ТЕХНОЛОГИЯ ЦИФРОВЫХ ПОДПИСЕЙ

Электронная цифровая подпись (ЭЦП), используется для аутентификации текстов, передаваемых по телекоммуникационным каналам. При таком обмене существенно снижаются затраты на обработку и хранение документов, убыстряется их поиск. Но возникает проблема аутентификации автора электронного документа, т.е. установления подлинности автора и отсутствия изменений в полученном электронном документе.

Целью аутентификации электронных документов является их защита от возможных видов злоумышленных действий, таких как :

- отказ (ренегатство) - пользователь *A* заявляет, что он не посылал сообщение пользователю *B*, хотя на самом деле посылал;
- модификация (переделка) - пользователь *B* изменяет сообщение и утверждает, что данные измененного сообщения посылал ему пользователь *A*;
- подделка (подмена) - пользователь *B* формирует сообщение и утверждает, что данные сформированного сообщения посылает ему пользователь *A;*
- активный перехват - пользователь *C* перехватывает сообщение между пользователями

A и *B c* целью скрытой модификации;

- маскировка (имитация, маскарад) - пользователь C посылает сообщение пользователю *B* от имени *A;*

повтор - пользователь *C* повторяет ранее переданное сообщение, которое пользователь *A* посылал ранее *B*.

Функционально цифровая подпись 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.

ЭЦП представляет собой относительно небольшое количество дополнительной цифровой информации, передаваемой вместе с подписываемым текстом.

ЭЦП реализуется при помощи асимметричных алгоритмов шифрования и хэш-функций.

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.

Таким образом, ЭЦП - данные, присоединяемые к передаваемому сообщению и подтверждающие, что владелец подписи заверил данное сообщение.

При этом получатель сообщения может проверить, что автором сообщения является именно владелец подписи, и что данные не были изменены в процессе передачи. В общем случае под ЭЦП понимается числовое значение, вычисляемое по сообщению с использованием секретного ключа подписывающего. Проверка ЭЦП осуществляется общеизвестной процедурой на основании открытого ключа.

Система ЭЦП включает две основные процедуры:

- формирования цифровой подписи;
- проверка цифровой подписи.

Процедура формирования цифровой подписи.

На подготовительном этапе этой процедуры абонент *A* - отправитель сообщения - генерирует пару ключей: секретный ключ k_A и открытый ключ K_A. Открытый ключ рассылается остальным абонентам сети для проверки подписи.

Для формирования цифровой подписи отправитель *A* прежде всего вычисляет значение хэш-функции *h(M)* подписываемого текста *M*. Хэш-функция служит для сжатия исходного подписываемого текста *M* в дайджест *m* - относительно короткое число, состоящее из фиксированного небольшого числа битов и характеризующее весь текст *M* в целом.

Под хэш-функцией понимается математическое или алгоритмическое преобразование заданного блока данных. Она обладает следующими свойствами: имеет бесконечную область определения, конечную область значений, необратима, изменение входного потока информации на один бит меняет около половины всех бит выходного потока.

К дайджесту m добавляется информация о том, кто подписывает документ, штамп времени и прочее. Далее абонент А шифрует получившуюся строку своим секретным ключом k_A. Получившийся зашифрованный набор бит и представляет собой подпись для данного текста M.

Сообщение *M* вместе с цифровой подписью отправляется в адрес получателя.

