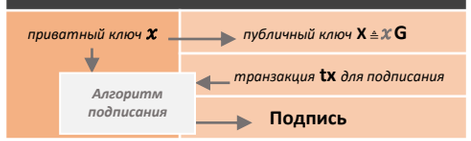
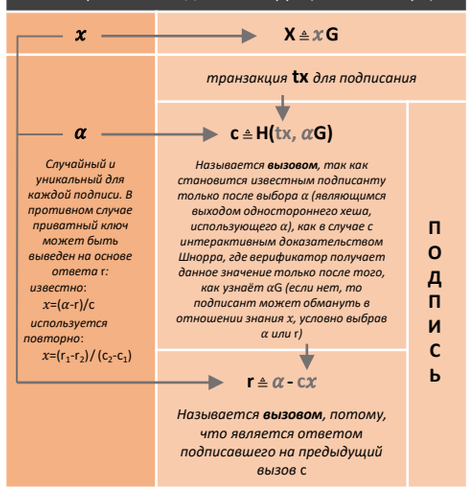
Унаследованная базовая подпись с ключами EC



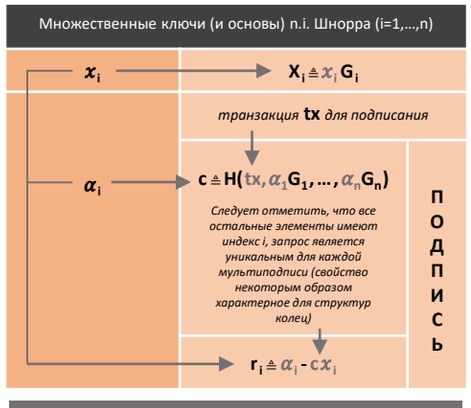
Алгоритм проверки $f(X, tx, Signature)$ → успешно / ? / отказ

Неитеративная подпись Шнорра (Фиат-Шамира)



$f(X, tx, c, r)$ → $H(tx, rG + cX) \overset{?}{=} c$ (rG + cX = αG если подпись в порядке)

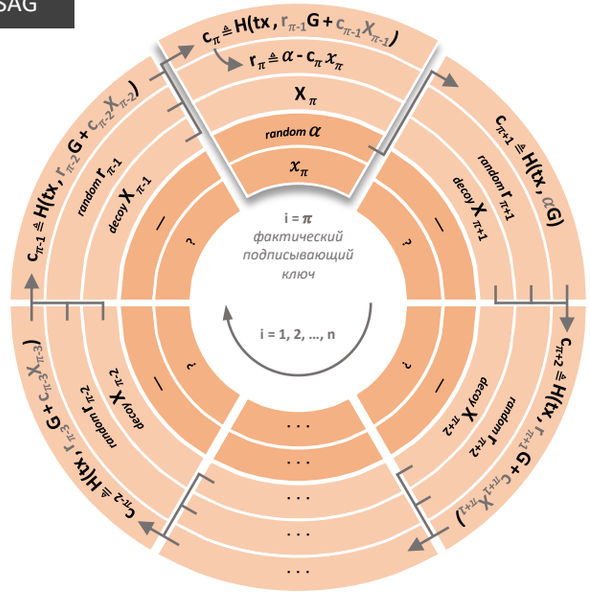
Множественные ключи (и основы) n.i. Шнорра (i=1,…,n)



$f(X_i, tx, c, r_i)$ → $H(tx, r_1G_1 + cX_1, \dots, r_nG_n + cX_n) \overset{?}{=} c$ (обязательство, создаваемое одновременно n подписей)

«Волшебство» колец состоит в нахождении особенностей предшествующих схем с ложными выходами с сохранением всего одного ФАКТИЧЕСКОГО подписанта (с технической точки зрения необходимо наличие множества X_i в алгоритме верификации, но единственного x в алгоритме подписания); и всё должно происходить без согласования действий между владельцами ключей.

