Лабораторная работа № 5. Электронная цифровая подпись (4 часа)

5.1 ЦЕЛЬ РАБОТЫ

Изучение механизма выработки и проверки значения электронной цифровой подписи с использованием криптографических алгоритмов с открытым ключем.

5.2 ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

Цифровая подпись для электронных документов играет ту же роль, что и подпись, поставленная от руки в документах на бумаге: это данные, присоединяемые к передаваемому сообщению, подтверждающие, что владелец подписи составил или заверил это сообщение. Получатель сообщения с помощью цифровой подписи может проверить, что автором сообщения является именно владелец подписи, и что в процессе передачи не была нарушена целостность полученных данных.

При разработке механизма цифровой подписи возникают следующие задачи:

1. Формирование подписи таким образом, чтобы её невозможно было подделать.
2. Обеспечение возможности проверки того, что подпись действительно принадлежит указанному субъекту.
3. Предотвращение отказа субъекта от своей подписи.

5.2.1 Классическая схема создания цифровой подписи

До того, как будет происходить формирование цифровой подписи, отправитель должен сгенерировать два ключа: открытый K_o и секретный K_c. При этом закрытый ключ должен быть известен только тому, кто подписывает сообщения, а открытый — любому желающему проверить подлинность сообщения.

При создании цифровой подписи по классической схеме отправитель должен выполнить следующие действия.

1. Вычислить хеш-образ m исходного сообщения M при помощи хеш-функции h.
2. Вычислить цифровую подпись S по хеш-образу сообщения с использованием секретного ключа K_c создания подписи.
3. Сформировать новое сообщение (M, S), состоящее из исходного сообщения и добавленной к нему цифровой подписи.

Получив подписанное сообщение (M', S), получатель должен выполнить следующие действия для проверки подлинности подписи и целостности полученного сообщения:

1. Вычислить хеш-образ m' сообщения M' при помощи хеш-функции h.

2. С использованием открытого ключа проверки подписи (K_o) извлечь хеш-образ m сообщения из цифровой подписи S.

3. Сравнить вычисленное значение m’ с извлеченным из цифровой подписи значением хеш-образа m. Если хеш-образы совпадают, то подпись признается подлинной.

