Cryptanalysis of a Quadratic Knapsack Cryptosystem

Amr M. Youssef

Concordia Institute for Information Systems Engineering,
Concordia University,
Montreal, Quebec, H3G 1M8, CANADA.
youssef@ciise.concordia.ca

Abstract. B. Wang and Y. Hu (Computers and Mathematics with Applications, Volume 59, Issue 1, 2010) proposed a knapsack-type public-key cryptosystem by introducing an easy quadratic compact knapsack problem and then using the Chinese remainder theorem to disguise the easy knapsack instant. In this paper, we present a heuristic stereotyped message attack that allows the cryptanalyst to recover the plaintext message when partial information about the original message is known. In particular, as shown by our experiments, for the proposed system parameter $n = 100$ which corresponds to a block length of 400 bits, exposing 60% of the plaintext allows the cryptanalyst to recover the remaining 160 bits of the message with a success probability of about 90% in about 2 hours.

Key words: Public-key cryptography; cryptanalysis; knapsack cryptosystem; lattice basis reduction; stereotyped message attack

1 Introduction

Knapsack-based cryptosystems [1] [2] were among the first public-key systems to be invented. Their seemingly NP-completeness nature and their high encryption/decryption speed made them very attractive. However, it was soon realized that the underlying knapsacks often have a low density [3] and hence they are vulnerable to lattice reduction attacks [4] [5].

In [6], B. Wang and Y. Hu proposed a knapsack-type public-key cryptosystem by introducing an easy quadratic compact knapsack problem and then using the Chinese remainder theorem to disguise the easy knapsack instant into a seemingly hard one. The authors showed that their cryptosystem achieves a high knapsack density [3] under the re-linearization attack model. Furthermore, by

showing that the underlying compact knapsack problem always has exponentially many solutions, they argued that it is computationally infeasible for the attacker to find all the solutions among which the attacker expects to pick out the plaintext. They also provided some analysis that shows that the proposed cryptosystem is secure against brute-force attacks and some known key-recovery attacks including simultaneous Diophantine approximation attacks and orthogonal lattice attacks [7].

In this paper, we present a heuristic attack against this system. Our attack allows the cryptanalyst to recover the plaintext message if part of it is exposed. Practically, this attack is applicable for situations where the encrypted messages have a specific structure (also referred to as stereotyped messages [8] [9] [10]). For example, the plaintext message that sends a daily session key may have the form "The secret for August 14, 2010 is $****$" where the actual secret is unknown. Similarly, a message that sends an account registration confirmation may have the form "Your login name is $****$ and your password is $****$".

The rest of the paper is organized as follows. A brief review of the relevant properties of the quadratic compact knapsack scheme is given in the next section. Our attack is described in section 3 and the computational experimental results that confirm its effectiveness are presented in section 4.

2 Description of the quadratic compact knapsack Scheme

In this section, we briefly review the features of the quadratic compact knapsack public key system that are relevant to our attack. Further details about the encryption and decryption operations as well as the key setup can be found in [6].

The plaintext message is encoded such that $\mathbf{M} = (m_1, \cdots, m_n)$ and $m_i \in \{0, 1, \cdots 15\}$ is encoded in 4 bits. Then the ciphertext is calculated as

$$c = \sum_{i=1}^{n} f_i m_i^2, \tag{1}$$

where $\mathbf{F} = (f_1, \cdots, f_n)^{tr}$ is the system public key generated by the key generation algorithm and tr denotes the transpose operation. For the purpose

of our cryptanalysis, the details of the key generation algorithm are irrelevant and we only need to note that, given the system parameters' specification in [6], $f_i \approx 225^2\ n^2\ (60.86)^{n-1}$. It is clear that exhaustive search for the plaintext message would require $16^n = 2^{4n}$ steps. A better meet-in-the middle attack [2] [3] requires $n16^{n/2}$ steps.

While the encryption operation as defined by (1) is a nonlinear function in the plaintext vector, the attacker can obtain a linear function just by setting $y_i = m_i^2$ and then try to attack the linearized version given by

$$c = \sum_{i=1}^{n} f_i y_i, 0 \leq y_i \leq 225. \tag{2}$$

However, in [6], it was shown that the density of the above system is given by

$$d \approx \frac{n\lceil log_2 226 \rceil}{log_2(225^3 n^3 60.86^{n-1})} \tag{3}$$

which is, for practical values of n, high enough to prevent low density attacks. Furthermore, this class of attacks is very unlikely to succeed in finding the original message $\mathbf{M}$ because many other vectors in the corresponding attack lattice are shorter than $\mathbf{M}$ [6].

