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

Звіт з виконання комп'ютерного практикуму

ДОСЛІДЖЕННЯ СУЧАСНИХ АЛГЕБРАЇЧНИХ КРИПТОСИСТЕМ

Виконали студентки
групи ФІ-32мн
Зацаренко А.Ю.
Футурська О.В.

Перевірив:
Фесенко А.В.

Київ — 2023

ЗВІТ

1.1 Мета проведення комп'ютерного практикуму

Дослідження особливостей реалізації сучасних алгебраїчних криптосистем на прикладі учасника першого раунду процесу стандартизації постквантової криптографії (NIST PQC) – NTRU-HRSS-KEM.

1.2 Постановка задачі

Провести детальний огляд теоретичних відомостей про обраний криптографічний алгоритм та реалізувати усі можливі його варіації. На основі цього виконати порівняльний аналіз алгоритму щодо постквантової стійкості.

1.3 Хід виконання роботи, опис труднощів, що виникали, та шляхи їх подолання

Хід роботи:

1) Розглянути основні теоретичні відомості щодо відповідного криптографічного алгоритму, які необхідні для подальшої реалізації;

2) Розробити програмну реалізацію криптосистеми згідно з обраним варіантом;

3) Перевірити код на коректність за допомогою відповідних тестів;

4) Дослідити схожі алгоритми та модифікації і провести порівняльний аналіз на швидкодію;

5) Розглянути можливість перенесення відомих атак на обраний алгоритм.

Опис труднощів та шлях вирішення:

– .

1.4 Детальний опис алгоритму NTRU-HRSS-KEM та його складових частин

У даному розділі буде наведено детальне теоретичне підґрунття, необхідне для виконання комп'ютерного практимуму.

1.4.1 NTRU

NTRU був вперше опублікований Гофштейном, Пайфером, Сільверманом у 1998 році. У цій оригінальній роботі поліноміальна алгебра і модульна арифметика використовуються для отримання криптосистеми, безпека якої ґрунтується на тому, що важко знайти надзвичайно короткі вектори в так званій решітці NTRU. Ця задача безпосередньо пов'язана з більш загальною задачею про найкоротший вектор (SVP) на решітках, яка вважається стійкою до квантових атак.

У цій криптосистемі всі об'єкти, такі як ключі, повідомлення та зашифровані тексти, представлені поліномами максимального степеня. Зокрема, разом з поліномами визначено дві операції, які надають їй кільцеву структуру.

Параметри:

1) просте n:

2) p, q – не обов'язково прості, НСД$(p, q) = 1, q > p$;

3) $\mathcal{L}_f, \mathcal{L}_g, \mathcal{L}_r, \mathcal{L}_m$ – чотири набори поліномів степені $n-1$ з цілими коефіцієнтами;

Кільце має вигляд $R = \mathbb{Z}[x]/(x^n - 1)$. Сам поліном записується в наступному вигляді:

$$f = \sum_{i=0}^{n-1} f_i x_i = \{f_0, f_1, \dots f_{n-1}\}$$

Крім цього, введемо операцію множення на R, яка позначається наступним знаком – $\circledast$ та яку задано як добуток циклічної згортки (конволюція). Отже, маємо, що

$$f \circledast g = h, \text{ де } h_k = \sum_{i=0}^{k} f_i g_{k-i} + \sum_{i=k+1}^{n-1} f_i g_{n+k-i} = \sum_{i+j \equiv k \ (mod \ n)} f_i g_j$$

Варто зазначити, що під мультиплікацією з модулем, мається на увазі зменшення коефіцієнтів за заданим модулем.

Позначимо F_p та F_q – відповідні обернені елементи. Після генерації отримуємо приватний ключ – пара (f, F_p):

$$f \circledast F_q \ mod \ q \equiv 1 \text{ і } f \circledast F_p) \ mod \ p \equiv 1$$

і публічний ключ h:

$$h \equiv f \circledast F_q \ (mod \ p).$$

Шифрування відбувається наступним чином:

$$c \equiv (p\text{ф} \circledast + m) \ mod \ q$$

Розшифрування:

$$F_p \circledast a \ (mod \ p), \text{ де } a \equiv f \circledast e \ (mod \ q)$$

NTRU вписується в загальні рамки ймовірнісної криптосистеми. Це означає, що шифрування включає в себе випадковий елемент, тому кожне повідомлення має багато можливих варіантів шифрування. Для відповідних значень параметрів існує надзвичайно висока ймовірність того, що процедура розшифрування відновить оригінальне повідомлення. Однак, деякі значення параметрів можуть призвести до випадкових невдач при розшифруванні, тому слід додавати декілька додаткових контрольних бітів у кожний блок повідомлення. Звичайною причиною невдалого розшифрування є неправильне центрування повідомлення, тобто коли значення $a \notin [-\frac{q}{2}, \frac{q}{2}]$, і чим далі від даної множини, тим більше значення ймовірності провалу.

