EMS Tunneling server / client build process on Ubuntu 20.04

Building localproxy (Tunneling server)

```
sudo apt update
sudo apt upgrade
sudo apt install git zlib1g wget cmake libssl-dev
git clone https://github.com/eguanatech/aws-iot-securetunneling-localproxy.git
git checkout imx6-eguana
wget https://github.com/boostorg/boost/releases/download/boost-1.81.0/boost-1.81.0.tar.gz
tar xzvf boost-1.81.0.tar.gz
cd boost-1.81.0
./bootstrap.sh
sudo ./b2 install link=static
cd ..
wget https://github.com/protocolbuffers/protobuf/releases/download/v3.17.3/protobuf-all-3.17.3.tar.gz
tar xzvf protobuf-all-3.17.3.tar.gz
cd protobuf-3.17.3
mkdir build
cd build
cmake ../cmake
make
sudo make install
cd ../../
git clone --branch v2.13.6 https://github.com/catchorg/Catch2.git
cd Catch2
mkdir build
cd build
cmake ../
make
sudo make install
cd ../../
mkdir build
cd build
cmake ../
make
```

You can find the built image localproxy and localproxytest at the bin folder. (i.e. aws-iot-securetunneling-localproxy/build/bin/).

localproxytest is the unit test suite that AWS team created, localproxy is the proxy server we need for tunneling.

The script I wrote for the proxy server is located at /home/ikayzo/Downloads/tunnel.sh. You can modify it if you need to.

Building aws-iot-device-client (Tunneling client on EMS)

```
sudo apt update
sudo apt install software-properties-common build-essential g++-arm-linux-gnueabihf gcc-arm-linux-gnueabihf gdb-multiarch
git clone https://github.com/eguanatech/aws-iot-device-client.git
cd aws-iot-device-client
git checkout master
mkdir build
cd build
```

```
wget --ca-certificate=/etc/ssl/certs/ca-certificates.crt https://www.openssl.org/source/openssl-1.1.1n.tar.gz
tar -xvzf openssl-1.1.1n.tar.gz
export INSTALL_DIR=/usr/lib/arm-linux-gnueabihf
cd openssl-1.1.1n
./Configure linux-generic32 shared \
      --prefix=$INSTALL_DIR --openssldir=$INSTALL_DIR/openssl \
      --cross-compile-prefix=/usr/bin/arm-linux-gnueabihf-
make depend
make
sudo make install
cd ../
cmake -DCMAKE_TOOLCHAIN_FILE=../cmake-toolchain/Toolchain-armhf.cmake ../
cmake --build . --target aws-iot-device-client
cmake --build . --target test-aws-iot-device-client
```

You can find the built image aws-iot-device-client at the build folder(aws-iot-device-client/build/) and test-aws-iot-device-client at the bin folder(aws-iot-device-client/build/test/). Since the image is cross-compiled with the arm toolchain these images won't be able to run on our host machine(x86). You need to scp these image to the EMC to run them.

After you build the aws-iot-device-client, copy this image to the yocto project and replace existing one(egear-manifest/source/meta-aws/recipes-iot/aws-iot-device-client/files/aws-iot-device-client). Build the yocto project after that. By doing this your Yocto image will include the latest version of device client you built.