2009 Математические методы криптографии №4(6)

DOI 10.17223/20710410/6/6

УДК 519.725, 519.816, 519.712.6

ПРОТОКОЛ АРГУМЕНТА ЗНАНИЯ СЛОВА КОДА ГОППЫ И ОШИБКИ ОГРАНИЧЕННОГО ВЕСА[1]

В. Е. Федюкович

Интропро, г. Киев, Украина

E-mail: vf@unity.net

Предложен новый протокол, позволяющий показать знание кодового слова кода Гоппы и полинома Гоппы, таких, что хэмминг-вес ошибки в искажённом кодовом слове не превышает заданный порог. Протокол является аргументом в предположении о сложности задачи поиска логарифма в используемой конечной группе и имеет специальное свойство нулевого разглашения в модели с честным Проверяющим.

Ключевые слова: *интерактивная система доказательства, аргумент, нулевое разглашение, схема привязки, код Гоппы.*

Введение

Рассматривается задача проверки утверждения об ошибке в искажённом кодовом слове кода Гоппы. Дополнительным условием является предоставление проверяющей стороне минимально необходимой информации о структуре кода, кодовом слове и весе ошибки. Рассматриваются интерактивные протоколы аргумента и протоколы аргумента знания значений, из которых получены экземпляры привязки. Выполняется вероятностная проверка утверждения о кодовом слове и весе ошибки. Произвольный Доказывающий, который не знает компонентов кодового слова и коэффициентов полинома Гоппы, удовлетворяющих проверяемым условиям, имеет только ничтожную вероятность успешно завершить протокол с честным Проверяющим. Рассматривается схема запрос-ответ, в которой запросом является значение переменной некоторого полинома над конечным полем, а проверяемым условием — равенство нулю этого полинома. Также рассматриваются полиномы, в которых компоненты кодового слова и коэффициенты полинома Гоппы заменяются ответами Проверяющего.

Полученный протокол имеет свойство полноты, свойство корректности в предположении о сложности задачи поиска логарифма в конечной группе, является аргументом знания кодового слова и полинома Гоппы, имеет специальное свойство нулевого разглашения в модели с честным Проверяющим. Ошибка корректности протокола экспоненциально мала, параметром безопасности является количество бит в двоичном представлении порядка группы.

Тезисное изложение результатов этой работы приведено в [1].

1. Обозначения и определения

1.1. Интерактивные системы и их свойства

Участника протокола называют *честным* (honest), если он следует спецификации протокола, либо *произвольным* (any) иначе. Функцию называют *ничтожной*, если она убывает быстрее любой степени своего аргумента. Будем говорить, что событие

[1] Результаты работы докладывались на Международной конференции с элементами научной школы для молодёжи, г. Омск, 7–12 сентября 2009 г.

происходит *почти всегда* (overwhelming probability), если вероятность того, что событие не произойдёт, ничтожна. Будем называть *преимуществом* разность вероятностей успешного завершения некоторого алгоритма распознавания, обычно выполяемого соперником, в случае предоставления сопернику возможности выбора входных данных и для случайно выбранных входных данных. *Алфавитом* называют конечное множество символов (букв алфавита); *языком* $\mathbb{L}$ называют некоторое подмножество множества всех слов в данном алфавите. Задачу принятия решения $x \stackrel{?}{\in} \mathbb{L}$ называют *распознаванием языка*. Задачу распознавания языка относят к классу сложности NP, если для любого слова x существует *NP-свидетельство* (witness) w, знание которого позволяет решить задачу за полиномиальное время; в этом случае язык называется *NP-языком*.

Определение 1. *Интерактивной системой аргумента* (interactive argument system) для NP-языка $\mathbb{L}$ и для некоторого предположения (assumption) будем называть интерактивную пару машин Тьюринга, работающих полиномиальное время, со словом x на общей входной ленте и свидетельством w на входной ленте Доказывающего, такую, что машина Проверяющего всегда выдает бинарный ответ (решение) и останавливается, а также имеются свойства полноты и корректности. *Полнота* (completeness): честный Проверяющий всегда принимает положительное решение (accept) для честного Доказывающего и $x \in \mathbb{L}$ (нулевая *ошибка полноты*). *Корректность* (soundness): честный Проверяющий принимает положительное решение для произвольного Доказывающего и $x \notin \mathbb{L}$ с ничтожной вероятностью при условии, что Доказывающий не может нарушить используемое предположение (ничтожная *ошибка корректности*).

