

# Window 환경에서 SEAL 라이브러리 설치

20200283 조범희

Window 환경에서 seal 라이브러리 설치를 진행하였습니다.

윈도우 설치의 경우 튜토리얼 영상과 첫 부분부터 달라 github 문서를 보고 따로 설치를 진행했습니다.

## Installing Microsoft SEAL

If you have root access to the system you can install Microsoft SEAL globally as follows:

```
cmake -S . -B build
cmake --build build
sudo cmake --install build
```

To instead install Microsoft SEAL locally, e.g., to `~/mylibs/`, do the following:

```
cmake -S . -B build -DCMAKE_INSTALL_PREFIX=~/mylibs
cmake --build build
sudo cmake --install build
```

Git clone 으로 파일을 C:/libs/SEAL 에 받은 후 Cmake 를 설치하고 위의 단계를 따랐습니다. Cmake 는 처음 사용해봐서 조금 공부가 필요했습니다. 최상위 디렉토리에는 반드시 CMakeList.txt 가 존재해야합니다. SEAL 폴더에서 PowerShell 을 통해 진행했고 sln 파일을 Visual Studio 로 열어 Install 솔루션을 build 했습니다. 그러나 에러가 발생했습니다.

```
1>1>
1>-- Install configuration: "Release"
1>-- Up-to-date: C:/Program Files (x86)/SEAL/include/SEAL-4.1/seal/util/config.h
1>CMake Error at cmake_install.cmake:36 (file):
1> file INSTALL cannot set permissions on "C:/Program Files
1> (x86)/SEAL/include/SEAL-4.1/seal/util/config.h": Permission denied.
1>
```

<에러 메시지>

```
# Install script for directory: C:/libs/SEAL

# Set the install prefix
if(NOT DEFINED CMAKE_INSTALL_PREFIX)
  set(CMAKE_INSTALL_PREFIX "C:/Program Files (x86)/SEAL")
endif()
```

<C:/libs/SEAL/build/cmake\_install.cmake>

위 cmake 파일을 보면 설치 폴더의 프리픽스가 Program Files (x86)/SEAL 입니다. 때문에 관리자 권한이 필요해 Visual Studio 의 빌드로는 실행되지 않았습니다. 이를 해결하기 위해 관리자 권한으로 PowerShell 을 실행해 수동으로 cmake install 을 진행했습니다.

```
PS C:\libs\SEAL> cmake --install build
-- Install configuration: "Release"
-- Installing: C:/Program Files (x86)/SEAL/include/SEAL-4.1/seal/util/config.h
-- Installing: C:/Program Files (x86)/SEAL/lib/seal-4.1.lib
-- Installing: C:/Program Files (x86)/SEAL/lib/cmake/SEAL-4.1/SEALTargets.cmake
-- Installing: C:/Program Files (x86)/SEAL/lib/cmake/SEAL-4.1/SEALTargets-release.cmake
-- Installing: C:/Program Files (x86)/SEAL/lib/cmake/SEAL-4.1/SEALConfig.cmake
-- Installing: C:/Program Files (x86)/SEAL/lib/cmake/SEAL-4.1/SEALConfigVersion.cmake
-- Installing: C:/Program Files (x86)/SEAL/include/SEAL-4.1/gsl
-- Installing: C:/Program Files (x86)/SEAL/include/SEAL-4.1/gsl/algorithm
-- Installing: C:/Program Files (x86)/SEAL/include/SEAL-4.1/gsl/assert
-- Installing: C:/Program Files (x86)/SEAL/include/SEAL-4.1/gsl/byte
-- Installing: C:/Program Files (x86)/SEAL/include/SEAL-4.1/gsl/gsl
-- Installing: C:/Program Files (x86)/SEAL/include/SEAL-4.1/gsl/gsl_algorithm
-- Installing: C:/Program Files (x86)/SEAL/include/SEAL-4.1/gsl/gsl_assert
-- Installing: C:/Program Files (x86)/SEAL/include/SEAL-4.1/gsl/gsl_byte
-- Installing: C:/Program Files (x86)/SEAL/include/SEAL-4.1/gsl/gsl_narrow
-- Installing: C:/Program Files (x86)/SEAL/include/SEAL-4.1/gsl/gsl_util
-- Installing: C:/Program Files (x86)/SEAL/include/SEAL-4.1/gsl/narrow
-- Installing: C:/Program Files (x86)/SEAL/include/SEAL-4.1/gsl/pointers
-- Installing: C:/Program Files (x86)/SEAL/include/SEAL-4.1/gsl/span
-- Installing: C:/Program Files (x86)/SEAL/include/SEAL-4.1/gsl/span_ext
-- Installing: C:/Program Files (x86)/SEAL/include/SEAL-4.1/gsl/string_span
-- Installing: C:/Program Files (x86)/SEAL/include/SEAL-4.1/gsl/util
-- Installing: C:/Program Files (x86)/SEAL/include/SEAL-4.1/seal/batchencoder.h
-- Installing: C:/Program Files (x86)/SEAL/include/SEAL-4.1/seal/ciphertext.h
```

