УДК 681.3

ТИПЫ И ПРИЛОЖЕНИЯ ПРОТОКОЛОВ С НУЛЕВЫМ РАЗГЛАШЕНИЕМ СЕКРЕТА

А. А. Демьянчук,
научный сотрудник
А. Ю. Мирин,
канд. техн. наук, старший научный сотрудник
Н. А. Молдовян,
доктор техн. наук, заведующий лабораторией
Санкт-Петербургский институт информатики и автоматизации РАН

Представлены новые варианты протоколов с нулевым разглашением на основе трудности задач дискретного логарифмирования и факторизации. Обсуждается способ доказательства стойкости схем электронной цифровой подписи построением их путем преобразования протоколов с нулевым разглашением секрета. Предложен ряд новых протоколов с нулевым разглашением, включая двухпроходные.

Ключевые слова *— криптографический протокол, аутентификация, открытый ключ, электронная цифровая подпись, задача дискретного логарифмирования, задача факторизации.*

Введение

Протоколы с нулевым разглашением относятся к двухключевым криптосистемам и реализуют процедуры строгой аутентификации удаленных абонентов телекоммуникационных систем, что определяет области их практического применения для обеспечения информационной безопасности современных информационных технологий. Кроме того, протоколы данного типа могут быть использованы как базовый механизм построения алгоритмов электронной цифровой подписи (ЭЦП), а также для обоснования стойкости последних [1]. В типовом случае протоколы без разглашения секрета (с нулевым разглашением секрета) относятся к криптосхемам с открытым ключом (ОК) и представляют собой многораундовую процедуру, в которой типовой раунд выполняется за три интерактивных шага, из которых первые два включают использование случайных значений, а именно, раунд представляет собой выполнение следующих трех шагов:

1) доказывающий (субъект, подлинность которого доказывается в ходе протокола) генерирует разовый случайный секретный ключ, вычисляет по нему разовый ОК и направляет последний проверяющему (пользователю, желающему убедиться в подлинности доказывающего, т. е. в том факте, что доказывающий знает секретный ключ, связанный с ОК);

2) после получения разового ОК проверяющий генерирует запрос в виде случайного бита e и посылает его доказывающему;

3) в зависимости от значения полученного запроса доказывающий вычисляет ответ, который направляет проверяющему.

В каждом раунде доказывающий, если он является подлинным, дает правильный ответ с вероятностью, равной единице, а нарушитель может дать правильный ответ с вероятностью 0,5. Увеличивая число раундов в протоколе, можно понизить вероятность обмана до сколь угодно низкой величины. В случаях применения некоторых трудных задач за счет использования массива ОК (вместо одного ОК), принадлежащих доказывающему, удается построить одно раундовый протокол, включающий три шага, аналогичных указанным ранее. В таких случаях запрос проверяющего на втором шаге генерируется в виде битовой строки достаточно большого размера. Уменьшение числа раундов имеет существенное значение для практического применения протоколов аутентификации пользователей.

Важными для практического использования протоколов с нулевым разглашением является решение следующих задач: 1) уменьшение размера ОК в трехпроходных протоколах и 2) обеспечение достаточной очевидности отсутствия передачи информации о личном секретном ключе (ЛСК) в ходе выполнения протокола.

Термин «нулевое разглашение секрета» (речь идет о нулевой утечке информации о секретном ключе, связанном с ОК) следует понимать в том смысле, что данные, передаваемые доказывающим проверяющему, могли бы быть выработаны проверяющим самостоятельно. Следует отметить, что изначально объявляется некоторая утечка информации о ЛСК, по которому доказывающий вычислил свой ОК. Это состоит в том, что ОК является общедоступным, и по нему теоретически можно вычислить ЛСК, однако это практически нереализуемо, поэтому допускаемая утечка считается приемлемой. Трудность вычисления ЛСК по ОК является верхней границей стойкости протокола. Важным является недопущение какой-либо дополнительной утечки информации о ЛСК в ходе выполнения протокола. Если это обеспечивается, то многократное выполнение протокола практически не снижает его стойкости, т. е. наблюдение атакующим любого числа процедур выполнения протокола не упрощает ему задачу вычисления ЛСК по ОК (теоретически можно допустить, что случайно могут повториться случайные значения запроса, и тогда нарушитель может предоставить правильные ответы, которые он уже наблюдал ранее, однако вероятность такого повтора пренебрежимо мала).

