Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н. Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н. Э. Баумана)

ФАКУЛЬТЕТ «Информатика, искусственный интеллект и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

ОТЧЕТ

по лабораторной работе № 4

по курсу «Защита Информации»

на тему: «Цифровая подпись»

Вариант № 1

Студент ИУ7-71Б _____ Корниенко К. Ю.
(Группа) (Подпись, дата) (И. О. Фамилия)

Преподаватель _____ Чиж И. С.
(Подпись, дата) (И. О. Фамилия)

2024 г.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ

Цель лабораторной работы — разработать программу, создающую цифровую подпись в соответствии с алгоритмами «MD5» и «RSA», и осуществляющую проверку подлинности документов по созданным подписям.

Задачи лабораторной работы:

1. провести анализ алгоритмов «MD5» и «RSA»;
2. описать процедуру создания цифровой подписи;
3. описать процедуру проверки подлинности документа по цифровой подписи;
4. релизовать программу.

1 Аналитический раздел

1.1 Алгоритм шифрования «RSA»

RSA — криптографический алгоритм с открытым ключом, основывающийся на вычислительной сложности задачи факторизации.

Алгоритм:

1. Выбираем два случайных простых числа p и q.
2. Вычисляем их произведение: $N = p \times q$.
3. Вычисляем функцию Эйлера: $\phi(N) = (p-1) \times (q-1)$.
4. Выбираем число e (обычно простое, но необязательно), которое меньше $\phi(N)$ и является взаимно простым с $\phi(N)$.
5. Ищем число d, обратное числу e по модулю $\phi(N)$:

$$d \times e \equiv 1 \pmod{\phi(N)}.$$

Найти его можно через расширенный алгоритм Евклида.

1.2 Алгоритм хеширования «MD5»

MD5 — 128-битный алгоритм хеширования, разработанный профессором Рональдом Л. Ривестом из Массачусетского технологического института (Massachusetts Institute of Technology, MIT) в 1991 году. Предназначен для создания «отпечатков» или дайджестов сообщения произвольной длины и последующей проверки их подлинности. Широко применялся для проверки целостности информации и хранения хешей паролей, однако признан небезопасным из-за малой длины получаемого хэша и простотой самого алгоритма.

2 Конструкторская часть

2.1 Разработка алгоритма

На рисунке 2.1 представлена схема создания и проверки цифровой подписи.

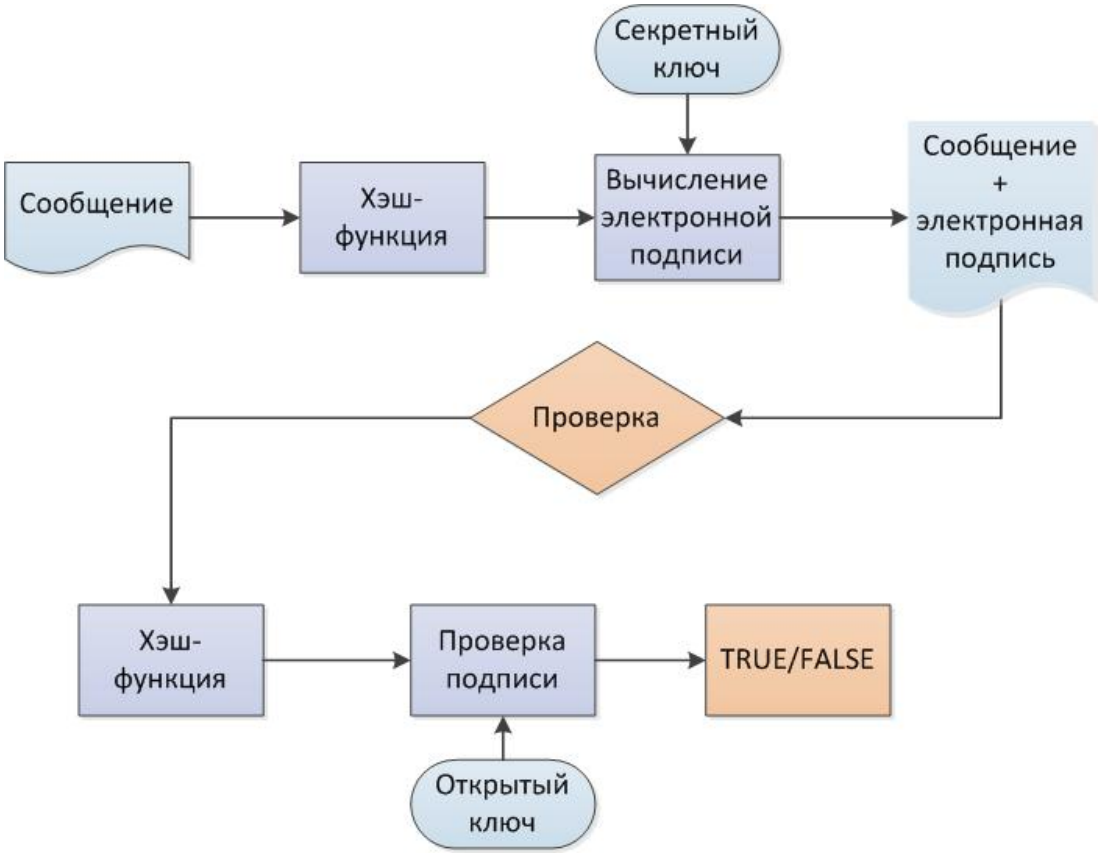

Рисунок 2.1 – Схема создания и проверки цифровой подписи

3 Технологическая часть

3.1 Средства реализации

Для реализации ПО был выбран язык C++ [1]. В данном языке есть все требующиеся инструменты для данной лабораторной работы. В качестве среды разработки была выбрана среда VS code [2].

3.2 Реализация алгоритма

На листинге ниже приведен алгоритм создания и проверки цифровой подписи.

Листинг 3.1 – Алгоритм создания и проверки цифровой подписи

```
bool checkSignature(const char* filename) {
    auto input = readFile(filename);
    const auto hash = md5(input);

    // Signature creation
    std::vector<long long> vec(hash.size());
    for (const auto c : hash)
        vec.push_back(static_cast<long long>(c));

    // Keys
    Keys keys = calculateRSAKeys();
    const auto sign = cryptMessage(vec, keys._private);

    const auto input2 = readFile(filename);

    // Signature verification
    const auto originalHash = md5(input2);

    auto signatureHash = encryptMessage(sign, keys._public);
    signatureHash = extractHash(signatureHash);

    return originalHash == signatureHash;
}
```

ЗАКЛЮЧЕНИЕ

В данной лабораторной работе:

— проведен анализ работы алгоритмов «RSA» и «MD5»;

— описан алгоритм создания и проверки цифровой подписи;

— реализован описанный алгоритм.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. *Josuttis N. M.* The C++ standard library: a tutorial and reference. — 2012.
2. *Code V. S.* Visual studio code // línea]. Available: https://code. visualstudio. com. — 2019.