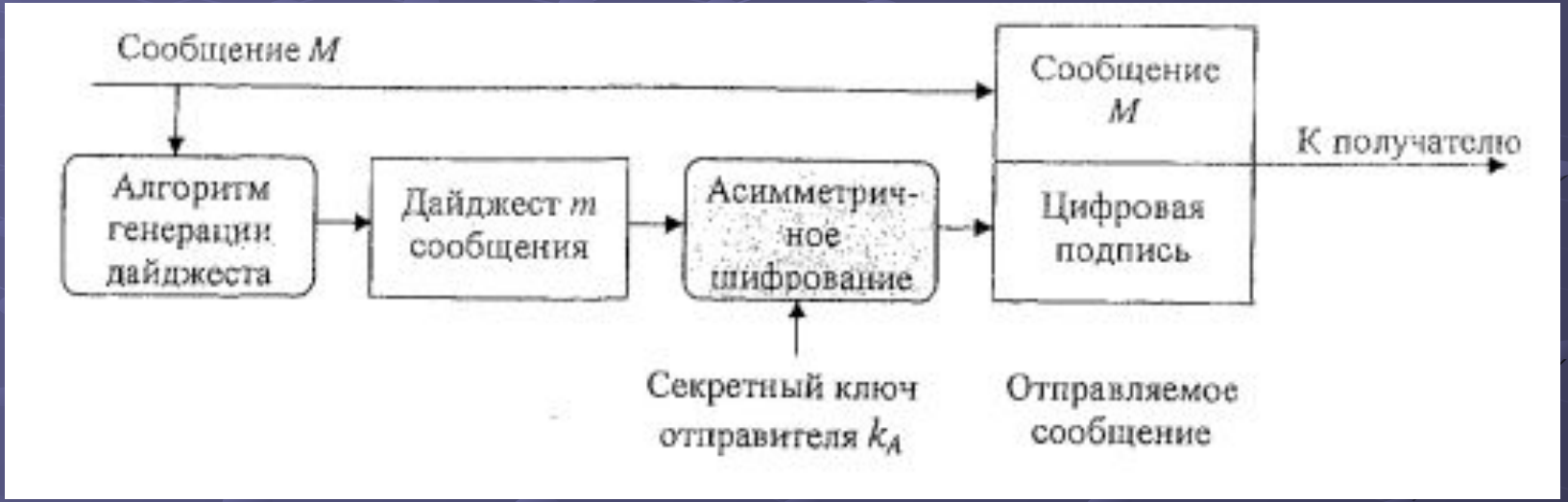

Рисунок 1 – Схема формирования электронной цифровой подписи

Процедура проверки цифровой подписи.

При проверке ЭЦП абонент *B* - получатель сообщения *M* - расшифровывает принятый дайджест *m* открытым ключом K_A отправителя *A*. Кроме того, получатель сам вычисляет с помощью хэш-функции *h(M)* дайджест *m'* принятого сообщения *M* и сравнивает его с расшифрованным. Если *m* и *m'* совпадают, то цифровая подпись является подлинной. В противном случае либо подпись подделана, либо изменено содержание сообщения.

Рисунок 2 - Схема проверки электронной цифровой подписи

Принципиальным моментом в системе ЭЦП является невозможность подделки ЭЦП пользователя без знания его секретного ключа. Поэтому необходимо защитить секретный ключ от НСД.

Важно отметить, что с точки зрения конечного пользователя процесс формирования и проверки цифровой подписи отличается от процесса криптографического закрытия передаваемых данных следующими особенностями.

При формировании цифровой подписи используется закрытый ключ отправителя, тогда как при зашифровке используется открытый ключ получателя. При проверке цифровой подписи используется открытый ключ отправителя, а при расшифровке - закрытый ключ получателя.

Проверить сформированную подпись может любое лицо, так как ключ проверки является открытым. При положительном результате проверки подписи делается заключение о подлинности и целостности полученного сообщения, т.е. о том, что это сообщение действительно отправлено тем или иным отправителем, и не было модифицировано при передаче по сети.

Схема подписи на основе алгоритма RSA.

Для создания подписи сообщения M отправитель:

1. Вычисляет сжатый образ (дайджест сообщения, хэш-образ) $R = H(M)$ с помощью хэш-функции H.
2. Зашифровывает полученный сжатый образ на своем секретном ключе и получает подпись $S=R^d \bmod n$, где $\{d,n\}$-закрытый ключ отправителя.

Для проверки подписи получатель:

1. Расшифровывает подпись S на открытом ключе отправителя, то есть вычисляет $R' = S^e \bmod n$, где $\{e, n\}$ - открытый ключ отправителя.