В настоящей статье представлены подходы к построению двухпроходных протоколов с нулевым разглашением, обеспечивающие существенное сокращение размера ОК и очевидное доказательство нулевого разглашения секрета.

Итеративные протоколы

Хорошо известным и апробированным протоколом с нулевым разглашением секрета является протокол Фиата — Шамира [2], в котором доказывающий (пользователь, подтверждающий свою подлинность) доказывает, что он знает значение квадратного корня из некоторого числа t, которое служит ОК. Для того чтобы только подлинный владелец ОК знал значение корня из него, задача извлечения корня должна быть сложной. Это имеет место в случае, если корень извлекается по специально выбранному составному числу. Предполагается, что такое число формируется доверительным центром, который выбирает два больших простых числа p и q и вычисляет значение $n = pq$. Далее уничтожаются числа p и q, а число n используется для формирования пользователями своих ОК. Каждый пользователь выбирает случайное число s такое, что $1 \le s \le n - 1$, и вычисляет значение $t = s^2 \bmod n$. Протокол состоит из многократного повторения раунда, включающего следующие три шага:

1) доказывающий выбирает случайное число v такое, что $1 \le v \le n - 1$, вычисляет значение $q = v^2 \bmod n$, называемое фиксатором, и посылает его проверяющему;

2) проверяющий отправляет доказывающему случайный бит $r \in \{0, 1\}$;

3) доказывающий вычисляет значение $x = vs^r \bmod n$ и направляет его проверяющему. Проверяющий считает ответ положительным, если выполняется соотношение $x^2 = qt^r \bmod n$. В ходе осуществления протокола выполняются z шагов. Вероятность того, что нарушитель (который не знает секрета s) при выполнении одного раунда может дать положительный ответ, равна 2^{-1}, следовательно, вероятность того, что нарушитель может быть принят за пользователя, знающего секрет s, составляет 2^{-z}. Выбирая в протоколе достаточно большое число раундов проверки, можно сделать сколь угодно низкой вероятность обмана.

Для устранения необходимости наличия в протоколе доверительного центра можно предложить использовать трудность задачи извлечения корней большой простой степени по простому модулю со специальной структурой [3], а именно простое число p, имеющее структуру $p = Nk^2 + 1$, где разрядности чисел k и N равны, соответственно, $|k| \ge 160$ бит и $|N| \approx 864$ бит. Выбор таких значений разрядности связан с заданием минимально приемлемого уровня стойкости ЭЦП, равного 2^{80} операций модульного умножения [4]. В протоколе, основанном на трудности извлечения корней k-й степени по модулю p, каждый пользователь выбирает случайное число s такое, что $1 \le s \le p - 1$, и вычисляет значение своего ОК $t = s^k \bmod p$. Протокол состоит из z-кратного повторения следующего трехшагового раунда:

1) доказывающий выбирает случайное число v такое, что $1 \le v \le p - 1$, вычисляет значение $q = v^k \bmod p$ и посылает его проверяющему;

2) проверяющий отправляет доказывающему случайный бит $r = 1$ или $r = 0$;

3) доказывающий вычисляет значение $x = vs^r \bmod p$ и направляет его проверяющему.

Проверяющий считает ответ положительным, если выполняется соотношение $x^k = qt^r \bmod p$. Вероятность обмана составляет 2^{-z}. Следует отметить, что генерируемый на втором шаге случайный запрос проверяющего является принципиальным моментом протокола, поскольку при известном запросе потенциальный нарушитель может легко ввести в заблуждение проверяющего. Рассмотрим две возможные схемы действий нарушителя в одном раунде. В случае ожидаемого запроса $r = 0$ нарушитель выбирает произвольное число v и передает проверяющему значение $q = v^k \bmod p$. Если он получит от проверяющего запрос $r = 0$, то направляет правильный ответ

$x = v$. Однако правильно ответить на запрос $r = 1$ нарушитель не имеет возможности. В случае ожидаемого запроса $r = 1$ нарушитель выбирает произвольное число v и направляет проверяющему число $q' = v^k/t \bmod p$. Если он получит от проверяющего запрос $r = 1$, то направляет ответ $x' = v$, который будет принят проверяющим за правильный, поскольку

$$q't = (v^k/t)t = v^k = x'^k \pmod p.$$

Однако на запрос $r = 0$ нарушитель правильно ответить не сможет.

