverteilung freigeschaltet werden. Auch kann der Shelly 1 sehr gut in Geräte eingebaut werden, um so auch mobile Geräte freizuschalten.

Für die verschiedenen Shelly-Produkte gibt es auch 3D-gedruckte Hutschienenhalter.

Steckerschutz und Sicherheitsplomben

Damit FabAccess-angebundene Maschinen in der eigenen Werkstatt auch durch Nutzer sachgemäß genutzt werden und diese nicht an FabAccess "vorbei arbeiten", weil sie die Maschinen einfach aus der Steckdose ziehen und woanders einstecken, gibt es verschiedene Konzepte und Möglichkeiten. Das Manipulieren durch Nutzer ist prinzipiell ein soziales Problem - also ein Vertrauensproblem - kann aber auch unabsichtlich erfolgen. Es ist unter anderem im Werkstatteinsatz von FabAccess zu klären, ob zum Beispiel das Vertauschen von Steckern in Schaltsteckdosen zu ungeahnten Konsequenzen führen könnte. Zum Beispiel könnte ein Gerät unbeabsichtigter Weise an gehen, obwohl es einen Wiederanlaufschutz geben sollte.

Auf dieser Seite werden einige dieser Konzepte und Gedanken vorgestellt und diskutiert. Wir gehen dabei auch auf mögliche Lösungen ein. Grundsätzlich häufig diskutiert wird zwischen den FabAccess Nutzern bw. Implementierern, welcher Schaltaktor der Sinnvollste ist. Je nach Beschaffenheit der Sache ist es manchmal ratsam eine einfache Schaltsteckdose zu nutzen, manchmal macht es mehr Sinn die Maschine zu öffnen und eine Schaltung direkt in die Maschine zu integrieren. Folgefragestellungen ergeben sich daraus jedoch häufig, zum Beispiel:

- was kostet es?
- verlieren wir die Garantie/CE des Geräts?
- ist die Installation und Nutzung sicher?
- ist der angeschlossene Schaltaktor zweckmäßig oder behindert er die Nutzer?
- ...

Maschinen, Handwerkzeuge, Sachen in 230 Volt und Standardstecker

Gewöhnliche Gerätschaften des Werkstattalltags wie Tauchsägen, Trennschleifer, Lötkolben oder einfach nur simple Schreibtischlampen verfügen in der Regel über ein fest angeschlossenes Kabel mit Stecker, oder aber einer Buchse, in welche ein gesondertes Kabel eingesteckt werden kann, um das Gerät beispielsweise portabler oder besser handhabbar zu machen, Manche Geräte haben sehr lange Kabel, andere sehr kurze - zum Beispiel elektrische Kettensägen. Manche Hersteller haben dabei außerdem noch ihr eigenes Steckbuchsensystem entwickelt (z.B. Festool).

Allgemein kennen wir auf der Seite der Steckdose Schukostecker (2-polig mit Schutzleiterkontakt in der Mitte, rund) und Eurostecker (2-polig, ohne Schutzleiter, flacher

Schukostecker und Eurostecker

Auf Seite des Verbrauchers kennen wir verschiedene Buchsen. Recht üblich und beliebt sind hierbei vorallem Kaltgerätestecker (C14 Stecker), der Mickeymouse-Stecker (C5 Stecker) und die Kleingerätebuchse (C7).




Festool Buchse, Kaltgerätestecker (C13), Mickeymouse-Stecker (C5), Kleingerätebuchse (C7)

Ein weiteres Thema bei Steckern und Kabeln sind Verlängerunskabel, die für verschiedene Steckbuchsensysteme entsprechende Kupplungen bereitstellen. Solche Verlängerungskabel sind beliebte und einfache Manipulationsmöglichkeiten für unser FabAccess-System.

Gedanken zur Abwägung: fest oder doch nicht fest?

