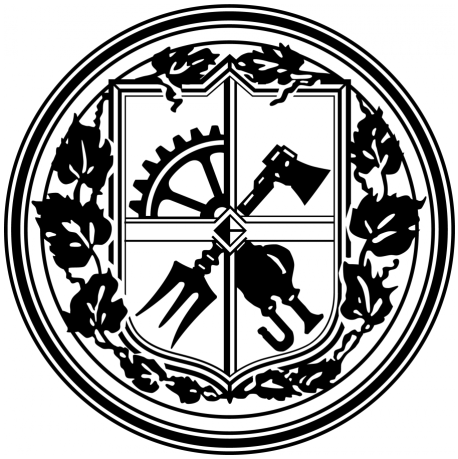
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ім. Ігоря СІКОРСЬКОГО»
ФІЗИКО-ТЕХНІЧНИЙ ІНСТИТУТ



Звіт за темою:

“ДОСЛІДЖЕННЯ СУЧАСНИХ АЛГЕБРАЇЧНИХ КРИПТОСИСТЕМ”

Виконали:
студенти групи ФІ-32мн
Баєвський Констянтин,
Шифрін Денис

Київ — 2023

ЗМІСТ

1 Мета практикуму

Дослідити особливості реалізації сучасних алгебраїчних криптосистем на прикладі учасників першого раунду процесу стандартизації постквантової криптографії (NIST PQC).

1. 1 Постановка задачі та варіант завдання

Бригада №4 ***Алгоритм LUOV***

Треба виконати	**Зроблено**
Повний теоретичний опис алгоритму	✓
Повний теоретичний опис реалізації основних алгебраїчних операцій, які використовує обраний алгоритм	✓
Реалізація алгоритму	
Результати проходження тестових даних	

2 Хід роботи та опис труднощів

Для початку було проаналізовано документацію щодо роботи алгоритму LUOV, аналізу проведених атак на даний алгоритм та відомих результататів досліджень. В процесі роботи було детально описано:

– криптографічний алгоритм LUOV та його складові частини;

– реалізації основних алгебраїчних операцій, які використовує даний алгоритм;

– результати порівняльного аналізу швидкодії даного алгоритму зі схожими алгоритмами (або модифікаціями алгоритму за допомогою заміни складових частин);

– наявні результати досліджень даного алгоритму;

– результати порівняльного аналізу стійкості даного алгоритму зі схожими алгоритмами з обґрунтуванням можливості застосування відомих атак.

В результаті було отримано повний теоретичний опис алгоритму та реалізації основних алгебраїчних операцій, які використовує обраний алгоритм. Дані описи будуть використовуватись безпосередньо для подальшої реалізації даного алгоритму.

При виконанні практикуму поки ніяких труднощів не виникло.

3 Результати дослідження

В результаті проробленої роботи було детально проаналізовано та теоретично описано алгоритм LUOV та реалізації основних алгебраїчних операцій, які використовує

даний алгоритм.

3. 1 Опис алгоритму

LUOV Signature Scheme for NIST PQC Project — це пост–квантовий алгоритм цифрового підпису, заснований на використанні еліптичних кривих. Він забезпечує безпеку підпису навіть у випадку, якщо квантові комп'ютери будуть створені.

Алгоритм LUOV Signature Scheme є одним із найперспективніших пост-квантових алгоритмів цифрового підпису. Він є безпечним, ефективним і відносно простим у реалізації.

LUOV = UOV + PRNG + Field Lifting + спрощений секретний ключ, де:

– UOV – одна з найстаріших і найкраще вивчених схем підпису в галузі багатоваріантної криптографії. Вона була розроблена Дж. Патаріном у 1997 році та витримала два десятиліття криптоаналізу. Схема UOV дуже проста, має маленькі підписи та швидка. Однак основним недоліком UOV є те, що відкритий ключ досить великий. LUOV покращує UOV, щоб прискорити алгоритм і зменшити розмір відкритого ключа.

– PRNG – у оригінальній схемі UOV спочатку навмання вибирається секретний ключ, а потім обчислюється відповідний відкритий ключ. Однак також можна випадково вибрати велику частину відкритого ключа, а потім обчислити відповідний секретний ключ і решту відкритого ключа. LUOV використовує цю техніку, так що більша частина відкритого ключа може бути згенерована з PRNG. Це значно зменшує розмір відкритого ключа.