Трехпроходные протоколы

Рассмотрим реализацию трехпроходного протокола на основе итеративного протокола, использующего ОК вида $t = s^k \bmod p$, где простое 1024-битовое число $p = Nk^2 + 1$ при некотором простом k размером не менее 160 бит, и описанного в предыдущем разделе. В отличие от итеративного протокола в трехпроходном протоколе предполагается, что каждый пользователь в качестве своего ОК имеет h значений $t_i = s_i^k \bmod p$, где s_i — секретные значения, $i = 1, 2, ..., h$. Это позволяет объединить h однобитовых запросов итеративного протокола в единственный h-разрядный запрос E, что обеспечивает сокращение числа проходов до трех в следующем протоколе:

1) доказывающий выбирает случайное число v такое, что $1 \le v \le p - 1$, вычисляет значение фиксатора $q = v^k \bmod p$ и посылает его проверяющему;

2) проверяющий генерирует случайное h-разрядное число $E = (e_1, e_2, ..., e_h)$ и отправляет его доказывающему в качестве своего запроса;

3) доказывающий вычисляет значение

$$W = v\prod_{i=1}^{h} x_i^{e_i} \bmod p$$

и отправляет его в качестве своего ответа на полученный запрос.

Проверяющий считает ответ положительным, если выполняется соотношение

$$W^k = q\prod_{i=1}^{h} t_i^{e_i} \bmod p.$$

Вероятность обмана составляет 2^{-h}, что определяется следующими действиями нарушителя, пытающегося выдать себя за владельца ОК $(t_1, t_2, ..., t_h)$. Нарушитель генерирует случайный запрос $E' = (e'_1, e'_2, ..., e'_h)$ и случайный ответ W, после чего вычисляет значение фиксатора $q = W^k\prod_{i=1}^{h} t_i^{-e_i} \bmod p$. Затем он на первом шаге протокола отправляет проверяющему полученное значение фиксатора, ожидая получить запрос $E = E'$, что является событием, имеющим вероятность 2^{-h}. При наступлении такого события нарушитель успешно проходит процедуру аутентификации.

Двухпроходные протоколы

Рассмотрим двухпроходный протокол с нулевым разглашением, основанный на трудности задачи факторизации, для которого доказательство нулевой утечки секрета является достаточно очевидным. В качестве ОК доказывающего используется натуральное число n, равное произведению двух больших простых чисел r и q, составляющих его ЛСК. Идея доказательства состоит в том, что доказывающий передает проверяющему значение, которое вычислено последним до того, как оно было вычислено доказывающим, поэтому никакой новой информации от доказывающего не передается проверяющему. Протокол включает следующие два шага:

1) проверяющий генерирует случайное 36-битовое число k и, используя метод последовательного возведения в квадрат, вычисляет значение $T = 2^{2^k} \bmod n$, после чего передает доказывающему значение k в качестве своего запроса, на который он ожидает ответ доказывающего;

2) доказывающий выполняет две последовательные операции возведения в степень, в результате чего за короткое время вычисляет значения $K = 2^k \bmod L(n)$, где $L(n)$ — обобщенная функция Эйлера от числа n, и $T = 2^K \bmod n$. Затем он сразу направляет проверяющему значение T в качестве своего ответа на полученный запрос.

Если проверяющий получил правильное значение T, т. е. то значение, которое он вычислил до направления своего запроса доказывающему, в течение временного интервала, длительность которого не превышает некоторое пороговое значение Δ, то им делается вывод о подлинности доказывающего. Значение Δ выбирается достаточно малым. Его определяют исходя из того, что при выборе значений k размером от 30 до 36 бит время вычисления значения $T = 2^{2^k} \bmod n$ методом последовательного возведения в квадрат в 1000 и более раз должно превышать величину Δ. Применение данного протокола на практике требует учета возможных вычислительных ресурсов у потенциального нарушителя. Если предполагается возможность применения нарушителем специализированных производительных ЭВМ (применение многопроцессорных ЭВМ не дает эффекта, так как процесс последовательного возведения в квадрат не может быть распараллелен), то проверяющему требуется выбрать бóльшую разрядность для значения k. Это означает, что ему потребуется потратить больше времени на вычисление значения T. Выбираемая разрядность k

определяется также и быстродействием канала связи, используемого в протоколе. Чем больше быстродействие канала, тем меньшее значение Δ может быть выбрано, т. е. тем меньшая разрядность числа k может быть использована. Последнее означает уменьшение времени вычислений, выполняемых проверяющим до направления своего запроса доказывающему.

