УДК 519.7:003.26

Артем Панченко
(Житомир, Україна)

ЧИСЛОВІ ОБЧИСЛЕННЯ: РОЗШИРЕНИЙ АЛГОРИТМ ЕВКЛІДА ТА ІМПЛЕМЕНТАЦІЯ RSA-КРИПТОГРАФІЇ

У статті досліджено застосування основних алгоритмів теорії чисел у криптографії на прикладі RSA. Особливу увагу приділено розширеному алгоритму Евкліда, що використовується для знаходження мультиплікативних обернених елементів і обчислення закритої експоненти. Розглянуто теоретичні засади, ключові компоненти RSA та проведено експериментальну оцінку продуктивності для ключів різних розмірів. Показано високу ефективність розширеного алгоритму та вплив низькорівневих оптимізацій на продуктивність у порівнянні з промисловими бібліотеками. Проаналізовано стійкість RSA та можливі оптимізації.

Ключові слова: *RSA, криптографія з відкритим ключем, теорія чисел, розширений алгоритм Евкліда, модульна арифметика, функція Ейлера, прості числа, мультиплікативний обернений елемент, генерація ключів, шифрування даних, криптографічна стійкість.*

The article examines the use of fundamental number-theoretic algorithms in cryptography through the example of RSA. Special attention is given to the Extended Euclidean Algorithm, which is essential for computing multiplicative inverses and deriving the private exponent. The work outlines the theoretical foundations, key RSA components and presents experimental performance results for various key sizes. The study demonstrates the high efficiency of the Extended Euclidean Algorithm and highlights the impact of low-level optimizations when compared with industrial libraries. The cryptographic strength of RSA and possible optimizations are also discussed.

Key words: *RSA, public key cryptography, number theory, extended Euclidean algorithm, modular arithmetic, Euler's totient function, prime numbers, multiplicative inverse, key generation, data encryption, cryptographic security.*

Вступ. Теорія чисел, одна з найдавніших галузей математики, знайшла своє практичне застосування в сучасній криптографії. RSA (Rivest-Shamir-Adleman) залишається одним із найпоширеніших асиметричних криптографічних алгоритмів, який використовує складність факторизації великих чисел для забезпечення конфіденційності. Центральну роль у його реалізації відіграє розширений алгоритм Евкліда, який дозволяє ефективно обчислювати мультиплікативні обернені елементи в модульній арифметиці.

Теоретичні основи

Основні поняття теорії чисел

Найбільший спільний дільник (НСД) двох чисел визначає найбільше натуральне число, яке ділить обидва числа без залишку. Два числа називаються взаємно простими, якщо їхній НСД дорівнює одиниці. Функція Ейлера φ(n) визначає кількість натуральних чисел, менших за n і взаємно простих з n. Для простого числа p виконується φ(p) = p-1, а для добутку двох простих φ(p×q) = (p-1)(q-1).

Теорема Ейлера стверджує, що для взаємно простих a і n виконується a^φ(n) ≡ 1 (mod n). З цієї теореми випливає можливість знаходження мультиплікативного оберненого елемента: якщо НСД(a,n) = 1, то існує b таке, що a×b ≡ 1 (mod n).

Алгоритм Евкліда та його розширення

Класичний алгоритм Евкліда базується на принципі: НСД(a,b) = НСД(b, a mod b). Алгоритм ітеративно застосовує ділення з залишком до нульового залишку. Складність становить O(log min(a,b)), що забезпечує високу ефективність навіть для великих чисел.

Розширений алгоритм Евкліда не лише знаходить НСД, але й обчислює коефіцієнти x та y в рівнянні Безу: ax + by = НСД(a,b). Ці коефіцієнти дозволяють знайти мультиплікативний обернений елемент у модульній арифметиці. Якщо НСД(a,b) = 1, то x є оберненим до a за модулем b.

Алгоритм працює рекурсивно: на кожній ітерації обчислюється частка q = r_(i-1) div r_i та оновлюються коефіцієнти r_(i+1) = r_(i-1) - q×r_i, s_(i+1) = s_(i-1) - q×s_i, t_(i+1) = t_(i-1) - q×t_i. Після досягнення r_i = 0, коефіцієнт t_(i-1) містить мультиплікативний обернений.

Математичні основи RSA

RSA використовує пару ключів: відкритий (n, e) для шифрування та закритий (n, d) для дешифрування, де n є добутком двох великих простих чисел p і q. Процес генерації включає обчислення модуля n = p×q, функції Ейлера φ(n) = (p-1)(q-1), вибір відкритого експонента e (зазвичай 65537) та обчислення закритого експонента d за допомогою розширеного алгоритму Евкліда: e×d ≡ 1 (mod φ(n)).

Коректність RSA випливає з теореми Ейлера. Шифрування: C = M^e mod n, дешифрування: M = C^d mod n. Підставляючи, отримуємо M = (M^e)^d = M^(ed) mod n. Оскільки ed ≡ 1 (mod φ(n)), то M^(ed) = M × (M^φ(n))^k ≡ M (mod n).

Реалізація основних компонентів

Генерація великих простих чисел

Для генерації криптографічно стійких простих чисел застосовується тест Міллера-Рабіна – імовірнісний алгоритм з поліноміальним часом виконання. Алгоритм базується на розкладанні n-1 = 2^r × d та перевірці умов для випадкових свідків. При k=40 раундах імовірність помилки не перевищує 2^(-80). Процес включає генерацію випадкового непарного числа та ітеративне тестування до знаходження простого кандидата.

