ПЕРСПЕКТИВНІ МЕТОДИ ТА СИСТЕМИ КРИПТОГРАФІЧНОГО ЗАХИСТУ ІНФОРМАЦІЇ

УДК 004.056.55 **DOI:10.30837/rt.2020.1.200.01**

А.М. ОЛЕКСІЙЧУК, д-р техн. наук, В.А. КУЛІБАБА, М.В. ЄСІНА, канд. техн. наук, С.О. КАНДІЙ, Є.В. ОСТРЯНСЬКА, І.Д. ГОРБЕНКО, д-р техн. наук

ОБҐРУНТУВАННЯ ПЕРСПЕКТИВНОГО ПОСТКВАНТОВОГО НАЦІОНАЛЬНОГО СТАНДАРТУ ЕЛЕКТРОННОГО ПІДПИСУ НА ОСНОВІ РЕШІТОК

Вступ

Важливою особливістю постквантового періоду у криптографії є суттєва невизначеність щодо вихідних даних для криптоаналізу та протидії в частині можливостей квантових комп'ютерів, їх математичного та програмного забезпечень, а також застосування квантового криптоаналізу до існуючих криптоперетворень та криптопротоколів. В якості основних методів обрано математичні методи електронного підпису (ЕП), що пройшли суттєвий аналіз та обґрунтування в процесі широких досліджень криптологами та математиками на найвищому рівні [3 – 18]. Вони детально описані та пройшли дослідження на першому етапі міжнародного конкурсу NIST США [23]. В процесі другого етапу прийнято ряд рішень стосовно об'єднання деяких кандидатів на постквантовий стандарт ЦП. Для подальших досліджень на 2-му етапі залишили 9 кандидатів [24]: CRYSTALS-DILITHIUM, FALCON, GeMSS, LUOV, MQDSS, Picnic, qTESLA, Rainbow та SPHINCS+. Три з них (Dilithium, FALCON, qTeSLA) засновані на стійкості алгебраїчних решіток (Lattice-based), чотири (GeMSS, LUOV, MQDSS, Rainbow) – на основі багатовимірних перетворень (multi-variate), один (SPHINCS+) – на стійкості геш-функції, один (Picnic) – на стійкості геш-функції та блокових потокових шифрів.

На наш погляд, національний стандарт Україні постквантового періоду повинен включати в себе мінімум три алгоритми, що базуються на різних видах математичних перетворень, що визнані світовим криптографічним співтовариством як такі, що можуть забезпечувати необхідний рівень стійкості в умовах квантового криптоаналізу.

Одним із видів криптографічних перетворень типу ЕП, що може бути включений в національний стандарт ЕП постквантового періоду, на наш погляд, може стати ЕП на основі застосування алгебраїчних решіток (Lattice-based) [24].

1. Огляд основних операцій в алгоритмі підпису Dilithium 2-го етапу конкурсу NIST

На рис.1 зображено узагальнену схему ЕП CRYSTALS-DILITHIUM, що подається у [2].

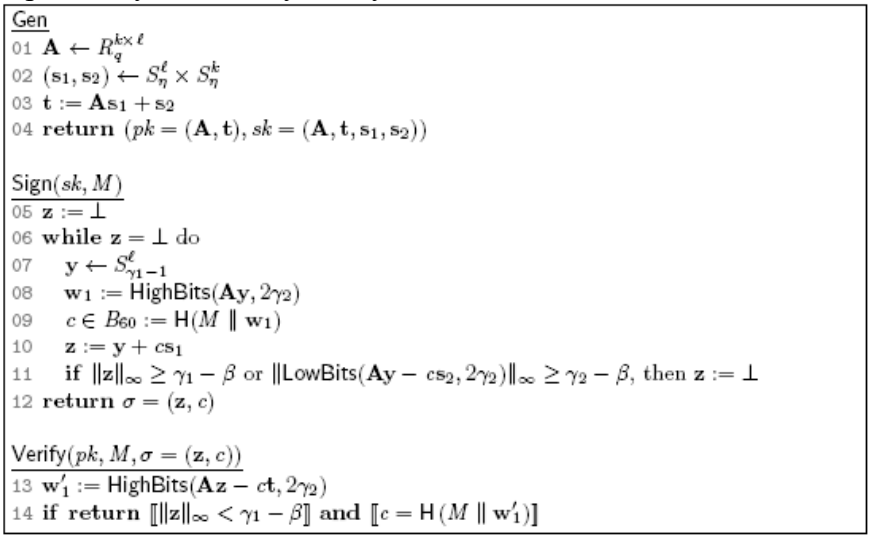

Рис. 1. Узагальнена схема ЕП CRYSTALS-DILITHIUM

Алгоритм належить до так званого класу "Fiat-Shamir з перериваннями"[3]. На рис. 1 подаються у спрощеному вигляді алгоритми генерації відкритого та секретного ключів, а також підпису і перевірки підпису.

