# Hello,

Please note that for this exercise you will be required to do coding in C/C++ language. The objective of the exercise is to allow you to practice usage of a common crypto library.

**Action required -** As a prerequisite of the workshop, you are required to install some SW on your machine. Please follow the instructions below.

# Setup preparation

**Linux**:

If you don’t have access to a Linux machine, you can use WSL2 (Windows Subsystem for Linux) on your Windows machine. The installation of WSL2 takes some time and requires a reboot.

Instructions

1. Follow this [link to install WSL2](https://docs.microsoft.com/en-us/windows/wsl/install-win10#manual-installation-steps) on your machine. Follow all the instructions until you are confident that you have WSL 2 installed.
2. Once you get the wsl shell, enter this command to get a File Explorer window to the WSL system (don’t miss the “.” at the end!)

explorer.exe .

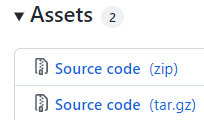
You can now use Drag & Drop or Copy-Paste to pass files to the WSL system.

Crypto library – we will use mbedTLS for this exercise. But you do need OpenSSL binary as well.

Development tools:

* Make sure you have dev tools like gcc and make installed:
* If not:
  + *sudo apt-get update*
  + *sudo apt install build-essential*

mbedTLS:

* *sudo apt install libmbedtls-dev*
* Another option:
  + All releases are in <https://github.com/Mbed-TLS/mbedtls/releases>
  + Select version 2.28.4 - [Release Mbed TLS 2.28.4 · Mbed-TLS/mbedtls (github.com)](https://github.com/Mbed-TLS/mbedtls/releases/tag/v2.28.4)
  + Scroll down to the Assets section and download the *mbedtls-x.y.z.tar.gz* file. E.g., 
  + Unzip the downloaded file: *tar -xvzf mbedtls-x.y.z.tar.gz*
* Build
  + Run *make* within *\mbedtls-x.y.z* folder

OpenSSL:

* Might be already installed – just try *openssl -h*
* Follow the instructions from - <https://nextgentips.com/2022/03/23/how-to-install-openssl-3-on-ubuntu-20-04/>