Заметим, что определение интерактивной системы доказательства в [2] предусматривает неограниченные ресурсы машины Доказывающего, ненулевую ошибку полноты и ошибку корректности, превышающую ошибку полноты на не-ничтожную функцию; параметром является $|x|$. В [3] рассматривались протоколы, свойство корректности которых выполняется при дополнительных предположениях о вычислительной сложности некоторых задач (computationally-sound proofs); для них был предложен термин «*протокол аргумента*». Протокол аргумента обычно имеет параметр, определяющий степень сложности задачи, содержащейся в используемом предположении.

Определение 2. Будем говорить, что протокол аргумента имеет свойство *знания* (is of knowledge), если существует алгоритм *extractor*, предполагающий управление машиной Доказывающего с возможностью возврата (rewind), работающий полиномиальное время, получающий на входе стенограмму протокола и почти всегда выдающий свидетельство Доказывающего при условии выполнимости предположения (ничтожная *ошибка знания*).

Введённое так понятие протокола аргумента со свойством знания отличается от понятия протокола доказательства знания [4] тем, что последний предусматривает неограниченные ресурсы Доказывающего и не-ничтожную ошибку корректности. Рассматривались [5] также протоколы с алгоритмом *extractor*, которые либо дают свидетельство, либо нарушают используемое предположение.

Определение 3 [2, 6, 7]. Говорят, что протокол имеет свойство *нулевого разглашения* (is zero knowledge), если существует *моделирующий алгоритм* (simulator), работающий полиномиальное время и в случае $x \in \mathbb{L}$ выдающий *моделируемую стенограмму* (simulated transcript), неотличимую от стенограммы протокола с Доказывающим. Протокол имеет свойство *нулевого разглашения в модели с честным Проверяющим* (is honest Verifier zero knowledge), если моделируемая стенограмма неотличима от всех

стенограмм протокола, в которых Проверяющий следует спецификации протокола. Протокол имеет свойство *специального нулевого разглашения* (is special honest Verifier zero knowledge), если моделирующий алгоритм получает на входе запросы Проверяющего и дает моделируемую стенограмму, неотличимую от всех стенограмм протокола, в которых запросы Проверяющего совпадают с запросами на входе моделирующего алгоритма.

Близкими к понятию нулевого разглашения, но не совпадающими с ним, являются следующие встречающиеся в литературе определения. Протоколы, дающие только ничтожное преимущество любому алгоритму Соперника (Adversary) найти свидетельство Доказывающего, называют *witness hiding*. Протоколы, дающие только ничтожное преимущество произвольному алгоритму Соперника, определяющему, какое из двух возможных значений является свидетельством, называют *witness indistinguishable*.

Способ Фиата — Шамира преобразования протокола в схему подписи [8] предусматривает произвольный алгоритм Проверяющего, выбирающий значение хэш-функции входного слова и имеющейся части стенограммы в качестве запроса, и в остальном следующий спецификации протокола; экземпляром подписи при этом является стенограмма протокола. Известен русскоязычный обзор протоколов и некоторых их свойств [9].

1.2. Схема привязки

Определение 4. *Схемой привязки* (commitment scheme) называют тройку алгоритмов $(Generate(), Commit(w), Open(C, w))$, такую, что экземпляр привязки C всегда успешно *раскрывается* значением w, из которого он был *создан*, а также имеются свойства связывания и скрытия. *Связывание* (binding): задача поиска альтернативного значения w', такого, что $Commit(w') = Commit(w)$, является сложной. *Скрытие* (hiding): экземпляр привязки дает только ничтожное преимущество любому алгоритму Соперника, определяющему, какое из двух значений использовалось для создания экземпляра привязки.