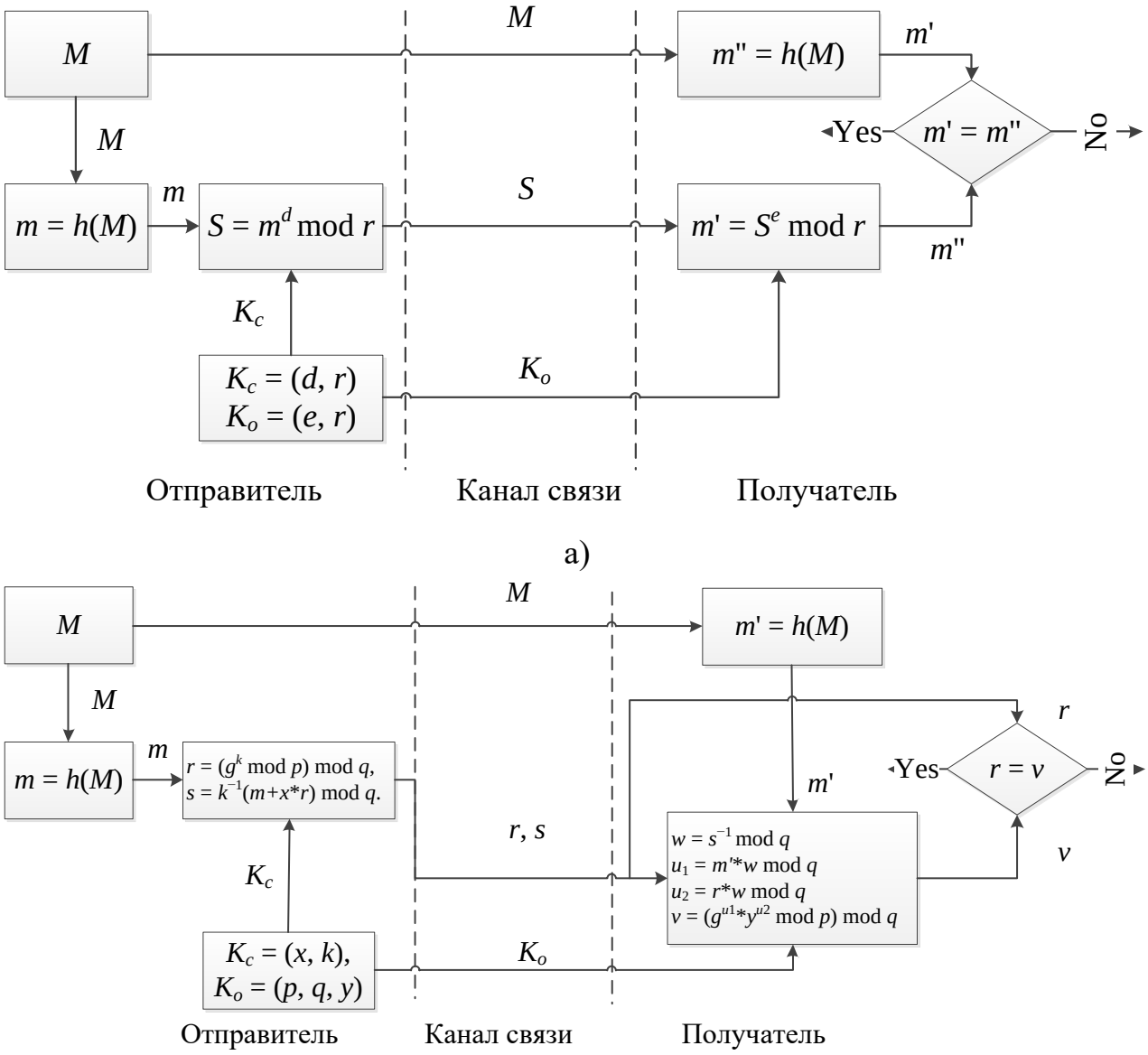

Рисунок 1. Обобщенная схема цифровой подписи
а) RSA, б) DSA

Фальсификация сообщения при его передаче по каналу связи возможна при получении злоумышленником секретного ключа K_c или за счет проведения успешной атаки против хеш-функции. Используемые в реальных приложениях хеш-функции обладают характеристиками, делающими атаку против цифровой подписи практически не осуществимой. Например, хеш-функция SHA-1, принятая в США в качестве стандарта в 1995 году,

формирующая 160-битовый хеш-образ при обработке сообщения блоками по 512 бит. С 2015 года происходит переход от использования хеш-функций SHA-1 и SHA-2 на использование SHA-3, которая может формировать хеш-образ длиной 224, 256, 386 или 512 бит.

5.2.2 Алгоритм цифровой подписи RSA

Первой и наиболее известной во всем мире конкретной системой электронной цифровой подписи стала система RSA, математическая схема которой была разработана в 1977 г. в Массачусетском технологическом институте США.

Для формирования подписи по алгоритму RSA сначала необходимо вычислить пару ключей: секретный ключ и открытый ключ, как это делается для криптосистемы RSA:

1. Выбираются два случайных простых числа p и q таких, что $p \approx q$.
2. Вычисляется их произведение $r = p*q$.
3. Вычисляется функция Эйлера для r $\varphi(r) = (p–1)*(q–1)$.
4. Выбирается открытая экспонента e такая, что $1<e<\varphi(r)$ и $(e, \varphi(r)) = 1$.
5. Вычисляется секретная экспонента d, удовлетворяющая условию $(e*d) \bmod \varphi(r) = 1$.

Пару значений $K_o = (e, r)$, которая является открытым ключом, автор передает партнерам по переписке для проверки его цифровых подписей. Значение $K_c = (d, r)$ сохраняется автором как секретный ключ подписи.

Если отправителю необходимо подписать сообщение M перед его отправкой, он сжимает сообщение M с помощью хеш-функции h в целое число m: $m = = h(M)$. Затем вычисляет цифровую подпись S под электронным документом M на основе хеш-образа m и секретного значения d:

$$S = m^d \bmod r. \qquad (5.1)$$

Пара (M, S) передается получателю как электронный документ M, подписанный цифровой подписью S, причем подпись S сформирована обладателем секретного ключа (d, r).

После приема пары (M', S) получатель вычисляет хеш-образ сообщения M' двумя различными способами. Прежде всего, он восстанавливает хеш-образ m, применяя криптографическое преобразование подписи S с использованием открытого ключа (e, r):

$$m = S^e \bmod r. \qquad (5.2)$$

Кроме того, он находит результат хеширования m' принятого сообщения M' с помощью такой же хеш-функции h: $m' = h(M)$.