SAG



$c_2 = H(tx, r_1G + c_1X_1)$ $c_3 = H(tx, r_2G + c_2X_2)$ … $c_n = H(tx, r_{n-1}G + c_{n-1}X_{n-1})$ $H(tx, r_nG + c_nX_n) \overset{?}{=} c_1$

$f(X_i, tx, c_1, r_i)$

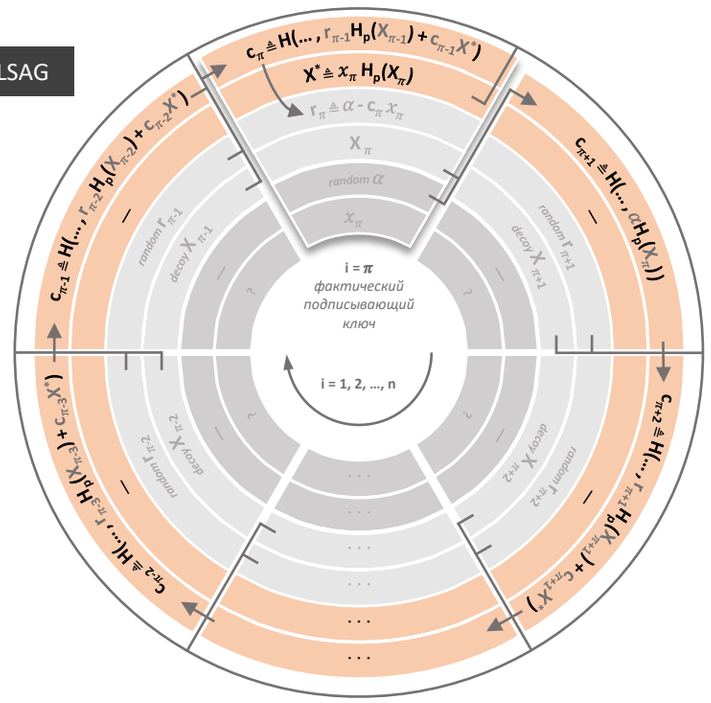
MLSAG

$c_{\pi+1} \triangleq H(tx, \alpha_m G, \dots, \alpha_m H_p(X_{\pi,m}), \dots, \alpha_1 G, \dots, \alpha_1 H_p(X_{\pi,1}))$

$c_i \triangleq H(tx, r_{i-1,m}G + c_{i-1}X_{i-1,m}, \dots, r_{i-1,m}H_p(X_{i-1,m}) + c_{i-1}X_m^*, \dots, r_{i-1,1}G + c_{i-1}X_{i-1,1}, \dots, r_{i-1,1}H_p(X_{i-1,1}) + c_{i-1}X_1^*)$

$f(X_{i,j}, tx, c_1, r_{i,j}, X_j^*)$ → X_j^* никогда ранее не было в блокчейне? / $l X_j^* \overset{?}{=} 0$ / $H(tx, r_{n,1}G + c_n X_{n,1}, r_{n,1}H_p(X_{n,1}) + c_n X_1^*, \dots, r_{n,m}G + c_n X_{n,m}, r_{n,m}H_p(X_{n,m}) + c_n X_m^*) \overset{?}{=} c_1$ ($c_n = c_n(c_{n-1}(\dots(c_2(c_1))))$ как ранее было в *SAG u bLSAG*)

