УДК 658.012.011.56:681.3.06

Карпінський М.П., Якименко І.З., Хомінчук А.А.

УДОСКОНАЛЕННЯ ПАРАЛЕЛЬНОГО АЛГОРИТМУ ПОШУКУ ПОРЯДКУ ЕЛІПТИЧНИХ КРИВИХ

В даній статті досліджено паралельну реалізацію алгоритму визначення неприводимості поліномів високих степенів. Запропоновано аналітичну модель, що дозволяє проаналізувати ефективність використання технології розпаралелення для визначення порядку еліптичної кривої. В статті наведено результати досліджень.

Аналіз стану захисту інформації в інформаційних системах свідчить, що в цілому стан розв'язання цієї задачі далекий від досконалості. Тим більше, з швидким розвитком комп'ютерної техніки, виникає потреба в побудові стійких і продуктивних криптографічних алгоритмів. До них можна віднести криптографічні алгоритми з використанням математичного апарату еліптичних кривих, які вимагають менших витрат ресурсів для забезпечення високого рівня захисту [1].

Особливий інтерес криптографії ЕК обумовлений такими перевагами – швидкодія та невелика довжина ключа. Поруч з цим існує низка взаємопов'язаних задач, для вирішення яких потрібно затратити великі часові та обчислювальні ресурси. Тому потрібні підходи, спеціальні алгоритми і засоби для швидкої роботи алгоритму і генерування стійких параметрів ЕК [2]. Одним з основних параметрів оцінки стійкості ЕК є її порядок [3]. Для вирішення задачі пошуку порядку можна скористатися наступними алгоритмами:

1) алгоритм «крок гіганта – крок малюка»;
2) r - Поларда метод;
3) l - Поларда метод;
4) алгоритм Еткіна та Еліза;
5) алгоритм Шуфа.

Дослідженнями алгоритмів пошуку порядку займалися багато відомих вітчизняних та зарубіжних криптографів. Паралельну реалізацію l - Поларда метода можна зустріти в працях Д.І. Горбенка, С.Збітнєва, удосконалення алгоритму Шуфа в працях Уткіна та Еліза [1]. Алгоритмами паралельної реалізації основних операцій в полях Галуа займалися відомі криптоаналітики М. Моргестерн, Е. Шамір [5].

Для подальших досліджень було запропоновано використати технологію паралельних обчислень в алгоритмі Шуфа, а саме розпарелелення процесу визначення неприводимості поліномів високих степенів, як альтернативу високопродуктивній обчислювальній техніці.

Отже, алгоритм Шуфа знаходить $t(\bmod l)$ для таких простих чисел l, що $\prod l > 4\sqrt{p}$. Потім за допомогою китайської теореми про лишки знайдемо значення t. Значення $t(\bmod 2)$ знайдемо так, як описано вище. Нехай в подальшому $l > 2, l$ - фіксоване просте число, $l \neq p$. Найбільш трудомісткими операціями даного алгоритму є:

піднесення числа до степеня за модулем [4];

визначення неприводимості поліномів високих степенів.

При дослідженні часових затрат визначення неприводимості поліномів високих степенів доцільно скористатися математичною моделлю, яка аналізує час послідовної реалізації [5]:

$$T_s = n^2 \cdot L(n) \cdot \log(p), \qquad (1)$$

де n - порядок полінома; $L(n) = \log(n) \cdot \log(\log(n))$; p – порядок поля.

Для паралельної реалізації алгоритму визначення неприводимості поліномів високих степенів, використаємо аналітичну модель:

$$T_p = w(n) \cdot n \cdot L(n) \cdot \log(p) \qquad (2)$$

Загальна оцінка показника прискорення:

$$S_p = \frac{T_s}{T_p} \qquad (3)$$

Якщо розглянути алгоритм Шуфа [4], то можна побачити, що в ньому, на певних етапах, зустрічається визначення неприводимості поліномів високих порядків. Розглянемо всі часткові випадки, для різних порядків поля – 32, 64, 128 і 256 біт. Відповідно порядок поля буде рівний:

при 32 бітах $p = 4294967295$;

при 64 бітах $p = 18446744073709551615$;

при 128 бітах $p = 3402823669\ \ 2093846346\ \ 3374607431\ \ 768211455$;

при 256 бітах

$p = 115792089210356248762697446949407573530086143415290314195533631308867097853951$

Отже, розглянемо перший випадок.

$$НСД\left((x^{p^2} - x)\ f_k(x)^2 (x^3 + ax + b) + f_{k-1}(x) f_{k+1}(x),\ f_l(x)\right) \neq 1 \qquad (4)$$

В даному випадку порядок поліному $n = p^2$. В таблиці 3.1 наведено результати моделювання.

Таблиця 1.

Часові затрати процесу процесу визначення взаємнопростості поліномів при $n = p^2$

Розрядність числа *p*	Послідовний алгоритм	Паралельний алгоритм			
		4 процесори		8 процесорів	
		Часові затрати	Прискорення	Часові затрати	Прискорення
32 біт	5.8395e+293	1.4599e+293	3.999	7.2994e+292	7.998
64 біт	1.1679e+294	2.9198e+293	3.950	1.4599e+293	7.860
128біт	2.3358e+294	5.8395e+293	3.830	2.9198e+293	7.802
256 біт	4.6716e+314	1.1979e+314	3.763	5.8763e+313	7.7560

На наступному етапі алгоритму Шуфа потрібно визначити взаємопростість наступних поліномів:

$$НСД\left((4f_w(x)^3(x^3+ax+b)^{\frac{p+3}{2}} - f_{w+2}(x)f_{w-1}(x)^2 + f_{w-2}(x)f_{w+1}(x)^2,\ f_l(x)\right) \neq 1 \quad (5)$$

В даному випадку степінь поліному $n = \frac{p+3}{2}$. В таблиці 3.2 наведено результат математичного моделювання такого поліному при різних порядках поля, на якому визначена еліптична крива.

