Journal of Theoretical and Applied Information Technology
15th May 2018. Vol.96. No 09

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

ON THE USAGE OF ARYABHATTA REMAINDER THEOREM FOR IMPROVED PERFORMANCE OF RPRIME RSA

[1]CH JL PADMAJA, [2]BEHARA SRINIVAS, [3]V.S.BHAGAVAN

[1]Research Scholar, Koneru Lakshmaiah Education Foundation, Department of Mathematics, India

[2]Head of the Department, Department of Technical Education, Andhra Pradesh, India

[3]Koneru Lakshmaiah Education Foundation, Department of Mathematics, India

E-mail: [1]padmajachivukula@gmail.com, [2]srinivasbehara45@gmail.com, [3]drvsb002@kluniversity.in

ABSTRACT

In the recent era, RSA is one of the widely known and largely used public key cryptosystems in the world. To improve the performance and speed of RSA cryptosystems, many variants to original RSA have been suggested in the literature. But there is no published literature on the evaluation of these faster variants based on the security aspects such as attacks and mitigation strategies. Through this article we intended to perform a classification of faster RSA variants, cryptanalytic attacks of RSA, mapping of different attacks to mitigation strategies and a complete evaluation of Faster RSA variants based on the severity of security threats. To assess the severity of threats, a Threat Severity Evaluation classification of these variants has been made. Also, we extended our study to look upon how the performance of RSA variants changes with a change in the decryption algorithms and build a suitability analysis to check the applicability of Aryabhatta Remainder Theorem(ART) in place of Chinese Remainder Theorem(CRT). RPrime RSA is the fastest known variant of modified RSA cryptosystems with no known security attacks or threats. RPrime uses Chinese Remainder Theorem for solving congruence equations in the decryption stage, which requires more modulo inverse operations. The decryption speed can be improved by reducing the number of modulo inverse operations required to solve the congruence. Aryabhatta Remainder Theorem takes only one modulo inverse operation to solve two congruence relations. When compared with Chinese Remainder Theorem, ART requires lesser modulo inverses. In this article, we have replaced RPrime's CRT with Aryabhatta Remainder Theorem. Performance of our new model is tested with larger modulo such as 2048 and 4096.

Keywords: *Aryabhatta Remainder Theorem, Attacks, Chinese Remainder Theorem, RSA Variants, Threat severity.*

1. INTRODUCTION

Cryptosystems are widely used in electronic payments, transactions for securely signing, encrypting, and decrypting information. Security sensitive applications and devices need faster encryption, decryption and signing mechanisms. In 1977 Rivest, Shamir and Adleman developed a Public key Cryptosystem which is ubiquitous with e-commerce and digital payments. This later became one of the most widely used public key cryptosystems with its applications in Net Banking, Secure Telephony, Smart cards and in other secure communications over networks. From being a modest Laboratory invention at MIT, it turned out into one the widely used cryptosystems in security–dependent products.

Many Researchers have identified some performance concerns with this cryptosystem and presented their own modified variants of RSA. Faster Variants such as Multi Prime RSA, Multi Power RSA, Rebalanced RSA, Batch RSA and RPrime RSA are prominent but researchers have identified that they too lack in some area or the other. The performance of RSA can be improved in three key areas, key generation, Encryption stage and decryption stage, the prime concern being the Key generation stage and the decryption stage.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

All the attacks on RSA cryptosystem are theoretical and it is a very time-taking process to attack this cryptosystem with the present-age computer technology. It may be possible to break the RSA system in future if technology such as quantum computers is introduced. All the present-day research is based on the assumption that if such situation arises, we must be ready with a stronger version of RSA system to counterfeit the attacks.

The overall aim of this research is to frame a more strengthened RSA cryptosystem both in terms of security as well as fast processing speed. In this paper, we have presented a comparative study of these faster variants of RSA based on identified security threats and decryptions speeds of algorithm. All of these variants use Chinese Remainder Theorem (CRT) in the decryption phase to obtain the message.

TRN Rao and Yang [18] suggested that public key cryptosystems could be improved on their performance front by using Aryabhatta Remainder theorem (ART) in place of the existing CRT. Also, Aryabhatta remainder theorem requires less modulo inversions to solve for residues when compared with the CRT. In this paper, we extend the application of ART to Rprime RSA in place of CRT and analyze how ART can improve the performance of Rprime RSA.

2. SECURITY ANALYSIS OF RSA VARIANTS

The security of RSA can be classified based on the functioning of the cryptosystem, involving information leakages and those prone to faults and bugs. Possible attacks on faster variants of RSA. The security of RSA can be classified based on the functioning of the cryptosystem, involving mathematical calculations such as factorization, public and private exponents, implementation information leakages and those prone to faults and bugs. Possible attacks on faster variants of RSA and applicability of Mitigation strategies is discussed here.

2.1 Batch RSA

Though Boneh and Shacham [1] suggest that there are no possible attacks on Batch RSA, as this variant uses a small public exponent, the attacks which are applicable to low encryption attacks (table 1) effect this variant of RSA. These attacks can be mitigated using random padding, ensuring the padding length is greater than 1/9 times the original message, using a larger public exponent of e=65537.