Grundsätzlich haben alle Stecker das Problem, dass man diese normalerweise einfach herausziehen kann. Das hilft uns bei Verwendung von FabAccess möglicherweise nicht immer. Es ist also aus Sicht des Manipulationsschutzes wünschenswert, wenn der Stecker nicht ohne Aufwand entfernt werden kann bzw. wenn er nicht im laufenden Betrieb locker

wird (was z.B. bei 3D-Druckern fatal ist). Allerdings verschärfen feste Stecker die Problematik, dass Geräte damit schnell unportabel werden oder vielleicht im Fehlerfall nicht schnell genug eine Abschaltung erfolgen kann (z.B. nicht mehr reagierendes Ultraschallbad, was durch Defekt an der Elektronik unbedienbar geworden ist).

Wie lassen sich Stecker schützen?

Eine simple Möglichkeit, Stecker vor dem Herausziehen zu schützen: Festkleben. Das ist allerdings eine unprofessionelle Einweglösung. Eine weitere Lösung: Den Stecker gegen eine feste Verlötung mit der Hauselektrik tauschen. Auch hier stellt sich die Frage, ob das machbar und sinnvoll ist. Eine gute Lösung ist deshalb generell eine reversible. Als guter Rat ist außerdem geboten, den Stecker eventuell an eine versteckte Position zu tun, zum Beispiel unter einen Tisch, in eine Nische, auf einen Schrank - jedoch ist das nicht in jedem Falle gewinnbringend oder kann einfach irritieren.

Nous A1T Safebox (Anti Tamper Seal) von Michael Prange

Ein Manipulationsschutz für Schaltsteckdosen vom Typ Nous A1T oder baugleich. Er besteht aus 2 Einzelteilen und wird dann mit einem Kabelbinder oder einer Sicherheitsplombe mit eindeutiger Kennzeichnung verschlossen. Die "Safebox" ist so schmal, dass sie in Steckdosenleisten in der Regel nicht stört bzw. kollidiert (zwei mit Safebox versehene Stecker können allerdings zumindest nicht direkt nebeneinander in einer gewöhnlichen Mehrfachsteckdosen angeordnet sein!). Durch die Geometrie können alle üblichen Schuko-Stecker benutzt werden - egal ob gerade oder gewinkelt. Auch schmale, 2-adrige Kabel funktionieren. Die Dateien (FreeCAD und STL) stehen zum Download bereit und können selbst per FDM 3D-Drucker angefertigt werden. Das Design ist bewusst offen gehalten, um Wärmestau zu unterbinden.

Wer beispielsweise auch beim Einsatz von Kabelbindern das Manipulieren erschweren will, der kann zum Beispiel Kabelbinder mit Sonderfarbe in der Werkstatt einsetzen, welche nicht von jedem Nutzer einfach so ausgetauscht werden können. Wie wäre es zum Beispiel mit Kabelbindern in Corporate Identity Farbe? Die Safebox selbst kann ebenso in dieser Farbe gedruckt werden, wenn ein geeignetes Filament verwendet wird, zum Beispiel in petrolgrün von 3DK.Berlin.

Kompatibel / getestet außerdem mit:

- NEO Coolcam Z-Wave Plus Smart Power Plug 2500 W

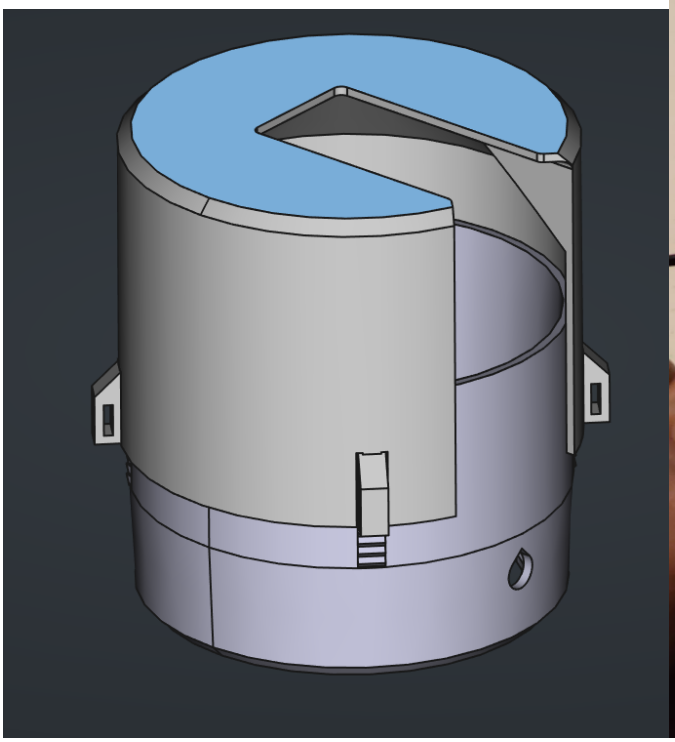
- wahrscheinlich auch mit GHome Gosund Smart Wi-Fi Plug EP2 (SP111) Schaltsteckdose kompatibel