2. Вычисляет сжатый образ $R = H(M)$ с полученного сообщения с помощью той же самой хэш-функции *H,* которую использовал отправитель.
3. Сравнивает полученные значения *R* и *R',* если они совпадают, то подпись верна.

Таким образом, *ЭЦП* - набор электронных цифровых символов, подтверждающий достоверность электронного документа, его принадлежность и неизменность содержания.

Закрытый ключ ЭЦП - последовательность электронных цифровых символов, известная владельцу подписи и предназначенная для создания ЭЦП.

Открытый ключ ЭЦП - последовательность электронных цифровых символов, доступная любому лицу и предназначенная для подтверждения подлинности ЭЦП в электронном документе.

Цифровая подпись на основе алгоритма Эль-Гамаля

Алгоритм Эль-Гамаля также можно использовать для формирования цифровой подписи. Группа пользователей выбирает общие параметры P и A. Затем каждый абонент группы выбирает свое секретное число Xi, 1<Xi<P-1, и вычисляет соответствующее ему открытое число

$$Y_i \ : \ Y_i = A^{X_i} \ mod \ P$$

Таким образом, каждый пользователь получает пару (закрытый ключ; открытый ключ) = (Xi,Yi).

Пусть пользователь 1 хочет подписать свое сообщение цифровой подписью и передать его пользователю 2. В этом случае алгоритм действий следующий.

Первый пользователь выбирает случайное секретное число k, взаимно простое с P-1, и вычисляет число $a = A^k \bmod P$

Затем с помощью расширенного алгоритма Евклида необходимо найти значение b в следующем уравнении:

m = (X1 * a +k * b) mod (P-1)

Пара чисел (a, b) будет цифровой подписью сообщения m.

Сообщение m вместе с подписью (a, b) отправляется пользователю 2.

Пользователь 2 получает сообщение m и с использованием открытого ключа первого абонента Y1 вычисляет два числа по следующим формулам:

$$c_1 = Y_1^a \times a^b \bmod P$$
$$c_2 = A^m \bmod P$$

Если c1=c2, то цифровая подпись первого пользователя верная. Для подписывания каждого нового сообщения должно каждый раз выбираться новое значение k.

Подписи, созданные с использованием алгоритма Эль-Гамаля, называются рандомизированными, так как для одного и того же сообщения с использованием одного и того же закрытого ключа каждый раз будут создаваться разные подписи (a,b), поскольку каждый раз будет использоваться новое значение k. Подписи, созданные с применением алгоритма RSA, называются детерминированными, так как для одного и того же сообщения с использованием одного и того же закрытого ключа каждый раз будет создаваться одна и та же подпись.

Пример вычисления и проверки цифровой подписи

Пусть абоненты, обменивающиеся через Интернет зашифрованными сообщениями, имеют следующие общие параметры: P = 11, A = 7.

Один из пользователей этой системы связи хочет подписать свое сообщение m=5 цифровой подписью, сформированной по алгоритму Эль-Гамаля. Вначале он должен выбрать себе закрытый ключ, например, X1=3 и сформировать открытый ключ $Y1 = 7^3 \bmod 11 = 2$. Открытый ключ может быть передан всем заинтересованным абонентам или помещен в базу данных открытых ключей системы связи.

Затем пользователь выбирает случайное секретное число k, взаимно простое с P-1. Пусть k=9 (9 не имеет общих делителей с 10). Далее необходимо вычислить число

$$a = A^k \ mod \ P = 7^9 \ mod \ 11 = 8$$

После этого с помощью расширенного алгоритма Евклида находится значение b в уравнении:

$$m = (X_1 \times a + k \times b) \ mod \ (P - 1),$$
$$5 = (3 \times 8 + 9 \times b) \ mod \ 10$$

Решением последнего уравнения будет значение b=9.

Таким образом, пара чисел (8, 9) будет цифровой подписью сообщения m=5.

