

Configuration Manual

MSc Research Project
Research Project

Dinesh Kanojiya
Student ID: X21104174

School of Computing
National College of Ireland

Supervisor: Jitendra Sharma

National College of Ireland
Project Submission Sheet
School of Computing



| | |
|-----------------------------|----------------------|
| Student Name: | Dinesh Kanojiya |
| Student ID: | X21104174 |
| Programme: | Research Project |
| Year: | 2018 |
| Module: | MSc Research Project |
| Supervisor: | Jitendra Sharma |
| Submission Due Date: | 15/08/2022 |
| Project Title: | Configuration Manual |
| Word Count: | 489 |
| Page Count: | 5 |

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

| | |
|-------------------|------------------|
| Signature: | |
| Date: | 11th August 2022 |

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

| | |
|--|--------------------------|
| Attach a completed copy of this sheet to each project (including multiple copies). | <input type="checkbox"/> |
| Attach a Moodle submission receipt of the online project submission , to each project (including multiple copies). | <input type="checkbox"/> |
| You must ensure that you retain a HARD COPY of the project , both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer. | <input type="checkbox"/> |

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

| | |
|----------------------------------|--|
| Office Use Only | |
| Signature: | |
| Date: | |
| Penalty Applied (if applicable): | |

Configuration Manual

Dinesh Kanojiya
X21104174

1 System BIOS Setup

The implementation is based on the Windows-based Intel SGX SDK; not all intel micro-processors offer Intel SGX. First, we must check that the system processor provides the Intel SGX features. This can be found in the BIOS configuration setting, where we need to enable it using toggle manually. Once it allows, then reboots the system.



Figure 1: Service Manager

To verify SGX is enabled, navigate to the service manager from windows start, then locate Intel SGX service and start it 1. This ensures that the Intel SGX is allowed and enclave options are ready to serve.

2 Visual Studio Professional and Intel SGX SDK:

First, download and install the Intel SGX SDK from the intel development centre Intel (n.d.). Second, install a visual studio professional (2013,2015,2017 and 2019) as the SGX SDK only supports these versions of visual studio professionals.

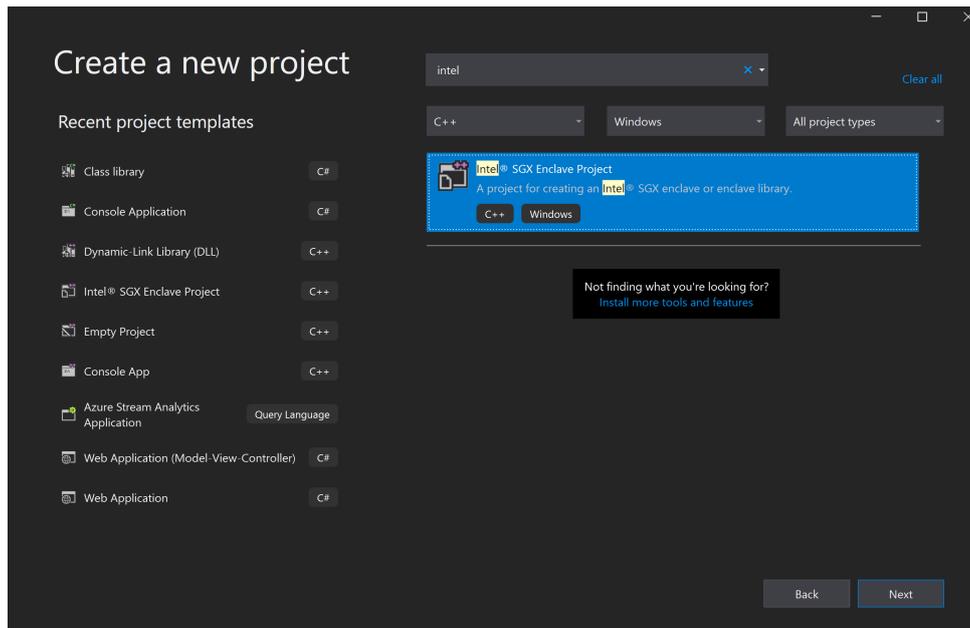


Figure 2: Visual Project Component

Once the above installation is done, it will add an extension of the Intel SGX project component to the visual studio 2. The SGX SDK is a C-based library which will easily be compatible with VS.

Once the Intel SGX project is chosen, allocate memory to the enclave either from a wizard or right-click on the project and select the memory size and several threads to be run on a single enclave.