Генерація основних складових ключа. Спочатку (рядок 01, рис.1) генерується матриця поліномів **A** розміру $k \times \ell$, кожен з елементів якої є поліномом у кільці $R_q = \mathbb{Z}_q[X]/(X^n+1)$. В процесі попереднього розгляду будемо вважати, що модуль $q=2^{23}-2^{13}+1$, а степінь полінома $n=256$. Потім генеруються (обчислюються) випадкові вектори, тобто множини поліномів секретного ключа $\mathbf{s}_1$ і $\mathbf{s}_2$ (рядок 02) відповідно з числом поліномів k та l. Коефіцієнти цих векторів (поліномів) є елементами поля R_η, тобто з (малими) коефіцієнтами з розміром не більше η (від $-\eta$ до η). Далі з використанням матриці **A** та секретного ключа $\mathbf{s}_1$ і $\mathbf{s}_2$ обчислюється друга частина відкритого ключа $t=As_1+s_2$ (рядок 03), а перша частина відкритого ключа задається значенням ρ. Всі алгебраїчні операції з поліномами в механізмі виконуються над кільцем полінома $R_q = \mathbb{Z}_q[X]/(X^n+1)$. Четвертий рядок показує вихідні значення відкритого P_k та секретного S_k ключів.

Алгоритм підпису у загальному виді. Спочатку в алгоритм перевірки ЕП вводяться значення секретного ключа S_k та повідомлення M, що підписується. Далі обчислюється вектор поліномів маскування **y** з коефіцієнтами, що є меншими, ніж γ_1 (рядок 07), а також обчислюється значення вектора поліномів **Ay.** На основі отриманого значення **Ay** обчислюються старші біти $\mathbf{w}_1$ ("біти високого порядку") коефіцієнтів у цьому векторі поліномів (рядок 08). $\mathbf{w}_1$ є вектором, що містить всі поліноми w1. Потім обчислюється поліном (рядок 09) c, що є поліномом у полі R_q з точно 60 символами ±1, а решта 0. Безпосередньо ЕП обчислюється у вигляді вектора поліномів $\mathbf{z}=\mathbf{y}+c\mathbf{s}_1$.

Якщо ЕП **z** вивести безпосередньо після його обчислення (рядок 10), то механізм ЕП Dilithium не буде безпечним через те, що при певних значеннях секретний ключ може бути компрометований. Щоб уникнути залежності **z** від секретного ключа та його витоку використовується відхилення вибірки. Для цього встановлюється значення параметру β як максимально можливий коефіцієнт $c\mathbf{s}_i$. Оскільки c має значення 60 ±1, а максимальний коефіцієнт в $\mathbf{s}_i$ дорівнює $\mathbf{s}_i$, то легко побачити, що $\beta \le 60\eta$. Якщо будь-який коефіцієнт **z** перевищує $\gamma_1 - \beta$, то процес ЕП відхиляється, а процедура ЕП повторюється. Також, якщо коефіцієнт бітів низького порядку вектора $\mathbf{Az}-c\mathbf{t}$ більше ніж $\gamma_2 - \beta$, то процес ЕП відхиляється, а процедура ЕП знову повторюється (рядок 11). Перша перевірка необхідна для безпеки ЕП, а інша – для його безпеки та правильності. Таким чином, процес ЕП повторюється, доти не будуть виконані дві наведені умови. Необхідно відмітити, що параметри β та γ_1 і γ_2 повинні бути вибрані, щоби очікувана кількість повторень ЕП була не надто висока (наприклад, від 4 до 7).

Алгоритм перевірки підпису у загальному виді. Спочатку перевірник обчислює $\mathbf{w}'_1$ як біти високого порядку вектора $\mathbf{Az}-c\mathbf{t}$. Далі ЕП приймається, якщо всі коефіцієнти **z** менше, ніж $\gamma_1 - \beta$ і, якщо c є геш-значенням повідомлення M, що перевіряється, та значення $\mathbf{w}'_1$ (рядок 13). Перевірка виконується за умови, якщо

$$\text{HighBits}(\mathbf{Az}-c\mathbf{t}, 2\gamma_2) = \text{HighBits}(\mathbf{Ay}, 2\gamma_2) \quad (1)$$

Доведено, що причиною цього є те, що для дійсного ЕП буде завжди виконуватись умова

$$\|\text{LowBits}(\mathbf{Ay}-c\mathbf{s}_2, 2\gamma_2)\|_\infty < \gamma_2-\beta. \quad (2)$$

А так як коефіцієнти $c\mathbf{s}_2$ менше ніж β, то додавання $c\mathbf{s}_2$ недостатньо для того, щоб викликати перенесення у старші коефіцієнті, збільшивши будь-який коефіцієнт низького порядку до величини щонайменше γ_2. Таким чином, рівняння (1) є коректним, і підпис перевіряється успішно.

2. Аналіз стійкості алгоритму Dilithium проти основних атак

Наразі в постквантовій криптології актуальними є завдання забезпечення криптографічної стійкості щодо квантових атак. Вона пов'язана з проблемою навчання з помилками.

