

УДК 519.7

Алгоритм (n, t) -пороговой доверенной цифровой подписи с Арбитром

Толюпа Е.А.

Ярославский государственный университет им. П. Г. Демидова
150000 Россия, г. Ярославль, ул. Советская, 14

e-mail: tolyupa@gmail.com

получена 15 мая 2013

Ключевые слова: доверенные цифровые подписи, пороговые доверенные цифровые подписи, разделение секрета

Предложен алгоритм (n, t) -пороговой доверенной цифровой подписи с Арбитром, позволяющий доверителю делегировать множеству $\mathfrak{P}$, состоящему из n участников, возможность подписывать сообщения от его имени. Доверитель разделяет доверенность между участниками $\mathfrak{P}$, таким образом, что только t ($t < n$) участников и Арбитр, объединившись, могут вычислить подпись. Таким образом, для подписания документа требуется согласие не менее чем t участников. Арбитр участвует в алгоритме в качестве третьего доверенного лица. Он завершает вычисление подписи на основании информации, полученной от t участников. Проверяющий может идентифицировать участников множества $\mathfrak{P}$ и доверителя. Главной особенностью алгоритма является то, что n участников, вычисляя подпись, не могут вычислить значения секретного ключа доверителя и доверенности.

Введение

Существуют ситуации, когда субъекту необходимо делегировать свои полномочия подписывать документы другому лицу. Для решения этой задачи используют механизм доверенных цифровых подписей (ДЦП, proxy signatures). ДЦП позволяют одному пользователю (*доверителю*) делегировать право подписи другому пользователю (*доверенной стороне*). Для делегирования прав доверитель вычисляет и передает (по защищенному или открытому каналу) *доверенность* доверенной стороне, которая на основании информации, содержащейся в доверенности, идентифицирует доверителя и создает пару доверенных ключей (x_p, y_p) , предназначенных соответственно для создания и проверки цифровой подписи по некоторому известному алгоритму. Любой проверяющий должен самостоятельно вычислить ключ y_p , используя открытую информацию из доверенности и открытые ключи доверителя и доверенной стороны.

Основоположниками теории протоколов ДЦП и разработчиками первого такого протокола являются М. Mambo, К. Usuda и Е. Okamoto [1]. Ими были сформулированы первые требования к безопасности протоколов ДЦП [1, 2] и позже были расширены в [3, 4]. Безопасная ДЦП должна удовлетворять следующим требованиям:

1. Проверимость — проверяющий может быть убежден, что подпись поставлена с согласия доверителя.
2. Стойкость к фальсификации — только назначенная доверителем сторона может создать верную доверенную подпись от лица доверителя. Другими словами, участник **A** и третья сторона, не выбранная им в качестве доверенного подписчика, не смогут создать верную ДЦП от имени доверенного участника **B**.
3. Строгая идентификация — каждый может идентифицировать соответствующую доверенную сторону из доверенной подписи.
4. Неотрекаемость — если доверенный подписчик создает подпись под документом, то в дальнейшем он не сможет заявить, что подпись выполнена кем-то другим.
5. Противостояние злоупотреблению — доверенная сторона не должна использовать ключ подписания для целей, не разрешенных доверителем в информации о полномочиях. В случае злоупотребления ответственность доверенного подписчика должна определяться явно.

Для борьбы со злоупотреблением со стороны доверенного участника S. Kim, S. Park и D. Won [3] предложили способ делегирования полномочий, включив в доверенность сообщение о полномочиях (warrant message) доверенной стороны, которое уточняет или ограничивает его права. В случае, если доверитель не может передать доверенность одной доверенной стороне, то для борьбы со злоупотреблением можно воспользоваться алгоритмами пороговых ДЦП [7–11]. Предложенные способы борьбы со злоупотреблением являются неэффективными для решения некоторых прикладных задач.

В работе изложены основные проблемы, возникающие при эксплуатации ДЦП, связанные со злоупотреблением со стороны доверенного лица. Изложен пороговый алгоритм Петерсена [6]. На его базе предложен алгоритм (n, t) -пороговой доверенной цифровой подписи с арбитром. Особенность предложенного алгоритма заключается в том, что множество доверенных лиц может вычислить ДЦП и не может вычислить значение доверенности.

1. Определения и общие обозначения

В рассматриваемых алгоритмах используется аппарат теории чисел: p и q — большие простые числа, причем $q|p-1$; $g \in \mathbb{Z}_p^*$, порядок g равен q и g является общеизвестным. Считается, что вычисление дискретного логарифма — трудоёмкая задача. Соответственно в роли односторонней функции $f(k)$ выступает $g^k \bmod p$.

Пусть **A** — доверитель, а $\mathfrak{P}$ — доверенное множество, состоящее из n доверенных сторон $p_1, \dots, p_n$. Доверитель располагает секретным ключом x_A и открытым ключом $y_A = g^{x_A} \bmod p$. Каждый участник p_i множества $\mathfrak{P}$ располагает ключевой парой (x_i, y_i) , где x_i — секретный и y_i открытый ключи, связанные соотношением $y_i = g^{x_i} \bmod p$. Обозначим через $X = \sum_{i=1}^n x_i \bmod q$ — секретный, а $Y = \prod_{i=1}^n y_i = g^{\sum_{i=1}^n x_i} \bmod p$ — открытый ключи множества $\mathfrak{P}$. Пусть имеется

множество $\mathfrak{PS}$, состоящее из t ($t \leq n$) участников множества $\mathfrak{P}$, и независимый **Арбитр**, которому доверяет **A** и все участники $\mathfrak{P}$.

