© Mis'ko V. Acceleration of Fermat's factorization method by decimation method with use of several bases // Ukrainian Scientific Journal of Information Security, 2015, vol. 21, issue 1, p. 64-68.

УСКОРЕНИЕ МЕТОДА ФЕРМА МЕТОДОМ ПРОРЕЖИВАНИЯ С ИСПОЛЬЗОВАНИЕМ НЕСКОЛЬКИХ БАЗ

Виталий Мисько

Институт проблем моделирования в энергетике им. Г.Е. Пухова НАН Украины, Украина

МИСЬКО Виталий Николаевич

Год и место рождения: 1991 год, г. Евпатория, АР Крым, Украина.
Образование: Институт специальной связи и защиты информации НТУУ «КПИ», 2013 год.
Должность: аспирант с 2014 года.
Научные интересы: информационная безопасность, криптография, математическое моделирование.
Публикации: две статьи.
E-mail: vitalik560@yandex.ru

Аннотация. *В основах существующих методов факторизации (метод решета числового поля, метод квадратичного решета) лежит принцип факторизации методом Ферма. Ускорение метода Ферма разложения чисел вида* $N=p*q$, *где* p *и* q *простые, на множители можно достичь за счет прореживания пробных значений, путем перехода к модульному уравнению* $x^2 \bmod B = (N \bmod B + y^2 \bmod B) \bmod B$, *где* B *некоторый модуль (база). Эффективность такого подхода увеличивается с увеличением базы* B. *Рост базы в свою очередь ведет к росту требуемого объема памяти, для хранения допустимых значений* $x \bmod B$ *и росту вычислительной сложности. Рост вычислительной сложности в дополнении с ростом требуемого объёма памяти превышает рост эффективности ускорения. В связи трудностями, возникающими при использовании модулей большого размера (*B*), предлагается использовать несколько модулей меньшего размера. При этом возникает задача эффективного их выбора. В данной работе проведен анализ эффективности, при использовании просеивания по двум базам* $b1$ *и* $b2$, *и даны сравнительные характеристики со случаем, когда используются просеивание по одной* $B = b1*b2$. *Определены условия эффективности просеивания для вариантов по одной и по двум базам.*

Ключевые слова: *ассиметричная криптография, факторизация, метод Ферма, прореживание, ускорение.*

Вступление

С момента появления в публичном доступе, алгоритм шифрования RSA (Rivest, Shamir, Adleman) стал одним из наиболее широко используемых асимметричных криптоалгоритмов на сегодняшний день. Этот криптоалгоритм может быть реализован в программных продуктах, встроенном программном обеспечении, аппаратных средствах [1]. Распространение этого алгоритма в информационно телекоммуникационных системах (ИТС) вызывает интерес к его криптоанализу, чему посвящено ряд публикаций [2, 3]. На данный момент все известные методы, которые применяются при криптоанализе RSA, работают только в отдельных случаях его реализации и, в общем случае по сложности либо приравниваются, либо превышают, по сложности, метод факторизации. Из чего следует, что криптостойкость данного алгоритма напрямую зависит от скорости работы метода факторизации чисел. Существующие методы факторизации (метод решета числового поля, метод квадратичного решета), основываются на ряде фундаментальных соотношений из классического алгоритма Ферма. Опираясь на это можно предположить, что усовершенствование метода Ферма, повлияет на скорость работы выше указанных методов. Поэтому исследование способов ускорения метода Ферма факторизации чисел вида $N=p*q$ является актуальным. Один из таких подходов [4] требует дополнительного изучения.

Постановка задачи

При использовании способа ускорения метода Ферма факторизации чисел вида $N=p*q$, где p и q простые, методом прореживания [4], определен рост эффективности прореживания с ростом базы B $(B \in N)$. Для реализации метода необходимо использовать три массива с количеством элементов равным B, и имеющих тип, который позволяют производить операции возведения в квадрат этих элементов. Из этого следует, что при использовании базы даже порядка 10^6 требуется 22 Гигабайта оперативной памяти. Что порождает проблему выделения необходимого объема памяти, для хранения допустимых значений $x \bmod B$. В связи с выше упомянутыми проблемами, возникающими

при использовании модулей большого размера (B), предлагается использовать более одного модуля меньшего размера. При этом возникает задача эффективного их выбора, чему посвящено настоящее исследование. В работе исследуется взаимное влияние баз B на эффективность метода факторизации.