Проблема навчання з помилками (LWE) визначається наступним чином. Нехай n, q є деякими натуральними числами, χ – деякий ймовірнісний розподіл над $\mathbf{Z}$ та s – секретний вектор (множина поліномів) у $\mathbf{Z}_q^n$. Ймовірнісний розподіл $L_{s,\chi}$ над $\mathbf{Z}_q^n \times \mathbf{Z}_q$ отримується обчисленням [3]

$$(a, c) = (a, \langle a, s \rangle + e) \in \mathbf{Z}_q^n \times \mathbf{Z}_q, \tag{3}$$

де $a \in \mathbf{Z}_q^n$ отримується з рівномірного розподілу та $e \in \mathbf{Z}$ з розподілу χ. В даному випадку атака Decision-LWE полягає у тому, щоб визначити, чи отримана пара $(a, c) \in \mathbf{Z}_q^n \times \mathbf{Z}_q$ з розподілу $L_{s,\chi}$, або рівномірного розподілу. Її Search-LWE складова полягає у знаходженні s з пари $(a, c) \in \mathbf{Z}_q^n \times \mathbf{Z}_q$ [3]. Вважається, що як проблема Decision-LWE, так і Search-LWE [2 – 7] з точки зору складності є еквівалентними та можуть бути зведені одна до одної за поліноміальний час і фактично є різними поглядами на одну і ту ж задачу. Розподіл χ для цих задач зазвичай є дискретним нормальним розподілом над кінцевим полем з математичним очікуванням рівним 0 та дисперсією, що характеризується параметром α. При цьому більшість атак на LWE полягають у знаходженні деякого вектору v з певною нормою на решітці L з фіксованим об'ємом $vol(L)$, але з різною розмірністю m, яка фактично характеризує оптимальну кількість пар $(a_i, c_i) \in \mathbf{Z}_q^n \times \mathbf{Z}_q$ необхідних для атаки.

Аналіз показав, що складність проблеми LWE точно знайдена лише асимптотично. Так, доведено [4, 6], що за певних умов складність вирішення LWE в просторі розмірності n становить щонайменше $2^{O(n)}$. Цей результат зручно використовувати для оцінки загальносистемних параметрів, проте конкретні оцінки складності крипостійкості досі не відомі. Таке пов'язано з тим, що атаки на LWE зводяться в кінцевому випадку до редукції решіток.

В останні 10 років у цьому напрямку є помітний суттєвий прогрес, що призводить до постійного уточнення та зміни оцінок. Він стосується більшості сучасних криптосистем, у яких використовуються варіанти LWE над поліноміальними кільцями (PR-LWE), тобто розподіл розглядається не над $\mathbf{Z}_q$, а над $\mathbf{Z}_q[X]/(f(x))$. При аналізі криптоперетворень засобом множення використовується поліном виду $f(x) = x^{2^n} + 1$ і відповідне поле $R_q = \mathbf{Z}_q[X]/(x^{2^n} + 1)$. При чому, якщо $(a_i, c_i) \in R_q \times R_q$, то задача має назву R-LWE. Коли $(a_i, c_i) \in R_q^d \times R_q$ – M-LWE відповідно.

Аналіз показує, що поліном $f(x) = x^{2^n} + 1$ обраний не випадково. Його властивості дозволяють здійснити доказ щодо стану захищеності асиметричного криптоперетворення щодо квантових атак. Також його властивості дозволяють використати для операцій множення поліномів швидке NTT перетворення, і як наслідок створювати швидкодіючі реалізації криптоперетворень. Однак, з теорії Галуа відомо [3, 4, 6], що поле $R_q = \mathbf{Z}_q[X]/(x^{2^n} + 1)$ має складну структуру підполів, що може бути використано для здійснення криптоаналізу. Проте, на нинішній час, на практиці такі атаки носять більше обмежений теоретичний характер, ніж практичний. Фактично сучасними криптологами ігноруються додаткові можливості, а R-LWE та M-LWE розглядаються як LWE. Це пояснюється тим, що для полінома $f(x) = x^{2^n} + 1$ доведено, що R-LWE та M-LWE є складнішими за LW атаки.

На основі аналізу визначено [7 – 15], що стосовно LWE можливо застосування таких атак:

1. Атака грубої сили, тобто повного перебору.
2. Традиційна атака зустріч посередині.
3. Атака на основі алгоритму Arora-Ge.
4. BKW, коли LWE зводиться до SIS атаки.
5. Primal attack (Search-LWE зводиться до BDD атаки).
6. Dual attack (Decision-LWE зводиться до SIS).
7. Зведення до uSVP атаки пошуку короткого вектора.

Деталі щодо кожного з видів, а також їх теоретичне обґрунтування наведено у [26].

3. Захищеність алгоритму ЕП від атак сторонніми каналами

В процесі проведення конкурсу на постквантовий стандарт ЕП особлива вимога висунута до захищеності кандидату на ЕП від атак сторонніми каналами. Тому така проблемна задача є актуальною, в першу чергу стосовно ЕП Crystals-Dilithium.

Дослідження стосовно алгоритм ЕП Crystals-Dilithium проведено за такими параметрами [2]:

- BKZ bock-size to break SIS = 475;
- BKZ bock-size to break LWE = 485;
- $k = 5; l = 4; \eta = 5; \zeta = 4; \beta = 275; \omega = 96$.