Table 1: Attacks on Batch RSA

Short Code	Type of Attack	Be Mitigated?
A11	Encryption Exponent [2]	✓
A12	Hastad's Attack [3]	
A13	Related Message Attack [4]	
A14	Short Padding Attack [5]	

2.2 RSA Re-Balanced

The low encryption attacks are not applicable to this variant of RSA and the only known attack (shown in table 2) is the Boneh's Lattice Factorization Attack [1] and this attack can be mitigated by choosing the length of k= 160 bits.

Table 2: RSA Rebalanced

Short Code	Type of Attack	Be Mitigated?
A22	Boneh Factorization Attack [1]	✓

2.3 RSA MPrime

There are a numerous possible attacks on MPrime RSA. As this variant uses a low private exponent (d), Weiner [6] exploited this weaker area of MPrime RSA and published the first known attack on this variant, and later other authors have worked on the areas where MPrime RSA can be attacked. For some attacks listed for this variant, there are no applicable mitigation strategies, the other attacks can be mitigated using a private exponent (d) of length >=300 bits.

Table 3: Attacks on MPrime RSA

Short Code	**Type of Attack**	**Be Mitigated?**
A2	General Number Field Sleeve GNFS [7, 8]	✗
A9	Attack on Low CRT decryption Exponent [9]	
A8	Wiener's Low Decryption Exponent Attack [6]	✓
A10	Partial Key exposure Attack [10]	

2.4 RSA MPower

M.Esgin et.al [11] present a partial key exposure attack on MPower RSA which can be mitigated by using a larger value of encryption exponent and keep the the LSB values secure. The side channel attack on RSA MPower can be mitigated by a secure implementation suggested by Kirtane and Pandu [12], where they use a secure Hensel Lifting algorithm implementation of MPower RSA.

Table 4: Attacks on MPower RSA

Short Code	Type of Attack	Be Mitigated?
A10	Partial Key exposure Attack [10]	✓
A17	Side Channel Attacks [13]	

2.5 RSA RPrime

RPrime is relatively new, and there are no known attacks on this faster RSA variant. The size of private exponent d is greater than160 bits and it offers a security of 2^{80}making the Factorization attacks useless [14]. Also, as the size of d is large, attacks on low decryption exponent are invalid against this RSA variant, making it highly secure when compared with the other RSA variants. Figure 1 shows the attacks on Faster RSA variants.

2.6 Threat Severity Evaluation

To evaluate the RSA variants based on the severity of the threats, a threat severity classification was developed based on Threat Severity Value (T.S.V).A threat severity value is obtained by the ratio of Number of Threats Applicable *n(T)* to the number of threats that can be mitigated *n(M.T).*

$$T.S.V = n(T) / n(M.T)$$

The Threat severity for a particular RSA variant decreases with the increase in the number of mitigated threats and vice versa. Table 5 shows the threat classification based on T.S.V.

Table 5: Threat Severity Classification

TS.V.	Threat Classification
If T.S.V<1	HIGH
T.S.V>1	LOW
T.S.V=0	No Threats Applicable

3. COMPARISON OF FASTER RSA VARIANTS

A comparison is done on the Faster RSA variants based on Threat severity classification, decryption speed, and the measure of gain based on different implementation strategies.

3.1 Security

The Table 6 shows a Classification of RSA variants based on Threat Severity Value mention in the section IV.

Table 6: Threat Severity comparison using Threat Severity Value

Variants	Attacks	Mitigation Strategies	Severity of Threats
Batch	A11	S4	*LOW*
	A12, A13	S4, S6	
	A14	S4, S7	
Re-Balanced	A22	S9	*LOW*
MPrime	A2, A9	Not Available	*HIGH*
	A8	S2, S4	
	A10	S4	
MPower	A10	S4	*LOW*
	A17	S5	
RPrime	No Known Attacks		*No Threats*

RPrime is a relatively newer variant when compared with the others and no there are no known attacks on this RSA variant. MPrime has the highest threat from cryptanalytic attacks, as there are a few possible attacks, which do not have any applicable mitigation strategy. Though there are several known attacks on BatchRSA, Rebalanced RSA and MPower RSA the threats can be mitigated using the applicable mitigation strategies mentioned in Appendix.

3.2 Speed Gain with Varied Implementation Approaches

There are several implementation approaches to improve the decryption speed on original RSA

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

such as using Chinese Remainder Theorem (CRT) + Garner's Theorem [15], CRT+Gauss Theorem [15], Aryabhatta Remainder Theorem [16] etc. In this study we have compared how the RSA varaints behave when only a CRT+Gauss theorem is applied and when CRT is applied in combination with Garner's Theorem. Only Garner's theorem is chosen for this comparison as Garner's theorem with CRT is made a defacto standard in Public Key Cryptography Standards (PKCS#1).

3.2.1 CRT + Gauss

Figure 2 shows the decryption time measure in microseconds using CRT+Gauss[15]. To conduct the experiment, for each faster RSA variant, and for each modulus (768, 1024 and 2048), 500 messages were encrypted and decrypted, the average speed for the execution was measured.

