Triptych: логарифмически масштабируемые связываемые кольцевые подписи и их применение

Саранг Ноезер (Sarang Noether) и Брендон Гуделл (Brandon Goodell)
Исследовательская лаборатория Monero (Monero Research Lab)
{sarang,surae}.noether@protonmail.com

24 января, 2020

Аннотация

Кольцевые подписи являются распространённой структурой, используемой с целью сокрытия реального подписанта среди неинтерактивного набора публичных ключей, указываемых во время подписания. В отличие от предшествующих подходов, предполагавших линейность размера подписи в рамках группы анонимных подписантов, существующие оптимальные решения требуют либо использования централизованных доверенных настроек, либо создания логарифмически масштабируемых подписей. Тем не менее некоторые подходы также обеспечивают связываемость, свойство, используемое для определения того, подписывал ли ранее подписант какое-либо другое сообщение, возможно, с ограничениями по выбору членов анонимной группы. Нами предлагается Triptych, семейство не требующих доверенных настроек связываемых кольцевых подписей, в основе которых лежат обобщённые доказательства с нулевым разглашением по открытым обязательствам до нуля. Нами приводятся примеры применения Triptych в составе протоколов транзакций с сокрытием подписанта путём расширения схемы до открытых параллельных обязательств в независимых анонимных группах. Подписи являются логарифмически масштабируемыми в рамках размера анонимной группы, и, несмотря на то, что сложность верификации линейна, доказательства допускают возможность групповой верификации. Мы демонстрируем, что в случае с практичным с точки зрения использования размером анонимной группы в составе распределённых протоколов Triptych обеспечивает конкурентные показатели даже при простом варианте реализации.

1 Введение

Будучи впервые представленными в работе [21] wb связи с группами RSA, кольцевые подписи позволили подписывать сообщения при помощи не определяемого предварительно набора публичных ключей без привлечения доверенного менеджера группы. Предшествующим конструкциям не хватало гибкости, и требовались либо централизованные настройки ключей, либо наличие фиксированных групп подписантов. В более поздней работе [2] были предложены более устойчивые модели безопасности, обеспечивающие невозможность фальсификации и анонимность и учитывающие реальные модели угроз, когда у злоумышленника имеется возможность взломать ключи, убедить честных подписантов включить эти ключи в анонимную группу или же заполучить подписи заранее.

Так как кольцевая подпись включает в себя анонимную группу публичных ключей, один из которых принадлежит настоящему подписанту, обнаружение того факта, что для подписания использовался тот же ключ, требуется дополнительное свойство, называемой связываемостью. Связываемая кольцевая подпись [16] позволяет верификаторам определить, подписывал ли (неизвестный) подписант сообщения и другие сообщения. Такая схема может использоваться в процессе выборов, когда необходимо гарантировать анонимность голосования, но при этом свой голос можно было отдать лишь единожды по определённому вопросу. Схема, предложенная в работе [16], имеющая ту же секретную структуру хеша, что и схема, приведённая в работе [22], представляет собой определённый интерес благодаря потенциальной гибкости с точки зрения связывания; несмотря на то, что в этом случае возможность связывания ограничена группами выбираемых, но неизменяемых анонимных подписантов,

она позволяет осуществлять связывание на основе конкретного случая. В ещё одной недавно появившейся работе [1] описано свойство связываемой анонимности, предполагающее использование наборов подписей и ограничение возможного умышленного повреждения ключей. Другие связанные с этим интересные свойства, такие как отслеживаемость [9, 8], подразумевают более сильные возможности, когда попытка подписать два сообщения при помощи одного и того же ключа позволяет верификатору идентифицировать подписанта.

Связываемые кольцевые подписи уже нашли своё применение в некоторых протоколах транзакций с сокрытием подписанта. В этом случае транзакции подтверждаются кольцевой подписью, в которой группа анонимных подписантов состоит из ранее сгенерированных выходов транзакций. Подпись демонстрирует, что подписант обладает приватным ключом к одному из таких выходов, но кто именно является подписантом не раскрывается, а свойство связываемости используется для того, чтобы убедить верификаторов в том, что выход не был использован ранее в другой подписи (что означало бы попытку двойной траты).

Практическим вопросом, в случае с применением связываемых кольцевых подписей в рамках протокола транзакций, является масштабирование размера подписи и времени верификации относительно размера анонимной группы. В случае с распространёнными схемами, подобными тем, что описаны в работах [19, 10], размер подписи и время верификации масштабируются линейно относительно размера анонимной группы, скрывающей подписанта; так как подобные подписи, как правило, включаются в публичную распределённую структуру данных, такую как блокчейн, существует баланс между размером анонимной группы и требованиями к хранению и верификации. Недавняя доработка протокола позволяет избежать ограничений, связанных с размером. Например, в работе [24] авторами предлагается протокол конфиденциальных транзакций, основанный на системе доказательства, размер которой масштабируется логарифмически относительно размера анонимной группы, и включающий в себя способ демонстрации баланса; доказательство диапазона обязательства по сумме облегчается в других схемах, подобных представленной в работе [4]. В работе [15] авторами используется более общий метод доказательства, пригодный для достижения той же цели. Тем не менее протокол демонстрирует улучшения с точки зрения сокращения размера за счёт интеграции доказательств диапазона обязательства напрямую в схему доказательства благодаря логарифмически масштабируемому размеру доказательства.

Другие протоколы транзакций с сокрытием подписанта, в основе которых не лежат связываемые кольцевые подписи, демонстрируют более конкурентные показатели производительности. Например, протоколы, подобные описанному в работе [13], обеспечивают теоретически максимальное сокрытие подписанта за счёт применения, помимо прочих, доказательства Меркла с нулевым разглашением, имеющее крайне малый размер и незначительное время верификации, но это достигается за счёт использования доверенных структурированных настроек, являющихся неизменным атрибутом, лежащим в основе системы доказательства [11]. Как и в этой работе, протокол транзакций, описанный в работе [14], также основан на том, что предлагается в работе [12]. Тем не менее он работает с обязательствами, схожими с теми, что используются в работе [18], например, но также указывает суммы и имеет ограничения по адресации и отслеживанию отправителя.

1.1 Наш вклад

Нами предлагается семейство связываемых кольцевых подписей под названием Triptych. Наши схемы представляют собой связываемую генерализацию системы доказательства одного из многих обязательств по нулю Грота [12] с оптимизацией Бутля [3], использованной для улучшения масштабирования размера доказательства и сложности верификации. В случае реализации самой простой версии Triptych доказывающая сторона демонстрирует, что ей известно открытие обязательства по нулю в рамках набора обязательств, а также, что ею при помощи того же открытия был построен связующий тег, в результате чего была получена связываемая кольцевая подпись. Затем мы изменяем определения кольцевой подписи Грота, включая в них свойство связываемости и ещё одно связываемое с ней свойство невозможности выборки.

В расширение Triptych нами включается множество независимых наборов обязательств. В этом случае доказывающая сторона делает то же, что и ранее, чтобы продемонстрировать, что ей известно открытие обязательства в одном наборе, равно как и структура связующего тега. Тем не менее дока-

зательство также демонстрирует, что доказывающей стороне известно открытие обязательства в том же самом месте во всех других наборах. Такая схема имеет непосредственное применение; в случае с некоторыми протоколами транзакций с сокрытием подписанта входы транзакций являются обязательствами по нулю, в отношении которых подписантом демонстрируется знание открытия. Каждое обязательство сопровождается ещё одним обязательством по сумме входа; гомоморфно смещая эти обязательства, и за счёт тщательного выбора случайности обязательства доказывающая сторона может продемонстрировать сбалансированность определённой транзакции.

Мы показываем, что Triptych позволяет создавать подписи, обладающие конкурентными показателями производительности в сравнении с другими современными связываемыми кольцевыми подписями, используемыми в случае с ограниченными по размеру анонимными группами. Нами подчёркивается, что подобные схемы также требуют линейного времени верификации, а это означает, что размер анонимной группы, используемый на практике, вероятно, будет ограничен по причинам производительности.