Для проведення експерименту було згенеровано 10000 ключів та виконано 10000 підписів. Результат залежності часу підпису від номеру ключа наведено на рис. 2. Для 10000 ключів максимальне відхилення від нормалізованого середнього (дисперсія) усіх вимірів часу підпису повинно знаходитися в інтервалі $-5.19676 <= d <= 6.62797(\%)$, щоб вважати, що час підпису не залежить від ключа. Номери ключів, для яких було отримано мінімальне та максимальне значення при повтореннях вимірів не повинні співпадати.

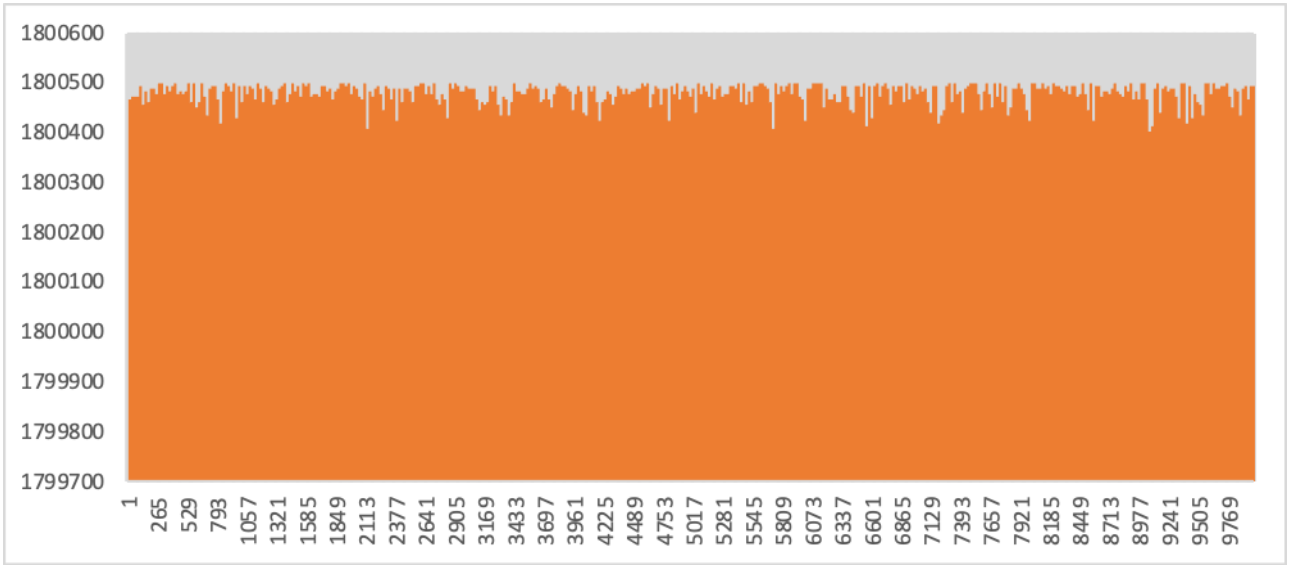

Рис. 2. Залежність часу підпису (у тактах процесору) від номеру ключа

Значення дисперсії $d \approx 2\%$, що свідчить про практично статистичну незалежність часу підпису від ключа, що є важливим з точки зору захищеності від атак сторонніми каналами.

4. Вибір параметрів 5-7 рівнів стійкості

Попередній аналіз показав, що значення параметрів, наведені в [20], табл. 3.1, не забезпечують при застосуванні в механізмі Dilithium стійкість ЕП від класичних атак на рівні 256 бітів. Фактично нижні оцінки стійкості наведено в рядках табл. 3.1 [20] з назвами Best Known Classical bit-cost і Best Known Quantum bit-cost окремо для кожної з двох задач – класичної та

квантової, на складності яких базується стійкість. Це задачі SIS та LWE. Для кожної з них зазначені параметри обчислюються за формулами

$$\text{Best Known Classical bit-cost (класична атака)} = 0,292b \tag{4}$$

$$\text{Best Known Quantum bit-cost (квантова атака)} = 0,265b, \tag{5}$$

де b є довжиною блоку (BKZ bock-size b to break SIS або LWE [2]). В ролі кінцевої оцінки стійкості використовується найменше з двох значень, обчислених для b, що є довжиною блоку для задачі SIS та задачі LWE відповідно.

Наприклад, для останнього стовпця табл. 3.1 в [2] (третя категорія стійкості за вимогами NIST) маємо: BKZ bock-size b to break SIS $= 605$. Отже, значення (4) та (5) дорівнюють відповідно $0,292b = 176$, $0,265b = 160$; BKZ bock-size b to break LWE $= 595$, і значення (4) і (5) дорівнюють відповідно $0,292b = 174$ та $0,265b = 158$ (в табл. 3.1 наведені значення з округленням до цілих, причому не завжди у той самий бік). Таким чином, кінцева нижня оцінка стійкості ЦП для випадку, що розглядається, є 174, тобто схема забезпечує стійкість проти класичної атаки на рівні не менше, ніж 2^{174} та відповідно квантової атаки на рівні не менше, ніж 2^{158}.

