Department of Electronic & Telecommunications Engineering

University of Moratuwa

EN3251 Internet of Things

Laboratory Exercise 4: Hardware Implementation of IoT System Components

2020 Batch **Semester 5**

Overview:

In this workshop you will get basic exposure to the Raspberry Pi, and how to establish a Local MQTT server and a CoAP server on it. You will then connect to these servers using NODE MCU devices as clients.

Objective:

In this workshop, we will

- Introduce Raspberry Pi and Server establishment.
- NODE MCU CoAP client and MQTT client with Local Servers.

Prerequisites:

- Laptop installed with Arduino IDE (Configured with ESP boards)
- Raspberry Pi (will be provided)
- SD card writer/ Adapter
- NODE MCU (will be provided)
- Working Californium CoAP server on Eclipse
  - A server with at least one GET method and one PUT method implemented.

Activities:

Configuring the Raspberry Pi:

Setup the Raspberry Pi with Wi-Fi configurations:

Step 1: Take out the SD card carefully and connect it to your Computer with an SD card writer.

Step 2: Open your Notepad or any other text editor and create a new file with the following content.

```
country=sl
update_config=1
ctrl_interface=/var/run/wpa_supplicant

network={
 scan_ssid=1
 ssid="F1"
 psk="qwertyuiop"
}
```

Rename and save it as wpa_supplicant.conf (Make sure the extension is correct)

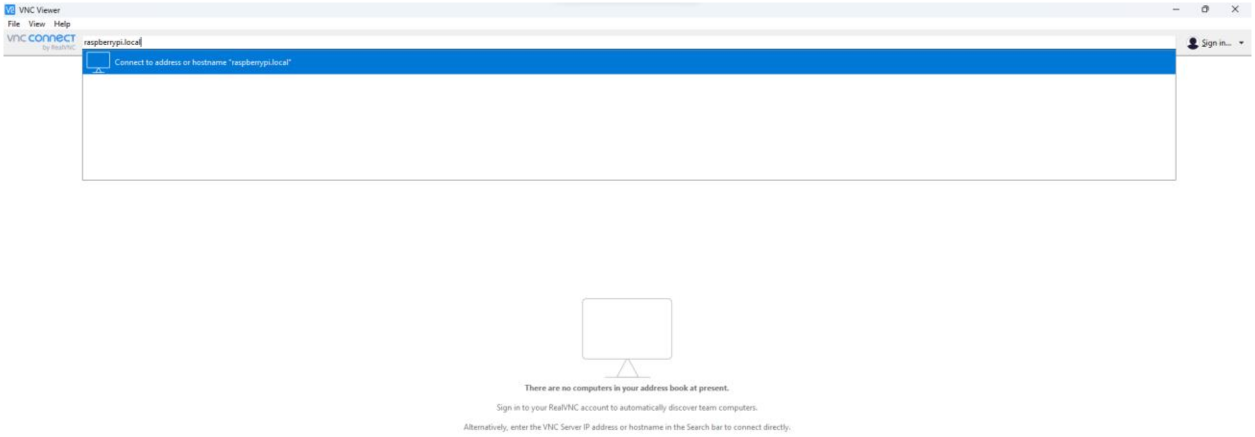
Step 3: Copy the file to boot partition on your SD card.

Step 4: Now, your Raspberry Pi should connect to your Wi-Fi SSID.

Connect to your Raspberry Pi via VNC viewer:

Step 1: Connect your laptop to the same Wi-Fi network as the Raspberry Pi.

Step 2: Enter your local IP address for the Raspberry Pi, or just enter "raspberrypi.local" in your address bar.



raspberrypi.local

Connect to address or hostname "raspberrypi.local"

There are no computers in your address book at present.

Sign in to your RealVNC account to automatically discover team computers.

Alternatively, enter the VNC Server IP address or hostname in the Search bar to connect directly.

Step 3: Now enter the provided credentials for your account and you will be able to get the GUI.

Install Required software:

Step 1: Open the Raspberry Pi terminal.

Step 2: Install JDK.

```
sudo apt install default-jdk
```

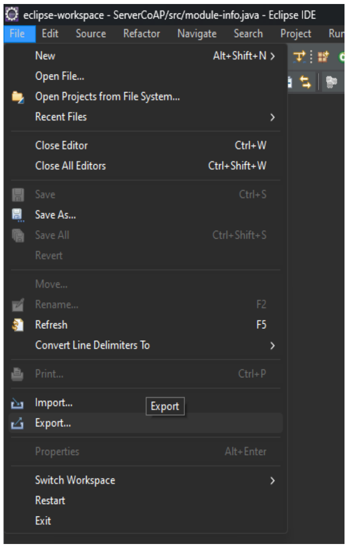
Step 3: Install mosquitto broker and mosquitto clients.

