конференция РусКрипто

ИТМО

Модифицированная электронная подпись на основе схемы Штерна

Ниткин Иван Сергеевич
магистрант ФБИТ ИТМО

Научный руководитель – Давыдов В.В., преподаватель, Университет ИТМО

21 марта 2024

Ниткин Иван
Университет ИТМО

Схема Штерна:

$\mathbb{P}$	$\mathbb{V}$
	n, k, ω $H^{n\times k}, h(\cdot): \{0,1\}^* \to \{0,1\}^l$ $\boldsymbol{sk}: s \in \{0,1\}^n : \mathrm{wt}(s) = \omega$ $\boldsymbol{pk}: y: Hs^T = y$
$\sigma \in S_n, u \in \{0,1\}^n$ $c_0 = h(\sigma \mid Hu^T),$ $c_1 = h(\sigma(u)),$ $c_2 = h(\sigma(u \oplus s))$	
	$\xrightarrow{\quad}$
	$b \in \{0,1,2\}$
$\xleftarrow{\quad}$	

$\mathbb{P}$	$\mathbb{V}$
Если $b = 0$:	
σ, u	$\xrightarrow{\quad}$
	$h(\sigma \mid Hu^T) \stackrel{?}{=} c_0$ $h(\sigma(u)) \stackrel{?}{=} c_1$
Если $b = 1$:	
$\sigma, u \oplus s$	$\xrightarrow{\quad}$
	$h(\sigma \mid H(u \oplus s)^T \oplus y) \stackrel{?}{=} c_0$ $h(\sigma(u \oplus s)) \stackrel{?}{=} c_2$
Если $b = 2$:	
$\sigma(u), \sigma(s)$	$\xrightarrow{\quad}$
	$h(\sigma(u)) \stackrel{?}{=} c_1$ $h(\sigma(u) \oplus \sigma(s)) \stackrel{?}{=} c_2$ $\mathrm{wt}(\sigma(s)) \stackrel{?}{=} \omega$

Ниткин Иван
Университет ИТМО

Схема Штерна:

Как обмануть схему Штерна без знания секрета:

Если $b \neq 1$:

$$s \to t_1 \in \{0,1\}^n: \ \mathrm{wt}(t_1) = \omega.$$

Если $b \neq 2$:

$$s \to t_2 \in \{0,1\}^n: \ y = Ht_2^T.$$

Если $b \neq 0$:

$$s \to t_1 \in \{0,1\}^n: \ \mathrm{wt}(t_1) = \omega,$$

$$c_0 = h(\sigma | H(u \oplus t_1)^T \oplus y).$$

Вероятность принятия доказательства злонамеренного доказывающего в **одном** раунде:

$$P = \frac{2}{3}$$

Ниткин Иван
Университет ИТМО

Электронная подпись на основе схемы Штерна:

Системные параметры:

$$n, k, \omega, \boldsymbol{\delta}$$

$$H^{n \times k}$$

$$h(\cdot): \{0,1\}^* \to \{0,1\}^l$$

$$\boldsymbol{f}(\cdot): \{0,1\}^* \to \{0,1,2\}^{\delta}$$

Генерация ключей:

$$\boldsymbol{sk}: s \in \{0,1\}^n : wt(s) = \omega$$

$$\boldsymbol{pk}: y: Hs^T = y$$

Алгоритм формирования подписи:

1. Повторить δ раз:
   1.1. Сгенерировать σ_i, u_i.
   1.2. Вычислить $c_i = c_{i0}|c_{i1}|c_{i2}$.
2. Вычислить $c = c_0| \dots |c_{\delta-1}$.
3. Вычислить $b = f(c|m)$. $b = \{0,1,2\}^{\delta}$.
4. δ раз:
   Для b_i, u_i, σ_i вычислить r_i:

Если $b_i = 0$, $r_i = u_i|\sigma_i$;

Если $b_i = 1$, $r_i = (u_i \oplus s)|\sigma_i$;

Если $b_i = 2$, $r_i = \sigma_i(u_i)|\sigma_i(s)$;

5. Вычислить $r = r_0| \dots |r_{\delta-1}$.
6. Вычисляется значение подписи $Sig = c|r$.