Довжина блоку b обчислюється за параметрами k, l, η з табл. 3.1 за допомогою окремих алгоритмів для кожної задачі SIS та LWE. Яких саме, потребує додаткового вивчення, але автори [2] використовують для цього методику, наведену в [2, 3]. Вони зазначають [2, с. 22], що ця методика дозволяє отримувати "консервативні" нижні оцінки стійкості, які "скоріш за все, не зміняться найближчим часом" з появою більш ефективних алгоритмів розв'язання задач SIS та LWE, зокрема, на квантових комп'ютерах. Автори [2] пишуть також про те, що отримані ними оцінки формально є менше в порівнянні з вимогами NIST, але вони вважають, що для параметрів з табл. 3.1 схема Dilithium задовольняє цим вимогам [20].

Таким чином, найвищий рівень класичної стійкості, що забезпечує Dilithium з параметрами, зазначеними в [2, табл. 3.1], є (принаймні) 2^{174}.

Загальні обмеження [2 – 4, 8], які необхідно врахувати при виборі параметрів для забезпечення стійкості на рівні $\lambda \in \{256, 384, 512\}$ бітів.

1. Геш-функція CRH, що використовується у схемі (наприклад, в рядках 07, 10 на рис. 4 [2]), повинна бути стійкою до колізій. Отже, довжина її вектору значень повинна бути, як мінімум, 2λ бітів. (Зокрема, при $\lambda = 256$ функція CRH повинна приймати значення довжини 512, а ні 384, як в оригінальній схемі на рис. 4 [2]).

2. Криптографічна функція H, яка використовується в рядку 16 на рис. 4, повинна бути стійкою відносно знаходження другого прообразу [3] і, отже, приймати, принаймні, 2^{λ} різних значень, що є поліномами з кільця R_q, які мають коефіцієнти $0, 1, -1$ та містять точно h ненульових коефіцієнтів (зауважимо, що кількість таких поліномів дорівнює $2^h \binom{n}{h}$).

При $\lambda = 256$ в [2] рекомендується використовувати параметри $n = 256$, $h = 60$, і умова

$$2^h \binom{n}{h} \geq 2^{\lambda} \tag{6}$$

виконується.

При $\lambda \in \{384, 512\}$ та $n = 256$ забезпечити виконання умови (6) неможливо, якщо $h \leq n/2 = 128$. Отже, треба збільшити n до 512; при цьому числа q, γ_1, γ_2 можна залишити такими самими як в [2]:

$$q = 2^{23} - 2^{13} + 1,$$

$$\gamma_1 = (q-1)/16,\ \gamma_2 = \gamma_1/2. \qquad (7)$$

3. Довжини векторів ρ та K, що використовуються відповідно в рядках 01 та 02 на рис. 4 в [2], повинні бути не менше ніж λ.

Таким чином, для забезпечення стійкості схеми цифрового підпису на рівні $\lambda \in \{256, 384, 512\}$ необхідно [25]:

1) використовувати геш-функцію CRH, значеннями якої є двійкові вектори довжини 2λ;

2) використовувати двійкові вектори ρ та K, що мають довжину λ;

3) покласти $n = 256$, якщо $\lambda = 256$; $n = 512$, якщо $\lambda \in \{384, 512\}$;

4) вибрати просте число $q \equiv 1 \bmod 2n$ та обчислити γ_1, γ_2 за формулою (7);

5) обчислити вагу c як найменше натуральне h, що задовольняє умові (3).

Зауважимо, що при $n = 512$, $q = 2^{23} - 2^{13} + 1$ час формування підпису може виявитися занадто великим; в цьому випадку треба збільшити q (приблизно в два рази).

В [1 – 3] немає чіткого викладення зазначених алгоритмів, є тільки певні вказівки. Тому на сьогодні вдалося "відновити" лише один з них, проте його застосування до параметрів з табл. 1 в [2] призводить до значень b, які є приблизно на 10 % більше наведених в цій таблиці.

5. Алгоритм оцінювання довжини блоку b для задачі LWE, пряма атака

Вхідні дані: натуральні числа k, l, η, n, q.

Алгоритм обчислень: для кожного $m = 0, 1, ..., nk$ виконати такі дії:

1) покласти $d = nl + m + 1$;

2) знайти найбільше натуральне $\tilde{b} = \tilde{b}(m) \le d$ таке, що

$$\eta\sqrt{\tilde{b}} \le \delta^{2\tilde{b}-d-1} q^{\frac{m}{d}}, \qquad (8)$$

де

$$\delta = \left(\left(\pi\tilde{b}\right)^{\frac{1}{\tilde{b}}} \frac{\tilde{b}}{2\pi e} \right)^{\frac{1}{2(\tilde{b}-1)}}.$$

Результатом алгоритму є число

$$b = \min_{0 \le m \le nk} \{\tilde{b}(m)\}. \qquad (9)$$

В табл. 1 наведені значення довжини блоку, отримані за допомогою наведеного вище алгоритму 1, та відповідні значення цього параметра з табл.1 в [2].

