Учреждение образования

«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Исследование криптографических хеш-функций

Студент: Николаева Е.В.

ФИТ 3 курс 5 группа

Преподаватель:

Савельева Маргарита Геннадьевна

Минск 2023

1. **Описание приложения**

Приложение написано на языке программирования C# и позволяет выполнить 2 задачи:

* хешировать входное сообщение;
* оценивать скорость выполнения алгоритма;

1. **Методика выполнения поставленных задач**

MD5 – алгоритм хеширования, в котором выполняются 4 этапа по 16 раз. Каждая операция в этапе представляет собой нелинейную функцию над тремя из *a*, *b*, *c* и *d*. Затем результат добавляется к четвертой переменной, подблоку текста *m* и константе *t*. Далее результат циклически сдвигается вправо на переменное число s бит и добавляет результат к одной и переменных *a*, *d*, *c* и *d*. В конце результат заменяет одну из этих переменных. Результат хеширования – конкатенация последних значений указанных переменных – 32×4 = 128 бит. Результат работы приложения для алгоритма хеширования MD5 представлен на рисунке 2.1

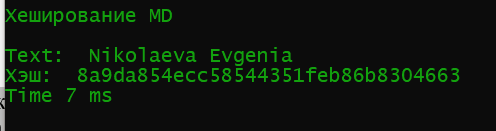


Рис. 2.1 – Реализация генерации тайного ключа

Были использованы библиотеки System.Security.Cryptography. Для хеширования были использованы функции изображены 2.2.

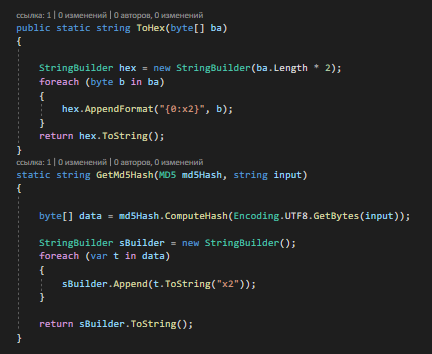


Рис. 2.2 – Реализация вычисления открытого ключа

SHA256 – хеш-функция из семейства алгоритмов SHA-2, которая предназначена для создания дайджестов для сообщений произвольной длины. Длина дайджеста – 256 бит. Исходное сообщение дополняется до нужной длины, а затем разбивается на блоки. Каждый блок – на 16 слов. Каждый блок сообщения пропускается через 64 итерации. На каждой итерации 2 слова преобразуются, функцию преобразования задают остальные слова. Результаты обработки каждого блока складываются, сумма – значение хеш-функции. Т.к. инициализация внутреннего состояния производится результатом обработки предыдущего блока, то нет возможности обрабатывать блоки параллельно.

Результат выполнения алгоритма SHA256 представлен на рисунке 2.3.

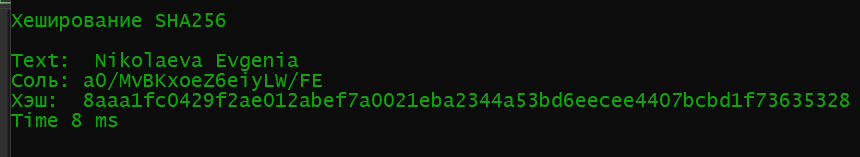


Рис. 2.3 – Результат алгоритма SHA256

Функция для хеширования с помощью алгоритма SHA256 представлена на рисунке 2.4.

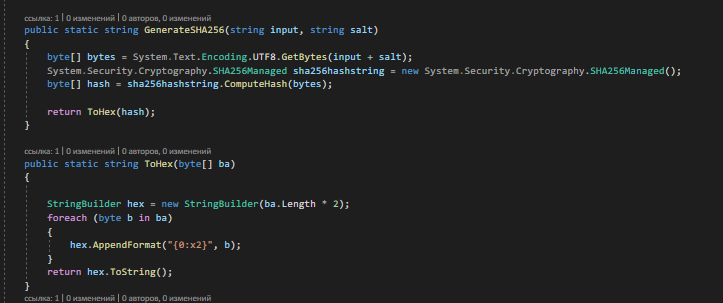


Рисунок 2.4 – Функция хеширования SHA256

В алгоритмах хеширования часто используется такое понятие как соль. Соль – строка данных, которая передается хеш-функции вместе с входными данными для вычисления хеша. Используется для усложнения определения прообраза хеш-функции методом перебора по словарю возможных значений, включая радужные атаки. Позволяет скрыть факт использования одинаковых прообразов при использовании разной соли.