The suggested security parameter for this system ($n = 100$) corresponds to a public-key size of about 6157 bits, and knapsack density of about 1.27 which is sufficiently high to prevent low density attacks. Furthermore, according to the analysis presented in [6], because of the non injectivity property of the system, the low-density subset-sum attack can find the valid plaintext only with probability $\approx \frac{1}{2^{151}}$.

3 The proposed attack

A lattice L is a discrete additive subgroup of $\mathbb{R}^m$. In particular any subgroup of $\mathbb{Z}^m$ is a lattice and such lattices are called integer lattices. In other words, a lattice consists of all integral linear combinations of a set of linearly independent vectors, i.e.,

$$L = \left\{ \sum_{i=1}^{d} a_i \mathbf{b}_i | a_i \in \mathbb{Z} \right\},$$

where the $\mathbf{b}_i$'s are linearly independent over $\mathbb{R}^m$. Such a set of vectors $\mathbf{b}_i$'s is called a lattice basis. All the bases have the same number, $d = dim(L)$, of elements, called the dimension or rank of the lattice.

Given an integer lattice basis as input, the goal of the lattice basis reduction algorithm is, to find a basis with short, nearly orthogonal vectors. Although determining the shortest basis is possibly an NP-complete problem, algorithms such as the LLL algorithm [5] can find a short basis in polynomial time with guaranteed worst-case performance. The reader is referred to [4], [13] for basic background of lattices in $\mathbb{Z}^m$ and various applications of lattice reduction in cryptography. Throughout the rest of this section, we describe the details of our proposed attack.

While there exists no efficient algorithm for solving a general quadratic Diophantine equations [11], [12], the specific constraints imposed on the solution space, i.e., the fact that $m_i \in \{0, 1, \cdots 15\}$, allows us to develop a heuristic algorithm to recover the plaintext message if part of $\mathbf{M}$, say, $\mathbf{M}^{'} = (m_{i_i}, m_{i_2} \cdots m_{i_l})$, is known.

It should be noted that for the traditional knapsack systems, partial exposure of the message can be utilized, almost in a trivial way, to reduce the dimensions of the lattice used in attacking these systems. However, for the quadratic knapsack system under consideration, utilizing the knowledge of $\mathbf{M}^{'}$ to directly attack a smaller instance of the knapsack in the form of Eq. (1) using the traditional lattice reduction based attack (where the unknown vector would be of length $n-l$ instead of n) does not allow us to recover the remaining unknown coordinates of $\mathbf{M}$ since, as explained in [6], the unknown vector is very likely to be larger than the basis of the reduced lattice. Our proposed attack proceeds in the following steps:

1. Formulate an initial basis $\mathbf{B}$ as follows

$$\mathbf{B} = \begin{pmatrix} \mathbf{I}_{n\times n} & -d_1 \times \mathbf{F} \\ \mathbf{0}_{1\times n} & -d_1 \times c \end{pmatrix}, \tag{4}$$

where $\mathbf{I}$ denotes the identity matrix, and c denotes the ciphertext corresponding to the partially exposed message. Choosing the constant d_1 large enough [13] ensures that the reduced (row) basis corresponding to $\mathbf{B}$ will be in the form

$$\hat{\mathbf{B}} = \begin{pmatrix} \mathbf{A}_{n\times n} & \mathbf{0}_{n\times 1} \\ \mathbf{x}_{1\times n} & d_1 \end{pmatrix}. \tag{5}$$

Remark 1. Let $\mathbf{Y}$ denote the vector $(m_1^2, \cdots m_n^2)$. The structure of $\mathbf{B}$ and $\hat{\mathbf{B}}$ implies that $\mathbf{Y}$ can be presented as an integer linear combination of the rows of $\mathbf{A}$. On the other hand, because of the properties of the reduced basis $\hat{\mathbf{B}}$, the rows of the matrix $\mathbf{A}$ are short and nearly orthogonal. Furthermore, since the vector $\mathbf{Y}$ is also relatively short, one expects that it is likely to be presented as a small integer linear combination of the rows of $\mathbf{A}$, i.e., one expects that $\mathbf{Y}$ can be presented as

$$\mathbf{Y} = \mathbf{s} \times \mathbf{A} \tag{6}$$

where $\mathbf{s}$ is a relatively short integer vector. Given $\mathbf{M}$, one can recover $\mathbf{s}$ by solving a set of linear equations. However, since only partial information about $\mathbf{M}$ is assumed to be known to the attacker, the number of possible solutions that one has to examine in order to find $\mathbf{s}$ is expected to be exponentially large. In order to overcome this problem, we use the lattice reduction algorithm one more time to find $\mathbf{s}$.

