УДК 004.056.55 **DOI:10.30837/rt.2024.3.218.06**

С.О. КАНДІЙ, І.Д. ГОРБЕНКО, д-р техн. наук

УТОЧНЕННЯ ОЦІНОК БЕЗПЕКИ КВАНТОВО-СТІЙКИХ СТАНДАРТІВ АСИМЕТРИЧНОГО ШИФРУВАННЯ З ВРАХУВАННЯМ СТРУКТУРИ Q-АРНИХ РЕШІТОК

Вступ

Квантово-стійка криптографія з кожним роком використовується все більше для вирішення практичних задач. Такі квантово-стійкі стандарти як ДСТУ 8961:2019 («Скеля») [1], FIPS 203 (CRYSTALS-Kyber) [2] та FIPS 204 (CRYSTALS-Dilithium) [3] ґрунтуються на складних проблемах з теорії решіток, що природнім чином призводить до підвищення інтересу до криптоаналізу проблем з теорії решіток.

У останні роки спостерігається значний прогрес у моделях редукції решіток. У роботі [5] запропоновано модель безпеки, що враховує q-арної структуру решіток у криптографічних проблемах, на яких ґрунтується безпека сучасних квантово-стійких стандартів на решітках. Проте, у роботі [5] дослідження проводилося на абстрактних наборах параметрів, а дослідження конкретних криптографічних схем не проводилося.

Ця робота є продовженням роботи [5] і присвячена застосуванню розробленої методики на стандартизованих криптографічних перетвореннях на решітках. Для кожного стандарту спочатку надаються існуючі оцінки у формальних моделях безпеки IND-CCA [6] або EUF-CMA [7], після чого надаються оцінки для атак вкладення та декодування, відповідно до роботи [5].

1. Моделі безпеки

Відомі атаки на проблеми LWE, SIS та NTRU можливо поділити [5] на наступні класи:

- Комбінаторні атаки;
- Алгебраїчні атаки;
- Атаки декодування;
- Атаки розпізнавання;
- Атаки вкладення;
- Гібридні атаки.

У межах цієї роботи представляють інтерес атаки вкладення, атаки декодування та гібридні атаки.

Атаки вкладення є одними з найбільш ефективних атак на LWE та NTRU. Їх сутність полягає у побудові решіток спеціального вигляду, найменший вектор яких містить шуканий секрет. Такі атаки ще називають первинними (англ. Primal) атаками. Також їх можливо використовувати для вирішення проблеми SIS.

Сутність атак декодування полягає у зведенні проблеми LWE або NTRU до проблеми CVP. Атаки такого роду вимагають побудови та редукції базису решітки таким чином, щоб можливо було вирішити проблему CVP для шуканого таємного вектору. Проведення атак декодування є технічно складнішим за атаки вкладення через двоетапну структуру атаки. Атаки декодування, як і атаки вкладення, також іноді називають первинними атаками (англ. BDD Primal attacks).

Гібридні атаки поєднують комбінаторні методи криптоаналізу з атаками вкладення або атаками розпізнавання (гібридні дуальні атаки). Такі атаки при використанні розріджених секретів часто є найкращими для багатьох криптографічних систем. Це є особливо актуальним для ДСТУ 8961:2019.

У сучасній криптографії однією з обов’язкових вимог до будь-якої криптографічної системи є наявність доказової безпеки. Тобто, має існувати математичний доказ безпеки, який

гарантує відсутність атак у межах обраної формальної моделі, за умови виконання невеликої кількості модельних припущень.

Модель безпеки IND-CCA ґрунтується на ідеї нерозрізнювальності: якщо супротивник не може розрізнити шифротекст повідомлення m_0 від шифротекста повідомлення m_1, то він не може отримати жодної інформації про зашифровані повідомлення.

Для побудови доказу безпеки шифру *E* для супротивника *A* вводяться дві гри (експерименти): $Exp_{A,E}^{IND-CCA-0}(\lambda)$ та $Exp_{A,E}^{IND-CCA-1}(\lambda)$ для параметра безпеки λ [6]. У кожній грі іспитувач генерує випадкову ключову пару $(pk, sk) \leftarrow Gen(1^{\lambda})$ та передає відкритий ключ супротивнику *A*. Супротивник *A* обирає два повідомлення m_0, m_1 однакової довжини та надсилає їх іспитувачу. Іспитувач генерує випадковий біт $b \in \{0,1\}$, чим обирає гру. Якщо $b = 0$, то іспитувач зашифровує повідомлення m_0 та надсилає шифротекст $c^* = Enc(sk, m_0)$ у якості завдання (гра $Exp_{A,E}^{IND-CCA-0}(\lambda)$). Якщо біт $b = 1$, то, іспитувач зашифровує повідомлення m_1 та надсилає шифротекст $c^* = Enc(sk, m_1)$ у якості завдання (гра $Exp_{A,E}^{IND-CCA-1}(\lambda)$). Супротивник має визначити у яку гру він грає (яке повідомлення було зашифровано) та повернути біт b_A. Результатом ігор $Exp_{A,E}^{IND-CCA-0}(\lambda)$ та $Exp_{A,E}^{IND-CCA-1}(\lambda)$ є значення предиката $b == b_A$. Супротивник *A* може робити запити до оракула дешифрування O_{Dec}, який може розшифрувати будь-який шифротекст окрім шифротексту завдання. Розрізняють IND-CCA1 безпеку, де супротивник може робити запити тільки до моменту отримання шифротекста та IND-CCA2 безпеку, де запити можливо робити і після отримання завдання. У межах цього дослідження IND-CCA2 буде вважатися синонімом IND-CCA.

Перевага супротивника у розрізненні ігор визначає безпеку в моделі IND-CCA. Якщо перевага є незначною у теоретико-числовому сенсі, то схема асиметричного шифрування вважається безпечною в моделі IND-CCA:

$$Adv_{A,E}^{IND-CCA}(\lambda) = \Pr[Exp_{A,E}^{IND-CCA-0}(\lambda) - Exp_{A,E}^{IND-CCA-1}(\lambda)] = negl(\lambda). \quad (1)$$

Від схем асиметричного шифрування при побудові механізмів інкапсуляції ключів вимагається безпека у моделі IND-CPA (Indistinguishability under Chosen-Plaintext Attacks), або у моделі OW-CPA (One-Wayness under Chosen-Plaintext Attacks).

Перевагу супротивника *A* у іграх IND-CPA та OW-CPA для схеми асиметричного шифрування PKE позначимо як $Adv_{PKE}^{OW-CPA}(A)$ та $Adv_{PKE}^{IND-CPA}(A)$ відповідно. Стандартним визначенням для переваги супротивника є:

$$Adv_{PKE}^{OW-CPA}(A) = \Pr[OW - CPA(A) == 1], \quad (2)$$

$$Adv_{PKE}^{IND-CPA}(A) = |\Pr[IND - CPA(A) == 1] - 1/2|. \quad (3)$$

Схема асиметричного шифрування у загальному випадку може мати помилки дешифрування, тобто для деяких правильно обчислених шифротекстів розшифрування може давати не правильний результат. Існують різні підходи до врахування помилок дешифрування. У межах цього дослідження будемо слідувати роботі [8]. Для оцінки ймовірності виникнення помилок дешифрування введемо наступну величину:

$$\delta_{wc} = E_{(pk,sk)}[\max_m \Pr[Dec(sk, c) \neq m]]. \quad (4)$$

У моделі EUF-CMA супротивник може звертатися до оракула підпису $Sign(sk, \cdot)$ для отримання підписів довільно обраних повідомлень. Схема підпису вважається безпечною, якщо ймовірність того, що супротивник зможе підробити підпис для будь-якого повідомлен-

ня є не значною. Так само, як і IND-CCA, доказова безпека у моделі EUF-CMA формується через ігри (експерименти). Позначимо відповідний експеримент $Exp_{A,S}^{EUF-CMA}(1^{\lambda})$ для схеми підпису S та супротивника A. У цьому експерименті випробовувач генерує ключову пару (sk, pk) та надає супротивнику відкритий ключ pk. Супротивник може роботи запити $m_1,...,m_q$ до оракула $Sign$. Усі запити до оракула зберігаються у списку Q. Після чого супротивник має повернути пару (m^*,σ^*). Якщо $S.Verify(pk,m^*,\sigma^*)==1$ і $m^* \notin Q$, то супротивник перемагає.