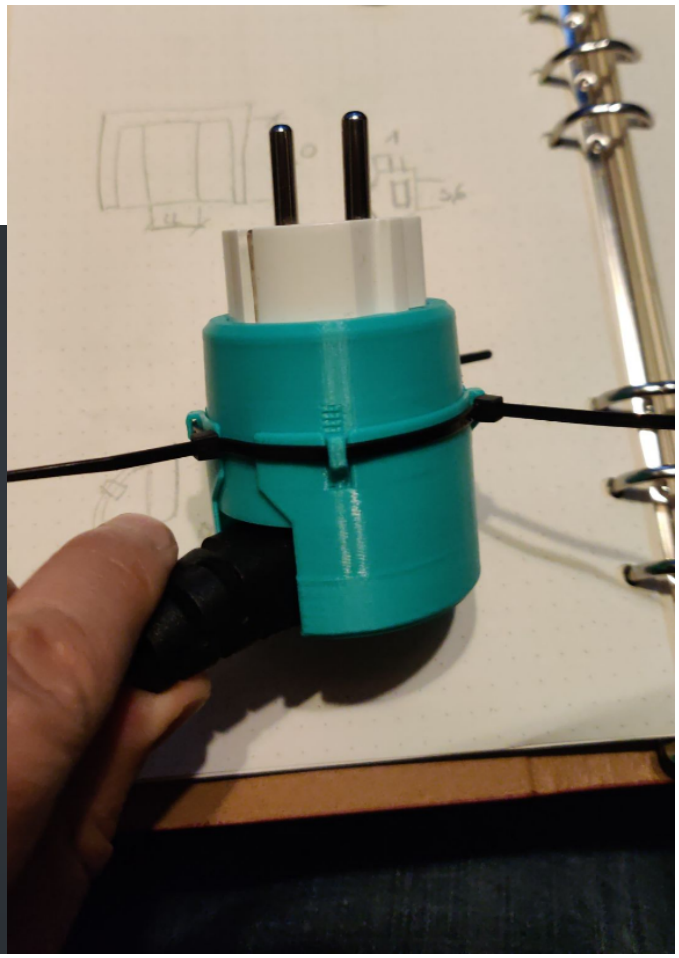
Quelldateien:

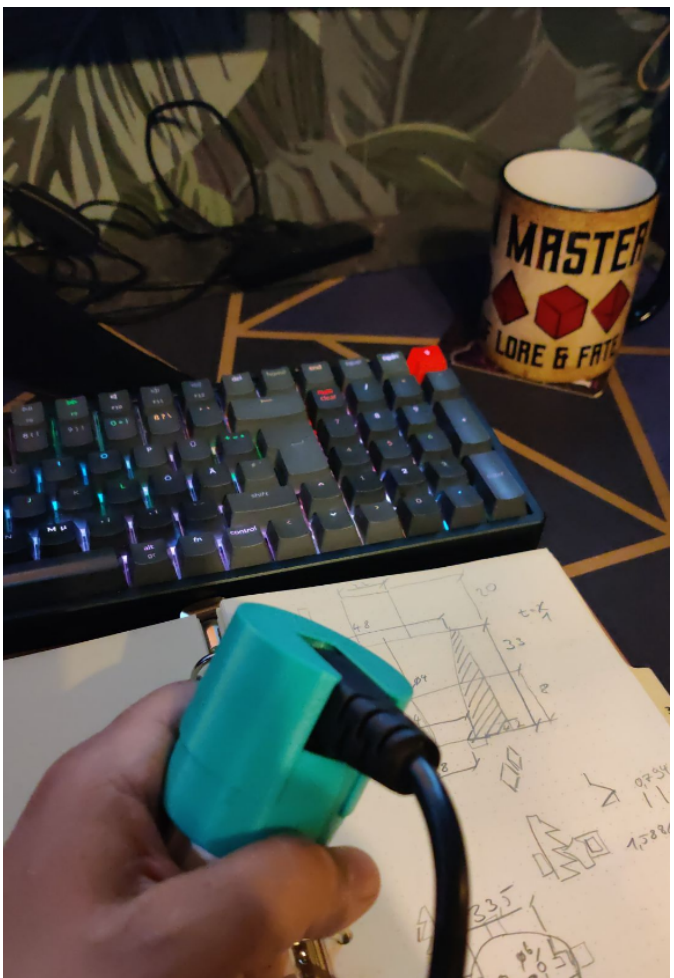
- https://gitlab.com/igami/nous-a1-safebox, oder
- https://hardware.fab-access.org/models?collection=2rqRJ4Jw (Mirror)