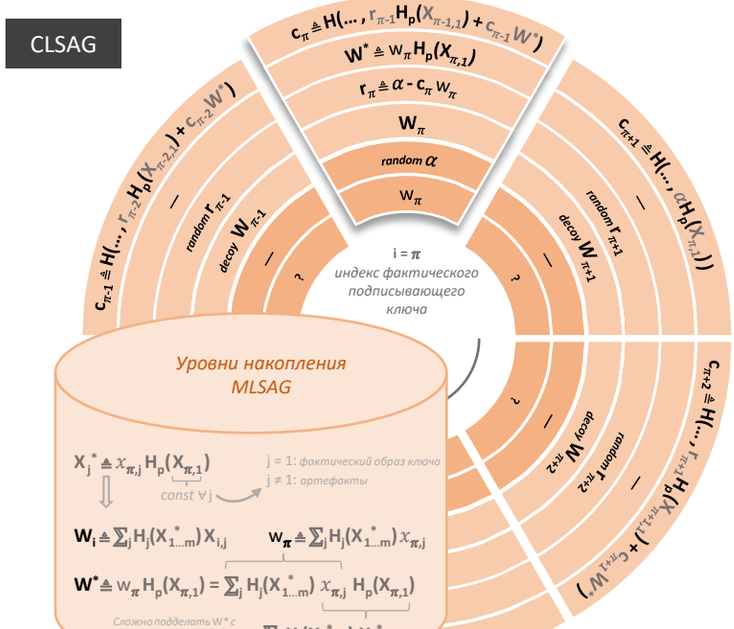
bLSAG



$c_{\pi+1} \triangleq H(tx, \alpha G, \alpha H_p(X_\pi))$ $c_i \triangleq H(tx, r_{i-1}G + c_{i-1}X_{i-1}, r_{i-1}H_p(X_{i-1}) + c_{i-1}X^*)$

$f(X_i, tx, c_1, r_i, X^*)$ → X^* никогда ранее не было в блокчейне? / $l X^* \overset{?}{=} 0$ / $H(tx, r_nG + c_n(c_{n-1}(\dots(c_2(c_1)))) X_n, r_nH_p(X_n) + c_n(c_{n-1}(\dots(c_2(c_1)))) X^*) \overset{?}{=} c_1$ (*как ранее было в SAG*)

CLSAG



$c_{\pi+1} \triangleq H(tx, \alpha G, \alpha H_p(X_{\pi,1}))$ $c_i \triangleq H(tx, r_{i-1}G + c_{i-1}W_{i-1}, r_{i-1}H_p(X_{i-1,1}) + c_{i-1}W^*)$

$f(X_{i,j}, tx, c_1, r_i, X_j^*)$ → X_1^* никогда ранее не было в блокчейне? / $l X_1^* \overset{?}{=} 0$ / $H(tx, r_nG + c_n W_n(X_j^*, X_{n,j}), r_nH_p(X_{n,1}) + c_n W^*(X_j^*)) \overset{?}{=} c_1$ (*снова:* $c_n = c_n(c_{n-1}(\dots(c_2(c_1))))$)

Вольные примечания по схемам колец

SAG (подпись спонтанной анонимной группы)

- значение индекса фактического подписанта (π) является случайным. В противном случае X_π можно было бы вывести на основе порядка параметров, содержащихся в подписи;
- запросы c_i строятся на основе элементов предыдущего уровня, а зависимости обозначены стрелками;
- окончательное определение r_π гарантирует, что зависимости, применяемые ко всем остальным c_i , по-прежнему будут применяться и к $c_{\pi+1}$ (даже если изначально будут вычислены на основе α), и запросы таким образом образуют замкнутую цепочку, кольцо: вот почему в подписи достаточно представить c_1 (это такой «сохранённый» одиночный запрос на свойство мультиподписи)

bLSAG (связываемая подпись SAG Бэка)

- bLSAG является схемой SAG, расширенной за счёт образа ключа X^* (чтобы исключить возможность двойной траты при сохранении анонимности, что обеспечивает при этом связываемость подписей), а также изменённых запросов c_i , позволяющих создать обязательство для этих образов ключей;
- $H_p(X_\pi)$ является тщательно выбираемой функцией, выдающей случайную точку в подгруппе базовых точек порядка, равного целому числу l на эллиптической кривой, и действующей в качестве генератора точки для образа ключа $X^* \triangleq x_\pi H_p(X_\pi)$