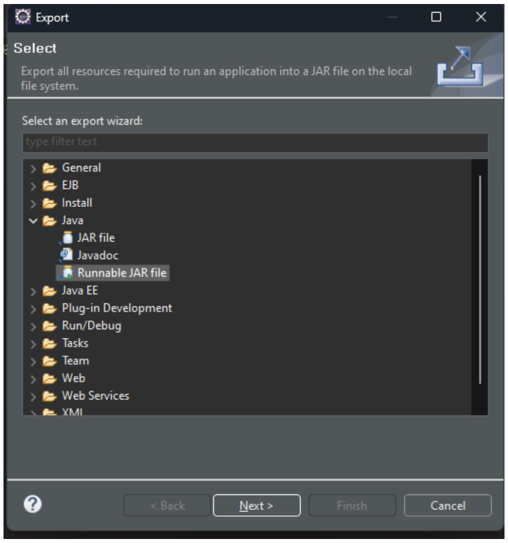
```
wget http://repo.mosquitto.org/debian/mosquitto-repo.gpg.key
sudo apt-key add mosquitto-repo.gpg.key
cd /etc/apt/sources.list.d/
sudo wget http://repo.mosquitto.org/debian/mosquitto-jessie.list
sudo wget http://repo.mosquitto.org/debian/mosquitto-stretch.list
sudo wget http://repo.mosquitto.org/debian/mosquitto-buster.list
sudo apt-get update
sudo apt-cache search mosquito
sudo apt-get install mosquitto mosquitto-clients
```

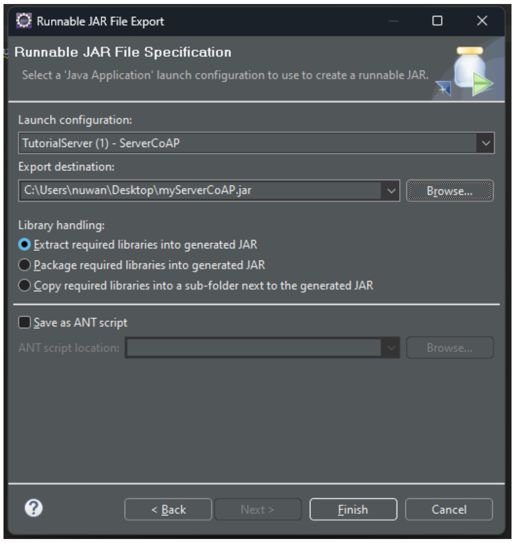
Implementing a CoAP server on Raspberry Pi

Export Runnable Jar file from:

Step 1: Open your Eclipse Environment, and go to File-> Export, and Java -> Runnable Jar File after that, specify an Export destination, select the following options and Finally press Finish.






Runnable JAR File Export
Runnable JAR File Specification
Select a 'Java Application' launch configuration to use to create a runnable JAR.
Browse...
Package required libraries into generated JAR
Copy required libraries into a sub-folder next to the generated JAR
Save as ANT script
ANT script location:
Browse...
< Back
Next >
Finish
Cancel

Launch configuration: TutorialServer (1) - ServerCoAP

Export destination: C:\Users\nuwan\Desktop\myServerCoAP.jar

Library handling: ● Extract required libraries into generated JAR

Step 2: Copy the portable jar file to the Raspberry Pi via VNC server.

Start your CoAP Server:

Step 1: Open a terminal on your Raspberry Pi, navigate to the location of runnable Jar file.

Step 2: Run the Jar file with Java,

```
java -jar myServerCoAP.jar
```

Test Your CoAP Server with the Cu Client:

Step 1: Acquire the Local Raspberry Pi IP address. Use the following command to do so.

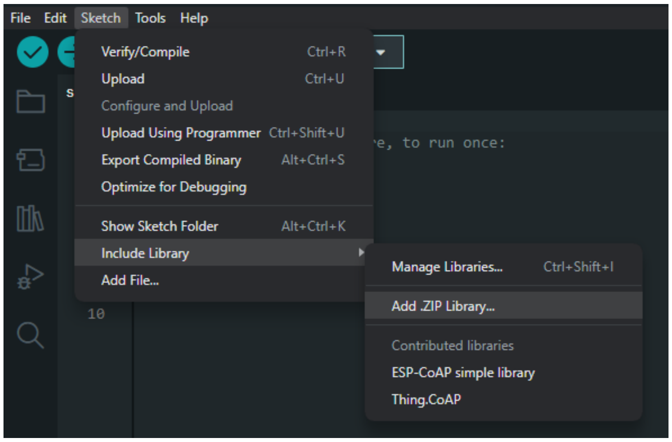
```
ifconfig
```

Step 2: Open your Cu Client in MS Edge and provide the RPI Ip address with the port.