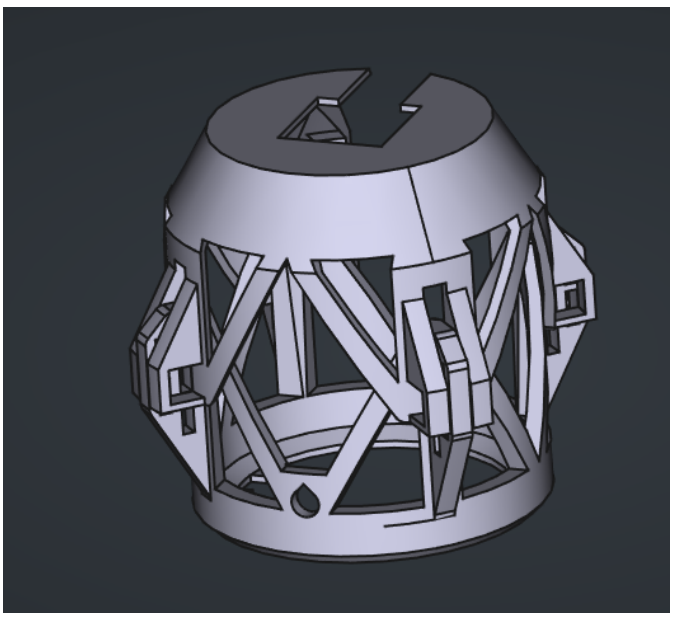
Siehe auch: https://forum.makerspace-gt.de/t/fabaccess-maschinenrechtverwaltung/854

Beispielfotos:






MASTER
LORE & FATE



> **Aber das Design der Safebox hat ein Loch beim Knopf. Ist das nicht unsicher?**

Ja das ist es prinzipiell! Aber nur, wenn die darin eingebaute Schaltsteckdose so konfiguriert ist, dass jeder den Knopf drücken kann! Das lässt sich z.B. bei Tasmota-basierten Schaltsteckdosen durch Konfiguration ändern! Siehe Nous A1T mit Tasmota Firmware für Details.

Andere Vorschläge:

- drehe die Schaltsteckdose um 180°, sodass der Knopf auf der anderen Seite gefangen ist und nicht gedrückt werden kann. Nachteil: Die LED wird verdeckt
- schließe das Loch im 3D-Design und drucke das Gehäuse transparent. Dann erkannt man den Status auch durch das Gehäuse hindurch

Kabelbinder versus Sicherheitsplomben

Sicherheitsplomben sind teurer, aber auch professioneller als Kabelbinder, denn sie haben eine eindeutige Kennzeichnung. Wer diese in der Werkstatt einsetzt, sollte emfehlenswerterweise ein Register führen, welche Maschine bzw. welcher Schaltaktor mit welcher Plombenkennzeichnung versehen wurde und wann. Außerdem sollte in diesem Falle auch im Prozess geklärt sein, wer diese Plomben aushändigt und montiert, damit niemand anderes solche Plomben in der Werkstatt einschmuggelt.