Схожий метод описан, как метод решета [5]. Но в методе решета используется следующая формула: $y^2 \bmod B = (x^2 \bmod B - N \bmod B) \bmod B$, где

$$y^2 - x^2 = N. \quad (1)$$

То есть анализ остатков происходил для y. Следует отметить, что использование метода Ферма эффективно при относительно близких значениях p и q. В этом случае изменение x на единицу приводит к изменению y на величину порядка $O(N^{1/4})$. Поэтому использование прореживания по x, предложенное в работе [4], где допустимые значения определяются для x из соотношения

$$x^2 \bmod B = (N \bmod B + y^2 \bmod B) \bmod B, \quad (2)$$

оказывается более эффективным.

Поиск эффективных модулей *B*

В рамках данного исследования будем полагать решенной задачу поиска эффективных (в смысле снижения числа пробных значений k в методе Ферма) модулей B, если определены правила, с помощью которых можно найти такие модули.

Пусть XN(B, NmodB) число значений xmodB, при которых уравнение (2) имеет хотя бы одно решение, для конкретных B и NmodB. Для известных B и NmodB определим коэффициент уменьшения числа пробных значений k в методе Ферма как отношение

$$\mathrm{Z(\ B,\ N\ mod\ B)} = \frac{B}{XN(B, N \bmod B)}, \quad (3)$$

которое, в дальнейшем будем называть ускорением метода Ферма(далее ускорение). Определим функцию, описывающую обобщенную характеристику ускорения $Z_{ср}(B)$ – среднее ускорение по всем вариантам значений NmodB [4].

Среднее ускорение определяется по формуле

$$Z(B)_{ср} = \frac{B * \phi(B)}{\sum_{i=1}^{K(B)} (N_i \bmod B)}, \quad (4)$$

где $\phi(B)$ - число Эйлера.

Поскольку величина $Z_{ср}(B)$ существенно зависит от структуры множителей B [4], возьмем модуль большого размера B, как произведение двух модулей малого размера (b1, b2). Для баз вида B = b1*b2 исследуем характеристику ускорения и сравним ее с ускорениями для баз b1 и b2, а именно сравним величины $Z_{ср}(B)$ и $(Z_{ср}(b1) * Z_{ср}(b2))$.

Зададим общий вид баз b1 и b2 формулой

$$b1 = 2^{k1} * c * b_{01},$$
$$b2 = 2^{k2} * c * b_{02}, \quad (5)$$

где:

– c, b_{01}, b_{02} – не четные;

– $НОД(b_{01}, b_{02}) = 1$;

– $k2 \ge k1, k1 \ge 0$.

Запишем общую формулу ускорения для баз b1 и b2 опираясь на структуру баз, описанной выше (5)

$$Z_{ср}(b1) = Z_{ср}(2^{k1}) * Z_{ср}(c) * Z_{ср}(b_{01}), \quad (6)$$

$$Z_{ср}(b2) = Z_{ср}(2^{k2}) * Z_{ср}(c) * Z_{ср}(b_{02}). \quad (7)$$

Отсюда рассчитаем произведение ускорений для двух баз:

$$Z_{ср}(b1) * Z_{ср}(b2) = \\ = Z_{ср}(2^{k1}) * Z_{ср}(2^{k2}) * (Z_{ср}(c))^2 * Z_{ср}(b_{01}) * Z_{ср}(b_{02}). \quad (8)$$

Определим отношение ускорения по двум базам (b1, b2) и ускорения базы, которая является произведением первых двух:

$$\frac{Z_{ср}(b1 * b2)}{Z_{ср}(b1) * Z_{ср}(b2)} = \\ = \frac{Z_{ср}(2^{k1+k2}) * Z_{ср}(c^2) * Z_{ср}(b_{01}) * Z_{ср}(b_{02})}{Z_{ср}(2^{k1}) * Z_{ср}(2^{k2}) * (Z_{ср}(c))^2 * Z_{ср}(b_{01}) * Z_{ср}(b_{02})}. \quad (9)$$

Сократим общие множители:

$$\frac{Z_{ср}(b1 * b2)}{Z_{ср}(b1) * Z_{ср}(b2)} = \frac{Z_{ср}(2^{k1+k2}) * Z_{ср}(c^2)}{Z_{ср}(2^{k1}) * Z_{ср}(2^{k2}) * (Z_{ср}(c))^2}. \quad (10)$$

Из уравнения (10) видно, что для оценки эффективности ускорения исследуемых баз необходимо учитывать не всю базу а только множитель c и показатели степени двойки k1 и k2.