3.2.2 CRT + Garner

The traditional CRT algorithm i.e., CRT + Gauss [15] needs a number of reductions on the modulus and a total of $O(b.t)^2$ operartions (a modulus m with b bit integer), whereas, the CRT + Garner [15] doesn't and only needs only $O(t.b^2)$operations, making it much more efficient than the former. Figure 3 shows the avaerage decryption time taken by each varaint using the CRT + Garner theorem.

From the figures 2 and 3, Rebalanced RSA and MPower RSA have 20% improvements in their decryption speed for a768 bit modulus using Garner's approach. RPrime has an overall better performance with the larger modulus and has the best decryption speed with a 1024bitmodulus.

Variations in the implementation approach did not have any effect on the Batch RSA and it had a very low decryption speed when compared with all other variants. Based on the above comparison we have made the following observations on each faster RSA variants and an overall ranking of has been provided for the RSA variants shown in Table 7.

3.3. Observations on Security and Speed

3.3.1 Batch RSA

- Needs to deal with overhead of agglomerating the messages [14].
- In Batch RSA every key must have a corresponding RSA certificate in the decryption server and as the number of batch increases the number of certificates to be maintained keeps growing. As the issuing and verification of certificates by CA's involves monetary transactions, it is can be said costly considering the certificates to be maintained and the money involved [1].
- The other issue with BATCH RSA is that it uses a very small public exponent such as 3,5,7,11,13 etc. Though D. Boneh and Shacham [1] suggests that there are no known attacks, low public exponents are prone to Attacks such as Related Message Attack [4], Hastad's Broadcast attack [3] and short padding Attack[5].
- If the public exponent of BATCH RSA is set to e=65537 then it increases the decryption time, losing its "FASTER" capabilities and making it a very slower than the other variants, also than the traditional RSA.

3.3.2 MPrime

- Paixao does not recommend this Faster variant for the use in smaller modulus where n= 768 bits [1].
- Also another limitation is that when we use a 1024 bit modulus only three primes should be considered as using more primes would again decrease its overall performance [1].

3.3.3 Re-balanced RSA

- Re-Balanced RSA offers a better decryption speed when compared with variants such Batch, MPrime and Mpower.
- Also, we can use a low encryption exponent to increase the decryption speed but at the cost of attacks such as security attacks such as Hastad's [3].

3.3.4 MPower RSA

- It is resilient to Lattice factoring attacks [1] as it can only factor integers of the $N= p^k q$, where K is large as it uses a factoring of $N=p^2q$, where k is very small and equal to 3 (Mpower uses a modulo of the form

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

$N=p^{k-1}q$).

- At times, some of the SSL cards do not support the modular inversion through Henselliftings [17], which is the major difference and improvement of Mpower over Mprime RSA. This restricts the applicability and advantages of MPower.

3.3.5 Rprime RSA

- This is really a faster variant of RSA and offers high decryption speed when compared to the other variants and there are no known possible attacks on this variant.

- When implemented in combination with the Garner's algorithm, it shows an even better performance in the decryption speeds.

Note:
Batch RSA and MPower RSA do not support the family of Public Key Cryptographic Standards (PKCS#1) when compared with other faster Variants.

Table 7: Ranking of Faster Variants of RSA

Variants	Security Ranking	Speed Ranking	Overall Ranking
Batch	2	5	4
Rebalanced	2	2	2
MPower	2	3	3
MPrime	5	4	5
RPrime	1	1	1

Based on the aforementioned observations, we have provided a ranking of the faster variants of RSA based on the severity of security threats, decryption speed and other considerations such as adhering to standards, CA's etc. RPrime RSA has the best overall ranking when compared with the other variatns and MPrime RSA has the least. MPrime RSA has a very decryption speed and is prone to threats of high severity such as GNFS[7, 8].

4. DECRYPTION OF RSA VARIANTS

All of studied RSA variants use Chinese Remainder Theorem in the decryption phase to obtain the message. TRN Rao and Yang [18] suggested that public key cryptosystems could be improved on their performance front by using Aryabhatta Remainder theorem in place of the existing CRT. Also, Arybhatta remainder theorem requires less modulo inversions to solve for residues when compared with the CRT [1]. In the following sections we provide the decryption process and a suitability analysis of ART to faster variants based on their decryption process.

4.1 RPrime RSA

- By selecting m size integers (d1, d2, .., dk) such that the G.C.D of primes and decryption exponent is equal to 1, the decryption exponent d is chosen, and d can be obtained by solving the congruence using Mi= Cdi mod pi where 1<= i<= k, and M = Cd mod N.

- The decryption process is same the MPrime RSA [1].

4.2 BatchRSA

- Fiat [19] observed that, when small exponents are used for encryption say e1 and e2, the cost of decrypting two cipher texts is similar to cost of decrypting one.

- The Batch RSA deals with public key exponents, based on a batch of k distinct and pairwise relative prime public key exponents e1,e2,..,en to obtain M which is equal to Ci 1/ei for i= 1, 2,..,k [4].

4.3 MPrimeRSA

- The decryption exponent is computed using d = e-1 mod φ(N) and g.c.d(e, φ(N))=1, where φ(N)= Πki=1 (p-1).

- The private key contains a tuple <N, d1, d2,..,dk> where 1<= i<= k; di= d mod (pi-1) [1].