Определение 5. Схема привязки имеет свойство *гомоморфизма* (homomorphic) относительно некоторых операций $(\oplus, \otimes)$, если

$$\forall(x_1, x_2) \quad Commit(x_1 \oplus x_2) = Commit(x_1) \otimes Commit(x_2).$$

Например, алгоритм инициализации схемы привязки Педерсена [10] к элементу x конечного поля $\mathbb{Z}_q$ предусматривает выбор циклической конечной группы $\mathbb{G}$ порядка q, а также двух элементов группы h, f так, что $\log_f(h)$ неизвестен и задача поиска логарифма в группе является сложной. Алгоритм создания экземпляра привязки выбирает случайное значение $r \in_R \mathbb{Z}_q$, создаёт элемент группы $C = h^x f^r$. Алгоритм раскрывания проверяет $C \stackrel{?}{=} h^x f^r$. Схема Педерсена имеет свойство аддитивного гомоморфизма и является протоколом доказательства знания значений, из которых был получен экземпляр привязки.

1.3. Протокол Шнорра

Ряд протоколов предусматривают бинарные запросы Проверяющего, так что вероятность для произвольного Доказывающего случайно предложить правильный ответ равна 1/2. Известны также протоколы с запросами, выбранными из множества S с большим количеством элементов, так что вероятность случайно предложить правильный ответ равна $1/|S|$. В протоколе Шнорра [11] запросы выбираются из конечного поля $\mathbb{Z}_q$, так что ошибка корректности ничтожна (параметром является длина битового представления q). Общей информацией сторон является генератор g циклической конечной группы $\mathbb{G}$ порядка q для некоторого большого простого q, а также

некоторый элемент $y \in \mathbb{G}$. Неформально, y играет роль публичного ключа проверки. Информацией Доказывающего является число b, такое, что $y = g^b$.

1) Доказывающий выбирает некоторое случайное $\alpha \in \mathbb{Z}_q$, получает элемент группы $A_0 = g^\alpha$ и пересылает A_0 Проверяющему.
2) Проверяющий выбирает некоторый запрос (challenge) $c \in \mathbb{Z}_q$ случайным образом и пересылает его Доказывающему.
3) Доказывающий вычисляет и пересылает Проверяющему ответ
$$B = cb + \alpha \pmod q. \tag{1}$$
4) Проверяющий принимает положительное решение, если
$$g^B y^{-c} A_0^{-1} = 1. \tag{2}$$

Протокол доказательства знания логарифма в конечной группе с бинарными запросами и ответами Доказывающего, полученными как значения линейного полинома, был предложен в работе Chaum, Evertse и van de Graaf [12].

1.4. Код Гоппы

Определение 6. *Кодовым словом* [13, 14] будем называть вектор $b = b_1 \dots b_N$ элементов поля, таких, что для некоторого полинома $g(z)$ и для дополнительных значений a_j, $j = 1, \dots, N$, выполняются условия $g(a_j) \neq 0$ и
$$\sum_{j=1}^{N} \frac{b_j}{z - a_j} \equiv 0 \pmod{g(z)}. \tag{3}$$

Множество всех таких кодовых слов называют *кодом Гоппы*, а $g(z)$ — *полиномом Гоппы*.

Двоичный код такого вида был предложен в работе [13], в [14] рассматривается код над полем, состоящим из $q = p^l$ элементов для некоторого простого p. В этой работе используется стандартная схемы привязки, в связи с чем рассматривается только случай $l = 1$.

Определение 7. *Ошибкой* в искажённом входном слове $w = w_1 \dots w_N$ будем называть вектор $w - b$, где b — кодовое слово; *весом ошибки* — количество ненулевых компонент в ней.

Пусть $g(z) = \sum\limits_{k=0}^{T} z^k g_k$. В этой работе проверяется справедливость утверждения, эквивалентного (3):
$$W^*(z) \equiv 0, \qquad W^*(z) = \sum_{j=1}^{N} b_j \sum_{k=1}^{T} g_k \frac{z^k - a_j^k}{z - a_j} \prod_{\substack{i=1, m=0 \\ i \neq j}}^{N} \sum_{m=0}^{T} g_m a_i^m. \tag{4}$$

Значения b_j, $j = 1, \dots, N$, и g_k, $k = 1, \dots, T$, (свидетельство Доказывающего) доступны Проверяющему только в виде экземпляров привязки и ответов (1) Доказывающего.

