УДК 681.3.06

В.И. Долгов[1], А.В. Неласая[2]

[1]*Харьковский национальный университет радиоэлектроники, Харьков*
[2]*Запорожский национальный технический университет, Запорожье*

ПРОГРАММНАЯ РЕАЛИЗАЦИЯ КРИПТОГРАФИЧЕСКИХ ОПЕРАЦИЙ НА ГИПЕРЭЛЛИПТИЧЕСКИХ КРИВЫХ

В работе рассматриваются особенности реализации криптографических операций на гиперэллиптических кривых на современных аппаратных платформах. Приводятся временные характеристики выполнения операций с дивизорами гиперэллиптических кривых в реализованных авторами криптографических библиотеках.

Ключевые слова: *криптосистема, гиперэллиптическая кривая, дивизор, конечное поле, скалярное умножение на гиперэллиптической кривой, 64-разрядная платформа.*

Введение

Большинство современных стандартов цифровой подписи, в частности украинский стандарт ДСТУ 4145-2002, основаны на арифметике в группе точек эллиптических кривых, определенных над конечными полями Галуа. Обобщением эллиптических кривых являются кривые более высокого рода – гиперэллиптические кривые[1]. Точки гиперэллиптической кривой не образуют группу, а в качестве групповой структуры используется якобиан кривой – факторгруппа дивизоров нулевой степени по подгруппе главных дивизоров. Основной особенностью гиперэллиптических кривых, делающей их привлекательным объектом для построения криптографических примитивов, является уменьшение размера элементов основного поля пропорционально роду кривой с сохранением заданного уровня криптостойкости. Это свойство вытекает из формулы Хассе-Вейля для границ порядка группы дивизоров:

$$\left\lceil \left(\sqrt{q}-1\right)^{2g} \right\rceil \le \#J/F_q \le \left\lfloor \left(\sqrt{q}+1\right)^{2g} \right\rfloor,$$

где q – характеристика поля, над которым определена кривая; g – род кривой.

Данное свойство позволяет отказаться от использования библиотек длинных чисел при реализации операций основного поля для практически приемлемых уровней стойкости.

Развитие вычислительной техники предоставляет новые возможности для повышения скорости реализации криптографических алгоритмов в частности, переход от 32-разрядного процессора к 64-разрядным позволяет повысить скорость реализации большинства алгоритмов. Однако существуют нюансы использования программного обеспечения инструментария разработчика на 64-битной платформе. В современных компиляторах C++ определен тип данных long long по стандарту имеющий размер как минимум 64 бита. В MS C++ этот тип имеет еще одно название «__int64». При использовании 32-битных процессоров удобство его использования является очевидным. Однако при использовании 64-битных машин этот тип также имеет размер 64 бита, в отличие от ожидаемых 128. Большинство компиляторов языка C++ не обладают встроенными возможностями обработки 128 разрядных целочисленных значений. Однако, компилятор GCC, используемый в основном под Linux, имеет встроенный тип __int128_t, что обеспечивает более гибкую обработку больших целочисленных значений.

Основной материал

Рассмотрим все возможные сочетания рода кривой с размером элементов основного поля, которые можно реализовать на современных процессорах без подключения внешних библиотек длинных чисел, и соответствующие им уровни криптографической стойкости, определяемые размером секретного ключа, который, в свою очередь, определяется порядком якобиана. Для реализации операций основного поля в общем случае необходимо, чтобы размер элементов поля был вдвое меньше размера регистров арифметико-логического устройства процессора. Это связано с тем, что размер результата умножения двух операндов вдвое превышает размер самих операндов. Однако его последующее приведение по модулю поля возвращает исходный размер. Другой вариант – ювелирная реализация операций умножения и приведения в поле на машинно-ориентированном языке ассемблер с привлечением дополнительных регистров – позволяет обойти эту проблему, хотя усложняет задачу программиста.

В табл. 1 приведены предельно достижимые уровни стойкости в зависимости от рода гиперэллиптической кривой, которые возможно достичь без подключения внешних библиотек длинных чисел на

компьютере с 64-разрядным процессором для компиляторов MS C++ и GCC двух вышеназванных вариантах реализации арифметики основного поля.

Для практического анализа скоростных показателей операций на кривых большого рода, а именно 5, 10, 20 и 40, авторами был разработан программный модуль HighCenusCurve на языке программирования C++ с использованием технологии шаблонов объектно-ориентированного программирования. В качестве шаблонного типа задается тип элементов основного поля, что позволило использовать один и тот же код для вычислений в конечных полях, размер элементов которых задавался типам unsigned long long, unsigned int, unsigned short, unsigned char в зависимости от рода кривой и требуемого уровня стойкости (табл. 2).

Таблица 1

Предельно достижимые уровни стойкости криптопреобразований на ГЭК

Род кривой	**MS C++ Длина модуля основного поля 32 бита(__int64)**	**GCC Длина модуля основного поля 64 бита (__int128_t)**	**MS C++ и ассемблер Длина модуля основного поля 64 бита (__int64)**
	Длина секретного ключа, бит		
1	32	64	64
2	64	128	128
3	96	192	192
4	128	256	256
5	160	320	320
10	320	640	640
20	640	1280	1280
40	1280	2560	2560

Таблица 2

Время скалярного умножения на кривых высоких родов для различных уровней стойкости

Род кривой	**Длина ключа**															
	160				**320**				**640**				**1280**			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
5	32	64	Unsigned long long	0,007	64	128			128	256			256	512		
10	16	32	unsigned int	0,484	32	64	Unsigned long long	0,228	64	128			128	256		
20	8	16	Unsigned short	11,08	16	32	Unsigned int	20,92	32	64	Unsigned long long	11,15	64	128		
40	4	8	Unsigned char	262,4	8	16	Unsigned short	619,5	16	32	Unsigned int	1129	32	64	Unsig-ned long long	636,4

Примечание: Заголовки столбцов для каждого уровня стойкости:

1 – Длина элементов основного поля; 2 – Требуемая длина элементов основного поля в машинном представлении;

3 – Тип данных языка C++ для хранения элементов основного поля; 4 – Время скалярного умножения (сек.)

