Практическая работа №11

ФОРМИРОВАНИЕ ЭЛЕКТРОННО-ЦИФРОВОЙ ПОДПИСИ ДОКУМЕНТА

Цель работы. Изучить принципы работы алгоритмов генерации и проверки электронной цифровой подписи.

Краткие сведения из теории

Электронная цифровая подпись

При обмене электронными документами по сети связи возникает проблема аутентификации автора документа и самого документа, т. е. установления подлинности автора и отсутствия изменений в полученном документе. В обычной (бумажной) информатике эти проблемы решаются за счет того, что информация в документе и рукописная подпись автора жестко связаны с физическим носителем (бумагой). В электронных документах на машинных носителях такой связи нет.

Электронная цифровая подпись (ЭЦП) используется для аутентификации текстов, передаваемых по телекоммуникационным каналам. Функционально она аналогична обычной рукописной подписи и обладает ее основными достоинствами:

– удостоверяет, что подписанный текст исходит от лица, поставившего подпись;

– не дает самому этому лицу возможности отказаться от обязательств, связанных с подписанным текстом;

– гарантирует целостность подписанного текста.

Цифровая подпись представляет собой относительно небольшое количество дополнительной цифровой информации, передаваемой вместе с подписываемым текстом. Система ЭЦП включает две процедуры: 1) постановки подписи; 2) проверки подписи. В процедуре постановки подписи используется секретный ключ отправителя сообщения, в процедуре проверки подписи – открытый ключ отправителя.

При формировании ЭЦП отправитель прежде всего вычисляет хэш-функцию $h(M)$ подписываемого текста M. Вычисленное значение хэш-функции $h(M)$ представляет собой один короткий блок информации m, характеризующий весь текст M в целом. Хэш-функция предназначена для сжатия подписываемого документа M до нескольких десятков или сотен бит. Хэш-функция $h(M)$ принимает в качестве аргумента сообщение (документ) M произвольной длины и возвращает хэш-значение $h(M) = m$ фикси-

рованной длины. Обычно хэшированная информация является сжатым двоичным представлением основного сообщения произвольной длины. Значение хэш-функции $h(M)$ сложным образом зависит от документа M и не позволяет восстановить сам документ M.

Затем число m шифруется секретным ключом отправителя. Получаемая при этом пара чисел представляет собой ЭЦП для данного текста M.

При проверке ЭЦП получатель сообщения снова вычисляет хэш-функцию $m = h(M)$ принятого по каналу сообщения M, после чего при помощи открытого ключа отправителя проверяет, соответствует ли полученная подпись вычисленному значению m хэш-функции.

Функция $h(M)$ – является хэш-функцией, если она удовлетворяет следующим условиям:

– исходный текст может быть произвольной длины;

– само значение $h(M)$ имеет фиксированную длину;

– значение функции $h(M)$ легко вычисляется для любого аргумента;

– восстановить аргумент по значению с вычислительной точки зрения – практически невозможно;

– функция $h(M)$ – однозначна.

Из определения следует, что для любой хэш-функции есть тексты-близнецы – имеющие одинаковое значение хэш-функции, так как мощность множества аргументов неограниченно больше мощности множества значений. Такой факт получил название «эффект дня рождения».

Наиболее известные из хэш-функций – MD2, MD4, MD5 и SHA.

Три алгоритма серии MD разработаны Ривестом в 1989-м, 90-м и 91-м годах соответственно. Все они преобразуют текст произвольной длины в 128-битную сигнатуру.

Алгоритм MD2 предполагает:

– дополнение текста до длины, кратной 128 битам;

– вычисление 16-битной контрольной суммы (старшие разряды отбрасываются);

– добавление контрольной суммы к тексту;

– повторное вычисление контрольной суммы.

Алгоритм MD4 предусматривает:

– дополнение текста до длины, равной 448 бит по модулю 512;

– добавление длины текста в 64-битном представлении;

– использование процедуры Damgard-Merkle с 512-битными блоками (в отличие от хэш-функции этот класс преобразований предполагает вычисление для аргументов фиксированной длины также фиксированных по длине значений), причем каждый блок участвует в трех разных циклах.