2. Let $\mathbf{M}^{'} = (m_{i_1}, m_{i_2} \cdots m_{i_l})$ denote the exposed l components of $\mathbf{M}$ (which are not necessarily contiguous) and let $\mathbf{A}^{'}$ be the $n \times l$ matrix constructed from the l columns of $\mathbf{A}$ corresponding to the known coefficients in $\mathbf{M}$, i.e., $\mathbf{A}^{'}[i][j] = \mathbf{A}[i][i_j]$, $i = 1, \cdots n$, $j = 1, \cdots l$. Form the basis

$$\mathbf{C} = \begin{pmatrix} \mathbf{I}_{n\times n} & -d_2 \times \mathbf{A}^{'}_{n\times l} \\ \mathbf{0}_{1\times n} & -d_2 \times \mathbf{Y}^{'}_{1\times l} \end{pmatrix} \tag{7}$$

where $\mathbf{Y}^{'} = (m_{i_i}^2, m_{i_2}^2 \cdots m_{i_l}^2)$. Again, the proper choice of the constant d_2 [13] ensures that the reduced basis corresponding to $\mathbf{B}$ will be in the form

$$\hat{\mathbf{C}} = \begin{pmatrix} \mathbf{S}_{(n+1-l)\times n} & \mathbf{0}_{(n+1-l)\times l} \\ \mathbf{z}_{l\times n} & -d_2 \times \mathit{\Pi}_{l\times l} \end{pmatrix} \tag{8}$$

where $\mathit{\Pi}$ is a permutation matrix. The $\mathbf{0}_{(n+1-l)\times l}$ on the top right corner of $\hat{\mathbf{C}}$ grantees that the first l columns of each row of $\mathbf{S} \times \mathbf{A}$ is equal to $\mathbf{Y}^{'}$ or an integer multiple of it.

3. Check if any of the rows of the matrix $\mathbf{S} \times \mathbf{A}$ satisfies the constraints on the message $\mathbf{M}$, i.e, with elements in the form of $m_j^2, m_j \in \{1, \cdots 15\}$. If such a condition is satisfied, let $\mathbf{s}$ denote the corresponding row in $\mathbf{S}$ and declare $\mathbf{s} \times \mathbf{A}$ as the original plaintext message, otherwise return (FAILURE).

The above steps are illustrated using the same example that the authors in [6] gave to support their argument about the security of their proposed system.

Example 1. Let

$$\begin{aligned}\mathbf{F} = (&11983552636085612996, 10999467547886443030, 15792325467390277628,\\ &10445813110882639381, 9252643203486974008, 17826100034189837380,\\ &1136144594347297305, 1012216192024971939, 10263527667452230037)^{tr},\end{aligned}$$

$\mathbf{M} = (3, 7, 15, 8, 6, 9, 11, 13, 10)$, and $c = 7980531210038881739482$. Following the steps of our attack, the initial basis $\mathbf{B}$ is given by

$$\mathbf{B} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -d_1 \times f_1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -d_1 \times f_2 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & -d_1 \times f_3 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -d_1 \times f_4 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & -d_1 \times f_5 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & -d_1 \times f_6 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & -d_1 \times f_7 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -d_1 \times f_8 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -d_1 \times f_9 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -d_1 \times c \end{pmatrix}$$

For $d_1 = 100000$, the reduced basis matrix corresponding to $\mathbf{B}$ is given by

$$\hat{\mathbf{B}} = \begin{pmatrix} 0 & 31 & 31 & -20 & 19 & -45 & -13 & -1 & 2 & 0 \\ 0 & -31 & 31 & 20 & -4 & -32 & -15 & -10 & 27 & 0 \\ -68 & 34 & 17 & -26 & 23 & 10 & 5 & -4 & 5 & 0 \\ 0 & 31 & 31 & 42 & -43 & -47 & 2 & -5 & -3 & 0 \\ 0 & 0 & 0 & 31 & 62 & -32 & 3 & -2 & -32 & 0 \\ 0 & 31 & -93 & 73 & 20 & -18 & -111 & 1 & 61 & 0 \\ -77 & -77 & -84 & -81 & -99 & -87 & -5 & -188 & -134 & 0 \\ 1648 & 1349 & 2229 & 933 & 789 & 2862 & -1590 & -3755 & -444 & 0 \\ 2785 & 2846 & -2315 & 519 & 2199 & -508 & 7169 & -7420 & 6456 & 0 \\ -1135 & -1117 & 304 & -334 & -816 & -297 & -1933 & 2839 & -1863 & 100000 \end{pmatrix}$$

