

Secret Key Leakage caused by Hamming-weight Timing Analysis on Modular Exponentiation

Dr. Mykola Karpinskyy
University of Bielsko-Biala
Bielsko-Biala, Poland

Ph.D. Student Lesya Vasytkiv
Computer Informational
Technology Faculty
Ternopil State Economy
University
Ternopil, Ukraine

M.S. Marcin Gizycki
Head of Academy of
Information Centre and
coordinator CNA, AATP,
CCAI
University of Bielsko-Biala
Bielsko-Biala, POLAND

Abstract – *In this Research Report the investigation of the performance of modern modular exponentiation methods has been done. There is shown that the β -ary method is the best for use in the asymmetric cryptosystems. This paper represents the mathematical background to estimate the secret information leakage risks during timing analysis the most general modern modular exponentiation methods. The comparison of the leakage risk of those methods has been done. Possible countermeasures to decrease the secret information leakage risk level have also been proposed. The theoretical material obtained in the paper can be useful for the investigation of the risk of successful DPA attack.*

Keywords: Timing Analysis, Performance, Secret Information Leakage Risk, Modular Exponentiation Methods.

1. Introduction

One of the perspective ways of the asymmetric cryptosystems research is analysis of exponentiation algorithms. These algorithms are used as basic operation of asymmetric cryptosystem ciphering. There are a number of different modern algorithms for modular exponentiation. In this paper algorithms [4, 5] based on binary method, beta-ary method and method of shift window have been investigated. Some basic theoretical relations of these methods have been considered. These relations allow making further investigation of the discussed modular exponentiation methods. One of the important parameters of comparing different methods is a performance. The investigation of the performance parameters of the methods has been done.

Side-channel attacks (SCA) are special way of cryptanalysis of modern means of cryptosystems [5].

Timing analysis (TA) is one of the simplest and easy-to-implement side-channel analysis (SCA) attacks. Such kind of attack can be very effective when the eavesdropper has the access to the encrypt tools [2]. So, development of the countermeasures to decrease the secret information leakage risk is a very important question. In this paper also the investigation of the resistance modular exponentiation methods to timing analysis has been done. The theoretical material obtained in the paper can be useful for the investigation of the risk of successful DPA attack.

2. Modern Exponentiation Algorithms

Most of the modern exponentiation algorithms can be realized in two variants accordingly to the directions of bit reading: from left to right (LTR) and from right to left (RTL) [4, 5]. Below the mathematical models of binary method, beta-ary method and method of shift window for both reading directions are presented.

In general situation all modular exponentiation algorithms get the numbers x , n and m as input and return the value $x^n \bmod m$ as a result.

For binary method the binary representation of number n is used. Then the representation of x^n from left to right is $x^n = x^{(n_{k-1} \dots n_0)_2} = (((((x^{n_{k-1}})^2 x^{n_{k-2}})^2 \dots x^{n_1})^2 x^{n_0})^2 \dots x^{n_0})^2$ and from right to left $x^n = x^{(n_{k-1} \dots n_0)_2} = (x^{2^0})^{n_0} \cdot \dots \cdot (x^{2^{k-1}})^{n_{k-1}}$.

In the beta-ary method number n is represented with base $\beta = 2^w$. So, for this method from left to right:

$x^n = x^{(n_{k-1} \dots n_0)_\beta} = (((\dots((x^{n_{k-1}})^\beta x^{n_{k-2}})^\beta \dots x^{n_1})^\beta x^{n_0})$ and from right to left:

$$x^n = x^{(n_{k-1} \dots n_0)_\beta} = (x^{\beta^0})^{n_0} \cdot \dots \cdot (x^{\beta^{k-1}})^{n_{k-1}} = \prod_{w=1}^{\beta-1} \left(\prod_{\{i|n_i=w\}} x^{\beta^i} \right)^w.$$

The method of shift window is based on dividing of binary representation of $n = (w_{i-1} \dots w_0)_2$ into the blocks with variable width. Taking this into account the mathematical model for method from left to right direction can be defined as $x^n = (((\dots((x^{(w_{i-1})_2})^{2^{|w_{i-1}|}} \cdot (x^{(w_{i-2})_2})^{2^{|w_{i-2}|}}) \dots) \cdot x^{(w_0)}.$

From right to left: $x^n = \prod_{i=0}^{l-1} x^{(w_i)_2} 2^{2^i} = \prod_{w \in \{1, 3, \dots, 2^w - 1\}} \left(\prod_{\{i|(w_i)_2=w\}} x^{2^i} \right)^w$, where $l_i = \sum_{j=0}^{i-1} |w_j|$, $|w|$ —the width of the

longest odd window (window which starts and finished with binary ‘1’).