Для оценки эффективности рассмотрим все возможные варианты:

1.Наибольший общий делитель баз b1 и b2 – четный:

1.1. НОД (b1, b2) $= 2^k$, где $k = 1$;

1.2. НОД (b1, b2) $= 2^k * c$, где $k = 1, c = 3$;

1.3. НОД (b1, b2) $= 2^k * c$, где $k = 1, c > 3, c - простое$;

1.4. НОД (b1, b2) $= 2^k$, где $k = 2$;

1.5. НОД (b1, b2) $= 2^k * c$, где $k = 2, c \ge 1, c - простое$;

1.6. НОД (b1, b2) $= 2^k * c$, где $k \ge 3, c \ge 1, c - простое$.

2. Наибольший общий делитель баз b1 и b2 – не четный.

Наибольший общий делитель баз *b1* и *b2* – четный

В случае, когда НОД (b1, b2) - четное, особое влияние на ускорение оказывает k1, k2 - показатели степеней множителей, на которые разлагается b1 и b2 (4). Поэтому, дальнейшее рассмотрение баз опирается на показатель степени двойки в базах.

Рассмотрим зависимость эффективности ускорения баз от вариантов показателей степени двойки:

1. $k1 = 1, k2 \ge k1$. Также необходимо учитывать значение c. Рассмотрим вариант, когда $c = 1$. Из заданных условий следует, что $НОД(b1, b2) = 2$. Результаты исследований (табл. 1) показали, что для таких баз всегда верно следующее утверждение $Z_{ср}(b1 * b2) \ge (Z_{ср}(b1) * Z_{ср}(b2))$.

Таблица 1

Значения $(Z_{ср}(b1)*Z_{ср}(b2))$ при $НОД(b1,b2)=2$

b1	$Z_{ср}$(b1)	b2	$Z_{ср}$(b2)	b1*b2	$Z_{ср}$(b1*b2)	$Z_{ср}$(b1)* $Z_{ср}$(b2)
16 (2*2*2*2)	4,00	82 (2*41)	2,00	1312	9,14	8,00
44 (2*2*11)	4,00	70 (2*5*7)	4,00	3080	21,33	16,00
60 (2*2*3*5)	8,00	86 (2*43)	2,00	5160	21,33	16,00
62 (2*31)	2,00	80 (2*2*2*2*5)	8,00	4960	18,29	16,00
74 (2*37)	2,00	80 (2*2*2*2*5)	8,00	5920	18,29	16,00
22 (2*11)	2,00	48 (2*2*2*2*3)	8,00	1056	18,29	16,00

Рассматривая вариант при $k1=1, k2 \geq k1$, следует исследовать ситуацию, когда $c>1$. В этом случае необходимо наложение дополнительных условий:

1.1 Разберем случай, когда $c=3$:

1.1.1 Если $НОД(2^{k1-1}*b_{01}*2^{k2-1}*b_{02},18)<18$, тогда $Z_{ср}(b1*b2)>(Z_{ср}(b1)*Z_{ср}(b2))$ (табл. 2);

1.1.2 Если $НОД(2^{k1-1}*b_{01}*2^{k2-1}*b_{02},18)=18$, тогда $Z_{ср}(b1*b2)<(Z_{ср}(b1)*Z_{ср}(b2))$;

Таблица 2

Значения $(Z_{ср}(b1)*Z_{ср}(b2))$ при $НОД(b1,b2)=6$

B1	$Z_{ср}$(b1)	b2	$Z_{ср}$(b2)	b1*b2	$Z_{ср}$(b1*b2)	$Z_{ср}$(b1)* $Z_{ср}$(b2)
6 (2*3)	2,00	42 (2*3*7)	4,00	252	14,40	8,00
24 (2*2*2*3)	5,33	42 (2*3*7)	4,00	1008	28,80	21,33
30 (2*3*5)	4,00	42 (2*3*7)	4,00	1260	28,80	16,00
42 (2*3*7)	4,00	78 (2*3*13)	4,00	3276	28,80	16,00
60 (2*2*3*5)	8,00	78 (2*3*13)	4,00	4680	38,40	32,00
78 (2*3*13)	4,00	90 (2*3*3*5)	7,20	7020	33,23	28,80