Следуя [5], с. 260, определим элементарный симметрический многочлен:

$$S_k(X_1, \dots, X_n) = \sum_{1 \leq i_1 < i_2 < \dots < i_k \leq n} X_{i_1} X_{i_2} \dots X_{i_k}$$

$$k = 1, 2, \dots, n$$

Определим функцию от переменных $(1, 2, \dots, n, k)$ следующим образом $\Omega_{j=1}^n(\mathbf{k}) = S_k(1, 2, \dots, n)$. Например, $\Omega_{j=1}^4(3) = 1 \cdot 2 \cdot 3 + 1 \cdot 2 \cdot 4 + 2 \cdot 3 \cdot 4 + 3 \cdot 4 \cdot 1 = 50$ или $\Omega_{j=1, j \neq 2}^4(2) = 1 \cdot 3 + 1 \cdot 4 + 3 \cdot 4 = 19$.

2. Алгоритм Педерсена

В этом разделе описан алгоритм Педерсена [6], на котором основывается авторский алгоритм (n, t) -пороговой доверенной цифровой подписи.

Пусть участники множества $\mathfrak{P}$ хотят выработать ключевую пару (X, Y) таким образом, что только t участников могут восстановить X .

Алгоритм состоит из следующих этапов:

2.1. Подготовительный этап

Каждый участник p_i выбирает случайный многочлен $f_i(z)$ степени не выше $t - 1$ такой, что $f_i(0) = x_i$:

$$f_i(z) = x_i + a_{i1}z + a_{i2}z^2 + \dots + a_{i,t-1}z^{t-1}.$$

Участник p_i вычисляет и отправляет всем участникам $g^{x_i}, g^{a_{i1}}, \dots, g^{a_{i,t-1}}$. Затем p_i вычисляет значение многочлена в точке $z = j$: $f_i(j) \bmod q$ ($j = 1, \dots, n$) и отправляет полученное значение участнику p_j по секретному каналу связи.

2.2. Вычисление секрета

Каждый p_j проверяет корректность полученной части секрета, вычисляя соотношение:

$$g^{f_i(j)} = g^{x_i} (g^{a_{i1}})^j \cdot (g^{a_{i2}})^{j^2} \cdot \dots \cdot (g^{a_{i,t-1}})^{j^{t-1}}.$$

Если условие для всех $f_i(j)$ выполнено, то p_j вычисляет $v_j = \sum_{i=1}^n f_i(j)$ как свою часть общего секрета. Пусть $F(z) = \sum_{i=1}^n f_i(z)$ – сумма многочленов всех участников $\mathfrak{P}$. Очевидно, что $F(0) = \sum_{i=1}^n x_i$.

После выполнения двух предыдущих этапов каждый участник p_j множества знает значение многочлена $F(z)$ в точке $z = j$: $v_j = F(j) = \sum_{i=1}^n f_i(j)$.

3. Недостатки алгоритмов

Алгоритм Педерсена позволяет множеству $\mathfrak{P}$ выработать пару ключей (X, Y) , но если t участников объединятся, то они смогут восстановить многочлены $f_i(z)$, что приведет к компрометации секретных ключей всех участников группы. Этот недостаток ограничивает применение алгоритма.

Пусть проверяющий отказывается доверять одной доверенной стороне, так как опасается злоупотреблений с ее стороны. В этом случае доверителю необходимо разделить доверенность между множеством из n участников таким образом, что подпись может быть вычислена только t участниками. Для этого можно воспользоваться алгоритмом (n, t) -пороговой доверенной цифровой подписи, в котором подпись под документом будет поставлена только в том случае, если t участников согласятся с содержанием документа.

В алгоритмах (n, t) -пороговых ДЦП [7–11] доверитель распределяет секрет (доверенность) между n участниками таким образом, что если t участников объединятся, то они смогут вычислить доверенность.

Например в [7] этап разделения доверенности между доверенными участниками выглядит следующим образом:

Действия доверителя:

1. Формирует информацию о полномочиях m_w ;
2. Генерирует случайное число $r \in \mathbb{Z}_q^*$ и вычисляет $R = g^r \bmod p$;
3. Вычисляет доверенность $s_A = r + x_A \cdot h(m_w || R) \bmod q$;
4. Генерирует многочлен $f_A(z)$ степени $t - 1$ такой, что $f_A(0) = s_A$:

$$f_A(z) = s_A + a_1 z + a_2 z^2 + \dots + a_{t-1} z^{t-1};$$

5. Вычисляет $f_A(i)$ ($i = 1, \dots, n$) и отправляет i -му участнику по секретному каналу. Публикует набор (m_w, R) и $A_j = g^{a_j}$ ($j = 1, \dots, t - 1$).

Видно, что s_A может быть восстановлен t участниками по формуле Лагранжа. В пороговых ДЦП [8–11] на этапе разделения секрета также используется многочлен степени $t - 1$. Если требования к параметрам системы безопасности будут оценены неверно и найдется t злоумышленников, то они смогут вычислить доверенность и передать значение другому множеству участников, которые смогут вычислить ДЦП от имени доверителя. Во избежание описанной ситуации необходимо включать в информацию о полномочиях m_w данные о точном составе множества $\mathfrak{P}$. Это не позволит множеству злоумышленников, отличному от $\mathfrak{P}$, воспользоваться доверенностью после ее компрометации. При каждой проверке ДЦП проверяющий должен контролировать, что все участники, вычислившие ДЦП, включены в состав доверенных, определенных информацией о полномочиях. Для этого проверяющему необходимо убедиться, что m_w содержит идентификационные номера всех участников, вычисливших ДЦП. Подпись считается некорректной, если хотя бы один из подписавших не найден в m_w . Контроль состава участников потребует выполнить t (согласно числу подписавших) или n (согласно числу участников $\mathfrak{P}$) проверок при

каждой верификации ДЦП. Таким образом, включение в m_w информации о составе множества $\mathfrak{P}$ увеличивает количество действий при проверке ДЦП минимум на t операций поиска элемента (идентификационного номера) в массиве из n элементов (множестве доверенных участников в информации о полномочиях). Трудоемкость бинарного поиска одного элемента оценивается как $O(\log n)$. Процедура проверки подписи выполняется многократно, и экономия числа операций является актуальной задачей.