Таким образом, были реализованы все поставленные задачи. Были исследованы асимметричные шифры.

1. **Оценить быстродействие выбранного алгоритма хеширования.**

Рассмотрим рисунок 2.5, где несколько раз произведены функции хеширования. Сразу же видно, что SHA256 будет намного быстрее и эффективнее. Построим график.

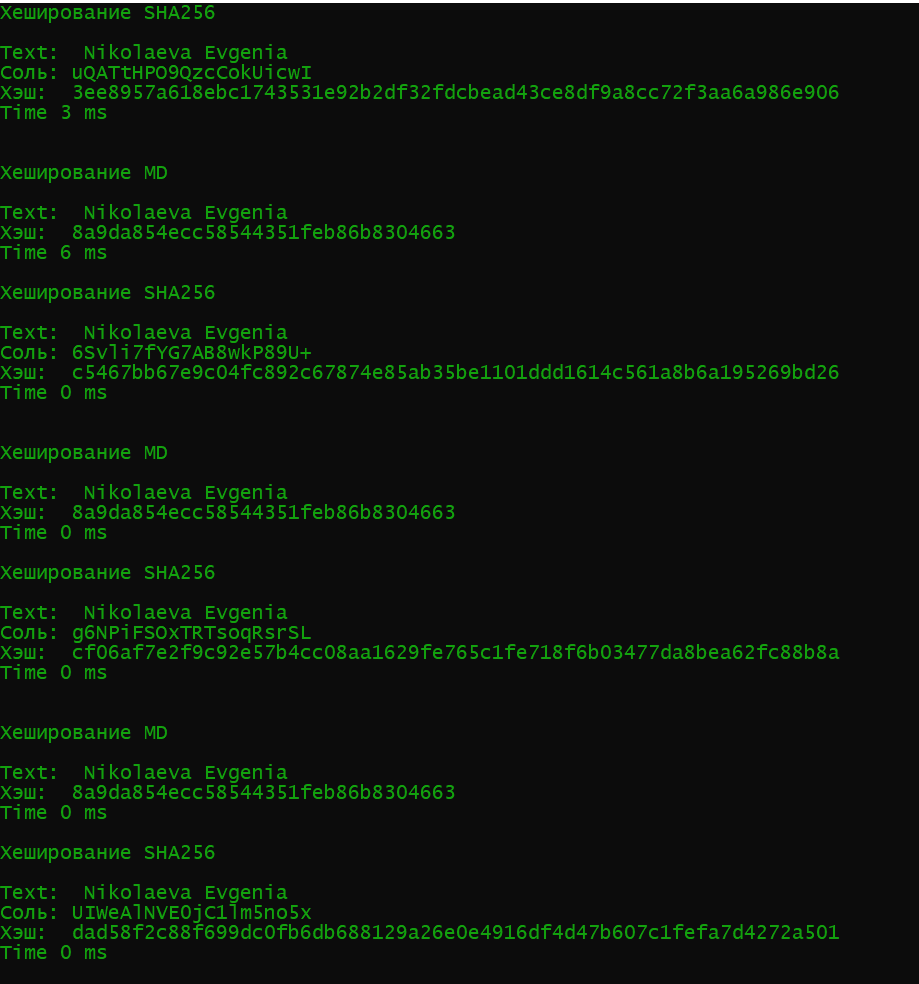


Рисунок 2.5 – Результат работы приложения

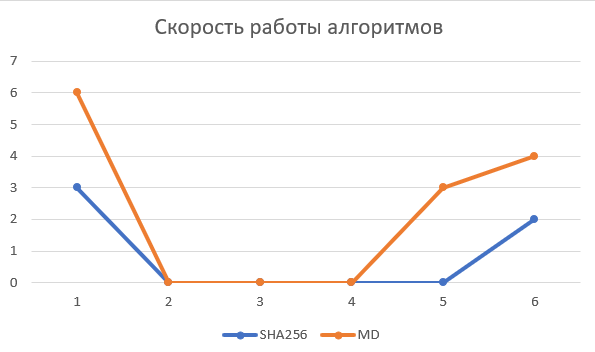


Рисунок 2.5 – График, описывающий скорость работы алгоритмов

Сразу на рисунке видно, что SHA256 быстрее, но так же он является более безопасным, чем MD5.

**Вывод**

В ходе лабораторной работы были приобретены навыки хеширования используя алгоритмы MD5 и SHA256. Были изучены основные принципы работы хеширования.

Также было разработано приложение, на языке программирования C#, для реализации задач, связанных с хешированием данных.