### Министерство науки и высшего образования Российской Федерации

1. Санкт-Петербургский политехнический университет Петра Великого
2. —
3. Институт компьютерных наук и кибербезопасности
4. Высшая школа кибербезопасности

**ЛАБОРАТОРНАЯ РАБОТА № 1**

1. **«Основы криптосистемы RSA»**
2. по дисциплине «Криптографические методы защиты информации»
3. Выполнил
4. студент гр. 5151001/00201 Устюгов А. А.

<*подпись*>

1. Проверил:
2. старший преподаватель, к.т.н. Тулинова А. В.

<*подпись*>

1. Санкт-Петербург
2. 2024

# Цель работы

Изучение криптосистемы RSA, реализация алгоритмов зашифрования и расшифрования сообщений, формирования и проверки электронной цифровой подписи.

# Порядок выполнения работы

1. Получить у преподавателя вариант задания и разработать программу П-1, которая:

* выполняет зашифрование и расшифрование файла с использованием алгоритма RSA. В режиме зашифрования программа должна принимать на вход файл сообщения (произвольного формата), открытый ключ RSA, ключ симметричного алгоритма; на выходе − возвращать зашифрованный файл, заголовок которого соответствует нотации ASN.1. Зашифрование файла необходимо производить с использованием вспомогательного симметричного алгоритма AES-256 в режиме CBC, реализация симметричного алгоритма не требуется. Ключ шифрования симметричного алгоритма – случайный или задан в программе. Случайный ключ шифрования алгоритма AES (32 байта) необходимо представить как число для шифрования алгоритмом RSA, порядок байтов – MSB. Старшие неиспользуемые цифры (байты) числа следует считать нулевыми. В режиме расшифрования программа П-1 должна принимать на вход зашифрованный файл, закрытый ключ RSA; на выходе – возвращать расшифрованное сообщение.
* выполняет формирование и проверку электронной подписи с использованием алгоритма RSA. При реализации алгоритма электронной подписи рекомендуется использовать хэш-алгоритм SHA-256. Поскольку хэш-образ, вычисленный по алгоритму SHA-256, может быть длиннее модуля RSA, при проверке подписи следует сравнивать значение ℎ(𝑚)(𝑚𝑜𝑑 𝑛) с подписью 𝑠 сообщения 𝑚. В режиме формирования подписи программа П-1 должна принимать на вход файл, который необходимо подписать, ключ подписи RSA; на выходе – возвращать отдельный файл подписи. В режиме проверки подписи программа должна принимать на вход сообщение и файл подписи; на выходе – возвращать результат «Подпись принимается» или «Подпись неверна».

# Ход работы

Внутри П-1 была реализована процедура, генерирующая параметры криптосистемы RSA следующим образом:

* Генерируется два больших простых числа p, q
* Генерируется простое число e, 1 < e < (p-1)(q-1)-1
* С помощью расширенного алгоритма Евклида вычисляется значение d

Зашифрование и расшифрование данных происходит с помощью симметричного алгоритма AES-256. Ключ алгоритма генерируется случайным образом с помощью функции get\_random\_bytes(32). Вектор инициализации задан в программе. Синхропосылка состоит из нулей. Зашифрование и расшифрование осуществляется функциями encrypt и decrypt соответственно.

Для зашифрования ключа используется RSA. В метод encrypt() передается сам ключ и открытый ключ RSA. Для зашифрования необходимо возвести сообщение в степень открытого ключа по модулю n. Для расшифрования возвести шифртекст в степень закрытого ключа по модулю n.