В алгоритме MD4 довольно быстро были найдены «дыры», поэтому он был заменен алгоритмом MD5, в котором каждый блок участвует не в трех, а в четырех различных циклах.

Алгоритм SHA (Secure Hash Algorithm) разработан NIST (National Institute of Standard and Technology) и повторяет идеи серии MD. В SHA используются тексты более 2^{64} бит, которые закрываются сигнатурой длиной 160 бит.

Принципиальным моментом в системе ЭЦП является невозможность подделки ЭЦП пользователя без знания его секретного ключа подписывания. В качестве подписываемого документа может быть использован любой файл. Подписанный файл создается из неподписанного путем добавления в него одной или более электронных подписей. Каждая подпись содержит следующую информацию:

– дату подписи;

– срок окончания действия ключа данной подписи;

– информацию о лице, подписавшем файл (Ф.И.О., должность, краткое наименование фирмы);

– идентификатор подписавшего (имя открытого ключа);

– собственно цифровую подпись.

Технология применения системы ЭЦП предполагает наличие сети абонентов, посылающих друг другу подписанные электронные документы. Для каждого абонента генерируется пара ключей: секретный и открытый. Секретный ключ хранится абонентом в тайне и используется им для формирования ЭЦП. Открытый ключ известен всем другим пользователям и предназначен для проверки ЭЦП получателем подписанного электронного документа. Иначе говоря, открытый ключ является необходимым инструментом, позволяющим проверить подлинность электронного документа и автора подписи. Открытый ключ не позволяет вычислить секретный ключ.

Для генерации пары ключей (секретного и открытого) в алгоритмах ЭЦП, как и в асимметричных системах шифрования, используются разные математические схемы, основанные на применении однонаправленных функций. Эти схемы разделяются на две группы. В основе такого разделения лежат известные сложные вычислительные задачи:

– факторизации (разложения на множители) больших целых чисел;

– дискретного логарифмирования.

Алгоритм электронной цифровой подписи RSA. Первой и наиболее известной во всем мире конкретной системой ЭЦП стала система RSA. Согласно этому алгоритму, сначала необходимо вычислить пару ключей (секретный ключ и открытый ключ). Для этого отправитель (автор) электронных документов вычисляет два больших простых числа – P и Q, затем находит их произведение $N = PQ$ и значение функции $\varphi(N) = (P - 1)(Q - 1)$. Далее отправитель вычисляет число E из условий: $E \le \varphi(N)$, НОД$[E, \varphi(N)] = 1$ и число D из условий: $D < N$, $ED \equiv 1 \ [\mathrm{mod}\ \varphi(N)]$. Пара чисел (E, N) является открытым ключом. Эту пару чисел автор передает партнерам по переписке для проверки его цифровых подписей. Число D сохраняется автором как

секретный ключ для подписывания.

Допустим, что отправитель хочет подписать сообщение M перед его отправкой. Сначала сообщение M (блок информации, файл, таблица) сжимают с помощью хэш-функции $h(M)$ в целое число m: $m = h(M)$. Затем вычисляют цифровую подпись S под электронным документом M, используя хэш-значение m и секретный ключ D: $S = m^D \pmod N$.

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

После приема пары (M, S) получатель вычисляет хэш-значение сообщения M двумя разными способами. Прежде всего он восстанавливает хэш-значение m', применяя криптографическое преобразование подписи S с использованием открытого ключа E: $m' = S^E \pmod N$. Кроме того, он находит результат хэширования принятого сообщения M с помощью такой же хэш-функции $h(M)$: $m = h(M)$.