The relate algorithms of binary, beta-ary methods and method of shift window are shown in [4, 5].

3. Investigation of the Performance

In the Table 1 there are definitions of the used marks for the operations of the investigated modular exponentiation algorithms.

Table 1. Definitions of the Timing Marks

Operation	Timing Mark
$a := b$	c
$z := x \bmod m$	b
$(x \cdot x) \bmod m$	r
$(x \cdot y) \bmod m$	s
$y^\beta \bmod m$	d
$n = (n_{k-1} \dots n_0)_2$	t
Finding the longest sequence of binary ‘1’ $n_i \dots n_j$	q

Below are obtained by the author the mathematical models to simulate the time requested to perform the modular exponentiation algorithms.

Table 2. The mathematical models to simulate the time requested to perform the modular exponentiation algorithms.

The modular exponentiation algorithms	From left to right	From right to left
Binary method	$T1_{(n)} = t + c + \lceil \log n \rceil \cdot r + H(n) \cdot s$	$T2_{(n)} = t + c + b + \lceil \log n \rceil \cdot r + H(n) \cdot s$
Beta-ary method	$T3_{(n,w)} = t + 2c + \frac{\lceil \log n \rceil}{w} \cdot d + \left(\frac{\lceil \log n \rceil}{w} + 2^w - 1 \right) \cdot s$	$T4_{(n,w)} = t + (2^w + 1) \cdot c + \frac{\lceil \log n \rceil}{w} \cdot d + \left(\frac{\lceil \log n \rceil}{w} - W_0 + 2^{w+1} - 2 \right) \cdot s$
Shift window method	$T5_{(n,w_i w_i)} = t + b + (2 + \lceil \log n \rceil - H(n) + p) \cdot c + (\lceil \log n \rceil + 1) \cdot r + (2^{ w_i -1} - 1 + p) \cdot s + p \cdot q$	$T6_{(n,w_i w_i)} = t + b + (2^{ w_i } - 2 + p) \cdot s + p \cdot q + (2^{2^{ w_i -2}} + \lceil \log n \rceil - H(n) + p) \cdot c + (\lceil \log n \rceil + 2^{ w_i -1} - 1) \cdot r$

Investigation of the modular exponentiation algorithms shows that the time requested to perform the algorithm depends on the number of bits in the binary representation of degree n .

On the Fig.1 it is presented the dependence of the performance of the discussed modular exponentiation method from $\lceil \log n \rceil$ value. Analysis of Fig.1 shows that the beta-ary method has the highest performance level.

On the Fig.2 the dependence of the performance value onto the Hamming weight (that is the main characteristic

of the encryption key) of the mathematical models beta-ary method from right to left has been presented. The analysis of the figure shows that some of the plots weakly depend on the Hamming weight, but others strongly. This information allows us to estimate the resistance of modular exponentiation algorithms to timing analysis attack.