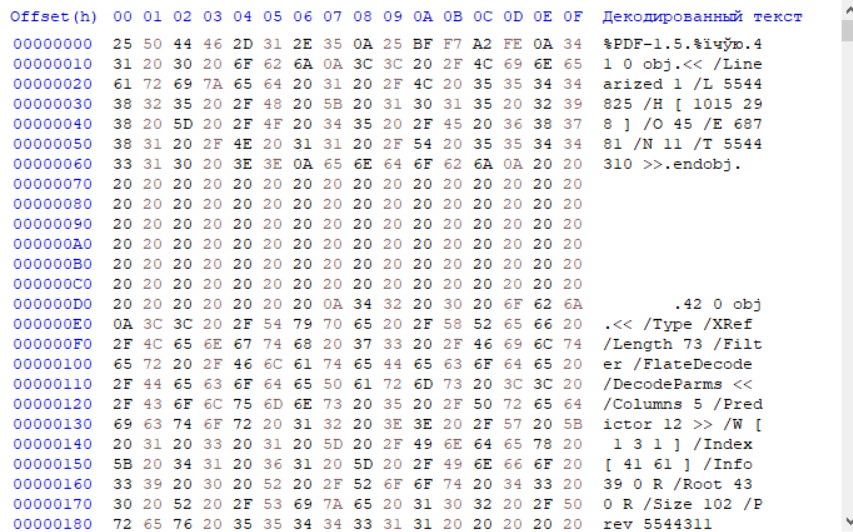


Рисунок 1 – файл для шифрования

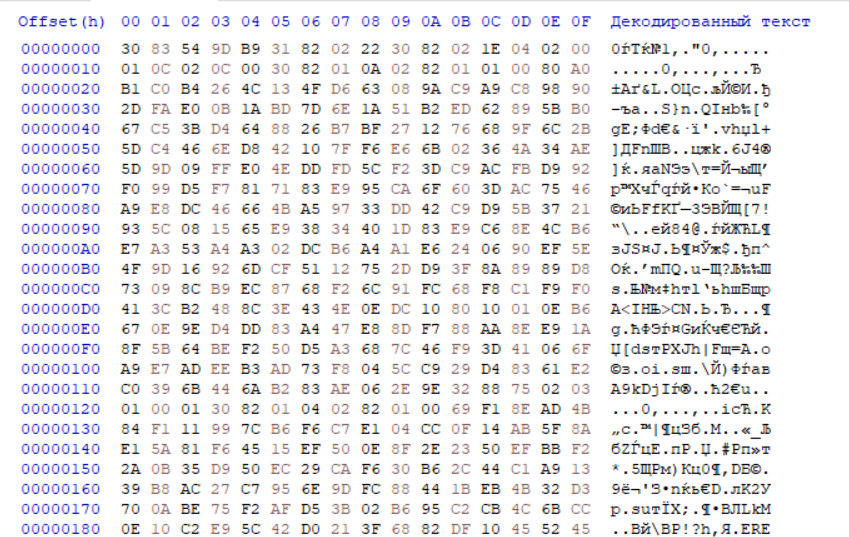


Рисунок 2 – зашифрованный файл

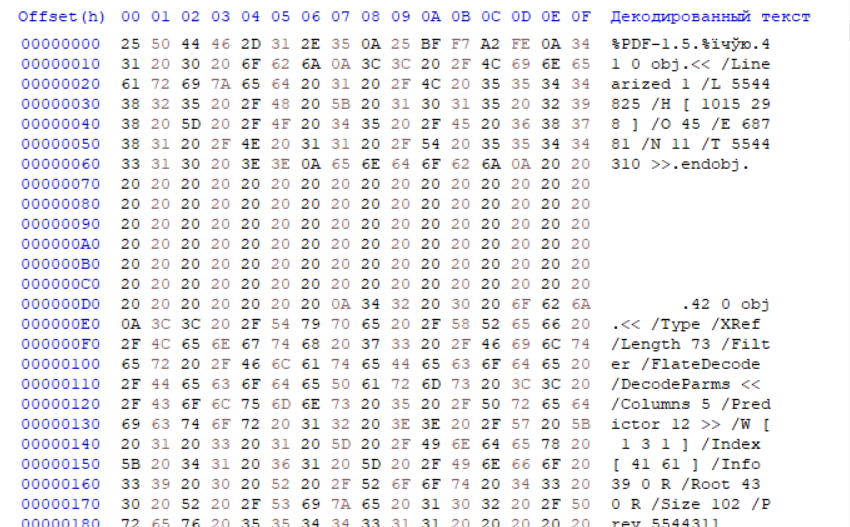


Рисунок 3 – расшифрованный файл

Для формирования подписи используется алгоритм SHA-256 и закрытый ключ алгоритма RSA. Для формирования подписи необходимо посчитать хэш от сообщения и затем возвести его в степень закрытого ключа по модулю n. Для проверки подписи необходимо возвести подпись в степень открытого ключа по модулю n, посчитать хэш от сообщения и сравнить их.