Таблиця 1

Значення щодо довжини блоку

(k, l)	(3, 2)	(4, 3)	(5, 4)	(6, 5)
η	7	6	5	3
BKZ bock-size b to break LWE [1]	200	340	485	595
Формула (9)	220	373	525	639

Як видно з табл. 1, формула (9) приводить до більших значень в порівнянні з [1]. Поряд з тим, значення довжини блоку, отримані за цій формулою, базуються тільки на розгляді однієї з двох атак на задачу LWE. Отже, з [2, п. 5.4], збільшення k та l на 1 приводить до збільшення стійкості приблизно на 30 бітів. Отже, виходячи з [1, табл. 1], де для $(k, l) = (6, 5)$,

$\eta = 3$ зазначено оцінку стійкості 2^{174}, можна припустити, що для стійкості 2^{256} достатньо покласти $(k, l) = (9, 8)$, $\eta = 3$.

6. Особливості гешування в алгоритмі ЦП Dilithium

Нехай B_h позначає сукупність елементів R, які мають коефіцієнти h, які є або −1, або 1, а інші є 0. Маємо $|B_h| = 2^h \cdot \binom{n}{h}$. Для нашого механізму ЕП необхідна криптографічна геш-функція, яка використовується для гешування полінома *c*. Алгоритм, який будемо використовувати для створення випадкового елементу у B_{60}, іноді називають «виворотною» версією Fisher-Yates перемішування [1] і його опис високого рівня наведено на рис. 3 (як правило, алгоритм повинен починатися з *i*=0, але оскільки існує 196 0, перші 195 ітерацій будуть просто встановлювати компоненти **c** у 0).

SampleInBall
01 Initialize $\mathbf{c} = c_0 c_1 \dots c_{255} = 00 \dots 0$
02 **for** $i := 196$ to 255
03 $j \leftarrow \{0, 1, \dots, i\}$
04 $s \leftarrow \{0, 1\}$
05 $c_i := c_j$
06 $c_j := (-1)^s$
07 **return c**

Рис. 3. Створення випадкового 256-елементного масиву з 60 ±1 та 196 0

Визначаємо роботу функції $\mathrm{H} : \mu \| \mathbf{w}_1 \mapsto c \in B_{60}$, описану на рис. 3, так як вона використовується в механізмі ЕП. Спочатку H гешує 48 байт μ, а потім відразу ж йдуть 128*k* байт для упаковки бітів представлення $\mathbf{w}_1$ в SHAKE-256. Протягом цих операцій функція стискає SHAKE-256, щоб отримати потік випадкових байтів змінної довжини. Перші 60 бітів у перших 8 байтах цього випадкового потоку інтерпретуються як 60 випадкових знакових бітів $s_i \in \{0, 1\}$, *i*=0,...,59. Інші 4 біта відкидаються. Далі H використовується для обчислення *c*. У кожній ітерації циклу для циклу використовується відхилення вибірки на елементи з {0,...,255}, доки він не отримується $j \in \{0, \dots, i\}$. Елемент у {0,...,255} отримується шляхом інтерпретації наступного байта випадкового потоку з SHAKE-256 як число в цьому наборі. Для підпису *s* використовується відповідний s_{i-196}.

Проведені дослідження показали, що сучасні та перспективні функції гешування для їх застосування в алгоритмах ЕП мають неодмінно відповідати таким вимогам стійкості:

- складність знаходження колізії $C_{col} \geq 2^{hlen/2}$;
- складність відновлення прообразу *M* $C_{preim} \geq 2^{hlen}$;
- складність знаходження іншого прообразу $C_{\mathrm{sec}\_preim} \geq 2^{hlen}$;
- складність знаходження колізії, усіченої на *ht* символів $C_{tr\_col} \geq 2^{(hlen-ht)/2}$.

Наведені вимоги дозволяють ввести безумовні критерії оцінки функцій гешування для криптографічних додатків.

7. Обґрунтування можливості застосування національного стандарту України в якості геш-функції для алгоритму Dilithium

Дві основні операції, що складають практично всю процедуру підпису та перевірки в алгоритмі Dilithium, – це розширення XOF (використовується SHAKE-128 і SHAKE-256) та множення у кільці полінома $R_q = Z_q[X]/(X^n + 1)$. Тому реалізації алгоритму на національному рівні повинні оптимізувати ці операції та працювати за постійний час.

Згідно [18] стандарт ДСТУ 7564:2014 визначає національний стандарт на функцію гешування. Функція гешування ДСТУ 7564:2014 забезпечує обчислення геш-значення з довжинами від 8 до 512 біт з кроком у 8 біт. Режим роботи для формування геш-значення довжиною n біт позначається як «Купина-n». Основними режимами роботи функції гешування, що рекомендуються до застосування, є «Купина-256», «Купина-384» і «Купина-512». Вона забезпечує обчислення геш-значення для повідомлення, що складається з бітової послідовності довжини від 0 біт (порожній рядок) до $2^{96}-1$ біт. При формуванні геш-значення повідомлення доповнюється, далі поділяється на l-бітні блоки $m_0,\ldots,m_t$, після чого виконується обробка кожного блоку шляхом ітеративного виконання функції стиснення ϕ. При обчисленнях формуються значення $h_i=\phi\left(h_{i-1},m_i\right)$, де $i=1,\ldots,t$, а також початкове значення $h_0=IV$. Після обробки останнього блоку повідомлення обчислюється результуюче геш-значення, тобто

$$H\left(M\right)=\Omega(h_t), \tag{10}$$

де Ω – завершальне перетворення, що повертає n – бітне значення, кратне 8 ($n\le\frac{l}{2}$).

