Министерство образования и науки РФ

Федеральное государственное автономное образовательное учреждение высшего профессионального образования <<Казанский (Приволжский) федеральный университет>>

ИНСТИТУТ МАТЕМАТИКИ И МЕХАНИКИ

КАФЕДРА АЛГЕБРЫ И МАТЕМАТИЧЕСКОЙ ЛОГИКИ

Направление: 01.03.01 - математика

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА
(бакалаврская работа)

О некоторых затемненных цифровых подписях и их адаптациях для платежных систем.

Работа завершена:
“___”_____ 2015 г. _____ (Е.А.Хохлова)

Работа допущена к защите:
Научный руководитель
Доктор физ.-мат. наук,
профессор кафедры алгебры
и математической логики КФУ

“___”_____ 2015 г. _____ (С.Н.Тронин)

Заведующий кафедрой алгебры
и математической логики КФУ
Доктор физ.-мат. наук, профессор
“___”_____ 2053 г. _____ (М.М. Арсланов)

Казань-2015

ОГЛАВЛЕНИЕ

Введение

Важным разделом современной криптографии является финансовая криптография. Одно из основных направлений финансовой криптографии - конструирование различных платежных систем, позволяющих осуществлять платежи дистанционно и в безналичной (электронной) форме. Математическим аппаратом, который позволяет конструировать такие системы, являются затемненные (слепые, blind) цифровые подписи. Первые такие подписи появились в 1980-х годах в работах голландского математика D. Chaum (Дэвида Чаума). В простейшем варианте затемненную подпись осуществляет банк, когда к нему обращается клиент с просьбой предоставить ему так называемую электронную банкноту. Номер (идентификатор) такой банкноты клиент генерирует сам, и банк каким-то образом должен сгенерировать подпись этого идентификатора. Однако, должно выполняться требование неотслеживаемости (как и для обычных денег): по электронной банкноте, в которой будет присутствовать её номер, банку должно быть невозможно установить кому он её выдал (подписал). Оказалось, что это возможно осуществлять с помощью затемненных подписей. Но для использования таких подписеё на практике, они должны удовлетворять ряду дополнительных требований. В частности, необходимо, чтобы клиент мог снимать со своего счета любую сумму, и информация об этой сумме присутствовала в подписанной электронной банкноте. То есть, необходимы затемненные подписи с дополнительным параметром, который клиент не в состоянии изменить, но банк может проконтролировать его при проверке подписи.

В большинстве из имеющихся в литературе затемненных подписей такая задача даже не ставится. В нашей работе предпринята попытка решить эту задачу для двух известных цифровых подписей: затемненной подписи Шнорра[7] и подписи из работы [5]. В общих случаях задачу удалось решить. Отметим, что по фининсовой криптографии имеется несколько достаточно свежих книг: Handbook [8], книга "Криптографические протоколы и их применение в финансовой комерческой деятельности"[7] и "Electronic Payment Systems for E-Commerce Second Edition"[9].

Опишем вкратце содержание работы. В главе 1 напоминаются основные понятия криптографии с открытым ключом. В главе 2 описываются некоторые известные затемненные подписи, прежде всего подписи основанные на обычных подписях RSA, Эль-Гамаля и Шнорра. В 3 главе основные идеи теории платежных систем. В частности, платежные системы Чаума и Бранса. В заключительной главе 4 решается основная задача работы, сформулированная выше. В-первом параграфе главы модифицируется затемненная подпись Шнорра. В модифицированном виде эта подпись теперь включает параметр, имеющий смысл снимаемой со счета суммы. В-третьем параграфе аналогичная задача решается для подписи из работы [5]. В обоих случаях дается обоснование модифицированных подписей.Во-втором параграфе мы рассмотриваем слепую подпись из работы[5]. При этом используется также работа [6]. Результат нашей работы докладывался на студенческой научной конференции КФУ 22 апреля 2015 года.

Глава 1.Общая информация о криптографии с открытым ключом

§ 1. Односторонние функции и хеш-функции.

Самая первая работа по асимметричным шифрам «Новые направления в современной криптографии» Уитфилда Диффи и Мартина Хеллмана, опубликованна в 1976 году. Они предложили метод получения секретных ключей для симметричного шифрования, используя открытый канал находясь под влиянием работы Ральфа Меркле (Ralph Merkle) о распространении открытого ключа,. В 2002 году Хеллман предложил называть данный алгоритм «Диффи - Хеллмана - Меркле», признавая вклад Меркле в изобретение криптографии с открытым ключом.

Работа Диффи-Хеллмана создала большой теоретический задел для открытой криптографии,но все-таки первой реальной криптосистемой с открытым ключом считают алгоритм RSA. Этот алгоритм назван по имени авторов - Рон Ривест (Ronald Linn Rivest), Ади Шамир (Adi Shamir) и Леонард Адлеман (Leonard Adleman).

В шифрования с открытым ключом для шифрования данных используется открытый ключ и его знают все пользователи, а для дешифрования секретный (закрытый) ключ и его знает только получатель закрытого сообщения.

В шифровании с открытым ключом тот, кто зашифровывает сообщение, не обязательно может его расшифровывать. Без ключа расщифрования не одно сообшение расшифровать нельзя. Алгоритмы шифрования с открытым ключом используют односторонние функции.

Определение 1.1.1. Одностороннии функции - это функции которые при заданном значении аргумента x относительно просто вычислить значение функции (x), однако, если известно значение функции $y = f(x)$, то нет простого пути для вычисления значения аргумента x.

Но не всякая необратимая функция подходит для использования в реальных криптосистемах.

Существует два важных и очевидных требования которые гарантируют надежную защиту информации в криптосистемах с открытым ключом:

1. Исходный текст должно быть невозможно восстановить на основе открытого ключа.

2. Определить закрытый ключ на основе открытого быть невозможным ни одному современному компьютеру.

Рассмотрим, как используя криптографию с открытым ключом, два человека могут отправить друг другу сообщения. Возьмем двых персонажей Алису и Боба. Вот по такому алгоритму Алиса может послать сообщение Бобу:

(1) Алиса и Боб согласовывают криптосистему с открытыми ключами.

(2) Боб посылает Алисе свой открытый ключ.

(3) Алиса шифрует свое сообщение и посылает его Бобу.

(4) Боб расшифровывает сообщение Алисы с помощью своего закрытого ключа.

Определение 1.1.2. Криптографическая хэш-функция h — это функция, определенная на битовых строках произвольной длины со значениями в строках битов фиксированной длины.

Ее значение часто называют хэш-кодом или хэш-значением. Также своего рода хэш-функции используются в информатике. Но важное отличие обычных хэш-функций от криптографических состоит в том, что в криптографии они должны быть односторонними.

Другими словами, должно быть невозможно в вычислительном отношении по элементу y из множества значений хэш-функции подобрать такой из области определения, при котором $h(x) = y$.

Рассмотрим очень простой пример хэш-функции. Хэш-код создается функцией H:

$$h = H(M)$$

Где M сообщение произвольной длины и h хэш-код фиксированной длины.

Хэш-функция H должна обладать следующими свойствами:

(1) Хэш-функция H должна применяться к данным любой длины.

(2) Хэш-функция H создает выход фиксированной длины.

(3) $H(M)$ не сложно вычисляется для любого значения M.

(4) $\forall$ данного значения хэш-кода h невозможно найти M такое, что $H(M) = h$.

(5) $\forall$ данного невозможно найти $y \neq x$, что $H(y) = H(x)$.

(6) Невозможно найти произвольную пару (x, y) такую, что $H(y) = H(x)$.

Использование хеш-функций даёт следующие преимущества:

1. Вычислительная сложность. Формировать хэш документа и подписывать его получается намного быстрее, чем подписывать сам документ.

2. Совместимость. Хеш-функцию можно использовать для преобразования любого входного текста в подходящий формат.

3. Целостность. Обычно большой электронный документ обычно нужно разделять на более малые блоки для применения ЭП. Использование хэш-функции не требуется разбиение на блоки.

§ 2. Общая идея электронной цифровой подписи.