Как видно из таблицы, время скалярного умножения значительно возрастает с увеличением рода кривой. К тому же кривые рода большего 4 не являются криптографически стойкими. Исходя из этого, для практического использования авторами были реализованы криптосистемы на гиперэллиптических кривых второго и третьего родов на 64-разрядной аппаратной платформе. Арифметика основного поля реализована на языке программирования ассемблер.

Для кривых второго рода элементы основного поля имеют размер до 128 бит. При программной реализации один элемент занимает два 64-разрядных регистра. Это позволяет достичь уровня стойкости, соответствующего длине ключа до 256 бит. Для кривых третьего рода элементы основного поля имею размер до 64 бит. Это позволяет достичь уровня стойкости, соответствующего длине ключа до 192 бит. Такая реализация позволила существенно улучшить скоростные характеристики операций сложения и дублирования дивизоров гиперэллиптической кривой, несмотря на сложность самих формул. Для реализации были выбраны аффинные координаты, поскольку отношение времени выполнения операций умножения и инверсии в основном поле равно около 2,5 и нет смысла избегать инвер-

сии за счет увеличения количества операций умножения почти вдвое, как это происходит при переходе к проективным координатам.

В табл. 3 приведено время операций с дивизорами на гиперэллиптической кривой третьего рода. В табл. 4 приведено время скалярного умножения на кривой третьего рода с помощью стандартного бинарного алгоритма с использованием итерационной формулы Кантора и с использованием явных формул сложения и дублирования дивизоров.

Таблица 3

Время выполнения операций с дивизорами на гиперэллиптической кривой третьего рода

Длина модуля кривой, бит	Длина ключа, бит	Сложение дивизоров, сек.		Дублирование дивизоров, сек.	
		Итерационная формула Кантора	Явная формула	Итерационная формула Кантора	Явная формула
56	168	6,25E-04	1,60E-05	6,01E-04	1,71E-05
58	174	6,21E-04	1,66E-05	6,08E-04	1,87E-05
60	180	6,35E-04	1,67E-05	6,09E-04	1,91E-05
62	186	6,29E-04	1,69E-05	6,08E-04	1,90E-05
64	192	6,38E-04	1,69E-05	6,18E-04	1,92E-05

Таблица 4

Время скалярного умножения на кривой третьего рода
с помощью стандартного бинарного алгоритма

Длина модуля кривой, бит	Длина ключа, бит	Итерационная формула Кантора, с.	Явные формула, с.
56	168	1,60E-01	4,46E-03
58	174	1,65E-01	4,62E-03
60	180	1,72E-01	4,79E-03
62	186	1,78E-01	4,98E-03
64	192	1,84E-01	5,26E-03

Таким образом, использование якобиана гиперэллиптических кривых в качестве групповой структуры для криптографических преобразований является целесообразным при реализации на 64-разрядной платформе, что позволяет отказаться от использования внешних библиотек длинных чисел для приемлемых уровней стойкости даже для кривых 2 и 3 родов. Кривых более высоких родов, не являющиеся криптографически стойкими, могут быть эффективно использованы, например, в целях криптоанализа.

Разработанное программное обеспечение успешно прошло тестовые испытания и может быть использовано для реализации протоколов цифровой подписи и направленного шифрования.

Список литературы

1. Menezes A. An Elementary Introduction to Hyperelliptic Curves [Электронный ресурс] / A. Menezes, Wu.Y. Zuccherato // R. : Published as Technical Report CORR 96-19 Department of C&O University of Waterloo. – Ontario: Canada, 1996. – P. 1-35. – Режим доступа к документу: www.cacr.math.uwaterloo.ca/techreports/1997/corr96-19.ps.

Поступила в редколлегию 22.03.2010

Рецензент: д-р техн. наук, проф. Л.М. Карпуков, Запорожский национальный технический университет, Запорожье.

ПРОГРАМНА РЕАЛІЗАЦІЯ КРИПТОГРАФІЧНИХ ОПЕРАЦІЙ НА ГІПЕРЕЛІПТИЧНИХ КРИВИХ

В.І. Долгов, Г.В. Неласа

В роботі розглядаються особливості реалізації криптографічних операцій на гіпереліптичних кривих на сучасних апаратних платформах. Наводяться часові характеристики виконання операцій з дивізорами гіпереліптичних кривих в реалізованих авторами криптографічних бібліотеках.

Ключові слова: *криптосистема, гіпереліптична крива, дивізор, кінцеве поле, скалярне множення на гіпереліптичній кривій, 64-розрядна платформа.*

SOFTWARE FOR CRYPTOGRAPHIC OPERATIONS ON HYPERELLIPTIC CURVES

V.I. Dolgov, G.V. Nelasa

The features of realization software for cryptographic operations on hyperelliptic curves on modern hardware are considered. The timing for cryptographic operations with divisors of hyperelliptic curves in author's realization of cryptographic libraries are illustrated.

Keywords: *cryptosystem, hyperelliptic curve, divisor, eventual field, scalar increase on a hyperelliptic curve, 64-bit platform.*