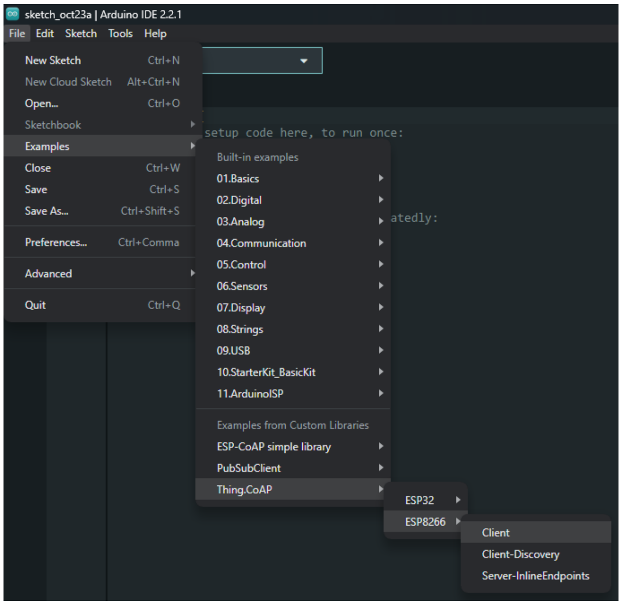
Implement a CoAP client on the NODE MCU:

Step 1: Download the NODE MCU CoAP Client as a zip file.

Step 2: Open Arduino IDE and go to *Sketch->Include Library->Add .ZIP Library* and select the download zip file.

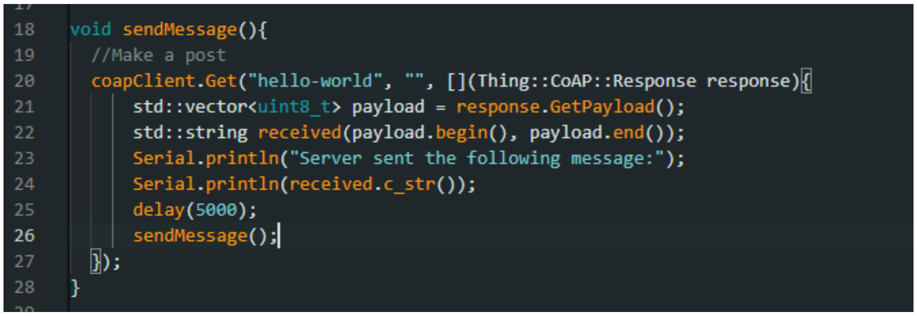


Step 3: Load the CoAP client code from examples. Go to *File->Examples->Thing.CoAP>ESP8266->Client*.



Step 4: Include your Wi-Fi credentials in the 15,16 lines and change the IP address to your RPi's IP address in line 47. (Note: You can find RPi's IP address by executing *ifconfig* in RPi terminal.

Step 5: Include the resource URI you want to connect to as follows. Here we connect to hello-world resource. (Note: Here we request a GET from the resource. In order to work this, the resource should be able to handle GET requests)



```
void sendMessage(){
  //Make a post
  coapClient.Get("hello-world", "", [](Thing::CoAP::Response response){
      std::vector<uint8_t> payload = response.GetPayload();
      std::string received(payload.begin(), payload.end());
      Serial.println("Server sent the following message:");
      Serial.println(received.c_str());
      delay(5000);
      sendMessage();
  });
}
```

Step 6: Upload the program to the ESP board and observe the results.

Implementing an MQTT Broker on Raspberry Pi

Configure mosquitto:

Step 1: Open RPi terminal

Step 2: Open mosquitto configuration file

```
sudo nano /etc/mosquitto/mosquitto.conf
```

Step 3: Add the following two lines in the end of the config file.

```
allow_anonymous true
listener 1883
```

After that press ctrl+o to save the file and then press ctrl+x to exit from nano editor.

Step 4: Run the mosquitto broker

```
sudo systemctl stop mosquitto.service
sudo mosquitto -c /etc/mosquitto/mosquitto.conf
```

Try following:

- Implement a PUT method with CoAP in Node MCU (use the GitHub repo and explore the library)
- Expose Both CoAP and MQTT servers to internet and connect your Node MCU with it.