Обычно на практике один пользователь связывается со многими другими пользователями, подлинность которых он желает проверить. Это означает, что он должен для каждого из проверяемых установить свое пороговое значение Δ или взять общее пороговое значение $\Delta_{общ}$, равное максимальному времени, требуемому для получения ответа, по всем проверяемым пользователям. В первом случае требуется индивидуальная настройка параметров протокола аутентификации, но достигается меньшее среднее время вычислений на первом шаге. Во втором случае устраняется необходимость индивидуальной настройки параметров, но увеличивается время вычислений на первом шаге.

Использование технического параметра канала связи, связанного с его быстродействием, вносит существенные ограничения на области применения данного протокола. Его значение в выполненном исследовании состоит в том, что он показывает принципиальную возможность построения двухпроходных протоколов с нулевым разглашением и существенного сокращения размера используемого ОК (в десятки и сотни раз в зависимости от допустимого значения вероятности обмана). Устранение привязки к быстродействию канала достигается в следующих двух протоколах. При этом также обеспечивается элементарное доказательство того, что в ходе протокола не происходит утечка информации о секрете (доказательство нулевой утечки: проверяющий получает от доказывающего ответ на свой запрос, который он уже знает).

Первый протокол основан на схеме Диффи — Хеллмана открытого согласования общего секретного ключа [5] и описывается следующим образом. Как и в схеме Диффи — Хеллмана, системными параметрами протокола являются большое простое число p и соответствующий ему первообразный корень $\alpha < p$. Причем для обеспечения стойкости протокола размер числа p должен быть не менее 1024 бит, а разложение числа $p - 1$ на простые множители должно содержать, по крайней мере, один большой простой множитель длины не менее 160 бит. Открытые ключи пользователей генерируются следующим образом. Каждый пользователь выбирает случайный секретный ключ x (длиной не менее 160 бит) и вычисляет ОК y по формуле

$$y = \alpha^x \bmod p.$$

Задаваемая изначально утечка информации о секретном ключе заключается в том, что ОК делается общеизвестным, и любой желающий имеет принципиальную возможность однозначно вычислить значение секретного ключа x, хотя эта возможность практически нереализуема. Многократное выполнение протокола, приводимого далее, не уменьшает сложность реализации указанной потенциальной возможности больше чем на число операций, выполненных в процессе осуществления протокола. Протокол включает следующие два шага:

1) проверяющий генерирует случайное число k и вычисляет значения $U = \alpha^k \bmod p$ и $Z = y^k \bmod p$, где y — ОК доказывающего, после чего передает доказывающему значение U в качестве своего запроса, на который ожидает ответ доказывающего;

2) доказывающий вычисляет значение $Z = U^x \bmod p$, после чего направляет проверяющему значение Z в качестве своего ответа на полученный запрос.

Если проверяющий получил правильное значение Z, т. е. то значение, которое он вычислил до направления своего запроса доказывающему, то им делается вывод о подлинности доказывающего. Возможны различные варианты реализации аналогичных протоколов с использованием различных вариантов построении схемы Диффи — Хеллмана, например, основанных на трудности задачи дискретного логарифмирования в скрытой циклической подгруппе конечной некоммутативной группы [6, 7] или на трудности задачи дискретного логарифмирования на эллиптической кривой [8]. В последнем случае обеспечивается существенное уменьшение вычислительной сложности протокола.

Второй вариант реализации двухпроходного протокола с нулевым разглашением, свободный от привязки к временным параметрам канала связи, основан на использовании алгоритма открытого шифрования. Например, при использовании алгоритма открытого шифрования, подобного криптосистеме RSA [9], протокол описывается следующим образом. Личный секретный ключ и соответствующий ему ОК формируются пользователем следующим путем. Выбираются два больших, не равных между собой простых числа r и q, вычисляется произведение $n = rq$ и значение обобщенной функции Эйлера от $L(n)$, равной наименьшему общему кратному чисел $r - 1$ и $q - 1$. После этого генерируется случайное 32-битовое число e, взаимно простое с $L(n)$, и вычисляется число d, удовлетворяющее условию

$$ed \equiv 1 \bmod L(n).$$

Пара значений n и e является ОК. Тройка чисел r, q и d составляет ЛСК пользователя. Чи-