Таблиця 2.

Часові затрати процесу розпаралелення НСД при $n = \frac{p+3}{2}$

Розрядність числа *p*	Послідовний алгоритм	Паралельний алгоритм			
		4 процесори		8 процесорів	
		Часові затрати	Прискорення	Часові затрати	Прискорення
32 біт	4.6725e+138	1.1681e+138	3.973	5.8407e+137	7.957
64 біт	9.3451e+138	2.3363e+138	3.946	1.1681e+138	7.932
128біт	1.8690e+139	4.6725e+138	3.890	2.3363e+138	7.596
256 біт	3.7380e+139	9.3451e+138	3.740	4.6725e+138	7.490

В алгоритмі Шуфа використовується також наступний поліном:

$$НСД\left((4f_w(x)^3(x^3+ax+b)^{\frac{p-1}{2}} - f_{w+2}(x)f_{w-1}(x)^2 + f_{w-2}(x)f_{w+1}(x)^2,\ f_l(x)\right) \neq 1 \quad (6)$$

В даному випадку степінь поліному $n = \frac{p-1}{2}$. В таблиці 3.3 наведено результат математичного моделювання такого поліному.

Таблиця 3.

Часові затрати процесу розпаралелення НСД при $n = \frac{p-1}{2}$

Розрядність числа *p*	Послідовний алгоритм	Паралельний алгоритм			
		4 процесори		8 процесорів	
		Часові затрати	Прискорення	Часові затрати	Прискорення
32 біт	4.6725e+138	1.1681e+138	3.963	5.8407e+137	7.943
64 біт	9.3451e+138	2.3363e+138	3.948	1.1681e+138	7.933
128біт	1.8690e+139	4.6725e+138	3.893	2.3363e+138	7.599

256 біт	3.7380e+139	9.3451e+138	3.745	4.6725e+138	7.501

Згідно з законом Амдала в ідеальному випадку прискорення наближається до кількості процесорів, що використовуються при обчисленнях. З таблиць 3.1-3.3 видно, що значення прискорення наближається до кількості процесорів, тому використання технологій паралельних обчислень для процесу визначення неприводимості поліномів є високоефективним рішенням. Оскільки прискорення практично не падає зі збільшенням кількості процесорів, то можна сказати, що для розпаралелення даної задачі можна використовувати максимальну кількість процесорів без зниження ефективності роботи алгоритму. Отже, при дослідженні стійкості алгоритмів шифрування з використанням математичного апарату ЕК, доцільно скористатися паралельним алгоритмом Шуфа, що дозволяє суттєво зменшити часовий ресурс для пошуку порядку ЕК.

Література

1. Elkies N.D. Elliptic and modular curves over finite fields and related computaional issues // Computational perspectives in number theory: Proc. of a Conf. in Honor of A.O. L.Atkin / J. T. Teitelbaum and D.A.Buell, editors. 1998. (Amer. Math. Soc. Inf. Press; V. 7). P. 21—76.
2. Elliptic Curve Cryptography. Certicom Research, 1999. Working Draft.
3. IEEE P1363 / D8(Draft Version 8). Standard Specifications for Public Key Cryptography.
4. Schoof R. Elliptic curves over finite fields and the computation of square roots mod p // Math. Comp. 1985. V. 44. P. 483—494.
5. Moshe Morgensteren, Eli Shamir – Parallel Algorithms for Arithmetics, Irreducibility and Factoring of GFq-Polynomials. Stanford University – 1983.

УДК 004.7

Гарасим Ю.Р., Дудикевич В.Б.

IDS – ТЕХНОЛОГІЯ ЗАХИЩЕНИХ КОРПОРАТИВНИХ МЕРЕЖ ЗВ’ЯЗКУ

В цій статті розглядається технологія захищених корпоративних мереж зв’язку – IDS, наводиться загальна архітектура системи виявлення втручань та окремих її елементів, що узагальнює принципи роботи сучасних IDS та дозволяє порівняти їхні функціональні можливості.

Вступ

Одним з найважливіших завдань систем інформаційної безпеки є спостереження за даними, що передаються у мережі зв’язку, щоб знаходити потенційно небезпечні дії щодо певного хоста для унеможливлення доступу зловмисника до неї. Існує безліч різновидів подібних систем, особливо багато їх з'явилося останніми роками.

Метою виявлення втручань є спостереження за мережевими ресурсами, для визначення їхньої аномальної поведінки або неправильного використання. Розробки в цій галузі незабаром виведуть технології безпеки на абсолютно нову ланку автоматизованої безпеки.

Система виявлення втручань

Система виявлення втручань (англ. Intrusion Detection System (IDS)) — програмний або апаратний засіб, який призначений для виявлення факту несанкціонованого доступу (втручання або мережевої атаки) в комп'ютерну (телекомунікаційну) систему або мережу зв’язку (рис. 1).

Використання IDS допомагає досягти декількох цілей: виявити втручання або мережеву атаку; спрогнозувати можливі атаки в майбутньому та виявити вразливості для запобігання їхньому подальшому розвитку; здійснити документування існуючих загроз; забезпечити контроль якості адміністрування з погляду безпеки, особливо у великих і складних мережах зв’язку; отримати корисну інформацію про втручання, які мали місце,