Учреждение образования

«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет информационных технологий

Кафедра информационных систем и технологий

Специальность 1-40 01 01 Программное обеспечение информационных технологий

**Отчёт по лабораторной работе №12**

**«Исследование алгоритмов генерации и верификации электронной цифровой подписи»**

Исполнитель:

Студент 3 курса группы 4

Гурина К. С.

Руководитель:

Ассистент Сазонова Д. В.

1. **Цель и задачи работы**

**Цель:** изучение алгоритмов генерации и верификации электронной цифровой подписи и приобретение практических навыков их реализации.

**Задачи:**

1. Закрепить теоретические знания по алгебраическому описанию и алгоритмам реализации операций генерации и верификации электронной цифровой подписи (ЭЦП).

2. Получить навыки практической реализации методов генерации и верификации ЭЦП на основе хеширования подписываемых сообщений и алгоритмов RSA, Эль-Гамаля и Шнорра, а также DSA.

3. Разработать приложение для реализации заданных алгоритмов генерации и верификации ЭЦП.

4. Оценить скорость генерации и верификации ЭЦП.

5. Результаты выполнения лабораторной работы оформить в виде описания разработанного приложения, методики выполнения экспериментов с использованием приложения и результатов эксперимента

**2. Теоретические сведения**

Электронная цифровая подпись – контрольная характеристика сообщения, которая вырабатывается с использованием личного ключа, проверяется с использованием открытого ключа, служит для контроля целостности и подлинности сообщения и обеспечивает невозможность отказа от авторства.

Таким образом, ЭЦП выполняет те же функции, что и собственноручная (поставленная «от руки») подпись:

• аутентифицирование лица, подписавшего сообщение;

• контроль целостности подписанного сообщения;

• защита сообщения от подделок;

• доказательство авторства лица, подписавшего сообщение, если это лицо отрицает свое авторство.

Важнейшие отличительные особенности ЭЦП:

• ЭЦП представляет собой бинарную последовательность (в отличие от графического образа, каковым является подпись от руки);

• указанная бинарная последовательность зависит от содержания подписываемого сообщения.

Электронная цифровая подпись – бинарная (или в ином виде) последовательность символов, являющаяся реквизитом электронного документа, зависящая от содержания этого документа и предназначенная для подтверждения целостности и подлинности электронного документа.

ЭЦП на основе RSA. Здесь можно рассматривать две ситуации:

• сообщение Мo подписывается и передается в открытом (незашифрованном) виде;

• сообщение Мo подписывается и передается в зашифрованном виде.

При этом подпись S вычисляется на основе соотношения:

S ≡ (H(Mo))dо modno.

Передаваемое сообщение М' = Мo||S. Соответственно, операция расшифрования на приемной стороне (получатель анализирует Мп||S) производистя с известной модификацией ключей:

H(Mo) ≡ (S) ео modno.

Далее вычисляется Н(Mп). Если H(Mo) = H(Mп), подпись верифицирована.

Если подписываемое сообщение М(М') также должно передаваться в зашифрованном виде, то обычно М' шифруется на стороне отправителя стандартным образом: с помощью открытого ключа получателя (еп и nп), который перед основным процессом верификации подписи расшифровывает послание своим тайным ключом: dп и nп. Далее осуществляются вычисления и анализ, как и в первом случае.

Основное отличие в применении расчетов в алгоритме Эль-Гамаля состоит в том, что результатом зашифрования является только одна пара чисел, а не пара для каждого блока исходного сообщения. Причем в рассматриваемом случае таким сообщением является хеш подписываемого документа: Н(Mо).

ЭЦП на основе DSA является одним из известных, нередко и сейчас применяемых. В алгоритме используются следующие параметры: p – простое число длиной от 64 до 1024 битов (число должно быть кратно 64); q – 160-битный простой множитель (р – 1). Далее вычисляется число g. Числа p, q, v могут использоваться группой лиц. Еще один элемент открытого ключа у вычисляется в соответствии с выражением

y ≡ gx mod p

где x < q; х – закрытый ключ.

Общая схема генерации и верификации ЭЦП приведена на рис. 1.1. Здесь H(m) – хеш подписываемого сообщения. ЭЦП состоит из двух чисел: r и s. Число k здесь играет такую же роль, что и одноименный параметр в шифре Эль-Гамаля.