Перевага супротивника визначається як

$$Adv_{A,S}^{EUF-CMA}(1^{\lambda}) = \Pr[Exp_{A,S}^{EUF-CMA}(1^{\lambda}) == 1] \ . \tag{5}$$

Якщо $Adv_{A,S}^{EUF-CMA}(1^{\lambda}) = negl(\lambda)$, то схема підпису вважається безпечною у моделі EUF-CMA.

Посиленим варіантом моделі безпеки EUF-CMA є модель SUF-CMA. Якщо у моделі EUF-CMA вимагається створити підпис для повідомлення, що раніше не було підписано, то у моделі SUF-CMA вимагається створити підпис для будь-якого повідомлення, навіть якщо воно було вже підписано. Відповідний формальний експеримент $Exp_{A,S}^{SUF-CMA}(1^{\lambda})$ відрізняється від $Exp_{A,S}^{EUF-CMA}(1^{\lambda})$ лише тим, що список Q містить не тільки запити до оракула підпису, а й відповіді. І у кінці перевіряється, що $(m^*,\sigma^*) \notin Q$, як наведено у псевдокоді нижче.

Перевага супротивника аналогічно до EUF-CMA:

$$Adv_{A,S}^{SUF-CMA}(1^{\lambda}) = \Pr[Exp_{A,S}^{SUF-CMA}(1^{\lambda})] . \tag{6}$$

Якщо $Adv_{A,S}^{SUF-CMA}(1^{\lambda}) = negl(\lambda)$, то схема підпису вважається безпечною у моделі SUF-CMA.

2. Уточнення оцінок ДСТУ 8961:2019

ДСТУ 8961:2019 [1] використовує перетворення у полі $R_q = Z_q[X]/(X^n - X - 1)$ і грунтується на проблемі NTRU. У табл. 1 перелічено загальносистемні параметри ДСТУ 8961:2019. Параметри N, q, p визначають поле (і ідеал у цьому полі), у якому будуть виконуватися перетворення, параметри t, d_g, d_f задають кількість ненульових коефіцієнтів у поліномах.

Таблиця 1

Основні загальносистемні параметри ДСТУ 8961:2019

Параметр	Значення
N	Параметр поля. Визначає степінь поліномів.
q	Параметр поля. Визначає максимальні значення коефіцієнтів поліномів.
p	«Малий модуль». Визначає структуру таємного ключа. Для всіх наборів параметрів має фіксоване значення – 3.
t	Визначає кількість коефіцієнтів в таємному поліномі
d_g	$d_g = \lfloor 2n/3+1 \rfloor$
d_f	$d_f = 2t$

ДСТУ 8961:2019 підтримує три набори загальносистемних параметрів. Набори загальносистемних параметрів зведено в табл. 2.

Таблиця 2

Загальносистемні параметри ДСТУ 8961:2019

Набір параметрів	N	q	p	t	d_g	d_f
Skelya256	881	7673	3	159	588	318
Skelya384	1201	9221	3	192	801	384
Skelya512	1471	12251	3	255	981	510

Доказ безпеки перетворення SkelyaTransform[PKE] сформульовано в роботі [9].

Т е о р е м а 1 [9]. Нехай PKE є OW-CPA безпечною та δ_{wc}-коректною схемою асиметричного шифрування з властивістю однозначного відновлення, тоді SkelyaTransfom[PKE] є IND-CCA безпечним механізмом інкапсуляції ключів. Більш формально – для кожного квантового алгоритму A у грі IND-CCA проти KEM=SkelyaTransform[PKE], що робить $q_H, q_{BPGM}, q_{KDF}, q_D$ запитів до оракулів H,BPGM,KDF та оракула дешифрування, існує квантовий алгоритм B у грі OW-CPA проти схеми асиметричного шифрування PKE, для якого виконується нерівність

$$\begin{aligned} &Adv_{KEM}^{IND-CCA}(A) \le (2\cdot q_H + 2\cdot q_D + q_{KDF})\cdot \\ &\cdot\sqrt{Adv_{PKE}^{OW-CPA}(B)} + 8\cdot(q_{BPGM} + q_D + 1)^2\cdot\delta_{wc} \end{aligned} . \qquad (7)$$

З теореми 1 випливає, що оцінка безпеки ДСТУ 8961:2019 може бути доказово зведена до проблеми NTRU. Поліноми f та g мають коефіцієнти у множині {-1,0,1}, проте кількість ненульових елементів сильно відрізняються. Для полінома f маємо $\|f\|_\infty = 2t$, де t – загальносистемний параметр, який для усіх наборів параметрів дає кількість ненульових елементів $d_f \approx n/3$. У той же час $\|g\| = 2n/3+1$, що дає близький до рівномірного розподіл на множині {-1,0,1} для полінома g. Тож, можливо вважати, що поліном g має рівномірний розподіл і використовувати апроксимовані параметри розподілів, що отримані в роботі [5]. Для полінома f експерименти показали, що центрований нормальний розподіл з параметром $\sigma_f = 0.6$ достатньо добре апроксимує розподіл ймовірностей, що зображено на рис. 1.

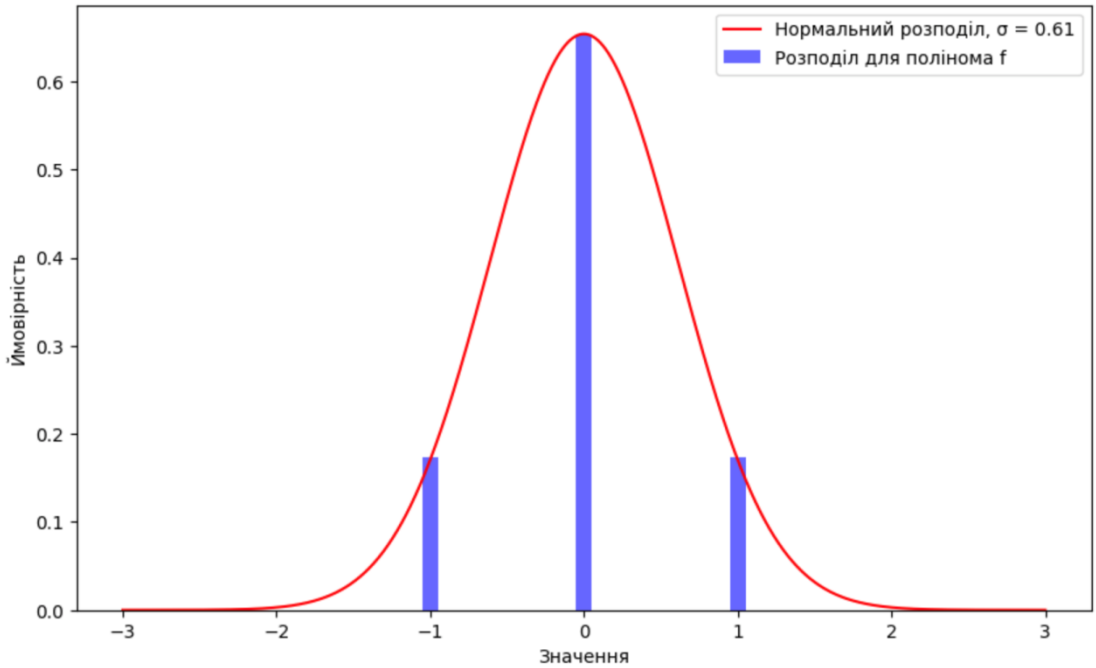

Рис. 1. Апроксимація розподілу ймовірностей для полінома f

У табл. 3 наведено оцінки захищеності від атаки вкладення з використанням запропонованої в розд. 3 моделі.