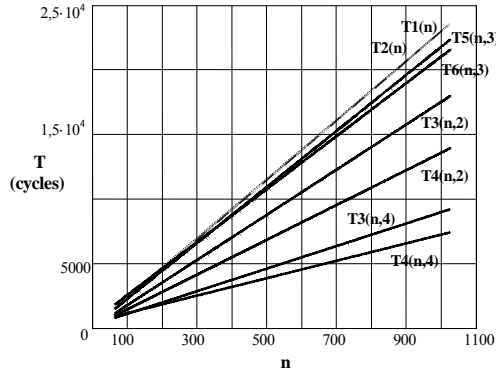


Fig. 1 Dependence of the performance on $\lceil \log n \rceil$ value

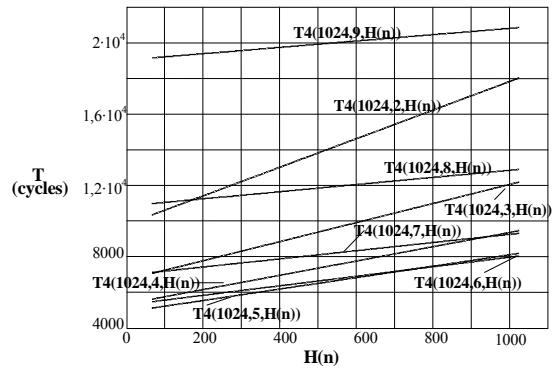


Fig. 2 Dependence on the Hamming weight value

4. Resistance to Timing Analysis

Timing analysis is one of the special kinds of cryptanalysis, which allows determining the secret encryption key on the basis of estimation of the performance time statistical parameters [2]. The latest investigation in this area shows that such kind of attack can be very efficient when the eavesdropper has the access to the encrypting tools. So, there is very important problem to find modular exponentiation method with the highest performance and resistance to Timing Analysis.

It was noticed that the main characteristic of the encryption key is the Hamming weight.

For the estimation of the resistance of the mathematical models of the discussed modular exponentiation method have to be differentiated based on Hamming weight $H(n)$.

The normalized resistance S of the noticed algorithms can be estimated as $S = \cos\left(\arctg \frac{dT_i}{dH(n)}\right)$.

In the table 3 the estimation of the parameters of every algorithms with $w = 4$ and $w_i = 3$ has been presented.

The analysis of the table shows that the beta-ary LTR method is absolutely resistant to timing analysis

Table 3. The estimation of the parameters of considered algorithms

Algorithm	Width n (bit)	Performance Time (cycles)	Normalized resistance S
Binary method from left to right	1024	23550	0.062
	2048	47110	
Binary method from right to left	1024	23560	0.062
	2048	47110	
B -ary method from left to right	1024	9204	1
	2048	18160	
B -ary method from right to left	1024	7412	0.243
	2048	14320	
Shift window method from left to right	1024	22350	0.430
	2048	44570	
Shift window method from right to left	1024	21560	0.114
	2048	42590	

5. Timing Models of Modular Exponentiation Methods

According to the mathematical models of the realization time the exponent bits influence on the values of: t_i , $d_{i,j}$, $s_{i,j}$.

To realize the attack cryptanalytic performs on the identical PC the similar exponentiation as real, to get the

times $\widehat{T}_{i,k-1,0}$ and $\widehat{T}_{i,k-1,1}$ (for every LTR method) or $\widehat{T}_{i,0,0}$ and $\widehat{T}_{i,0,1}$ (for every RTL method, accordingly) for the exponents 0 and 1. After that he/she can construct the table of differences between real and guessed timings in the way that was shown in [2].

Cryptanalytic can find the exponent n_{k-2} (or n_1 for every method from left to right) and continue so on for the consideration the other exponent bits n_{k-3}, \dots, n_0 (n_2, \dots, n_{k-1}).

Let j_0 is a particular value of j in the corresponding algorithm and let $g = \begin{cases} 0, & \text{for the exponent 0} \\ 1, & \text{for the exponent 1} \end{cases}$.

Let $\widehat{s}_{i,j_0,g} > 0$ for β -ary method LTR (because it doesn't depend on n_i) and $\widehat{s}_{i,j_0,g} = \begin{cases} 0, & g = 0 \\ > 0, & g = 1 \end{cases}$ is the time of the multiplication for β -ary method from right to left, when $n_i = 1$. Then the times

$$\widehat{T}_{i,j_0,g \beta LTR} = t_i + 2c_i + (\beta - 1)s_i + \sum_{j=k-1}^{j_0+1} (d_{i,j} + s_{i,j}) + (d_{i,j_0} + \widehat{s}_{i,j_0,g}), \quad (1)$$

$$\widehat{T}_{i,j_0,g \beta RTL} = t_i + (\beta + 1)c_i + b_i + \sum_{\substack{j=0 \\ n_j=1}}^{j_0-1} d_{i,j} + \sum_{\substack{j=0 \\ n_j=1}}^{j_0-1} (d_{i,j} + s_{i,j}) + (d_{i,j_0} + \widehat{s}_{i,j_0,g}). \quad (2)$$

