SEAL Compilation and Usage Instructions for Mac OS

Author: Owen Lo

June 2019

1 Introduction

The goal of this document is to compile the SEAL source code and run it on Mac OS. More specifically, we demonstrate the following:

1. Compile the SEAL C++ source code (Sec. 2.2).
2. Compile SEAL for .NET Core (Sec. 2.3).
3. Use SEAL for Homomorphic Voting application [1] (Sec. 2.4).

2 Compilation

2.1 Prerequisite Requirements

The following compilation software and tools are necessary:

- CMake (>=3.10). CMake can be installed via Homebrew.
- Xcode toolchain (>= 9.3).
- .NET Core 2.X and Visual Studio Community.

2.2 Compiling SEAL

Download the Microsoft SEAL source code from [2].

Navigate to the root folder of the SEAL source code and run the following commands:

```
mkdir ~/mylibs/
cd native/src
cmake -DCMAKE_INSTALL_PREFIX=~/mylibs . -DCMAKE_BUILD_TYPE=Release
make
make install
```

The compiled SEAL source code is now located in the **mylibs** folder. This folder can be deleted once the instructions in the next section (Sec. 2.3) are carried out.

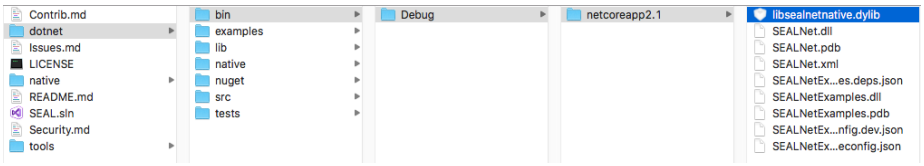

Figure 1: Output directory of libsealnetnative.dylib

2.3 Compiling SEAL for .NET

Return to the root directory of the SEAL source code and run the following commands:

```
cd dotnet/native
cmake -DCMAKE_PREFIX_PATH=~/mylibs . -DCMAKE_BUILD_TYPE=Release
make
```

If compilation was successful, the dynamic link library **libsealnetnative.dylib** is now located in *dotnet/bin/Debug/netcoreapp*2.1/ as depicted in Fig. 1. The next section demonstrates how one may use this library in the Homomorphic Voting application.

2.4 Use SEAL for Homomorphic Voting application

Download the SEAL Homomorphic Encrypting Voting code from [1].

Open the solution in Visual Studio. Build solution. Navigate to the root directory of the Homomorphic voting source code. Make a copy of the dynamic link library **libsealnetnative.dylib** (created in Sec. 2.3) and paste it into the *dotnet/lib/* directory within the Homomorphic voting source code. The project should now run successfully.

References

[1] SEAL Homomorphic Encryption Voting. `https://github.com/blockpass-identity-lab/seal-homomorphic-encryption-voting`, July 2019. Blockpass Identity Lab, Edinburgh, UK.

[2] Microsoft SEAL (release 3.2). `https://github.com/Microsoft/SEAL`, February 2019. Microsoft Research, Redmond, WA.