Так як геш-функція, що описана в стандарті ДСТУ 7564:2014, забезпечує необхідні для алгоритму довжини геш-значення та є колізійно-стійкою[24], а також *відповідає усім безумовним критеріям* [26], нами пропонується її використання в якості основної функції гешування у новому постквантовому національному стандарті ЕП України.

Висновки

1. За результатами першого етапу конкурсу авторами проекту Crystals-Dilithium було запропоновано пропозиції щодо його удосконалення у плані побудови системних параметрів, що забезпечать більш високі рівні стійкості [3].

2. Актуальною є проблема обґрунтування необхідності та розробки удосконаленої версії ЕП Dilithium, що може забезпечувати в постквантовий період 5, 6 та 7 рівні захищеності від найбільш загрозливих атак [15, 16].

3. Різниця між версіями на першому і другому етапах конкурсу NIST полягає у тому, що початкова ентропія удосконаленого ЕП, тобто у рандомізованій версії, може встановлюватись випадковим чином, тоді як у 1-й, детермінованій версії, вона визначалась тільки у вигляді геш-значення ключа та безпосередньо повідомлення *M*, що підписується.

4. В механізмі Dilithium при генеруванні ключів та загальних параметрів використовуються засоби з рівноймовірним розподілом. Також такі операції як множення поліномів та їх округлення легко реалізуються з однаковою часовою складністю. Вказане забезпечує захист від атак сторонніми каналами на основі різної складності множення поліномів тощо.

5. Використання секретних послідовностей на основі дискретного нормального гаусового розподілу є надзвичайно нетривіальним і може легко призвести до незахищених реалізацій [22]. Це пояснюється тим, що хоча дуже «обережна» реалізація може запобігти подібним атакам, але неможливо припускати, що загальнодоступний механізм типу Crystals-Dilithium, що містить багато тонкощів, завжди буде досконало реалізований.

6. В механізмі Crystals-Dilithium зроблена спроба мінімізувати суму довжин відкритого ключа та ЕП. Внаслідок цього механізм Dilithium має, у порівнянні з іншими механізмами на алгебраїчних решітках, найменше поєднання розміру підпису та розмірів відкритих ключів, з однаковими рівнями безпеки.

7. В механізмі Crystals-Dilithium є можливість оперативної зміни рівня безпеки. Модульність реалізації та можливість оперативної зміни рівня безпеки пов’язані з тим, що по суті в механізмі Crystals-Dilithium виконуються всього дві операції. При генеруванні ключів та па-

раметрів виконується операція розгортання ключів та параметрів, а також множення у кільці полінома $\mathbb{Z}_q[X]/(X^n+1)$.

8. Безпеку механізму ЕП, що наведено на рис. 1, можна довести в моделі випадкового оракула (ROM), ґрунтуючись на складності двох проблем. Перша – це стандартна задача LWE над кільцями багаточленів, в якій пропонується відрізняти (A, $t:=As_1+s_2$) від (A, u), де u – рівномірно випадкове.

9. Механізм Crystals-Dilithium може бути взятий за основу, одним із кандидатів для розробки національного стандарту ЕП з використанням стандартизованих в Україні криптографічних алгоритмів, таких як функція гешування, наведена у ДСТУ 7564:2014 [27].

Список літератури:

1. Donald Knuth The Art of Computer Programming, volume 2. Addison-Wesley, 3 edition, 1997. P. 145.

2. Lyubachevsky V., Ducas L., Kiltz E. [et all] CRYSTALS–Dilithium. Techn. rep. NIST (2017) / https://crc.nist.gov./projects/post-quantum-cryptogtraphy/round-1-submissions.

3. Bos J.W., Costello C., Ducas L. [et all] Frodo: take of the ring! Practical, quantum-secure key exchange from LWE // Proc. of ACM CCS 16, ACM Press, Okt. 2006. P. 1006-1018.

4. Albrecht M.R., Goepfert F., Virdia F., Wunderer T. Revisiting the expected cost of solwing uSVP and applications to LWE // Cryptology ePrint Archive, Report 2017/815, http://eprint.iacr.org/2017/815.

5. Rueckert M., Schneider M. Estimating the security of lattice-based cryptosystems // Cryptology ePrint Archive, Report 2010/137, http://eprint.iacr.org/2010/0137.

6. Albrecht M.R., Player R., Scott S. On the concrete hardness of learning with errors // Cryptology ePrint Archive, Report 2015/046, http://eprint.iacr.org/2015/046.