For Sliding Window Method it can be done in the same way. Now can be calculated:

$$\Delta T_{i\beta LTR} = T_{iLTR} - \widehat{T}_{i,j_0,g LTR} = e_i + \sum_{j=j_0-1}^0 (d_{i,j} + s_{i,j}) + (s_{i,j_0} - \widehat{s}_{i,j_0,g}), \quad (3)$$

$$\Delta T_{i\beta RTL} = T_{iRTL} - \widehat{T}_{i,j_0,g RTL} = e_i + \sum_{j=j_0+1}^{k-1} d_{i,j} + \sum_{\substack{j=j_0+1 \\ n_j=1}}^{k-1} s_{i,j} + (s_{i,j_0} - \widehat{s}_{i,j_0,g}), \quad (4)$$

$$\Delta T_{iSWLTR} = T_i - \widehat{T}_{i,j_0,g} = e_i + (p - p_{j_0})s_i + (p - p_{j_0})c_i + (p - p_{j_0})q_i + \sum_{j=j_0-1}^0 s_{i,j} + \sum_{\substack{j=j_0-1 \\ n_j=0}}^0 c_{i,j} + (c_{i,j_0} - \widehat{c}_{i,j_0,g}), \quad (5)$$

$$\Delta T_{iSWRTL} = T_i - \widehat{T}_{i,j_0,g} = e_i + (p - p_{j_0})c_i + (p - p_{j_0})s_i + (p - p_{j_0})d_i + (p - p_{j_0})q_i + \sum_{j=j_0+1}^{k-1} s_{i,j} + (s_{i,j_0} - \widehat{s}_{i,j_0,g}) \quad (6)$$

If $\widehat{s}_{i,j_0,g}$ was correctly guessed, so $\widehat{s}_{i,j_0,g} \equiv s_{i,j_0}$. From this, it follows that $\Delta T_{i\beta LTR} = e_i + \sum_{j=j_0-1}^0 (d_{i,j} + s_{i,j})$ and

$$\Delta T_{i\beta RTL} = e_i + \sum_{j=j_0+1}^{k-1} d_{i,j} + \sum_{\substack{j=j_0+1 \\ n_j=1}}^{k-1} s_{i,j}.$$

For Sliding Window Method if $c_{i,j_0} \equiv \widehat{c}_{i,j_0,g}$ then $\Delta T_{iSWLTR} = e_i + (p - p_{j_0})(s_i + c_i + q_i) + \sum_{j=j_0-1}^0 s_{i,j} + \sum_{\substack{j=j_0-1 \\ n_j=0}}^0 c_{i,j}$ and

$$\Delta T_{iSWRTL} = e_i + (p - p_{j_0})(s_i + c_i + d_i + q_i) + \sum_{j=j_0+1}^{k-1} s_{i,j}, \text{ accordingly.}$$

But in the reality $\widehat{s}_{i,j_0,g} \neq s_{i,j_0}$ or $c_{i,j_0} \neq \widehat{c}_{i,j_0,g}$, so that means that correct guessing is difficult. That is why the probability of successful attack should be estimated.

6. Secret information Leakage Risk

Let us calculate the variance of the random variable $T - \widehat{T}_{j_0,g}$ with the next conditions:

1. g is correct (i.e. n_j is correctly guessed), then the variances

$$\sigma^2(\Delta T)_{\beta LTR} = \sigma^2 \left(e + \sum_{j=j_0-1}^0 (d_j + s_j) \right) = \sigma^2(e) + j_0 \sigma^2(d) + j_0 \sigma^2(s), \quad (7)$$

$$\sigma^2(\Delta T)_{\beta RTL} = \sigma^2 \left(e + \sum_{j=j_0+1}^{k-1} d_j + \sum_{\substack{j=j_0+1 \\ n_j=1}}^{k-1} s_j \right) = \sigma^2(e) + (k - j_0 - 1) \sigma^2(d) + \frac{1}{2} (k - j_0 - 1) \sigma^2(s), \quad (8)$$