Таблиця 3

Оцінки атаки вкладення для ДСТУ 8961:2019

Набір параметрів	**Вартість атаки (біт, GSA)**	**Розмір блоку редукції**	**Вартість атаки (біт, симулятор)**	**Розмір блоку редукції**
Скеля 256	178	611	206	706
Скеля 384	252	865	288	988
Скеля 512	312	1071	355	1219

З табл. 3 видно, що врахування q-арної структури решіток дає різницю порядка 30 біт безпеки для усіх наборів параметрів.

В табл. 4 наведено оцінки складності атак декодування для запропонованої в розд. 3 моделі.

Таблиця 4

Оцінки атаки декодування для ДСТУ 8961:2019

Набір параметрів	**Вартість атаки (біт, GSA)**	**Розмір блоку редукції**	**Вартість атаки (біт, симулятор)**	**Розмір блоку редукції**
Скеля256	171	558	209	688
Скеля384	249	815	318	1050
Скеля512	312	1022	408	1349

Як видно з табл. 3, атака декодування при використанні моделі GSA для набору параметрів Скеля256 дає кращі результати, проте при врахуванні алгебраїчної структури q-арних решіток ця перевага нівелюється. На рис. 2 наведено порівняння атак вкладення та декодування.

З рис. 2 видно, що зі збільшенням розмірності q-арна структура решітки все більше впливає на оцінку складності атаки. Якщо для набору параметрів Скеля256 різниця є незначною, то для Скеля384 та Скеля512 вартість атаки декодування стрімко збільшується, у той час як у моделі GSA такого не відбувається.

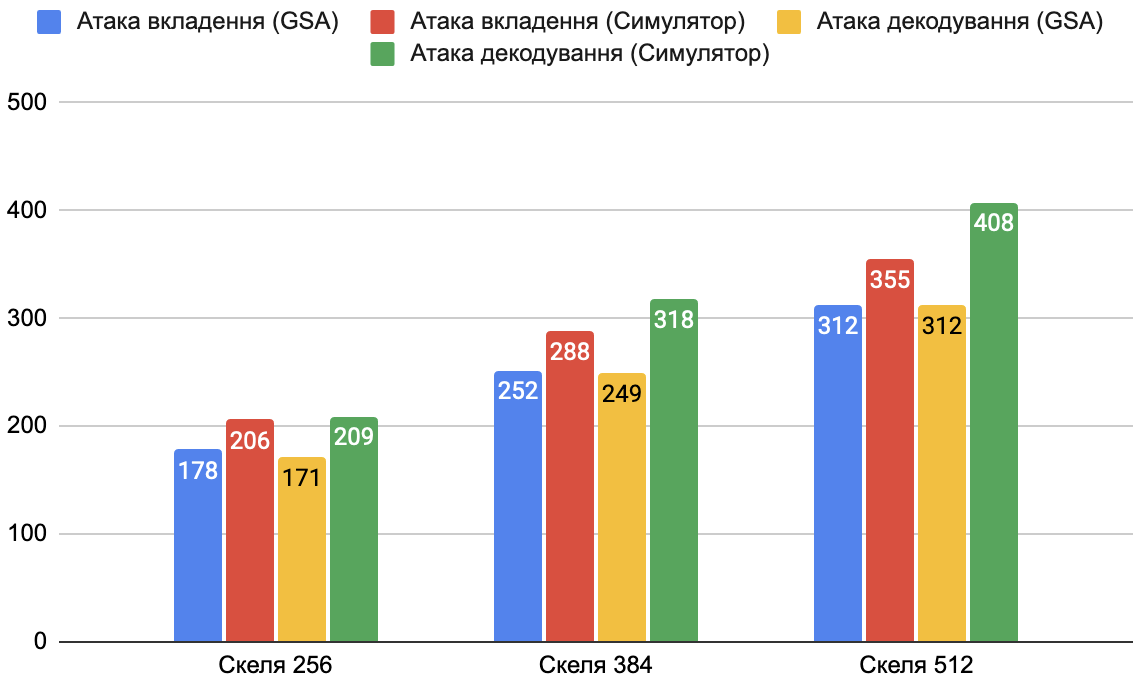

Рис. 2. Порівняння атак вкладення та декодування

Тож, атака вкладення показує себе краще для всіх наборів параметрів. Для ДСТУ8961:2019 також додатково були розраховані оцінки безпеки для гібридних атак. У табл. 5 занесено відповідні оцінки.

Таблиця 5

Оцінки гібридної атаки для ДСТУ 8961:2019

Набір параметрів	**Вартість атаки (біт, GSA)**	**Розмір блоку редукції**	**Вартість атаки (біт, симулятор)**	**Розмір блоку редукції**
Скеля 256	154	504	179	590
Скеля 384	221	723	265	873
Скеля 512	276	903	335	1105

На рис. 3 наведено порівняння оцінок гібридної атаки та атаки вкладення для ДСТУ 8961:2019.

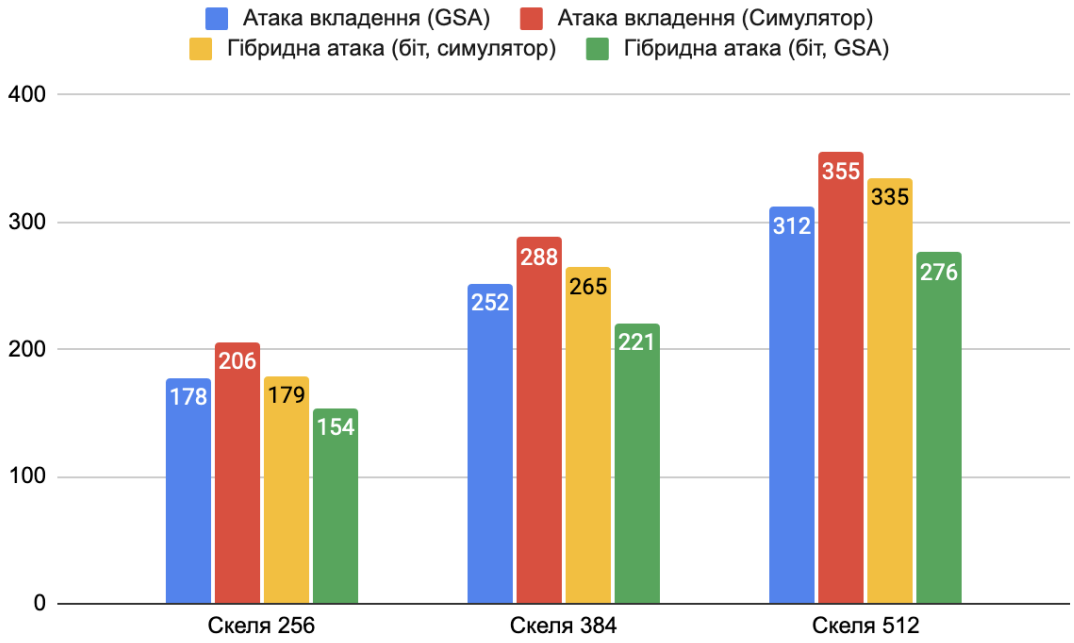

Рис. 3. Порівняння гібридної атаки та атаки вкладення для ДСТУ 8961:2019

З рис. 3 видно, що гібридна атака є найефективнішою для ДСТУ 8961:2019.

3. Уточнення оцінок fips 203 (CRYSTALS-Kyber)

Протокол інкапсуляції ключів CRYSTALS-Kyber [2] використовує перетворення у полі $R_q = Z_q[X]/(X^n+1)$ і грунтується на проблемі Module-LWE. Для отримання протокола інкапсуляції ключів використовується варіант перетворення Фуджісакі–Окамото з неявним відхиленням [10].