- The public key is pair of <e, N> similar to the original RSA [15].

- To obtain the plain text from the cipher, Mi= Cdi mod pi where 1<= i<= k, and M =Cd mod N is obtained using CRT [1].

- This decryption is similar to the RSA CRT [1].

- Singh [20] in his master thesis has integrated the improved Aryabhatta

Algorithm [21] with Lehmers Algorithm [15] on the MultiPrime RSA [1].

- The results suggest that there is an improvement of decryptions speeds when compared with the CRT implementation.

4.4 MPowerRSA

- Except the change made to the Modulus N, for the values of r >=3, where N=pr-1q, the process of encryption is same as the original RSA .

- Decryption use a Hensel Lifting Method [1] M1= Cr1 mod p and M2=Cr2 mod q. Thus M1e = C mod p and M2e = C mod q.

4.5 Rebalanced RSA

- Some of the RSA variants use a private key containing a triplet but here the private key contains of a quadruple <dp, dq, p, q>, where dp and dq are two integers of length w bits, where g.c.d (dp, p-1) = g.c.d (dq, q-1) =1 and dp ≡ dq mod 2.

- Also, d should satisfy a condition where d = dq mod (p-1) and d = dp mod (p-1).

- Decryption is similar to the MPrime RSA which uses the same decryption process of RSA CRT [1]

5. SELECTION OF A PARTICULAR VARIANT OF RSA FOR OUR RESEARCH

To select a particular variant of RSA, a suitability analysis is made based on different criteria such as Decryption speed, speed gain, Time complexity of decryption and decryption model. We are only concerned about the decryption phase as we apply ART during the decryption of a Message (M) from a Cipher (C). For the decryption we consider N as 1024 bits, w=160 and r =3. As batch RSA and MPower RSA use different algorithms in decryption stage and are not similar to the original RSA or RSA CRT [1], they are not suitable to apply ART as suggested by TRN Rao and Yang [18]. Hence we excluded Batch RSA and MPower RSA from the comparison in Table 8.

Table 8: Suitability Comparison

Variant	Time Complexity	Overall Ranking	Decryption process
RPrime	$r\ x\ O(w(n/r)^2)$	1	Same as RSA CRT[1]
MPrime	$r\ x\ O((n/r)^3)$	5	Same as RSA CRT[1]
ReBalanced	$2\ x\ O(w(n/2)^2)$	6.14	Same as RSA CRT[1]

RPrimeRSA has a better time complexity and uses the original RSA decryption process, whereas RebalancedRSA has only half of RPrimeRSA's speed while using Garner's algorithm [21] and is below RPrimeRSA in the overall ranking comparison. Aryabhatta remainder theorem was applied on MPrimeRSA [18]. Hence from the above comparison we prefer that RPrimeRSA be used to carry out our research.

6. RPrime RSA

Observations from our work in the above sections suggest that RPrime RSA [14] has a higher ranking in terms of decryption speed or security threats when compared with the other variants. In the following sections we discuss the algorithm of RPrime RSA, usage of Aryabhatta Remainder Theorem (ART) [20] in place of Chinese Remainder Thorem and analyze how ART can improve the performance of RPrimeRSA.

- This algorithm uses multiple primes with each prime of size log(n/k) bits with G.C.D (p1-1, p2-1, .., pk-1) =2.

- The decryption exponent d is chosen by selecting m size integers (d_1, d_2, .., d_k) such that the G.C.D of primes and decryption exponent is equal to 1 and d can be obtained by solving the congruence using the CRT.

- The encryption process is same as the Rebalanced RSA [6] and decryption process is same the MultiPrime RSA [1].

- There are no known attacks on RPRime RSA but since it uses MultiPrimeRSA's decryption, an improvement can be made by using ART in place of CRT in the decryption stage than the Key generation phase.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

6.1 Usage Of Aryabhatta Remainder Theorem

The problem of two residues can be solved with only one modular inverse operation using the original Aryabhatta remainder theorem, which is an improvement over the CRT[1]. Rao and Yang [18] have introduced and improved Aryabhatta Algorithm which can be applied to RSA cryptosystems which is comparable with other Remainder theorems in place such as CRT + Gauss and CRT +Garner. A. Singh [20] in his master thesis has implemented the improved Aryabhatta Algorithm [21] with Lehmers Algorithm [15] on the MultiPrime RSA [1]. The results suggest that there is an improvement of decryptions speeds when compared with the CRT implementation.

As, the decryption process of RPrime RSA[14] is based on MultiPrime RSA[1], we chose to apply the Aryabhatta Remainder Theorem on the RPrime RSA. The following sections show the algorithm, implementation and the results.

7. DECRYPTION OF RPRIME USING ARYABHATTA REMAINDER THEOREM

Aforementioned, the decryption phase of RPrime RSA is same as MultiPrimeRSA[1]. A. singh [20] in his master thesis presented a modified version of Arybhatta Remainder Theorem, combining Rao and Yang's [18] Improved Aryabhatta Algorithm and Lehmers Algorithm [15]. We suggest that same algorithm with slight changes is applicable to RPrime RSA. The following is our RPrime ART algorithm, which does modular reductions by computing at once rather than every time we run the loop.