2 Предварительная информация

2.1 Публичные параметры

Допустим, $\mathbb{G}$ является циклической группой, в рамках которой задача логарифмирования является сложной, и допустим, что $\mathbb{F}$ является полем скалярных величин $\mathbb{G}$. Допустим $\mathcal{H} : \{0,1\}^* \to \mathbb{F}$ является криптографической хеш-функцией. Допустим, G и H являются генераторами $\mathbb{G}$ с неизвестным отношением по дискретному логарифму. Допустим, $N = n^m$ является параметром размера, где $n > 1$ и $m > 1$. Допустим, $\{G_{j,i}\}_{j,i=0}^{m-1,n-1}$ является набором генераторов $\mathbb{G}$ с неизвестным отношением друг к другу по дискретному логарифму, к G и H. Допустим, U является генератором $\mathbb{G}$. Следует отметить, что все генераторы могут быть созданы с публичной случайностью; например, здесь вполне подойдёт соответствующая хеш-функция с разделением домена. Все подобные публичные параметры могут включать в себя глобальную контрольную строку, известную всем участникам; мы исключаем их из определений алгоритма и транскрипта Фиата-Шамира из соображений удобочитаемости.

2.2 Обязательство Педерсена

Допустим, Com является простой с вычислительной точки зрения гомоморфной схемой обязательства, прекрасно скрывающей подписанта. В данной работе предполагается использование схемы доказательства Педерсена: для $x, r \in \mathbb{F}$ определяем $\mathrm{Com}(x,r) \equiv xG + rH$ как обязательство по значению x с коэффициентом случайности r. Это выражение может быть тривиально расширено до значений матрицы поддержки; для $\{x_{j,i}\}, r \in \mathbb{F}$, определяем $\mathrm{Com}(x,r) \equiv rH + \sum_{j,i} x_{j,i} G_{j,i}$. В частности, следует отметить, что обязательства Педерсена подобным же образом являются гомоморфными.

2.3 Другая система представления

Для целых чисел или элементов поля i, j дельта Кронекера $\delta(i,j)$ равна 1, если $i = j$, и 0, если иначе, при этом выход выбирается для соответствующей группы.

Иногда нами используется нижний индекс в форме i_j, чтобы обозначить разрядность j для i, при которой разделение i берётся по основе n с дополненной длиной m:

$$\sum_{j=0}^{m} i_j n^j = i$$

Данная система представления будет чётко обозначена во избежание путаницы.

3 Протокол: связываемое обязательство по одному из многих

Мы хотим построить схему связываемой кольцевой подписи, в рамках которой подписант, которому известно открытие обязательства, мог бы подписывать сообщения, используя набор, содержащий другие

обязательства, открытия к котором ему были бы неизвестны. Вместе с доказательством знания подписант также предоставляет связующий тег, то есть образ открытия подписывающего обязательства под верифицируемой псевдослучайной функцией, используя метод, описанный в работе [6], представленный ранее в работах [15, 24]. Частично надёжность системы доказательства основана на надлежащем построении этого связующего тега. По получении верификатор может проверить, появлялся ли данный связующий тег ранее в каком-либо из других действительных доказательств; если нет, инъективность гарантирует верификатору, что этот (неизвестный) подписант ранее не создавал какой-либо другой подписи.

Точнее, мы изменяем схему, предложенную Бутлем [3], которая сама по себе является обобщением схемы, предложенной Гротом [12]. . Мы создаём сигма-протокол для следующего отношения:

$$\mathcal{R}_{\text{link}} = \left\{ \{M_i\}_{i=0}^{N-1} \subset, J \in; (l \in \mathbb{Z}, r \in \mathbb{F}) : M_l = rG \text{ and } U = rJ \right\}$$

Протокол представлен на рисунках 1 и 2.

Следует отметить, что этот протокол может быть неинтерактивным, если воспользоваться эвристическим подходом Фиата-Шамира, в рамках которого запрос верификатора создаётся с использованием устойчивой к конфликтам хеш-функции (моделирующей случайный оракул) и транскрипт доказательства [7].

Мы показываем, что сигма-протокол является полным, надёжным и предполагает нулевое разглашение (точные определения являются общепринятыми и содержатся в работе [12]). Неформально нам необходимо, чтобы протокол обладал следующими свойствами:

- *Идеальная полнота.* При наличии известного свидетельства по утверждению в отношении доказательства честная доказывающая сторона всегда может убедить честного верификатора в действительности свидетельства.

- *Особая надёжность.* При наличии утверждения в отношении доказательства, если доказывающая сторона может правильно ответить на множество запросов верификатора, это будет означать возможность выделения свидетельства для этого утверждения.

- *Особое нулевое разглашение для честного верификатора.* При наличии любого утверждения и запроса верификатора можно смоделировать транскрипт, который будет принят честным верификатором без знания им соответствующего свидетельства.

Теорема 1. *Протокол, показанный на рисунках 1 и 2 , является совершенно полным, предполагает особое нулевое разглашение для честного верификатора, а также обладает особой надёжностью* $(m + 1)$.

Доказательство. Доказательство подобно представленному в работе [3].

Сначала нами демонстрируется совершенная полнота. Предположим, верификатор получает доказательство, сгенерированное честной доказывающей стороной. В Уравнении 1 используется тождественное равенство

$$\sum_{i=0}^{n-1} \sigma_{j,i} = 1$$

для всех $0 \leq j < m$. Уравнение 2 является подобным и использует тождественное равенство

$$(\sigma_{j,i})^2 = \sigma_{j,i}$$

$\underline{\mathcal{P}_{\text{link}}(\{M_i\}, J; (l, r)) :}$

- Выбираем случайные $r_A \in \mathbb{F}$ и $\{a_{j,i}\}_{i=1,j=0}^{n-1,m-1} \subset \mathbb{F}$. Set

$$\{a_{j,0}\}_{j=0}^{m-1} \equiv -\sum_{i=1}^{n-1} a_{j,i}$$

  и задаём $A \equiv \text{Com}(a, r_A)$.

- Задаём $\{\sigma_{j,i}\}_{i,j=0}^{n-1,j-1} \subset \mathbb{F}$ так, чтобы $\sigma_{j,i} \equiv \delta(l_j, i)$ (используя наше представление разложения), и выбираем случайное $r_B \in \mathbb{F}$. Задаём $B \equiv \text{Com}(\sigma, r_B)$.

- Выбираем случайное $r_C \in \mathbb{F}$ и задаём $C \equiv \text{Com}(a(1 - 2\sigma), r_C)$.

- Выбираем случайное $r_D \in \mathbb{F}$ и задаём $D \equiv \text{Com}(-a^2, r_D)$.

- Задаём коэффициенты $\{p_{k,j}\}_{k,j=0}^{N-1,m-1}$ так, чтобы

$$p_k(x) \equiv \prod_{j=0}^{m-1} (\sigma_{j,k} x + a_{j,k}) = \delta(l, k)\, x^m + \sum_{j=0}^{m-1} p_{k,j} x^j$$

  для всех $k \in [0, N)$ (используя наше представление разложения).

- Выбираем случайное $\{\rho_j\}_{j=0}^{m-1} \subset \mathbb{F}$.

- Задаём $\{X_j\}_{j=0}^{m-1} \subset$ так, чтобы

$$X_j \equiv \sum_{k=0}^{N-1} p_{k,j} M_k + \rho_j G$$

- Задаём $\{Y_j\}_{j=0}^{m-1} \subset$ так, чтобы

$$Y_j \equiv U \sum_{k=0}^{N-1} p_{k,j} + \rho_j J$$

$\underline{\mathcal{P} \to \mathcal{V} :}$
$A, B, C, D, \{X_j\}, \{Y_j\}$

$\underline{\mathcal{V} \to \mathcal{P} :}$
$\xi \in \{0, 1\}^*$

$\underline{\mathcal{P}(\xi) :}$

- Задаём $\{f_{j,i}\}_{i=1,j}^{n-1,m-1}$ так, чтобы $f_{j,i} \equiv \sigma_{j,i}\xi + a_{j,i}$.