Несколько ПЛОХИХ генераторов образов ключей

$H_p(X_\pi) \triangleq n(X_\pi)G$
$\Rightarrow X^* \triangleq x_\pi n(X_\pi)G = n(X_\pi)x_\pi G = n(X_\pi)X_\pi$
позволяет найти фактического подписанта после ряда попыток

$H_p(X_\pi) \triangleq G_2$
$\Rightarrow X_1^* \triangleq x_{\pi,1} G_2 \qquad X_2^* \triangleq x_{\pi,2} G_2$
$\Rightarrow X_1^* - X_2^* = (x_{\pi,1} - x_{\pi,2})G_2$
предыдущий отправитель , переводящий средства $X_{\pi,1}$ и $X_{\pi,2}$, может вычислить значения в скобках (благодаря обмену в рамках алгоритма, подобного алгоритму Диффи-Хеллмана, на основе скрытых адресов), а следовательно, может воспользоваться эвристическим подходом, чтобы объединить будущие случаи использования $X_{\pi,1}$ и $X_{\pi,2}$

$H_p(X_\pi) \triangleq X_\pi \triangleq x_\pi G$
$\Rightarrow X_1^* - X_2^* = (x_{\pi,1} - x_{\pi,2})G$
(как и в предыдущем случае, но с добавлением некоторых алгебраических выражений, а также необходимостью в использовании G, чтобы избавиться от остающегося приватного ключа траты в пользу публичного)

- Проверка $l X^* = 0$ в рамках алгоритма верификации необходима, чтобы избежать двойной траты из-за «гибкости» образа ключа. В запросах мы имеем следующее:
  $c_i = H(\dots c_{i-1} X^*)$
  тем не менее X^* можно заменить на ложное $X^* + P_h$, где P_h является точкой в подгруппе порядка h на эллиптической кривой, кофактором, который, если злоумышленник (после ряда попыток) найдёт все c_i , умножается на h, и в этом случае:
  $c_i(X^* + P_h) = c_i X^* + c_i P_h = c_i X^*$
  поскольку любая точка, умноженная на порядок своей подгруппы, даёт ноль. К счастью, $l(X^* + P_h) \neq 0$, так как будучи простым числом l не может быть умножена на h

MLSAG (многоуровневая связываемая подпись SAG)

- MLSAG представляет собой стек из множества bLSAG с имеющимися на каждом уровне запросами ci (поэтому один отдельный вызов на каждом 3D уровне является обязательством на всех уровнях);
- даже несмотря на то, что это не является требованием схемы, в случае с Monero значение индекса фактического подписанта (π) должно быть случайным, но использоваться при этом на всех уровнях, что обеспечит возможность кластеризации между уровнями для злоумышленника, что позволит составить обоснованное предположение о фактических ключах: вот почему в случае с транзакциями со множеством входов (где можно достичь максимальной экономии) лучше избегать использования всего одной подписи MLSAG

CLSAG (сжатая связываемая подпись SAG)

- Данная схема в настоящее время используется Monero. Это bLSAG с «псевдо ключами» w_π и W_i, полученными путём накопления ключей на различных уровнях MLSAG. Она обеспечивает обратную связываемость (что подразумевает создание обычного образа ключа) только для $X_{\pi,1}$;
- W^* сама по себе не исключает возможности двойной траты , но строится на основе фактических артефактов X_1^* и $X_{\pi,1}^*$ (вот почему они используются в алгоритме верификации).

Примечания

В основе данной шпаргалки лежат материалы, взятые из работы От нуля к Монеро: Второе издание (особенно главы 2) и упомянутые источники: система обозначений отличается совсем немного и имеет лишь «минимальные» исключения, чтобы сфокусировать внимание на последовательном представлении ключевых свойств колец (например, отсутствуют префиксы ключей и доменное разделение хешей).

Перевод данной шпаргалки выполнил v1docq47