Определение 1.2.1. Электронная цифровая подпись(ЭЦП) - номер электронного документа, который нужен для защиты данного электронного документа от подделки. Он получается в результате криптографического преобразования информации с использованием закрытого ключа ЭЦП и позволяющий опозновать владельца сертификата ключа подписи, а также установить отсутствие изменения информации в электронном документе, а также нет возможности отказатказаться подписавшемуся.

Электронная подпись предназначена для определения лица, который подписал электронный документ, и является аналогом собственноручной подписи в случаях, предусмотренных законом. Использование электронной подписи позволяет осуществить:

1. Контроль за целостностью передаваемого документа.
2. Защиту от подделки документа.
3. Невозможность отказа от авторства.
4. Доказательное подтверждение авторства документа.

Поскольку подписываемые документы — непостоянного объёма, в схемах ЭП чаще всего подпись ставится не на сам документ, а на его хэш.

Протокол 2.1. Асимметричная схема. Асимметричные схемы ЭЦП являются к криптосистемами с открытым ключом. В схемах цифровой подписи подписывание производится с применением закрытого ключа, а проверка — с применением открытого.

Определение 1.2.2. Пусть $X \subseteq A^*$ — множество исходных сообщений, $S = V_n$ — множество значений цифровой подписи, K — множество ключей. *Схемой цифровой подписи* будем называть набор (X, S, K, Sig, Ver), в котором:

1) $Sig : X \times K \to S$— алгоритм формирования цифровой подписи;

2) $Ver : X \times S \times K \to \{0, 1\}$— алгоритм проверки цифровой подписи. При этом должно выполняться условие: если $M \in X$, $s \in S$, $k \to K$, то $Ver_k(M, s) = 1$ в том и только в том случае, если $Sig_k(M) = s$.При этгом проверяющий не знает секретный ключ K. Ему известна только функция Ver_k в целом.

Схема цифровой подписи охватывает три процесса:

1) Генерация ключевой пары. Используя алгоритма генерации ключа выбирается закрытый ключ, вычисляется соответствующий ему открытый ключ.

2) Формирование подписи. Для заданного электронного документа с помощью закрытого ключа вычисляется подпись.

3) Проверка подписи. Для данных документа и подписи с помощью открытого ключа определяется действительность подписи.

Для того, чтобы использование цифровой подписи имело смысл, необходимо выполнение двух условий:

1) Проверка подписи производится открытым ключом,который должен соответствовать именно тому закрытому ключу, который использовался при подписании.

2) Без обладания закрытым ключом должно быть вычислительно сложно создать цифровую подпись.

§ 3. Криптосистема RSA.

Определение 1.3.1. RSA – криптографическая система открытого ключа, обеспечивающая такие механизмы защиты как шифрование и цифровая подпись (аутентификация – установление подлинности).

Криптосистема RSA разработана в 1977 году и названа в честь ее разработчиков Ronald Rivest, Adi Shamir и Leonard Adleman. Криптосистема RSA основана на теореме Эйлера, согласно которой для любых взаимно простых целых чисел и n, где $< n$, выполняется соотношение

$$M^{\varphi(n)} \equiv 1 \pmod{n}$$

В криптосистеме RSA в качестве числа используется сообщение, которое необходимо подписать или зашифровать. Должно выполняться условие взаимной простоты чисел и n .

Выбираются два различных больших простых числа p, q. Положим $N = pq$. Выбирается E, $1 < E < \varphi(N)$ с условием НОД$(E, \varphi(N)) = 1$. Пусть число d таково, что $Ed = 1 + \varphi(N)t$.

Шифруется сообщение — битовая строка. Ее можно рассматривать как двоичную запись натурального числа m. Условие: $0 \leq m \leq N - 1$.

Секретный ключ: p, q, d. Отправитель сообщения m (Алиса) знает N и E. Получатель зашифрованного сообщения C (Боб) знает все, что знает Алиса, и еще секретный ключ (фактически ему достаточно знать d). Если бы существовали эффективные методы разложения на сомножители , то, разложив N на сомножители p и q, можно было бы получить частный (private) ключ d. Таким образом надежность криптосистемы RSA основана

на трудноразрешимой – практически неразрешимой – задаче разложения N на сомножители так как в настоящее время эффективного способа поиска сомножителей не существует.

Протокол 3.1.

1) **Шифрование** $C = m^E \pmod N$.

2) **Дешифрование** $m = C^d \pmod N$.

RSA.Цифровая подпись

Предположим, Алиса хочет послать Бобу сообщение M , при этом Боб должен быть уверен, что сообщение не было взломано и что автором сообщения действительно является Алиса. Алиса создает цифровую подпись S возводя M в степень d и умножая на модуль n: Секретный ключ знает подписывающий (Алиса).

Протокол 3.2. 1) **Подписание** $S = m^d \pmod N$, где d и N - частный ключ Алисы.

2) Алиса отправляет Бобу сообщение m и подпись S.

3) **Проверка подписи** $m \equiv S^E \pmod N$, где E и N - открытый (public) ключ Алисы.

Использование хэш-функциии

Рассмотрим схему электронной цифровой подписи по алгоритму RSA.

$$S = h(M)^d \pmod N; \qquad h(M) \equiv S^E \pmod N.$$

Для вычисления цифровой подписи используется криптографическое преобразование по алгоритму RSA.

Субъект, который желает пересылать подписанные им документы, должен сформировать два ключа открытый и закрытый.

Пару значений (E, N)-открытый ключ подписи, отправитель - Алиса передаёт получателю - Бобу.Эти значения будут использоваться для проверки подлинности сообщений.А также открытый ключ помогает подтвердить принадлежность сообщения отправителю. Значение d вместе с модулем N - секретный ключ, который будет использоваться отправителем для постановки подписей под своими сообщениями.

1. Алиса сжимает сообщение M в целое число $m = h(M)$.

2. Алиса вычисляет значение цифровой подписи S для сообщения M

$$S = m^d \pmod{N}$$

   И отправляет подписанное сообщение $(, S)$ Бобу.

3. Боб проверяет подлинность сообщения и принадлежит ли оно Алисе.

4. Боб сжимает полученное сообщение M' при помощи криптографической хеш-функции h, которая идентична той, которая была использована Алисой, в целое число m'.

5. Боб выполняет расшифрование открытым ключом E отправителя m оригинального сообщения, преобразуя значение подписи S по алгоритму RSA:

$$m = S^E \pmod{N}$$

6. Боб сравнивает полученные значения m' и m. Если данные значения совпадают, т. е.

$$S^E \pmod{N} = h(M)$$

то он признает полученное сообщение подлинным и принадлежащим Алисе.

§ 4. Криптосистема Эль-Гамаля.

Криптосистема с открытым ключом,основанная на трудности вычисления дискретных логарифмов в конечном поле. Криптосистема включает в себя алгоритм шифрования и алгоритм цифровой подписи. Схема Эль-Гамаля лежит в основе стандартов электронной цифровой подписи в США и России

Схема была предложена Тахером Эль-Гамалем в 1984 году.

Исходные данные: p — большое простое число, g — порождающий элемент циклической группы $\mathbb{F}_p^*$, x — секретный ключ, который знает только Боб, $1 < x < p-1$.

Вычисляется открытый ключ: $y = g^x \pmod p$. Он известен и Алисе, и Бобу.

Протокол 4.1. Шифрование

1) Генерируется случайное число k, $0 < k < p-1$. Это — так называемый сеансовый ключ, или эфемерный ключ, или ключ подписания.

2) Первая часть зашифрованного сообщения: $C_1 = g^k \pmod p$.

3) Вторая часть зашифрованного сообщения: $C_2 = my^k \pmod p$.

4) Алиса посылает Бобу пару чисел (или элементов $\mathbb{F}_p$): C_1, C_2.

Заметим, что при каждом шифровании применяется свой кратковременный ключ. Поэтому, шифруя одно сообщение дважды, мы получаем разные шифротексты.

