## ESP8266 avec EspEasy et Jeedom

## Installation

#### 1 <u>Hardware</u>

Carte NodeMCU Lua Lolin v3 Module ESP8266 (ESP-12F)



Taille de la carte: 5,5cm x 3cm Serial: Rx & Tx Flash: 4MB

#### 1.1 GPIO on ESP8266

GPIO Number	GPIO Name	Boot State	Precautions and information	
0	D3	HIGH for boot LOW for serial programming	Pullup resistor on most boards	
1	Тх		Used as serial (TX) May be used but observe that signal might flicker due to serial activity during boot.	
2	D4	HIGH for boot HIGH for serial programming	Connected to onboard LED (low active) Used as serial1 (TX1)	
3	Rx		Used as serial (RX) May be used but observe that signal might flicker due to serial activity during boot.	
4	D2		Generally used as SDA (I2C)	
5	D1		Generally used as SCL (I2C)	
6			Do not use! Reserved for SPI + flash	
7			Do not use! Reserved for SPI + flash	
8			Do not use! Reserved for SPI + flash	
9			Use with caution! Reserved for SPI + flash	
10			Use with caution! Reserved for SPI + flash	
11			Do not use! Reserved for SPI + flash	
12	D6			
13	D7			
14	D5			
15	D8	LOW for boot	Pulldown resistor on most boards	
16	D0		No PWM (counter not possible). No internal pullup available. Used in sleep mode for wakeup	

#### 2 Software

- Flash de la carte avec

ESP\_EspEasy\_ESP82xx\_mega\_20201022\_normal\_ESP8266\_4M1M.bin Origine du software: <u>https://github.com/letscontrolit/ESPEasy/releases</u>

- Nécessite un MQTT broker.

Utilisation de mosquitto qui est disponible sur Debian apt install mosquitto apt install mosquitto-clients (optionnel) Utilisation de la configuration par défaut.

Attention: pas de user/password requis pour les connexions. Les échanges sont en clair et accessible à tout un chacun.

- Pour suivre les messages MQTT, utilisation du software mqtt explorer url: <u>http://mqtt-explorer.com/</u>
- Sur Jeedom, installation du plugin MQTT de Lunarok

## Sécurisation des échanges MQTT

Si l'on veut sécuriser les échanges MQTT, il faut que le broker nécessite une authentification pour accepter les connexions.

Malheureusement la combinaison username, password est transmise en clair. Idéalement, Il faudrait que la connexion soit protégée par ssl.

Toute la configuration de sécurisation est effectuée dans le broker MQTT. Une fois celle-ci en place, les clients doivent s'y conformer pour se connecter, souscrire et publier.

Procédure 1) Créer un fichier user, password nommé passwordfile Celui-ci est de la forme

> user1:password1 user2:password2

....

2) Encrypter les password présents dans le fichier. Cela s'effectue avec la commande mosquitto\_passwd qui est disponible dans le paquet mosquittoclients.

Exécuter la commande

mosquitto\_passwd -U passwordfile

Le fichier passwordfile contient toujours les password mais ils sont encryptés.

3) Déplacer le fichier dans le directory de configuration mosquitto mv passwordfile /etc/mosquitto/passwordfile

- 4) Modifier le fichier de configuration de mosquitto vi /etc/mosquitto/mosquitto.conf
- Ajouter les lignes suivantes avant la ligne include-dir allow\_anonymous false password\_file /etc/mosquitto/passwordfile
- 5) Recharger la configuration de mosquitto
- a) trouver le PID du programme mosquitto
  - ps -ef | grep mosquitto
- b) Recharger la configuration de mosquitto sudo kill -HUP <PID>

6) Configurer EspEasy et Jeedom pour utiliser cette authentification.

Cette procédure est extraite du post suivant http://www.steves-internet-guide.com/mqtt-username-password-example/

Sécurisation par SSL/TLS

Cette configuration est relativement lourde. Elle nécessite une Certificate Authority. Une personelle peut convenir.

http://www.steves-internet-guide.com/mosquitto-tls/

### **Installation ESPEasy**

Sur machine windows, lancer le programme ESP.Easy.Flasher. Installer les deux fichiers de polices de caractères. Attention: Run with elevated privilege (admin)

J'ai dû désinstaller l'antivirus pour faire fonctionner ce programme.

	Connecter le module Esp sur u	une porte USB du PC.	(Alimentation 5V p	ar le PC)
--	-------------------------------	----------------------	--------------------	-----------

	SP Easy Flash	her (0.04.007)	×	Domo: pas do porto ush
	COM port	No online COM ports found! ~	0	
	⊡Only active ports Firmware (.bi	n) AESP_Easy_mega_20201022_normal_ESP8266_4MIM.bin ~	Open folder	
		Baud rate 115200 🗸 🗆 Force -DOUT	Update	
-	⊡ <u>P</u> o Name UNIT espn	nsg I ash action Pause after flash (manual reboot) Number Admin password 1 xxxxxx		
	WiFi S	SID (main) WiFi password (main) □ Fixed IP		Préciser les infes de post flach
	WIH fullo:	52hdax xxxxxxxx 192 168 U 123		Preciser les infos de post flash
		n custom serial commands (Settinos\SerialCommands txt)	only.	<u>.</u>
	Rulest	EMPTY FOLDER ~	Open	
	Rules2	EMPTY FOLDER ~	Open	
	Rules3	EMPTY FOLDER ~	Open	
P	Rules4	EMPTY FOLDER ~	Open	3
	No active 1 (USB) C Please ci AGENCYR AGENCYB	e (USB) COM ports found! OM ports found. opy: .TTF .TTF		Pour flasher
	✓ <u>S</u> ave a	is default settings Open serial monitor 🖌 🖌 <u>H</u> ash ESP Easy	y FW	

Après le flash, utiliser le bouton Reset sur la carte pour la redémarrer.