Возможно избежать контроля состава участников, если разделить доверенность так, что вступление в сговор t и более доверенных участников не позволит ее вычислить. В этом случае проверяющему достаточно убедиться в корректности ДЦП, чтобы сделать вывод, что подпись сформирована множеством доверенных участников.

Добиться желаемого возможно в случае применения (n, t) -пороговой ДЦП с Арбитром, в которой доверенность (секрет) распределяется таким образом, что Арбитр и участники множества $\mathfrak{P}$ не способны ее вычислить отдельно друг от друга. Предположим, что при эксплуатации пороговой ДЦП с Арбитром более чем t участников множества $\mathfrak{P}$ оказались злоумышленниками. В этом случае они не смогут восстановить доверенность и, например, передать ее другому множеству. Следовательно, нет необходимости включать в информацию о полномочиях сведения о точном составе $\mathfrak{P}$. Проверяющему достаточно убедиться в корректности ДЦП, чтобы сделать вывод о том, что участники, вычислившие подпись, имеют полномочия от доверителя. Процедура верификации предложенной пороговой ДЦП состоит из двух этапов — вычисление открытого доверенного ключа и проверка подписи с использованием установленного алгоритма. Процедура верификации подписи из [7] вдобавок к перечисленному требует контроля состава участников. Процедура вычисления открытого доверенного ключа и проверки подписи в обоих случаях состоит из операций умножения и возведения в степень в группе $\mathbb{Z}_p^*$. Предложенная реализация не требует контроля состава участников и позволяет экономить t операций поиска элемента в массиве при проверке полномочий доверенных участников.

Алгоритм полезен при решении следующей задачи. Пусть доверителю необходимо проверять и подписывать большое количество документов, поступающих от географически распределенных участников документооборота. Он может быть не в состоянии это сделать из-за ограниченного канала связи до участников и возникающей нагрузки на собственные вычислительные ресурсы. В этом случае он может делегировать право проверять и подписывать корректные документы доверенному участнику. В этом случае проверяющий должен при каждой проверке подписи принимать решение о том, можно ли доверять участнику, который проверил документ и поставил подпись от имени доверителя. В прикладных задачах достаточно сложно убедить проверяющего в благонадежности одного доверенного участника. Таким образом, разумно делегировать полномочия проверки и подписи множеству из n доверенных участников так, что не менее чем t ($t < n$) участников могут вычислить подпись. В этом случае проверяющему не надо принимать решение о доверии одному участнику — он доверяет множеству из t участников. Применение пороговой ДЦП с Арбитром обусловлено необходимостью уменьшить количество операций, требующихся для проверки ДЦП. Если проверяющий использует мобильное

устройство для проверки подписи, то уменьшение числа операций снижает нагрузку на аппаратную часть и позволяет увеличить время автономной работы.

4. Предложенный алгоритм

Алгоритм (n, t) -пороговой ДЦП с Арбитром должен удовлетворять следующим требованиям:

1. Участники множества $\mathfrak{P}$ не имеют возможности вычислить доверенность;
2. Доверитель участвует в алгоритме один раз — в момент вычисления и распределения доверенности между участниками множества $\mathfrak{P}$ и Арбитром;
3. Секретный ключ доверителя x_A должен использоваться многократно, что не приводит к компрометации системы;
4. Арбитр не может самостоятельно вычислить подпись для произвольного документа;
5. Секретный доверенный ключ X_P не зависит от состава участников, которые вычисляют ДЦП.

Для безопасной реализации алгоритма (n, t) -пороговый ДЦП с арбитром необходимо модифицировать алгоритм М. Mambo, К. Usuda и Е. Okamoto [1]. Пусть доверенная сторона располагает секретным ключом x_B и открытым $y_B = g^{x_B} \bmod p$.

Действия доверителя:

1. Генерирует случайное число $r \in \mathbb{Z}_q^*$ и вычисляет $R = g^r \bmod p$;
2. Вычисляет $s_A = (x_A + r \cdot R) \bmod q$ и посылает (s_A, R) доверенной стороне по защищенному каналу связи.

Действия доверенной стороны:

1. Для идентификации доверителя проверяет сравнение: $g^{s_A} = y_A \cdot R^R \pmod{p}$; Если оно выполнено, то
2. Генерирует случайное число $c \in \mathbb{Z}_q^*$ и вычисляет $C = g^c \bmod p$;
3. Вычисляет доверенный секретный ключ:

$$x_p = (s_A + x_B \cdot y_B + c) \bmod q.$$

Действия проверяющего:

1. Зная (из подписи) параметры R и C , открытые ключи доверителя (y_A) и доверенной стороны (y_B), вычисляет доверенный ключ y_p для проверки подписи по правилу:

$$y_p = y_A \cdot R^R \cdot y_B^{y_B} \cdot C \bmod p.$$

В случае, если требуется включить в доверенность информацию о полномочиях, то алгоритм необходимо изменить следующим образом.