Як зазначається в оригінальній роботі, шифрування та дешифрування з NTRU надзвичайно швидкі, а створення ключів є простим процесом.

Крім NTRU-HRSS-KEM, у конкурсі також був представлений алгоритм NTRU Prime. Це варіант NTRU, який використовує скінченне поле виду $(Z/q)[x]/(x^n - x - 1)$, де n –просте.

1.4.2 OW CPA-безпечна схема шифрування NTRU HRSS

NTRUEncrypt High-Res Security Standard (NTRU HRSS) – це криптографічний алгоритм на основі решітки, який належить до пост-квантової криптографії, який був вперше опублікований у 2017 році. Він є вдосконаленням оригінального алгоритму NTRU і призначений для забезпечення більш високого рівня захисту від атак, в тому числі з боку квантових комп'ютерів.

NTRU-HRSS - це схема шифрування з відкритим ключем, яка захищена OW CPA. Це поняття безпеки, яка гарантує, що схема шифрування захищена від атак на обраний відкритий текст (CPA), і додатково забезпечує складність отримання додаткової інформації про відкритий текст із зашифрованого, навіть за наявності потужних обчислювальних ресурсів.

Криптосистема є прямою параметризацією NTRU. Її конструктивна новизна полягає у виборі просторів вибірок для повідомлень, сліпих поліномів і закритих ключів.

Ці простори були обрані таким чином, щоб

а) NTRU-HRSS була коректною (дешифрування ніколи не дає збою);

б) допускала просту і ефективну реалізацію з постійним часом;

в) уникала зайвих параметрів, характерних для інших реалізацій NTRU.

Основні зміни порівняно з NTRU:

1) операції виконуються безпосередньо з S_n, щоб уникнути поширених проблем безпеки, пов'язаних з підкільцем S_1. Хоча можна реалізувати NTRU безпосередньо в S_n і не використовувати R_n взагалі, все ж елементи S_n підносяться в R_n, щоб скористатися зручними обчислювальними і геометричними особливостями R_n;

2) параметри підбираються таким чином, щоб повністю виключити збої при розшифруванні, і робиться це без обмеження простору ключа і повідомлення;

3) усувається будь-яка необхідність у розподілах з фіксованою вагою. Всі процедури вибірки обрано таким чином, щоб вони допускали прості та ефективні реалізації з постійним часом.

Параметри:

1) просте n, для якого:

а) порядок 2 в $(\mathbb{Z}/n)^\times$ дорівнює $(n-1)$, тобто 2 – генератор даної мультиплікативної групи;

б) порядок 3 в $(\mathbb{Z}/n)^\times$ дорівнює $(n-1)$, тобто 3 – генератор даної мультиплікативної групи.

У даній роботі ми використовуватимемо $n = 701$, щоб забезпечити 128-бітову безпеку для постквантів.

2) $p = 3$;

3) $q = 2^{3.5+log\ n}$, у нас це $q = 8192$;

Визначимо

$$\mathcal{T} = \{v \in \{-1, 0, 1\}^n : v_{n-1} = 0\}, \text{ та}$$
$$\mathcal{T}_+ = \{v \in \mathcal{T} : <x \circledast v, v> \geqslant 0\}$$

Тоді простори:

$$\mathcal{L}_f = \mathcal{L}_g = \mathcal{T}_+ \text{ і } \mathcal{L}_r = \mathcal{L}_m = \mathcal{T}$$

Крім цього, введемо наступні поняття:

– Φ_n – поліном вигляду $\dfrac{(x^n-1)}{(x-1)} = x^{n-1} + x^{n-2} + \ldots + 1$, має бути незвідним за модулем як p та q. Це позбавляє від перевірки на обернений під час генерації ключів і робить процес більш придатним для реалізації з постійним часом. Цікаво, що для NTRU Prime умова незвідності вимагається лише для q, в той час як для NTRU ця вимога взагалі була не обов'язковою, хоча і рекомендованою;

– S – фактор-кільце $\mathbb{Z}[x]/(\Phi_n)$;

– S/p – фактор-кільце $\mathbb{Z}[x]/(p, \Phi_n)$.