Without loss of generality, assume that the first four coordinates of $\mathbf{M}$ are known to the attacker. Then, with $d_2 = 10000$, we form the basis

$$\mathbf{C} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -d_2A[1][1] & -d_2A[1][2] & -d_2A[1][3] & -d_2A[1][4] \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -d_2A[2][1] & -d_2A[2][2] & -d_2A[2][3] & -d_2A[2][4] \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -d_2A[3][1] & -d_2A[3][2] & -d_2A[3][3] & -d_2A[3][3] \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & -d_2A[4][1] & -d_2A[4][2] & -d_2A[4][3] & -d_2A[4][4] \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -d_2A[5][1] & -d_2A[5][2] & -d_2A[5][3] & -d_2A[5][4] \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & -d_2A[6][1] & -d_2A[6][2] & -d_2A[6][3] & -d_2A[6][4] \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & -d_2A[7][1] & -d_2A[7][2] & -d_2A[7][3] & -d_2A[7][4] \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & -d_2A[8][1] & -d_2A[8][2] & -d_2A[8][3] & -d_2A[8][4] \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -d_2A[9][1] & -d_2A[9][2] & -d_2A[9][3] & -d_2A[9][4] \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -d_2m_1^2 & -d_2m_2^2 & -d_2m_3^2 & -d_2m_4^2 \end{pmatrix}$$

$$= \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -310000 & -310000 & 200000 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 310000 & -310000 & -200000 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 680000 & -340000 & -170000 & 260000 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 310000 & 310000 & 420000 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -310000 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -310000 & 930000 & -730000 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 770000 & 770000 & 840000 & 810000 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -16480000 & -13490000 & -22290000 & -9330000 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 11350000 & 11170000 & -3040000 & 3340000 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -90000 & -490000 & -2250000 & -640000 \end{pmatrix}$$

Then the reduced basis corresponding to $\mathbf{C}$ is given by

$$\hat{\mathbf{C}} = \begin{pmatrix} 1 & 0 & 0 & 1 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \mathbf{1} & \mathbf{-1} & \mathbf{-1} & \mathbf{1} & \mathbf{0} & \mathbf{1} & \mathbf{1} & \mathbf{0} & \mathbf{0} & 0 & 0 & 0 & 0 \\ 2 & 2 & 0 & 1 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 270 & 22 & 232 & -596 & 164 & 181 & 402 & 81 & 70 & 0 & 0 & 0 & 0 \\ 300 & -123 & -547 & -111 & -94 & -342 & -518 & -88 & -55 & 0 & 0 & 0 & 0 \\ -302 & 806 & -985 & -147 & 225 & -633 & 906 & -198 & -271 & 0 & 0 & 0 & 0 \\ 28 & -194 & -21 & 229 & -127 & -22 & -447 & -12 & 14 & -d_2 & 0 & 0 & 0 \\ -87 & 394 & -349 & -233 & 160 & -216 & 582 & -62 & -102 & 0 & 0 & -d_2 & 0 \\ -233 & 158 & -67 & 215 & 9 & -57 & 166 & -32 & -50 & 0 & -d_2 & 0 & 0 \\ 181 & -106 & 172 & -242 & 31 & 124 & 1 & 51 & 59 & 0 & 0 & 0 & -d_2 \end{pmatrix}$$

Finally we have $\mathbf{S} \times \mathbf{A} =$

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 186 & -62 & -9 & 0 & -59 \\ \mathbf{-9} & \mathbf{-49} & \mathbf{-225} & \mathbf{-64} & \mathbf{-36} & \mathbf{-81} & \mathbf{-121} & \mathbf{-169} & \mathbf{-100} \\ 0 & 0 & 0 & 0 & 31 & -93 & -172 & -14 & 154 \\ -1665 & -9065 & -41625 & -11840 & 50590 & -66675 & 125289 & 535710 & 162787 \\ 8541 & 46501 & 213525 & 60736 & 94651 & 386466 & -422351 & 183495 & -147024 \\ -17505 & -95305 & -437625 & -124480 & 23027 & -265575 & 608468 & -190273 & 372141 \end{pmatrix}$$

It is clear that the second row of the matrix above corresponds to the original message $\mathbf{M}$ and the second row in $\mathbf{S}$ corresponds to $\mathbf{s} = (1, -1, -1, 1, 0, 1, 1, 0, 0)$ in (6). One should also note that $\mathbf{M}$ might be recovered even if less than four coordinates of it were exposed to the attacker. For example, knowing $\mathbf{M}' = (m_2, m_3, m_4)$ allows the recovery of the rest of $\mathbf{M}$ using the same procedure above.