Таблица 3

Значения $(Z_{ср}(b1)*Z_{ср}(b2))$ при $НОД(b1,b2)=6$

B1	$Z_{ср}$(b1)	b2	$Z_{ср}$(b2)	b1*b2	$Z_{ср}$(b1*b2)	$Z_{ср}$(b1)* $Z_{ср}$(b2)
6 (2*3)	2,00	72 (2*2*2*3*3)	9,60	432	16,62	19,20
6 (2*3)	2,00	36 (2*2*3*3)	7,20	216	11,08	14,40
24 (2*2*2*3)	5,33	90 (2*3*3*5)	7,20	2160	33,23	38,40
66 (2*3*11)	4,00	72 (2*2*2*3*3)	9,60	4752	33,23	38,40
36 (2*2*3*3)	7,20	66 (2*3*11)	4,00	2376	22,15	28,80
18 (2*3*3)	3,60	84 (2*2*3*7)	8,00	1512	22,15	28,80

1.2 Когда c – простое и $c>3$ (табл. 4), в таких ситуациях:

1.2.1 Если $НОД(2^{k1-1}*b_{01}*2^{k2-1}*b_{02},2*c)=2$ или $НОД(2^{k1-1}*b_{01}*2^{k2-1}*b_{02},2*c)=2*c$, тогда $Z_{ср}(b1*b2)<(Z_{ср}(b1)*Z_{ср}(b2))$;

1.2.2 В остальных случаях - $Z_{ср}(b1*b2)>(Z_{ср}(b1)*Z_{ср}(b2))$;

Таблица 4

Значения $(Z_{ср}(b1)*Z_{ср}(b2))$ при $НОД(b1,b2)=10$

B1	$Z_{ср}$(b1)	b2	$Z_{ср}$(b2)	b1*b2	$Z_{ср}$(b1*b2)	$Z_{ср}$(b1)* $Z_{ср}$(b2)
10 (2*5)	2,00	90 (2*3*3*5)	7,20	900	21,18	14,40
40 (2*2*2*2*5)	5,33	70 (2*5*7)	4,00	2800	23,53	21,33
30 (2*3*5)	4,00	50 (2*5*5)	2,94	1500	12,35	11,76
60 (2*2*3*5)	8,00	70 (2*5*7)	4,00	4200	31,37	32,00
10 (2*5)	2,00	80 (2*2*2*2*5)	8,00	800	13,45	16,00
50 (2*5*5)	2,94	60 (2*2*3*5)	8,00	3000	16,46	23,53

1.3 Когда c – составное, всегда выполняется неравенство $Z_{ср}(b1*b2)\leq(Z_{ср}(b1)*Z_{ср}(b2))$.

2.Рассмотрим следующую показатель степени двойки, $k1=2, k2 \geq k1$. Как и предыдущем варианте следует выделить случай, когда:

2.1 $c=1$. То есть, когда $НОД(b1,b2)=4$ (табл. 5). Определены условия ускорений в таких случаях:

2.1.1 если $k_2 \geq 3$, то $Z_{ср}(b1*b2)<(Z_{ср}(b1)*Z_{ср}(b2))$;

2.2 если $k_2<3$, то $Z_{ср}(b1*b2)=(Z_{ср}(b1)*Z_{ср}(b2))$. При $c>1$. Необходимо провести анализ. Для данного случая всегда справедливы неравенства:

$$Z_{ср}(2^{k1+k2})<Z_{ср}(2^{k1})*Z_{ср}(2^{k2}), \quad (11)$$

$$Z_{ср}(c^2)<(Z_{ср}(c))^2. \quad (12)$$

Неравенства (11), (12) справедливы для всех k2 ($k2 \in N$) и c, с учетом предыдущих условий. Исходя из этого, можно утверждать, что при $k1=2$, $c>1$, справедливо неравенство – $Z_{ср}(b1*b2)<(Z_{ср}(b1)*Z_{ср}(b2))$.