$$\sigma^2(\Delta T)_{SW LTR} = \sigma^2(e) + (p - p_{j_0}) (\sigma^2(s) + \sigma^2(c) + \sigma^2(q)) + j_0 \sigma^2(s) + \frac{1}{2} j_0 \sigma^2(c), \quad (9)$$

$$\sigma^2(\Delta T_i)_{SW RTL} = \sigma^2(e) + (p - p_{j_0}) (\sigma^2(s) + \sigma^2(c) + \sigma^2(d) + \sigma^2(q)) + \frac{1}{2} (k - j_0 - 1) \sigma^2(s). \quad (10)$$

If supposed that time of exponentiation $z = z^\beta \bmod m$ equals $(\beta - 1)s$, that mean $d = (\beta - 1)s$, so $\sigma^2(\Delta T)_{\beta LTR} = \sigma^2(e) + \beta j_0 \sigma^2(s)$, $\sigma^2(\Delta T)_{\beta RTL} = (k - j_0 - 1)(\beta - \frac{1}{2}) \sigma^2(s)$.

2. g is incorrect. Then for β -ary method from left to right can be only one case $\begin{cases} \hat{s}_{i,j_0,g} \neq 0 \\ s_{i,j_0} \neq 0 \end{cases}$ and so

$$\sigma^2(\Delta T)_{\beta LTR} = (\beta + 1)(j_0 + 2) \sigma^2(s).$$

For β -ary method from right to left and for Sliding Window Method can be two cases:

a) $\begin{cases} \hat{s}_{i,j_0,g} \neq 0 \\ s_{i,j_0} \neq 0 \end{cases}$ (for β -ary method from right to left) or $\begin{cases} c_{i,j_0} \neq 0 \\ \hat{c}_{i,j_0,g} \neq 0 \end{cases}$ (for Sliding Window Method), then:

$$\sigma^2(\Delta T)_{\beta RTL} = \sigma^2(e) + ((k - j_0 - 1)(\beta - \frac{1}{2}) + 2) \sigma^2(s), \quad (11)$$

$$\sigma^2(\Delta T)_{SW LTR} = \sigma^2(e) + (p - p_{j_0}) (\sigma^2(s) + \sigma^2(c) + \sigma^2(q)) + j_0 \sigma^2(s) + (\frac{1}{2} j_0 + 2) \sigma^2(c), \quad (12)$$

$$\sigma^2(\Delta T)_{SW RTL} = \sigma^2(e) + (p - p_{j_0}) (\sigma^2(s) + \sigma^2(c) + \sigma^2(d) + \sigma^2(q)) + (\frac{1}{2} (k - j_0 - 1) + 2) \sigma^2(s), \quad (13)$$

b) $\begin{cases} s_{i,j_0} \neq 0 \\ \hat{s}_{i,j_0,g} = 0 \end{cases}$ (for β -ary method) or $\begin{cases} c_{i,j_0} = 0 \\ \hat{c}_{i,j_0,g} \neq 0 \end{cases}$ (for Sliding Window Method), then:

$$\sigma^2(\Delta T)_{\beta RTL} = \sigma^2(e) + ((\beta - \frac{1}{2})(k - j_0 - 1) + 1) \sigma^2(s), \quad (14)$$

$$\sigma^2(\Delta T_i)_{SW LTR} = \sigma^2(e) + (p - p_{j_0}) (\sigma^2(s) + \sigma^2(c) + \sigma^2(q)) + j_0 \sigma^2(s) + (\frac{1}{2} j_0 + 1) \sigma^2(c), \quad (15)$$

$$\sigma^2(\Delta T_i)_{SW RTL} = \sigma^2(e) + (p - p_{j_0}) (\sigma^2(s) + \sigma^2(c) + \sigma^2(d) + \sigma^2(q)) + (\frac{1}{2} (k - j_0 - 1) + 1) \sigma^2(s). \quad (16)$$

This variance can be used as the criterion of the guessing about exponent bits correctness, as the column of the table with the correctly guessing has a variance which is $2\sigma^2(s)$ for β -ary method from left to right and $\sigma^2(s)$ or $2\sigma^2(s)$ for β -ary method from right to left and $\sigma^2(c)$ for Sliding Window method from left to right or $2\sigma^2(c)$ for Sliding Window method from right to left lower than another data columns. So this feature will allow estimating the risk of secret information leakage during timing analysis of binary method modular exponentiation.