Определение 8. *Проверочным полиномом кодового слова* для некоторого значения аргумента $z = d$ и для некоторых начальных случайных значений β_1, для $j = 1, \dots, N$ и α_k для $k = 1, \dots, T$ протокола Шнорра будем называть
$$R^*(y) = \sum_{j=1}^{N} B_j(y) \sum_{k=1}^{T} G_k(y) \frac{d^k - a_j^k}{d - a_j} \prod_{\substack{i=1, m=0 \\ i \neq j}}^{N} \sum_{m=0}^{T} G_m(y) a_i^m,$$
$$B_j(y) = y b_j + \beta_j, \qquad G_k(y) = y g_k + \alpha_k.$$

Определение 9. *Проверочным полиномом ошибки* для некоторых начальных случайных значений β_j, $j = 1, \ldots, N$, протокола Шнорра будем называть

$$E^*(y) = \prod_{j=1}^{N} (B_j(y) - yw_j), \qquad B_j(y) = yb_j + \beta_j.$$

Нетрудно убедиться, что $\deg(R^*(y)) = N + 1$, старший коэффициент $R^*(y)$ равен $W^*(d)$, а степень $E^*(y)$ равна весу ошибки.

Алгебраическое декодирование искажённого кодового слова в этой работе не рассматривается.

1.5. Вероятностная проверка утверждений о полиномах

Пусть $f(z)$ — некоторый ненулевой полином с коэффициентами из конечного поля порядка q. Количество корней такого полинома не превышает $\deg(f(z))$. Вероятность выбрать корень случайно путём выбора произвольного значения аргумента из поля при условии отсутствия информации о коэффициентах не превышает $\deg(f(z))/q$.

2. Протокол для слова кода Гоппы и ошибки ограниченного веса

Протокол P

Общая информация Проверяющего и Доказывающего: порядок q и пара элементов группы (h, f), дополнительные значения a_j, $j = 1, \ldots, N$, искажённое кодовое слово $w_1 \ldots w_N$, экземпляры привязки к коэффициентам полинома V_k, $k = 0, \ldots, T$, и к компонентам кодового слова W_j, $j = 1, \ldots, N$, верхняя граница веса ошибки S.

Информация Доказывающего: коэффициенты полинома g_k и дополнительные значения θ_k, компоненты кодового слова b_j и дополнительные значения φ_j, такие, что $V_k = h^{g_k} f^{\theta_k}$, $W_j = h^{b_j} f^{\varphi_j}$ и $g(z) = \sum\limits_{k=0}^{T} z^k g_k$, $\sum\limits_{j=1}^{N} \frac{b_j}{z - a_j} \equiv 0 \pmod{g(z)}$, $|\{j \mid b_j \neq w_j\}| \leqslant S$.

Доказывающий демонстрирует знание значений b_j и g_k, из которых были получены экземпляры привязки W_j и V_k, а также справедливость утверждения о весе ошибке.

1) Проверяющий выбирает запрос $d \in \mathbb{Z}_q$ и пересылает его Доказывающему.
2) Доказывающий выбирает в $\mathbb{Z}_q$ значения α_k, ζ_k, $k = 0, \ldots, T$; β_j, η_j, $j = 1, \ldots, N$; μ_t, $t = 0, \ldots, N$; τ_s, $s = 0, \ldots S$; получает коэффициенты r_t, p_s, экземпляры привязки U_k, Q_j, R_t, P_s:

$$\sum_{j=1}^{N} (yb_j + \beta_j) \sum_{k=1}^{T} (yg_k + \alpha_k) \frac{d^k - a_j^k}{d - a_j} \prod_{\substack{i=1, \\ i \neq j}}^{N} \sum_{m=0}^{T} (yg_m + \alpha_m) a_i^m = \sum_{t=0}^{N} y^t r_t,$$

$$\prod_{j=1}^{N} (yb_j + \beta_j - yw_j) = \sum_{s=0}^{S} y^s p_s,$$

$$U_k = h^{\alpha_k} f^{\zeta_k}, \qquad Q_j = h^{\beta_j} f^{\eta_j}, \qquad R_t = h^{r_t} h^{\mu_t}, \qquad P_s = h^{p_s} h^{\tau_s}.$$