Если вычисленные значения совпадают, т. е. $h(M) = S^e \bmod r$, то получатель признает пару (M', S) подлинной.

Например, подпишем сообщение "БГУИР". Сначала получим его хеш-образ. Он равен $h(M)$=93 (см. раздел 4.2.1 для лабораторной работы № 4 Функция хеширования). Далее сгенерируем открытый и закрытый ключи:

1. Выберем $p = 17$, $q = 19$.
2. Вычислим $r = 17*19 = 323$.
3. Вычислим $\varphi(r) = (p–1)*(q–1) = 16*18 = 288$.
4. Выберем открытую экспоненту $e = 43$, взаимно простую с $\varphi(r) = 288$.
5. На основе e и $\varphi(r)$ вычислим закрытую экспоненту $d = 67$, используя расширенный алгоритм Евклида.

Тогда открытый ключ будет равен (43, 323), а закрытый – (67, 323). Далее подписываем сообщение:

$$S = m^d \bmod r = 93^{67} \bmod 323 = 206.$$

После чего отправляем сообщение, состоящее из самого текста и подписи: {БГУИР, 206}. Пусть при передачи сообщение было изменено, и получатель получил {БРУИР, 206}. Для проверки подписи сначала он вычисляет хеш-образ полученного сообщения "БРУИР":

$$H_1 = (H_0+M_1)^2 \bmod n = (100+2)^2 \bmod 323 = 10404 \bmod 323 = 68,$$
$$H_2 = (H_1+M_2)^2 \bmod n = (68+18)^2 \bmod 323 = 7396 \bmod 323 = 290,$$
$$H_3 = (H_2+M_3)^2 \bmod n = (290+21)^2 \bmod 323 = 96721 \bmod 323 = 144,$$
$$H_4 = (H_3+M_4)^2 \bmod n = (144+10)^2 \bmod 323 = 23716 \bmod 323 = 137,$$
$$H_5 = (H_4+M_5)^2 \bmod n = (137+18)^2 \bmod 323 = 24025 \bmod 323 = 123.$$

С другой стороны из цифровой подписи с помощью известного ему открытого ключа (43, 323) получатель вычисляет хеш-образ, переданный отправителем:

$$S = m^e \bmod r = 206^{43} \bmod 323 = 93.$$

Так как два вычисленных значений 123 и 93 не равны, то подпись признается недействительной.

5.2.3 Алгоритм цифровой подписи DSA

Алгоритм DSA (Digital Signature Algorithm – алгоритм цифровой подписи) был предложен Национальным институтом стандартов и технологий в августе 1991. Данный алгоритм вместе с криптографической хеш-функцией SHA-2 является частью DSS (Digital Signature Standard – стандарт цифровой подписи) − криптографического стандарта электронной цифровой подписи, используемой в США. DSA основан на трудности вычисления дискретных логарифмов и базируется на схеме, первоначально представленной Эль-Гамалем и Шнорром.

Алгоритма цифровой подписи DSA состоит в следующем. Сначала необходимо получить секретный и открытый ключи, для этого выполнить следующие действия:

1. Выбрать большое простое число q.

2. Выбрать простое число p такое, что q является делителем $(p-1)$.

3. Подобрать число g такое, что для него верно $g = h^{(p-1)/q} \bmod p$, где h – некоторое произвольное число из интервала $(1, p-1)$, и при этом $g>1$. В большинстве случаев значение $h = 2$ удовлетворяет этому требованию.

4. Закрытый ключ отправителя x выбирается случайно из интервала $(0, q)$.

5. Открытый ключ вычисляется из закрытого ключа по формуле:

$$y = g^x \bmod p. \tag{5.3}$$

Вычислить y по известному x довольно просто (используя алгоритм быстрого возведения в степень). Однако, имея открытый ключ y, вычислительно невозможно определить x, который является дискретным логарифмом y по основанию g.

Открытой информацией являются значения p, q и y, закрытой – x. При этом значения p и q могут быть общими для группы пользователей, а значение y и x – для каждого свое.

Подпись сообщения выполняется по следующему алгоритму:

1. Получаем хеш-образ исходного сообщения $h(M)$. При использовании формулы 4.2 (см. раздел 4.2.1 для лабораторной работы № 4 Функция хеширования) вычисления необходимо выполнять по модулю числа q.