Дешифрование

$$m = C_2/C_1^x \pmod{p}.$$

Чтобы расшифровать пару данных $C = (C_1, C_2)$, производят следующие преобразования:

$$\frac{C_2}{C_1^x} = \frac{mH^k}{G^x k} = \frac{mG^x k}{G^x k} = m$$

Эль-Гамаль. Цифровая подпись

Исходные данные те же, но секретным ключом x владеет подписывающий (Алиса). Имеется также открытая хэш-функция h, $0 \leq h(m) \leq p-2$.

Протокол 4.2. Подписание

Алиса подписывает сообщение m следующим образом.

1) Генерируется случайное число k, $0 < k < p-1$. В отличие от случая шифрования, здесь требуется, чтобы НОД$(k, p-1) = 1$.

2) Вычисляется первая часть подписи: $s_1 = g^k \pmod{p}$.

3) Вторая часть подписи s_2 ищется как решение уравнения:

$$h(m) \equiv xs_1 + ks_2 \pmod{(p-1)}.$$

4) Подпись сообщения m — пара чисел s_1, s_2. таким образом, Алиса отправляет Бобу три числа: m, s_1, s_2.

Проверка подписи

Боб не знает ни x, ни k, но знает p, g, y, h.

Проверка подписи состоит в проверке сравнения:

$$g^{h(m)} \equiv y^{s_1} s_1^{s_2} \pmod{p}.$$

Если сравнение выполняется, то подпись принимается, если нет, то отвергается.

Криптостойкость основана на трудности решения *задачи о дискретном логарифме*: по известным p, g, y, если число p достаточно велико, вычислительно очень трудно найти решение сравнения:

$$y \equiv g^x \pmod{p}.$$

§ 5. Цифровая электронная подпись Шнорра.

Стойкость схемы Шнорра основывается на трудной задаче вычисления дискретных логарифмов.У нас есть два персонажа Алиса и Боб. Алиса хочет передать сообщение Бобу. Для генерации пары ключей сначала выбираются два простых числа, p и p' так, чтобы $p'|p-1$. $g \in Z_p = 0, 1, ..., p-1$ - элемент порядка p' , т.е $g^{p'} \equiv 1 \pmod{p'}$ $y = g^x \pmod{p}$. $0 < x < p'$ - секретный ключ Алисы, подписывающей сообщение M, $0 < M < p$. p, p',g,R все эти числа могут быть свободно опубликованы. Для подписи сообщений используется открытая хэш-функция h.

Предварительные вычисления:

Пользователь Алиса выбирает случайное число $0 < k < p'$ ("эфемерный ключ"), и вычисляет $r = g^k \pmod{p}$.

Подпись сообщения M:

Для того, чтобы подписать сообщение M Алисе необходимо выполнить следующие действия:

(1) вычисляется $e = h(M||r) \pmod{p'}$,

(2) вычисляется $s = (r + xe) \pmod{p'}$. Подписью являются значения x и e, их нужно выслать Бобу.

Проверка подписи для сообщения M:

Боб вычисляет $r' = g^s * y^{-e}$ и $e = h(M||r')$. Если $e = e'$, то он считает подпись верной.

При одинаковом уровне безопасности длина подписей для подписи Шнорра короче, чем для RSA.

§ 6. Система цифровой электронной подписи DSA.

Определение 1.5.1. DSA — алгоритм с использованием открытого ключа для создания электронной подписи, но не для шифрования. Секретное создание хеш-значения и публичная проверка её — только один человек может создать хеш-значение сообщения, но любой может проверить её корректность. Основан на вычислительной сложности взятия логарифмов в конечных полях.

Для подписывания сообщений необходима пара ключей — открытый и закрытый. При этом закрытый ключ знает только подписывающий сообщение, а открытый — любому желающему проверить подлинность сообщения. Также все знают параметры самого алгоритма.

Выбор хеш-функции $H(x)$. Хеш-функция должна преобразовать любое сообщение в число p, q - простые числа $q|p-1,\ g \in F^* = 0, 1, \ldots, p-1$ и имееет порядок q

Протокол 5.1. Генерация открытого и закрытого ключа

1. Закрытый ключ $x \in (0, q)$
2. Открытый ключ $y = g^x \pmod p$

Открытые параметры - (p, q, g, y). Закрытый параметр x.

Протокол 5.2. Подпись сообщения.

1. Выбораем случайное число $k \in (0; q)$

2. Вычисляется $r = (g^k \pmod p) \pmod q$

3. Вычисляется $s = (k^{-1}(H(m) + x \cdot r)) \pmod q$

4. Выборается другое k, если оказалось, что $r = 0$ или $s = 0$

Подписью - (r, s)

Протокол 5.3. Проверка подписи.

1. Вычисляется $w = s^{-1} \pmod q$

2. Вычисляется $u_1 = (H(m) \cdot w) \pmod q$

3. Вычисляется $u_2 = (r \cdot w) \pmod q$

3. Вычисляется $v = ((g^{u_1} \cdot y^{u_2}) \pmod p) \pmod q$

Подпись верна, если $v = r$

Глава 2. Затемненные (слепые) подписи.

Определение 2.6.1. Затемненная (слепая) подпись (Blind Signature) — разновидность электронной цифровой подписи, особенностью которой является то, что подписывающая сторона не может точно знать содержимое подписываемого документа.

Затемненная (слепая) подпись (blind signature) используется в протоколах электронных платежей, которые основанны на использовании электронной банкноты. Банк выдает влальцу монеты уникальный номер, который и является слепой подписью. Протокол затемненной подписи должен давать возможность подписать сообщение, текст которого не известен подписывающему лицу. В отличие от обычной цифровой подписи слепая подпись позволяет вычислить маску и в последствии снять ее так, что подпись остается верной. Кроме того, наложить и снять маску можно без знания ключа подписи. При снятие маски не должно быть возможности изменить сообщение. Таким образом необходимо использовать односторинии функции, что бы вычислимость функций была только в одну сторону.

§ 1.Затемненная подпись RSA.

Первая реализация слепых подписей была осуществлена Чаумом с помощью криптосистемы RSA:

1) Алиса выбирает случайное число k, взаимно простое с n. Затем она маскирует m, вычисляя: $t = mk^e \pmod{n}$

2) Боб подписывает $t : t^d = (mk^e)^d \pmod{n}$

3) Алиса снимает маскировку с t^d, вычисляя $s = t^d/k \pmod{n}$.

4) Результатом является $s = t^d/k \pmod{n}$

§ 2. Затемненная подпись Эль-Гамаля.

Расмотрим алгоритм в общем виде. Боб распологает секретным ключом x.

Открытый ключ:

a) Простое число p;

b) Эллиптическая кривая ϵ над полем F_p;

c) Точка на P кривой ϵ $(P \neq O)$;

d) Точка $\Pi = [x]P$

открытое сообщение Алисы - это точка M на кривой ϵ.

Алгоритм шифрования:

1) Выбирается случайным образом число k.

2) Вычисляется $C_1 = [k]P$.

3) Вычисляется $C_2 = M + [k]\Pi$.

4) Зашифрованное сообщение - это пара $(_1, C_2)$.

Расшифрование. Боб вычисляет $M = C_2 - [x]C_1$.

Теперь рассмотрим алгоритм поподробнее. Открытым ключом банка является уравнение эллиптической кривой, образующая точка Q, точка P, простой порядок группы p. Оба участника протокола умеют вычислять хэш-функцию h. Секретным ключом банка является показатель l такой, что $P = lQ$. Подпись вырабатывается для сообщения m, $0 < m < p$.

Протокол "слепой"подписи:

1) Банк выбирает случайный показатель $\bar{k}$, $0 < \bar{k} < p'$ вычисляет точку $\bar{R} = \bar{k}Q$, проверяет, что $h(\bar{R}) \neq 0$, и посылает точку $\bar{R}$ пользователю. Если $h(\bar{R}) = 0$, то заменяется случайный показатель.

