

Practica 4. Comunicacions a través de la pila TCP/IP

Objectius de la pràctica

L'objectiu de la pràctica és conèixer el funcionament del protocol TCP/IP i treballar sobre aquesta pila per muntar un sistema de transferència de dades entre un dispositiu i un servidor, analitzant com són els paquets necessaris.

Per tal d'assolir l'objectiu, treballarem amb una doble vessant. Per una banda la primera part de la pràctica estarà basada en els dispositius que hem fet servir a la pràctica anterior. Aquesta primera part de la pràctica es realitzarà amb l'ajut dels professors de pràctiques de l'assignatura.

Per altra banda, per tal d'analitzar el comportament dels diferents protocols ens instal·larem un software de simulació que ens permetrà l'estructura dels diferents paquets que estem enviant per la xarxa. El simulador que farem servir es el Packet Tracer de Cisco. Aquesta part de la pràctica serà prèviament explicada pels professors de l'assignatura.

Realització pràctica

Creació del canal recol·lector de dades

Per tal de crear un canal on agafar totes les dades que enviarem des de el nostre dispositiu, necessitem crear un compte. Podeu fer-ho individualment o treballar en grup tots els membres del laboratori. Per crear el vostre canal aneu a la següent adreça:

<https://thingspeak.com>

Obteniu un compte gratuït. Podeu especificar que sou estudiants.

ThingsSpeak és on anem a guardar les dades de la nostra mota i on podrem veure les dades que estem recollint. Recorda que per poder veure i analitzar les dades heu d'anar a ThingSpeak.com i activar un compte.

Un cop tenim el compte hem de crear un canal. Els canals de ThingSpeak és el lloc on les dades resten emmagatzemades. Crearem un nou canal seleccionant Channels, My Channels, i aleshores clicant a New Channel. Anomenem aquest canal com ESP8266_Nom_RSSI. El nom que posem al camp 1 és RSSI. Cliqueu a "Save Channel" per finalitzar el procés.

El firmware que hem de carregar a la mota ha de fer el següent:

- 1.- Fer un scan per detectar les xarxes WiFi que té disponibles
- 2.- Des de consola, indicar-li a quina xarxa WiFi es pot connectar
- 3.- Fer una lectura del RSSI que té amb aquesta xarxa

4.- Connectar-se al servidor remot de thingspeak.com per tal de transmetre la dada de RSSI.

Tenim dues formes de fer-ho. La primera correspon a un format genèric, on ataquem a thingspeak com a qualsevol servidor sobre el que vulguem actuar. Aquests serien els passos:

- i) Definició de la xarxa WiFi sobre la que treballem, característiques del servidor i interval de post que definim inicialment (Figura 1):

```
#include <ESP8266WiFi.h>

// Wi-Fi Settings
const char* ssid = "XXX"; // your wireless network name (SSID)
const char* password = "XXXXXXX"; // your Wi-Fi network password

WiFiClient client;

// ThingSpeak Settings
const int channelID = XXX;
String writeAPIKey = "XXXXXXXXXXXXXXXX"; // write API key for your ThingSpeak Channel
const char* server = "api.thingspeak.com";
const int postingInterval = 20 * 1000; // post data every 20 seconds
```

Figura 1. SSID, PWD, identificador de canal i APIKey per poder carregar les dades al servidor.

- ii) Connexió com a client podeu fer servir el següent codi, on a partir de l'objecte client, enviem al servidor la informació desitjada:

```
void loop() {
  if (client.connect(server, 80)) {

    // Measure Signal Strength (RSSI) of Wi-Fi connection
    long rssi = WiFi.RSSI();

    // Construct API request body
    String body = "field1=";
    body += String(rssi);

    Serial.print("RSSI: ");
    Serial.println(rssi);

    client.println("POST /update HTTP/1.1");
    client.println("Host: api.thingspeak.com");
    client.println("User-Agent: ESP8266 (nothans)/1.0");
    client.println("Connection: close");
    client.println("X-THINGSPEAKAPIKEY: " + writeAPIKey);
    client.println("Content-Type: application/x-www-form-urlencoded");
    client.println("Content-Length: " + String(body.length()));
    client.println("");
    client.print(body);

  }
}
```

Figura 2. Software de connexió amb el servidor especificat.

- iii) Finalment recordeu tancar la connexió fent servir la funció stop().

```
client.stop();

// wait and then post again
delay(postingInterval);
```

Figura 3. Tancament de la connexió i retard

Expliqueu **detalladament** el funcionament del codi.

Una altra manera de fer-ho és utilitzant la llibreria ThingSpeak.h proporcionada per la plataforma i que facilita tota la connexió.