сла r и q должны иметь специальную структуру, в частности, они должны иметь разрядность не менее 512 бит, и каждое из чисел $r - 1$ и $q - 1$ должно содержать в своем разложении один большой простой множитель. Процедура открытого шифрования сообщения $M < n$ описывается формулой

$$C = M^e \bmod n.$$

Процедура расшифрования криптограммы C описывается формулой

$$M = C^d \bmod n.$$

Корректность процедуры расшифрования легко доказывается с использованием обобщенной теоремы Эйлера, согласно которой для любого числа M, взаимно простого с n, имеет место соотношение

$$M^{L(n)} \equiv 1 \bmod n.$$

Двухпроходный протокол с нулевым разглашением на основе данного алгоритма открытого шифрования имеет следующий вид:

1) проверяющий генерирует случайное сообщение $M < n$ и зашифровывает его по ОК (n, e) доказывающего: $C = M^e \bmod n$. Затем направляет доказывающему значение C в качестве своего запроса, на который ожидает ответ доказывающего;

2) доказывающий расшифровывает криптограмму C по своему ЛСК d, используя формулу $M = C^d \bmod n$. Полученное значение M доказывающий направляет проверяющему в качестве своего ответа на полученный запрос.

Если проверяющий получил правильное значение M, т. е. то значение, которое он сгенерировал на первом шаге протокола, т. е. до направления своего запроса доказывающему, то им делается вывод о подлинности доказывающего. Возможны различные варианты реализации аналогичных протоколов с использованием различных алгоритмов открытого шифрования, например алгоритмом открытого шифрования Эль-Гамаля [10]. При реализации протокола с нулевым разглашением на основе алгоритма Эль-Гамаля, построенного с использованием вычислений на эллиптической кривой, обеспечивается существенное повышение производительности протокола. Также могут быть построены производительные алгоритмы последнего типа при использовании алгоритмов открытого шифрования, основанных на трудности задачи дискретного логарифмирования в скрытой циклической подгруппе конечной некоммутативной группы [11].

Преобразование в схемы цифровой подписи

Протоколы с нулевым разглашением секрета имеют и другое важное применение, которое состоит в синтезе на их основе схем ЭЦП. Многие из протоколов с нулевым разглашением, а именно протоколы с предварительной передачей фиксатора проверяющему, могут быть легко преобразованы в схему ЭЦП, хотя в некоторых случаях размер подписи может быть настолько большим, что практическое применение протоколов будет нецелесообразным. Трехпроходные протоколы с нулевым разглашением позволяют получить схемы ЭЦП, пригодные для практического использования. Примером таких схем является протокол ЭЦП Фиата — Шамира [2, 12], выведенный из одноименного протокола с нулевым разглашением. Общая идея построения схемы ЭЦП на основе протокола с нулевым разглашением заключается в следующем. Подписывающий генерирует конкретное значение фиксатора. Далее, в зависимости от фиксатора и подписываемого документа, он вычисляет значение запроса, после чего вычисляет ответ на запрос. Пара чисел, включающая запрос и ответ, является цифровой подписью к документу. Таким образом, построенные алгоритмы ЭЦП относятся к рандомизированным криптосхемам, для которых к одному и тому же документу может быть выработано практически произвольное число различных подписей. Для того чтобы подделка подписи была практически невозможной, схема ЭЦП строится таким образом, что после вычисления значения запроса вычислительно трудно изменить значение фиксатора. Это может быть обеспечено вычислением значения запроса как значения стойкой хэш-функции от значения фиксатора с присоединенным к нему сообщением. В этом случае реализуется зависимость запроса от каждого бита фиксатора и каждого бита подписываемого документа. Преобразуем в соответствии с этой схемой трехпроходный протокол из предыдущего раздела в схему ЭЦП. Пусть подписывающий владеет ОК $(t_1, t_2, ..., t_h)$, где $t_i = s_i^k \bmod p$; k^2 — большой простой делитель числа $p - 1$; s_i — секретные значения, $i = 1, 2, ..., h$. Алгоритм формирования подписи к сообщению M зададим в следующем виде:

1) подписывающий генерирует случайное число $v < p$ и вычисляет значение фиксатора $q = v^k \bmod p$;

2) затем он, используя некоторую заранее оговоренную h-битовую хэш-функцию F_H, вычисляет значение $E = F_H(q, M) = (e_1, e_2, ..., e_h)$, являющееся первым элементом генерируемой ЭЦП;

3) далее он вычисляет значение

$$W = v\prod_{i=1}^{h} x_i^{e_i} \bmod p,$$

являющееся вторым элементом генерируемой ЭЦП.