2) Пользователь проверяет, что точка $\bar{R}$ лежит на кривой, выбирает случайный показатель α, $0 < \alpha < p?$, вычисляет точку $R = \alpha\bar{R}$, проверяет, что $h(R) \neq 0$, вычисляет коэффициент $\beta \equiv \dfrac{h(R)}{h(\bar{R})} \pmod{p'}$ вычисляет замаскированное сообщение для открытого сообщения m и отправляет $\bar{m}$ в банк. Точка $\bar{R}$ должна лежить на кривой, иначе это может быть расценено как попытка банка узнать некоторую информацию о содержании сообщения m.

3) Банк проверяет, что $\bar{m} \neq 0$, вычисляет подпись $\bar{s} \equiv l \cdot h(\bar{R}) + \bar{k} \cdot \bar{m} \pmod{p'}$ для сообщения под маской и отправляет ее пользователю. Если $\bar{m} = 0$, то создание подписи немедленно ведет к раскрытию ключа, в этом случае протокол прерывается.

4) Пользователь проверяет выполненяется ли равенство $\bar{s}Q = h(\bar{R})P + \bar{m}\bar{R}$ для сообщения под маской. Если равенство выполняется, то подпись верна. Затем пользователь снимает маску, вычисляя подпись для исходного сообщения: $s \equiv \bar{s}\beta \pmod{p'}$.

Проверка подписи:

1) Для точки R вычисляется хэш-функция. Если $h(R) = 0$ или $m = 0$, то подпись считается недействительной.

2) Если $h(R) \neq 0$, $m \neq 0$, то проверяется выполнение равенства $sQ = h(R)P + mR$. Если равенство выполняется, то подпись верна

§ 3. Затемненная подпись Шнорра.

Рассмотрим как реализуется протокол слепой подписи на основе электронной цифровой подписи Шнорра. В протоколе участвуют: Покупатель и Банк. Пусть покупатель желает получить подпись к сообщению m

p, q - простые числа $q|p-1$, $g \in F^* = 0, 1, \ldots, p-1$ и имееет порядок q ($g^q \equiv 1 \pmod p$, $g^j \not\equiv 1 \pmod p$ при $0 < j < q$). h - хэш-функция. $y = g^x \pmod p$, $1 < x < q$, где x -секретный ключ банка. Рассмотрим ситуацию когда покупатель снимает со счета сумму t.

Транзакция снятия со счета.

1) Банк генерирует $0 < k < q$ и вычисляет $r = g^k \pmod p$ и отправляет r покуупателю.

2) Покупатель вычисляет a)$r' = rg^{-\varepsilon}y^{-\tau} \pmod p$, где $0 < \varepsilon, \tau < q$ - случайные числа (затемняющие элементы)

   b)$e' = h(m||r')$

   c)$e = e' + \tau \pmod q$ и отсылает банку e

3) Банк подписывает: $s = k - xe \pmod q$ и отправляет s покупателю.

   (Отсюда $g^s = g^k g^{-xe} = ry^{-e} \pmod p$ $r = g^s y^e \pmod p$))

4) Покупатель формирует подпись: (e', s'), где

$$e' = e - \tau \pmod q$$

$$s' = s - \varepsilon \pmod q$$

Таким образом, электронная банкнота - это (m, e', s', t)

Проверка подписи. Возведем g в степень $s' = s - \varepsilon = k - xe - \varepsilon \pmod q$, то получим $g^{s'} = g^k g^{-xe} g^{-\varepsilon} \pmod p = ry^{-e} g^{-\varepsilon} \pmod p$. Но $e = e' + \tau \pmod q$, поэтому $g^{s'} = ry^{-e} g^{-\varepsilon} = ry^{-\tau} g^{-\varepsilon} y^{-e'} \pmod p = r' y^{e'}$. Отсюда получаем $r' = g^{s'} y^{e'} \pmod p$. Далее сравниваем e' и $h(m||r')$ по $\pmod q$. Если они равны, то подпись принимается.

Глава 3.Системы электронных платежей.

§ 1. Общие сведения о системах электронных платежей.

Электронные платежные системы (СЭП). Электронной платежной системой мы называем любой комплекс специальных аппаратных и программных средств, которые позволяют проводить электронные расчеты. В последнее время СЭП стали очень актуальными. Создание систем электронных платежей одно из основных направлений финансовой криптографии. Электронные деньги - это один из видов электронных расчетов. Единица электронной наличности — не что иное, как финансовое обязательство банка или другого финансового учреждения. Создание системы электронных платежей требует от нас решения всех трех задач криптографии:

- обеспечение конфиденциальности
- обеспечение целостности
- обеспечение неотслеживаемости.

В СЭП в первую очередь должно быть невозможно подделать электронные монеты,а во-вторых должна быть неотслеживаемость платежей.Система электронных платежей состоит из набора протоколов, основными из которых являются протоколы, реализующие транзакцию снятия со счета и транзакцию платежа.

В платежных системах обычно три участника:

Рассмотрим технологию расчетов в платежных системах. Банк выбирает большие простые секретные числа p, q. $N = pq$ - открыто.

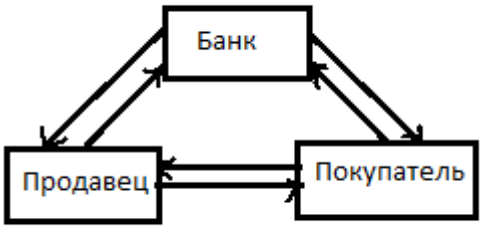

Рис. 1: Участники платежной системы.

$ed = 1 + \Phi(N)t$, где e- открытый ключ, d - секретный ключ. Также дана открытая односторонняя функция $f : Z_N \to Z_N$.

Транзакция снятия со счета:

1) Покупатель генерирует случайное m - номера банкнот.

2) Покупатель вычисляет $f(m)$.

3) Покупатель генерирует случайное число $r \in Z_N$ $Z_N = 0, 1, 2, \cdots, N-1$, $r \neq 0$, НОД$(r; N) = 1$.

4) Покупатель вычисляет $f(n)r^e \pmod N$ и отправляет банку.

5) Банк вычисляет $(f(n)r^e)^d \equiv f(n)^d \pmod N$.

6) Банк отправляет $f(m)r \pmod N$ покупателю.

7) Банк снимает со счета покупателя 1 денежную единицу.

8) Покупатель генерирует r' и t', ввиду того, что НОД$(r; N) = 1$ существуют r' и t' такие, что $rr' = 1 + Nt$

9) Покупатель $f(m)r \pmod N$ умножает на $r' \pmod N$ и получает $f(m) \pmod N$, $(m, f(m)^d \pmod N)$ - электронная монета.

Транзакция платежа: Покупатель посылает продавцу $(m, f(m)^d \pmod N)$

Транзакция депозита: Продавцу нужно, чтобы банк зачислил 1 денежную единицу на его счет.

1) Продавец отправляет банку $(m, f(m)^d \pmod N)$

2) Банк проверяет подпись. Если $(a) = (b)$, то банкнота подлиная.

(a) $m \mapsto f(m) \pmod N$

(b) $f(m)^d \mapsto (f(m)^d)^e \equiv f(m) \pmod N$

3) Банк проверка уникальность монеты: сравнивает $(m, f(m)^d \pmod N)$ с данными, хранящимися у него о прошлых платежах. Если такая банкнота не использовалась ранее, то платеж принимается.

4) $(m, f(m)^d)$ отправляется на хранение.

5) Банк перечисляет на счет продавца 1 денежную единицу и убирает 1 денежную единицу из той базы данных, которая используется для хранения информации о снятых с чьих-то счетов суммах, еще не перечисленных на другие счета.

§ 2. СЭП Брандса.

СЭП Брандса - это анонимная автономная система электронных платежей. Это достаточно эффективная платежная система.За осову данной СЭП берется затемненная подпись Шнорра. В СЭП Брандса используется следующее свойство слепой подписи Шнорра: чтобы пройти аутентификацию нужно знать величину x - секретный ключ.