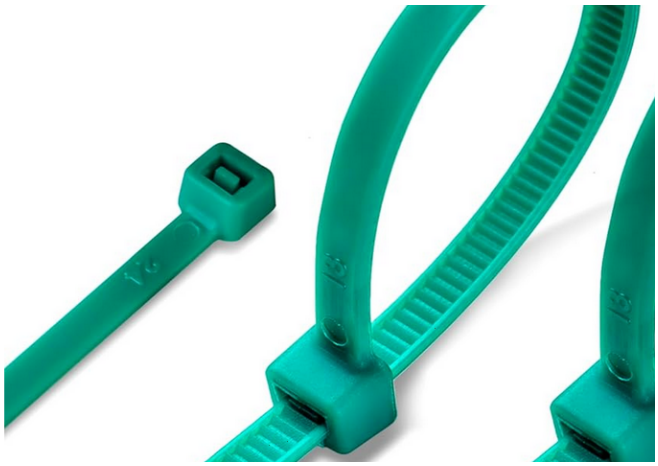


Foto von Sicherheitsplomben und von Kabelbindern in "FabAccess Farbe"

Wie lassen sich Buchsen schützen?

Ähnlich zur Safebox von Michael Prange lassen sich auch Konzepte auf Buchsenseite umsetzen. Zunächst gibt es beispielsweise für C13-Kaltgerätebuchsen Varianten, die ein einfaches Herausziehen erschweren, indem ein spezieller Hebel gedrückt werden muss. Diese sind dafür ausgelegt, dass das Gerät nicht unkontrollierten Stromausfall erleidetet und werden häufig im IT-Sektor angewendet (Fallbeispiel: Putzkraft im Serverraum stolpert über ein Kabel und alles geht spontan aus). Es gibt verschiedene Hersteller solcher Spezialkabel wie beispielsweise PROCOM mit ihrem Produkt "IEC Lock":

https://www.youtube-nocookie.com/embed/eUHOjCzW2mQ?si=-YnBK1fwuhkdomCI

> ToDo: Finde eine mechanische Lösung, um das Kabel mit Sperriegel vor dem Entsperren zu Sichern (analog zur Nous A1T Safebox) - zum Beispiel durch Klemmen, Schrauben, Plomben, Kabelbinder, ...

Weitere Möglichkeiten des Manipulationsschutzes

Neben der Safebox besteht auch die Möglichkeit, ganze Schaltsteckdosen oder die Verbindungsstellen von Verlängerungskabeln in gesonderte Gehäuse einzubetten, wie im

folgenden Beispiel einer wasserdichten Kabelbox:

Shelly 1 Plus

Mit den Shelly Plus Modellen hat Shelly eine neue Generation von schaltbaren WLAN-Steckdosen und Adaptern veröffentlicht. Diese unterscheiden sich durch die Gen2-API und müssen daher anders angesteuert werden.

Shelly Plug S

Der Shelly Plug S ist das Äquivalent zum Nous A1T. Er kann jedoch weniger Strom als Dauerlast schalten.

Shelly Plug 1

Mit dem Shelly Plug 1 können bis zu 16A Dauerlast geschaltet werden.

Tipps zu Shelly Generationen

https://support.shelly.cloud/de/support/solutions/articles/103000044249-was-ist-der-unterschied-zwischen-ger%C3%A4ten-der-generation-1-alte-generation-und-der-generation-2-neu

> Der Hauptunterschied zwischen den Geräten der Gen. 1 und der Gen. 2 ist der Prozessor. Gen. 1 verwendet ESP8266 und Gen. 2 verwendet ESP32. Die Geräte der Gen. 2 (auch New-GEN genannt) haben mehr Speicher und einen schnelleren Chip. Sie verfügen über mehr Funktionen und Merkmale und unterstützen außerdem BLE (derzeit nur Inklusion), benutzerdefinierte Skripte, sichere Anfragen (HTTPS und MQTTS) und eine Wi-Fi Range Extender Option.

Es wird empfohlen, Shellys der Generation 2 zu verwenden, da diese natives TLS mit MQTT bieten.

Gosund EP2