Let assume that d , c , q and s is normally distributed. Let $N(\mu_d, \sigma_d^2)$, $N(\mu_c, \sigma_c^2)$, $N(\mu_q, \sigma_q^2)$ are distributing of d , c , q , and $N(\mu_s, \sigma_s^2)$ - of s .

Let $N(\mu_0, \sigma_0^2)$ is a distributing of expecting value ΔT , where $\sigma_{0\beta_{LTR}}^2 = j_0\sigma_d^2 + j_0\sigma_s^2 = \beta j_0\sigma_s^2$ or $\sigma_{0\beta_{RTL}}^2 = (k - j_0 - 1)(\beta - \frac{1}{2})\sigma_s^2$, accordingly.

Accordingly to the analysis of Secret Information Leakage Risk in [3, 5]:

$$P(S_W^2 > S_V^2) \approx P(2\sigma_0\sqrt{K}Z + \sigma_s K > 0) = P\left(Z > -\frac{\sigma_s \sqrt{K}}{\sigma_0} \frac{1}{2}\right) = \Phi\left(\frac{\sigma_s \sqrt{K}}{\sigma_0} \frac{1}{2}\right), \quad (17)$$

where $\Phi\left(\frac{\sigma_s \sqrt{K}}{\sigma_0} \frac{1}{2}\right)$ is the area under the standard normal curve from $-\infty$ to Z .

From this, the risk of secret information leakage for β -ary and Sliding Window methods can be estimated as:

$$P_{\beta_{LTR}}(S_W^2 > S_V^2) \approx P\left(Z > -\sqrt{\frac{K}{4\beta j_0}}\right), \quad (18)$$

$$P_{\beta_{RTL}}(S_W^2 > S_V^2) \approx P\left(Z > -\sqrt{\frac{K}{(k - j_0 - 1)(2\beta - 1)}}\right) \quad (19)$$

$$P_{SW_{LTR}}(S_W^2 > S_V^2) \approx P\left(Z > -\sqrt{\frac{K}{4((p - p_{j_0})\left(\frac{\sigma_q^2}{\sigma_c^2} + \frac{\sigma_s^2}{\sigma_c^2} + 1\right) + j_0 \frac{\sigma_s^2}{\sigma_c^2} + \frac{1}{2}j_0)\right)}\right) \quad (20)$$

$$P_{SW_{RTL}}(S_W^2 > S_V^2) \approx P\left(Z > -\sqrt{\frac{K}{4((p - p_{j_0})\left(\frac{\sigma_c^2}{\sigma_s^2} + \frac{\sigma_d^2}{\sigma_s^2} + \frac{\sigma_q^2}{\sigma_s^2} + 1\right) + \frac{1}{2}(k - j_0 - 1))}\right)} \quad (21)$$

With the increasing of K , the probability of the success attack is increasing too. It is also obvious that the risk of the secret information leakage is increasing relatively to the number of correctly guessed bits, since the entropy is decreasing.

7. Estimation of the dependence of Secret Information Leakage Risk on number of correctly guessed bits of Exponent

In [3, 4] there are the averaged abstract approximations for time requested to compute c , b , t , q , s , d operations. In the next estimations we assume these approximations.

To compare the analyzed modular exponentiation methods, let us make a raw assumption that $p_{j_0} = p \cdot \frac{j_0}{k}$ and than, from probability approximation, it arises that $p - p_{j_0} = \frac{j_0}{3}$. So, this will allow us to compare the risk trends for analyzed methods as below.

The dependences of secret information leakage risk on j_0 for binary [5], β -ary and Sliding Window methods from left to right and from right to left, where number of experiments equal 100 and exponent has 1024 bits, are shown on Figs. 1 and 2, accordingly.