   Доказывающий пересылает U_k, Q_j, R_t, P_s Проверяющему.
3) Проверяющий выбирает запрос $c \in \mathbb{Z}_q$ и пересылает его Доказывающему.
4) Доказывающий получает ответы и пересылает их Проверяющему:

$$\Psi_k = cg_k + \alpha_k, \qquad \Theta_k = c\theta_k + \zeta_k, \qquad \Omega_j = cb_j + \beta_j, \qquad \Phi_j = c\varphi_j + \eta_j,$$

$$\Delta = \sum_{t=0}^{N} c^t \mu_t, \qquad \Delta' = \sum_{s=0}^{S} c^s \tau_s.$$

5) Проверяющий получает значения проверочного полинома:

$$\Gamma = \sum_{j=1}^{N} \Omega_j \sum_{k=1}^{T} \Psi_k \tfrac{d^k - a_j^k}{d - a_j} \prod_{\substack{i=1,\\ i\neq j}}^{N} \sum_{m=0}^{T} \Psi_m a_i^m, \qquad \Gamma' = \prod_{j=1}^{N} (\Omega_j - c w_j).$$

Проверяющий рассматривает демонстрацию как убедительную, если

$$h^{\Psi_k} f^{\Theta_k} V_k^{-c} = U_k \ \wedge \ h^{\Omega_j} h^{\Phi_j} W_j^{-c} = Q_j \ \wedge \ h^{\Gamma} f^{\Delta} \prod_{t=0}^{N} R_t^{-c^t} = 1 \ \wedge \ h^{\Gamma'} f^{\Delta'} \prod_{s=0}^{S} P_s^{-c^s} = 1.$$

Нетрудно проверить, что протокол P обладает свойством полноты: честный Доказывающий всегда принимает положительное решение, если вес ошибки в искажённом кодовом слове не превышает порог.

Лемма 1. Любой Доказывающий, получающий приемлемые ответы, не совпадающие со значениями линейных полиномов, нарушает предположение о вычислительной сложности задачи поиска логарифма.

Доказательство. Пусть имеется Доказывающий, способный с более чем ничтожной вероятностью находить приемлемые ответы Ψ_k, Θ_k на запрос Проверяющего c, отличные от значений линейных полиномов $cg_k + \alpha_k$, $c\theta_k + \zeta_k$. Для любой пары элементов циклической группы (h, f), $f \neq 1$, найдется некоторое значение $\chi \in \mathbb{Z}_q$, такое, что $h = f^{\chi}$. Тогда $V_k = f^{\chi g_k + \theta_k}$, $U_k = f^{\chi\alpha_k + \zeta_k}$, проверяемое условие $\chi\Psi_k + \Theta_k - c(\chi g_k + \theta_k) - (\chi\alpha_k + \zeta_k) = 0$. Предположим, что $\Psi_k - cg_k - \alpha_k \neq 0$. Тогда $\log_f(h) = -(\Theta_k - c\theta_k - \zeta_k)/(\Psi_k - cg_k - \alpha_k)$. Аналогично для ответов (Ω_j, Φ_j). ■

Теорема 1. Протокол P является аргументом со свойством знания в предположении сложности задачи поиска дискретного логарифма.

Доказательство. Алгоритм *extractor* выполняет возврат Доказывающего после получения ответов $\Psi_k, \Theta_k, \Omega_j, \Phi_j, \Delta, \Delta'$ на запрос c к состоянию ожидания запроса Проверяющего и получает от Доказывающего повторные ответы $\bar{\Psi}_k, \bar{\Theta}_k, \bar{\Omega}_j, \bar{\Phi}_j, \bar{\Delta}, \bar{\Delta}'$ на повторный запрос $\bar{c} \neq c$ без изменения начальных случайных значений α_k, β_j. *Extractor* вычисляет свидетельство из любой такой пары приемлемых ответов: $g_k = (\Psi_k - \bar{\Psi}_k)/(c - \bar{c})$, $b_j = (\Omega_j - \bar{\Omega}_j)/(c - \bar{c})$. ■

Теорема 2. Протокол P является аргументом в предположении сложности задачи поиска дискретного логарифма.

