**HO CHI MINH CITY UNIVERSITY TECHNOLOGY AND EDUCATION**

🞠◊🞠◊🞠



**GVHD**

**Phạm Công Thiện**

**TOPIC:**

**Connect to any board with sensors and update to mobile phone.**

**Introduction to IoT**

**GROUP: 09**

**Bùi Tuấn Anh – 17110001**

**Lê Phương Nam – 17110054**

**Đỗ Nguyễn Hoàng Hải – 17110024**

**SEMESTER : 2 – YEAR: 2019-2020**

**TP. HỒ CHÍ MINH –**

Preface

Nowadays, information technology is considered an important key industry of countries, especially developing countries, conducting industrialization and modernization like our country. The information explosion and the strong development of technical technology, to develop we need to computerize all sectors.

Along with the rapid development of computer hardware, the software has become more diverse, abundant, more complete and effective support for human consumption. The current software supports users with convenient use, fast processing time, and some highly automated operations.

Thus, the software development not only requires precision, handling a lot of practical work but also must meet other requirements such as speed, compatibility, user-friendly interface, familiar, paradigm reality to the computer to the user ease of use and high security ... software saves a huge amount of time, energy of users, increasing accuracy and efficiency.

1. Descibe the topic:
   1. Desciption:

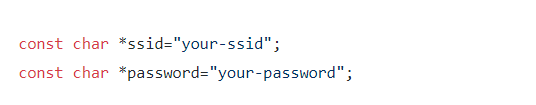
In this topic, our team choose ESP8266 board and DTH11 sensor to connect to mobile app to measure temperature.

The sensor will measure the temperature and put the data on AWS, then the application will retrieve data and show on the screen.

1. Device
   1. Handle ESP8266 board connect to wifi network
      * Handle connecting to a wifi network

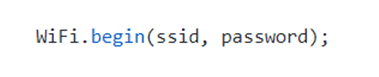


+ Setting ssid and password for wifi

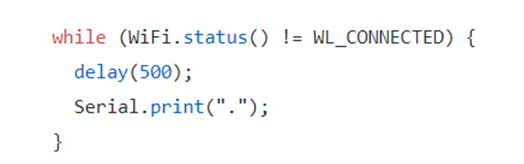


Notes : update these values suitable for your network

+ For connection we use :



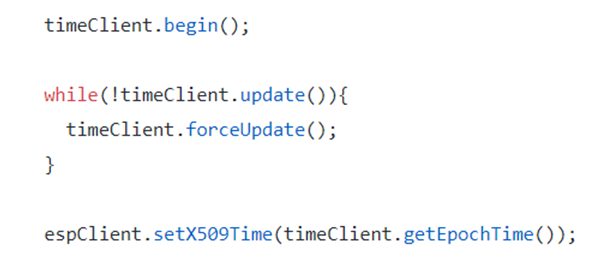
We need setting ssid, password for module can connect to a Access Point



The station have to connect to a Access Point. If it return false, it can not do any action. If it return true it means connect sucessful and you have to check status connection



Function use to take IP address of ESP station



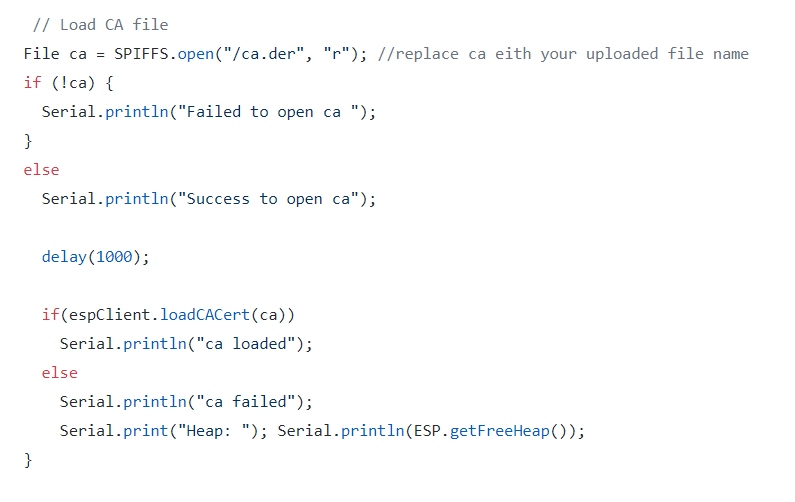
We need the current date/time to perform the TLS handshake because the certificate calidation process needs to knows the current time to work properly. WiFiClientSecure.setX509Time in order to feed it with the current time.

* 1. Handle connecting to Aws
     + We need to create Thing on AWS
     + Load file certificate

+ A private key for the device. AWS IoT will also generate it together with the device certificate.