Для схемы цифровой подписи Шнорра выбираются два достаточно больших простых числа p, q, таких, что $q|p-1$, $g \in G_q$ - образующий элемент подгруппы $\mathbf{Z}_p^*$ порядка q. Случайное число $\in Z_q$ - секретный ключ схемы цифровой подписи. Открытый ключ схемы - набор чисел (p, q, g, h), где $h = g_x$; m - подписываемое сообщение. Подписывающий выбирает $w \in Z_q$, вычисляет $c = (m, a) \in Z_q$, $r = w + cx$ и посылает $[m, c, r]$ проверяющему. Последний должен проверить: $g^r = ah^c \mod p$. Если результат положительный он принимает подпись, в противном случае он её отвергает.

Система электронных платежей Брандса полностью описывается комплектом из пяти протоколов, которые мы и рассмотрим.

1) *Инициализация системы.* Протокол инициализации выполняется один раз при начале работы с СЭП . Банк выбирает тройку порождающих (g, g_1, g_2) группы G_q простого порядка и число $x \in_R \mathbb{Z}_q^*$. Кроме того, он выбирает две хэш-функции H и H_0. H отображает пятерки элементов группы G_q в $\mathbb{Z}_q^*$, а H_0 - пары элементов G_q - в $\mathbb{Z}_q$. Кроме того, H_0 зависит, например, от некоторого значения id, идентифицирующего продавца , а также - от времени и даты t выполнения транзакции. Отмечается, что этот формат функции H_0 выбран лишь

для примера. Банк публикует описание группы G_q (простые числа p и q, если $G_q \subset \mathbb{Z}_p^*$), тройку (g, g_1, g_2) и функции H, H_0 в качестве своего открытого ключа. Секретным ключом банка является число x. В описаниях протоколов появляется также еще некоторое значение h, которое в работе нигде не определяется, но из анализа протоколов можно понять, что $h = g^x$ и это значение должно публиковаться как часть открытого ключа.

Подпись $sign(A, B)$ банка для пары (A, B), $A, B \in G_q$ есть четверка (z, a, b, r), где $z, a, b \in G_q$, $r \in \mathbb{Z}_q$, такая, что

$g^r = h^{H(A,B,z,a,b)}a, A^r = z^{H(A,B,z,a,b)}b.$

2) *Открытие счета.* Протокол открытия счета выполняется однажды для каждого нового плательщика при вводе его в систему электонных платежей. Покупатель выбирает число $u_1 \in_R \mathbb{Z}_q$ и вычисляет $I = g_1^{u_1}$. Если $Ig_2 \neq 1$, то покупатель передает значение I банку, а u_1 хранит в секрете. Для безопасности банка существенно, чтобы значения I были различными для разных клиентов. Банк вычисляет $z = (Ig_2)^x$ и передает это значение покупателю.

3) *Снятие со счета.* Прежде чем снять электронную монету со счета, покупатель должен доказать банку, что он является владельцем данного счета. Затем:

   1. Банк выбирает одно число $w \in_R \mathbb{Z}_q$ и посылает $a = g^w$ и $b = (Ig_2)^w$ покупателю.

2. Покупатель выбирает 3 числа $s \in_R \mathbb{Z}_q^*$, $x_1, x_2 \in_R \mathbb{Z}_q$ и вычисляет $A = (Ig_2)^s$, $B = g_1^{x_1} g_2^{x_2}$ и $z' = z^s$. Кроме того, покупатель выбирает числа $u, v \in_R \mathbb{Z}_q$ и вычисляет $a' = a^u g^v$ и $b' = b^{su} A^v$. Затем он вычисляет $c' = H(A, B, z', a', b')$ и посылает запрос $c = c'/u \bmod q$ банку.

3. Банк посылает ответ $r = (cx + w) \bmod q$ и снимает со счета покупателя соответствующую сумму. Покупатель принимает ответ тогда и только тогда, когда $g^r = h^c a$ и $(Ig_2)^r = z^c b$. Если эти условия выполнены, покупатель вычисляет $r' = (ru + v) \bmod q$. Пара (A, B) и подпись банка (z', a', b', r') для нее образуют электронную монету.

4) *Платеж.* В транзакции платежа покупатель и продавец выполняют следующий протокол.

1. Покупатель посылает продавцу электронную монету: A, B, $sign(A, B)$.

2. Если $A \neq 1$, то продавец вычисляет $d = H_0(A, B, id, t)$, где id - идентификатор продавца, а t – дата и время транзакции. Продавец посылает покупателю значение d.

3. Покупатель вычисляет $r_1 = (d(u_1 s) + x_1) \bmod q$ и $r_2 = (ds + x_2) \bmod q$ и посылает их продавцу. Продавец принимает монету тогда и только тогда, когда $sign(A, B)$ является подписью для (A, B) и $g_1^{r_1} g_2^{r_2} = A^d B$.

5) *Депозит.* Продавец отправляет банку $A, B, sign(A, B), (r_1, r_2)$, а также дату и время транзакции платежа t. Если $A = 1$, то банк не принимает монету. В противном случае он вычисляет d, используя полученные данные и идентификатор id продавца $Seller$. Затем банк проверяет, что $g_1^{r_1} g_2^{r_2} = A^d B$ и что $sign(A, B)$ является подписью для (A, B). Если все корректно, то банк проверяет, не была ли монета потрачена ранее. Если нет, то банк запоминает (A, t, r_1, r_2) и кладет монету на счет продавца.

   Если данная электронная монета уже была потрачена ранее, то у банк есть в наличии две несовпадающие тройки (d, r_1, r_2) и (d', r_1', r_2') и может вычислить идентификатор нарушителя $g_1^{(r_1 - r_1')/(r_2 - r_2')}$.

§ 3. СЭП Чаума.

Общая схема СЭП Шаума. СЭП на базе затемненной подписи - одни из лучших известных анонимных платежных систем, которые наиболее близко имитируют обычную монету. Основной криптографической особенностью этих систем электронных платежей то, что в них используется так называемая затемненная, цифровая подпись, идея которой предложена в работах Чаума (Chaum). Основные этапы ее функционирования таковы.

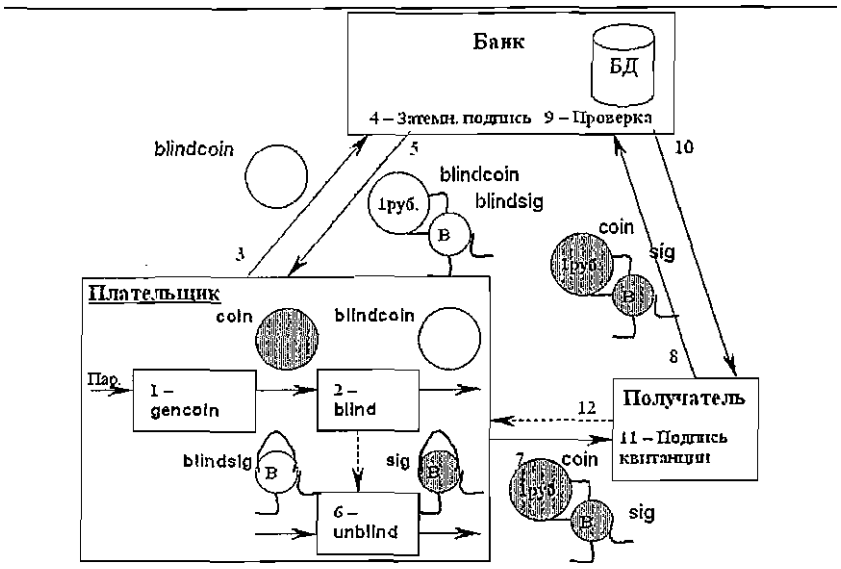

Рис. 2: Основные этапы функционирования.

1) Банк установливает начальные параметры.

   Снятие со счета

2) Используя операцию gencoin плательщик создает монету coin и получает "форму"для монеты котрою дальше использует в платеже.

3) Плательщик затемняет монету операцией blind получает «затемненную» монету blindcoin.

4) Плательщик отправляет «затемненную» монету в банк вместе с требованием о снятии денег со счета, в котором обозначен размер монеты и номер счета.

5) Банк списывает нужную сумму со счета плательщика и подписывает затемненную монету blindcoin подписью blindsig, которая генерируется при помощи специального секретного ключа.