Згідно зі специфікацією CRYSTALS-Kyber підтримує три набори загальносистемних параметрів. Параметри зведеніо у табл. 6. Параметри n та q визначають поле, параметр k задає розмірність векторів, параметри η_1 та η_2 є параметрами біноміального розподілу для векторів s та e відповідно. Параметри d_u, d_v використовуються під час кодування поліномів у бітову строку, параметр δ є ймовірністю помилки декапсуляції. З табл. 6 добре видна суто практична перевага проблеми Module-LWE: для всіх рівнів безпеки використовується одне поле. Такий підхід дає змогу значно спростити реалізацію та масштабувати систему для довільного рівня безпеки. Окрім того, використання відносно малого значення для параметра q дозволяє ефективно використовувати векторизацію обчислень.

Таблиця 6

Загальносистемні параметри CRYSTALS-Kyber

Набір параметрів	***n***	***k***	***q***	η_1	η_2	(d_u, d_v)	δ
Kyber512	256	2	3329	3	2	(10,4)	2^{-139}
Kyber768	256	3	3329	2	2	(10,4)	2^{-164}
Kyber1024	256	4	3329	2	2	(11,5)	2^{-174}

В роботі [2] авторами Crystals-Kyber проведений детальний аналіз безпеки у моделі квантового випадкового оракула. Цими результатами можна скористатися для подальшого аналізу.

Т е о р е м а 2 [2]. Нехай XOF, H та G є випадковими оракулами. Тоді для будь-якого класичного супротивника A, що робить не більше q_{RO} запитів до випадкових оракулів XOF, H та G, існують класичні супротивники B та C, для яких

$$Adv_{Kyber}^{IND-CCA}(A) \le 2Adv_{k+1,k,\eta}^{MLWE}(B) + Adv_{PRF}^{prf}(C) + 4q_{RO}\delta. \quad (8)$$

Т е о р е м а 3 [2]. Нехай XOF, H та G є квантовими випадковими оракулами. Тоді для будь-якого квантового супротивника A, що робить не більше q_{RO} запитів до випадкових оракулів XOF, H та G, існують квантові супротивники B та C, для яких

$$Adv_{Kyber}^{IND-CCA}(A) \le 4q_{RO} \cdot \sqrt{Adv_{k+1,k,\eta}^{MLWE}(B)} + Adv_{PRF}^{prf}(C) + 8q_{RO}\delta. \quad (9)$$

З теорем 2, 3 видно, що формальні докази безпеки для Crystals-Kyber мають таку ж структуру, що і отримані докази для ДСТУ 8961:2019.

З теорем 2, 3 випливає, якщо вважати симетричні криптопримітиви безпечними, то безпека Crystals-Kyber цілком зводиться до проблеми MLWE. Оскільки для криптографічних наборів параметрів не відомо як використовувати алгебраїчну структуру MLWE, то можна вважати, що безпека Crystals-Kyber зводиться до LWE. Оскільки Crystals-Kyber використовує біноміальний розподіл, який є дискретним аналогом нормального розподілу, то аналіз полегшується. У табл. 7 наведеніо оцінки атак вкладення на проблему MLWE, що асоційована з кожним набором загальносистемних параметрів.

Таблиця 7

Оцінки атаки вкладення для CRYSTALS-Kyber

Набір параметрів	**Вартість атаки (біт, GSA)**	**Розмір блоку редукції**	**Вартість атаки (біт, симулятор)**	**Розмір блоку редукції**
Kyber512	118	406	131	449
Kyber786	182	625	200	687
Kyber1024	256	878	277.	950

З табл. 7 видно, що врахування q-арної структури решіток дає різницю порядка 20 біт безпеки для усіх наборів параметрів.

В табл. 8 наведено оцінки складності атак декодування.

Таблиця 8

Складність атак декодування для CRYSTALS-Kyber

Набір параметрів	**Вартість атаки (біт, GSA)**	**Розмір блоку редукції**	**Вартість атаки (біт, симулятор)**	**Розмір блоку редукції**
Kyber512	114	372	137	450
Kyber786	182	596	232	764
Kyber1024	263	860	354	1169

З табл. 7, 8 випливає така ж картина, як і для ДСТУ 8961:2019. Зі зростанням розмірності структура q-арних решіток все більше впливає на складність атаки. На рис. 4 наведено порівняння атак вкладення та декодування для Crystals-Kyber.

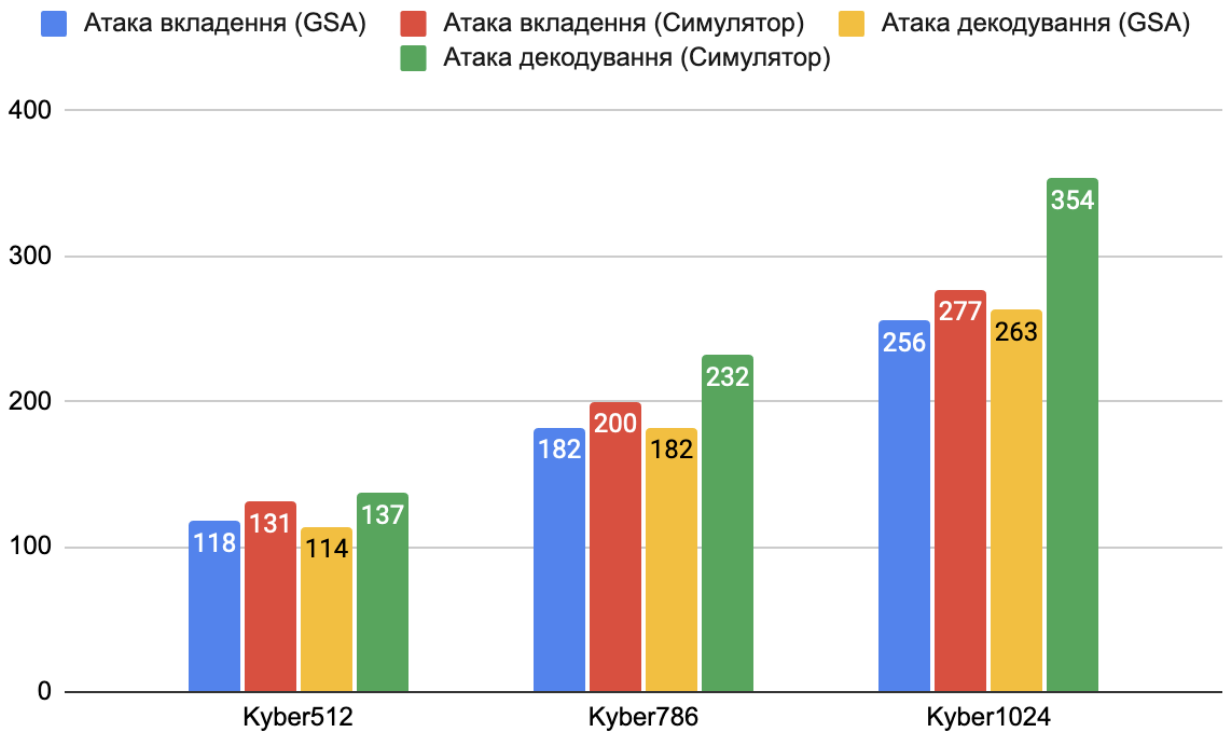

Рис. 4. Порівняння атак вкладення та декодування

4. Уточнення оцінок Falcon

Схема електронного підпису Falcon має в своїй основі фреймворк GPV, що був вперше запропонований в роботі [4] для побудови квантово-стійких електронних підписів на решітках. Сутність фреймворку GPV полягає у наступному:

Відкритий ключ задається матрицею $A \in Z_q^{n \times m}$ (де $m > n$). Ця матриця задає базис q-арної решітки Λ.