Таблица 5

Значения $(Z_{ср}(b1) * Z_{ср}(b2))$ при $НОД(b1,b2) = 4$

B1	$Z_{ср}$(b1)	b2	$Z_{ср}$(b2)	b1*b2	$Z_{ср}$(b1*b2)	$Z_{ср}$(b1)* $Z_{ср}$(b2)
4 (2*2)	2,00	16 (2*2*2*2)	4,00	64	5,33	8,00
8 (2*2*2)	2,67	92 (2*2*23)	4,00	736	9,14	10,67
68 (2*2*17)	4,00	88 (2*2*2*11)	5,33	5984	18,29	21,33
12 (2*2*3)	4,00	76 (2*2*19)	4,00	912	16,00	16,00
20 (2*2*5)	4,00	84 (2*2*3*7)	8,00	1680	32,00	32,00
60 (2*2*3*5)	8,00	92 (2*2*23)	4,00	5520	32,00	32,00

3. При $k1 \geq 3, k2 \geq k1$, можно воспользоваться уравнением (10) из анализа которого, при данных условиях можно сделать вывод, что при любых c, k1,k2 справедливо неравенство $Z_{ср}(b1 * b2) < (Z_{ср}(b1) * Z_{ср}(b2))$.

Наибольший общий делитель нечетный

В случае, когда базы b1 и b2 взаимно простые, ускорения для исследуемых баз равны.

Таблица 6

Значения $(Z_{ср}(b1) * Z_{ср}(b2))$ при $НОД(b1,b2) = 1$

b1	$Z_{ср}$(b1)	b2	$Z_{ср}$(b2)	b1*b2	$Z_{ср}$(b1*b2)	$Z_{ср}$(b1)* $Z_{ср}$(b2)
18 (2*3*3)	3,60	25 (5*5)	2,94	450	10,59	10,59
45 (3*3*5)	7,20	98 (2*7*7)	2,65	4410	19,07	19,07
77 (7*11)	4,00	96 (2*2*2*2*2*3)	9,14	7392	36,57	36,57

В остальных случаях, когда $НОД(b1,b2) > 1$, имеет место следующее неравенство:

$$Z_{ср}(b1 * b2) \leq (Z_{ср}(b1) * Z_{ср}(b2)).$$

Общие случаи по сравнению ускорений исследуемых баз

$Z_{ср}(b1 * b2) = (Z_{ср}(b1) * Z_{ср}(b2))$;

– базы b1 и b2 взаимно простые;

– НОД (b1, b2) $= 2^k$, *где* $k = 2$ но только тогда, когда ни одна из баз не делится нацело на 2^3;

$Z_{ср}(b1 * b2) > (Z_{ср}(b1) * Z_{ср}(b2))$;

– НОД (b1, b2) $= 2^k$, *где* $k = 1$;

– НОД (b1, b2) $= 2^k * c$, *где* $k = 1, c = 3$, Если $НОД(2^{k1-1} * b_{01} * 2^{k2-1} * b_{02}, 18) < 18$;

– НОД (b1, b2) $= 2^k * c$, *где* $k = 1, c > 3, c - простое$, при условиях:

– $НОД(2^{k1-1} * b_{01} * 2^{k2-1} * b_{02}, 2 * c) \neq 2$;

– $НОД(2^{k1-1} * b_{01} * 2^{k2-1} * b_{02}, 2 * c) \neq 2 * c$;

$Z_{ср}(b1 * b2) < (Z_{ср}(b1) * Z_{ср}(b2))$;

– НОД (b1, b2) $= 2^k * c$, *где* $k = 1, c = 3$, Если $НОД(2^{k1-1} * b_{01} * 2^{k2-1} * b_{02}, 18) = 18$;

– НОД (b1, b2) $= 2^k * c$, *где* $k = 1, c > 3, c - простое$, при условиях:

– $НОД(2^{k1-1} * b_{01} * 2^{k2-1} * b_{02}, 2 * c) = 2$;

– $НОД(2^{k1-1} * b_{01} * 2^{k2-1} * b_{02}, 2 * c) = 2 * c$;

– НОД (b1, b2) $= 2^k$, *где* $k = 2$ и хотя бы одна из баз делится нацело на 2^3;

– НОД(b1, b2) = 2^k*c, где k = 2, c > 1

НОД (b1, b2) $= 2^k * c$, *где* $k = 2, c > 1$;

– НОД(b1, b2) $\geq 2^k$*c, где $k \geq 3$, $c \geq 1$;

– НОД(b1, b2) не четное.