Если любой другой пользователь сети желает проверить цифровую подпись в сообщении, он должен получить из базы данных открытый ключ первого пользователя (он равен 2), вычислить два числа c1 и c2 и сравнить их.

$$c_1 - Y_1^a \times a^b \; mod \, P = 2^8 \times 8^9 \; mod \, 11 = 10,$$
$$c_2 = A^m \; mod \, 11 = 10$$

Так как c1=c2, то цифровая подпись первого пользователя в сообщении m=5 верная.

Стандарты на алгоритмы цифровой подписи

Стандарт цифровой подписи DSS

Во многих странах сегодня существуют стандарты на электронную (цифровую) подпись. Стандарт цифровой подписи DSS (Digital Signature Standard – DSS) был принят в США в 1991 году и пересмотрен в 1994 году. В основе стандарта лежит алгоритм, называемый DSA (Digital Signature Algorithm) и являющийся вариацией подписи Эль-Гамаля. В алгоритме используется однонаправленная хэш-функция H(m). В качестве хэш-алгоритма стандарт DSS предусматривает использование алгоритма SHA-1.

Рассмотрим сам алгоритм генерации ЭЦП. Вначале для группы абонентов выбираются три общих (несекретных) параметра p, q и a:

- параметр p должен быть простым числом длиной от 512 до 1024 бит.
- q – простое число длиной 160 бит; между p и q должно выполняться соотношение $p = bq + 1$ для некоторого целого b.
- число a, удовлетворяющее неравенству $1 < a < p-1$ и являющееся корнем уравнения $a^q \bmod p = 1$.

Зная эти числа, каждый абонент системы случайно выбирает число x, удовлетворяющее неравенству $0<x<q$, и вычисляет $y = a^x \bmod p$.

Число x будет секретным ключом пользователя, а число y — открытым ключом. Вычислить y по известному x довольно просто. Однако, имея открытый ключ y, вычислительно невозможно определить x, который является дискретным логарифмом y по основанию q .

Пусть имеется сообщение m, которое один из пользователей желает подписать. Для генерации подписи пользователь должен выполнить следующие действия:

1. Вычислить значение хэш-функции $h = H(m)$ для сообщения m. Значение хэш-функции должно лежать в пределах $0 < h < q$.

2. Затем сгенерировать случайное число k, $0 < k < q$.
3. Вычислить $r = (a^k \bmod p) \bmod q$.
4. Определить $s = [k^{-1}(H(m) + x*r)] \bmod q$

В результате пользователь получит для сообщения m подпись, состоящую из пары чисел (r,s). Сообщение вместе с подписью может быть послано любому другому абоненту системы. Проверить подпись можно следующим образом:

1. Вычислить значение хэш-функции $h = H(m)$ для сообщения m.
2. Проверить выполнение неравенств $0 < r < q$, $0 < s < q$.

3. Вычислить $w = s^{-1} \bmod q$;

4. $u1 = [H(m) * w] \bmod q$

5. $u2 = r * w \bmod q$

6. $v = [(a^{u1} * y^{u2}) \bmod p] \bmod q$

Проверить выполнение равенства $v = r$. Если $v = r$, то подпись считается подлинной, иначе подпись считается недействительной.

В силу сложности вычисления дискретных логарифмов злоумышленник не может восстановить k из r или x из s, а следовательно, не может подделать подпись. По той же самой причине автор сообщения не сможет отказаться от своей подписи, так как никто кроме него не знает закрытого ключа x.

Стандарт цифровой подписи ГОСТ Р34.10-94

В России принят стандарт ГОСТ Р34.10-94 "Информационная технология. Криптографическая защита информации. Процедуры выработки и проверки электронной цифровой подписи на базе асимметричного криптографического алгоритма". В этом стандарте используется алгоритм, аналогичный алгоритму, реализованному в стандарте DSS.

Вначале, так же как и по стандарту DSS, для группы абонентов выбираются три общих (несекретных) параметра p, q и a.

По-разному вычисляется компонента s подписи. В ГОСТ Р34.10-94 компонента s вычисляется по формуле

$s = (k(H(m) + x(r) \bmod q,$

а в DSS компонента s вычисляется по формуле

$s = [k^{-1} ((H(m) + x(r))] \bmod q.$

Последнее отличие приводит к соответствующим отличиям в формулах для проверки подписи.