Обчислення мультиплікативного оберненого

Знаходження закритого експонента d виконується розширеним алгоритмом Евкліда за O(log² φ(n)) операцій. Алгоритм перевіряє, що НСД(e, φ(n)) = 1, та обчислює d таке, що e×d ≡ 1 (mod φ(n)). Для e = 65537 та типових значень φ(n) обчислення займає мілісекунди навіть для 4096-бітних ключів.

Швидке піднесення до степеня за модулем

Метод бінарного піднесення до степеня зменшує складність з O(n) до O(log n) операцій. Алгоритм використовує бінарне представлення експоненти: на кожному кроці база

зводиться в квадрат, і якщо біт експоненти одиничний, результат множиться на базу. Усі операції виконуються за модулем n. Це дозволяє виконати шифрування з e=65537 лише за 17 операцій множення.

Обробка текстових повідомлень

Схема PKCS#1 v1.5 padding додає до повідомлення випадкові байти для забезпечення рандомізації: 0x00 || 0x02 || PS || 0x00 || M, де PS – випадкова послідовність ненульових байтів. Повідомлення розбивається на блоки розміру, меншого за $\log_2(n)$ біт. Для 2048-бітного ключа максимальний розмір повідомлення становить 245 байт після padding. Дешифрування виконує обернені операції з перевіркою формату padding.

Експериментальне дослідження

Методологія та результати

Тестування проводилося для ключів розміром 1024, 2048 та 4096 біт з 100 повторами кожного експерименту. Генерація 1024-бітних ключів займає 450-600 мс, 2048 біт – 3-4 секунди, 4096 біт – 25-35 секунд. Шифрування 100-символьного повідомлення для 1024 біт виконується за 15-20 мс, для 2048 біт – за 35-45 мс, для 4096 біт – за 120-140 мс.

Дешифрування є найбільш ресурсоємною операцією: 1024 біт – 40-50 мс, 2048 біт - 180-220 мс, 4096 біт – 1200-1500 мс. Це відповідає теоретичній складності $O(\log^3 n)$. Внесок розширеного алгоритму Евкліда у час генерації ключів становить менше 5% навіть для 4096-бітних чисел.

Порівняння з бібліотеками

OpenSSL генерує 2048-бітні ключі за 100-150 мс проти 3-4 секунд власної реалізації (прискорення в 20-30 разів). Шифрування в OpenSSL виконується за 2-3 мс проти 35-45 мс (прискорення в 12-15 разів), дешифрування – за 5-8 мс проти 180-220 мс (прискорення в 30-40 разів). Це пояснюється оптимізованим кодом на C/асемблері та використанням апаратних інструкцій.

Аналіз безпеки

Безпека RSA залежить від складності факторизації модуля n. Найкращі алгоритми факторизації мають субекспоненційну складність: для 1024 біт потрібно приблизно 2^80 операцій, для 2048 біт – 2^112. Сучасні стандарти (NIST, BSI) вимагають мінімум 2048 біт для довгострокового захисту до 2030 року. Ключі 1024 біт вважаються застарілими.

Розширений алгоритм Евкліда є критичним компонентом RSA, забезпечуючи ефективне обчислення закритого експонента. Без нього генерація ключів була б непрактично повільною. Складність $O(\log^2 n)$ робить алгоритм масштабованим для будь-яких практичних розмірів ключів.

Оптимізації включають китайську теорему про залишки (прискорення дешифрування в 4 рази) та метод Монтгомері для захисту від атак по сторонніх каналах. Основним обмеженням є низька швидкість порівняно з симетричним шифруванням – на практиці RSA використовується лише для обміну ключами та цифрових підписів.

Висновки. Робота демонструє взаємозв'язок фундаментальних алгоритмів теорії чисел та практичних криптографічних застосунків. Розширений алгоритм Евкліда залишається ключовим компонентом сучасних систем захисту інформації завдяки своїй ефективності та математичній елегантності.

Експериментальні дослідження підтвердили теоретичні оцінки складності та квадратичну залежність часу виконання від розміру ключа. Розширений алгоритм Евкліда вносить лише $O(\log^2 n)$ до загальної складності генерації ключів, що підкреслює його винятково важливу роль у забезпеченні практичності RSA.

Для практичних застосунків критично важливим є використання перевірених криптографічних бібліотек. Мінімальний розмір ключа має становити 2048 біт, а для довгострокового захисту – 3072 або 4096 біт з урахуванням зростання обчислювальних можливостей та розвитку квантових комп'ютерів.

ДЖЕРЕЛА ТА ЛІТЕРАТУРА

1. Cormen T., Leiserson C., Rivest R., Stein C. Вступ до алгоритмів. 3-тє вид. Cambridge : MIT Press, 2009. 1312 с.

2. Crandall R., Pomerance C. Прості числа: обчислювальний підхід. 2-ге вид. New York : Springer, 2005. 597 с.

3. Hoffstein J., Pipher J., Silverman J. Вступ до математичної криптографії. 2-ге вид. New York : Springer, 2014. 538 с.

4. Menezes A., van Oorschot P., Vanstone S. Довідник з прикладної криптографії. Boca Raton : CRC Press, 1996. 816 с.

5. Miller G. L. Гіпотеза Рімана та тести на простоту чисел. *Journal of Computer and System Sciences.* 1976. Т. 13. С. 300–317.

6. Rivest R., Shamir A., Adleman L. Метод отримання цифрових підписів та криптосистем з відкритим ключем. *Communications of the ACM.* 1978. Т. 21, № 2. С. 120–126.