– Field Lifting – LUOV генерує пару ключів над бінарним полем F2, але підносить ці ключі до розширення поля (наприклад, F 27,F247), щоб використовувати їх для підпису та перевірки повідомлень. Це значно зменшує розмір ключів (оскільки кожен коефіцієнт є одним бітом), але не впливає на складність розв'язування системи поліномів, оскільки розв'язки живуть лише в полі розширення.

– Спрощений секретний ключ – UOV, як і багато інших схем MQ, має властивість, що секретний ключ не є унікальним. Існує величезна кількість секретних ключів, які можуть відповідати одному відкритому ключу. Однак не всі ці можливі секретні ключі однаково ефективні. LUOV вибирає певний секретний ключ, щоб генерація та підписання ключів відбувалися набагато швидше.

Розглянемо еліптичну криву E над полем F_p, де p – просте число. Нехай G – точка на кривій E, а n – деяке число, що ділить порядок групи $E(F_p)$.

Алгоритм 0.1. Генерація ключів для LUOV.
Приймає в якості вхідних даних приватний набір і виробляє відкритий ключ (публічний набір, Q2) і приватний ключ (приватний набір).

– InitializeAndAbsorb(private seed): функція ініціалізації губки з заданим приватним набором.

– SqueezePublicSeed(private sponge): функція вичавлювання публічного набору з приватної губки. Публічний набір використовується для отримання відкритого ключа і оприлюднюється.

– SqueezeT(private sponge): функція вичавлювання значення T з приватної губки. Значення T використовується для отримання приватного ключа і зберігається в секреті.

– InitializeAndAbsorb(public seed): функція ініціалізування іншої губки з публічним набором.

– SqueezePublicMap(public sponge): функція вичавлювання трьох значень (C, L, Q1) з публічної губки. Значення C і L використовуються для отримання відкритого ключа, а значення Q1 використовується для отримання приватного ключа.

– FindQ2(Q1, T): функція обчислення значення Q2 з Q1 і T. Значення Q2 є частиною відкритого ключа.

– Return (public seed, Q2) and private seed: повернення відкритого ключа (публічний набір, Q2) і приватного ключа (приватний набір).

Саме функція FindQ2(Q1, T) є важливою частиною алгоритму. Вона забезпечує безпеку алгоритму, оскільки її важко обчислити, знаючи лише значення Q1 і T.

Приватний ключ для схеми підпису LUOV – це послідовність із 256 випадкових бітів (які використовуються для заповнення Keccak1600 Sponge) і просто кодується як послідовність із 32 байтів.

Відкритий ключ схеми підпису LUOV – це послідовність із 32 байтів (які використовуються для заповнення Keccak Sponge) і матриці m на $\frac{m(m+1)}{2}$ із двійковими записами. Матриця кодується шляхом об'єднання стовпців і доповнення результату нульовими бітами, щоб отримати послідовність бітів, довжина яких ділиться на 8. Потім послідовність інтерпретується як послідовність байтів, де перші біти мають найменші значення.

Публічне початкове число, представлене 32 байтами, отримують із губки шляхом стиснення (видавлювання) 32 байтів. Ця операція називається SqueezePublicSeed.

Алгоритм 0.2. LUOV Signature Scheme працює наступним чином:
Вхід: відкритий ключ (s_1,s_2), приватний ключ s_3, повідомлення M.
Вихід: підпис (R,S).

1) Генеруються випадкові числа a та b в діапазоні $[0, n-1]$.
2) Обчислюється точка $Q = aP+bG$, де G і H – довільно обрані точки на еліптичній криві E.
3) $R = (M + s_1Q) \bmod n$.
4) $S = (s_2Q + R) \bmod n$.

Перевірка підпису:

Для перевірки підпису (R, S) повідомлення M виконується наступна операція:

– $V = (RS) \bmod n$.

Якщо $V = M$, то підпис є дійсним.

Схема підпису може використовуватись в **режимі відновлення повідомлень**. Використання відновлення повідомлення не впливає на алгоритм створення підпису. Ту саму пару ключів можна використовувати для підпису повідомлень у режимі відновлення повідомлень і в режимі доданого підпису, підпис для M у режимі доданого підпису не пов'язаний з підписом для того самого повідомлення в режимі відновлення повідомлення, оскільки інший байт додається до повідомлення в кожному режимі.