Тоді канонічний S/q-представник полінома a – це єдиний поліном b степеня не більше $n-1$ з коефіцієнтами в $\{1, 2, \ldots q-1\}$ такий, що $a \sim b$ як елементи S/q.

Генерація ключа:

1) Обираємо f та g з $\mathcal{L}_f$ і $\mathcal{L}_g$;

2) Знаходимо таке F_q, що $(f \circledast F_q)\ mod\ q \equiv 1$ в S;

3) Знаходимо таке F_p, що $(f \circledast F_p)\ mod\ p \equiv 1$ в S;

4) $h = (p \circledast (x-1) \circledast g \circledast F_q)\ mod\ q$.

Вихід: приватний ключ (f, F_p) і публічний ключ h.

У попередніх варіаціях NTRU f вважалося коротким елементом R з оберненими як в R/p, так і в R/q. З параметрами попереднього розділу, кожен ненульовий елемент $\mathcal{T}$ є оберненим як елемент S/p та S/q. Оберненість в S/p та S/q є достатньою для процедури дешифрування, тому можна відмовитись від перевірки на наявність оберненого у R/p та R/q. Все одно потрібно обчислювати обернені, але цей процес ніколи не дає збоїв.

Шифрування:

Вхід: повідомлення $m \in \mathcal{L}_m$.

1) Обираємо r з $\mathcal{L}_r$;

2) $c = (r \circledast h + LiftP(m))\ mod\ q$, де $LiftP(m) = (x-1) \circledast m_0$,

при $m_0 \in \mathcal{T}$ і $LiftP(m) = m$, у випадку S/p;

Вихід: шифротекст c.

У попередніх інстанціях NTRU r та m були обрані так, щоб мати коефіцієнти з $\{-1, 0, 1\}$ з визначеною кількістю коефіцієнтів, що приймають кожне значення. У даній схемі r та m приймають довільні значення на $\mathcal{T}$.

Дешифрування:

Вхід: шифроекст c

1) $v = (c \circledast f)\ mod\ q$;

2) $u = (u \circledast F_p)\ mod\ p$;

3) $m^{'} = (u - u_{n-1} \cdot \Phi_n)\ mod\ p$.

Вихід: $m^{'}$.

Коректність:

Алгоритм шифрування NTRU-HRSS з параметрами $p = 3$ та $q > 8\sqrt{2}n$ буде працювати коректно, якщо виконуватиметься наступна вимога:

$$c \circledast f = |r \circledast h + LiftP(m) \circledast f| < \frac{q}{2}$$

Розписавши h та $LiftP(r)$, отримаємо наступне:

$$c \circledast f = |r \circledast p \circledast (x-1) \circledast g + LiftP(m) \circledast f| < \frac{q}{2}$$
$$c \circledast f = |p \circledast LiftP(r) \circledast g + LiftP(m) \circledast f| < \frac{q}{2}$$

Згідно з лемою, описаною та доведеною в офіційному документі[1], маємо:

$$LiftP(r) \circledast g \leqslant \sqrt{2}n$$

Тоді отримаємо, що

$$c \circledast f = |p \circledast LiftP(r) \circledast g + LiftP(m) \circledast f| =$$
$$= |3 \circledast LiftP(r) \circledast g + LiftP(m) \circledast f| < 4\sqrt{2}n < \frac{q}{2}$$

Варто зазначити, що точний розподіл, з якого беруться f і g, впливає на безпеку. Генерація ключів використовує загальну функцію $Sample\mathcal{T}_+$, яку можна розглядати як вибірку з рівномірного розподілу на $Sample\mathcal{T}_+$. Однак це складно і довго, тому варто розглянути інші варіанти. Спочатку зазначимо, що будь-яку процедуру вибірки з $\mathcal{T}$ можна перетворити на процедуру вибірки з $\mathcal{T}_+$ з втратою не більше одного біта в ентропії її вихідного розподілу.

Функція вибірки з $Sample\mathcal{T}_+$ обирає $v \in \mathcal{T}$ і потім умовно застосовує парне перевертання знаку індексу до v, якщо $<xv, v> < 0$.

Спрощена процедура вибірки з $Sample\mathcal{T}$ витягує $n-1$ коефіцієнт незалежно з центрованого біноміального розподілу з параметром $t = 2$, а потім зменшує ці коефіцієнти за модулем p.

