

Eclipse Paho MQTT Support for 4DIAC-FORTE

Requirements:

Linux VM (e.g. Mintlinux 17.3 64 Bit) with cross compiler toolchain installed (e.g. built with crosstool-ng)

1. on the Raspberry Pi

Openssl Library is required for Paho MQTT C Client.

```
plc@conmeleon-cl:~/ $ sudo apt-get update
plc@conmeleon-cl:~/ $ sudo apt-get install libssl-dev
```

Shutdown the Raspi and move microSD card with USB adapter to desktop PC.

2. on the Linux VM

Redirect the USB SD card reader to the Linux VM and copy the include header and libraries to your toolchain.

Directory

/usr/include/openssl

File

/usr/include/arm-linux-gnueabi/openssl/opensslconf.h

```
herwig@workshop /media/herwig/859cf567-e7d3-4fa7-82b8-cb835cd272c6/usr/include $ cp -a
openssl ~/develop/toolchains/arm-raspi-linux-gnueabi/arm-raspi-linux-
gnueabi/sysroot/usr/include/
```

```
herwig@workshop /media/herwig/859cf567-e7d3-4fa7-82b8-cb835cd272c6/usr/include/arm-
linux-gnueabi/openssl $ cp -a opensslconf.h ~/develop/toolchains/arm-raspi-linux-gnueabi/arm-
raspi-linux-gnueabi/sysroot/usr/include/openssl/
```

Files

/usr/lib/arm-linux-gnueabi/libssl.a

/usr/lib/arm-linux-gnueabi/libssl.so

/usr/lib/arm-linux-gnueabi/libssl.so.1.0.0

/usr/lib/arm-linux-gnueabi/libcrypto.a

/usr/lib/arm-linux-gnueabi/libcrypto.so

/usr/lib/arm-linux-gnueabi/libcrypto.so.1.0.0

Directories

/usr/lib/arm-linux-gnueabi/openssl-1.0.0

```
herwig@workshop /media/herwig/859cf567-e7d3-4fa7-82b8-cb835cd272c6/usr/lib/arm-linux-
gnueabi $ cp -a libssl.* ~/develop/toolchains/arm-raspi-linux-gnueabi/arm-raspi-linux-
gnueabi/sysroot/usr/lib/
```

```
herwig@workshop /media/herwig/859cf567-e7d3-4fa7-82b8-cb835cd272c6/usr/lib/arm-linux-
gnueabi $ cp -a libcrypto.* ~/develop/toolchains/arm-raspi-linux-gnueabi/arm-raspi-linux-
gnueabi/sysroot/usr/lib/
```

```
herwig@workshop /media/herwig/859cf567-e7d3-4fa7-82b8-cb835cd272c6/usr/lib/arm-linux-
gnueabi $ cp -a openssl-1.0.0 ~/develop/toolchains/arm-raspi-linux-gnueabi/arm-raspi-linux-
gnueabi/sysroot/usr/lib/
```

3. Download Sources of Eclipse Paho MQTT on the Linux VM

Clone the Eclipse Paho MQTT sources from Github (the Repository given on the Eclipse Paho MQTT Website and in the 4DIAC documentation did not work for me).

```
herwig@workshop ~/develop/repos/github $ git clone https://github.com/eclipse/paho.mqtt.c.git
```

4. cross compile MQTT C/C++ Library on the Linux VM

```
herwig@workshop ~/develop/repos/github/paho.mqtt.c $ export CC=arm-raspi-linux-gnueabi-hf-gcc
```

```
herwig@workshop ~/develop/repos/github/paho.mqtt.c $ export TARGET_PLATFORM=Arm
```

```
herwig@workshop ~/develop/repos/github/paho.mqtt.c $ make
```

5. copy library binaries and include files to toolchain directory on the Linux VM

Library binaries can be found in paho.mqtt.c/build/output directory and include files are in paho.mqtt.c/src

```
herwig@workshop ~/develop/repos/github/paho.mqtt.c/build/output $ cp -a libpaho*  
~/develop/toolchains/arm-raspi-linux-gnueabi-hf/arm-raspi-linux-gnueabi-hf/sysroot/usr/lib/
```

```
herwig@workshop ~/develop/repos/github/paho.mqtt.c/src $ cp -a MQTT*.h  
~/develop/toolchains/arm-raspi-linux-gnueabi-hf/arm-raspi-linux-gnueabi-hf/sysroot/usr/include/
```

5.a also copy the library binaries to the /usr/lib/arm-linux-gnueabi-hf directory of the RaspberryPi

```
herwig@workshop /media/herwig/859cf567-e7d3-4fa7-82b8-cb835cd272c6/usr/lib/arm-linux-gnueabi-hf $ cp -a ~/develop/repos/github/paho.mqtt.c/build/output/libpaho* .
```

6. enable Eclipse Paho MQTT Option in FORTE with Cmake

Enable the FORTE_COM_PAHOMQTT option and set FORTE_COM_PAHOMQTT_DIR to directory where you copied the paho libraries to


```
May 14 12:02:19 conmeleon-c1 mosquitto[1301]: Starting network daemon::  
mosq....  
May 14 12:02:19 conmeleon-c1 systemd[1]: Started LSB: mosquitto MQTT  
v3.1 me....  
Hint: Some lines were ellipsized, use -l to show in full.
```

Now test the MQTT broker functionality:

```
plc@conmeleon-c1:/etc/apt/sources.list.d $ mosquitto_sub -h 127.0.0.1 -t  
Test/#
```

Use a different terminal session to publish a value to the new Test/ topic.

```
plc@conmeleon-c1:~ $ mosquitto_pub -h 127.0.0.1 -t Test/Temp -m "23.7"  
plc@conmeleon-c1:~ $ mosquitto_pub -h 127.0.0.1 -t Test/Temp -m "31.2"
```

The first terminal session we used for subscribing should now show the published temperature values.

```
plc@conmeleon-c1:/etc/apt/sources.list.d $ mosquitto_sub -h 127.0.0.1 -t  
Test/#  
23.7  
31.2
```