Проверка подлинности ЭЦП (E, W) состоит в неявно заданной проверке выполнимости соотношения $W^k = q\prod_{i=1}^{h} t_i^{e_i} \bmod p$, используемого для проверки правильности ответа доказывающего в протоколе аутентификации с нулевым разглашением секрета. Поскольку значение q не задано в явном виде, процедура проверки подписи включает следующие шаги:

1) вычисляется значение фиксатора

$$q = W^k\prod_{i=1}^{h} t_i^{-e_i} \bmod p;$$

2) вычисляется значение хэш-функции $F_H(q, M) = E' = (e'_1, e'_2, ..., e'_h)$;

3) сравниваются значения E' и E. Если $e'_i = e_i$ для всех $i = 1, 2, ..., h$, то подпись принимается как подлинная. В противном случае подпись отклоняется как ложная.

Доказательство стойкости алгоритмов ЭЦП

Для обоснования безопасности схем ЭЦП, основанных на сложности задачи дискретного логарифмирования, может быть использован подход, основанный на синтезе протоколов с нулевым разглашением, из которых выводится схема ЭЦП, стойкость которой следует обосновать. В рамках данного подхода можно показать, что в ряде известных схем ЭЦП размер рандомизирующего параметра подписи может быть уменьшен в 2 раза без снижения стойкости. Эта возможность определяется тем, что для получения стойкой схемы ЭЦП достаточно получить низкую вероятность генерации ожидаемого запроса при фиксированном документе, а для устранения атак, связанных с возможностью модифицирования подписываемого документа в уравнение проверки ЭЦП, можно включить дополнительное значение другой хэш-функции (имеющей в 2 раза большую разрядность), вычисляемое только от документа. Рассмотрим обоснование стойкости схемы ЭЦП, построенной в предыдущем разделе.

В предположении, что изменение фиксатора после вычисления запроса $E = F_H(q, M)$ является практически невыполнимой задачей (это фактически является предположением о стойкости используемой хэш-функции), можно констатировать следующие факты.

1. Генерация произвольного числа подписей не упрощает задачу вычисления ЛСК подписывающего по его ОК.

2. Подделка подписи может быть выполнена с вероятностью 2^{-h} при использовании процедур с низкой трудоемкостью. Для получения большой вероятности удачной подделки подписи к заданному документу требуется выполнить порядка 2^h операций умножения по модулю p.

3. Для получения большой вероятности удачной подделки подписи в атаке с возможностью модифицирования подписываемого документа требуется выполнить порядка $2^{h/2}$ операций умножения по модулю p (это значение трудоемкости подделки определяется вычислительной сложностью нахождения коллизии хэш-функции с использованием парадокса о днях рождения [13]). Это означает, что для получения 80-битовой стойкости схемы ЭЦП требуется использовать, по крайней мере, 160-битовую хэш-функцию.

Эти факты позволяют говорить, что построенная из протокола схема ЭЦП является настолько стойкой, насколько стойким является протокол с нулевым разглашением, положенный в ее основу. Покажем, что разработанная схема ЭЦП может быть отнесена к доказуемо стойким криптосхемам в том смысле, что в предположении о стойкости используемой хэш-функции, т. е. о практической невозможности замены фиксатора после получения значения запроса E, алгоритм ее взлома имеет вычислительную сложность одного порядка со сложностью трудной задачи, положенной в ее основу. Пусть имеется некоторая атака, позволяющая подделать подпись без использования слабостей хэш-функции, т. е. вычислить значение ответа W по заданному значению запроса для различных используемых хэш-функций F_H и F'_H. Тогда, используя данную атаку, можно сгенерировать случайное значение фиксатора q и вычислить два правильных ответа W и W' для каждого из случаев использования F_H и F'_H, определяющих получение разных запросов $E = F_H(q, M)$ и $E' = F'_H(q, M)$. Правильные ответы удовлетворяют следующим соотношениям:

$$W^k = q\prod_{i=1}^{h} t_i^{e_i} \bmod p \text{ и } W'^k = q\prod_{i=1}^{h} t_i^{e'_i} \bmod p.$$

Выполнив деление первого соотношения на второе, получаем $\left(W/W'\right)^k = \prod_{i=1}^{h} t_i^{e_i - e'_i} \bmod p$.