Безпека алгоритму:

Безпека алгоритму LUOV Signature Scheme базується на тому, що важко обчислити точку P, знаючи лише точку Q і пару чисел (a, b). Ця задача є NP–складною, і навіть якщо квантові комп'ютери будуть створені, то її рішення буде вимагати значних ресурсів.

Особливості реалізації алгоритму:

– У реалізації алгоритму використовується еліптична крива над полем F p, де p – число з 128 біт.

– Для генерації випадкових чисел використовується криптографічний генератор псевдовипадкових чисел.

– Для обчислення точки $aP + bG$ використовується алгоритм швидкого множення еліптичних кривих.

Основні алгебраїчні операції, які використовує LUOV, це:

– Додавання еліптичних кривих:

$$Q = aP + bG \tag{0.1}$$

– Множення числа на точку еліптичної кривої:

$$s_1 Q = aP + bGs_1 \tag{0.2}$$

3. 2 Продуктивність LUOV

Рівень безпеки	**Розмір секретного ключа**	**Розмір відкритого ключа**	**Розмір підпису**	**Генерація ключів**	**Підписання**	**Перевірка**
1	32 Б	11,5 Кб	239 Б	1,1 М циклів	224 К цикли	49 К циклів
3	32 Б	35,4 Кб	337 Б	4,6 М циклів	643 К цикли	152 К цикли
5	32 Б	82,0 КБ	440 Б	9,7 М циклів	1,1 М циклів	331 К циклів

Таблиця 1 – Характеристики продуктивності оптимізованої реалізації LUOV

Характеристики продуктивності оптимізованої реалізації LUOV з постійним часом AVX2. Для покращення продуктивності дана реалізація виконує деякі попередні

обчислення пари ключів. Для впровадження без інструкцій AVX2 або з різними рівнями попереднього обчислення звертаємося до документа NIST PQC Round2.

3. 3 Атака QuantumHammer

Гібридна атака QuantumHammer — це комбінація двох атак: атаки трасування бітів, увімкненої за допомогою впровадження помилки Rowhammer, і атаки розділяй і володарюй, яка використовує трасування бітів як оракул. Використовуючи трасування бітів, зловмисник, маючи доступ до помилкових підписів, зібраних за допомогою атаки Rowhammer, може відновити секретні ключові біти, хоча й повільно. Використовується атака «розділяй і володарюй», яка в свою чергу використовує структуру в частині LUOV, яка генерує ключі, і ефективніше розв'язує систему рівнянь для секретного ключа з кількома бітами ключа, відновленими за допомогою трасування бітів. Було продемонстровано першу успішну атаку в дикій природі на LUOV, яка відновила всі 11 тисяч бітів ключа менш ніж за 4 години активної атаки Rowhammer. Частина постобробки є дуже паралельною, тому її можна тривіально пришвидшити, використовуючи скромні ресурси. QuantumHammer не робить жодних нереалістичних припущень, вимагає лише спільного розміщення програмного забезпечення (без фізичного доступу), і тому може використовуватися для націлювання на спільні хмарні сервери або в інших середовищах ізольованого програмного середовища.

Таким чином атака Rowhammer може призвести до серйозних наслідків через перевертання бітів в інших процесах і витік ключової інформації. QuantumHammerattack поєднує в собі обидві слабкі сторони, щоб розпочати успішну атаку з відновленням повного секретного ключа схеми.

3. 4 Атака Nested Subset Differential Attack

Модифікована версія диференціальної атаки підполя під назвою «Вкладена диференціальна атака підмножини» повністю порушує половину параметрів, встановлених у раунді 2 версії Lifted Unbalanced Oil and Vinegar. Автори звели атаку на ці набори параметрів до проблеми розв'язання квадратних рівнянь над простим полем F2. Це робить їхню атаку достатньо ефективною для практичного виконання. Оскільки дана атака не використовувала незбалансовану структуру LUOV, її можна розглядати як метод вирішення піднятих квадратичних систем загалом. Автори вважають, що необхідні додаткові дослідження для розв'язання такого типу квадратичних систем за

допомогою атаки NSDA. Також було проведено експериментальні атаки на фактичні параметри LUOV і змогли підробити підпис менш ніж за 210 хвилин.

4 Висновки

В даному практикумі наведено повний теоретичний опис алгоритму з усіма деталями та відомими результатами досліджень. Також проведено теоретичний порівняльний аналіз обраного алгоритму зі схожими алгоритмами та дослідження відомих атак на даний алгоритм.