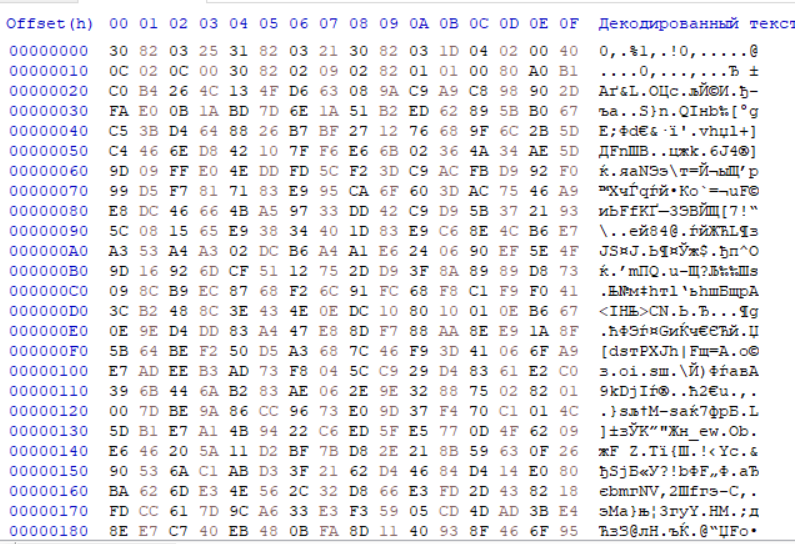


Рисунок 4 – файл подписи

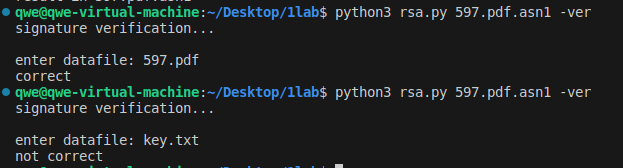


Рисунок 5 – проверка подписи

# Ответы на контрольные вопросы

1. Какие задачи положены в основу безопасности системы RSA?

Криптосистема RSA основана на сложности задачи разложения больших чисел на простые множители.

2. Показать, что схема RSA работает корректно для любого сообщения 𝑚 ∈ ℤ⁄𝑛ℤ.

Покажем, что для выполняется .

Всего возможно два случая:

Поскольку числа *e* и *d* являются взаимно обратными по модулю 𝜑(𝑛)=(p-1)(q-1), то есть для некоторого целого *k* верно следующее:

Тогда

Второй случай:

А значит и

Таким образом, для всех m выполняется:

Аналогично показывается то, что для выполняется .

Тогда, из КТО следует:

3. Доказать, что задача разложения числа n на множители и задача вычисления функции Эйлера 𝜑(𝑛) полиномиально эквивалентны.

Если известно разложение числа n, то значение 𝜑(𝑛)=(p-1)(q-1) вычислимо с не более чем квадратичной сложностью.

Если известны значения n и 𝜑(𝑛), то делители p и q числа n являются целочисленными корнями квадратного уравнения

Сложность решения этого уравнения оценивается кубическим полиномом от .

4. Показать, что схема RSA обладает свойством гомоморфности относительно операции умножения.

Определение гомоморфизма:

В нашем случае, , где x – открытый текст, k открытый ключ, n – модуль RSA.

# Вывод

В данной лабораторной работе были изучены основы и варианты практического использования криптосистемы RSA. Была реализована программа, позволяющая шифровать и расшифровать сообщения, формировать и проверять электронную цифровую подпись. Хранение параметров криптосистем в открытом виде внутри кода программы является небезопасным, поэтому было принято решение генерировать параметры внутри разработанной программы.