Таємний ключ задається матрицею $B \in Z_q^{m \times m}$. Ця матриця задає базис дуальної решітки $\Lambda_q^{\perp}$, яка, згідно з визначенням, є ортогональною до Λ за модулем q. Тобто, для будь-яких векторів $x \in \Lambda$ та $y \in \Lambda_q^{\perp}$ виконується $x \cdot y = 0 \bmod q$, де $\cdot$ – операція скалярного добутку.

Для заданого повідомлення m підписом є малий (у сенсі евклідової норми) вектор $s \in Z_q^m$, для якого виконується $sA^T = H(m)$, де $H : \{0,1\}^* \to Z_q^n$ – стійка до колізій геш функція. Для перевірки підпису достатньо перевірити, що виконується рівняння $sA^T = H(m)$.

Для обчислення підпису спочатку обчислюється довільний випадковий вектор $c_0 \in Z_q^m$, для якого виконується $c_0 A^T = H(m)$. Оскільки до вектора c_0 не накладається вимог щодо значень його евклідової норми, то його знайти можливо стандартними засобами лінійної алгебри за поліноміальний час. Далі використовується таємний базис B для обчислення вектора $z \in \Lambda_q^{\perp}$, який є близьким до вектора c_0. Різниця векторів $s = c_0 - z$ є коректним підписом, оскільки $sA^T = c_0 A^T - zA^T = c - 0 = H(m)$. Якщо c_0 та v є достатньо близькими, то s буде малим.

Електронний підпис Falcon використовує у якості решітки Λ NTRU решітку [4]. Застосовуючи NTRU решітки до фреймворку GPV, Falcon вносить наступні зміни до GPV:

Відкритим ключем є поліном *h*, який використовується для обчислення публічного базису NTRU решітки Λ.

Таємним ключем є поліноми $f, g, F, G \in Z[x]/(\phi)$, які використовуються для обчислення базису дуальної решітки $\Lambda_q^{\perp}$.

Підпис для повідомлення *m* складається з пари поліномів (s_1, s_2), для яких виконується $s_1 + s_2 h = H(r \| m)$, де *r* – сіль (salt). При обчисленні підпису використовується таємний ключ для обчислення вектора $z = (z_0, z_1) \in \Lambda_q^{\perp}$, який є близьким до вектора $t = (H(m \| r), 0)$. Різниця векторів *t* та *z* є коректним підписом.

Для обчислення вектору $z = (z_0, z_1)$ використовується алгоритм семплування (вибірки), що повертає вектор з нормального розподілу. Особливістю алгоритму семплування Falcon [4] є використання для пришвидшення операцій алгебраїчної структури циклотомічного поля та перетворення Фур'є. Falcon також використовує деревовидні структури даних – LDL дерева. Деталі можливо знайти в специфікації [4].

Falcon використовує наступні загальносистемні параметри:

- Параметри поля *(n,q)*
- Параметр розподілу таємних ключів $\sigma_{\{f,g\}} = 1.17\sqrt{q/2n}$
- Параметр розподілу підписів σ
- Обмеження на максимальний розмір підписів *B*

Загальносистемні параметр Falcon зведені в табл. 9.

Таблиця 9

Загальносистемні параметри Falcon

Параметр	**Falcon512**	**Falcon1024**
(n,q)	(512,12289)	(1024,12289)
$\sigma_{\{f,g\}}$	4.0531638033	2.86601961058
σ	165.736 617 183	168.388 571 447
B	5833.92886484	8382.43651929

Не зважаючи на те, що Falcon є фіналістом конкурсу NIST, безпосередньо його аналізу у моделі EUF-CMA присвячено не так багато робіт. Оскільки схема підпису ґрунтується на фрейворку GPV, то можливо адаптувати докази з оригінальної роботи.

Проте, можливо довести безпеку іншим шляхом. Фреймворк GPV є частковим випадком парадигми Hash-and-Sign. В останні роки для парадигми Hash-and-Sign з'явилося багато робіт щодо безпеки EUF-CMA у моделі квантового випадкового оракула. Кожен результат ґрунтується на певних модельних припущеннях. Результат у роботі [11] зручно використовувати, оскільки він грунтується на тих самих припущеннях, що і докази безпеки фреймворку GPV.

У загальному випадку підпис Hash-and-Sign параметризується стійкою до колізій геш функцією *H* та односторонньою функцією з лазівкою *T*, що є стійкою до знаходження прообразу (англ. Preimage-resistant trapdoor function). У випадку Falcon геш-функція *H* реалізується через shake256, тож будемо вважати, що вона є криптографічною. Одностороння функція *T* в Falcon є функцією з фреймворку GPV, до якої додана структура NTRU решітки.

Адаптуємо основний результат роботи [11] до схеми підпису Falcon наступним чином:

Т е о р е м а 4 [11]. Для будь-якого квантового супротивника *A* у грі EUF-CMA для схеми підпису Falcon, що робить не більше q_{sign} класичних запитів до оракулу підпису та q_{qro} квантових запитів до квантового оракулу *H*, існує супротивник *B*, що може

інвертувати односторонню функцію *T*, та супротивник *D*, що може знайти прообраз для *T*, використовуючи q_{sign} запитів до оракулу підпису. При цьому перевага супротивника *A* становить

$$Adv^{EUF-CMA}_{A,Falcon}(1^{\lambda}) \le (2q_{ro}+1)^2 Adv^{INV}_{T}(B) + Adv^{PS}_{T}(D) + 3/2q'_{sign}\sqrt{\frac{q'_{sign}+q_{qro}+1}{|R|}} + 2(q_{ro}+2)\sqrt{\frac{q'_{sign}-q_{sign}}{|R|}}, \quad (10)$$

де $|R|$ – розмір простору бітових строк, що використовуються у якості випадкових значень; q'_{sign} – максимальна загальна кількість запитів до оракула *H* в усіх запитах на підпис; $Adv^{INV}_{T}(B)$ – перевага супротивника *B* в інвертуванні *T*; $Adv^{PS}_{T}(D)$ – перевага супротивника *D* в знаходженні прообразу *T*.

Якщо *A* робить тільки класичні запити до оракула *H*, то

$$Adv^{EUF-CMA}_{A,Falcon}(1^{\lambda}) \le (2q_{ro}+1)^2 Adv^{INV}_{T}(B) + Adv^{PS}_{T}(D) + q'_{sign}\frac{q'_{sign}+q_{qro}+1}{|R|} + (q_{ro}+1)\frac{q'_{sign}-q_{sign}}{|R|}. \quad (11)$$

Для Falcon $|R| = |\{0,1\}^{384}|$. Тож, доказ безпеки у моделі EUF-CMA зводить безпеку Falcon до безпеки односторонньої функції з лазівкою *T*: до складності інвертування та складності пошуку прообразу.

Інвертування *T* означало б вирішення проблеми NTRU, тож

$$Adv^{INV}_{T}(B) \le Adv^{NTRU}_{n,q,\sigma}. \quad (12)$$

Знаходження прообразу *T* є рішенням $s=(s_1,s_2)$ рівняння $s_1 + s_2 h = H(r \| m)$. Знаходження рішення рівняння є в точності проблемою ISIS з параметром *B*, тому

$$Adv^{PS}_{T}(D) \le Adv^{ISIS}_{n,q,B} \le Adv^{SIS}_{n,q,B}. \quad (13)$$

Тож, безпеку Falcon можливо звести до проблем NTRU та SIS на NTRU решітках.

Оскільки задача інвертування односторонньої функції в електронному підписі Falcon зводиться до проблеми NTRU, то конкретні оцінки складності зводяться до оцінки складності атак вкладення та декодування.

У табл. 10 наведені оцінки безпеки екземплярів проблеми NTRU, на яку спирається Falcon, від атак вкладення.