7. Rachel Player. Parameter selectionin lattice-based cryptography.

8. Gottfried Herold, Elena Kirshanova, and Alexander May. On the asymptoticcomplexity of solving LWE.Designs, Codes and Cryptography, Jan 2017.

9. Shi Bai and Steven D. Galbraith. Lattice decoding attacks on binary LWE.In Willy Susilo and Yi Mu, editors, ACISP 14, vol. 8544 of LNCS, p. 322–337. Springer, Heidelberg, July 2016.

10. Sanjeev Arora and Rong Ge. New algorithms for learning in presence of errors.In Luca Aceto, Monika Henzinger, and Jiri Sgall, editors, ICALP 2011, Part I, vol. 6755 of LNCS, p. 403–415. Springer, Heidelberg, July 2011.

11. Martin R. Albrecht, Carlos Cid, Jean-Charles Faugère, and Ludovic Perret. Algebraic algorithms for LWE. Cryptology ePrint Archive, Report 2014/1018,2014. http://eprint.iacr.org/2014/1018.

12. Martin R. Albrecht, Carlos Cid, Jean-Charles Faugère, Robert Fitzpatrick, andLudovic Perret. On the complexity of the BKW algorithm on LWE. Designs, Codes and Cryptography, 74:325–354, 2015.

13. Martin R. Albrecht, Jean-Charles Faugère, Robert Fitzpatrick, and Ludovic Perret. Lazy modulus switching for the BKW algorithm on LWE. In Hugo Krawczyk, editor,PKC 2014, vol. 8383 of LNCS, p. 429–445. Springer,Heidelberg, March 2014.

14. Richard Lindner and Chris Peikert. Better key sizes (and attacks) for LWE-based encryption. In Aggelos Kiayias, editor,CT-RSA 2011, vol. 6558 ofLNCS, p. 319–339. Springer, Heidelberg, February 2011.

15. Avrim Blum, Adam Kalai and Hal Wasserman. Noise-tolerant learning, the parity problem, and the statistical query model // Journal of the ACM,50(4):506–519, July 2003.

16. НДР «Визначення напрямків розвитку математичних методів та дослідження перспектив їх застосування для створення сучасних та перспективних криптографічних алгоритмів та протоколів» (Шифр «Скіл»). Т. 9. «Проект стандарту електронного підпису на алгребраїчній решітці для постквантового періоду». Харків, 2018. 127 с.

17. Daniel J. Bernstein, Daira Hopwood, Andreas Hülsing, Tanja Lange, Ruben Niederhagen, Louiza Papachristodoulou, Michael Schneider, Peter Schwabe and Zooko Wilcox-O'Hearn. SPHINCS: Practical stateless hash-based signatures. In Elisabeth Oswald and Marc Fischlin, editors, EUROCRYPT 2015, Part I, vol. 9056 of LNCS, p. 368–397, Sofia, Bulgaria, April 26–30, 2015. Springer, Heidelberg, Germany.

18. ДСТУ 8961:2019 Інформаційні технології. Криптографічний захист інформації. Алгоритми асиметричного шифрування та інкапсуляції ключів.

19. Ducas L., Lepoint T., Lyubachevsky V. [et all] CRYSTALS – Dilithium: digital sygnatures from module lattices / https://cryptojedi.org/papers/dilithium-20170617.pdf.

20. Alkim E., Ducas L., Poeppelmann T., Schwabe P. Post-quantum key exchange – a new hope / http://cryptojedi.org/papers/#newhope, 2016.

21. Bos J.W., Costello C., Ducas L. [et all]. Frodo: take of the ring! Practical, quantum-secure key exchange from LWE // Proc. of ACM CCS 16, ACM Press, Okt. 2006, P. 1006-1018.

22. Albrecht M.R., Player R., Scott S. On the concrete hardness of learning with errors // Cryptology ePrint Archive, Report 2015/046, http://eprint.iacr.org/2015/046.

23. Post-Quantum Cryptography [Electronic resource]. Access mode: https://csrc.nist.gov/projects/post-quantum-cryptography.

24. Горбенко Ю. І. Аналіз шляхів розвитку криптографії після появи квантових комп'ютерів / Ю. І. Горбенко. Р. С. Ганзя // Комп'ютерні системи та мережі : Вісник нац. ун-ту «Львівська політехніка». 2014. № 806. С. 40–49.

25. Damien Stehlé and Ron Steinfeld. Making NTRU as secure as worst-case problems over ideal lattices. In Kenneth G. Paterson, editor, EUROCRYPT 2011, volume 6632 of LNCS, pages 27–47, Tallinn, Estonia, May 15–19, 2011. Springer, Heidelberg, Germany.

26. Горбенко І. Д. Постквантова криптографія та механізми її реалізації / І. Д. Горбенко, О. О. Кузнєцов, О. В. Потій, Ю. І. Горбенко, Р. С. Ганзя, В. А. Пономар // Радіотехніка. 2016. Вип. 186. С. 32-52.

27. ДСТУ 7564:2014 Інформаційні технології. Криптографічний захист інформації. Функція гешування.

Харківський національний
університет імені В. Н. Каразіна;
АТ «Інститут інформаційних технологій»

Надійшла до редколегії 07.01.2020