3 Visual Studio Project Configuration

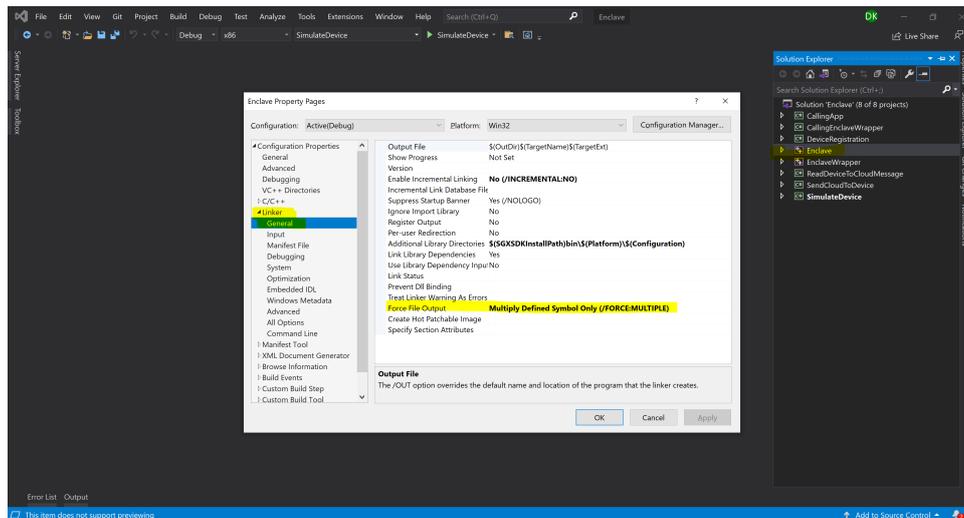


Figure 3: C++ Project Configuration

A few SGX APIs services might not be compatible with VS 2019 to make them compatible with the solution; we need to navigate project properties and choose Linker-; General-; Force File OutPut and set Multiply Defined Symbol Only (/FORCE: MULTIPLE)3. This will allow SGX SDK to be compiled successfully. The signed.DLL gets generated, which will expose methods to calling projects.

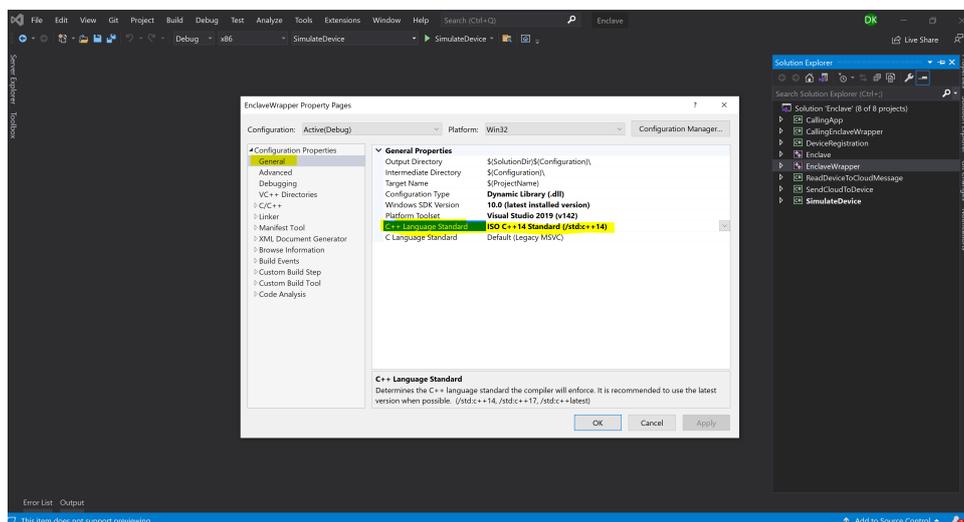


Figure 4: C++ Project Configuration

To consume signed.DLL, we need to create C++ based project this will inherit the SGX library from the enclave project. First, we need to set the compiler version for this project, right-click on the project, go to properties, under Configuration -; General -; C++ language standard set to ISO C++14 Standard (/std:c++14) 4. This will compile

C++ and C projects in one go.