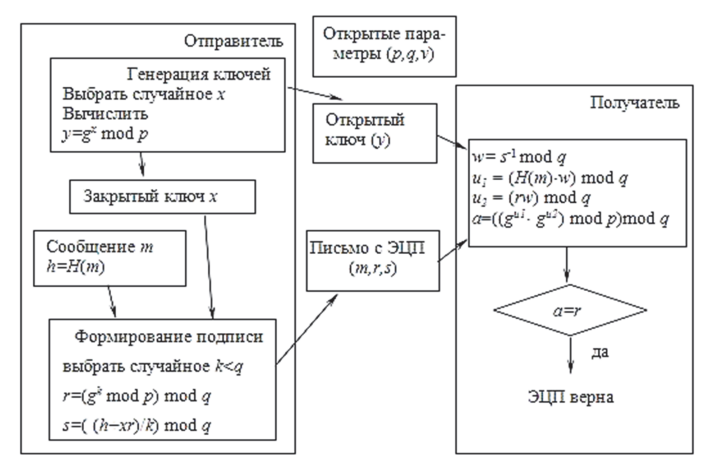


Рисунок 1.1 – Общая схема генерации и верификации ЭЦП DSA

Алгоритм ЭЦП К. Шнорра (К. Schnorr) является вариантом алгоритма ЭЦП Эль-Гамаля. Одной из особенностей ЭЦП Эль-Гамаля является то, что число p должно быть очень большим, чтобы сделать действительно трудной проблему дискретного логарифма. Рекомендуемая длина p должна составлять по крайней мере 1024 бита. Чтобы уменьшить размер подписи, Шнорр предложил новую схему, но с уменьшенным размером подписи.

Ключевая информация: p – простое число в диапазоне от 512 до 1024 битов; q –160-битное простое число, делитель (p – 1); любое число g (g ≠ 1) такое, что

gq ≡ 1 mod p.

Числа p, g, q являются открытыми и могут применяться группой пользователей. Выбирается число х < q (х является тайным ключом) и вычисляется последний элемент открытого ключа:

y ≡ g–х mod p.

Секретный ключ имеет длину не менее 160 битов.

Для подписи сообщения Мо выбирается случайное число k (1 < k < q) и вычисляет параметр а:

а ≡ gk mod p.

Далее вычисляется хеш от канкатенации сообщения Мо и числа а:

h = H(Mo||a).

Обратим внимание, что хэш-функция непосредственно не применяется к сообщению. Создается хеш-образ подписываемого сообщения, спереди присоединенного к числу а. Далее вычисляется значение b:

b ≡ (k + xh) mod q

Получателю отправляются М' = Мо||S; S = {h, b}.

Для проверки подписи получатель вычисляет

Х ≡ gb yh (mod p)

Затем он проверяет выполнение равенства:

h = Н(Mп||Х)

Подпись достоверна, если равенство выполняется. Основные вычисления для генерации подписи могут производиться предварительно. Порядок величин х и h – около 140 двоичных разрядов, порядок числа k – около 70–72 разрядов. С учетом этого сложность операций умножения можно считать ничтожно малой по сравнению с модульным умножением в схеме RSA.

**3. Ход работы**

**Практическое задание:**

1. Разработать оконное приложение, реализующее один из алгоритмов хеширования из указанного преподавателем семейства (MD или SHA; или иного). При этом можно воспользоваться доступными готовыми библиотеками. Язык программирования – на свой выбор. Приложение должно обрабатывать входные сообщения, длина которых определяется спецификацией на реализуемый алгоритм.

2. Оценить быстродействие выбранного алгоритма хеширования.

3. Результаты оформить в виде отчета по установленным правилам.

**Ход работы**

Алгоритм RSA основан на сложности факторизации больших чисел. В процессе работы используются две большие простые числа, их произведение и значения, взаимно простые с произведением. Для реализации генерации цифровой подписи на основе алгоритма RSA была написана функция createDigitalSignature, представленная на рисунке 3.1

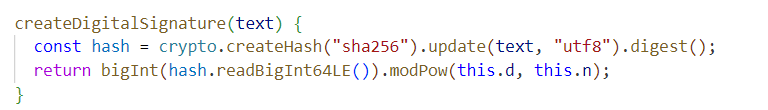


Рисунок 3.1 – Генерация ЭЦП алгоритмом RSA

Для верификации подписи была разработана функция verifyDigitalSignature, представленная на рисунке 3.2.

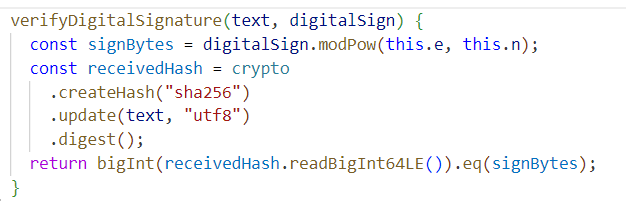


Рисунок 3.2 – Верификация ЭЦП алгоритмом RSA

Функция verifyDigitalSignature хеширует исходное сообщение, декодирует цифровую подпись с помощью публичного ключа, а затем сравнивает полученное значение с хешем сообщения.

Результат работы представлен на рисунке 3.3.

