

# Systemy monitoringu IoT

## Protokół integracyjny MQTT



<b>Nazwa dokumentu:</b>	Protokół MQTT
<b>Wersja:</b>	1.1
<b>Data wydania:</b>	11.01.2021
<b>Status:</b>	Aktualny

Niniejszy dokument zawiera zastrzeżone informacje techniczne, które są własnością firmy SYNGEOS. Kopiowanie tego dokumentu i przekazywanie jej zawartości innym firmom są zabronione bez wyraźnej zgody właściciela. Wszystkie dostarczane specyfikacje mogą zostać zmienione w dowolnym czasie.

## Punkt dostępu MQTT

Do integracji z systemami zewnętrznymi na serwerze IoT dostępny jest interfejs MQTT (Message Queue Telemetry Transport). System wspiera protokół MQTT w wersji 3.1.1.

Punkt dostępu MQTT jest dostępny pod adresem:

`mqtt://iot-endpoint.syngeos.pl:8883/`

## Autoryzacja i uprawnienia

Dostęp do interfejsu możliwy jest po zainicjowaniu połączenia szyfrowanego z następującymi parametrami:

- protokół szyfrowania: TLS v1.2 z PSK (pre-shared key)
- identyfikator PSK: `iot`
- klucz PSK: `b4c8540f5f4329352e5460a79c8195cd`
- `clientId` - unikalny identyfikator klienta (patrz niżej)

### `clientId`

Klient powinien identyfikować się w brokerze MQTT parametrem `clientId` w formacie

`urn:{namespace}:{id}`

gdzie:

- `{namespace}` jest przestrzenią nazw identyfikującą typ urządzenia. Należy używać stałej wartości `im40`.
- `{id}` jest unikalnym identyfikatorem urządzenia w ramach przestrzeni nazw. Identyfikator ten powinien być ciągiem maksymalnie 16 znaków i być na stałe przypisany do urządzenia, np. numer IMEI - *International Mobile Equipment Identity*.

Przykładowa wartość `clientId`: `urn:im40:123456781234567`

Pole `clientId` identyfikuje klienta w brokerze MQTT, ale nie w serwerze. Serwer do identyfikacji używa pól `"namespace"` i `"imei"` w komunikacie (patrz niżej).

## Komunikaty MQTT

Klient powinien wysyłać do brokera MQTT komunikaty używając metody PUBLISH. Serwer IoT nasłuchuje komunikatów przychodzących na temat `"iot/sensor"`. Komunikaty przychodzące na inne tematy będą ignorowane przez serwer.

Przykład użycia klienta Eclipse Mosquitto do wysłania komunikatu (wartość `{json_body}` należy zastąpić komunikatem w wymaganym formacie):

```
mosquitto_pub -h iot-endpoint.syngeos.pl -t "iot/sensor" -m
"{\"version\":\"1.0.0\", \"namespace\":\"im40\", \"imei\":\"1234567812
34567\", \"ssn\":\"26012345678912345\", \"latitude\":52.3007, \"longitu
de\":16.8204, \"gpsFix\":true, \"signalRssi\":-
63.9, \"status\":\"OK\", \"info\":\"\", \"sensors\": [{\"code\":\"pm2.5\"
```

```
"\", \"value\":73.45},{\"code\":\"pm10\", \"value\":57.51}}]" -p 8883 -
-psk-identity iot --psk b4c8540f5f4329352e5460a79c8195cd -i
urn:im40:123456781234567
```

Klient powinien wysyłać do brokera MQTT komunikaty w następującym formacie JSON:

```
{
  {
    "version": "1.0.0",
    "namespace": "im40",
    "imei": "123456781234567",
    "ssn": "26012345678912345",
    "latitude": 52.3007,
    "longitude": 16.8204,
    "gpsFix": true,
    "signalRssi": -63.9,
    "status": "OK",
    "info": "",
    "sensors": [
      {
        "code": "pm2.5",
        "value" : 73.45
      },
      {
        "code": "pm10",
        "value" : 57.51
      }
    ]
  }
}
```

Znaczenie poszczególnych pól w komunikacie opisano w tabeli poniżej.