Ниткин Иван
Университет ИТМО

Электронная подпись на основе схемы Штерна:

Системные параметры:

$$n, k, \omega, \boldsymbol{\delta}$$

$$H^{n \times k}$$

$$h(\cdot): \{0,1\}^* \to \{0,1\}^l$$

$$\boldsymbol{f}(\cdot): \{0,1\}^* \to \{0,1,2\}^{\delta}$$

Генерация ключей:

$$\boldsymbol{sk}: s \in \{0,1\}^n : \mathrm{wt}(s) = \omega$$

$$\boldsymbol{pk}: y : Hs^T = y$$

Алгоритм проверки подписи:

1. Вычислить $b = f(c|m).\ b = b_0\,|b_1\,|\ldots|b_{\delta-1}$
2. Для каждого b_i выполнить проверку:

Если $b_i = 0$:

$$c_{i0} \stackrel{?}{=} h\left(\sigma_i \middle| Hu_i^T\right), c_{i1} \stackrel{?}{=} h\left(\sigma_i(u_i)\right)$$

Если $b = 1$:

$$c_{i0} \stackrel{?}{=} h(\sigma_i | H(u_i \oplus s)^T \oplus y), c_{i2} \stackrel{?}{=} h\left(\sigma_i(u_i \oplus s)\right)$$

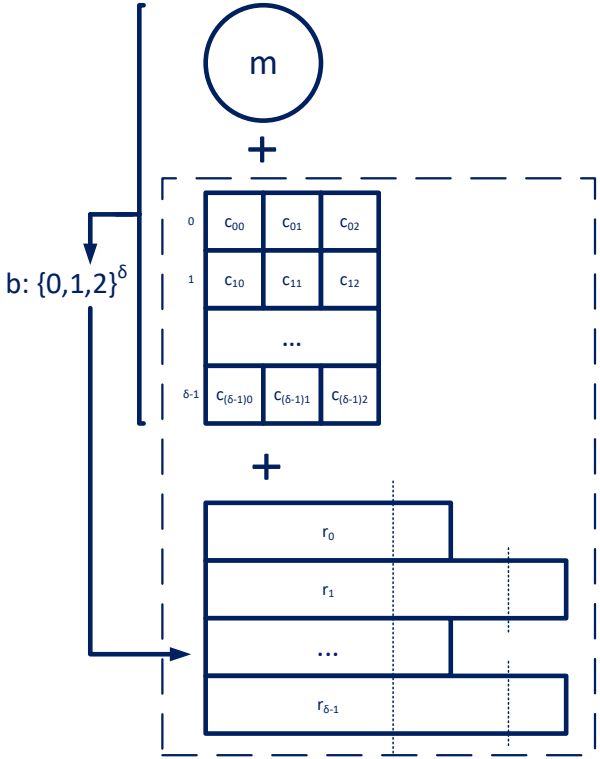
Если $b = 2$:

$$c_{i1} \stackrel{?}{=} h\left(\sigma_i(u_i)\right), c_{i2} \stackrel{?}{=} h\left(\sigma_i(u_i \oplus s)\right), \mathrm{wt}(s) \stackrel{?}{=} \omega$$

Подпись принимается если пройдены все проверки.

Ниткин Иван
Университет ИТМО

Подпись Шиповник:



n	620[1]	2896[2]
k	310	1488
ω	68	318
δ	137	137
l	(112)	512
$\lvert c_i \rvert$, бит	$3l = 336$	$3l = 1536$
$\lvert r_{\sigma(s)} \rvert$, бит	$2n = 1240$	$2n = 5792$
$\lvert r_\sigma \rvert$, бит	$n + n \cdot log_2 n = 6200$	$n + n \cdot log_2 n = 36199$

[1] *Partha Sarathi Roy, Kirill Morozov, Kazuhide Fukushima, & Shinsaku Kiyomoto. (2019). Evaluation of Code-based Signature Schemes.*

[2] *Victoria Vysotskaya, & Ivan Chizhov. (2021). The security of the code-based signature scheme based on the Stern identification protocol.*

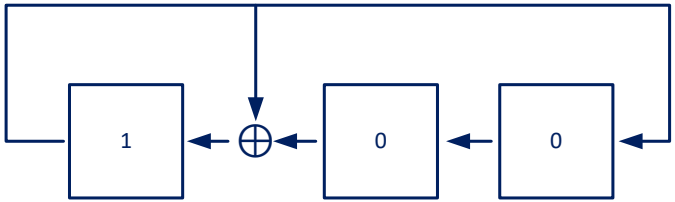
Ниткин Иван
Университет ИТМО

Алгоритм генерации перестановки при помощи РСЛОС:

LSFR_perm_gen(N):

1. Построить РСЛОС на основе примитивного многочлена;
2. Инициализировать РСЛОС случайным значением IV от 1 до N;
3. **Повторить** $2^L - 1$ раз:
   3.1. Прочитать значение регистра;
   3.2. **Если** значение состояния $\leq N$ – записать его в вектор перестановки;
   3.3. Выполнить 1 такт работы РСЛОС.

