27.12.2023 // FabAccess FAQ

FabAccess Tasmota Actor konfigurieren

Actor verfügbar machen

```
git clone https://gitlab.com/fabinfra/fabaccess/actors/tasmota.git adapters/tasmota
chmod +x adapters/tasmota/main.py
```

Actor in bffh.dhall konfigurieren

```
    actors =
    {
        tasmota_F0A4FB =
        {
            module = "Process",
            params =
            {
                cmd = "/usr/local/lib/bffh/adapters/tasmota/main.py",
                args = "--host mqtt --tasmota F0A4FB"
            }
        }
    },
```

Noua A1T / Tasmota konfigurieren

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

PowerOnState auf 0 setzen, um eine Art Wiederanlaufschutz zu haben

> PowerOnState Control power state when the device is powered up. More information
> 0 / OFF = keep power(s) OFF after power up
> 1 / ON = turn power(s) ON after power up
> 2 / TOGGLE = toggle power(s) from last saved state
> 3 = switch power(s) to their last saved state (default)
> 4 = turn power(s) ON and disable further power control
> 5 = after a PulseTime period turn power(s) ON (acts as inverted PulseTime mode)

Ggf. FriendlyName auf etwas lesbares ändern

!! Fuktioniert das noch mit dem Tasmota Actor? !!

> FriendlyName<x> 1 = Reset friendly name to firmware default
> <value> = set friendly name (32 char limit)

Tasmota aktualiseren

> Upgrade 1 = download firmware from OtaUrl and restart
> 2 = (ESP32 only) download safeboot firmware based on OtaUrl and restart into safeboot
> <value> = download firmware from OtaUrl if <value> is higher than device version

Werbserver ausschalten und konfiguration nur über MQTT machen

!! Ist dann noch ein separates Netzwerk erforderlich? !!

> Webserver 0 = stop web server
> 1 = start web server in user mode
> 2 = start web server in admin mode

WiFi einrichten

> SSId<x> <x> = 1…2
> <value> = set AP<x> Wi-Fi SSID and restart
> 1 = reset AP<x> Wi-Fi SSID to firmware default (STA_SSID1 or STA_SSID2) and restart
> SSID are limited to 32 characters. Do not use special characters or white spaces in the SSID

> Password<x> <x> = 1…2
> <value> = set AP<x> Wi-Fi password and restart
> 1 = reset AP<x> Wi-Fi password to firmware default (STA_PASS1 or STA_PASS2) and restart
> Passwords are limited to 64 characters. Do not use special characters or white spaces in the password.
> Note that Password and Password1 are equivalent commands.

MQTT einrichten

> MqttHost 0 = clear MQTT host field and allow mDNS to find MQTT host
> 1 = reset MQTT host to firmware default (MQTT_HOST) and restart
> <value> = set MQTT host and restart (do NOT use .local)

> MqttUser 0 = clear MQTT user name
> 1 = reset MQTT user name to firmware default (MQTT_USER) and restart
> <value> = set MQTT user name and restart

> MqttPassword 0 = clear MQTT password
> 1 = reset MQTT password to firmware default (MQTT_PASS) and restart
> <value> = set MQTT password and restart (min 5 chars)

>

> MqttPort 1 = reset MQTT port to firmware default (MQTT_PORT) and restart
> <value> = set MQTT port between 2 and 32766 and restart

MQTT mit TLS

mosquitto mit traefik einrichten

https://medium.com/@synoniem/running-mosquitto-mqtt-server-with-docker-traefik-and-lets-encrypt-a1f6cbb864cc

ggf. noch mit crowdsec absichern

https://goneuland.de/traefik-v2-3-reverse-proxy-mit-crowdsec-im-stack-einrichten/

ESP8266 unterstützen kein TLS mit Tasmota und muss selbst gebaut werden

https://tasmota.github.io/docs/TLS/#compiling-tls-for-esp8266
!! Durch FabInfra bereitstellen, damit neue Nous per OTA geflashes werden können? !!

platform.ini

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

platformio_tasmota_env.ini

```
lib_extra_dirs              =
                              ${common.lib_extra_dirs}
                              lib/lib_ssl
```

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_
```

Backup / Restore einrichten

Hardware empfehlung

Automatisches Deployment via Ansible?

Hilfsmittel

Links

https://gitlab.com/fabinfra/fabaccess/dockercompose

https://fab-access.readthedocs.io/en/v0.3/installation/server_docker.html

MQTT Explorer

https://mqtt-explorer.com/

Version #1
Erstellt: 2024-10-14 16:09:13 CEST von Mario Voigt (Stadtfabrikanten e.V.)
Zuletzt aktualisiert: 2025-02-25 21:22:13 CET von Mario Voigt (Stadtfabrikanten e.V.)