6) Банк отправляет подпись blindsig обратно плательщику, который проверяет ее.

Платеж с депозитом

7) Плательщик снимает затемнение с подписи при помощи операции unblinid и получает цифровую подпись sig, которая является подписью для исходной формы монеты. Для этой операции он использует параметры, сохраненные при выполнении операции blind.

8) Плательщик отправляет (coin, sig) получателю.

9) Получатель просто отправляет это сообщение в банк, чтобы банк проверил тратилась ли повторно монета.

10) Банк проверяет подпись и проверяет по базе данных, что эта монета не была положена на депозит ранее, если эти проверки заканчиваются успешно, он вводит монету в базу данных и добавляет сумму платежа на счет получателя.

11) Банк оповещает получателя о результатах проверок и о депозите. Если предыдущие шаги завершились успешно, получатель подписывает квитанцию, если нет, получатель не принимает платеж.

12) Получатель передает квитанцию плательщику, поставляет ему товары или оказывает услуги.

Криптографическая реализация СЭП Шаума.

Расмотрим реализацию системы основанную на схеме цифровой подписи RSA, концепцию которой мы рассмотрели выше.

1) Банк выбирает ключевую пару схемы подписи RSA для подписания монет: $sk_B = (n, d_B)$, $pk_B = (n, e_B)$. Открытые ключи банк распространяет среди участников системы во время установки начальных параметров. Выбирается также хеш-функция, которая будет использоваться при генерации монет.

2) Плательщик генерирует монету при помощи операции gencoin.Для этого он выбирает случайную величину $c \in_R Z_m$, $|m| = |n| - |hash(c)|$. Монета $coin = (c||hash(c))$

3) Плательщик выполняет затемнение используя операцию blind, т. е. выбирает случайный затемняющий множитель $r \in_R Z_m$ и пологает $blindcoin = coin * r_B^e \pmod{n}$

4) Генерация подписи полностью совпадает с обычной схемой RSA: $blindsig = blindcoin^{d_B} \pmod{n}$. Таким образом, $blindsig = coin^{d_B} * r^{e_B * d_B} = coin^{d_B} * r \pmod{n}$

5) Для снятия затемнения с подписи (операция unblind), плательщик берет r, которое он сохраняет от операции затемнения, и вычисляет $sig = \frac{blindsig}{r}$. Подпись имет вид: $sig = coin^{d_B} \pmod{n}$.

6) Для проверки подписи, построенной таким образом монеты, достаточно обычной проверки подписи по схеме RSA и проверке хеш-кода величины .

§ 4. СЭП из работы Чаума.

Данная система электроннах платежей так же как и предыдущая основана на схеле электронной подписи RSA. Банк выдает банктноты только достоинством в 1 монету.

Пусть $td \equiv 1 \pmod{\varphi(N)}$ $0 < t, d < \varphi(N)$ и $td = 1 + \phi(N)$

Введем некоторые обозначения: Вместо $n^d \pmod{N}$ пишем $n^{\frac{1}{i}} \pmod{N}$ Корень степени равен i-му нечетному простому числу соответствует 2^{i-1} денежных единиц. Что означает: Например, при $i = 1$ степень корня будет равнв 3, а количество денежных единиц $2^{i-1} = 1$, при $i = 2$ степень корня будет равнв 5, количество денежных единиц $2^{i-2} = 2$ и так далее.

номер i	1	2	3	4	5	6
степень корня	3	5	7	11	13	17
количесство единиц	1	2	4	8	16	32

Рис. 3: Таблица количества денежных единиц.

Для того, чтобы получить 1 денежную единицу нужно проделать следующие операции:

1) Покупатель отправляет банку $r^3 f(n) \pmod{N}$, где r- затемняющий множитель.

2) Банк находит d $3d \equiv 1 \pmod{\varphi(N)}$. Число 3 известно покупателю.

3) Банк отправляет покупателю $(r^3 f(n))^d = (r^3 f(n))^{\frac{1}{3}} = r f(n)^{\frac{1}{3}} \pmod{N}$

1 денежная единица в электронной форме - это $(n, f(n))^{\frac{1}{3}} \pmod N$)

Проверка $(f(n)^{\frac{1}{3}})^3 \equiv f(n) \pmod N$

Рассмотрим как снимать другие денежные суммы. Допустим мы хотим снять 4 денежных единицы.

1) Покупатель отправляет банку $r^7 f(n) \pmod N$, где r- затемняющий множитель.

2) Банк находит d $7d \equiv 1 \pmod{\varphi(N)}$. Число 7 известно покупателю.

3) Банк отправляет покупателю $(r^7 f(n))^d = (r^7 f(n))^{\frac{1}{7}} = rf(n)^{\frac{1}{7}} \pmod N$

Составные суммы.

Любую сумму представима однозначно в виде числа:

$$sum = 2^{i_1 - 1} + 2^{i_2 - 1} + \cdots + 2^{i_n - 1}$$

$i_1, i_2, \ldots, i_n$ - номера нечетных простых чисел. $p_{i_1}, p_{i_2}, \ldots, p_{i_l}$ - любые натуральные числа.

Рассмотрим $t = p_{i_1}, p_{i_2}, \ldots, p_{i_l} < \varphi(N) = (p-1)(q-1)$.

Банк находит d $td \equiv 1 \pmod{\varphi(N)}$.

Покупатель $r^f(n) \pmod N$

Банк отправляет покупателю $(r^t f(n))^d \pmod N$

Электронные деньги соответствующие сумме sum - это $(n, f(n)^d \pmod N)$

Рассмотрим пример когда банк выдает только суммы кратые 15 денежным единицам, а покупателю нужно заплатить только 5 денежных единиц.

Распишем 15, как $15 = 1 + 2 + 4 + 8 = 2^0 + 2^1 + 2^2 + 2^3$. Тогда подпись банка на банкноте, это - корень h-ой степени, где $h = 3 * 5 * 7 * 11$. n_1 - номер банкноты продавца. $r^f(n_1) \pmod N$. Сама банкнота имеет вид $(n_1, f(n_1)^d \pmod N)$. Ввведе еще один модуль RSA для копилки: $N_1 = p_1 q_1$. Представим $5 = 1 + 4 = 2^{1-1} + 2^{3-1}$ $w = 3 * 7$. $h = w * 5 * 11$ $w(5 * 11 * d) \equiv 1 \pmod{\varphi(N)}$ $\frac{1}{w} = 5 * 11 * d$

$$f(n_1)^{\frac{1}{w}} = f(n_1)^{5*11*d} = (f(n_1)^d)^5 * 11 \pmod N$$

То есть, чтобы из суммы в 15 денежных единиц сделать сумму в 5 денежных единиц нужно взять $f(n_1)^d \pmod N$ и возвести в степень $5*11$.

Результат: $(n_1, f(n_1)^{\frac{1}{w}} \pmod N)$

Сумма соответствующая 10 денежным единицам - это $(n_1, f(n_1)^{\frac{1}{5*11}} \pmod N)$

Как работает копилка:

1) Выбираем случайное число j, случайный затемняющий множитель s_1 , $\text{НОД}(S_1, N_1) = 1$

2) Покупатель $s_1^{5*11} f(j) \pmod{N_1}$

3) Покупатель отправляет продавцу $(n_1, f(n_1)^{\frac{1}{3*7}} \pmod N), f(j) s_1^{5*11} \pmod{N_1}$

4) Продавец отправляет в банк $(n_1, f(n_1)^{\frac{1}{3*7}} \pmod N), f(j) s_1^{5*11} \pmod{N_1}$

Что делает банк:

1) Проверяет, что $(n_1, f(n_1)^{\frac{1}{3*7}} \pmod N)$ подлиная банкнота в 5 денежных единиц.

2) Была ли банкнота с n_1 ранее использована.

3) Если нет, то 5 денежных единиц переводятся на счет продавца, информация о потраченных n_1 в специальный реестр.

4) Далее банк извещает продавца о конце платежа.

Смысл дальнейшего: Покупатель получает свои 10 денежных единиц с новым номером банкноты j.