Если соблюдается равенство вычисленных значений, т. е. $S^E \pmod N = h(M)$, то получатель признает пару (M, S) подлинной. Доказано, что только обладатель секретного ключа D может сформировать цифровую подпись S по документу M, а определить секретное число D по открытому числу E не легче, чем разложить модуль N на множители. Кроме того, можно строго математически доказать, что результат проверки цифровой подписи S будет положительным только в том случае, если при вычислении S был использован секретный ключ D, соответствующий открытому ключу E. Поэтому открытый ключ E иногда называют "идентификатором" подписавшего.

К недостаткам алгоритма электронной цифровой подписи RSA можно отнести следующее:

1 При вычислении модуля N, ключей E и D для системы цифровой подписи RSA необходимо проверять большое количество дополнительных условий, что сделать практически трудно. Невыполнение любого из этих условий делает возможным фальсификацию цифровой подписи со стороны того, кто обнаружит такое невыполнение. При подписании важных документов нельзя допускать такую возможность даже теоретически.

2 Для обеспечения криптостойкости цифровой подписи RSA по отношению к попыткам фальсификации на уровне, например, национального стандарта США шифрования информации (алгоритм DES), т. е. 10^{18}, необходимо использовать при вычислениях N, D и E целые числа не менее 2^{512} (или около 10^{154}) каждое, что требует больших вычислительных затрат, превышающих на 20–30 % вычислительные затраты других алгоритмов цифровой подписи при сохранении того же уровня криптостойкости.

3 Цифровая подпись RSA уязвима к так называемой мультипликативной атаке. Иначе говоря, алгоритм цифровой подписи RSA позволяет противнику без знания секретного ключа D сформировать подписи под теми доку-

ментами, у которых результат хэширования можно вычислить как произведение результатов хэширования уже подписанных документов.

Алгоритм электронной цифровой подписи Эль Гамаля (EGSA). Название EGSA происходит от слов El Gamal Signature Algorithm (алгоритм цифровой подписи Эль Гамаля). Идея EGSA основана на том, что для обоснования практической невозможности фальсификации цифровой подписи может быть использована более сложная вычислительная задача, чем разложение на множители большого целого числа, – задача дискретного логарифмирования. Кроме того, Эль Гамалю удалось избежать явной слабости алгоритма цифровой подписи RSA, связанной с возможностью подделки цифровой подписи под некоторыми сообщениями без определения секретного ключа.

Для того чтобы генерировать пару ключей (открытый ключ – секретный ключ), сначала выбирают некоторое большое простое целое число P и большое целое число G, причем $G < P$. Отправитель и получатель подписанного документа используют при вычислениях одинаковые большие целые числа P ($\sim 10^{308}$ или $\sim 2^{1024}$) и G ($\sim 10^{154}$ или $\sim 2^{512}$), которые не являются секретными.

Отправитель выбирает случайное целое число X, $1 < X \le (P - 1)$ и вычисляет $Y = G^X \bmod P$. Число Y является открытым ключом, используемым для проверки подписи отправителя. Число Y открыто передается всем потенциальным получателям документов. Число X является секретным ключом отправителя для подписывания документов и должно храниться в секрете.

Для того чтобы подписать сообщение M, сначала отправитель хэширует его с помощью хэш-функции $h()$ в целое число m: $m = h(M)$, $1 < m < (P - 1)$ и генерирует случайное целое число K, $1 < K < (P - 1)$ такое, что K и $(P - 1)$ являются взаимно простыми. Затем отправитель вычисляет целое число a: $a = G^K \bmod P$ и, применяя расширенный алгоритм Евклида, вычисляет с помощью секретного ключа X целое число b из уравнения $m = Xa + Kb \pmod{(P - 1)}$.

Пара чисел (a, b) образует цифровую подпись S, проставляемую под документом M. Тройка чисел (M, a, b) передается получателю, в то время как пара чисел (X, K) держится в секрете.

После приема подписанного сообщения (M, a, b) получатель должен проверить, соответствует ли подпись $S = (a, b)$ сообщению M. Для этого получатель сначала вычисляет по принятому сообщению M число $m = h(M)$, т. е. хэширует принятое сообщение.