As was noted in Section IV $\Phi\left(\frac{\sigma_s \sqrt{K}}{\sigma_0} \frac{\sqrt{K}}{2}\right)$ is the area under the standard normal curve from $-\infty$ to Z . So, the secret information risk is the lowest in the case, when β -ary Method from left to right or Sliding Window Method from left to right is used.

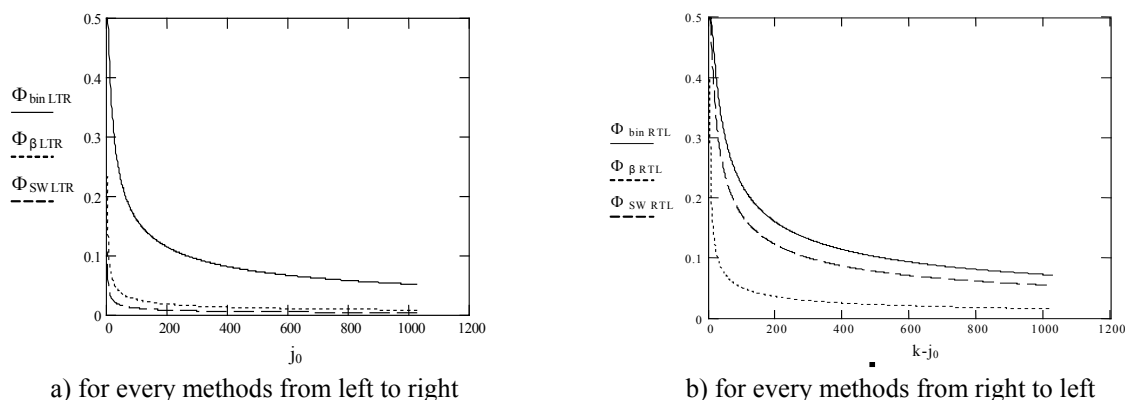


Fig. 3 Dependences of secret information leakage risk on j_0

8. Conclusion

In this paper the mathematical model to estimate the performance, normalized resistance, risk of secret information leakage during timing analysis of general modular exponentiation methods has been shown. Form the practical point of view, these results allow to make the consistent choice of the modular exponentiation method for the implementation in the real-world application systems with taking into account the existent modern attacks.

The obtained theoretical results also can be used to develop the similar probability models for modern modular exponentiation algorithms.

There are two major approaches to decrease the risk of secret information leakage during Timing analysis attack:

- 1) increasing of the measurement error $\sigma^2(e)$ by implementing the additional random calculations to decrease the possibility of correct secret key bits guessing;
- 2) decreasing K – the number of messages encrypted with the same key to decrease the probability of secret information leakage risk to the value 0.5.

The theoretical material obtained in the paper can be useful for the investigation of the risk of successful DPA attack.

9. References

- [1] E.Biham and A.Shamir, Differential Cryptanalysis of the Data Encryption Standard, Springer-Verlag, 1993.
- [2] Muir J. Techniques of Side Channel Cryptanalysis. // A thesis presented to the University of Waterloo in fulfillment of the thesis requirement for the degree of Master of Mathematics in Combinatorics and Optimization, Waterloo, Ontario, Canada, 2001.
- [3] Vasylytsov I., Vasylykiv L., Vasylykiv N., Chyrka M. Investigation of Modern Exponentiation Algorithms // Proceedings of the International Conference TCSET'2004 "Modern Problems of Radio Engineering, Telecommunications and Computer Science" (24-28 February 2004, Lviv-Slavsko, Ukraine). – Lviv: Publishing House of Lviv Polytechnic National University. – 2004. – Pp.291-293
- [4] Karpinskyy M., Vasylytsov I., Vasylykiv L. Estimation of the Secret Information Leakage Risk during Timing Analysis of Binary Modular Exponentiation Method // Proceedings of the 2-nd International Conference ACSN-2005 "Advanced Computer Systems and Networks: Design and Application" (21-23 September 2005, Lviv, Ukraine). – Lviv: Publishing House of Lviv Polytechnic National University. – 2005. – Pp. 132-135.
- [5] Moldovyan A.A., Moldovyan V.A., Sovetov V.Y. Cryptography: - Series "Textbooks for institutes. Special literature". – Spb.: Publishing "Lan", 2000. – 224 p. (in Russian)