Центрований біноміальний розподіл з параметром t визначається, як

$$\sum_{i=1}^{t} b_i - b_{t+i}, \text{ де } b_1, b_2, \ldots b_{2t} - \text{рівномірні випадкові біти}$$

Процес завжди споживає рівно $2t(n-1)$ випадкових бітів. Результуючий розподіл відноситься до типу розподілів, що демонструє симетрію відносно певної центральної точки (для будь-яких значень p і t) і прямує до рівномірного розподілу зі збільшенням t.

[1] Andreas Hülsing, Joost Rijneveld, John Schanck, and Peter Schwabe: High-speed key encapsulation from NTRU

1.4.3 CCA2-безпечний механізм інкапсуляції ключів (KEM)

Адаптивна атака на вибраний шифрований текст (скорочено CCA2) - це інтерактивна форма атаки на вибраний шифрований текст, в якій зловмисник спочатку надсилає декілька шифрованих текстів для розшифрування, вибраних адаптивно, а потім використовує результати для розпізнавання цільового шифротексту, не звертаючись до оракула щодо шифротексту.

В оригінальній роботі зазначається INDCCA2 безпека. Якщо криптосистема володіє властивістю нерозрізнення, то зловмисник не зможе розрізнити пари шифротекстів на основі зашифрованого ними повідомлення. Властивість нерозрізнення під час атаки з обраним відкритим текстом вважається основною вимогою для більшості криптосистем з відкритим ключем.

Криптосистема вважається безпечною з точки зору нерозрізнення, якщо жоден зловмисник, зашифрувавши повідомлення, випадково вибране з двоелементного простору повідомлень, визначеного зазделегідь, не зможе ідентифікувати вибір повідомлення з ймовірністю, значно кращою, ніж ймовірність випадкового вгадування ($p = \frac{1}{2}$). Якщо будь-який зловмисник може розпізнати вибраний шифротекст з ймовірністю, значно більшою за $\frac{1}{2}$, то вважається, що він має «перевагу» у розпізнаванні шифротексту, і схема не вважається безпечною з точки зору нерозрізненості. Це визначення охоплює поняття того, що в безпечній схемі противник не повинен дізнатися ніякої інформації, побачивши шифротекст. Таким чином, противник не повинен мати змоги розгадати шифр краще, ніж якби він вгадував його випадковим чином.

Шифрування з відкритим ключем (PKE) - це просто використання можливості зашифрувати щось за допомогою відкритого ключа, а потім розшифрувати його іншою стороною за допомогою закритого ключа. Однак, методи інкапсуляції ключів (KEM) також використовують відкритий і закритий ключі, але природа KEM полягає в тому, що він

генерує симетричний ключ як частину свого процесу.

PKE з достатньо великим технічним простором можна досить тривіально перетворити на KEM. А сам KEM можна перетворити на PKE, додавши трохи симетричного шифрування.

У квантовому розумінні, існує багато чого, що має бути передано між процесами. Тому автоматична генерація симетричного ключа має сенс. Іноді це просто через обмеження, пов'язані з тим, як працює квантовий алгоритм. Але зазвичай це пов'язано з тим, що для того, щоб відбулася комунікація, потрібно щось передати, і тому KEM просто має більше сенсу.

NTRU-HRSS-KEM - це KEM із захистом CCA2, який також був представлений у 2017 році як спроба перетворити вищезгадане безпечне шифрування OW CPA на KEM із захистом CCA2. Конструкція використовує загальне перетворення зі схеми шифрування з відкритим ключем, що захищена OW CPA. Як пряме перетворення KEM дозволяє уникнути механізму паддінгу NAEP, який використовується в стандартному NTRU.

Принцип роботи KEM-перетворення полягає в наступному. Спочатку з простору повідомлень схеми шифрування вибирається випадкове повідомлення m. Цей текст шифрується за допомогою випадкового рядку бітів (coins), детерміновано отриманих з m за допомогою геш-функції, яка пізніше моделюється як випадковий оракул. Ключ сеансу отримується з m за допомогою іншого оракула. Нарешті, виводиться шифротекст і сеансовий ключ.

Алгоритм декапсуляції розшифровує зашифрований текст, щоб отримати m, виводить випадкові рядки (coins) з m і повторно шифрує m, використовуючи ці рядки (coins). Якщо отриманий шифротекст збігається з раніше надісленим, то він генерує сеансовий ключ з m.

Подібний KEM на основі NTRU був запропонований Мартіном Стамом (Martijn Stam) у 2005 році; основні відмінності від цієї роботи полягають у виборі параметрів для NTRU та включенні додаткового гешу, який додається до зашифрованого тексту. Додатковий геш дозволяє довести

безпеку у квантово-доступній моделі випадкового оракула.