AWS súhrn

1. Úprava kódu pre jeden senzor
    - Knižnice

```
#include <OneWire.h>
#include <DallasTemperature.h>
#include <WiFi.h>
#include <PubSubClient.h>
#include <WiFiClientSecure.h>
#include "secrets.h"
#include <ArduinoJson.h>
```

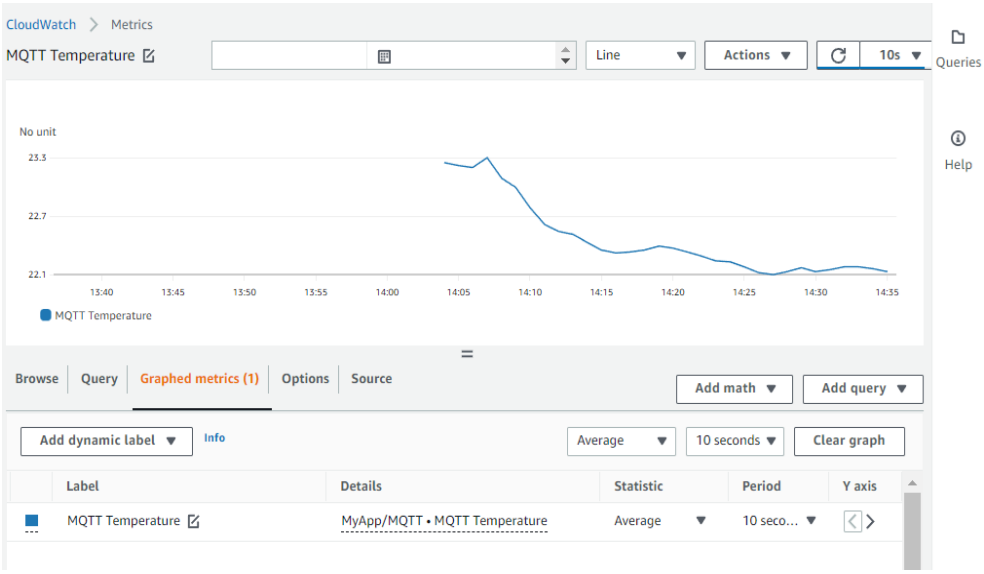
    - Dôležité časti kódu priamo pre AWS

```
// Set up the client certificate and private key
  wifiClient.setCertificate(AWS_CERT_CRT);
  wifiClient.setPrivateKey(AWS_CERT_PRIVATE);
  wifiClient.setCACert(AWS_CERT_CA);

 // Set up MQTT client
  mqttClient.setServer(AWS_IOT_ENDPOINT, 8883);
  mqttClient.setCallback(callback);

 // Connect to the AWS IoT MQTT broker and subscribe to the topic
  while (!mqttClient.connected()) {
    Serial.print("Connecting to AWS IoT broker...");
    if (mqttClient.connect(THINGNAME)) {
      Serial.println("Connected to AWS IoT MQTT broker");
      mqttClient.subscribe(AWS_IOT_SUBSCRIBE_TOPIC);
      Serial.println("Subscribed to AWS MQTT topic");
    } else {
      delay(1000);
    }
  }


   mqttClient.loop();
// Request temperature from DS18B20 sensor
    if (mqttClient.connected()) {
    sensors.requestTemperatures();

// Get temperature value in Celsius
    float temperature = sensors.getTempCByIndex(0);

// Create a JSON object and set its values
    StaticJsonDocument<64> jsonDoc; jsonDoc["temperature"] = temperature;

// Convert the JSON object to a string and publish payload to the MQTT broker
    String payload; serializeJson(jsonDoc, payload);
    mqttClient.publish(AWS_IOT_PUBLISH_TOPIC, payload.c_str());

// Print temperature value to serial console
    Serial.print("Temperature: "); Serial.print(temperature);
    Serial.println(" °C");
    delay(5000); }
```

2. Registrácia zariadenia do IoT Core
3. Vytvorenie certifikátov, endpoint
4. Testovanie získaných dát – integrovaný MQTT test client
5. Prepojenie s cloudom - vytvorenie log groups, metrický filter

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "iot:Connect",
      "Resource": "arn:aws:iot:us-east-1:867003613246:client/ESP32_DS18"
    },
    {
      "Effect": "Allow",
      "Action": "iot:Publish",
      "Resource": "arn:aws:iot:us-east-1:867003613246:topic/ESP32_DS18/pub"
    },
    {
      "Effect": "Allow",
      "Action": "iot:Subscribe",
      "Resource": "arn:aws:iot:us-east-1:867003613246:topicfilter/ESP32_DS18/sub"
    },
    {
      "Effect": "Allow",
      "Action": "iot:Receive",
      "Resource": "arn:aws:iot:us-east-1:867003613246:topic/ESP32_DS18/sub"
    }
  ]
}
```

- povolenia pre CloudWatch



- automaticky vytvorený dashboard