RPRimeART(N_i, p_i, c_{pi}, d_{pi}, m_{pi})
Begin
1) **Define** $c_{pi} := c \bmod p_i$ for i = 1,2,3,..,k.
2) **Define** $d_{p_i} := d \bmod p_i - 1$ for i=1,2,3,..,k.
3) **Define** $m_{pi} := c_{pi}^{d_{pi}} \bmod p_i$ for i = 1,2,3,..,k.
4) **Set** $M_i := 1$
5) ***Loop***:**for**i from 2 to k
6) ***do***
7) $M_i = M_i p_i^{-1}$
8) $C_i = M_i \bmod p_i$
9) ***End Loop***
10) **Set** u:= m_{p1}
11) ***Loop***: **for**i from 2 to k
12) ***do***
13) $v = (m_{pi} - u) C_i \bmod p_i$
14) $u = u + v \Pi^{i-1}_{j=1} p_j$
15) ***End Loop***
16) **Set**c := u
17) **return** c
End

7.1 Implementation Results of RPrime ART

RPrime ART has been implemented using the BigInteger class in Java to test for modulo sizes of 4096 bit on Linux and Macintosh operating systems with Core i3 Processor and 4gb RAM. The implementation has been carried out in both single system as well as over LAN.

We have considered a sample size of 50 Messages and calculated the average decryption time for the RPrime CRT and RPrime ART algorithms. Figure 5 below shows the implementation result of RPrime RSA on our Sample string "Improving the decryption speed of RPrime RSA using Aryabhatta Remainder Theorem".

Two comparisons were made due to the nature of RPrime RSA [14] having variations in the number of primes (p_1, p_2,..,p_k), mostly, also in the above sections comparisons were made on 3 primes, so we have tested our implementation on number of primes ranging from 4 to 7 primes with a constant modulo size of 512 as shows in Figure 6.

The second set of comparisons was made against fixed number of primes, i.e., we chose 3 primes, as shown in Figure 7. Decryption speed is measured in microseconds. A comparison was made on RPrime CRT and RPrime ART with 1024, 2048 and 4096 size moduli.

These results show that RPrime ART performs better with fixed modulo and variable moduli. There is a considerable improvement of decryption speed when the modulus is 4096. Though there is only slight improvement when the modulo size is less than or equal to 2048.

8. CONCLUSION

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

In this paper, a review of faster variants of RSA and mapping is carried out for the possible attacks and the available threat mitigation strategies to RSA cryptosystem. Further, a performance evaluation is conducted on different Fast variants of RSA such as Batch RSA, MPower RSA, Rebalanced RSA, MPrime RSA and RPrime RSA, of how these variants perform in cases of encryption and decryption, severity of threats and also checked for the difference in implementation mechanisms. Also, a ranking is provided based on decryption speed and severity of threats. The empirical data suggests that RPrime RSA possesses a higher degree of security and is very fast in decrypting a cipher text when compared to all other faster variants of RSA.

For analyzing the performance of faster variants of RSA during decryption phase, we have made a selection of faster RSA variants based on the decryption model, time complexity and overall ranking from our work. From our observations, RPrime RSA and Rebalanced RSA use the same decryption process as the MPrime RSA. RPrime RSA has a much better overall ranking and time complexity when compared with Rebalanced RSA and so we have selected it to carry out our research in this area. Through this article, we have proposed a new algorithm to increase the existing decryption speed of RPrime RSA by using Aryabhatta Remainder Theorem (ART) in place of Chinese Remainder Theorem (CRT). With the application of our new implementation technique, the decryption speed of the RPrime RSA improved significantly when compared with the traditional CRT algorithm. Also, a comparison was made over the performance of RPrimeART based on 2048 and 4096 size moduli. Our new technique went successful in improving the speed of RPrime in case of larger moduli.

REFERENCES:

[1] D. Boneh and H. Shacham, "Fast variants of RSA", *CryptoBytes*, Vol.5, No.1, 2000, pp.1-9.

[2] A.May and M. Ritzenhofen, "Solving systems of modular equations in one variable: how many RSA-encrypted messages does Eve need to know?" PKC'08, *LNCS*, vol. 2146, 2008, pp. 37–46.

[3] J. Hastad, "Solving simultaneous modular equations of low degree", SIAM Journal on Computing, Vol. 17, 1988, pp. 336–341

[4] M.K. Franklin and M.K. Reiter, "A linear protocol failure for RSA with exponent three", *Crypto'95*, Rump Session, 1995.

[5] D. Coppersmith, M. Franklin, J. Patarin, and M. Reiter, " Low-exponent RSA with related messages", EUROCRYPT'96, *LNCS*, vol. 1070, 1996, pp. 1–9.

[6] M. J. Wiener, "Cryptanalysis of short RSA secret exponents", *IEEE Transactions on Information Theory*, vol.36, No.3, 1990, pp. 553–558.

[7] R.M. Elkenbracht-Huizing, "An Implementation of the Number Field Sieve", *Experiment. Math.* Vol.5, 1996, pp. 231-253.

[8] C. Pomerance, "A Tale of Two Sieves" *North American Mathematical Society,* vol. 43, 1996, pp. 1473-1485.