Результат: IV, вектор перестановки длины N.



Пример 1.

Необходимо сгенерировать перестановку длины $N = 5$ при помощи РСЛОС.

1. Выберем длину РСЛОС:

$$L = \lceil log_2 N \rceil = 3$$

2. Выберем примитивный многочлен степени $L = 3$ над полем $GF(2)$: $x^3 + x + 1$.

3. Построим РСЛОС.

4. Инициализируем РСЛОС произвольно выбранным значением $IV = 4 = 100_2$:

5. Выполним период работы РСЛОС:

№ такта	*Состояние РСЛОС*		*Вектор перестановки*
0	4	100_2	(4)
1	5	101_2	(4,5)
2	7	111_2	(4,5)
3	3	011_2	(4,5,3)
4	6	100_2	(4,5,3)
5	1	001_2	(4,5,3,1)
6	2	010_2	(4,5,3,1,2)

Ниткин Иван
Университет ИТМО

Алгоритм расширения перестановки:

Perm_extention(perm, γ):

1. **Повторить** γ раз:
   1.1. Сгенерировать число pos от 0 до $len(perm)$;
   1.2. Записать число в вектор значений;
   1.3. Сформировать вектор перестановки следующим образом:
      1.3.1. Добавить pos первых значений вектора перестановки $perm$;
      1.3.2. Добавить значение $len(perm) + 1$;
      1.3.3. Добавить оставшиеся значения

Результат: Вектор перестановки длины n. Вектор значений длины γ, который задает перестановку длины n на основе перестановки длины $n - \gamma$.

Пример 2.

Необходимо на основе перестановки $(4,5,3,1,2)$ *сгенерировать перестановку длины* 9.

1. Вычислим количество необходимых расширений $\gamma = 9 - 5 = 4$.

2. Выполним необходимое количество расширений.

№ расширения	*Случайное значение*	*Границы выбора значения*	*Вектор перестановки*	*Вектор значений*
0	–	–	(4,5,3,1,2)	[4]
1	2	$0-5$	$(4,5,6,3,1,2)$	[4, 2]
2	1	$0-6$	$(4,7,5,6,3,1,2)$	[4, 2, 1]
3	0	$0-7$	$(8,4,7,5,6,3,1,2)$	[4, 2, 1, 0]
4	6	$0-8$	$(8,4,7,5,6,3,9,1,2)$	[4, 2, 1, 0, 6]

Ниткин Иван
Университет ИТМО

Подпись Шиповник-М:

Системные параметры:

$$n, k, \omega, \delta, \boldsymbol{\gamma}$$

$$H^{n \times k}$$

$$h(\cdot): \{0,1\}^* \to \{0,1\}^l$$

$$f(\cdot): \{0,1\}^* \to \{0,1,2\}^\delta$$

$$\boldsymbol{p}(\boldsymbol{x}): \deg\big(p(x)\big) = [log_2(n-\gamma)]$$

Генерация ключей:

$$\boldsymbol{sk}: s \in \{0,1\}^n : wt(s) = \omega$$

$$\boldsymbol{pk}: y: Hs^T = y$$

Алгоритм формирования подписи:

1. Повторить δ раз:
   1.1. Сгенерировать u_i.
   1.2. Сгенерировать
   $\boldsymbol{\sigma_i, v_i} =$ **`Perm_extention(LSFR_perm_gen(`$\boldsymbol{n-\gamma}$**`), `$\boldsymbol{\gamma}$**`)`**
   1.2. Вычислить $c_i = c_{i0}|c_{i1}|c_{i2}$.
2. Вычислить $c = c_0| \dots |c_{\delta-1}$.
3. Вычислить $b = f(c|m)$. $b = \{0,1,2\}^\delta$.
4. δ раз:
   Для b_i, u_i, σ_i вычислить r_i:

Если $b_i = 0$, $r_i = u_i|\boldsymbol{v_i}$;

Если $b_i = 1$, $r_i = (u_i \oplus s)|\boldsymbol{v_i}$;

Если $b_i = 2$, $r_i = \sigma_i(u_i)|\sigma_i(s)$;

5. Вычислить $r = r_0| \dots |r_{\delta-1}$.
6. Вычисляется значение подписи $Sig = c|r$.