Действия доверителя:

1. Формирует информацию о полномочиях m_w ;
2. Генерирует случайное число $r \in \mathbb{Z}_q^*$ и вычисляет $R = g^r \bmod p$;

3. Вычисляет $s_A = (x_A \cdot h(m_w) + r \cdot R) \bmod q$ и посылает (s_A, R, m_w) доверенной стороне по защищенному каналу связи.

Действия доверенной стороны:

1. Для идентификации доверителя проверяет сравнение: $g^{s_A} = y_A^{h(m_w)} \cdot R^R \pmod{p}$.

Действия проверяющего:

1. Зная (из подписи) параметры R, C и m_w , открытые ключи доверителя (y_A) и доверенной стороны (y_B), вычисляет доверенный ключ y_p для проверки подписи по правилу:

$$y_p = y_A^{h(m_w)} \cdot R^R \cdot y_B^{y_B} \cdot C \bmod p.$$

Для простоты изложения будет использоваться модификация без информации о полномочиях.

Алгоритм (n, t) -пороговой ДЦП с арбитром состоит из следующих этапов:

4.1. Разделение доверенности (секрета)

Действия Доверителя:

1. Генерирует случайное число $r \in \mathbb{Z}_q^*$ и вычисляет $R = g^r \bmod p$;
2. Вычисляет доверенность $s_A = x_A + r \cdot R \bmod q$;
3. Генерирует многочлен $f_A(z)$ степени n такой, что $f_A(0) = s_A$:

$$f_A(z) = s_A + a_1 z + a_2 z^2 + \dots + a_{t-1} z^{t-1} + a_n z^n; \quad (1)$$

4. Вычисляет и публикует значения $g^{s_A}, g^{a_1}, g^{a_2}, \dots, g^{a_{t-1}}, g^{a_n}$;
5. Вычисляет $f_A(j)$ ($j = 1, \dots, n$) и отправляет j -му участнику по секретному каналу связи, публикует R . Таким образом, каждый участник $\mathfrak{P}$ получает значение многочлена $f_A(z)$ в точке, соответствующей его порядковому номеру;
6. Направляет Арбитру коэффициент a_n по секретному каналу.

4.2. Действия участников множества $\mathfrak{P}$

Каждый участник p_i имеет пару ключей (x_i, y_i) , связанных соотношением $y_i = g^{x_i} \bmod p$, и может вычислить $Y = \prod_{i=1}^n y_i \pmod{p}$.

4.2.1. Разделение секрета между участниками множества $\mathfrak{P}$

Действия множества $\mathfrak{P}$:

Для каждого $i = 1, 2, \dots, n$ участник p_i производит следующие действия:

1. Генерирует случайное число $c_i \in \mathbb{Z}_q^*$, вычисляет и отправляет всем участникам множества $\mathfrak{P}$ значение $C_i = g^{c_i} \bmod p$.
2. Генерирует многочлен $g_i(z)$, $\deg(g_i(z)) = t - 1$, $g_i(0) = Y \cdot x_i + c_i$:

$$g_i(z) = (x_i \cdot Y + c_i) + b_{i1} z + b_{i2} z^2 + \dots + b_{i,t-1} z^{t-1}.$$

Вычисляет и отправляет всем участникам $\mathfrak{P}$ значения $g^{(x_i \cdot Y + c_i)}, g^{b_{i1}}, g^{b_{i2}}, \dots, g^{b_{i,t-1}}$.

3. Вычисляет значение многочлена $g_i(z)$ в точке $z = j$: $g_i(j)$ ($j = 1, \dots, n; j \neq i$). Отправляет $g_i(j)$ участнику p_j по секретному каналу связи.

4.2.2. Вычисление секрета

Для каждого $j = 1, 2, \dots, n$ участник p_j производит следующие действия:

4. Проверяет корректность полученных частей секрета, вычисляя соотношения:

$$g^{f_A(j)} = g^{s_A} \cdot (g^{a_1})^j \cdot (g^{a_2})^{j^2} \cdot \dots \cdot (g^{a_{t-1}})^{j^{t-1}} \cdot (g^{a_n})^{j^n};$$

$$g^{g_i(j)} = g^{(x_i \cdot Y + c_i)} \cdot (g^{b_{i1}})^j \cdot (g^{b_{i2}})^{j^2} \cdot \dots \cdot (g^{b_{i,t-1}})^{j^{t-1}}, \quad i = 1, \dots, n.$$

5. Если условие выполнено для всех $g_i(j)$ и $f_A(j)$, то p_j вычисляет $f_A(j) + \sum_{i=1}^n g_i(j)$ как свою часть секрета.

Обозначим $G(z) = f_A(z) + \sum_{i=1}^n g_i(z)$. Секретный доверенный ключ равен:

$$X_P = G(0) = s_A + Y \sum_{i=1}^n x_i + \sum_{i=1}^n c_i. \quad (2)$$

Каждый p_j владеет частью секрета $G(j) = f_A(j) + \sum_{i=1}^n g_i(j)$.

После выполнения этих шагов **Арбитр** и участники множества $\mathfrak{P}$ могут участвовать в этапе вычисления ЭЦП для документа.

4.3. Вычисление ЭЦП для документа

Без ограничения общности будем считать, что множество $\mathfrak{PS}$ состоит из первых t участников $(p_1, \dots, p_t)$ множества $\mathfrak{P}$.

Вычисление ЭЦП для документа M выполняется с использованием алгоритма Шнорра.