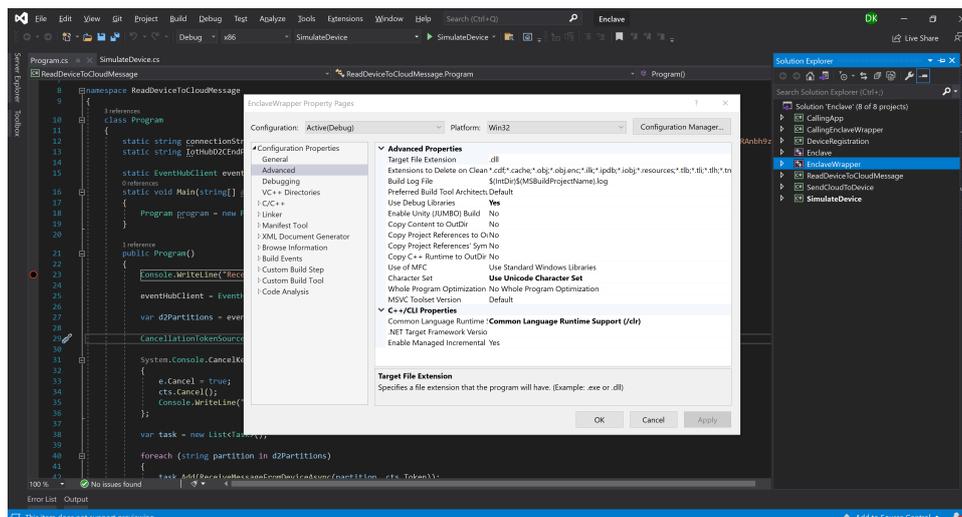


Figure 5: C++ Project Configuration

The C++ project needs to talk to the C project. To do so, we need to set the Common Language Runtime (CLR) of the .Net framework under Configuration -> Advanced -> Common Language Runtime set to Common Language Runtime Support (/clr) 5. After compilation, this will create a COM-based DLL, which can easily be consumed by any C based application.

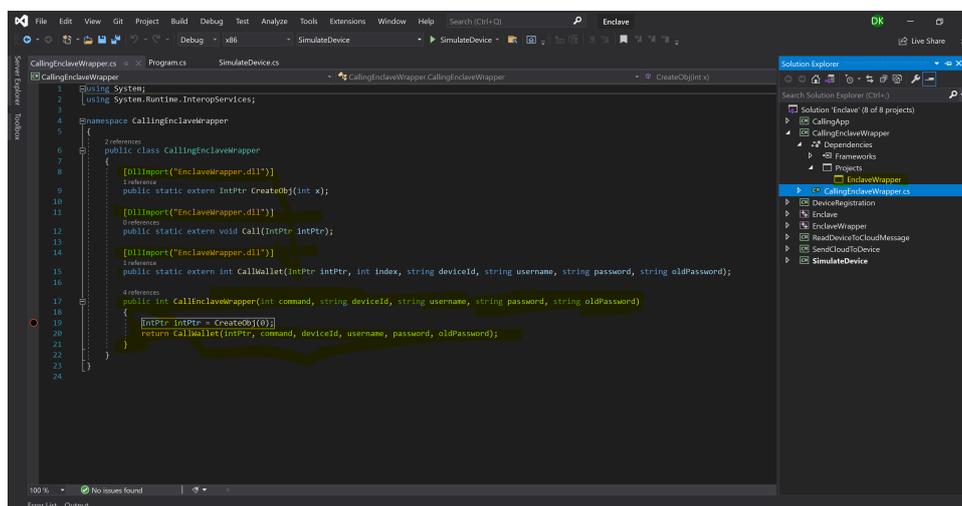


Figure 6: C++ Project Configuration

Create C console application and add reference of COM-based DLL 6. This will expose only the public method through which we must pass a required parameter to access the enclave. On top of this project, we need to create another C console application which will interact with the Azure cloud and serve as the IoT home device.

4 Azure IoT

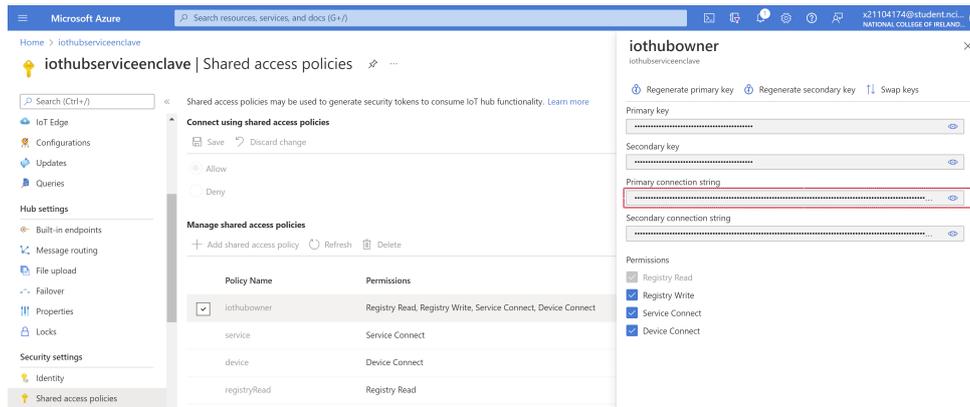


Figure 7: C++ Project Configuration

Login into the Azure cloud portal and create a new IoT hub service using the wizard. Once the IoT hub is created, then from the left panel, navigate to accessibility and copy the primary connection string⁷. This connection string will use inside the C code for data exchange.

References

Intel (n.d.). Intel[®] software guard extensions (intel[®] sgx) driver for windows*.