- Задаём $z_A \equiv r_A + \xi r_B$ and $z_C \equiv \xi r_C + r_D$.

- Задаём $z \equiv r\xi^m - \sum_{j=0}^{m-1} \rho_j \xi^j$.

$\underline{\mathcal{P} \to \mathcal{V} :}$
$\{f_{j,i}\}_{j=0,i=1}^{m-1,n-1}, z_A, z_C, z$

Рис. 1: Сигма-протокол для $\mathcal{R}_{\text{link}}$

$\underline{\mathcal{V}_{\text{link}}(\{M_i\}, J):}$

- Для $0 \leq j < m$, допустим, что $f_{j,0} \equiv \xi - \sum_{i=1}^{n-1} f_{j,i}$.
- Принимаем, только если:

$$A + \xi B = \text{Com}(f, z_A) \quad (1)$$

$$\xi C + D = \text{Com}(f(\xi - f), z_C) \quad (2)$$

$$\sum_{k=0}^{N-1} M_k \left(\prod_{j=0}^{m-1} f_{j,k_j} \right) - \sum_{j=0}^{m-1} \xi^j X_j - zG = 0 \quad (3)$$

$$U \sum_{k=0}^{N-1} \left(\prod_{j=0}^{m-1} f_{j,k_j} \right) - \sum_{j=0}^{m-1} \xi^j Y_j - zJ = 0 \quad (4)$$

Рис. 2: Сигма-протокол для $\mathcal{R}_{\text{link}}$ (продолжение)

для всех $0 \leq j < m$. Уравнение 3 содержит:

$$\begin{aligned}
& \sum_{k=0}^{N-1} M_k \left(\prod_{j=0}^{m-1} f_{j,k_j} \right) - \sum_{j=0}^{m-1} \xi^j X_j - zG \\
= & \sum_{k=0}^{N-1} M_k p_k(\xi) - \sum_{j=0}^{m-1} \xi^j \left(\sum_{k=0}^{N-1} p_{k,j} M_k + \rho_j G \right) - zG \\
= & \sum_{k=0}^{N-1} M_k \left(p_k(\xi) - \sum_{j=0}^{m-1} \xi^j p_{k,j} \right) - \sum_{j=0}^{m-1} \xi^j \rho_j G - zG \\
= & \sum_{k=0}^{N-1} M_k \xi^m \delta(l,k) - \sum_{j=0}^{m-1} \xi^j \rho_j G - \left(r\xi^m - \sum_{j=0}^{m-1} \rho_j \xi^j \right) G \\
= & \xi^m r G - \sum_{j=0}^{m-1} \xi^j \rho_j G - \xi^m r G + \sum_{j=0}^{m-1} \xi^j \rho_j G \\
= & 0
\end{aligned}$$

Уравнение 4 решается подобным образом:

$$\begin{aligned}
& U\sum_{k=0}^{N-1}\left(\prod_{j=0}^{m-1} f_{j,k_j}\right) - \sum_{j=0}^{m-1}\xi^j Y_j - zJ \\
= \;& U\sum_{k=0}^{N-1} p_k(\xi) - \sum_{j=0}^{m-1}\xi^j\left(U\sum_{k=0}^{N-1} p_{k,j} + \rho_j J\right) - zJ \\
= \;& U\sum_{k=0}^{N-1}\left(p_k(\xi) - \sum_{j=0}^{m-1}\xi^j p_{k,j}\right) - \sum_{j=0}^{m-1}\xi^j\rho_j J - zJ \\
= \;& U\sum_{k=0}^{N-1}\xi^m\delta(l,k) - \sum_{j=0}^{m-1}\xi^j\rho_j J - \left(r\xi^m - \sum_{j=0}^{m-1}\rho_j\xi^j\right)J \\
= \;& \xi^m U - \sum_{j=0}^{m-1}\xi^j\rho_j G - \xi^m r J + \sum_{j=0}^{m-1}\xi^j\rho_j G \\
= \;& 0
\end{aligned}$$

так как $J = r^{-1}U$ в действительном доказательстве. Следовательно, протокол является совершенно полным.

Затем мы показываем, что протокол является особым с точки зрения нулевого разглашения для честного верификатора. Для этого мы строим имитатор, который при известном случайном вызове верификатора ξ позволяет создать транскрипт доказательства с идентичным для действительного доказательства распределением.

Прежде всего, отметим, что имитатор, представленный в доказательстве Леммы 1 в работе [3] , осуществляет преобразование точно так же, как и в нашем случае. Если имитатор случайным образом равномерно выбирает $B \in$, упомянутая лемма гарантирует правильность моделирования элементов доказательства $A, C, D, z_A, z_C, \{f_{j,i\neq 0}^{(u)}\}$; на базе этого мы можем вычислить каждый элемент $f_{j,0}^{(u)}$. Кроме того, в действительном доказательстве B является независимым элементом и так же распределяется равномерно.

Элементы доказательства $\{X_j\}_{j=1}^{m-1}$ и $\{Y_j\}_{j=1}^{m-1}$ являются независимыми и распределяются в действительном доказательстве равномерно, так как набор $\{\rho_j\}$ является случайным, а задача дискретного логарифмирования в $\mathbb{G}$ сложной, поэтому имитатор и может выбирать равномерно и случайным образом. Верификация требует, чтобы X_0 и Y_0 уникально определялись другими элементами соответствующих наборов как в реальных доказательствах, так и имитатором.

Наконец, z в действительных доказательствах при известном ξ, распределяется равномерно, и имитатор также может выбрать этот элемент равномерно и случайным образом. Следовательно, протокол является особым с точки зрения нулевого разглашения для честного верификатора.

Остаётся показать, что протокол обладает особой надёжностью $(m+1)$, и при этом $m > 1$. Для этого нам необходимо построить механизм извлечения, который при известных действительных ответах $m+1$ на отдельные запросы верификатора $m+1$ по одному и тому же изначальному утверждению сможет создать действительное свидетельство.

Предположим, что для заданного утверждения у нас имеется набор отдельных запросов верификатора $m+1$ $\{\xi_e\}_{e=0}^{m}$, соответствующих уникальным действительным ответам в форме:

$$\left\{\{f_{j,i}^{(e)}\}, \{z_e\}\right\}_{e=0}^{m}$$

На основе 3-особой надёжности, описанной в [3], и $m > 1$ получаем действительные выделенные $\{\sigma_{j,i}\}_{j,i=0}^{m-1,n-1}$ и $\{a_{j,i}\}_{j,i=0}^{m-1,n-1}$, а связывающее свойство Педерсена гарантирует, что (с высокой степенью вероятности) мы получим:

$$f_{j,i}^{(e)} = \sigma_{j,i}\xi_e + a_{j,i}$$

для всех $e \in [0, m]$. Используя выделенные значения, вычисляем

$$p_k(\xi) \equiv \prod_{j=0}^{m-1} (\sigma_{j,k}\xi + a_{j,k})$$

для всех $k \in [0, N)$. Выделение $\{\sigma_{j,i}\}_{j,i=0}^{m-1,n-1}$ тут же даёт нам значение подписывающего индекса l.