2. Выбирается случайное число k из $(0, q)$, уникальное для каждого подписи.

3. Вычисляется значение r и s по формулам:

$$\begin{aligned} r &= (g^k \bmod p) \bmod q, \\ s &= k^{-1}(h(M)+x*r) \bmod q. \end{aligned} \tag{5.4}$$

4. Если одно из полученных значений r или s будет равно 0, то необходимо повторить вычисления для другого значения k. Иначе, подписью будет пара значений (r, s).

Таким образом, сообщение с подписью будет иметь вид $\{M, r, s\}$.

Для того чтобы проверить подлинность подписи, сначала из полученного сообщения $\{M', r, s\}$вычисляется хеш-образ $h(M')$, после чего находят значение v, используя формулы (5.5). Подпись признается подлинной, если $v = r$.

$$
\begin{aligned}
&w = s^{-1} \bmod q,\\
&u_1 = h(M)*w \bmod q,\\
&u_2 = r*w \bmod q,\\
&v = (g^{u1}*y^{u2} \bmod p) \bmod q.
\end{aligned}
\qquad (5.5)
$$

Приведем пример данного алгоритма подписи. Возьмем приведенное выше сообщение "БГУИР", хеш-образ которого равен 93. Далее сгенерируем открытый и закрытый ключи для создания подписи. Для этого выберем случайные простые числа q и p, пусть они будут равны соответственно 107 и 643. Как видно $p-1$ (642) делится на q (107) без остатка. Тогда число $g = 2^{(643-1)/107} \bmod 643 = 64$. Далее выберем случайное число $x = 45$, которое будет секретным ключом и храниться в секрете, и вычислим для него открытый ключ по формуле (5.3): $y = g^x \bmod p = 64^{45} \bmod 643 = 181$. Значение y является открытой информацией.

Вычислим цифровую подпись для сообщения. Для этого возьмем его хеш-образ $h(M) = 93$, сгенерируем случайное число $k = 31$, и вычислим r, s по формулам (5.4):

$$
\begin{aligned}
&r = (g^k \bmod p) \bmod q = (64^{31} \bmod 643) \bmod 107 = 36,\\
&s = k^{-1}(h(m) + x*r) \bmod q = \frac{1}{31}(93 + 45*36) \bmod 107 = 31^{\varphi(q)-1}*1713 \bmod 107 = 38.
\end{aligned}
$$

Так как оба полученных значения r и s не равны 0, то подпись будет равна паре значений (36, 38). И отправляемое сообщение будет иметь вид: {БГУИР, 36, 38}.

Для проверки подлинности подписи получатель выполняет следующие действия. Сначала он вычисляет хеш-образ сообщения "БГУИР", которое равно 93. Далее вычисляет значение v по формулам (5.5):

$w = s^{-1} \bmod q = 38^{105} \bmod 107 = 31$,
$u_1 = h(M)*w \bmod q = 93*31 \bmod 107 = 101$,
$u_2 = r*w \bmod q = 36*31 \bmod 107 = 46$,
$v = (g^{u1}*y^{u2} \bmod p) \bmod q = (64^{101}*181^{46} \bmod 643) \bmod 107 = (((64^{101} \bmod 643)*(181^{46} \bmod 643)) \bmod 643) \bmod 107 = 36$.

Так как $r = v$ (36 = 36), то подпись является подлинной.

5.3 ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

1. Изучить теоретический материал по лабораторной работе.
2. Реализовать программное средство вычисления и проверки электронной цифровой подписи:
    a. вариант №1 -по схеме RSA
    b. вариант №2 -по схеме DSA

3. Реализованное программное средство должно вырабатывать ЭЦП и проверять значение ЭЦП для произвольного текстового файла (*.txt), размером более 1kb. Для вычисления хеш-образа сообщения необходимо использовать хеш-функцию, реализованную в рамках лабораторной работы № 4.
4. Длины ключей во всех алгоритмах должны быть достаточно большими (512 бит - 1024 бита). Можно использовать для работы классы работы с большими числами (BigInteger C#, Java).

Пояснение по выбору варианта:

Ваш_Номер_Варианта=(Номер_Вашей_зачетной_книжки) mod 2+1

Например №=123456
(123456) mod 2 +1 = 0+1=1