Повторяя такую процедуру, можно получить достаточно большое число соотношений последнего вида, из которых легко найти представление элементов ОК ($t_1, t_2, ..., t_h$) в виде $t_i = s_i^k \bmod p$ для некоторого известного значения s_i (для всех значений $i = 1, 2, ..., h$). Это означает, что предположенная атака решает вычислительно трудную задачу извлечения корней большой простой степени k по модулю $p = Nk^2 + 1$, т. е. гипотетическая

атака имеет сложность одного порядка со сложностью решения использованной трудной задачи.

Отметим, что аналогичным способом можно дать формальное доказательство стойкости схем ЭЦП [4, 14] с малым размером ОК, которые основаны на трудности задачи извлечения корней большой простой степени по простому модулю со специальной структурой. Это можно выполнить, предложив протокол с нулевым разглашением, из которого затем вывести схему ЭЦП, стойкость которой требуется обосновать. Примеры такого обоснования стойкости других алгоритмов ЭЦП приводятся в работе [1].

Заключение

В настоящей статье предложен ряд протоколов с нулевым разглашением, которые представляют собой разные варианты решения поставленных исследовательских задач уменьшения размера ОК в протоколах с малым числом проходов и обеспечения достаточной очевидности отсутствия передачи информации о ЛСК в ходе выполнения протокола. Разработанные протоколы совмещают в себе использование ОК сравнительно малого размера и малое число проходов, что имеет существенное практическое значение. Также разработаны новые протоколы с нулевым разглашением, отличающиеся использованием трудности задачи извлечения корней большой простой степени по простому модулю, имеющему специальную структуру, и построена схема ЭЦП путем преобразования одного из предложенных протоколов. Дано формальное доказательство стойкости построенной схемы ЭЦП.

Литература

1. **Молдовян А. А., Молдовян Д. Н., Васильев И. Н., Головачев Д. А.** Протоколы с нулевым разглашением секрета и обоснование безопасности схем цифровой подписи // Вопросы защиты информации. 2011. № 4. С. 6–11.
2. **Молдовян Н. А., Молдовян А. А.** Введение в криптосистемы с открытым ключом. — СПб.: БХВ-Петербург, 2005. — 286 с.
3. **Молдовян Н. А.** Вычисление корней по простому модулю как криптографический примитив // Вестник СПбГУ. 2008. Сер. 10. Вып. 1. С. 101–106.
4. **Молдовян А. А., Молдовян Н. А.** Новые алгоритмы и протоколы для аутентификации информации в АСУ // Автоматика и телемеханика. 2008. № 7. С. 157–169.
5. **Diffie W., Hellman M. E.** New Directions in Cryptography // IEEE Transactions on Information Theory. 1976. Vol. IT-22. P. 644–654.
6. **Молдовян Д. Н.** Примитивы криптосистем с открытым ключом: конечные некоммутативные группы четырехмерных векторов // Информационно-управляющие системы. 2010. № 5. С. 43–50.
7. **Moldovyan D. N.** Non-Commutative Finite Groups as Primitive of Public-Key Cryptoschemes // Quasigroups and Related Systems. 2010. Vol. 18. P. 165–176.
8. **Болотов А. А., Гашков С. Б., Фролов А. Б.** Элементарное введение в эллиптическую криптографию: Протоколы криптографии на эллиптических кривых. — М.: КомКнига, 2006. — 274 с.
9. **Коутинхо С.** Введение в теорию чисел. Алгоритм RSA. — М.: Постмаркет, 2001. — 323 с.
10. **ElGamal T.** A public key cryptosystem and a signature scheme based on discrete logarithms // IEEE Transactions on Information Theory. 1985. Vol. IT-31. N 4. P. 469–472.
11. **Молдовян Д. Н.** Конечные некоммутативные группы как примитив криптосистем с открытым ключом // Информатизация и связь. 2010. № 1. С. 61–65.
12. **Fiat A., Shamir A.** How to prove yourself: Practical solutions to identification and signature problems // Advances in cryptology — CRYPTO'86. Springer-Verlag LNCS, 1987. Vol. 263. P. 186–194.
13. **Pieprzyk J., Hardjono Th., Seberry J.** Fundumentals of Computer Security. — Berlin: Springer-Verlag, 2003. — 677 p.
14. **Moldovyan N. A.** Digital Signature Scheme Based on a New Hard Problem // Computer Science J. of Moldova. 2008. Vol. 16. N 2(47). P. 163–182.