Видно, что p_k имеет степень m, только если $k = l$. Следовательно, существуют коэффициенты $\{\overline{X}_j, \overline{Y}_j\}_{j=0}^{m-1}$, уникально вычисленные на основе утверждения и выделенных значений, и уравнения 3 и 4, таким образом, приобретают следующую форму:

$$\begin{aligned} \xi^m M_l + \sum_{j=0}^{m-1} \xi^j \overline{X}_j &= zG \\ \xi^m U + \sum_{j=0}^{m-1} \xi^j \overline{Y}_j &= zJ \end{aligned}$$

Строим матрицу Вандермонда V, в которой ряд e является вектором $(1, \xi_e, \ldots, \xi_e^m)$. Так как все ξ_e являются явными, ряды V охватывают $\mathbb{F}^{m+1}$; следовательно, существуют такие весовые значения $\{\theta_e\}_{e=0}^m$, что полученная линейная комбинация рядов даёт нам вектор $(0, \ldots, 0, 1)$. То есть $\sum_{e=0}^m \theta_e \xi_e^j = \delta(j, m)$.

Таким образом, для каждого из предыдущих двух уравнений можно построить линейную комбинацию по e. Прежде всего:

$$M_l = \sum_{e=0}^m \theta_e \xi_e^m M_l + \sum_{e=0}^m \theta_e \left(\sum_{j=0}^{m-1} \xi_e^j \overline{X}_j \right) = \left(\sum_{e=0}^m \theta_e z_e \right) G$$

Таким образом, мы выделяем $r \equiv \sum_{e=0}^m \theta_e z_e$. Затем:

$$U = \sum_{e=0}^m \theta_e \xi_e^m U + \sum_{e=0}^m \theta_e \left(\sum_{j=0}^{m-1} \xi_e^j \overline{X}_j \right) = \left(\sum_{e=0}^m \theta_e z_e \right) J$$

Это подразумевает, что $rJ = U$, как и требовалось. Следовательно, протокол обладает особой надёжностью $(m + 1)$, что делает доказательство полным. □

4 Безопасность: связываемая кольцевая подпись

Неформально связываемая кольцевая подпись является структурой, позволяющей подписывать сообщения, используя выбираемую реальным подписантом анонимную группу возможных подписантов (называемую *кольцом*). Действительная подпись убеждает верификатора в том, что подписанту известен (по крайней мере) один из приватных ключей членов кольца. Структура является связываемой в том случае, если можно определить, что две подписи были сгенерированы при помощи одного и того же приватного ключа, независимо от членов анонимной группы.

В качестве исходных нами используются определения безопасности, приводимые в работе [12], а определения правильности и невозможности подделки взяты напрямую из более поздних работ, таких как [1]. Тем не менее мы изменяем определение анонимности так, чтобы учитывались связующие теги, и злоумышленнику пришлось бы выбирать по крайней мере между двумя честными подписантами, взломать которых он не успел. Чтобы учесть связующие свойства, предполагаемые нашей структурой, мы используем разумное определение связываемости, предложенное в работе [1] и использующее теоретико-множественный подход. Нами используется прямое определение невозможности фабрикации, когда злоумышленник производит целевую подпись на основе ключа честного подписанта после получения доступа к подписывающему оракулу и оракулу повреждения, а затем производит новую подпись, которая может быть связана.

Более формально структура *связываемой кольцевой подписи* (LRS) представляет собой набор из алгоритмов KeyGen, Sign, Verify и Link, обладающих определёнными свойствами. Предполагается, что каждый алгоритм обладает рядом публичных параметров.

- $\text{KeyGen}(r) \to (x, X)$: генерирует секретный ключ x и соответствующий публичный ключ X опционально с коэффициентом случайности r; не будучи указанным, секретный ключ выбирается равномерно случайным образом.
- $\text{Sign}(x, M, R) \to \sigma$: Генерирует подпись σ для сообщения $M \in \{0,1\}^*$ относительно кольца $R = \{X_1, \ldots, X_n\}$. При этом предполагается, что x является секретным ключом, соответствующим некоторому $X_i \in R$, сгенерированному KeyGen.
- $\text{Verify}(\sigma, M, R) \to \{0,1\}$: Проверяет подпись σ сообщения M относительно кольца R. Выдаёт 0, если подпись отклонена, и 1, если подпись принята.
- $\text{Link}(\sigma, \sigma') \to \{0,1\}$: Определяет, были ли подписи σ и σ' подписаны при помощи одного и того же приватного ключа. Выдаёт 0, если подписи были подписаны разными приватными ключами, и 1, если при помощи одного и того же приватного ключа.

Нам требуется, чтобы LRS обладала свойствами правильности, анонимности, невозможности подделки, связываемости и невозможности фабрикации.

Правильность подписи требует, чтобы честно сгенерированную подпись всегда можно было верифицировать.

Определение 1 (Правильность)**.** Рассмотрим игру между претендентом и злоумышленником по вероятностному полиномиальному времени $\mathcal{A}$:

- Претендент запускает алгоритм $\text{KeyGen} \to (x, X)$ и передаёт ключи $\mathcal{A}$.
- Злоумышленник $\mathcal{A}$ выбирает кольцо таким образом, чтобы $X \in R$, а сообщение $M \in \{0,1\}^*$, и отправляет их претенденту.
- Претендент подписывает сообщение, используя $\text{Sign}(x, M, R) \to \sigma$.

Если $\Pr[\text{Verify}(\sigma, M, R) = 1] = 1$, мы можем утверждать, что LRS является *совершенно правильной*.

Следует отметить, что мы не требуем, чтобы какой-либо из участников кольца (за исключением X) был сгенерирован при помощи KeyGen. Тем не менее на практике распределённые приложения могут потребовать дополнительных ограничений, касающихся публичных ключей, используемых анонимными группами. Это допускает возможность того, что $\mathcal{A}$ выберет участников кольца со злым умыслом.

Невозможность подделки требует, чтобы злоумышленник, не обладающий приватными ключами участников кольца, не мог сгенерировать действительную подпись для какого-либо сообщения, использующего такое кольцо.

Определение 2 (Невозможность подделки)**.** Рассмотрим следующую игру между претендентом и злоумышленником по вероятностному полиномиальному времени $\mathcal{A}$:

- Злоумышленник $\mathcal{A}$ получает доступ к оракулу публичного ключа GenOracle, который (при i^{th}-ном вызове) запускает алгоритм $\text{KeyGen} \to (x_i, X_i)$ и выдаёт значение X_i злоумышленнику $\mathcal{A}$.
- Злоумышленник $\mathcal{A}$ получает доступ к оракулу повреждения $\text{CorruptOracle}(i)$, который выдаёт значение x_i, если оно соответствует запросу, отправленному GenOracle.
- Злоумышленник $\mathcal{A}$ получает доступ к подписывающему оракулу $\text{SignOracle}(X, M, R)$, который запускает алгоритм $\text{Sign}(x, M, R) \to \sigma$ и выдаёт подпись σ злоумышленнику $\mathcal{A}$. При этом предполагается, что X соответствует запросу, отправленному GenOracle, и что $X \in R$.
- Затем $\mathcal{A}$ выводит (σ, M, R) так, чтобы SignOracle не получал запроса $(-, M, R)$, все ключи в R были сгенерированы по запросам, отправленным GenOracle, а также чтобы ни один ключ в R не был повреждён CorruptOracle.

Если $\Pr[\text{Verify}(\sigma, M, R) = 1] \approx 0$, мы можем утверждать, что LRS *нельзя поделать* путём внутреннего повреждения.

Свойство анонимности требует, чтобы (учитывая, что кольцо включает в себя по крайней мере двух неповреждённых участников) злоумышленник мог только догадываться, кто является действительным подписантом честной подписи.

Определение 3 (Анонимность)**.** Рассмотрим игру между претендентом и злоумышленником по вероятностному полиномиальному времени $\mathcal{A}$:

- Злоумышленник $\mathcal{A}$ получает доступ к оракулу публичного ключа GenOracle и оракулу повреждения CorruptOracle.
- Злоумышленник $\mathcal{A}$ выбирает сообщение $M \in \{0,1\}^*$, кольцо R и индексы i_0 и i_1 и отправляет их претенденту. Нам необходимо, чтобы $X_{i_0}, X_{i_1} \in R$ так, чтобы оба ключа были сгенерированы по запросу, отправленному GenOracle, и ни один из ключей не был запрошен CorruptOracle.
- Претендент выбирает равномерно случайный бит $b \in \{0,1\}$, генерирует подпись $\text{Sign}(x_{i_b}, M, R) \to \sigma$ и отправляет её злоумышленнику $\mathcal{A}$.
- Злоумышленник $\mathcal{A}$ выбирает бит $b' \in \{0,1\}$.