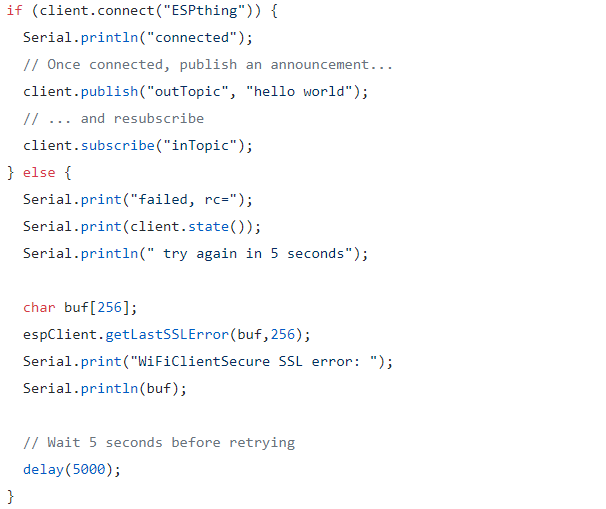
+ The Amazon CA certificate. The CA certificate can be downloaded



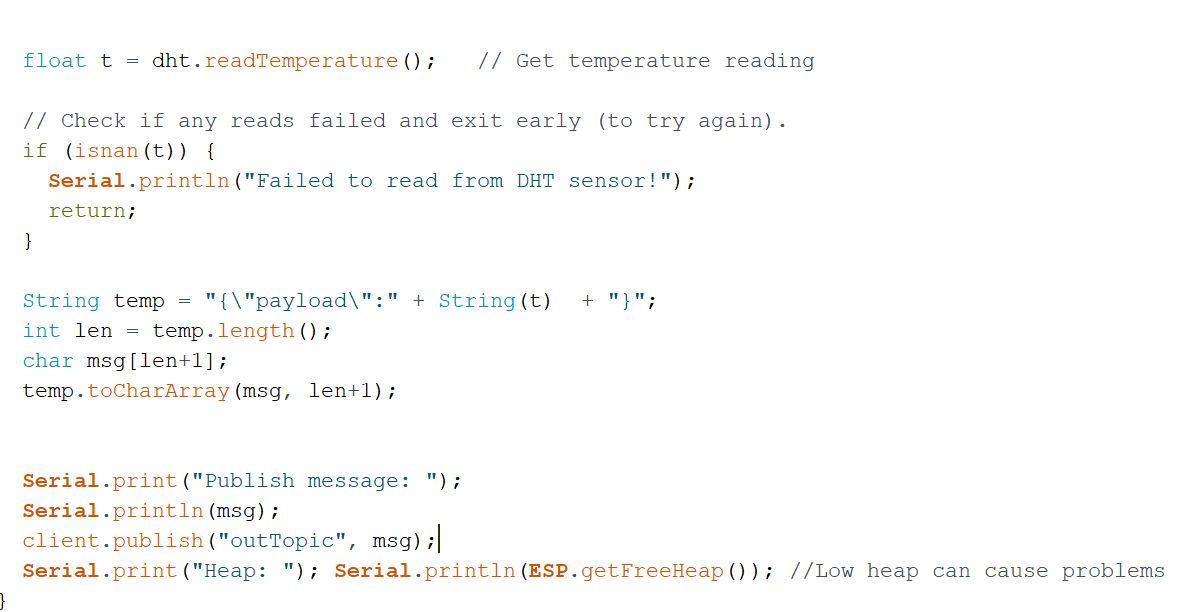
+A certificate for the device. AWS IoT will generate it for you.



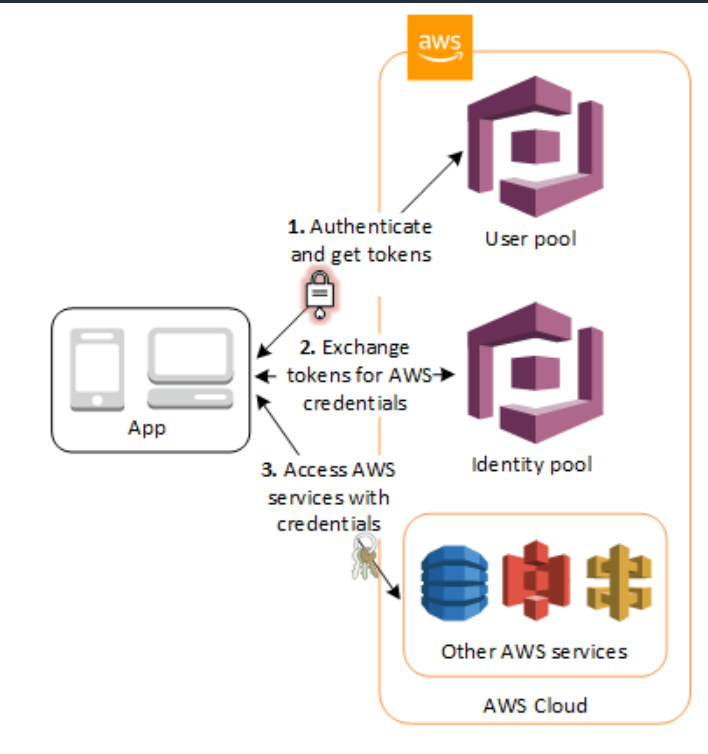
* + - The TLS handshake is processed by the WifiClientSecure class from Arduino Core for ESP8266. In this case the connection to MQTT is not working you can retrieve the error message with a call to WifiSecureClient.getLastSSLErrror