Ниткин Иван
Университет ИТМО

Подпись Шиповник-М:

Системные параметры:

$$n, k, \omega, \delta, \gamma$$

$$H^{n \times k}$$

$$h(\cdot): \{0,1\}^* \to \{0,1\}^l$$

$$f(\cdot): \{0,1\}^* \to \{0,1,2\}^\delta$$

$$p(x): \deg(p(x)) = [log_2(n-\gamma)]$$

Генерация ключей:

$$\boldsymbol{sk}: s \in \{0,1\}^n : wt(s) = \omega$$

$$\boldsymbol{pk}: y: Hs^T = y$$

Алгоритм проверки подписи:

1. Вычислить $b = f(c|m).\ b = b_0\,|b_1\,|\ldots|b_{\delta-1}$
2. Для каждого b_i выполнить проверку:

Если $b_i = 0$:

2.1. **Восстановить σ_i по значению v_i;**

2.2. $c_{i0} \stackrel{?}{=} h\left(\sigma_i \middle| Hu_i^T\right), c_{i1} \stackrel{?}{=} h(\sigma_i(u_i))$

Если $b = 1$:

2.1. Восстановить σ_i по значению v_i;

2.2. $c_{i0} \stackrel{?}{=} h(\sigma_i|H(u_i \oplus s)^T \oplus y), c_{i2} \stackrel{?}{=} h(\sigma_i(u_i \oplus s))$

Если $b = 2$:

$c_{i1} \stackrel{?}{=} h(\sigma_i(u_i)), c_{i2} \stackrel{?}{=} h(\sigma_i(u_i \oplus s)), \mathrm{wt}(s) \stackrel{?}{=} \omega$

Подпись принимается если пройдены все проверки.

Ниткин Иван
Университет ИТМО

Уровень криптографической стойкости подписи Шиповник

1. Подделка подписи.

$$-\log_2\left(\frac{2}{3}\right)^{\delta} = 80 \text{ бит}$$

Конкретные значения приведены для параметров:

n	620
k	310
ω	68
δ	137
l	112

2. Раскрытие секретного ключа.

– Перебор значений:

$$\log_2\binom{n}{\omega} \approx 305 \text{ бит}$$

– Решение задачи синдромного декодирования:

$$k = 310 \text{ бит}$$

– Декодирование по информационным совокупностям:

$$k = 310 \text{ бит}$$

– На основе «ответа» $u \oplus s$*:*

$$n = 620 \text{ бит}$$

– На основе «ответа» $\boldsymbol{\sigma(s)}$:

$$-\log_2 \frac{\omega!(n-\omega)!}{n!} = \log_2\binom{n}{\omega} \approx 305 \text{ бит}$$

Ниткин Иван
Университет ИТМО

Уровень криптографической стойкости подписи Шиповник

Алгоритм атаки на подбор значения секретного ключа по известному значению $\sigma(s)$:

1. Выбрать случайную перестановку σ_s.
2. Вычислить $\sigma_s(\sigma(s))$.
3. Вычислить $H\left(\sigma_s(\sigma(s))\right)^T$.
4. Проверить $y \stackrel{?}{=} H\left(\sigma_s(\sigma(s))\right)^T$.

Если проверка пройдена – значение секретного ключа восстановлено.

Вероятность успешной атаки:

$$P_S = \frac{Q_1 Q_0}{Q} = \frac{\omega!\,(n-\omega)!}{n!} = \binom{n}{\omega}^{-1}$$

Q – общее количество возможных перестановок;
Q_1 – количество способов поместить ω единиц двоичного вектора s на ω позиций.
Q_0 – количество способов поместить $n-\omega$ нулей двоичного вектора s на $n-\omega$ позиций

Ниткин Иван
Университет ИТМО

Уровень криптографической стойкости подписи Шиповник-М

Алгоритм атаки на подбор значения секретного ключа по известному значению $\sigma(s)$:

1. Выбрать случайную перестановку σ_s.
2. Вычислить σ_s^{-1}: $\sigma_s^{-1}(\sigma) = e$, где $e = (1, \dots, n)$
3. Вычислить $\sigma_s^{-1}(\sigma(s))$.
3. Вычислить $H * \left(\sigma_s^{-1}(\sigma(s))\right)^T$.
4. Проверить $y \stackrel{?}{=} H * \left(\sigma_s^{-1}(\sigma(s))\right)^T$.