Если $\Pr[b' = b] \approx 1/2$ и злоумышленник $\mathcal{A}$ не сделал какого-либо запроса повреждения после получения случайного бита претендента, мы можем утверждать, что LRS является *анонимной*.

Можно увидеть, что данное определение позволяет злоумышленнику повредить или со злым умыслом сгенерировать все, кроме двух ключей, входящих в кольцо. Некоторые определения позволяют злоумышленнику повредить большее количество ключей, но мы покажем, что они не действуют в случае с нашей связываемой структурой, когда злоумышленник, обладающий приватным ключом участника кольца, может определить, является ли этот участник подписантом, изучив связывающий тег подписи.

Свойство связываемости требует, чтобы злоумышленник не мог произвести $k+1$ не связываемых подписей на основе комбинированной группы, состоящей из k публичных ключей.

Определение 4 (Связываемость)**.** Рассмотрим игру между претендентом и злоумышленником по вероятностному полиномиальному времени $\mathcal{A}$:

- Для $i \in [0, k-1]$ злоумышленник $\mathcal{A}$ создаёт публичный ключ X_i, сообщение M_i, кольцо R_i и подпись σ_i.
- Злоумышленник $\mathcal{A}$ создаёт ещё одно сообщение M, кольцо R и подпись σ.
- Все наборы элементов $(X_i, M_i, R_i, \sigma_i)$ и (M, R, σ) отправляются претенденту.
- Претендент проверяет следующее:
  - $|V| = k$, где $V \equiv \bigcup_{i=0}^{k-1} R_i$.
  - чтобы $X_i \in V$.
  - чтобы $R_i \subset V$.
  - $\text{Verify}(\sigma_i, M_i, R_i) = 1$ для всех i.
  - $\text{Verify}(\sigma, M, R) = 1$.
  - для всех $i \neq j$, мы должны получить $\text{Link}(\sigma_i, \sigma_j) = \text{Link}(\sigma_i, \sigma) = 0$.
- Если все проверки будут пройдены, $\mathcal{A}$ выигрывает.

Если $\mathcal{A}$ побеждает в игре с ничтожной вероятностью для всех k, мы можем утверждать, что LRS является *связываемой*.

Невозможность фабрикации требует, чтобы злоумышленник мог сгенерировать подпись, которая будет связана с честной подписью.

Определение 5 (Невозможность фабрикации)**.** Также рассмотрим следующую игру между претендентом и злоумышленником по вероятностному полиномиальному времени $\mathcal{A}$:

- Злоумышленник $\mathcal{A}$ получает доступ к оракулу публичного ключа GenOracle.
- Злоумышленник $\mathcal{A}$ получает доступ к оракулу повреждения CorruptOracle.
- Злоумышленник $\mathcal{A}$ получает доступ к подписывающему оракулу SignOracle.
- Злоумышленник $\mathcal{A}$ выбирает публичный ключ X, сгенерированный по запросу, отправленному GenOracle, но не представленному в качестве запроса CorruptOracle. Выбирает сообщение $M \in \{0,1\}^*$ и кольцо R так, чтобы $X \in R$. Запрашивает SignOracle$(X, M, R) \to \sigma$.
- Затем злоумышленник $\mathcal{A}$ создаёт набор элементов (M', R', σ') и отправляет (M', R', σ') претенденту вместе с (X, M, R, σ).
- Если Verify$(\sigma', M', R') = 0$ или если σ' была получена по запросу, отправленному SignOracle, претендент прерывает процесс.

Если $\Pr[\text{Link}(\sigma, \sigma') = 1 \approx 0$, мы можем утверждать, что LRS *нельзя сфабриковать*.

5 Применение: связываемая кольцевая подпись

Структуры, предлагаемые в работах [12, 3], описывают использование подобного сигма-протокола с целью построения простой схемы кольцевой подписи. Вносимые нами изменения позволяют расширить их благодаря свойствам связываемости и невозможности фабрикации. Мы кратко описываем, как сделать это.

Теорема 2. *Протокол, представленный на рисунке 3, позволяет построить связываемую кольцевую подпись.*

Доказательство. Совершенная правильность вытекает непосредственно из совершенной полноты системы доказательства, используемой $\mathcal{R}_{link}$.

Подобным образом вытекает и анонимность, так как система предполагает особое нулевое разглашение честному верификатору, а следовательно, свидетельство является неотличимым [5]. Таким образом, любое преимущество злоумышленника с точки зрения нарушения анонимности может возникнуть в результате появления возможности отличить либо обязательства по входу, либо связующие теги подписей. Так как честно сгенерированные обязательства Педерсена по входу обеспечивают совершенное сокрытие, они неотличимы от элементов $\mathbb{G}$, равномерно выбранных случайным образом; мы допускаем по определению, что по крайней мере два таких обязательства присутствуют в такой подписи. Кроме того, честно сгенерированные связующие теги создаются на основе односторонней псевдослучайной функции, и, следовательно, в рамках модели случайного оракула независимо и равномерно распределяются из других элементов доказательства и обязательств по входу.

Доказательство невозможности подделки, описанное в работе [12], основано на (особой) надёжности лежащего в основе сигма-протокола; в случае с нашим изменением оно напрямую применяется к $\mathcal{R}_{link}$, и мы не повторяем его описания в этой работе.

Чтобы продемонстрировать связываемость, сначала отметим, что Link просто сравнивает связующие теги, поэтому подписи будут связаны исключительно в том случае, если у них будет общий связующий тег. Предположим, что злоумышленник может победить в игре связываемости с ничтожной вероятностью для некоторого $k > 1$. Так как все предлагаемые подписи верифицируются, надёжность подразумевает выделение свидетельства x_i из подписи σ_i for all i и свидетельства x из σ. Следует отметить, что все $\{x_i\}$ и x не совпадают. Если $x_i = x_j$ для $i \neq j$, то соответствующие связующие теги J_i и J_j будут такими, что $x_i J_i = x_j J_j = U$; значит, $J_i = J_j$что противоречит Link$(\sigma_i, \sigma_j) = 0$. То же обоснование используется, чтобы продемонстрировать, что x является также отличным. Надёжность

KeyGen(r) :

- Если не указано значение $r \in \mathbb{F}$, оно выбирается равномерно случайным образом.
- Вычисляем $R = rG$.
- Возвращаем $(x, X) = (r, R)$.

Sign(x, M, R) :

- Допустим, $R = \{X_0, \ldots, X_{N-1}\}$ так, что $X_l = x_l G$.
- Вычисляем $J \equiv x_l^{-1} U$.
- Запускаем $\mathcal{P}_{\text{link}}(R, J; (l, x_l)) \to a$ (вплоть до запроса верификатора).
- Задаём $\xi \equiv \mathcal{H}(M, R, a)$.
- Запускаем $\mathcal{P}_{\text{link}}(\xi) \to z$ (после запроса верификатора).
- Возвращаем $\sigma = (a, z, J)$.

Verify(σ, M, R) :

- Допустим, $R = \{X_0, \ldots, X_{N-1}\}$ так, что $X_l - x_l G$.
- Допустим, $\sigma = (a, z, J)$.
- Задаём $\xi \equiv \mathcal{H}(M, R, a)$.
- Возвращаем $\mathcal{V}_{\text{link}}(R, J, a, z)$.

Link(σ, σ') :

- Мы косвенно допускаем, что σ и σ' были верифицированы ранее.
- Допустим, $\sigma = (a, z, J)$ and $\sigma' = (a', z', J')$.
- Если $J = J'$, возвращаем 1. В противном случае возвращаем 0.