Для каждого $i = 1, 2, \dots, t$ участник p_i производит следующие действия:

1. Генерирует случайное число $k_i \in \mathbb{Z}_q^*$;
2. Публикует значение $g^{k_i} \bmod p$;
3. Зная опубликованные значения $g^{k_j} \bmod p$ ($j = 1, \dots, t$) всех участников, вычисляет $K = \prod_{j=1}^t g^{k_j} \bmod p$;
4. Вычисляет значение хеш-функции $H(M||K)$.
5. Вычисляет:

$$S_i = k_i \cdot \prod_{j=1, j \neq i}^{n+1} \frac{i-j}{0-j} + G(i) \cdot H(M||K). \quad (3)$$

Действие p_i на этом шаге изменяет коэффициенты $G(z)$ следующим образом:

- свободный член умножается $H(M||K)$ и увеличивается на k_i ;
- коэффициент при z^m умножается на $H(M||K)$ и увеличивается на

$$k_i \cdot (-1)^{n-m} \Omega_{j=1, j \neq i}^{n+1} (n-m) \cdot \prod_{j=1, j \neq i}^{n+1} \frac{1}{-j}.$$

6. Генерирует $n - t + 1$ многочленов степени, $t - 1$:

$$\begin{aligned} d_{i,n}(0) &= k_i \cdot \prod_{j=1, j \neq i}^{n+1} \frac{1}{-j}; \\ d_{i,n-1}(0) &= k_i \cdot (-1)^1 \Omega^{n+1}(1) \cdot \prod_{j=1, j \neq i}^{n+1} \frac{1}{-j}; \\ &\dots \\ d_{i,t}(0) &= k_i \cdot (-1)^{n-t} \Omega^{n+1}(n-t) \cdot \prod_{j=1, j \neq i}^{n+1} \frac{1}{-j}. \end{aligned}$$

Применяя алгоритм Педерсена, участники множества $\mathfrak{PS}$ восстанавливают общие значения $K_n = \sum_{i=1}^t d_{i,n}(0)$, $K_{n-1} = \sum_{i=1}^t d_{i,n-1}(0)$, $\dots$, $K_t = \sum_{i=1}^t d_{i,t}(0)$. Где K_m ($m = t, \dots, n$) – это величина, на которую изменится коэффициент при z^m многочлена $G(z)$ после того, как каждый p_i выполнит шаги 1 – 5 п. 4.3.

Когда придет время восстанавливать секрет, каждый участник множества $\mathfrak{PS}$ отправит Арбитру по секретному каналу набор (i, S_i) . Произвольный участник передаст Арбитру по секретному каналу связи набор $\{K_m \mid m = t, \dots, n\}$.

4.4. Действия Арбитра

Пусть $S(z)$ – многочлен, равный S_i в точке $z = i$, $\deg(S(z)) = n$. Обозначим $S'(z)$ многочлен, состоящий в точности из t первых членов $S(z)$, $\deg(S'(z)) = t - 1$.

Для вычисления ЭЦП Арбитр выполняет следующие шаги:

1. Для каждого $i = 1, \dots, t$ Арбитр вычисляет значения многочлена $S'(z)$ в точке $z = i$:

$$S'(i) = S_i - (K_t) \cdot i^t - \dots - (K_{n-1}) \cdot i^{n-1} - (a_n \cdot H(M||K) + K_n) \cdot i^n.$$

2. Зная t точек многочлена $S'(z)$, вычисляет значение $S'(0)$ по интерполяционной формуле Лагранжа. Полученное значение возвращает участникам множества в качестве подписи для документа M :

$$Sign(M, X_P) = S'(0) = \sum_{i=1}^t k_i + G(0) \cdot H(M||K). \quad (4)$$

Доверенной подписью для M является:

$$\sigma = (Sign(M, X_P), R, H(M||K), y_A, y_1, \dots, y_n, \prod_{i=1}^n C_i)$$

4.5. Проверка подписи

На основе информации, содержащейся в σ , проверяющий самостоятельно вычисляет открытый доверенный ключ:

$$Y_P = y_A \cdot R^R \cdot Y^Y \cdot \prod_{i=1}^n C_i \bmod p.$$

Затем проверяющий вычисляет:

$$K_v = g^{\text{Sign}(M, X_P)} \cdot Y_P^{-H(M||K)}$$

Если $H(M||K) = H(M||K_v)$, то подпись считается корректной. Это означает, что подпись поставлена от имени доверителя с согласия t участников множества $\mathfrak{P}$.

4.6. Добавление новых участников в множество $\mathfrak{P}$

В предложенном алгоритме нет возможности увеличить число участников множества $\mathfrak{P}$ без выбора нового многочлена $f_A(z)$. Пусть доверитель вычислит значение многочлена $f_A(z)$ в точке $z = n + 1$ и передаст его участнику под номером $n + 1$. В этом случае $n + 1$ участников, объединившись, смогут восстановить $f_A(z)$, что приведет к компрометации доверенности.

Если в прикладной задаче потребуется возможность увеличивать число доверенных участников, то это можно решить следующим образом. Доверитель выбирает многочлен $f_A(z)$ степени $s > n$ следующего вида:

$$f_A(z) = s_A + a_1 z + a_2 z^2 + \dots + a_{t-1} z^{t-1} + a_s z^s.$$

В том случае, если $n + 1 \leq s$, то участники множества $\mathfrak{P}$ не могут восстановить многочлен. Таким образом, для добавления участника доверителю необходимо вычислить значение $f_A(n + 1)$ и отправить его новому участнику. Чтобы включить в состав множества $\mathfrak{P}$ нового участника, необходимо заново выполнить п. 4.2, так как добавление участника изменит открытый ключ Y .