Chercher sur l'adresse IP prise par la carte.

S'y connecter avec un browser.

Le user est "admin" avec le password spécifié en post flash.

## **Configuration EspEasy**

La documentation de EspEasy est accessible à l'url: <u>https://espeasy.readthedocs.io/en/latest/</u>

1 <u>Config</u>		
ESP Easy Mega: esp	omsg	
∆Main <b>≉Config</b> ్ల	Controllers <i>≸</i> Hardware <b>∢</b> Devices ⇔Rules ⊠Notifications <b>∢</b> Tools	
Main Settings		
Unit Name:	espmsg	
Unit Number:	1 3	
Append Unit Number to hostname:		Mettre un numéro différent pour chaque esp
Admin Password:	••••	
Wifi Settings		
SSID:	full852hdax	
WPA Key:	•••••	
Fallback SSID:		
Fallback WPA Key:		
WPA AP Mode Key:		
Client IP filtering		
Client IP block level:	Allow Local Subnet	
Access IP lower range:	192.168.11.0	
Access IP upper range:	192.168.11.255	
WiFi IP Settings		
ESP WiFi IP:		
ESP WiFi Gateway:		Conseil: Spécifier une adresse IP fixe
ESP WiFi Subnetmask:		
ESP WiFi DNS:		
	Note: Leave empty for DHCP	
Sleep Mode		
Sleep awake time:	0 (sec] (?	
	Note: 0 = Sleep Disabled, else time awake from sleep	
Sleep time:	60 : [sec (max: 14856)]	
Sieep on connection failure:		
	Submit	

# 2 <u>Advanced config</u> Accessible via Tools/Advanced

ESP Easy Mega: esp	nsg	
∆Main <b>∗</b> Config ⊝C	ontrollers <b>⊀</b> Hardware <b>≹</b> Devices :→Rules <sup>I</sup> Notifications <b>₹</b> Tools	
Advanced Settings (1)		
Rules Settings		J
Hules: Old Engine:		Pour avoir rules
Tolerant last parameter:		
SandToHTTD wait for activ	Note: Perform less strict parsing on last argument of some commands (e.g. publish and s	
Send IOT I IP Walt for ack:		
NTP Settings		
NTP Hostname:	3.be.pool.ntp.org	ntp config
		1
DST Settings		
Start (week dow month):	- Sun	
Start (Week, uow, month):	Mar ,	
Start /localtime a c 05 01	2 hourst	
oran (iocanime, e.g. 2n→3h):		
	Last	
⊏па (weeк, aow, month):	• Oct	
	•	
End (localtime, e.g. 3h→2h):		
091:		
Location Settings		
Timezone Offset (UTC +):	60 c) [minutes]	
Latitude:	0.000000 : [°]	Timezone offset: UTC +1
Longitude:	0.000000 : [°]	L
	Note: Longitude and Latitude are used to compute sunrise and sunset	
Log Settings		
Syslog IP:		
Syslog UDP port:	<b>514</b> ©	
Syslog Log Level:	None	
Syslog Facility:	Kernel	
Serial Log Level:	Info •	
Web Log Level:	Info -	
Carial Cattings		
Enable Serial port:		
Baud Rate:	115200 :	
Inter-ESPEasy Network		
ODP port:		
Special and Experimental Se	ttings	
Webserver port:	80 0	
	Note: Requires reboot to activate	
Fixed IP Octet:	0	
WD I2C Address:	0 (decimal)	
I2C ClockStretchLimit:	<b>0</b> ©	
Use SSDP:		
Connection Failure Threshold:	<b>0</b> $\odot$	
Force WiFi B/G:		
Hestart WIFI Lost Conn: Force WiFi No Sleep:		
	Note: Change WiFi sleep settings requires reboot to activate	
Periodical send Gratuitous ARP:		
CPU Eco Mode:		
	Note: Node may miss receiving packets with Eco mode enabled	
	Submit	

#### 3 **Controllers**

Pour dialoguer avec Jeedom, il faut indiquer ou se situe les MQTT broker. C'est le rôle du Controller.

ESP Easy Mega: es	omsg				
∆Main *Config 🤅	Controllers Alardware	es :→Rules	INNOTIFICATIONS	Tools	
Controller Settings					
Protocol:	Home Assistant (openHAB) MQTT		•	?	
Locate Controller:	Use IP address		•		
Controller IP:	192.168.11.78			$\leq$	IP du broker mosquitto
Controller Port:	1883 0				
Controller Queue					
Minimum Send Interval:	100 c [ms]				
Max Queue Depth:	<b>10</b> °				
Max Betries:	10				
Full Queue Action:					
			•		
Check Reply:	Ignore Acknowledgement		•		
Client Timeout:	100 c [ms]				
Credentials					
Use Extended Credentials:					
Controller User:					
Controller Password:					
MOTT					
Controller Client ID:	%sysname% %unit%				
Unique Client ID on					
Reconnect: Current Client ID:	espmsg_1_1				
	Note: Updated on load of this page				
Publish Retain Flag:					
Controller Subscribe:	%sysname%/#				
Controller Publish:	%sysname%/%tskname%/%valname%				/// %tskname% correspond à "name"
Controller LWT Topic:					dans devices et %valname% au nom
LWT Connect Message:				de la variable.	
LWT Disconnect Message:					
Send LWT to broker:					
Will Retain:					
Enabled:					
	Close Submit				

Jeedom n'est pas reconnu en tant que tel comme Controller mais celui pour "Home Assistant MQTT" fonctionne très bien.