Рис. 3: Linkable ring signature using $\mathcal{R}_{\text{link}}$

также подразумевает, что для всех i существует такое $X_i \in R_i$, что $x_i G = X_i$; подобным образом существует такое $X \in R$ что $xG = X$. Согласно допуску имеем

$$\{X_0, \ldots, X_{k-1}, X\} \subset \left(\bigcup_{i=0}^{k-1} R_i\right) \cup R \subset V.$$

Тем не менее следует отметить, что $|\{X_0, \ldots, X_{k-1}, X\}| = k+1$, при $|V| = k$ является противоречием.

Наконец, нами демонстрируется невозможность фабрикации, и мы допускаем, что злоумышленник обладает ничтожным преимуществом с точки зрения нарушения этого свойства.

Поскольку $\mathrm{Verify}(\sigma', M', R') = 1$, надёжность предполагает выделение такого свидетельства $x' \in \mathbb{F}$, что $x'G \in R'$; также у нас есть такое свидетельство x, что $xG \in R$, исходя из известной подписи σ. Так как $\mathrm{Link}(\sigma, \sigma') = 1$, соответствующие связующие теги J и J' будут равны по определению: следовательно, согласно надёжности $xJ = x'J' = U$ получаем $x = x'$. Тем не менее злоумышленник не запрашивал CorruptOracle, используя X, а это означает ничтожное нарушение задачи дискретного логарифмирования. □

6 Протокол: параллельное связываемое обязательство по одному из многих

В данном разделе нами будет описано изменение сигма-протокола для $\mathcal{R}_{\mathrm{link}}$, позволяющее доказать знание множества обязательств в отдельных наборах $d > 1$ с тем же расположением индексов с сохранением свойства связываемости только в первом наборе обязательств. Эта версия Triptych обладает теми же функциональными возможностями, что и схема d-связываемой кольцевой подписи, предложенная в работе [10], даже несмотря на то, что точная модель безопасности будет несколько иной. Затем мы показываем, как применить такую схему к протоколу транзакций с сокрытием подписанта так, чтобы сохранилась возможность демонстрации баланса.

Мы хотим создать сигма-протокол для следующего отношения с некоторым заданным векторным показателем $d > 1$.

$$\mathcal{R}_{\mathrm{par}} = \left\{ \{M_{i,\alpha}\}_{i,\alpha=0}^{N-1,d-1} \subset^d, J \in G; \left(l, \{r_\alpha\}_{\alpha=0}^{d-1}\right) : \{M_{l,\alpha} = r_\alpha G\}_{\alpha=0}^{d-1} \text{ and } U = r_0 J \right\}$$

Это требует внесения минимальных изменений в протокол для $\mathcal{R}_{\mathrm{link}}$, поэтому нами приводятся только изменённые элементы доказательства, структура и верификация которых показаны на рисунке 4. Все остальные элементы доказательства генерируются и верифицируются идентичным образом.

Теорема 3. *Протокол, представленный на рисунке 4 , является совершенно полным, предполагает особое нулевое разглашение для честного верификатора и $(m+1)$ особую надёжность.*

Доказательство. В рамках модели случайного оракула выделение свидетельства в форме $r_0 + \sum_\alpha \mu_\alpha r_\alpha$ подразумевает такое знание всех $\{r_\alpha\}$, что $r_\alpha G = M_{l,\alpha}$, подобно аргументам накопления ключей, описанным в работе [17]. То же выделение показывает, что $\left(r_0 + \sum_\alpha \mu_\alpha r_\alpha\right) J = U + \sum_\alpha \mu_\alpha K_\alpha$, а это в свою очередь подразумевает, что $r_0 J = U$, что и требовалось.

Остальная часть доказательства имеет лишь тривиальные изменения доказательства $\mathcal{R}_{\mathrm{link}}$. □

7 Применение: протокол транзакций с сокрытием подписанта

Параллельная структура, показанная на рисунке 4, может использоваться при $d = 2$ в рамках протокола транзакций с сокрытием подписанта.

Предположим, пользователь желает создать транзакцию, содержащую W ранее сгенерированных выходов, а также сгенерировать T новых выходов. Пользователь перемешивает использованные выходы в большом списке, состоящем из N выходов $\{M_{k,0}\}_{k=0}^{N-1}$ таким образом, что существуют некоторые индексы $\{l_u\}_{u=0}^{W-1}$, где каждое $M_{l_u,0} = r_u$ для некоторого известного приватного ключа r_u. Кроме того, предположим, что каждое $M_{l_u,0}$ имеет обязательство по сумме в форме $M_{l_u,1} \equiv \mathrm{Com}(a_u, s_u)$ для суммы

$\underline{\mathcal{P}_{\mathrm{par}}\left(\{M_{i,\alpha}\}, J; (l, \{r_\alpha\})\right):}$

- Задаём $K_\alpha \equiv r_\alpha J$ для $\alpha \in (0, d)$.
- Задаём $\mu_\alpha \equiv \mathcal{H}\left(\alpha, \{M_{i,\alpha}\}, J, \{K_\alpha\}\right)$ для $\alpha \in (0, d)$.
- Задаём $\{X_j\}_{j=0}^{m-1} \subset$ так, чтобы

$$X_j \equiv \sum_{k=0}^{N-1} p_{k,j} \left(M_{k,0} + \sum_{\alpha=1}^{d-1} \mu_\alpha M_{k,\alpha} \right) + \rho_j G$$

- Задаём $\{Y_j\}_{j=0}^{m-1} \subset$ так, чтобы

$$Y_j \equiv \left(U + \sum_{\alpha=1}^{d-1} \mu_\alpha K_\alpha \right) \sum_{k=0}^{N-1} p_{k,j} + \rho_j G$$

$\underline{\mathcal{P} \to \mathcal{V}:}$
$\{K_\alpha\}, \{X_j\}, \{Y_j\}$

$\underline{\mathcal{V} \to \mathcal{P}:}$
$\xi \in \{0,1\}^*$

$\underline{\mathcal{P}(\xi):}$

- Задаём $z \equiv \left(r_0 + \sum_{\alpha=1}^{d-1} \mu_\alpha r_\alpha \right) \xi^m - \sum_{j=0}^{m-1} \rho_j \xi^j$.

$\underline{\mathcal{P} \to \mathcal{V}:}$
z

$\underline{\mathcal{V}_{\mathrm{par}}(\{M_{i,\alpha}\}, J):}$

- Задаём $\mu_\alpha \equiv \mathcal{H}\left(\alpha, \{M_{i,\alpha}\}, J, \{K_\alpha\}\right)$ для $\alpha \in (0, d)$.
- Принимаем исключительно в том случае, если

$$\begin{aligned} \sum_{k=0}^{N-1} \left(M_{k,0} + \sum_{\alpha=1}^{d-1} \mu_\alpha M_{k,\alpha} \right) \left(\prod_{j=0}^{m-1} f_{j,k_j} \right) - \sum_{j=0}^{m-1} \xi^j X_j - zG &= 0 \\ \left(U + \sum_{\alpha=1}^{d-1} \mu_\alpha K_\alpha \right) \sum_{k=0}^{N-1} \left(\prod_{j=0}^{m-1} f_{j,k_j} \right) - \sum_{j=0}^{m-1} \xi^j Y_j - zJ &= 0 \end{aligned}$$

Рис. 4: Сигма-протокол (сокращённый) для $\mathcal{R}_{\mathrm{par}}$

a_u и маски s_u. (Все остальные $M_{k,0}$ также имеют соответствующее $M_{k,1}$, но структура этих точек в нашем случае не имеет значения.)