Для этого:

1) Банк вычисляет $\frac{1}{5*11} \pmod{\varphi(N_1)}$ то, есть $(5*11)\sigma \equiv 1 \pmod{\varphi(N_1)}$

$$(s_1^{5*11} f(j))^\sigma = s_1 f(j)^\sigma \pmod{N_1}$$

2) Банк отправляет продавцу, а продавец покупателю: $s_1 f(j)^\sigma \pmod{N_1}$

3) Покупатель убирает s_1 $\text{НОД}(s_1, N_1) = 1$ и возвращает банку копилку $(j, f(j) \pmod{N_1})$

4) Банк проверяет $(f(j)^\sigma)^{5*11} = f(j)$

Если проверка сходится, то 10 денежных единиц заносятся на счет покупателя.

Глава 4. Слепые подписи адаптированные для платежных систем.

§ 1.Слепая подпись Шнорра с параметром.

p, q - простые числа $q|p-1$, $g \in F^* = 0, 1, \ldots, p-1$ и имееет порядок q ($g^q \equiv 1 \pmod p$, $g^j \not\equiv 1 \pmod p$ при $0 < j < q$). h - хэш-функция. $y = g^x \pmod p$, $1 < x < q$, где x -секретный ключ банка. Рассмотрим ситуацию когда покупатель снимает со счета сумму t.

Транзакция снятия со счета.

1) Банк генерирует $0 < k < q$ и вычисляет $r = g^k \pmod p$ и отправляет r покуупателю.

2) Покупатель вычисляет a)$r' = rg^{-\varepsilon}y^{-\tau} \pmod p$, где $0 < \varepsilon, \tau < q$ - случайные числа (затемняющие элементы)

   b)$e' = h(m||r')$

   c)$e = e' + \tau \pmod q$ и отсылает банку e и сумму t - не секретную.

3) Банк подписывает: $s = k - t - xe \pmod q$ и отправляет s покупателю.

   (Отсюда $g^s = g^k g^{-t}(g^x)^{-e} = rg^{-t}y^{-e} \pmod p$ $r = g^s y^e g^t \pmod p$)

4) Покупатель формирует подпись: (e', s'), где

$$e' = e - \tau \pmod q$$

$$s' = s - \varepsilon \pmod q$$

Таким образом, электронная банкнота - это (m, e', s', t)

Проверка подписи. Возведем g в степень $s' = s - \varepsilon = k - t - xe - \varepsilon \pmod q$, то получим $g^{s'} = g^k g^{-t} (g^x)^{-e} g^{-\varepsilon} \pmod p = r y^{-e} g^{-\varepsilon} g^{-t} \pmod p$. Но $e = e' + \tau \pmod q$, поэтому $g^{s'} = r y^{-e} g^{-\varepsilon} g^{-t} = r y^{-\tau} g^{-\varepsilon} y^{-e'} g^{-t} \pmod p = r' y^{e'} g^{-t}$. Отсюда получаем $r' = g^{s'} y^{e'} g^t \pmod p$. Далее сравниваем e' и $h(m||r')$ по $\pmod q$. Если они равны, то подпись принимается.

Подпись $s = k - t - xe \pmod q$ легко изменить так, чтобы она соответствовала любой сумме $t_1 < t$

$$s = s_t = k - t - xe \pmod q$$

$$s_{t_1} = s_t + (t - t_1) = k - t_1 - xe \pmod q$$

§ 2. Слепая подпись подпись из работы [5].

Сначала рассмотрим просто затемненную (слепую) подпись из работ [5]. Банк задает начальные параметры: p, q - простые числа $q|p-1$, $g \in F^* = 0, 1, \ldots, p-1$ и имееет порядок q ($g^q \equiv 1 \pmod p$), $g^j \not\equiv 1 \pmod p$ при $0 < j < q$). $y = g^x \pmod p$, $1 < x < q$, где x -секретный ключ банка.

1) Подписывающий выбирает $\hat{k}_1, \hat{k}_2, b_1, b_2 \in Z_q$

2) Вычисляет $\hat{r}_1 = g^{\hat{k}_1} \pmod p$, $\hat{r}_2 = g^{\hat{k}_2} \pmod p$ такие, что $g \subset d(\hat{r}_i, q) = 1$. (Если это не выполняется выбираются другие $\hat{k}_i$).

3) Подписывающий посылает $(\hat{r}_1, \hat{r}_2, b_1, b_2)$ покупателю.

4) Покупатель выбирает случайные числа (a, b, c, d, e).

5) После получения $\hat{r}_1$, $\hat{r}_2$ покупатель вычисляет:

$$r_1 = \hat{r}_1^{ab_1} g^c \pmod p$$

$$r_2 = \hat{r}_2^{bb_2} g^e \pmod p$$

$$r = (r_1, r_2)^d \pmod p$$

6) m- сообщение (номер банкноты). Покупатель также вычисляет:

$$\hat{m}_1 = m\hat{r}_1 \frac{r^{-1}}{2} ad \pmod q$$

$$\hat{m}_2 = m\hat{r}_2 \frac{r^{-1}}{2} bd \pmod q$$

7) Покупатель отправляет в банк $\hat{m}_1$, $\hat{m}_2$.

8) Банк вычисляет:

$$\hat{s}_1 = x\hat{r}_1 + \hat{k}_1 b_1 \hat{m}_1 \pmod{q}$$

$$\hat{s}_2 = x\hat{r}_2 + \hat{k}_2 b_2 \hat{m}_2 \pmod{q}$$

9) Банк отправляет покупателю $\hat{s}_1,\ \hat{s}_2$

10) Покупатель вычисляет:

$$s_1 = \hat{s}_1 \hat{r}_1^{-1} \frac{r}{2} + cdm \pmod{q}$$

$$s_2 = \hat{s}_2 \hat{r}_2^{-1} \frac{r}{2} + edm \pmod{q}$$

$$s = s_1 + s_2 \pmod{q}$$

11) (m, r, s)- электронная банкнота.

Проверка подписи:

Нужно проверить: $g^s \equiv y^r r^m \pmod{p}$

$$\begin{aligned} g^s \equiv g^{s_1+s_2} \pmod{p} &\equiv g^{\hat{s}_1\hat{r}_1^{-1}\frac{r}{2}+\hat{s}_2\hat{r}_2^{-1}\frac{r}{2}+cdm+edm} \pmod{p} \\ &\equiv g^{(x\hat{r}_1+\hat{k}_1 b_1 \hat{m}_1)\hat{r}_1^{-1}\frac{r}{2}+(x\hat{r}_2+\hat{k}_2 b_2 \hat{m}_2)\hat{r}_2^{-1}\frac{r}{2}+cdm+edm} \pmod{p} \\ &\equiv g^{(x\frac{r}{2}+\hat{k}_1 b_1 \hat{m}_1 \hat{r}_1^{-1}\frac{r}{2})+(x\frac{r}{2}+\hat{k}_2 b_2 \hat{m}_2 \hat{r}_2^{-1}\frac{r}{2})+cdm+edm} \pmod{p} \qquad (1) \\ &\equiv g^{xr+\hat{k}_1 b_1 \hat{m}_1 \hat{r}_1^{-1}\frac{r}{2}+\hat{k}_2 b_2 \hat{m}_2 \hat{r}_2^{-1}\frac{r}{2}+cdm+edm} \pmod{p} \\ &\equiv g^{xr+\hat{k}_1 b_1 m \hat{r}_1 \frac{r^{-1}}{2} a d \hat{r}_1^{-1}\frac{r}{2}+\hat{k}_2 b_2 m \hat{r}_2 \frac{r^{-1}}{2} b d \hat{r}_2^{-1}\frac{r}{2}+cdm+edm} \pmod{p} \end{aligned}$$

$$
\begin{aligned}
&\equiv g^{xr+\hat{k}_1 b_1 mad+\hat{k}_2 b_2 mbd+cdm+edm} \pmod{p} \\
&\equiv g^{xr+m(\hat{k}_1 ab_1 d+k_2 bb_2 d+cd+ed)} \pmod{p} \\
&\equiv g^{xr} g^{m(\hat{k}_1 ab_1 d+k_2 bb_2 d+cd+ed)} \pmod{p} \\
&\equiv y^r r^m \pmod{p}
\end{aligned} \tag{2}
$$

С $y = g^x \pmod{p}$ и $r = (r_1 r_2)^d = r_1^d r_2^d = \hat{r}_1^{ab_1 d} g^{cd} \hat{r}_2^{bb_2 d} g^{ed} = g^{\hat{k}_1 ab_1 d+k_2 bb_2 d+cd+ed} \pmod{p}$ проверка проходит успешно.