[9] M. K. Dubey, "Cryptanalytic Attacks and Countermeasures on RSA Proceedings of the Third International Conference 805 on Soft Computing for Problem Solving, Advances in Intelligent Systems and Computing 258, DOI: 10.1007/978-81-322-1771-8_70, © Springer India, 2014.

[10] C. Cid, "Cryptanalysis of RSA: A Survey", SANS Institute InfoSec Reading Room, 2003.

[11] M.F. Esgin, M.S. Kiraz, and O. Uzunkol, "A new partial key exposure attack on multi-power RSA", *International Conference on Algebraic Informatics*, Springer International Publishing, 2015.

[12] V. Kirtane and C. PanduRangan, "Side Channel Attack Resistant Implementation of Multi-Power RSA using Hensel Lifting", *IACR Cryptology ePrint Archive*, 2008, pp. 368.

[13] T. Finke, M. Gebhardt and W. Schindler, "New side-channel attack on RSA prime generation", CHES'09, *LNCS*, vol. 5747, 2009, pp. 141–155.

[14] C.A.M. Paixao, "An efficient variant of the RSA cryptosystem", *IACR Cryptology ePrint Archive*, 2003, p.159

[15] A.J.Menezes, P. C. van Oorschot, and S. A. Vanstone, "Handbook of Applied Cryptography", (CRC Press Series on Discrete Mathematics and Its Applications), CRC Press, Boca Raton, FL, 1996.

[16] T. Takagi, "Fast RSA-type Cryptosystem Modulo p^k q. In H. Krawczyk, ed., Proceedings of Crypto '98, 1462 of *LNCS*, Springer-Verlag, 1998, pp. 318–326.

[17] H. Cohen, "A Course in Computational Algebraic Number Theory", *Vol 138 of*

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Graduate Texts in Mathematics, Springer-Verlag, 1996

[18] T.R.N. Rao and C.H. Yang, "Aryabhatta Remainder Theorem: Relevance to Public-key Crypto-algorithms", *Circuits, Systems and Signal Processing*, Vol. 25, No. 1, 2006.

[19] A.Fiat, " Batch RSA", *Advances in Cryptology*, Proceedings of Crypto '89, vol. 435, 1989, pp.175–185.

[20] A.Singh, "Improving the RSA Crypto Computations", *Master Thesis*, University of Louisiana, 2006

[21] H. Garner, "The residue number system", *IRE Transactions on Electronic Computers*, vol. 8, 1959, pp. 140—147.

Journal of Theoretical and Applied Information Technology
15th May 2018. Vol.96. No 09

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

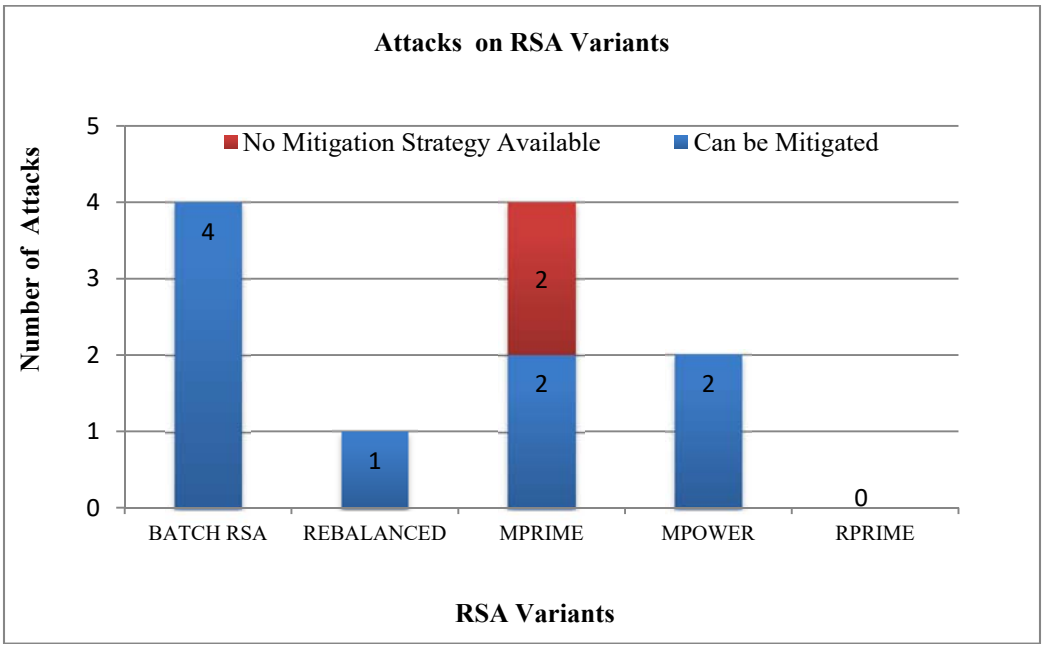
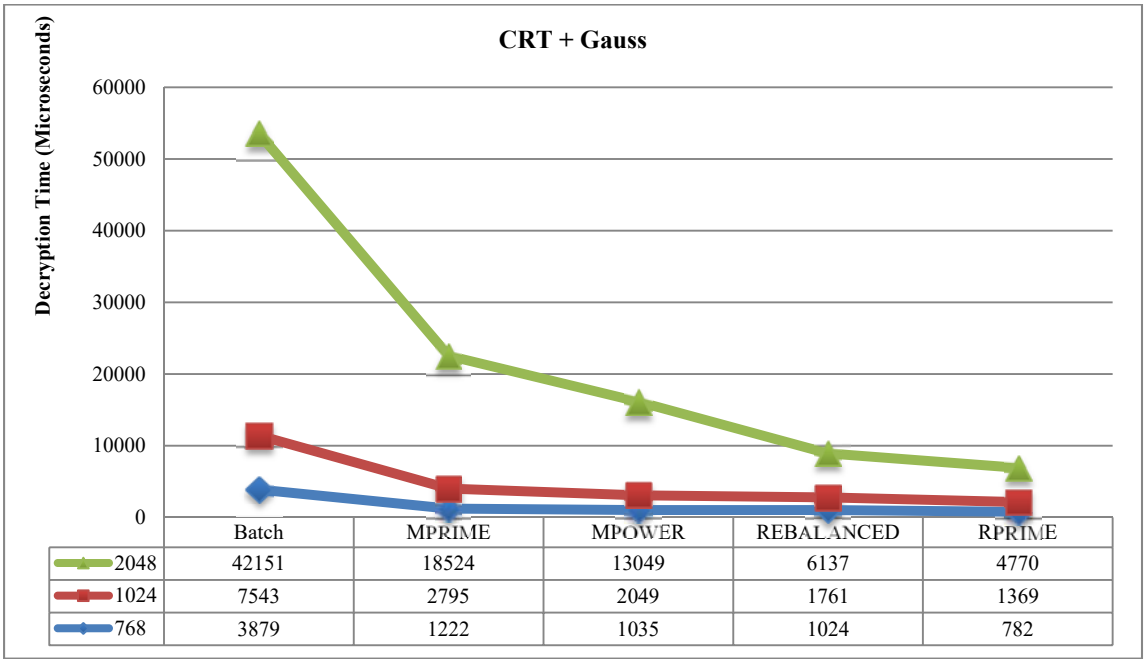