Пользователь генерирует W вспомогательных обязательств $P'_u \equiv \mathrm{Com}(a_u, s'_u)$ по тем же суммам, но с другими масками $\{s'_u\}$, равномерно выбираемым случайным образом из $\mathbb{F}$. Затем пользователь генерирует W доказательств траты, каждое из которых содержит следующие входы доказывающей стороны для $u \in [0, W)$:

$$\mathcal{P}_{par}(\{M_{k,0}\}_{k=0}^{N-1}, \{M_{k,1} - P'_u\}_{k=0}^{N-1}, r_u^{-1}U; (l_u, r_u, s_u - s'_u))$$

Для $j \in [0, T)$ пользователь генерирует свежий выход в форме $Q_j \equiv \mathrm{Com}(b_j, t_j)$ для суммы b_j и маску t_j. Маски выбираются таким образом, что для каждого $j \in [1, T)$ имеется t_j , равномерно выбираемая случайным образом из $\mathbb{F}$. Затем выбираем

$$t_0 \equiv \sum_{u=0}^{W-1} s'_u - \sum_{j=1}^{T-1} t_j$$

и включаем все вспомогательные обязательства $\{P'_u\}$ в транзакцию.

Чтобы верифицировать такую транзакцию, верификатор сначала производит верификацию каждого доказательства траты, чтобы гарантировать, что оно является действительным. Затем верификатор убеждается в том, что

$$\sum_{u=0}^{W-1} P'_u - \sum_{j=0}^{T-1} Q_j = 0$$

так, чтобы транзакция была сбалансирована. Процесс успешен, так как сумма обязательств равна нулю исключительно в том случае, если разница входа и выхода равна нулю, что и происходит, поскольку структура обязательства Педерсена является вычислительно связной.

8 Эффективность

Доказательства Triptych масштабируются логарифмически вместе с размером вводимой анонимной группы; это лучшее асимптотическое масштабирование, известное для кольцевых подписей и не требующее доверенных настроек в случае с непарными группами. Среди связанных протоколов, основанных на методе сжатия внутреннего произведения [4], можно назвать [15] и RingCT 3.0 [15]. Тем не менее сложно напрямую сравнивать эффективность этих протоколов. Omniring использует все подписи ко входам транзакций, доказательства диапазона входов и баланс в рамках структуры одиночного доказательства; однако протокол не позволяет проводить более эффективную групповую верификацию доказательств путём объединения общих генераторов. Несмотря на то, что ранняя версия RingCT 3.0 использовала отдельные доказательства по входу, доказательства диапазона и баланса, в самой последней версии все доказательства по входам и балансу смешаны вместе, но доказательства диапазона берутся из стороннего источника, что позволяет построить эффективную структуру, описанную в работе [4]; это делается за счёт требования, согласно которому количество выходов должно быть в два раза больше или же должно быть надлежащее заполнение (что сказывается на времени верификации). Таким образом, в целях сравнения мы немного изменили более раннюю версию RingCT 3.0, подправив свойство надёжности, которое имеется в обновлённой версии, игнорируя размер и издержки, связанные с верификацией элементов, не связанных с доказательствами. Ещё более прямое сравнение касается CLSAG, схемы линейно масштабируемой связываемой кольцевой подписи [10].

Нами приводится сравнение по размерам и верификации параллельного варианта реализации Triptych с $d = 2$ Нами приводится сравнение по размерам и верификации параллельного варианта реализации Triptych с 2-CLSAG. В случае со сравнением масштабирования верификации нами также учитывается использование группового объединения в Triptych и RingCT 3.0, при котором генераторы являются общими для множества доказательств и используются лишь единожды при верификации. Далее, так как верификация в случае с обеими этими схемами сводится к проверке того, равны ли результаты нескольких операций мультискалярного умножения нулю, мы можем использовать выборочное взвешивание, чтобы верификация группы, включающей в себя множество доказательств,

свелось к одному мультискалярному умножению. Использование эффективных алгоритмов мультискалярного умножения подобных тем, что описаны в работах [23, 20], означает, что n-мультискалярное умножение будет выражено как $O(n/\log n)$. акая групповая форма не применима к CLSAG, где процесс верификации представляет собой последовательность вычисления хеш-функций.

В Таблице 1 приводится сравнение размеров доказательства/подписи и сложности верификации этих схем в виде функции размера анонимной группы N и общего размера B. Сложность верификации разбита на ряд операций хеширования к $\mathbb{F}$ (обозначены как $\mathcal{H}$), операций хеширования к $\mathbb{G}$ (обозначены как $\mathbb{H}$) и операций i-мультискалярного умножения с размером $k(i)$. На рисунке 5размер показан как функция размера анонимной группы входа с тем допуском, что элементы $\mathbb{G}$ и $\mathbb{F}$ занимают 32 Кбайта. Следует отметить, что, несмотря на то, что доказательства RingCT 3.0 немного меньше при большом N, доказательства Triptych меньше при $N < 512$, ограниченный, но разумный диапазон для практичного времени верификации. Также следует отметить, что в данной работе мы не сравниваем напрямую применение Triptych в рамках полного протокола транзакций с RingCT 3.0, так как использование вспомогательных данных, не относящихся к протоколу/подписи, в таком протоколе может отличаться в зависимости от варианта реализации.

	Размер ($\mathbb{G}$)	Размер ($\mathbb{F}$)	Верификация (общая групповая)
CLSAG [10]	2	$N+1$	$B(N+2)\mathcal{H} + BN\mathbb{H} + 2BNk(3)$
RingCT 3.0 [24]	$2\lg(N)+9$	9	$k(B[2N+2\lg(N)+9]+2N+5)$
Triptych (данная работа)	$2\lg(N)+6$	$\lg(N)+3$	$k(B[2N+2\lg(N)+2]+2\lg(N)+3)$

Таблица 1: Размеры доказательств и сложность верификации при размере анонимной группы N и общем (групповом) размере B

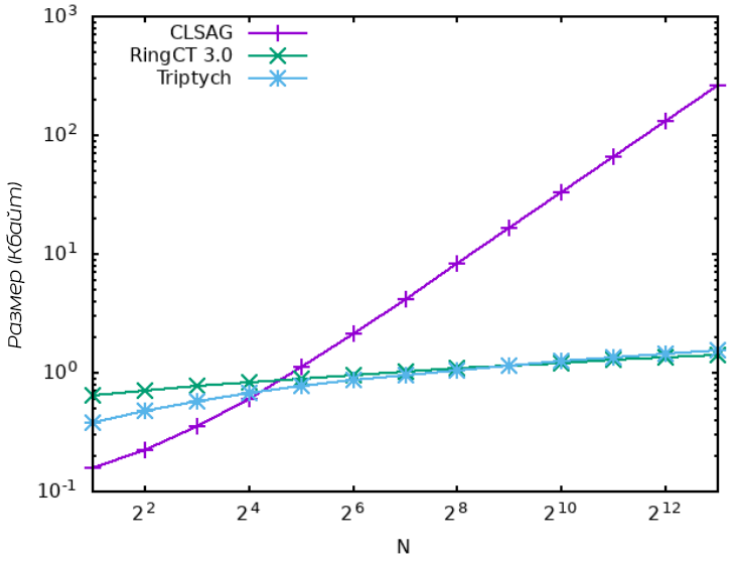

Рис. 5: Размеры доказательств для анонимной группы входа с размером N

9 Будущая работа

Существует возможность дальнейшего расширения системы доказательства Triptych таким образом, чтобы она поддерживала доказательство знания открытий множества обязательств в пределах *одной и той же* анонимной группы и позволяла создавать связующие теги для каждого открытия и демонстрировать баланс напрямую в пределах одного доказательства. Такая структура, будучи интегрированной в протокол транзакций, будет гораздо эффективнее, чем та, что представлена в настоящей работе, но определения безопасности, применимые к такой структуре, пока ещё проходят оценку.

Список литературы

[1] Michael Backes, Nico Döttling, Lucjan Hanzlik, Kamil Kluczniak, and Jonas Schneider. Ring signatures: Logarithmic-size, no setup—from standard assumptions (Кольцевые подписи: логарифмическое масштабирование без настроек при стандартных допусках). In Yuval Ishai and Vincent Rijmen, editors, *Advances in Cryptology – EUROCRYPT 2019*, pages 281–311, Cham, 2019. Springer International Publishing.