В результате процедура проверки подписи по ГОСТ Р34.10-94 заключается в следующем. Получив [m, (r, s)], получатель вычисляет

$w = H(m)^{-1} \bmod q,$

$u1 = w * s \bmod q$,

$u2 = (q-r) * w \bmod q$,

$v = [(a^{u1} * y^{u2}) \bmod p] \bmod q$.

Затем проверяется равенство вычисленного значения v и полученного в составе ЭЦП параметра r. Подпись считается корректной, если v = r.

Пример создания и проверки подписи по стандарту ГОСТ Р34.10-94

Пусть p = 23, q = 11, a =6 (проверяем: $6^{11} \bmod 23 = 1$)

Создание подписи.

Предположим, пользователь А выбрал в качестве закрытого ключа число x=8. После этого он вычисляет открытый ключ по формуле $y=a^x \bmod p$. То есть $y = 6^8 \bmod 23 = 18$.

Для создания подписи пользователь А выбирает случайное число k = 5.

Пусть результат вычисления хэш-функции для сообщения H(m) = 9.

Подпись сообщения состоит из двух чисел (r, s):

$r = (a^k \bmod p) \bmod q = (6^5 \bmod 23) \bmod 11 = 2,$

$s = (k^* H(m) + x * r) \bmod q = (5*9 + 8*2) \bmod 11 = 6,$

Таким образом, подпись сообщения состоит из пары чисел (2, 6).

Проверка подписи.

Получив сообщение вместе с подписью (2,6), получатель вычисляет

$w = H(m)^{-1} \bmod q = 9^{-1} \bmod 11 = 5,$

$u1 = w * s \bmod q = 5 * 6 \bmod 11 = 8,$

$u2 = (q-r)*w \bmod q = (11-2)*5 \bmod 11 = 1,$

$v = [(a^{u1}*y^{u2}) \bmod p] \bmod q = [(6^{8}*18^{1}) \bmod 23] \bmod 11 = 2$

Так как v = r, то подпись считается верной.

Функция хэширования

В качестве исходного значения для вычисления ЭЦП берется не сам электронный документ, а его хэш-значение или дайджест.

Хэш-значение *h(M)* - это дайджест сообщения *M*, т. е. сжатое двоичное представление основного сообщения *M* произвольной длины.

Функция хэширования (хэш-функция) представляет собой преобразование, на вход которого подается сообщение переменной длины *M*, а выходом является строка фиксированной длины *H= h(M)*.

Функция хэширования позволяет сжать подписываемый документ *M* до 128 и более бит (в частности до 128 или 256 бит), тогда как *M* может быть размером в мегабайт или более.

Следует отметить, что значение хэш-функции *h(M)* зависит сложным образом от документа *M* и не позволяет восстановить сам документ *M*.

Функция хэширования обладает следующими свойствами:

- хэш-функция может быть применена к аргументу любого размера;
- выходное значение хэш-функции имеет фиксированный размер;
- хэш-функцию достаточно просто вычислить для любого аргумента; скорость вычисления хэш-функции должна быть такой, чтобы скорость выработки и проверки ЭЦП при использовании хэш-функции была значительно больше, чем при использовании самого сообщения;

- хэш-функция чувствительна к всевозможным изменения в тексте M, таким как вставки, выбросы, перестановки и т.п.;
- хэш-функция однонаправлена, т.е. обладает свойством необратимости, иными словами, задача подбора документа *M'* который обладал бы требуемым значением хэш-функции, должна быть вычислительно неразрешима;
- вероятность того, что значения хэш-функций двух различных документов (вне зависимости от их длин) совпадут, ничтожно мала, т.е. для любого фиксированного *x* с вычислительной точки зрения невозможно найти x' для *x*, такое, что $h(x') = h(x)$.

Таким образом, функция хэширования может использоваться для обнаружения изменений сообщения. В этом качестве хэш-функция используется для контроля целостности сообщения при формировании и проверке ЭЦП.

Хэш-функции широко используются также для аутентификации пользователей.