§ 3. Слепая подпись из работы [5] с параметром.

Теперь изменим предыдущую слепую подпись так, чтобы в процессе формирования затемнной подписи покупатель передал банку параметр t, причем в подписи этот параметр должен сохраниться. Параметр t не является секретным и является частью электронной банкноты.

Введем параметр t в $\hat{s}_1$ и $\hat{s}_1$. Получим:

$$\hat{s}_1 = xt\hat{r}_1 + \hat{k}_1 b_1 \hat{m}_1 \pmod{q}$$

$$\hat{s}_2 = xt\hat{r}_2 + \hat{k}_2 b_2 \hat{m}_2 \pmod{q}$$

Протокол будет иметь следующий вид: p, q - простые числа $q|p-1$, $g \in F^* = 0, 1, \ldots, p-1$ и имееет порядок q ($g^q \equiv 1 \pmod{p}$, $g^j \not\equiv 1 \pmod{p}$ при $0 < j < q$). $y = g^x \pmod{p}$, $1 < x < q$, где x -секретный ключ банка.

1) Подписывающий выбирает $\hat{k}_1, \hat{k}_2, b_1, b_2 \in Z_q$

2) Вычисляет $\hat{r}_1 = g^{\hat{k}_1} \pmod{p}$, $\hat{r}_2 = g^{\hat{k}_2} \pmod{p}$ такие, что $g \subset d(\hat{r}_i, q) = 1$. (Если это не выполняется выбираются другие $\hat{k}_i$).

3) Подписывающий посылает $(\hat{r}_1, \hat{r}_2, b_1, b_2)$ покупателю.

4) Покупатель выбирает случайные числа (a, b, c, d, e).

5) После получения $\hat{r}_1$, $\hat{r}_2$ покупатель вычисляет:

$$r_1 = \hat{r}_1^{ab_1} g^c \pmod{p}$$

$$r_2 = \hat{r}_2^{bb_2} g^e \pmod{p}$$

$$r = (r_1, r_2)^d \pmod{p}$$

6) m- сообщение (номер банкноты). Покупатель также вычисляет:

$$\hat{m}_1 = m\hat{r}_1 \frac{r^{-1}}{2} ad \pmod{q}$$

$$\hat{m}_2 = m\hat{r}_2 \frac{r^{-1}}{2} bd \pmod{q}$$

7) Покупатель отправляет в банк $\hat{m}_1$, $\hat{m}_2$.

8) Банк вычисляет:

$$\hat{s}_1 = xt\hat{r}_1 + \hat{k}_1 b_1 \hat{m}_1 \pmod{q}$$

$$\hat{s}_2 = xt\hat{r}_2 + \hat{k}_2 b_2 \hat{m}_2 \pmod{q}$$

9) Банк отправляет покупателю $\hat{s}_1$, $\hat{s}_2$

10) Покупатель вычисляет:

$$s_1 = \hat{s}_1 \hat{r}_1^{-1} \frac{r}{2} + cdm \pmod{q}$$

$$s_2 = \hat{s}_2 \hat{r}_2^{-1} \frac{r}{2} + edm \pmod{q}$$

$$s = s_1 + s_2 \pmod{q}$$

11) (m, r, s, t)- электронная банкнота.

Проверка подписи:

Нужно проверить: $g^s \equiv y^{rt} r^m \pmod{p}$

$$
\begin{aligned}
g^s &\equiv g^{s_1+s_2} \pmod p \equiv g^{\hat{s}_1\hat{r}_1^{-1}\frac{r}{2}+\hat{s}_2\hat{r}_2^{-1}\frac{r}{2}+cdm+edm} \pmod p \\
&\equiv g^{(xt\hat{r}_1+\hat{k}_1b_1\hat{m}_1)\hat{r}_1^{-1}\frac{r}{2}+(xt\hat{r}_2+\hat{k}_2b_2\hat{m}_2)\hat{r}_2^{-1}\frac{r}{2}+cdm+edm} \pmod p \\
&\equiv g^{(xt\frac{r}{2}+\hat{k}_1b_1\hat{m}_1\hat{r}_1^{-1}\frac{r}{2})+(xt\frac{r}{2}+\hat{k}_2b_2\hat{m}_2\hat{r}_2^{-1}\frac{r}{2})+cdm+edm} \pmod p \\
&\equiv g^{xtr+\hat{k}_1b_1\hat{m}_1\hat{r}_1^{-1}\frac{r}{2}+\hat{k}_2b_2\hat{m}_2\hat{r}_2^{-1}\frac{r}{2}+cdm+edm} \pmod p \\
&\equiv g^{xtr+\hat{k}_1b_1m\hat{r}_1\frac{r^{-1}}{2}ad\hat{r}_1^{-1}\frac{r}{2}+\hat{k}_2b_2m\hat{r}_2\frac{r^{-1}}{2}bd\hat{r}_2^{-1}\frac{r}{2}+cdm+edm} \pmod p \\
&\equiv g^{xtr+\hat{k}_1b_1mad+\hat{k}_2b_2mbd+cdm+edm} \pmod p \\
&\equiv g^{xtr+m(\hat{k}_1ab_1d+k_2bb_2d+cd+ed)} \pmod p \\
&\equiv g^{xtr}g^{m(\hat{k}_1ab_1d+k_2bb_2d+cd+ed)} \pmod p \\
&\equiv y^{rt}r^m \pmod p
\end{aligned}
\tag{3}
$$

С $y = g^x \pmod p$ и $r = (r_1r_2)^d = r_1^dr_2^d = \hat{r}_1^{ab_1d}g^{cd}\hat{r}_2^{bb_2d}g^{ed} = g^{\hat{k}_1ab_1d+k_2bb_2d+cd+ed} \pmod p$ проверка проходит успешно.

Список литературы

[1] Молдовян Н.А., Молдовян А.А. Ввведение в криптосистемы с открытым ключом. — СПб.: БХВ-Петербург, 2005. — 388 с.

[2] Молдовян Н.А., Практикум по криптосистемам с открытым ключом. — СПб.: БХВ-Петербург, 2007. — 304 с.

[3] Смарт Н. Криптография.— М.: Техносфера, 2005.—528 с.

[4] Введение в криптографию/ Под общ. ред. В.В. Ященко.— 3-е изд., доп. — М.: МЦНМО: "ЧеРо", 2000. — 288 с.

[5] Cheng-Chi Lee, Min-Shiang Hwang, Wei-Pang Yang. A new blind signature based on the discrete logarithm problem for untraceability Applied Mathematics and Computation. — 2005. — Vol. 170. — issue 2. — P. 837–841.

[6] Ting Wu, Jin-Rong Wang. Comment: A new blind signature based on the discrete logarithm problem for untraceability Applied Mathematics and Computation. — 2005. — Vol. 164. — issue 3. — P. 999–1005.

[7] Запечников С.В. Криптографические протоколы и их применение в финансовой комерческой деятельности: Учебное пособие для вузов.— М.: Горячая линия-Телеком, 2007. — 320 с.

[8] Burton Rosenberg. Handbook of financial criptography and security. Taylor and Francis Group, LLC. Chapman and Hall/CRC is an imprint of Taylor and Francis Group, an Informa business, —2011. —584 с.

[9] Donal O'Mahony, Michael Peirce, Hitesh Tewari. Electronic Payment Systems for E-Commerce Second Edition. ARTECH HOUSE,INC. 685 Canton Street Notwod, MA 02062. — 2001.— 345 c.