[2] Adam Bender, Jonathan Katz, and Ruggero Morselli. Ring signatures: Stronger definitions, and constructions without random oracles (Кольцевые подписи: усиленные определения и структуры без применения случайных оракулов). In Shai Halevi and Tal Rabin, editors, *Theory of Cryptography*, pages 60–79, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

[3] Jonathan Bootle, Andrea Cerulli, Pyrros Chaidos, Essam Ghadafi, Jens Groth, and Christophe Petit. Short accountable ring signatures based on DDH (Короткие учитываемые кольцевые подписи на базе DDH). In Günther Pernul, Peter Y A Ryan, and Edgar Weippl, editors, *Computer Security – ESORICS 2015*, pages 243–265, Cham, 2015. Springer International Publishing.

[4] Benedikt Bünz, Jonathan Bootle, Dan Boneh, Andrew Poelstra, Pieter Wuille, and Greg Maxwell. Bulletproofs: Short proofs for confidential transactions and more (Bulletproofs: короткие доказательства конфиденциальных транзакций и многое другое). In *2018 IEEE Symposium on Security and Privacy (SP)*, pages 315–334. IEEE, 2018.

[5] Ronald Cramer, Ivan Damgård, and Berry Schoenmakers. Proofs of partial knowledge and simplified design of witness hiding protocols (Доказательства частичного знания и упрощённая схема протоколов сокрытия свидетельства). In Yvo G. Desmedt, editor, *Advances in Cryptology — CRYPTO '94*, pages 174–187, Berlin, Heidelberg, 1994. Springer Berlin Heidelberg.

[6] Yevgeniy Dodis and Aleksandr Yampolskiy. A verifiable random function with short proofs and keys (Верифицируемая случайная функция с короткими доказательствами и ключами). In Serge Vaudenay, editor, *Public Key Cryptography - PKC 2005*, pages 416–431, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg.

[7] Amos Fiat and Adi Shamir. How to prove yourself: Practical solutions to identification and signature problems (Как самостоятельно произвести доказательство — практические решения проблем идентификации и подписи). In Andrew M. Odlyzko, editor, *Advances in Cryptology — CRYPTO' 86*, pages 186–194, Berlin, Heidelberg, 1987. Springer Berlin Heidelberg.

[8] Eiichiro Fujisaki. Sub-linear size traceable ring signatures without random oracles (Сублинейные отслеживаемые по размеру кольцевые подписи, не использующие случайного оракула). In Aggelos Kiayias, editor, *Topics in Cryptology – CT-RSA 2011*, pages 393–415, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.

[9] Eiichiro Fujisaki and Koutarou Suzuki. Traceable ring signature (Отслеживаемая кольцевая подпись). In Tatsuaki Okamoto and Xiaoyun Wang, editors, *Public Key Cryptography – PKC 2007*, pages 181–200, Berlin, Heidelberg, 2007. Springer Berlin Heidelberg.

[10] Brandon Goodell, Sarang Noether, and Arthur Blue. Compact linkable ring signatures and applications (Компактные кольцевые подписи и их применение). Cryptology ePrint Archive, Report 2019/654, 2019. `https://eprint.iacr.org/2019/654`.

[11] Jens Groth. On the size of pairing-based non-interactive arguments (По вопросу размера неинтерактивных аргументов, основанных на попарном объединении). In Marc Fischlin and Jean-Sébastien Coron, editors, *Advances in Cryptology – EUROCRYPT 2016*, pages 305–326, Berlin, Heidelberg, 2016. Springer Berlin Heidelberg.

[12] Jens Groth and Markulf Kohlweiss. One-out-of-many proofs: Or how to leak a secret and spend a coin (Доказательства по одному из многих или как выведать секрет и потратить монету). In Elisabeth

Oswald and Marc Fischlin, editors, *Advances in Cryptology - EUROCRYPT 2015*, pages 253–280, Berlin, Heidelberg, 2015. Springer Berlin Heidelberg.

[13] Daira Hopwood, Sean Bowe, Taylor Hornby, and Nathan Wilcox. Zcash protocol specification (Спецификация протокола ZCash). *Tech. rep. 2016–1.10. Zerocoin Electric Coin Company, Tech. Rep.*, 2016.

[14] Aram Jivanyan. Lelantus: Towards confidentiality and anonymity of blockchain transactions from standard assumptions (Lelantus: движение в направлении конфиденциальности и анонимности блокчейн транзакций с учётом стандартных допусков). Cryptology ePrint Archive, Report 2019/373, 2019. `https://eprint.iacr.org/2019/373`.

[15] Russell W. F. Lai, Viktoria Ronge, Tim Ruffing, Dominique Schröder, Sri Aravinda Krishnan Thyagarajan, and Jiafan Wang. Omniring: Scaling up private payments without trusted setup - formal foundations and constructions of ring confidential transactions with log-size proofs (Omniring: масштабирование анонимных платежей без доверенных настроек — формальные основы и схемы конфиденциальных транзакций с логарифмически масштабируемыми доказательствами). Cryptology ePrint Archive, Report 2019/580, 2019. `https://eprint.iacr.org/2019/580`.

[16] Joseph K. Liu, Victor K. Wei, and Duncan S. Wong. Linkable spontaneous anonymous group signature for ad hoc groups (Связываемая подпись спонтанной анонимной группы для специализированных групп). In Huaxiong Wang, Josef Pieprzyk, and Vijay Varadharajan, editors, *Information Security and Privacy*, pages 325–335, Berlin, Heidelberg, 2004. Springer Berlin Heidelberg.

[17] Gregory Maxwell, Andrew Poelstra, Yannick Seurin, and Pieter Wuille. Simple Schnorr multi-signatures with applications to Bitcoin (Использование простых мультиподписей Шнорра в Bitcoin). *Designs, Codes and Cryptography*, 87(9):2139–2164, Sep 2019.

[18] Ian Miers, Christina Garman, Matthew Green, and Aviel D Rubin. Zerocoin: Anonymous distributed e-cash from bitcoin (Zerocoin: анонимная распределённая электронная валюта на базе Bitcoin). In *2013 IEEE Symposium on Security and Privacy*, pages 397–411. IEEE, 2013.

[19] Shen Noether, Adam Mackenzie, and the Monero Research Lab. Ring confidential transactions (Кольцевые конфиденциальные транзакции). *Ledger*, 1(0):1–18, 2016.

[20] Nicholas Pippenger. On the evaluation of powers and monomials (По вопросу оценки показателей степени и одночленов). *SIAM Journal on Computing*, 9(2):230–250, 1980.

[21] Ronald L. Rivest, Adi Shamir, and Yael Tauman. How to leak a secret (Как узнать секрет). In Colin Boyd, editor, *Advances in Cryptology — ASIACRYPT 2001*, pages 552–565, Berlin, Heidelberg, 2001. Springer Berlin Heidelberg.

[22] C. P. Schnorr. Efficient signature generation by smart cards (Создание эффективных подписей при помощи смарт-карт). *Journal of Cryptology*, 4(3):161–174, Jan 1991.

[23] Ernst G Straus. Addition chains of vectors (problem 5125) (Дополнительные цепочки векторов (задача 5125)). *American Mathematical Monthly*, 70(806-808):16, 1964.

[24] Tsz Hon Yuen, Shi-feng Sun, Joseph K. Liu, Man Ho Au, Muhammed F. Esgin, Qingzhao Zhang, and Dawu Gu. RingCT 3.0 for blockchain confidential transaction: Shorter size and stronger security (RingCT 3.0 для проведения конфиденциальных транзакций в блокчейне: меньший размер при более высоком уровне безопасности). Cryptology ePrint Archive, Report 2019/508, 2019. `https://eprint.iacr.org/2019/508`.