5. Подробно о преобразованиях

Многочлен $G(z)$, $\deg(G(z)) = n$, можно восстановить по формуле Лагранжа, зная значения в $n + 1$ различных точках $z_1, z_2, \dots, z_{n+1}$:

$$G(z) = \sum_{i=1}^{n+1} G(z_i) v_i(z), \quad (5)$$

где

$$v_i(z) = \prod_{j=1, j \neq i}^{n+1} \frac{z - z_j}{z_i - z_j}. \quad (6)$$

Так как известны значения $G(z)$ в точках $z_1 = 1, z_2 = 2, \dots, z_{n+1} = n + 1$, то (5) и (6) можно записать следующим образом:

$$G(z) = \sum_{i=1}^{n+1} G(i)v_i(z), \quad (7)$$

$$v_i(z) = \prod_{j=1, j \neq i}^{n+1} \frac{z-j}{i-j}.$$

Теорема 1. *Добавление числа k к значению многочлена $G(z)$ в точке $z = l$ изменит коэффициенты многочлена следующим образом:*

1. коэффициент при свободном члене увеличится на $k \cdot (-1)^n \Omega_{j=1, j \neq i}^{n+1}(n) \cdot \prod_{j=1, j \neq i}^{n+1} \frac{1}{i-j}$;
2. коэффициент при x^m увеличится на $k \cdot (-1)^{n-m} \Omega_{j=1, j \neq i}^{n+1}(n-m) \cdot \prod_{j=1, j \neq i}^{n+1} \frac{1}{i-j}$.

Доказательство. Если к $G(l)$ прибавить k , то (7) будет иметь вид:

$$G(z) = \sum_{i=1, i \neq l}^{n+1} G(i)v_i(z) + (G(l) + k) \cdot v_l(z) = \sum_{i=1}^{n+1} G(i)v_i(z) + k \cdot v_l(z). \quad (8)$$

Приведем многочлен $v_l(z)$ к нормальному виду:

$$\begin{aligned} v_l(z) &= \\ &= \prod_{j=1, j \neq l}^{n+1} \frac{1}{l-j} z^n + (-1)^1 \Omega_{j=1, j \neq l}^{n+1}(1) \cdot \prod_{j=1, j \neq l}^{n+1} \frac{1}{l-j} \cdot z^{n-1} + (-1)^2 \Omega_{j=1, j \neq l}^{n+1}(2) \cdot \prod_{j=1, j \neq l}^{n+1} \frac{1}{l-j} \cdot z^{n-2} + \dots \\ &\quad \dots + (-1)^{n-1} \Omega_{j=1, j \neq l}^{n+1}(n-1) \cdot \prod_{j=1, j \neq l}^{n+1} \frac{1}{l-j} \cdot z + (-1)^n \Omega_{j=1, j \neq l}^{n+1}(n) \cdot \prod_{j=1, j \neq l}^{n+1} \frac{1}{l-j}. \end{aligned} \quad (9)$$

Из (8), (9) нетрудно видеть, что коэффициент при x^m увеличится на

$$k \cdot (-1)^{n-m} \Omega_{j=1, j \neq l}^{n+1}(n-m) \cdot \prod_{j=1, j \neq l}^{n+1} \frac{1}{l-j}, \quad (10)$$

а коэффициент при свободном члене (x^0) увеличится на

$$k \cdot (-1)^n \Omega_{j=1, j \neq l}^{n+1}(n) \cdot \prod_{j=1, j \neq l}^{n+1} \frac{1}{l-j}. \quad (11)$$

□

Для вычисления ЭЦП в (3) случайное число выбирается равным $k_i \in \mathbb{Z}_q^*$, после чего значение точки $G(i)$ увеличивается на $k_i \cdot \prod_{j=1, j \neq i}^{n+1} \frac{i-j}{0-j}$. Подставив это значение в (11), получаем:

$$k_i \cdot \prod_{j=1, j \neq i}^{n+1} \frac{i-j}{0-j} \cdot (-1)^n \Omega_{j=1, j \neq i}^{n+1}(n) \cdot \prod_{j=1, j \neq i}^{n+1} \frac{1}{i-j} = k_i \cdot \frac{(-1)^n \Omega_{j=1, j \neq i}^{n+1}(n)}{\prod_{j=1, j \neq i}^{n+1} -j} = k_i \cdot \frac{\prod_{j=1, j \neq i}^{n+1} j}{\prod_{j=1, j \neq i}^{n+1} j} = k_i$$

Отсюда видно, что свободный член увеличится в точности на целое число k_i .

Подставив $k_i \cdot \prod_{j=1, j \neq i}^{n+1} \frac{i-j}{0-j}$ в (10), можно сделать вывод, что коэффициент при x^m увеличится на

$$k_i \cdot (-1)^{n-m} \Omega_{j=1, j \neq i}^{n+1}(n-m) \cdot \prod_{j=1, j \neq i}^{n+1} \frac{1}{-j}.$$

6. Замечания по стойкости алгоритма

Основной идеей алгоритма является использование многочлена (1) степени n . Распределение секрета между n участниками не позволяет восстановить многочлен, так как для этого необходимо знать значения в $n+1$ точках. Вид многочлена (1) определяется следующим правилом:

Определение 1. Многочлен $f_A(z) = f'(z) + f''(z)$, где $\deg(f'(z)) = t-1$, $\deg(f''(z)) = n$.

$$\begin{aligned} f'(z) &= s_A + \sum_{i=1}^{t-1} a_i \cdot z^i, a_{t-1} \neq 0; \\ f''(z) &= a_n \cdot z^n, a_n \neq 0; \end{aligned}$$

$$f_A(z) = s_A + \sum_{i=1}^{t-1} a_i z^i + a_n z^n = \underbrace{s_A + a_1 \cdot z + a_2 \cdot z^2 + \dots + a_{t-1} \cdot z^{t-1}}_{f'(z)} + \underbrace{a_n \cdot z^n}_{f''(z)}$$

При анализе стойкости алгоритма можно говорить о следующих видах угроз:

1. Раскрытие секретного ключа x_i участника p_i на этапе разделения секрета между участниками множества $\mathfrak{P}$;
2. Раскрытие доверенности, что позволит вычислить доверенный секретный ключ без ведома доверителя;
3. Возможность вычисления Арбитром ДЦП для сообщения без участия t лиц из множества $\mathfrak{P}$.