Figure 1: Number of Attacks on RSA Variants

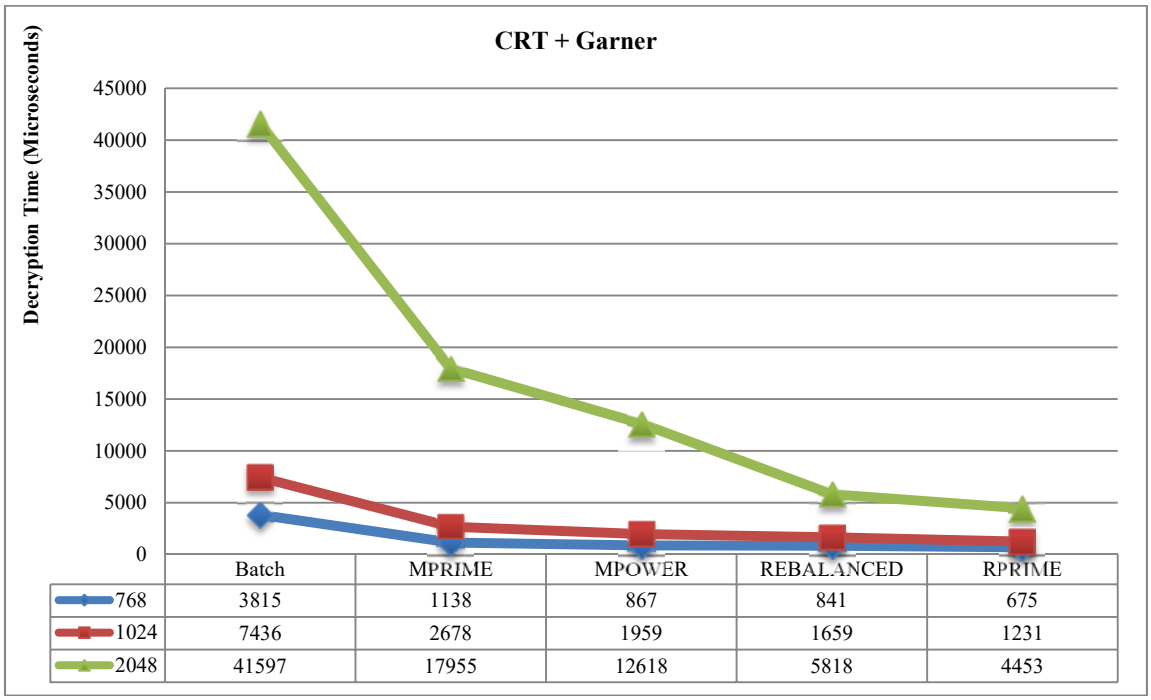


	Batch	MPRIME	MPOWER	REBALANCED	RPRIME
2048	42151	18524	13049	6137	4770
1024	7543	2795	2049	1761	1369
768	3879	1222	1035	1024	782