Затем получатель вычисляет значение $A = Y^a a^b \pmod P$ и признает сообщение M подлинным, если $A = G^m \pmod P$. Иначе говоря, получатель проверяет справедливость соотношения $Y^a a^b \pmod P = G^m \pmod P$.

Можно строго математически доказать, что последнее равенство будет выполняться тогда, и только тогда, когда подпись $S = (a, b)$ под документом

M получена с помощью именно того секретного ключа X, из которого был получен открытый ключ Y. Таким образом, можно надежно удостовериться, что отправителем сообщения M был обладатель именно данного секретного ключа X, не раскрывая при этом сам ключ, и что отправитель подписал именно этот конкретный документ M.

Выполнение каждой подписи по методу Эль Гамаля требует нового значения K, причем это значение должно выбираться случайным образом. Если нарушитель раскроет когда-либо значение K, повторно используемое отправителем, то он сможет раскрыть секретный ключ X отправителя.

Для примера выберем числа P =11, G = 2 и секретный ключ X = 8. Вычислим значение открытого ключа: $Y = G^X \bmod P = Y = 2^8 \bmod 11 = 3$. Предположим, что исходное сообщение M характеризуется хэш-значением m = 5. Для того чтобы вычислить цифровую подпись для сообщения M, имеющего хэш-значение m = 5, сначала выберем случайное целое число K = = 9. Убедимся, что числа K и (P – 1) являются взаимно простыми. Действительно, НОД (9, 10) = 1.

Далее вычисляем элементы a и b подписи: $a = G^K \bmod P = 2^9 \bmod 11 = 6$, элемент b определяем, используя расширенный алгоритм Евклида:

$$m = Xa + Kb \ [\bmod (P - 1)]. \qquad (1)$$

При m = 5, a = 6, X = 8, K = 9, P = 11 получаем 5 = (6×8 + 9×b)(mod 10) или $9 \times b \equiv -43 \pmod{10}$. Решив данное линейное уравнение, получим: b = 3. Цифровая подпись представляет собой пару: a = 6, b = 3.

Далее отправитель передает подписанное сообщение. Приняв подписанное сообщение и открытый ключ Y = 3, получатель вычисляет хэш-значение для сообщения M: m = 5, а затем вычисляет два числа:

1) $Y^a a^b \pmod P = 3^6 \times 6^3 \pmod{11} = 10 \pmod{11}$;

2) $G^m \pmod P = 2^5 \pmod{11} = 10 \pmod{11}$.

Так как эти два целых числа равны, принятое получателем сообщение признается подлинным.

Схема Эль Гамаля является характерным примером подхода, который допускает пересылку сообщения M в открытой форме вместе с присоединенным аутентификатором (a, b). В таких случаях процедура установления подлинности принятого сообщения состоит в проверке соответствия аутентификатора сообщению.

Схема цифровой подписи Эль Гамаля имеет ряд преимуществ по сравнению со схемой цифровой подписи RSA:

1 При заданном уровне стойкости алгоритма цифровой подписи целые числа, участвующие в вычислениях, имеют запись на 25 % короче, что уменьшает сложность вычислений почти в два раза и позволяет заметно сократить объем используемой памяти.

2 При выборе модуля P достаточно проверить, что это число является

простым и что у числа (P – 1) имеется большой простой множитель (т. е. всего два достаточно просто проверяемых условия).

3 Процедура формирования подписи по схеме Эль Гамаля не позволяет вычислять цифровые подписи под новыми сообщениями без знания секретного ключа (как в RSA).

Однако алгоритм цифровой подписи Эль Гамаля имеет и некоторые недостатки по сравнению со схемой подписи RSA. В частности, длина цифровой подписи получается в 1,5 раза больше, что, в свою очередь, увеличивает время ее вычисления.

Российский стандарт электронной цифровой подписи. В российском алгоритме цифровой подписи, определяемом стандартом ГОСТ Р 34.10-94, используются следующие параметры:

p – большое простое число длиной от 509 до 512 либо от 1020 до 1024 бит;

q – простой сомножитель числа (p – 1), имеющий длину от 254 до 256 бит;

a – любое число, меньшее (p – 1), причем такое, что $a^q \bmod p = 1$;

x – некоторое число, меньшее q;

$y = a^x \bmod p$.

