УДК 003.026:004.056 **DOI:10.30837/rt.2023.4.211.04**

Д.В. ГАРМАШ

АНАЛІЗ ПІДПИСУ FALCON В ПОРІВНЯННІ З ІНШИМИ ПІДПИСАМИ. ФРЕЙМВОРКИ GPV ТА РАБІНА

Вступ

Квантовий комп'ютер може зруйнувати більшість, якщо не всі традиційні криптосистеми, що використовуються на практиці, а саме – всі системи на основі задачі факторизації цілих чисел (наприклад RSA) або завдання дискретного логарифмування (як традиційних, так і на еліптичних кривих Діффі – Хеллмана і DSA; а також всю криптографію, засновану на спаровуваннях). Деякі класичні криптосхеми, що базуються на обчислювально-складних завданнях, сильно відрізняються від зазначених вище і їх набагато складніше вирішити, вони залишаються незалежними від квантових обчислень. У даній роботі проведено огляд алгоритму Falcon.

1. Порівняння FALCON та схеми CRYSTALS-DILITHIUM

1.1. Crystals-Dilithium

Dilithium – схема підпису, яка наслідує концепцію створення схеми підпису зі схеми ідентифікації, використовуючи Fiat-Shamir з перериванням. Безпека даної схеми може бути скорочена до проблем безпеки Модульного-навчання з помилками (MLWE) і Модульного короткого цілочисельного рішення (MSIS). Вона створюється з метою дозволу швидким множенням використовувати NTT перетворення і уникати появи випадковості з дискретного поширення Gaussian замість вибору зразка з однорідного поширення.

Безпека Dilithium заснована на основоположних проблемах MLWE та MSIS. На даний момент не існує жодної ефективної атаки, що використовує модульну структуру і розглядається в якості безпеки, еквівалентної проблемам MLWE та MSIS.

На перевагу до інших пропозицій щодо підпису Dilithium підбирає з одноманітного поширення, уникаючи складний та неефективний відбір з дискретного поширення Gaussian. Модульна структура Dilithium забезпечує, що поліноміальне множення завжди виконується у тому ж самому кільці незалежно від рівня безпеки, який робить її легким для перемикання між рівнями. Множення може бути виконано ефективно через власні «дружні» параметри NTT. Використовуючи фокус для стискання відкритого ключа з фактором 2, Dilithium має найменший відкритий ключ плюс розмір підпису схеми на основі решіток, що використовують одноманітний відбір [3].

1.2. Falcon

Falcon – схема підпису, чий дизайн заснований на базі фреймворку Gentry-Peikert-Vaikuntanathan (GPV) для підписів на основі решіток, що використовують приховані функції. Він створює екземпляри даного тлумачення за допомогою решіток СКПН і ефективного зразку Gaussian, який створює схему, що є доказово безпечною на основі припущення, що SIS є складним, особливо у використаних решітках. Falcon був створений таким чином, що усі арифметичні операції можуть бути обчислені, використовуючи ефективні техніки Fourier-перетворення.

Він не вимагає (але може використовувати) одиницю з плаваючою крапкою і працювати ефективно на базі мікропроцесорів різного виду, включаючи Intel x86 і ARM cores. Постійний у часі шаблон Gaussian може бути використаний у Falcon.

Математична безпека Falcon покладається на твердість проблеми SIS над кільцями СКПН, яка виграє над довгою історією криптоаналізу для криптосистеми СКПН. Найбільш відомі атаки – загальні техніки решіток: не існує поширеного засобу для використання додаткової кільцевої структури, представленої у решітках СКПН. Для оцінювання безпеки проти

алгоритмів скорочення решіток, Falcon вживлює метод «Core-SVP», який також використовувався багатьма іншими представленнями NTT на основі решіток [2].

Коротко, Falcon – дуже компактний (найменший комбінований розмір відкритого ключа і підпису серед усіх кандидатів НІСТ) і ефективна схема пост-квантового підпису, чия безпека скорочується до добре оцінених припущень. Обрана кільцева структура і помилкове поширення дозволяються для ефективних реалізацій на основі FFT, які частково відміняють несприятливий вплив виконання помилкового поширення Gaussian і призводять до задовільного представлення на практиці. Насправді, можливо найбільшим недоліком Falcon здається складність розуміння усіх деталей тлумачення і реалізації схеми правильно.