Если проверка пройдена – значение секретного ключа восстановлено.

Вероятность успешной атаки:

$$P_{SM} = \frac{Q_1 Q_0}{Q}$$

Q – общее количество возможных перестановок;
Q_1 – количество способов поместить ω единиц двоичного вектора s на ω позиций.
Q_0 – количество способов поместить $n - \omega$ нулей двоичного вектора s на $n - \omega$ позиций

Уровень криптографической стойкости подписи Шиповник-М

Вычисление значения Q для подписи Шиповник-М:

Количество расширений	Количество значений вектора		Мощность множества перестановок
0	1	→	n
1	2	→	$n(n-1)$
⋮	⋮	→	⋮
$\gamma-1$	γ	→	$\prod_{j=0}^{\gamma-1}(n-j)$
γ	$\gamma+1$	→	$\prod_{j=0}^{\gamma}(n-j)=\frac{n!}{(n-\gamma-1)!}$

$$Q=\frac{n!}{(n-\gamma-1)!}.$$

Вычисление значения Q_1 для подписи Шиповник-М:

q_1 - количество единиц в первых $n-\gamma$ разрядах вектора s.

$$Q_1=\frac{\omega!}{(q_1-1)!}$$

Вычисление значения Q_0 для подписи Шиповник-М:

q_0 - количество единиц в первых $n-\gamma$ разрядах вектора s.

$$Q_0=\frac{(n-\omega)!}{(q_0-1)!}=\frac{(n-\omega)!}{(n-\gamma-q_1-1)!}$$

Ниткин Иван
Университет ИТМО

Уровень криптографической стойкости подписи Шиповник-М

Вероятность успешной атаки:

$$P_{SM} = \frac{Q_1 Q_0}{Q} =$$

$$= \frac{\frac{\omega!}{(q_1 - 1)!} \cdot \frac{(n - \omega)!}{(n - \gamma - q_1 - 1)!}}{\frac{n!}{(n - \gamma - 1)!}} =$$

$$= P_S \cdot \frac{(n - \gamma - 1)!}{(q_1 - 1)!\,(n - \gamma - q_1 - 1)!}$$

Выводы (предварительные):

1. Перестановку длины n можно задать при помощи $n - 1$ значений длины $\log_2 n$, без потери уровня криптостойкости.

2. Может быть вычислено минимальное значение γ, обеспечивающее стойкость при атаке на подбор значения секретного ключа по известному значению $\sigma(s)$ на уровне общей стойкости подписи Шиповник.

Ниткин Иван
Университет ИТМО

Сравнение схем подписи Шиповник и Шиповник-М:

Показатель	Шиповник	Шиповник-М
n	620	620
k	310	310
ω	68	68
δ	137	137
l	112	112
γ	–	**341**
$\lvert c_i \rvert$, бит	$3l = 336$	$3l = 336$
$\lvert r_{\sigma(s)} \rvert$, бит	$2n = 1240$	$2n = 1240$
$\lvert r_\sigma \rvert$, бит	$n + n \cdot log_2 n = 6200$	$n + (\gamma + 1) \cdot log_2 n =$ **3698**
Уровень криптостойкости, бит	80	80
Уровень криптостойкости при атаке на подбор s, бит	305	**80**

Результаты эксперимента:

Наименование показателя	Отношение показателя подписи Шиповник-М к подписи Шиповник
Среднее время подписи сообщений	1,109
Среднее время проверки подписи сообщения	1,139
Полезный размер подписи	0,656
Фактический размер подписи	0,607

Ниткин Иван
Университет ИТМО

Итоги:

1. Предложен **алгоритм генерации перестановки.** Такая перестановка описывается при помощи **вектора значений сокращенной длины**.

2. Предложена **модифицированная схема электронной подписи** на основе схемы Штерна. Разработанная схема позволяет **сократить размеры подписи.**

3. Рассчитано значение параметра «количество расширений при генерации перестановки», которое **позволяет обеспечить уровень криптостойкости** модифицированной подписи **на уровне базовой подписи.**

@INITKIN

Спасибо за внимание!

ITMO more than a UNIVERSITY

Ниткин Иван
exebopen@gmail.com