

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 умножению к e d , т.е. $de \equiv 1 \pmod{\varphi(n)}$.

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 $E(m)$:

$$c = E(m) = m^e \pmod{n}$$

1.1.3 Расшифрование

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

$$m = D(c) = c^d \pmod{n}$$

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 случайный 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:

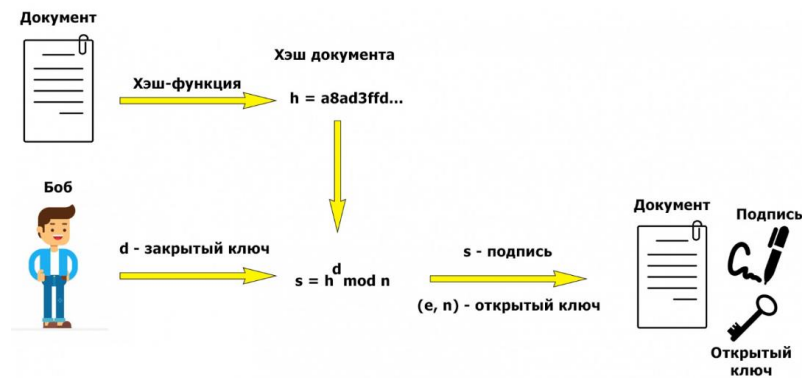


Рисунок 1 - Формирование подписи

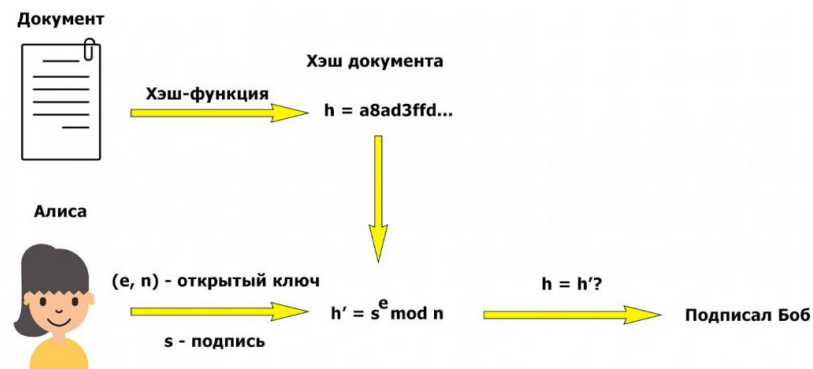


Рисунок 2 - Проверка подписи

2 Пример работы программы

Сгенерированные ключи:

```
e: MFwwDQYJKoZIhvcNAQEBBQADSwAwSAJBAlaU6GX4dPiDys6MIloE2FANLFL/QQvEU1+sUXevJi2E6gPThj18hU/0XtNTP7vT080LcdzjsF/EAnxxyikL6kCAwEAAQ==
d: MIIBVAIBADANBgkqhkiG9w0BAQEFAASCAT4wggE6AgEAAKEAhpToZf0+INizow1WigTYUA0sWX9BC8RSL6xRd68mLYTqA906PXyFT/Re01M
/u9M7w4tx300wX8QCfHHKKQvQIDAQABAKAZLkpL35sUNZ0NV50cAWwVmyeo5JLgPbYSVrTJb8n7CLJImrmDDvX68zYq1YBZi6mvIyBxKXiL0nCyceQ5LTN1AiEAu6hvc01nGdtHr6D0b0Qqp
l5bm6ueaexpUDXSuaBGNLcCIQC3mBxEJHibpRNaBTihhfL0zxsK6KSBpT563pxQ8AQenwIghrcwHthRgtrwFVeRQ4YZRLWvZfP
+RuogRcW4mvePt1sCIAV9nxcKnHDLNxbvJwdtoh6AMvbYXlMhRqKtld012Wmx4iEA4iRT8BucY9iJEX6Zt9cY6IV2ALuZ1vpxm6302q3AwwE=
```

Рисунок 3 - Генерация ключей

Пусть Боб хочет передать следующее сообщение:

$m = \text{Hello, Alice}$

Рисунок 4 – Сообщение

Результат хэширования сообщения с использованием MD5:

$h: [80, -5, 41, 15, -97, -30, -91, 109, 96, 81, 52, -66, -36, 97, -86, 5]$

Рисунок 5 - Хэш-код сообщения

Шифрование сообщения:

$c: [101, 76, 104, 81, 52, 83, 99, 48, 51, 118, 115, 67, 121, 110, 81, 104, 43, 115, 71, 115, 47, 50, 113, 69, 67, 53, 114, 117, 71, 103, 110, 57, 110, 116, 57, 85, 69, 77, 115, 112, 66, 98, 115, 121, 88, 81, 80, 43, 52, 90, 112, 81, 52, 122, 82, 82, 48, 87, 107, 65, 87, 112, 118, 113, 56, 74, 114, 70, 69, 118, 68, 119, 50, 77, 119, 112, 55, 103, 47, 115, 111, 72, 54, 82, 110, 81, 61, 61]$

Рисунок 6 – Шифротекст

Расшифрование шифротекста:

$h': [80, -5, 41, 15, -97, -30, -91, 109, 96, 81, 52, -66, -36, 97, -86, 5]$

Рисунок 7 - Расшифрованный текст

Проверка подписи:

$h = h'$

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