Так само, як і їхні фізичні аналоги, цифрові підписи призначені для підтвердження того, що документ видано чи схвалено коректним чином. Разом зі схемами шифрування вони відіграють важливу роль у безпеці електронних комунікацій. Оскільки в цифровому світі все можна відтворити, цифрові підписи не можуть використовувати ті ж принципи, що й фізичні; натомість вони покладаються на складність математичних проблем.

Підписувач зберігає при собі закритий ключ, який він використовує кожного разу, коли обчислює підпис. З цим закритим ключем пов'язаний відкритий ключ, який він може публічно надіслати будь-кому. Кожного разу, коли підписувачу потрібно підписати повідомлення, він використовує свій особистий ключ, щоб вирішити якусь математичну складну задачу, яка залежить лише від повідомлення та відкритого ключа; рішенням буде підпис. З іншого боку, верифікатор генерує ту саму проблему (оскільки вона залежить лише від відкритих елементів) і використовує свій відкритий ключ, щоб перевірити, що підпис справді є вирішенням проблеми. Однак відкритий ключ не допомагає верифікатору самостійно вирішити проблему, якщо все це може здатися трохи абстрактним [3].

2. Схема Рабіна: приклад на основі факторизації

Відомо, що проблема розкладання на множники є складною: задано два дуже великі цілі числа p і q (скажімо, 1000 цифр кожне), комп'ютер може обчислити їхній результат $N = p \times q$ моментально, але відновлення (p, q) із заданим N недоступне для сучасних комп'ютерів. Є проблеми, які важко розв'язати, знаючи лише N, але вони вирішуються, враховуючи його розкладання на множники (p, q). Розглянемо, наприклад, задачу обчислення квадратного кореня: задане ціле число y, ми хочемо знайти таке ціле число $x^2 = y \bmod N$. Якщо ми знаємо лише N, це складна задача на класичному комп'ютері (принаймні, немає відомого ефективного методу її розв'язання), але перевірити, чи є x правильним рішенням, легко: просто перевірити $x^2 = y \bmod N$. Однак, якщо ми знаємо розкладання $N = p \times q$, то ця задача легко вирішується за допомогою цього алгоритму:

1) Обчислити квадратний корінь з y за модулем p і q. Є багато способів зробити це, наприклад алгоритм Тонеллі – Шенкса.

2) Скористатися китайською теоремою про залишки, щоб поєднати ці квадратні корені за модулями p і q у квадратний корінь за модулем N [4].

У цьому прикладі показано, які виникають проблеми (обчислення квадратного кореня):

1) Легко перевірити, важко вирішити за допомогою відкритого ключа N.

2) Легко вирішити за допомогою закритого ключа (p, q).

Криптографія з відкритим ключем використовує переваги асиметрії між тим, що можна досягти відкритим і закритим ключами. Наприклад, схема підпису Рабіна базується на конкретній проблемі, викладеній вище. Ця схема працює наступним чином:

1) Підписувач підписує повідомлення, спочатку надсилаючи його випадковій цілі y (використовуючи геш-функцію, тип функції, яка надсилає вхідні дані до випадкових на вигляд виходів). Потім він використовує свій особистий ключ (p, q), щоб обчислити квадратний корінь з $y \bmod N$: це рішення x слугуватиме підписом повідомлення.

2) Верифікатор використовує відкритий ключ N, щоб переконатися, що x є дійсним підписом повідомлення, перевіривши, що $x^2 = H(msg) \bmod N$.

Цікаво, що можна показати, що обчислення квадратного кореня та розкладання на множники є еквівалентними проблемами: якщо обчислення квадратного кореня за модулем N є складним, то складним є і розкладання N на множники. Схема Рабіна дотримується парадигми гешування, потім підписування: повідомлення спочатку гешується до цільового виклику, а рішенням для цього виклику є підпис. Falcon дотримується тієї ж парадигми, але замість цілочисельної факторизації використовує решітку [6].