Пусть после завершения действий п. 4.2 группа из t участников объединится и вычислит значение секрета $g_i(0) = x_i Y + c_i$ участника p_i и значение $\sum_{i=1}^n g_i(0) = Y \sum_{i=1}^n x_i + \sum_{i=1}^n c_i$. Покажем, что это не приведет к компрометации секретного ключа x_i и доверенности s_A . Действительно, зная $g_i(0)$ и Y , из уравнения $g_i(0) = x_i Y + c_i$ нельзя вычислить x_i , так как c_i неизвестно.

Зная $\sum_{i=1}^n g_i(0)$, для вычисления доверенности s_A из равенства (2) можно составить уравнение

$$s_A = G(0) - Y \sum_{i=1}^n x_i - \sum_{i=1}^n c_i = G(0) - \sum_{i=1}^n g_i(0).$$

Многочлен $G(z) = f_A(z) + \sum_{i=1}^n g_i(z)$ имеет степень n и не может быть восстановлен t ($t \leq n$) участниками, соответственно, недостаточно информации для получения $G(0)$ и вычисления доверенности s_A .

Для вычисления значения подписи $S'(0)$ Арбитр располагает следующей информацией:

- старшим коэффициентом многочлена (1) — a_n ;
- наборами (S_i, i) всех участников множества $\mathfrak{P}\mathfrak{S}$;
- набором $\{K_m \mid m = t, \dots, n\}$.

Покажем, что, владея этими данными, Арбитр не сможет вычислить секретный доверенный ключ $X_P = G(0)$, который в дальнейшем позволил бы ему вычислять ДЦП без участия t лиц из множества $\mathfrak{P}$. Сообщение M не является секретным и доступно любому участнику.

Для того, чтобы узнать $G(0)$, необходимо решить уравнение (4):

$$G(0) \cdot H(M||K) = - \sum_{i=1}^t k_i + S'(0).$$

Для решения необходимо вычислить $\sum_{i=1}^t k_i$. Пусть, для получения $\sum_{i=1}^t k_i$, Арбитр воспользуется известными ему значениями $K_t, \dots, K_n$ и составит следующую систему из $n - t + 1$ уравнений, где $n = \deg(f_A(z))$, которую необходимо решить относительно k_i ($i = 1, \dots, t$):

$$\begin{aligned} K_n &= k_1 \prod_{j=1, j \neq 1}^{n+1} \frac{1}{j} + k_2 \prod_{j=1, j \neq 2}^{n+1} \frac{1}{j} + \dots + k_t \prod_{j=1, j \neq t}^{n+1} \frac{1}{j} \\ &\dots \\ K_t &= k_1 \cdot (-1)^{n-t} \prod_{j=1, j \neq 1}^{n+1} \frac{1}{-j} + \dots + k_t \cdot (-1)^{n-t} \prod_{j=1, j \neq t}^{n+1} \frac{1}{-j} \end{aligned}$$

Арбитру необходимо решить систему из $n - t + 1$ уравнений с t неизвестными. Составим из коэффициентов системы матрицу $\mathcal{A}$. Согласно теореме Кронекера–Капелли совместная система имеет одно решение тогда и только тогда, когда ранг матрицы $\mathcal{A}$ равен числу неизвестных. В данном случае $\text{rank} \mathcal{A} \leq n - t + 1$, соответственно система имеет больше одного решения, если $n - t + 1 < t$. Без ограничения общности будем считать, что отличный от нуля минор порядка $n - t + 1$ составлен из коэффициентов при первых $n - t + 1$ неизвестных. Если перенести в каждом из уравнений системы в левую часть все члены с неизвестными $k_{n-t+2}, \dots, k_t$, то левая часть будет содержать $2t - n - 1$ свободных неизвестных k_i ($i = n - t + 2, \dots, t$).

В связи с тем, что $k_i \in \mathbb{Z}_q^*$, получаем, что количество возможных решений равно q^{2t-n-1} . Большой порядок группы $\mathbb{Z}_q^*$ осложняет поиск нужного решения методом перебора. Отсюда следует вывод:

Теорема 2. Для безопасной реализации (n, t) -пороговой доверенной цифровой подписи необходимо, чтобы выполнялось соотношение $\frac{n+1}{2} < t$, где $n = \deg(f_A(z))$.

Если учесть, что n и t – целые числа, то очевидно, что требование Утверждения 2 будет выполняться в случае нечетного n , когда $\frac{n+1}{2} < t$, в случае четного n , когда $\lceil \frac{n+1}{2} \rceil \leq t$, где $\lceil \frac{n+1}{2} \rceil$ – результат от деления, округленный до большего целого.

В п. 4.6 доверитель генерирует многочлен $f_A(z)$, $\deg(f_A(z)) = s > n$. В этом случае справедливо выражение $n < s < 2t - 1$, где n и t – число участников множеств $\mathfrak{P}$ и $\mathfrak{PS}$ соответственно. Утверждение 2 ограничивает возможность доверителя увеличивать количество доверенных сторон, как описано в п. 4.6.