Доказательство. Рассмотрим произвольный алгоритм Доказывающего с искажённым кодовым словом, содержащим ошибку веса, превышающего S. Тогда полином $\hat{E}(y) = E^*(y) - \sum_{s=0}^{S} y^s p_s$ ненулевой для любых p_s. Вероятность случайного выбора корня c полинома $\hat{E}(y)$ не превышает N/q. Пусть $\hat{E}(c) \neq 0$. Тогда такой Доказывающий также получает нетривиальный логарифм $\log_f(h) = (-\Delta' + \sum_{s=0}^{S} c^s \tau_s)/\hat{E}(c)$. Таким образом, вероятность для честного Проверяющего ошибочно принять положительное решение на этапе проверки веса ошибки не превышает N/q.

Предположим, Доказывающий создал экземпляры привязки к некоторым значениям b_j и g_k, которые не являются кодовым словом. Тогда полином $W^*(z)$ (4) ненулевой, $\deg(W^*(z)) = T - 1$, вероятность выбрать корень d полинома $W^*(z)$ не превышает $(T-1)/q$. Пусть $W^*(d) \neq 0$. Тогда старший коэффициент полинома

$\hat{R}(y) = R^*(y) - \sum\limits_{t=0}^{N} y^t r_t$ ненулевой для любых r_t, вероятность выбрать корень c полинома $\hat{R}(y)$ не превышает $(N+1)/q$. Пусть $\hat{R}(c) \neq 0$. Тогда такой Доказывающий также получает нетривиальный логарифм $\log_f(h) = (-\Delta + \sum\limits_{t=0}^{N} c^s \mu_t)/\hat{R}(c)$. Таким образом, вероятность для честного Проверяющего ошибочно принять положительное решение на этапе проверки кодового слова не превышает $(N+T)/q$.

Таким образом, вероятность ошибочно принять положительное решение в рамках протокола для честного Проверяющего ничтожна, параметром является длина битового представления порядка группы q. ■

Теорема 3. Протокол P имеет свойство специального нулевого разглашения.

Доказательство. Рассмотрим следующий моделирующий алгоритм.

Входные параметры: запросы $(d, c) \in \mathbb{Z}_q$ Проверяющего.

1) Проверяющий выбирает в $\mathbb{Z}_q$ случайные элементы Ψ_k, Θ_k, $k = 0, \ldots, T$; Ω_j, Φ_j, $j = 1, \ldots, N$; Δ, Δ'.
2) Проверяющий выбирает в группе $\mathbb{G}$ случайные элементы U_k, $k = 0, \ldots, T$; Q_j, $j = 1, \ldots, N$; R_t, $t = 1, \ldots, N$; P_s, $s = 1, \ldots, S$.
3) Проверяющий получает

$$\Gamma = \sum_{j=1}^{N} \Omega_j \sum_{k=1}^{T} \Psi_k \frac{d^k - a_j^k}{d - a_j} \prod_{\substack{i=1, \\ i \neq j}}^{N} \sum_{m=0}^{T} \Psi_m a_i^m, \qquad \Gamma' = \prod_{j=1}^{N} (\Omega_j - c w_j),$$

$$U_k = h^{\Psi_k} f^{\Theta_k} V_k^{-c}, \quad Q_j = h^{\Omega_j} h^{\Phi_j} W_j^{-c}, \quad R_0 = h^{\Gamma} f^{\Delta} \prod_{t=1}^{N} R_t^{-c^t}, \quad P_0 = h^{\Gamma'} f^{\Delta'} \prod_{s=1}^{S} P_s^{-c^s}.$$

Моделируемой стенограммой является строка из элементов d, U_k, Q_j, R_t, P_s, c, Ψ_k, Θ_k, Ω_j, Φ_j, Δ, Δ'.