Таблиця 10

Атаки вкладення для проблеми NTRU

Falcon	**Вартість атаки (біт, GSA)**	**Розмір блоку редукції**	**Вартість атаки (біт, симулятор)**	**Розмір блоку редукції**
Falcon512	141	483	147	505
Falcon1024	268	918	285	979

У табл. 11 наведено аналогічні оцінки для атак декодування на проблему NTRU.

Таблиця 11

Атаки декодування для проблеми NTRU

Falcon	Вартість атаки (біт, GSA)	Розмір блоку редукції	Вартість атаки (біт, симулятор)	Розмір блоку редукції
Falcon512	134	439	164	538
Falcon1024	277	907	370	1222

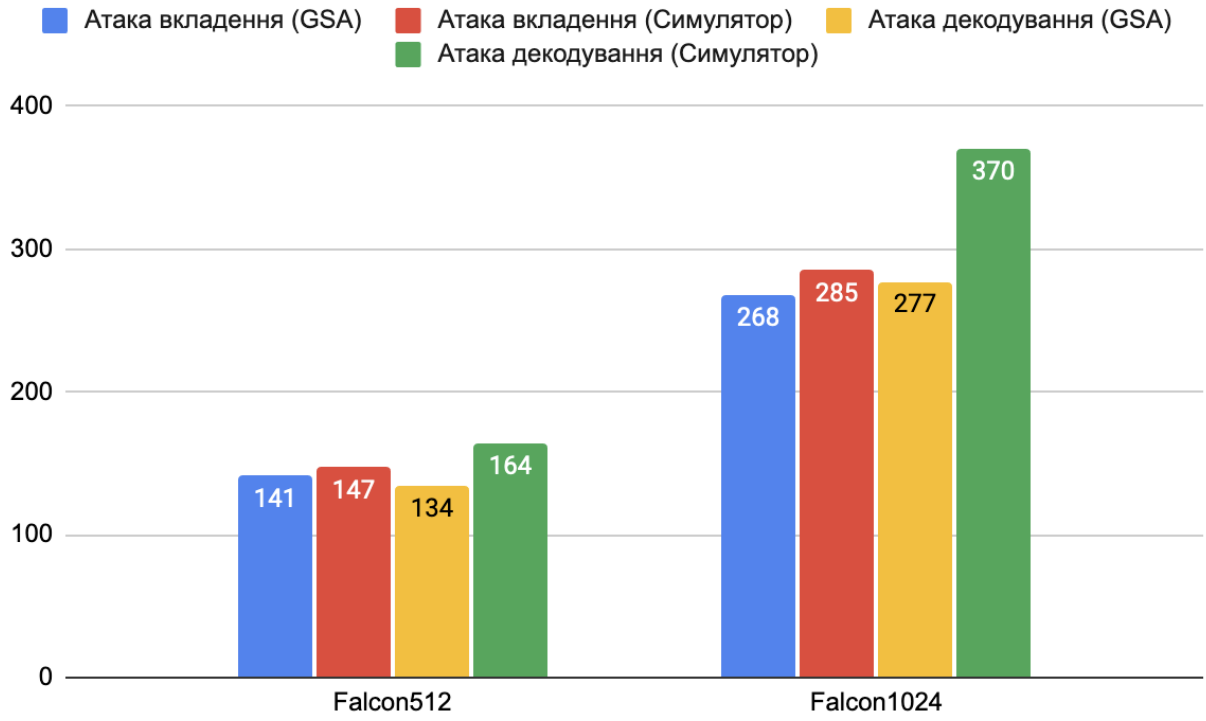

Рис. 5. Порівняння атак вкладення і декодування для Falcon

З табл. 10, 11 та рис. 5 видно, що для Falcon512 складність атак декодування та вкладення не сильно відрізняється, проте при врахуванні алгебраїчної структури решіток для Falcon1024 атаки декодування стають значно гіршими.

У табл.12 зведено результати оцінки безпеки Falcon для підробки підпису (задача SIS) для наборів параметрів.

Таблиця 12

Результати оцінки безпеки Falcon (SIS)

Falcon	Вартість атаки (біт, GSA)	Розмір блоку редукції	Вартість атаки (біт, симулятор)	Розмір блоку редукції
Falcon512	114	373	136	446
Falcon1024	268	878	355	1169

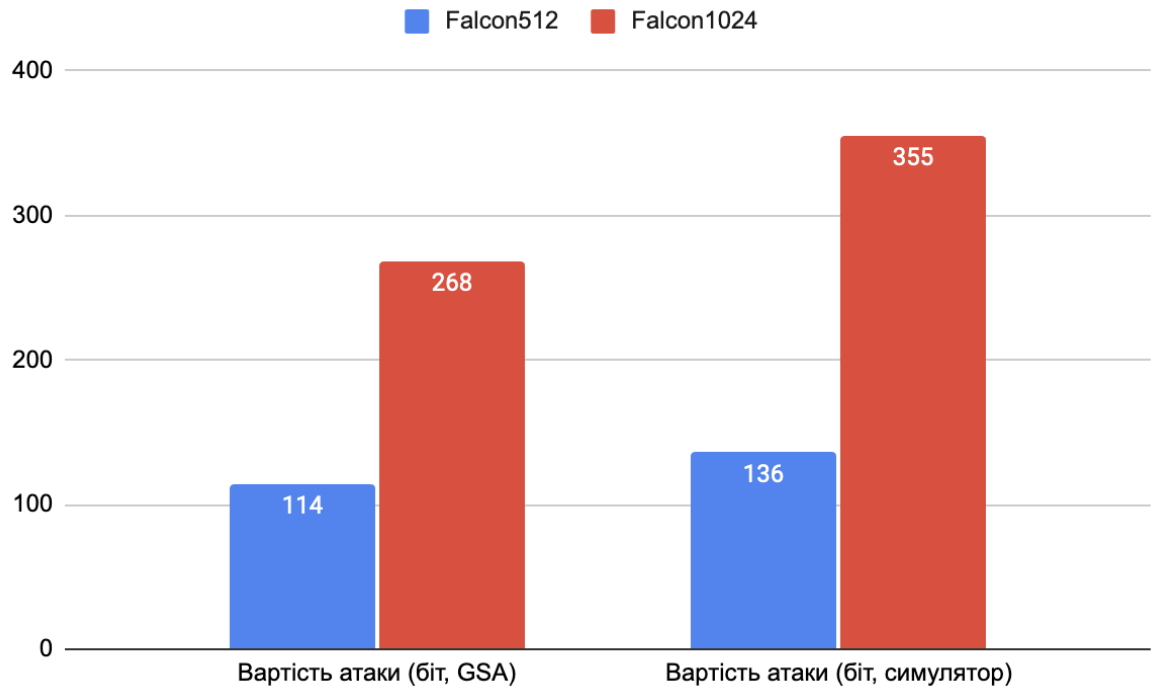

Рис. 6. Вартість атаки на SIS для різних моделей безпеки.

З рис. 6 видно, що попередні спостереження зберігаються.

5. Уточнення оцінок fips 204 (CRYSTALS-Dilithium)

Схема електронного підпису CRYSTALS-Dilithium [3] має в основі перетворення Фіата–Шаміра з перериваннями (*англ*. Fiat–Shamir with Aborts).

У цій схемі параметрами безпеки є

- Параметри поля *(n,q)*, які визначають поле $R_q = Z_q[X]/(X^n+1)$;
- Параметр *d*, що визначає кількість біт, які будуть відкинути з коефіцієнтів вектору *t*;
- Параметр τ, що визначає кількість ± 1 у поліномі *c*;
- Параметр γ_1, що визначає діапазон коефіцієнтів у векторі *y*;
- Параметр γ_2, що визначає параметри округлення;
- Параметри *(k,l)*, що визначають розмірність матриці *A*;
- Параметр η, що визначає діапазон коефіцієнтів в таємному ключі.