Кроме того, этот алгоритм использует однонаправленную хэш-функцию $h(M)$. Стандарт ГОСТ Р 34.11-94 определяет хэш-функцию, основанную на использовании стандартного симметричного алгоритма ГОСТ 28147-89.

Первые три параметра p, q и a являются открытыми и могут быть общими для всех пользователей сети. Число x является секретным ключом. Число y является открытым ключом.

Чтобы подписать некоторое сообщение m, а затем проверить подпись, выполняются следующие шаги:

1 Пользователь A генерирует случайное число k, причем $k < q$.

2 Пользователь A вычисляет значения $r = (a^k \bmod p) \bmod q$, $s = \{xr + k\,[h(m)]\} \bmod q$.

Если $h(m) \bmod q = 0$, то значение $h(m) \bmod q$ принимают равным единице. Если $r = 0$, то выбирают другое значение k и начинают снова.

Цифровая подпись представляет собой два числа: $r \bmod 2^{256}$ и $s \bmod 2^{256}$.

Пользователь A отправляет эти числа пользователю B.

3 Пользователь B проверяет полученную подпись, вычисляя последовательно величины $v = h(m)^{q-2} \bmod q$, $z_1 = (sv) \bmod q$, $z_2 = ((q - r)v) \bmod q$, $u = (a^{z_1} \cdot y^{z_2} \bmod p) \bmod q$.

Если $u = r$, то подпись считается верной.

Следует также отметить, что в данном стандарте ЭЦП параметр q имеет длину 256 бит. Западных криптографов вполне устраивает q длиной примерно 160 бит. Различие в значениях параметра q является отражением

стремления разработчиков российского стандарта к получению более безопасной подписи.

Порядок выполнения работы

1 Изучить краткие сведения из теории.

2 По последней цифре шифра из таблицы 1 необходимо выбрать сообщения, которые будут подвергаться шифрованию.

Таблица 11 – **Исходные сообщения**

Параметр	Последняя цифра шифра				
	1	2	3	4	5
m	5	7	9	11	13
Параметр	Последняя цифра шифра				
	6	7	8	9	0
m	15	17	19	21	23

3 По предпоследней цифре шифра из таблицы 2 необходимо выбрать ключи шифрования.

4 По первой цифре шифра из таблицы 3 необходимо выбрать параметры шифрования.

Таблица 2 – **Ключи шифрования**

Параметр	Предпоследняя цифра шифра									
	1	2	3	4	5	6	7	8	9	0
X_2	2	11	9	6	13	11	12	11	6	9

Таблица 3 – **Параметры алгоритмов шифрования**

Параметр	Первая цифра шифра									
	1	2	3	4	5	6	7	8	9	0
P_3	6	12	16	15	13	11	9	7	5	4
Q_2	13	7	5	7	8	9	7	11	13	11
P_4	23	22	21	20	19	18	17	16	15	14
G_2	5	7	9	11	12	10	8	6	4	2
K_7	9	11	13	7	7	5	11	2	3	4

5 Создать ЭЦП для хешированного сообщения m с использованием алгоритма:

– RSA с параметрами P_3, Q_2.

– Эль Гамаля с параметрами P_4, G_2, X_2, K_7.

Содержание отчета

1 Цель работы.

2 Исходные данные.

3 Результаты расчетов.

4 Вывод по работе.

Контрольные вопросы

1 Дайте понятие криптографии.
2 Что относится к криптошрафическим методам защиты информации?
3 В чем принцип асимметричных методов шифрования?
4 Почему асимметричные методы шифрования называются криптосистемами с закрытым ключом?
5 Что такое электронный документооборот?
6 Приемущества электронного документа.
7 ЭЦП с использованием алгоритма RSA.
8 ЭЦП с использованием алгоритма Эль Гамаля.