« 로컬 디스크 (C:) > Program Files (x86) > SEAL				SEAL 검색
이름	수정된 날짜	유형	크기	
include	2023-07-02 오전 4:05	파일 폴더		
lib	2023-07-02 오전 4:05	파일 폴더		

<정상적으로 설치됨>

여기서 include 에는 헤더파일이, lib 에는 라이브러리 파일이 들어있습니다. 이 파일들을 Visual Studio 에 추가 라이브러리와 헤더파일 경로를 추가하면 이제 #include 를 통해 Seal 라이브러리를 사용할 수 있습니다.

추가 포함 디렉터리	C:\Program Files (x86)\SEAL\include\SEAL-4.1
추가 #using 디렉터리	
추가 DLL 디렉터리	

<헤더파일 경로추가>

추가 종속성	seal-4.1.lib;\$(CoreLibraryDependencies);%(AdditionalDependencies)
모든 기본 라이브러리 무시	

<추가 종속성으로 어떠한 lib 파일을 추가할지 미리 알려줌>

사용사 난위 디니택션	아니요
추가 라이브러리 디렉터리	C:\Program Files (x86)\WSEALWlib
라이브러리 조소서 링크	예

<라이브러리 파일 경로>

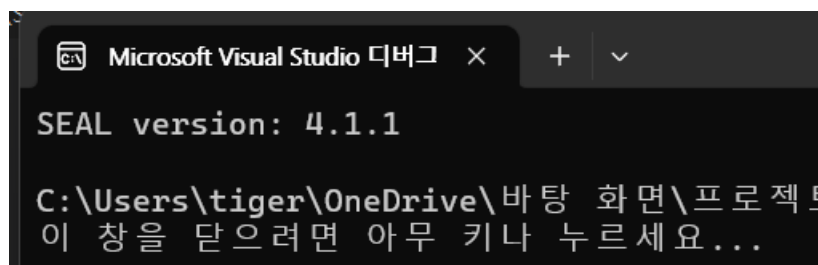
플랫폼 도구 집합	Visual Studio 2022 (v143)
C++ 언어 표준	ISO C++17 표준(/std:c++17)
C 언어 표준	기본값(레거시 MSVC)

<언어 표준 17로 설정>

```
#include <iostream>
#include <seal/seal.h>

int main() {
    seal::EncryptionParameters params(seal::scheme_type::bfv);
    std::cout << "SEAL version: " << SEAL_VERSION_MAJOR << "." << SEAL_VERSION_MINOR << "." << SEAL_VERSION_PATCH << std::endl;
    return 0;
}
```

제대로 설치되었는지 확인합니다. 이 코드는 라이브러리가 제대로 동작하는지를 보여주고 SEAL 버전을 확인합니다.



```
Microsoft Visual Studio 디버그
SEAL version: 4.1.1

C:\Users\tiger\OneDrive\바탕 화면\프로젝트
이 창을 닫으려면 아무 키나 누르세요...
```

<결과>