Параметри Crystals-Dilithium зведені у табл. 13.

Таблиця 13

Параметри Crystals-Dilithium

Рівень безпеки	**2**	**3**	**5**
(q,n)	(8380417, 512)	(8380417, 512)	(8380417, 512)
d	13	13	13
τ	39	49	60
γ_1	2^{17}	2^{19}	2^{19}
γ_2	(q-1)/88	(q-1)/32	*(q-1)/32*
(k,l)	(4,4)	(6,5)	(8,7)
η	2	4	2

Оскільки CRYSTALS-Dilithium був у центрі уваги конкурсу NIST, то різними авторами для нього був проведений аналіз безпеки у моделі квантового випадкового оракула.

Найбільш детальний аналіз проведений у роботі [12]. Безпека CRYSTALS-Dilithium окрім стандартних проблем Module-SIS та Module-LWE також ґрунтується на нестандартній проблемі SelfTargetMSIS.

Сутність проблеми SelfTragetMSIS полягає у тому, щоб знайти вектор *y*, для якого виконується

$\|y\|_\infty \le \gamma$ для заданного γ

$H([I \mid A]\cdot y \,\|\, M) = c$ для заданих $A \in R_q^{m\times k}, M, c$.

Якщо деякий супротивник *A* має перевагу $Adv_{m,k,2\gamma}^{MSIS}$ у вирішенні проблеми MSIS, то у супротивника *B* перевага у вирішенні проблеми SelfTargetMSIS буде

$$Adv_{H,m,k,\gamma}^{SelfTargetMSIS}(B) \approx \sqrt{Adv_{m,k,2\gamma}^{MSIS}(A)/Q_H}\,, \tag{14}$$

де Q_H – кількість запитів до квантового випадкового оракула *H*.

Для CRYSTALS-Dilithium перевага супротивника *A* у грі SUF-CMA складає

$$Adv_{Dilithium}^{SUF-CMA} \le Adv_{k,l,D}^{MLWE} + Adv_{H,k,l+1,\zeta_1}^{SelfTargetMSIS} + Adv_{k,l,\zeta_2}^{MSIS} + 2^{-\alpha+1} \tag{15}$$

Де α є мінімальною ентропією схеми,

$$\begin{aligned} \zeta_1 &= \max\{\gamma_1 - \beta, 2\gamma_2 + 1 + 2^{d-1}\cdot\rho\},\\ \zeta_2 &= \max\{2(\gamma_2 - \beta), 4\gamma_1 + 2\} \end{aligned} \tag{16}$$

Мінімальну ентропію α для CRYSTALS-Dilithium можливо розрахувати як

$$\alpha > nl\cdot\log(\min(q/((4\gamma_1+1)(4\gamma_2+1)), 2\gamma_2 - 1)\,, \tag{17}$$

якщо $2\gamma_1, 2\gamma_2 < \sqrt{q/2}$ та $l \le k$.

З формул вище випливає, що для оцінки безпеки схеми необхідно оцінити складність вирішення проблеми MLWE з параметрами *k,l,D*, проблем MSIS з параметрами k, l, ζ_1 та $k, l+1, \zeta_2$. І мінімальна ентропія схеми повинна перевищувати цільовий рівень безпеки.

Оскільки алгебраїчну структуру проблем MLWE та MSIS для криптографічних наборів параметрів невідомо як використовувати, то можливо розглядати відповідні проблеми LWE та SIS.

У табл. 14 зведено результати оцінки безпеки CRYSTALS-Dilithium від атак вкладення для моделі безпеки.

Таблиця 14

Оцінка складності атаки вкладення для Crystals-Dilithium

Набір параметрів	**Складність атаки (біт, GSA)**	**Розмір блоку редукції**	**Складність атаки (біт, Симулятор)**	**Розмір блоку редукції**
Dilithium2	123	424	170	583
Dilithium3	182	625	236	811
Dilithium5	252	864	331	1134

У табл. 15 зведено результати оцінки безпеки CRYSTALS-Dilithium від атак вкладення для моделі безпеки.

Таблиця 15

Оцінка складності атаки декодування для Crystals-Dilithium

Набір параметрів	Складність атаки (біт, GSA)	Розмір блоку редукції	Складність атаки (біт, Симулятор)	Розмір блоку редукції
Dilithium2	124	406	171	565
Dilithium3	186	609	247	234
Dilithium5	261	246	352	1164

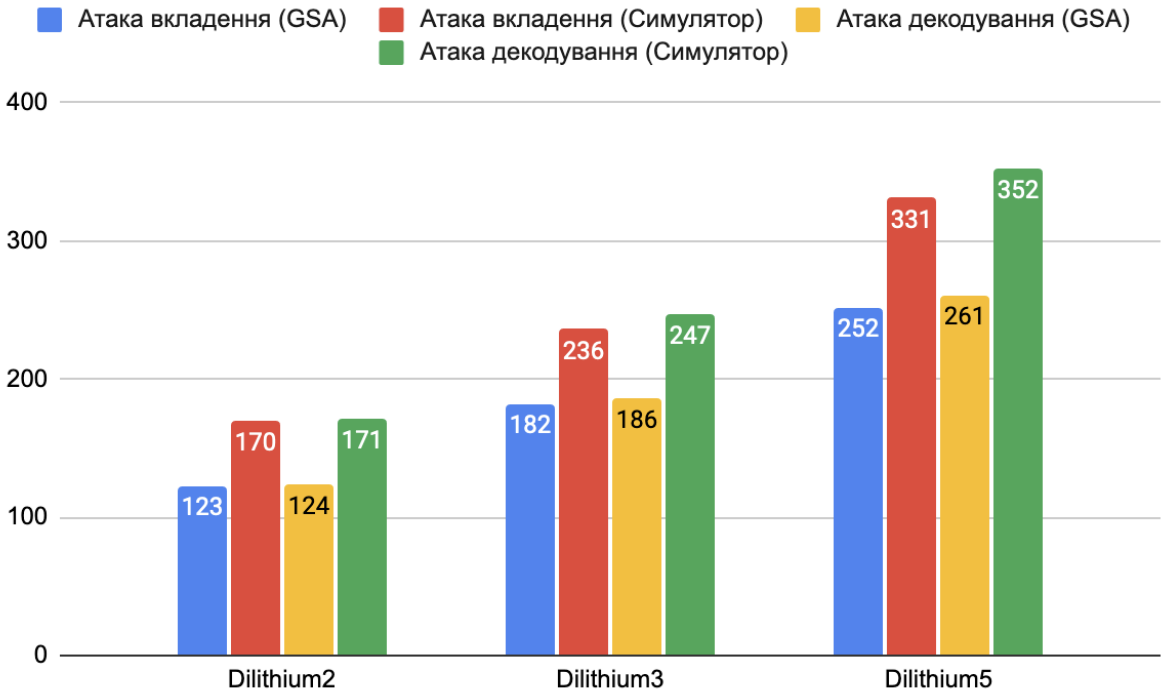

Рис. 7. Порівняння атак вкладення та декодування

У табл. 16 зведено результати оцінки безпеки CRYSTALS-Dilithium (SIS) для наборів параметрів, що представлені в табл. 1.

