

ESP-NOW to MQTT gateway PEER-TO-NODERED

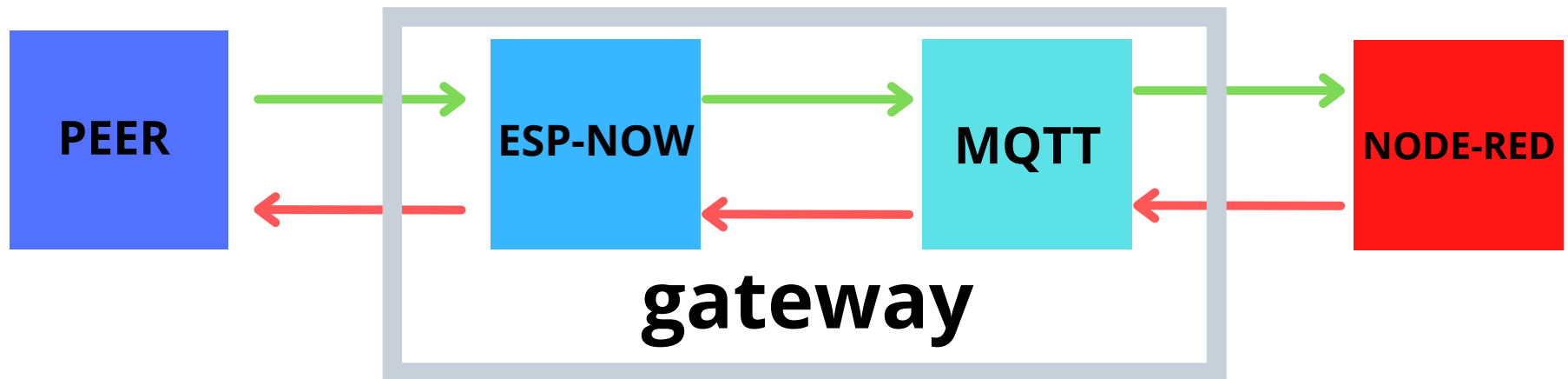
packet = topic | {message}

len = packet.length()

topic | {message}

\$\$MAClenpacket

{MAC:,message{}}



- Peer send message and topic to publish
- ESP-NOW pack MAC, topic and message and transmit it
- MQTT unpack full message and create topic. Send MAC & message content
- NODE-RED receive MAC & message content

ESP-NOW to MQTT gateway

NODE-RED TO PEER

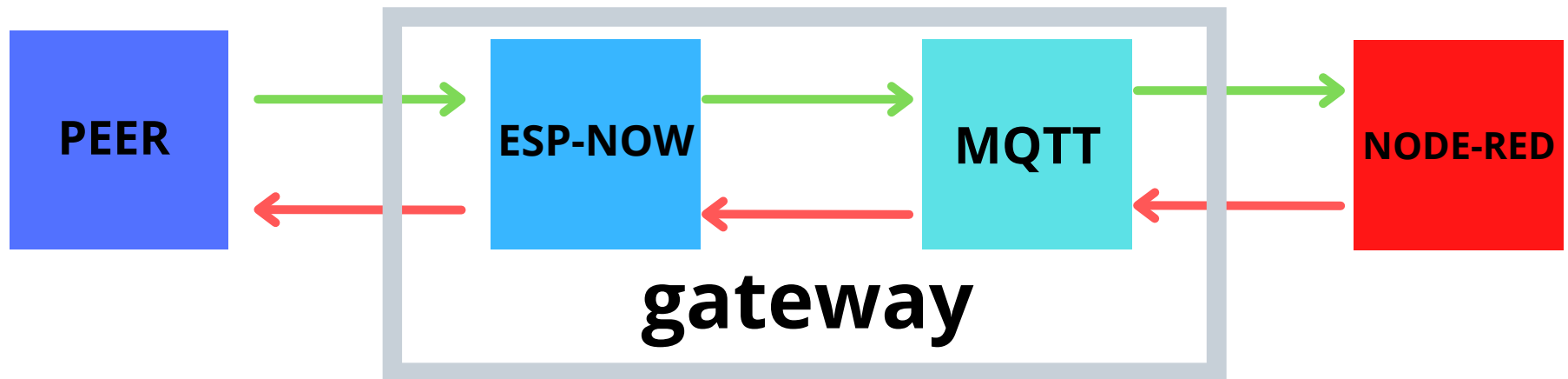
packet = topic | {message}

len = packet.length()

topic | {message}

\$\$MAClenpacket

{MAC;topic;message{}}



- NODE-RED send MAC,topic and message content (whatever)
- MQTT unpack full message and create packet. Convert MAC (HEX2Byte) and send it by UART.
- ESP-NOW read MAC, topic and message and transmit it to peer.
- Peer do something cool

ESP-NOW to MQTT gateway

Keep-in-mind

- Peer and ESP-NOW are sender/receiver.
- ESP-NOW needs to know WiFi MAC of peer and must be in AP mode. In peer side (if you're using ESP8266), looks like this:

```
//Put WiFi in AP mode.  
WiFi.mode(WIFI_AP);  
wifi_set_opmode(STATIONAP_MODE);
```

- ESP-NOW needs to know if the peer is in AP mode or whatever. We specify this when we declare the peer (in ESP32):

```
//adding normal friends  
memcpy(peer2.peer_addr, mac_peer2, 6);  
peer2.channel = WIFI_CHANNEL;  
peer2.encrypt = 0;  
peer2.ifidx = ESP_IF_WIFI_AP; |
```

ESP-NOW to MQTT

gateway

Keep-in-mind

- ESP-NOW can have a custom MAC address, and peer needs to know it. Will be the gateway MAC, associated with the gateway. Code in ESP-NOW (ESP32):

```
//Ponemos el WiFi en modo AP.  
WiFi.mode(WIFI_AP);  
//Establecemos la MAC para esta ESP  
esp_wifi_set_mac(ESP_IF_WIFI_AP, gatewayCustomMac); // esp32 code  
Serial.print("MAC: "); Serial.println(WiFi.softAPmacAddress());
```

- Code in peer (ESP8266):

```
//Especificamos el rol del ESP-NOW (0=OCIOSO, 1=MAESTRO, 2=ESCLAVO y 3=MAESTRO)  
esp_now_set_self_role(ESP_NOW_ROLE_CONTROLLER);  
//Emparejamos con el otro ESP  
esp_now_add_peer(macPasarela, ESP_NOW_ROLE_CONTROLLER, WIFI_CHANNEL, key, 0);  
//Especificamos las claves para enviar la información de manera encriptada.  
esp_now_set_peer_key(macPasarela, key, 16);
```