Пусть доверителю необходимо реализовать алгоритм при $n = 8$ без возможности добавлять нового участника в множество $\mathfrak{P}$. В этом случае доверитель в п. 4.1 выбирает многочлен $f_A(z)$ такой, что $\deg(f_A(z)) = n$, тогда исходя из Теоремы 2 следует, что $t \geq 5$. Таким образом можно реализовать схемы предложенного алгоритма $(n=8, t=5)$; $(n=8, t=6)$; $(n=8, t=7)$. Если необходимо реализовать схему с возможностью добавления новых участников, тогда согласно п. 4.6 доверитель выбирает многочлен $f_A(z)$, $\deg(f_A(z)) = s > n$, тогда схему можно безопасно реализовать, если выполняется неравенство $n < s < 2t - 1$. Таким образом можно реализовать схемы $(n=8, t=6)$ с возможностью добавления 2-х новых участников и $(n=8, t=7)$ с возможностью добавления 4-х участников.

Заключение

Таким образом, применяя указанный алгоритм, доверитель может быть уверен, что в случае вступления в сговор n участников множества $\mathfrak{P}$ доверенность не будет скомпрометирована. Подпись для документа может быть сформирована только в том случае, если t участников согласятся с его содержанием. Процедура верификации подписи требует от проверяющего только вычисления доверенного открытого ключа и проверки корректности ДЦП и не требует дополнительных манипуляций по удостоверению полномочий участников множества $\mathfrak{P}$. В случае необходимости последнего проверяющий вынужден контролировать, содержит ли информация о полномочиях идентификационные номера всех участников, вычисливших ДЦП. Это потребует выполнить t (согласно числу подписавших) или n (согласно числу участников $\mathfrak{P}$) проверок при каждой верификации. Предложенный алгоритм позволяет не проверять полномочия подписантов, что экономит минимум t проверок при верификации ДЦП.

Предложенный алгоритм позволяет противостоять злоупотреблению со стороны доверенных подписчиков и может быть использован для борьбы с инсайдерами в организациях или для разграничения доступа в информационных системах с единым центром доверия.

Список литературы

1. Mambo M., Usuda K., and Okamoto E. Proxy signatures: Delegation of the power to sign messages // IEICE Trans. Fundamentals. 1996. V. E79-A. No. 9. P. 1338–1353.
2. Mambo M., Usuda K., Okamoto E. Proxy signatures for delegating signing operation // Proc. of 3rd ACM Conference on Computer and Communications Security (CCS'96). ACM Press, 1996. P. 48–57.
3. Kim S., Park S., and Won D. Proxy signatures, revisited // Information and Communications Security (ICICS'97). LNCS 1334, Springer-Verlag, 1997. P. 223–232.
4. Lee B., Kim H., and Kim K. Strong proxy signature and its applications // Proceedings of the 2001 Symposium on Cryptography and Information Security (SCIS'01), Vol. 2/2. Oiso, Japan, Jan. 23–26, 2001. P. 603–608.
5. Кострикин А.И. Введение в алгебру. Основы алгебры: Учебник для вузов. М.: Физматлит, 1994. (*Kostrikin A.I. Vvedenie v algebru. Osnovy algebrы: Uchebnik dlya vuzov. M.: Fizmatlit, 1994 [in Russian].*)
6. Pedersen T. A Threshold Cryptosystem without a Trusted Party // Eurocrypt 1991. LNCS 547. Springer-Verlag, 1991. P. 522–526.
7. Sun H. M. An efficient nonrepudiable threshold proxy signatures with known signers // Computer Communications. 1999. 22(8). P. 717–722.
8. Hwang M.-S., Lin I.-C., and Lu K.-F. A secure nonrepudiable threshold proxy signature scheme with known signers // International Journal of Informatica. 2000. 11(2). P. 1–8.
9. Hsu C.-L., Wu T.-S., and Wu T.-C. New nonrepudiable threshold proxy signature schemem with known signers // The Journal of Systems and Software. 2001. 58. P. 119–124.
10. Yang C.-Y., Tzeng S.-F. and Hwang M.-S. On the efficiency of nonrepudiable threshold proxy signatures with known signers // The Journal of Systems and Software. 2003. 22(9). P. 1–8.
11. Tzeng S.-F., Hwang M.-S., and Yang C.-Y. An improvement of nonrepudiable threshold proxy signature schemem with known signers // Computers & Security. 2004. 23. P. 174–178.

An Algorithm of (n, t) -Threshold Proxy Signature with an Arbitrator

Tolyupa E.A.

P.G. Demidov Yaroslavl State University, Sovetskaya str., 14, Yaroslavl, 150000, Russia

Keywords: proxy signature, threshold proxy signature, secret sharing

The paper presents an (n, t) -threshold proxy signature scheme with an Arbitrator which enables an original signer to delegate the signature authority to sign a message on behalf of the original signer to proxy group $\mathfrak{P}$ of n members. The original signer distributes the proxy key among the proxy group members in such a way that not less than t proxy signers and the Arbitrator can cooperatively sign messages on behalf of the original signer. Thus, for signing the document it is necessary to have agreements of not less than t members. The Arbitrator is a trusted third party. It receives the information from the t members and completes the calculation of the digital signature. A verifier can identify the original signer and the members of the proxy group $\mathfrak{P}$. The main feature is that n members of the proxy group can not calculate the proxy key and the original signer's secret key.

Сведения об авторе:

Толюпа Евгений Алексеевич,

Ярославский государственный университет им. П.Г. Демидова,
аспирант