Компоненты стенограммы имеют равномерное распределение; зависимости между компонентами моделируемой стенограммы, использованные на шаге 3 моделирующего алгоритма для расчёта U_k, Q_j, R_0, P_0, совпадают с зависимостями между компонентами стенограммы протокола с Доказывающим. ■

Заключение

Предложен протокол для проверки утверждения о весе ошибки в искажённом кодовом слове и схема запрос-ответ для проверки справедливости утверждений о полиномах. Предложенная схема запрос-ответ также использовалась для протоколов проверки условий о множествах [15], графах [16, 17] и строках [18].

ЛИТЕРАТУРА

1. *Федюкович В. Е.* Протокол аргумента знания слова кода Гоппы и ошибки ограниченного веса // Прикладная дискретная математика. Приложение. 2009. № 1. С. 30–32.
2. *Goldreich O., Micali S., Wigderson A.* Proofs that yield nothing but their validity for all languages in NP have zero-knowledge proof systems // J. ACM. 1991. V. 38. No. 3. P. 691–729.
3. *Brassard G., Chaum D., Crépeau C.* Minimum disclosure proofs of knowledge // J. Comput. Syst. Sci. 1988. V. 37. No. 2. P. 156–189.
4. *Bellare M., Goldreich O.* On defining proofs of knowledge // CRYPTO. 1992. P. 390–420.
5. Computationally convincing proofs of knowledge / G. Brassard, C. Crépeau, S. Laplante, C. Léger // STACS. 1991. P. 251–262.

6. *Bellare M., Micali S., Ostrovsky R.* The (true) complexity of statistical zero knowledge // STOC. 1990. P. 494–502.
7. *Cramer R., Damgård I., Schoenmakers B.* Proofs of partial knowledge and simplified design of witness hiding protocols // CRYPTO. 1994. P. 174–187.
8. *Fiat A., Shamir A.* How to prove yourself: Practical solutions to identification and signature problems // CRYPTO. 1986. P. 186–194.
9. *Варновский Н. П.* Типы нулевого разглашения // XI Междунар. школа-семинар «Синтез и сложность управляющих систем». Нижний Новгород, 2000.
10. *Pedersen T. P.* Non-interactive and information-theoretic secure verifiable secret sharing // CRYPTO. 1991. P. 129–140.
11. *Schnorr C. P.* Efficient identification and signatures for smart cards // CRYPTO. 1989. P. 239–252.
12. *Chaum D., Evertse J. H., van de Graaf J.* An improved protocol for demonstrating possession of discrete logarithms and some generalizations // EUROCRYPT. 1987. P. 127–141.
13. *Гоппа В. Д.* Новый класс линейных корректирующих кодов // Проблемы передачи информации. 1970. Т. 6. № 3. С. 24–30. `http://mi.mathnet.ru/eng/ppi1748`.
14. *Гоппа В. Д.* Рациональное представление кодов и (L, g)-коды // Проблемы передачи информации. 1971. Т. 7. № 3. С. 41–49. `http://mi.mathnet.ru/eng/ppi1647`.
15. *Федюкович В. Е.* Изменчивые ключи подписи // Информационные технологии и системы (ИТиС'09): сборник трудов конференции. [Электронный ресурс] М.: ИППИ РАН, 2009. С. 396–400. `http://www.iitp.ru/ru/conferences/539.htm` (3.12.2009).
16. *Федюкович В. Е.* Протокол аргумента для цикла Гамильтона // Препринт IACR. 2008. `http://eprint.iacr.org/2008/363` (23.08.2008).
17. *Fedyukovych V.* Protocols for graph isomorphism and hamiltonicity // 9th Central European Conference on Cryptography — Trebic'09, June 23–26, 2009. The proceedings will be published as a special issue of Tatra Mountains Mathematical Publications.
18. *Федюкович В. Е., Шарапов В. Г.* Протокол демонстрации K-кратного вхождения строки //Информационные технологии и системы (ИТиС'08): сборник трудов конференции. [Электронный ресурс] М.: ИППИ РАН, 2008. С. 459–466. `http://www.iitp.ru/upload/content/340/itas08_proceedings.pdf` (9.09.2008).