Выводы

Из анализа различных вариантов модулей $b1$ и $b2$, и проведенных численных экспериментов, можно сделать вывод, что в случае, когда удовлетворяется одно из приведенных условий:

– НОД (b1, b2) $= 2^k$, *где* $k = 1$;

– НОД (b1, b2) $= 2^k * c$, *где* $k = 1, c = 3$, Если $НОД(2^{k1-1} * b_{01} * 2^{k2-1} * b_{02}, 18) < 18$;

– НОД (b1, b2) $= 2^k * c$, *где* $k = 1, c > 3, c - простое$, при условиях:

– $НОД(2^{k1-1} * b_{01} * 2^{k2-1} * b_{02}, 2 * c) \neq 2$;

– $НОД(2^{k1-1} * b_{01} * 2^{k2-1} * b_{02}, 2 * c) \neq 2 * c$;

где исследуемые базы вида

$$b1 = 2^{k1} * c * b_{01}, b2 = 2^{k2} * c * b_{02},$$

справедливо следующее неравенство $Z_{ср}(b1 * b2) > (Z_{ср}(b1) * Z_{ср}(b2))$, из чего следует, что в таких условиях эффективнее использовать базу B. Во всех же остальных случаях эффективнее использовать, просеивание по двум базам $b1$ и $b2$.

Литература

[1] Жилин А.В. Использование RSA алгоритма для обеспечения задач криптографической защиты информации в современных информационно-телекоммуникационных системах / А.В. Жилин, А.В. Корнейко, В.В. Мохор // Захист інформації. – 2013. – Т.15, № 3. – С. 225-231.

[2] Song Y. Yan. Cryptanalytic attacks on RSA / Song Y. Yan – Springer Science and Business Media, Inc. 2008. – P.255

[3] Горбенко И.Д. Анализ каналов уязвимости системы RSA / И.Д. Горбенко, В.И. Долгов, А.В. Потий, В.Н. Федорченко // Безопасность информации. – 1995. – № 2. – С.22-26.

[4] Винничук С.Д., Жилин А.В., Мисько В.Н. Ускорение метода ферма факторизации чисел вида n = pq, где p и q простые, методом прореживания

[5] Кнут Д. Искуство программирования том 2 (3-е изд.) 2001 – 423 с.

УДК 511:003.26.09 (045)

Міська В. М. Прискорення методу Ферма методом проріджування з використання декількох баз

Анотація. *В основі існуючих методів факторизації (метод решета числового поля, метод квадратичного решета) полягає принцип факторизації методом Ферма. Прискорення методу Ферма розкладання чисел виду N = pq, де p та q прості, на множники можна досягнути за рахунок проріджування пробних значень, шляхом переходу до модульного рівняння* $x^2 \bmod B = (N \bmod B + y^2 \bmod B) \bmod B$*, де B деяка основа модуля (база). Ефективність такого підходу збільшується тоді, коли зростає база B. Збільшення бази у свою чергу веде до зростання необхідної кількості пам'яті, для збереження допустимих значень xmodB та зростання обчислювальної складності. Збільшення обчислювальної складності разом зі збільшенням необхідного об'єму пам'яті перевищує зростання ефективності прискорення. У зв'язку з проблемами, які виникли при використанні основ більшого розміру (B), пропонується використовувати декілька основ меншого розміру. При цьому виникає задача ефективного їх вибору. У даній роботі проведено аналіз ефективності, при використанні просіювання за двома базами b1 та b2, та наведені порівнювальні характеристики з варіантом, коли використовується просіювання за однією базою B=b1*b2. Визначені умови ефективності просіювання для варіантів за однією та за двома базами.*

Ключові слова: *асиметрична криптографія, факторизація, метод Ферма, проріджування, прискорення.*

Mis'ko V. Acceleration of Fermat's factorization method by decimation method with use of several bases

Abstract*. Fermat's factorization method is the basis of the existing methods of factorization (General number field sieve, Quadratic sieve). We can achieve acceleration of Fermat's factorization method, by moving to modular equation* $x^2 \bmod B = (N \bmod B + y^2 \bmod B) \bmod B$ *where B some module (base). The effectiveness of this approach increases when increasing base* B*. When you increased base, you must to increase the required amount of memory to store permissible values* $xmodB$ *and increase computational complexity. The increase in computational complexity with increasing the required memory bigger than growth of the efficiency of acceleration. We have many problems arising when using larger bases (*B*), that is why we propose to use multiple small bases. We have a problem, of effective choice of the bases. This paper analyzes the efficiency, the use of screening on two bases* $b1$ *and* $b2$*, and give comparable characteristics of the option, when used for screening one base* $B = b1*b2$*. Described conditions for the effectiveness of screening options for one and two bases.*

Key words: *public key cryptography, factorization, Fermat's factorization method, decimation, acceleration.*

Отримано *17 лютого 2015 року*, затверджено редколегією *4 березня 2015 року*