Sean Evans
CS 6058
Data / Security and Privacy
Spring 2018
Project 1

Description

The project was implemented using C++, and uses the C++ Standard Library and Standard Template Library (STL), and Boost Libraries. The encryption and decryption functionality is implemented using STL algorithms. The key generation functionality is implemented using psuedorandom number generation functionality in the C++ Standard Library. The build system for the project was implemented using CMake. The project was implemented and tests were run on a Windows 10 system with the Cygwin environment installed.

Key Distribution Testing

In order to prove that the keys generated are uniformly distributed, the probability of each key in the key space must be calculated. If the probability of each of the keys are 1 / K, where K is the number of possible keys, then the key generator generated uniformly distributed keys.

Test data from program output is included below.

```
key distribution test parameters
 security parameter = 3
 iterations = 5000

key distribution test result data
 key | count  | probability | histogram
 --- | ------ | ----------- | ---------
 0x0 | 0x0285 | 0.129       | ********************************************************
 0x1 | 0x0275 | 0.126       | ******************************************************
 0x2 | 0x0257 | 0.120       | **************************************************
 0x3 | 0x0289 | 0.130       | *********************************************************
 0x4 | 0x027D | 0.127       | *******************************************************
 0x5 | 0x0274 | 0.126       | ******************************************************
 0x6 | 0x025C | 0.121       | *************************************************
 0x7 | 0x0261 | 0.122       | **************************************************
```

Per this test data, we can conclude that the key generation function generates uniformly distributed keys as the probabilities of each key are approximately 12.5%, or 1 / 8, as would be expected from a key space of 2^3.

Run Time Testing

The mean run time of the encryption function on 128-bits of data on the test machine was approximately 308 nanoseconds.

Test data from program output is included below.

```
run time test parameters
 security parameter = 128 bits
 iterations         = 5000

run time test results
 min run time      =        200 ns
 max run time      =       3400 ns
 mean run time     =        308 ns
 run time variance =       7400 ns
 median run time   =        300 ns
 total run time    =    1543700 ns
```