Таблиця 16

Оцінка складності атаки декодування для Crystals-Dilithium

Dilithium	SIS (автори)	SIS (наша)	SelfTargetSIS (автори)	SelfTargetSIS (наша)
Dilithium2	123	113	121	111
Dilithium3	186	135	175	127
Dilithium5	265	208	253	199

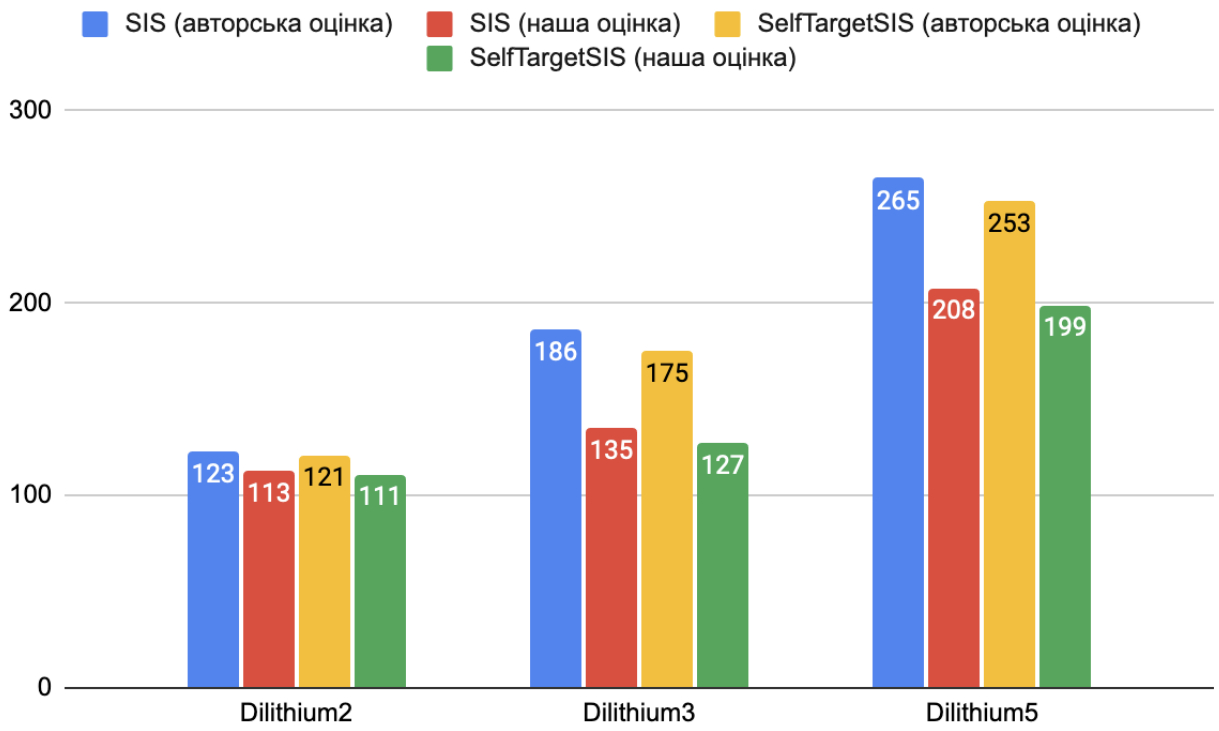

Рис. 8. Порівняння атак на SIS

Висновки

1. Уточнено оцінки безпеки механізмів інкапсуляції ключів ДСТУ 8961:2019 та Crystals-Kyber. В залежності від набору параметрів різниця між оцінками у моделі GSA та моделі, що враховує алгебраїчну структуру q-арних решіток, складає 20–30 біт безпеки. Причому, уточнені оцінки показують, що існуючі атаки є менш ефективними, ніж припускалося при використанні моделі GSA.

2. Атака декодування для 1 рівню безпеки NIST показує кращі результати за атаку вкладення. Для третього рівня безпеки NIST атака декодування має приблизно однакову складність з атаками вкладення. Проте, для п'ятого рівня безпеки атаки декодування значно програють атакам вкладення. Це пояснюється тим, що у атаках декодування форма базису сильніше впливає на параметри атаки.

3. Уточнено оцінки безпеки електронних підписів Falcon та Crystals-Dilithium. В залежності від набору параметрів різниця між оцінками у моделі GSA та моделі, що враховує алгебраїчну структуру q-арних решіток, також складає 20-30 біт безпеки. Причому, уточнені оцінки показують, що існуючі атаки є менш ефективними, ніж припускалося при використанні моделі GSA. Для атак декодування ефективність атаки стрімко падає з ростом розмірності решіток, у той час як для атак вкладення різниця не є такою сильною.

Список літератури:

1. ДСТУ 8961:2019. Інформаційні технології. Криптографічний захист інформації. Алгоритм асиметричного шифрування та інкапсуляції ключів. Чинний від 21.12.2019. Вид. офіц. Київ : УкрНДНЦ, 2019. 72 с.

2. National Institute of Standards and Technology (2024) Module-lattice-based key-encapsulation mechanism standard, CSRC. Available at: https://csrc.nist.gov/pubs/fips/203/final (Accessed: 13 October 2024).

3. National Institute of Standards and Technology (2024a) Module-lattice-based digital signature standard, CSRC. Available at: https://csrc.nist.gov/pubs/fips/204/final (Accessed: 13 October 2024).

4. [PDF] falcon: Fast-fourier lattice-based compact signatures over NTRU | Semantic scholar. Available at: https://www.semanticscholar.org/paper/Falcon:-Fast-Fourier-Lattice-based-Compact-over-Fouque-Hoffstein/423e31b1b96ffa0559078961963baeeb98f01e19 (Accessed: 13 October 2024).

5. Kandii S.O. and Gorbenko, I.D. Assessing the influence of the algebraic structure of q-ary lattices on the complexity of cryptanalysis of problems on lattices // Radiotekhnika. 2024. No217. P. 79–99. doi:10.30837/rt.2024.2.217.07.
6. Bellare M. et al. (1998) Relations among notions of security for public-key encryption schemes // Lecture Notes in Computer Science. 1998. P. 26–45. doi:10.1007/bfb0055718.
7. Goldwasser S., Micali S. and Rivest R.L. A digital signature scheme secure against adaptive chosen-message attacks // SIAM Journal on Computing. 1988. 17(2). P. 281–308. doi:10.1137/0217017.
8. Hövelmanns K., Hülsing A. and Majenz C. Decryption failures and the Fujisaki-Okamoto Transform, Cryptology ePrint Archive. Available at: https://eprint.iacr.org/2022/365.pdf (Accessed: 13 October 2024).
9. Kandii S.O. and Gorbenko I.D. Analysis of DSTU 8961:2019 in the quantum random Oracle Model // Radiotekhnika. 2023. No214. P. 7–16. doi:10.30837/rt.2023.3.214.01.
10. Lyubashevsky V. Fiat-Shamir with aborts: Applications to lattice and factoring-based signatures // Lecture Notes in Computer Science. 2009. P. 598–616. doi:10.1007/978-3-642-10366-7_35.
11. Kosuge H. and Xagawa K. Probabilistic hash-and-sign with retry in the quantum random Oracle Model // Lecture Notes in Computer Science. 2024. P. 259–288. doi:10.1007/978-3-031-57718-5_9.
12. Kiltz E., Lyubashevsky V. and Schaffner C. A concrete treatment of Fiat-Shamir signatures in the quantum random-oracle model // Lecture Notes in Computer Science. 2018. P. 552–586. doi:10.1007/978-3-319-78372-7_18.

Надійшла до редколегії 07.09.2024

Відомості про авторів:
Кандій Сергій Олегович – Харківський національний університет імені В. Н. Каразіна, аспірант кафедри кібербезпеки інформаційних систем, мереж і технологій, навчально-науковий інститут комп'ютерних наук та штучного інтелекту, АТ «Інститут Інформаційних технологій», науковий консультант; Україна; e-mail: sergeykandy@gmail.com; ORCID: https://orcid.org/0000-0003-0552-8341
Горбенко Іван Дмитрович – д-р техн. наук, професор, Харківський національний університет імені В.Н. Каразіна, професор кафедри кібербезпеки інформаційних систем, мереж і технологій, навчально-науковий інститут комп'ютерних наук та штучного інтелекту, АТ «Інститут інформаційних технологій», головний конструктор; Україна; e-mail: i.d.gorbenko@karazin.ua; ORCID: https://orcid.org/0000-0003-4616-3449