2.1. Як влаштовані алгебраїчні решітки

Схема Рабіна не є постквантовою. Дійсно, її основну проблему, розкладання на множники, можна швидко вирішити за допомогою великомасштабного квантового комп'ютера. Однак його ідеї високого рівня можна адаптувати для роботи над проблемами решіток, які, як припускають, протистоять квантовим зловмисникам.

По суті, решітка – це нескінченна кількість точок, розташованих у вигляді сітки. Наприклад, на малюнку нижче зображена двовимірна решітка. Загалом, решітка може існувати в будь-якій додатній кількості вимірів.

Решітка має нескінченну кількість точок, що, звичайно, викликає питання практичності: чи потрібно зберігати нескінченну кількість точок? Звичайно, відповідь – ні, ми можемо бути ефективнішими. Першим кроком до практичності є робота лише з q-ірними ґратками; це решітки, координати точок яких є цілими і «обертають» деяке ціле число q, тобто, якщо ми зменшуємо за модулем q координати точки решітки, результатом все одно буде точка решітки [5].

2.2. Як відбувається створення цифрового підпису Falcone

Загальна інформація: фреймворк GPV.

Falcon слідує структурі, представленій у 2008 р. Гентрі, Пейкертом і Вайкунтатаном, яку скорочено називають фреймворком GPV. Деталі їх роботи можуть бути досить технічними, але ідея високого рівня полягає в наступному:

1) Відкритий ключ є довгою основою q-ї решітки.

2) Приватний ключ є (по суті) короткою основою тієї ж решітки.

3) Під час процедури підписання підписувач:

- генерує випадкове значення v;
- обчислює ціль $c = H(m||v)$, де H – геш-функція, яка надсилає вхідні дані до випадкової точки (на сітці), m – повідомлення;
- використовує свої знання про короткий базис для обчислення точки решітки v поблизу цілі c;
- на виході отримує (m, s), де $s=c-v$.

4) Верифікатор приймає підпис (m, s) в такому випадку, якщо вектор v короткий, та $H(m||v)-s$ є точкою на решітці, згенерованою його відкритим ключем [7].

2.3. NTRU решітки

Першим кроком для створення екземпляра GPV-структури є вибір класу криптографічно жорстких решіток: повинна бути можливість побудувати короткий і довгий базис для тієї самої решітки, щоб будь-кому з довгим базисом було важко знайти близькі вектори з такою ж точністю, як із коротким базисом. Для дидактичних цілей у прикладі використовується решітка розмірності $n = 2$. Однак на практиці ця розмірність недостатня для забезпечення безпеки: у низьких розмірностях алгоритми зменшення решітки, такі як LLL, можуть швидко відновити короткий базис з довгого базису. Подібно до того, як для захисту від класичних комп'ютерів RSA вимагає чисел у кілька тисяч біт, для захисту від класичних і квантових комп'ютерів криптосистеми на основі решітки зазвичай потребують розмірів у порядку величини $n = 1024$.

Зберігання баз таких великих розмірів може бути дорогим: кінцевий відкритий ключ легко може бути більшим за мегабайт. Щоб уникнути цієї проблеми, типово працювати зі структурованими решітками, де цілий базис можна отримати обертанням коефіцієнтів кількох початкових базисних векторів. Це значно зменшує розмір бази для зберігання. Falcon використовує решітки NTRU, які є класом таких структурованих решіток. Їх використання дозволяє зменшити розмір відкритого ключа до менш ніж 1,8 кілобайт. З моменту створення більше ніж 20 років решітки NTRU успішно витримали ретельну перевірку [8].

2.4. Швидка вибірка Фур'є

Другим кроком є вибір алгоритму для обчислення векторів тісної решітки на кроці 3.3 схеми підпису. Хоча алгоритм округлення Бабаї є дуже ефективним, відомо, що його не слід використовувати тут. Дійсно, результат алгоритму Бабаї завжди є паралелепіпедом, що має форму використовуваного базису, тому його використання призведе до повільного витоку закритого ключа. Натомість у своїй структурі Гентрі та його співавтори рекомендують працювати з модифікацією алгоритму Бабаї, де:

- кожен коефіцієнт округлюється випадковим чином. Це гарантує відсутність витоку інформації про закритий ключ;
- кожне округлення враховує попередні. Це дозволяє відбирати ближчі вектори, ніж із простим округленням.