- i) El primer que cal fer es carregar la llibreria que ThingSpeak ha desenvolupat. Aneu al Administrador de Llibreries del IDE de Arduino, cliqueu en Herramientas + Administrador de Bibliotecas. Busqueu la llibreria ThingSpeak i instal·leu-la.
- ii) Afegiu la llibreria de ThingSpeak.h al vostre programa

```
#include <ThingSpeak.h>
```

- iii) Configureu el dispositiu com client de WiFi per connectar-vos a internet a través del mòbil o compartint la WiFi des de l'ordinador, tal i com es mostra a la Figura 1. Caldrà inicialitzar la connexió amb ThingSpeak

```
ThingSpeak.begin(client)
```

- iv) Pujeu les dades necessàries que voleu mostrar. En el cas particular que es mostra a la figura 4, es mostraven 4 paràmetres. A vosaltres només es demana mostrar 1. La RSSI del vostre dispositiu. En cas que vulgueu ampliar la pràctica, parleu amb els professors i us donaran dispositius addicionals per monitoritzar les condicions climàtiques d'allà on esteu, seguint el format següent:

```

//=====

//=====
// Data transmission to ThingSpeak database
// hcho = hcho/5.0;
// relativeHumidity = relativeHumidity/100.0;
// temperature = temperature/200;
ThingSpeak.setField(1, WiFi.RSSI());
ThingSpeak.setField(2, temp);
ThingSpeak.setField(3, humidity);
ThingSpeak.setField(4, co2);

// write to the ThingSpeak channel
int x = ThingSpeak.writeFields(ChannelID, API_Key);
// if(x == 200){
//   Serial.println("Channel update successful.");
// }
// else{
//   Serial.println("Problem updating channel. HTTP error code " + String(x));
// }

}

```

Figura 4. Transmissió de diferents paràmetres a la base de dades de ThingSpeak

Veieu que efectivament, es mostra a la web de ThingSpeak la vostra dada, tant en format gràfic, com en format numèric. Feu una captura i afegiu-lo a la vostra entrega explicant detalladament el procés que heu seguit.

Creació d'un Punt d'Accés

Fins ara, hem treballat amb el nostre mòdul configurant-lo com a client per tal que es connecti a una xarxa WiFi. En aquest exercici el que farem es que sigui el nostre mòdul qui generi una xarxa WiFi amb la qual ens podem connectar i transmetre informació.

Per configurar el mòdul ESP8266 com a un punt d'accés hem de passar de la configuració normal a la de WIFI_AP. Això es fa utilitzant la comanda `WiFi.mode(WIFI_AP)`. Haurem d'establir el nom de la nostra WiFi i quina clau volem fer servir, tal i com es mostra en la figura 5.

```
#include <ESP8266WiFi.h>

char ssid[] = "MyWiFi";
char password[] = "password";

void setup() {
  // put your setup code here, to run once:
  Serial.begin(9600);
  Serial.println("WiFi access point test");
  WiFi.mode(WIFI_AP);
  WiFi.softAP(ssid, password);
  Serial.println("WiFi ON");
}

void loop() {
  // put your main code here, to run repeatedly:
}
```

Figura 5. Configuració del mòdul ESP8266 com Access Point

Farem servir la funció `softAP` per tal de passar el nom i la contrasenya de la WiFi que volem crear. Amb això ja tindrem la nostra WiFi funcionant i ens podrem connectar a ella des de qualsevol dispositiu, per exemple altres motes.

A partir de tot això, munteu una arquitectura client-servidor, on el client fa una petició de dades i el servidor les subministra. Teniu llibertat per escollir que i com transmetre. Per exemple podeu demanar un valor aleatori, o muntar un xat fent servir la consola... Transmeteu les dades entre client i servidor fent servir un fitxer JSON. Opcionalment, podeu també demanar als professors de pràctiques una placa per mesurar Temperatura, Pressió i humitat i connectar-la al vostre dispositiu i tenir una estació meteorològica.

Simulació amb Packet Tracer

Descarregueu-vos el programa Packet Tracer de forma gratuïta des de la següent adreça:

<https://www.netacad.com/es/courses/packet-tracer>

Feu el curs "Introduction to Packet Tracer" per entendre el funcionament del programa.

Configura una xarxa senzilla amb els següents dispositius:

- 1.- Quatre PCs. Assigneu a cada PC una IP privada del tipus 192.168.0.x amb màscara 255.255.255.0. Ho podeu fer clicant en cada PC, anant a configuració, seleccionant la interfície FastEthernet i indicant IP i màscara.
- 2.- Dos routers connectats entre ells pel port sèrie i un switch per cada router. Escolliu-lo amb com a mínim quatre boques per connectar els PCs.
- 3.- Feu un ping des de un dels ordinadors a qualsevol altre. Aneu a Desktop, seleccioneu Command Prompt i, al terminal que s'obre, feu un ping a qualsevol dels altres ordinadors. Per veure que està passant cliqueu a Event List.
- 4.- A l'Event List podeu veure tot el tràfic que s'està generant en la xarxa. Si feu un ping podeu analitzar el format de la trama Ethernet, el datagrama IP i el paquet ICMP.

Expliqueu els diferents passos i com es relacionen els punts 1, 2 i 3 amb el punt 4.

Un cop fet això, connecta el núvol a la teva xarxa a través del router de sortida. Configureu NAT dinàmic en tots dos routers, assignant un pool de 2 IPs públiques a cada router. A l'altra banda del núvol connecteu un servidor web. Connecteu, amb l'ajuda dels professors la vostra xarxa amb el servidor.

Un cop connectats al servidor i, fent servir l'sniffer que subministra packet tracer, identifiqueu els diferents protocols involucrats en la comunicació entre servidor i client.

A partir d'aquesta informació, compareu els protocols i els resultats amb la tasca feta en la primera part de la pràctica.