4 Computational Experiments

The structure of the basis $\mathbf{C}$ used in the second step of the attack implies that the attack fails only in situations corresponding to the cases where there are at least $(n + 1 - l)$ vectors in the form $\mathbf{w}_i = \mathbf{v}_i \times \mathbf{A}, i = 1, \cdots, n + 1 - l$, whose first l coordinates are given by $\mathbf{Y}^{'}$ or an integer multiple of it and $\mathbf{v}_i$ is shorter than $\mathbf{s}$. Consequently, the vector $\mathbf{s}$ will not show in the reduced basis $\hat{\mathbf{C}}$. Because of the heuristic nature of the lattice reduction algorithm, determining an analytical expression for the probability of this event, and consequently the failure probability of our attack, seems to be an intractable problem. However, it

is intuitively expected that increasing l should improve the probability of success (i.e., as $l \rightarrow n$, prob(success) $\rightarrow 1$) since the existence probability of such short vectors, $\mathbf{w}_i, i = 1, \cdots, n+1-l$, with such constraints on its first l coordinates should diminish as l increases.

In this section, we provide some experimental results that confirm the effectiveness of our proposed attack. The performance of our attack was evaluated using Maple version 10 on a lenovo X61 laptop running Intel Core Duo CPU@1.6 GHz with 2 GB of memory.

For small values of $n \leq 50$, our attack succeeds with a good probability even if the exposed portion of the message, l, is less than $n/2$. On the other hand, for practical large values of n, the attack succeeds with non negligible probability only when l exceeds $n/2$. In particular, assuming that 60% of the original plaintext is pre-exposed to the cryptanalyst, for $n = 80$, our attack always succeeded in recovering the remaining $(0.4 \times 4 \times 80 = 128)$ message bits in about 86 minutes on average. For the suggested suggested security parameter, $n = 100$, our attack succeeded in recovering the remaining $4 \times 40 = 160$ bits of the message for 9 out of 10 cases in about 121 minutes on average.

5 Conclusions

The security level of the quadratic compact knapsack public key cryptosystem proposed by Wang and Hu is overestimated. In particular, for the suggested value of the system parameters, our stereotyped message attack allows the cryptanalyst to efficiently recover the unknown part of the plaintext with high success probability if about 60% of the original message was known to the attacker.

References

1. R.C. Merkle and M.E. Hellman, *Hiding information and signatures in trapdoor knapsacks*, IEEE Trans. on Inform. Theory, vol. 24, pp. 525-530, 1978.
2. A J. Menezes, P. C. van Oorschot and S A. Vanstone, *Handbook of Applied Cryptographic Research*, CRC Press, 1996.
3. A. M. Odlyzko, *The Rise and Fall of Knapsack Cryptosystems*. In Cryptology and Computational Number Theory, volume 42 of Proc. of Symposia in Applied Mathematics, pp. 75-88. A.M.S., 1990.

4. P. Nguyen and J. Stern, *Lattice reduction in cryptology: An update*, Algorithmic Number Theory, Proc. of ANTS-IV, Springer-Verlag, LNCS 1838, pp. 85-112, 2000.
5. A. K. Lenstra, H. W. Lenstra and L. Lovász, *Factoring Polynomials with Rational Coefficients*, Math. Ann. 261, pp. 515-534, 1982.
6. B. Wang and Y. Hu, *Quadratic compact knapsack public-key cryptosystem*, Computers and Mathematics with Applications, Volume 59, Issue 1, pp. 194-206, January 2010.
7. P. Nguyen, J. Stern, Merkle-Hellman revisited: A cryptanalysis of the Qu-Vanstone cryptosystem based on group factorizations, Proc. of Crypto'97, LNCS-1294, Springer-Verlag, Berlin, pp. 198-212, 1997.
8. S. Y. Yan, *Cryptanalytic attacks on RSA*, 1st edition, Springer, 2008.
9. M. J. Hinek, *Cryptanalysis of RSA and its variants*, CRC Press, Taylor & Francis Group, 2009.
10. D. Boneh, *Twenty Years of Attacks on the RSA Cryptosystem*, Notices of the AMS, vol. 46, pp. 203-213, 1999.
11. T. Nagell, *Introduction to Number Theory*, New York: Wiley, 1951.
12. Rainer Dietmann, *Small solutions of quadratic diophantine equations*, Proc. of Mondon Math. Soc., (3) 86, pp. 545-582, 2003.
13. Antoine Joux, *Algorithmic Cryptanalysis*, Chapman & Hall/CRC Cryptography and Network Security Series, 2009.