Отриманий алгоритм, який часто називають семплером GPV, безпечніший і кращий, ніж алгоритм округлення Бабаї. Як додаткове вдосконалення, Falcon використовує структуру решіток NTRU, щоб зробити семплер GPV швидшим на два порядки. Falcon, який називається швидкою дискретизацією Фур'є, можна розглядати як гібрид між дискретизатором GPV і швидким перетворенням Фур'є, яке широко використовується в обробці сигналів. Нижче можна побачити результати роботи двох алгоритмів Falcone (512 та 1064 біт) [7]:

VARIANT	KEYGEN/S	SIGN/S	VERIFY/S	IPKI (BYTES)	ISIGI (BYTES)
Falcon-512	143	6081	37175	897	618
Falcon-1024	50	3075	17697	1793	1234

Висновки

1. Схеми цифрового підпису на решітках є основними претендентами на перемогу в конкурсі NIST PQC. Тому, їх подальший детальний аналіз та порівняння щодо основних характеристик стійкості є першочерговою задачею. Схема FALCON, як фіналіст другого етапу, потребує особливої уваги, оскільки має нетиповий дизайн, що використовує арифметику з плаваючою крапкою.

2. Одним із основних завдань конкурсу NIST США є розробка та прийняття постквантового чи постквантових стандартів ЕП. Зараз фаворіти – CRYSTALS-DILITHIUM та FALCON. Причому, подальше вирішення проблеми безпеки, тобто доведення криптографічної стійкості двох кандидатів-фіналістів, на стандарт ЕП FALCON, може ґрунтуватись на проблемах теорії та практики алгебраїчних решіток.

3. Можна дійти висновків, що Falcon використовує схему високого рівня (систему GPV) із двома компонентами (решітки NTRU та швидка вибірка Фур'є). Цілочисельний модуль завжди однаковий, q = 12289. Відкритий ключ – це один вектор із n цілих чисел від 0 до q–1, аналогічно для кожного підпису. Falcon доступний у двох варіантах, для n=512 або n=1024. Вони націлені на високу ефективність і високий рівень безпеки відповідно.

Список літератури:

1. Lili Chen, Stephen Jordan, Yi-Kai-Liu, Dustin Moody, Rene Peralta, Ray Perlner, Daniel Smith-Tone. Report on Post – Quatum Cryptography. Nistir 8105 (draft).

2. Інтернет-ресурс. Режим доступу http://www.nkj.ru/archive/articles/5309/

3. Горбенко, Ю.І. Методи побудування та аналізу, стандартизація та застосування криптографічних систем : монографія ; заг. ред. І.Д. Горбенко. Харків : Форт, 2015. 959 с

4. Потій О.В, Горбенко Ю.І., Ганзя Р.С., Пономар В.І. // Матеріали V-ї міжнар. наук.-техн. конф. «Захист інформації і безпеки інформаційних систем». Львів, 2016 р., 02.06 – 03.06. С. 52.

5. Reinier Broker. Constructing supersingular elliptic curves // J. Comb. Number Theory, (3): pp. 269 – 273, 2009.

6. McGrew D., Curcio M. Hash-Based Signatures draft-mcgrew-hash-sigs00[Електронний ресурс] / D. McGrew, M. Curcio. Режим доступу: https://tools.ietf.org/html/draftmcgrew-hash-sigs-00 .

7. Lili Chen, Stephen Jordan, Yi-Kai-Liu, Dustin Moody, Rene Peralta, Ray Perlner, Daniel Smith-Tone. Report on Post – Quantum Cryptography. NISTIR 8105 (DRAFT). https://www.google.com.ua/search ?

8. Bernstein D. J. Grover vs. McEliece ; N. Sendrier, editor. Post-Quantum Cryptography // Third International Workshop, PQCrypto 2010. Darmstadt, Germany, May 25–28, 2010. Proceedings, vol. 6061 of Lecture Notes in Computer Science, pages 73 – 80. Springer, 2010.

Надійшла до редколегії 03.12.2022

Відомості про автора:

Гармаш Дмитро Васильович – Харківський національний університет імені В. Н. Каразіна, аспірант кафедри безпеки інформаційних систем і технологій, факультет комп'ютерних наук; Україна; e-mail: donni.dima@gmail.com