DSP Lab 2016-17, JNI using Intel-SGX

Clindo Devassy K, Subhadeep Manna

February 17, 2017

Contents

1 Abstract

Software applications frequently need to work with private information. The security of these information is crucial. Intel have introduced the Intel Software Guard Extensions (Intel SGX). A method to secure our code and data from disclosure. Intel SGX could be coded using C/C++. This project creates a JNI implementation for SGX calls and with those implementation creates an arithmetic evaluator. This arithmetic evaluator can be accessed remotely.

2 Application Flow

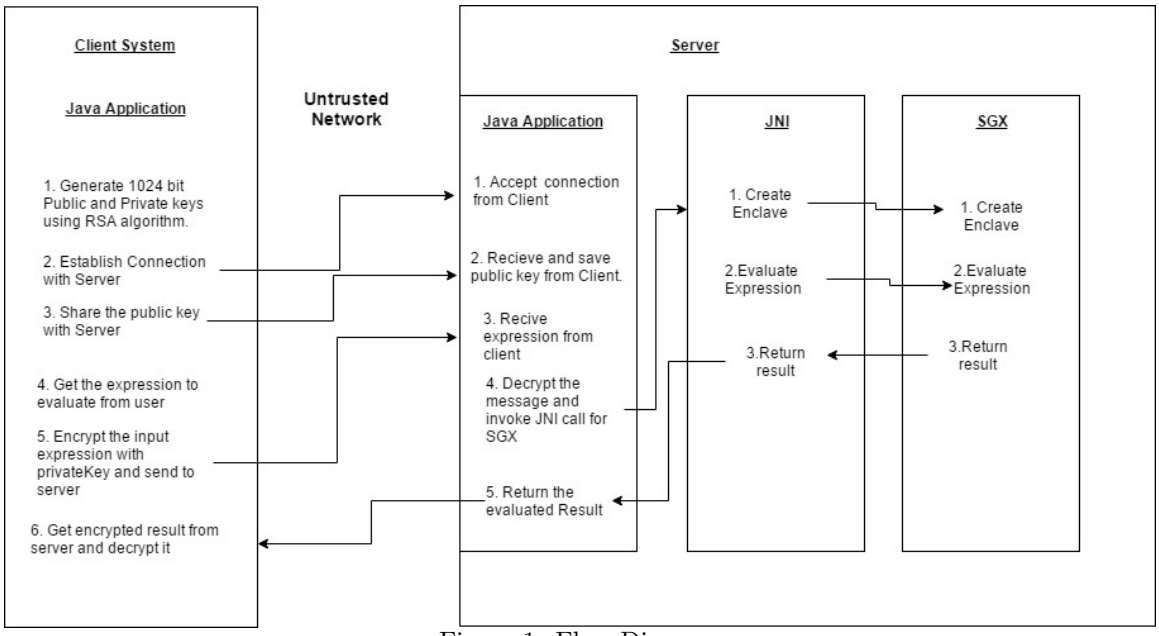

Figure 1: Flow Diagram

3 Prerequisites

1. JDK 1.8 is installed on both client and server
2. The Server should be a SGX enabled machine.

4 SGX Server Usage

1. Copy the server side project for SGX
2. Run 'make'
3. export LD_LIBRARY_PATH='/opt/intel/sgxsdk/sdk_libs/'
4. javac -cp . JavaApp.java
5. java -Djava.library.path=. JavaApp

5 Java Client-Server Usage

5.1 Server

1. Copy the file Server.java
2. Compile it using javac Server.java
3. Run the server with java Server

5.2 Client

1. Copy Client.java and EncryptionUtil.java to same folder
2. Compile client using javac Client.java
3. Run the server with java Client

Note: We have save common erros and its solutions inside the Errors and Troubleshooting directory in the repository

6 Pending tasks

1. Integrating the java client-server with SGX server
2. Remote Attestation of the server
3. Exception Handling in Client-Server Communication

7 Questions

1. Should the public key received by the server be saved in side the enclave or outside the enclave?
2. Should we transfer the computed result using OCALL outside the enclave? Currently we are just making a test OCALL and printing it directly.