* + - Read data from sensor and publish message to aws



1. Instructions on how connect android devices to connect to aws iot core:



picture 1 Source : https://docs.aws.amazon.com/cognito/latest/developerguide/what-is-amazon-cognito.html

To connect to any aws services .First the client app should authenticate itself then the aws cognito will associate an identity to the client app and authorize it to which aws servcices based on the role of the identity.

1. **AWS Amplify:**
   1. **Using amplify to connect android app to the cloud**

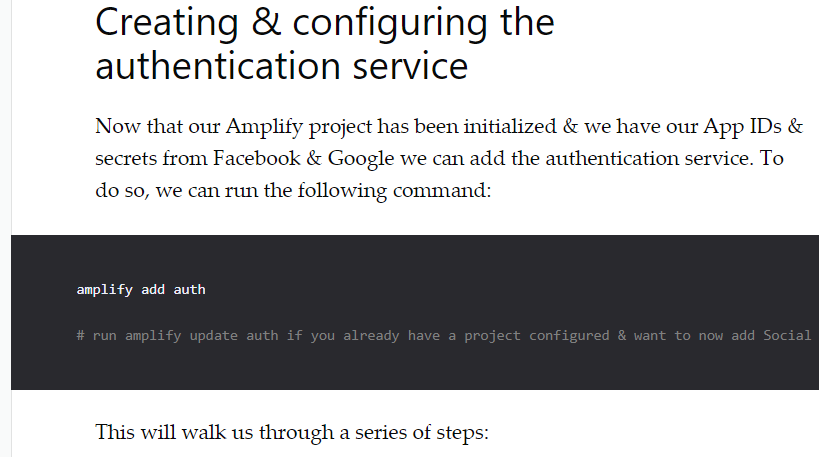
This component will let you interact with the aws cloud and aws services . To use this you will have to use an IAM account .

In this document , we will use aws amplify to init your app with the aws cloud. In details, we will create a user pool and an identity pool to connect to aws service.

Follow the instructions in the link below to create the users pool and identity pool in aws .

<https://docs.amplify.aws/lib/auth/getting-started/q/platform/android#automated-setup>

Or you can follow the link below to get users pool and identity pool in aws.



picture 2 Source : https://dev.to/dabit3/the-complete-guide-to-user-authentication-with-the-amplify-framework-2inh

<https://dev.to/dabit3/the-complete-guide-to-user-authentication-with-the-amplify-framework-2inh>

In this link use the code that I pasted here.



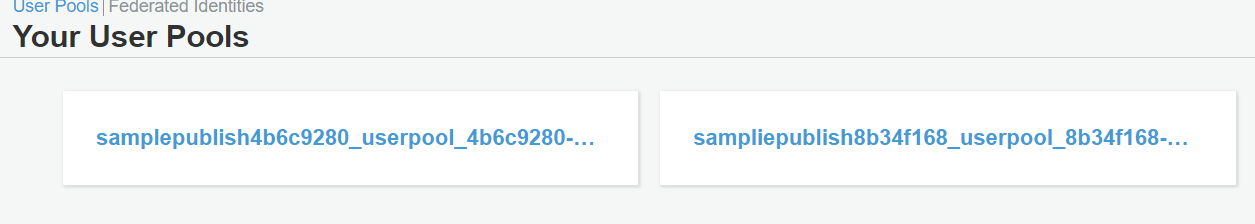
picture 3 Source <https://docs.amplify.aws/sdk/auth/federated-identities/q/platform/android#google-login-in-your->mobile-app

<https://docs.amplify.aws/sdk/auth/federated-identities/q/platform/android#google-login-in-your-mobile-app>

When your application runs this piece of code, it will prompt a login UI .

Now you go to the aws console then go to the cognito service. You will go to user pools.

From there you will see the user pool that you created using amplify CLI earlier.



picture 4 Source : Aws cognito console users pool

In the user pools console, you create a user then confirmed just like the image below



picture 5 Source : AWS cognito console users pool

You use this account to login to your application then cognito will create an identity assciate with the user you created.

Then you attach role for the identity pool . To ease of use , you can attach administrator role.

Go to your application’s identity pool then click edit identity pool you will see where to attach a role.

After that you can follow the instructions in the link below to connect to your aws iot core and use its api.

<https://docs.amplify.aws/lib/pubsub/getting-started/q/platform/js>

1. REFERENCES: