Цель работы: реализовать криптосистему RSA. При постановке подписи использовать хеш-функцию MD5.

1 Описание

1.1 RSA

1.1.1 Генерация ключей

Выбирается два случайных числа p и q заданного размера. Затем вычисляется $n = pq$ и значения функции Эйлера $\varphi(n)$. Выбирается целое e, такое, что $1 < e < \varphi(n)$ и взаимно простое с $\varphi(n)$.

Вычисляется обратное по умножение к e d, т.е. $de \equiv 1 \bmod \varphi(n)$.

Пара $(e, n)-$ является открытым ключом, в то время как пара $(d, n)-$закрытым.

1.1.2 Шифрование

Пусть Боб хочет послать Алисе сообщение m. В таком случае, Бобу необходимо взять пару открытых ключей (e, n), открытый текст m и зашифровать с использованием функции $E(m)$:

$$c = E(m) = m^e \bmod n$$

1.1.3 Расшифрование

Алиса на своей стороне принимает зашифрованное сообщение c и с помощью закрытого ключа (d, n) расшифровывает с использованием функции $D(c)$:

$$m = D(c) = c^d \bmod n$$

1.1.4 Алгоритм с сеансовым ключом

Однако чаще используется алгоритм шифрования, где Боб сначала шифрует сеансовый ключ и потом, с его помощью, шифрует сообщения. В таком случае алгоритм шифрования имеет такую последовательность: Боб создает случайные сеансовый ключ m и зашифровывает его с помощью функции $E(m)$, которая была описана ранее. Затем шифруется само сообщение с помощью функции $E(m)$, но в качестве ключа будет выступать сеансовый.

Алиса принимает зашифрованный сеансовый ключ c и с помощью закрытого ключа расшифровывает его, используя функцию $D(c)$, описанную ранее. После этого она расшифровывает сообщение, но в качестве ключа опять же использует сеансовый ключ.

1.1.5 Подпись

Для подписи Бобу необходимо выполнить:

$$s = m^d \bmod n$$

Алиса получает и расшифровывает сообщение от Боба, а также получает его подпись s. Используя свой секретный ключ, она проверяет:

$$m' = s^e \bmod n$$

Затем, она проверяет, что $m \equiv m'$. Наглядно данную схему можно описать так:

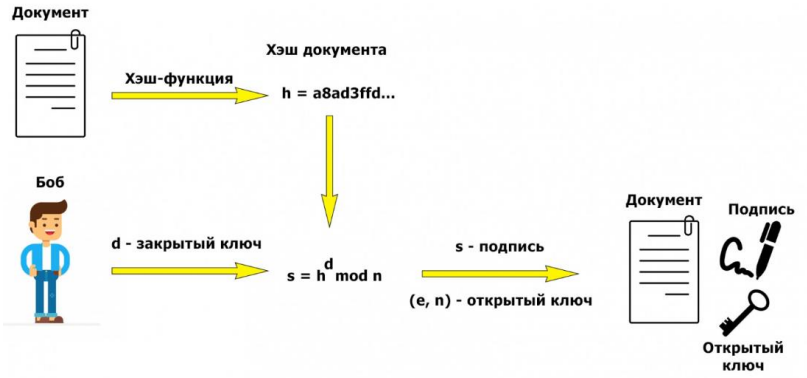

Рисунок 1 - Формирование подписи

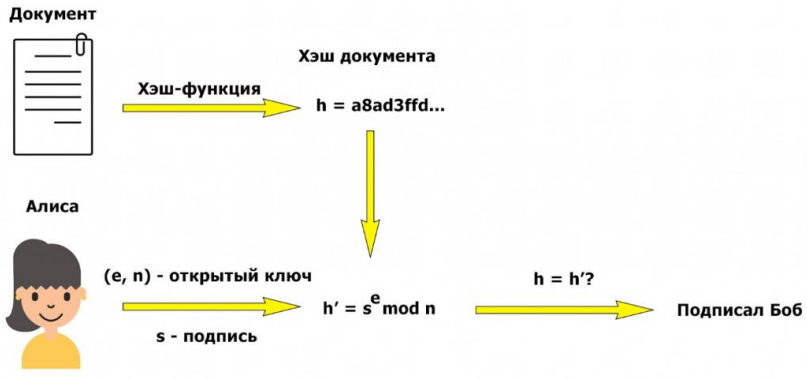

Рисунок 2 - Проверка подписи

2 Пример работы программы

Сгенерированные ключи:

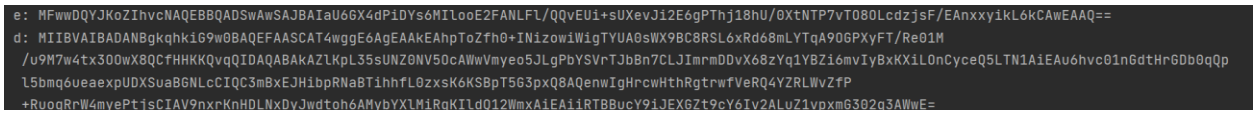

Рисунок 3 - Генерация ключей

Пусть Боб хочет передать следующее сообщение:

Рисунок 4 – Сообщение

Результат хэширования сообщения с использованием MD5:

Рисунок 5 - Хэш-код сообщения

Шифрование сообщения:

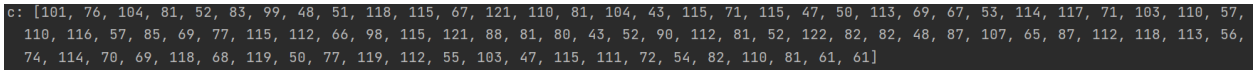

Рисунок 6 – Шифротекст

Расшифрование шифротекста:

Рисунок 7 - Расшифрованный текст

Проверка подписи:

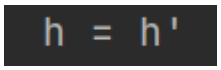

Рисунок 8 - Проверка подписи

Как видно, хэш-коды совпали, а значит, личность Боба подтверждена.

3 Выводы

Таким образом, в ходе выполнения работы была реализована система RSA, а также реализована проверка подписи с использованием хэш-функции MD5.

Список используемых источников:

1) Ященко В.В. Введение в криптографию. Под общей ред. В. В. Ященко — СПб.: Питер, 2001. - 288 с.