Nazwa	Opis	Przykładowa wartość	Pole obowiązkowe
version	Numer wersji firmware urządzenia. Należy używać wersjonowania semantycznego 2.0.0 w formacie: MAJOR.MINOR.PATCH	"1.0.0"	tak
namespace	Przestrzeń nazw identyfikującą typ urządzenia. Należy używać stałej wartości <i>im40</i> . Komunikaty o innych wartościach zostaną odrzucone	"im40"	tak
imei	Identyfikator urządzenia IMEI (International Mobile Equipment Identity). Ten	"123456781234567"	tak

	sam numer musi być również podany w clientId		
ssn	Numer seryjny karty SIM (SIM Serial Number)	"1234567890123456789"	tak
latitude	Szerokość geograficzna w stopniach (wartości dodatnie dla półkuli północnej, ujemne dla południowej)	52.3007	nie
longitude	Długość geograficzna w stopniach (wartości dodatnie dla półkuli wschodniej, ujemne dla zachodniej)	16.8204	nie
gpsFix	Czy GPS odbiera sygnał (FIX)	true	nie
signalRssi	Poziom sygnału RSSI w dBm	-63.8	tak
status	Status urządzenia. Musi przyjąć jedną z wartości: <ul style="list-style-type: none"> <li>OK - urządzenie działa poprawnie</li> <li>DEVICE_ERROR - urządzenie nie działa poprawnie, dane z czujników zostaną zignorowane</li> <li>SENSOR_ERROR - co najmniej jeden czujnik nie działa poprawnie</li> </ul>	"OK"	tak
info	Dodatkowa informacja do celów diagnostycznych. W polu można umieścić opis błędu, jeżeli status jest różny od "OK".	" "	nie
sensors	Tablica czujników. Każdy czujnik w tablicy posiada dwa pola: <ul style="list-style-type: none"> <li>code - kod czujnika</li> <li>value - wartość czujnika</li> </ul> Szczegóły patrz niżej	[{"code": "pm2.5", "value": 73.45}]	nie

### Czujniki

Pole "sensors" w komunikacie zawiera tablicę z wartościami pomiarowymi czujników działających w urządzeniu. Jeżeli dany czujnik nie działa, nie został wykryty lub nie

pobrano jeszcze z niego wartości należy nie przysyłać o nim informacji w komunikacie. W szczególności tablica czujników może być pusta (np. pierwszy komunikat po restarcie urządzenia).

Każdy typ czujnika identyfikowany jest przez unikalny kod (pole "code"). Urządzenie może być wyposażone w następujące czujniki:

Kod (code)	Nazwa	Jednostka miary
temperature	temperatura zewnętrzna	°C
humidity	wilgotność względna zewnętrzna	%
pressure	ciśnienie atmosferyczne	hPa
pm1	stężenie pyłów zawieszonych PM1	µg/m3
pm2.5	stężenie pyłów zawieszonych PM2.5	µg/m3
pm10	stężenie pyłów zawieszonych PM10	µg/m3
co	stężenie tlenku węgla CO	mg/m3
o3	stężenie ozonu O3	µg/m3
so2	stężenie dwutlenku siarki SO2	µg/m3
no2	stężenie dwutlenku azotu NO2	µg/m3
h2s	stężenie siarkowodoru H2S	µg/m3
ch2o	stężenie formaldehydu CH2O	µg/m3
noise	poziom dźwięku	dB
soil_temp	temperatura gleby	°C
soil_vwc	zawartość wody w glebie (VWC)	%
soil_aper	pozorna przenikalność elektr. gleby	-

Częstość wysyłania komunikatów do serwera powinna wynosić nie więcej niż jeden raz na pięć minut.