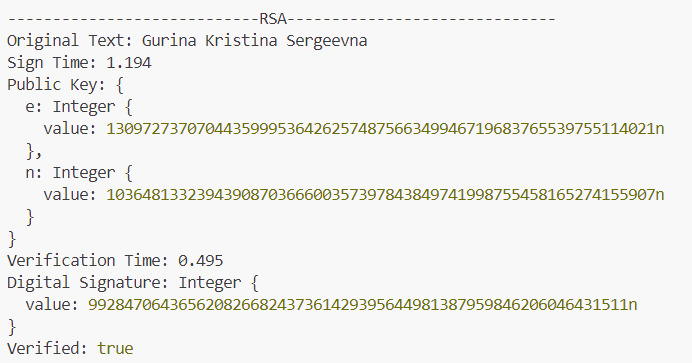


Рисунок 3.3 – Результат работы алгоритма RSA

Алгоритм ElGamal основан на сложности вычисления дискретного логарифма. Включает в себя использование большого простого числа и генератора группы.

Была написана функция getPublicKey, которая возвращает публичный ключ, состоящий из p, g и y. Функция createDigitalSignature хеширует сообщение с помощью SHA-256, генерирует случайное число k, взаимно простое с p-1, Вычисляет первую часть подписи как g в степени k по модулю p. Вычисляет вторую часть подписи, используя хеш сообщения и секретный ключ x. Код функции createDigitalSignature представлен на рисунке 3.4.



Рисунок 3.4 – Генерация ЭЦП алгоритмом Эль-Гамаля

Функция verifyDigitalSignature хеширует исходное сообщение, декодирует цифровую подпись с помощью публичного ключа, а затем сравнивает полученное значение с хешем сообщения. Код функции createDigitalSignature представлен на рисунке 3.5.

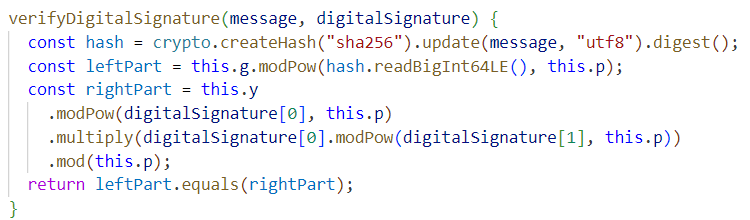


Рисунок 3.5 – Верификация ЭЦП алгоритмом Эль-Гамаля

Результат работы представлен на рисунке 3.6.

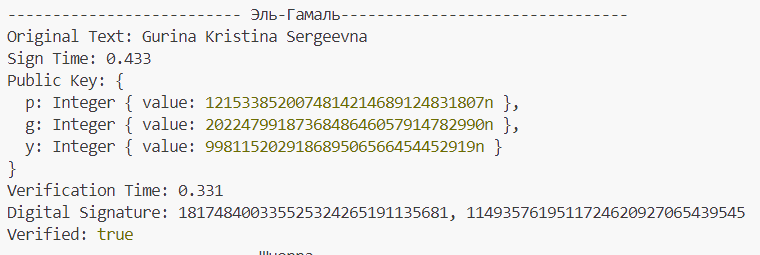


Рисунок 3.6 – Результат работы алгоритма Эль-Гамаля

Алгоритм Шнорра основан на сложности дискретного логарифма в группе. Использует большое простое число и генератор подгруппы.

Функция getPublicKey возвращает публичный ключ, состоящий из p, q, g и y.

Функция generateDigitalSignature генерирует случайное число k, вычисляет временное значение a как g в степени k по модулю p, хеширует сообщение с добавленным значением a, вычисляет первую часть подписи как хеш сообщения, вычисляет вторую часть подписи, используя секретный ключ x и случайное число k.

Код функции createDigitalSignature представлен на рисунке 3.7.



Рисунок 3.7 – Генерация ЭЦП алгоритмом Шнорра

Функция verifyDigitalSignature вычисляет значение x, используя публичный ключ и подпись, хеширует сообщение с добавленным значением x, сравнивает первую часть подписи с вычисленным хешем

Результат работы представлен на рисунке 3.8.

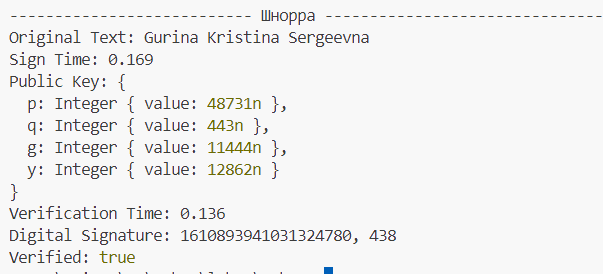


Рисунок 3.8 – Результат работы алгоритма Шнорра

**Вывод**

В ходе выполнения лабораторной работы был изучен принцип генерации и верификации ЭЦП на основе алгоритмов RSA, Эль-Гамаля и Шнорра. Было разработано приложение, выполняющее генерацию и верификацию ЭЦП на основе данных алгоритмов и была оценена скорость генерации и верификации подписи.