Figure 2: Decryption time using CRT+Gauss

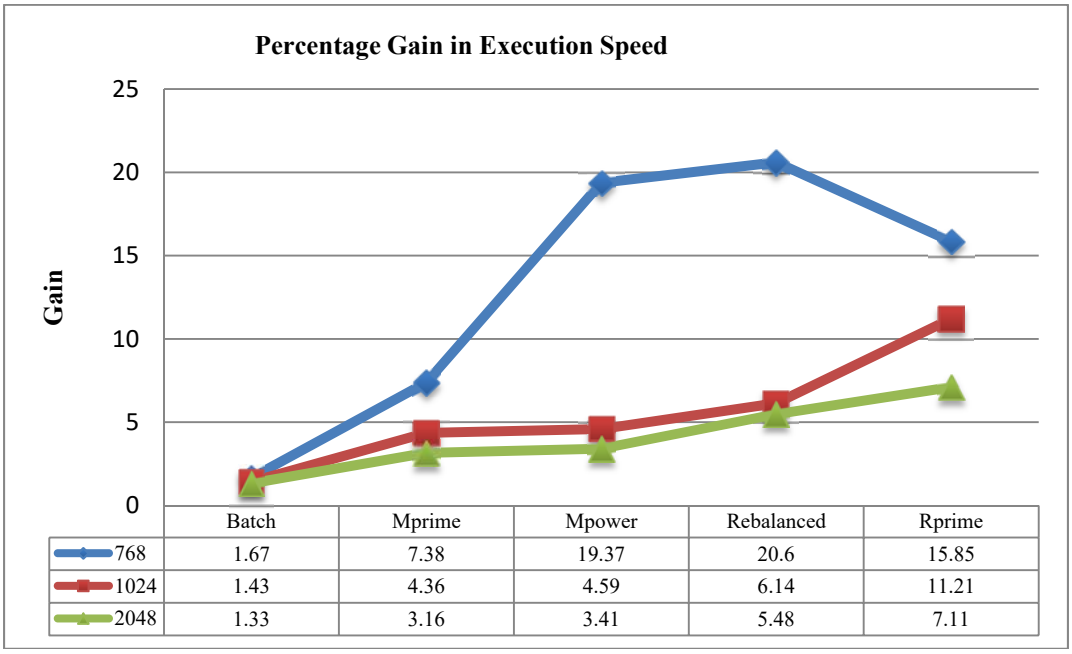
Journal of Theoretical and Applied Information Technology
15th May 2018. Vol.96. No 09

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195



Batch	MPRIME	MPOWER	REBALANCED	RPRIME
768	3815	1138	867	841	675
1024	7436	2678	1959	1659	1231
2048	41597	17955	12618	5818	4453

Figure 3: Decryption time using CRT+Garner



Batch	Mprime	Mpower	Rebalanced	Rprime
768	1.67	7.38	19.37	20.6	15.85
1024	1.43	4.36	4.59	6.14	11.21
2048	1.33	3.16	3.41	5.48	7.11

Figure 4: Percentage gain in execution speed

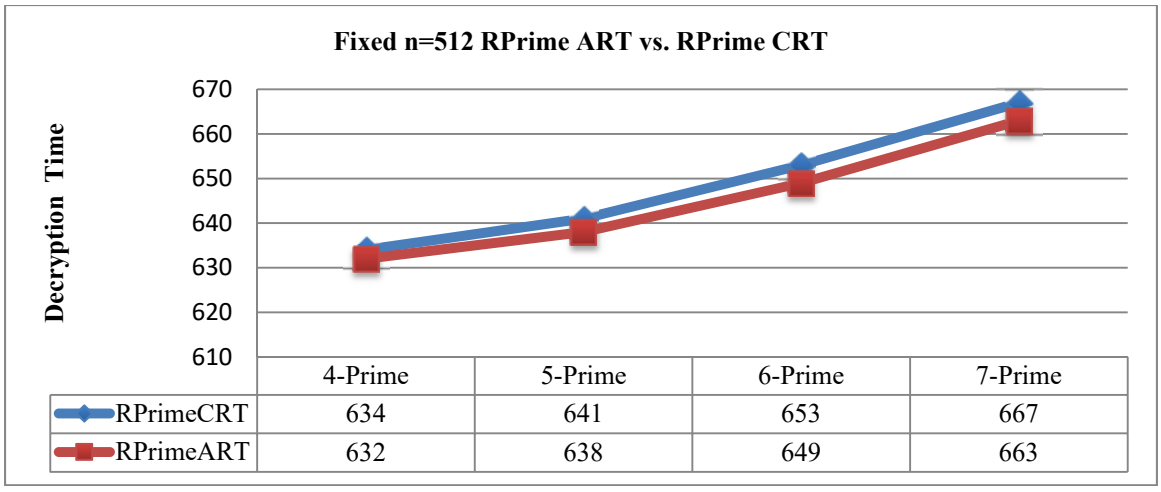
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

```
1  import java.io.DataInputStream;
2  import java.io.IOException;
3  import java.math.BigInteger;
4  import java.util.Random;
5
6  public class RPrimeART
7  {
8      private BigInteger p;
9      private BigInteger q;
10     private  static BigInteger N;
```

Result RPrimeART — Locked

```
$javac RPrimeART.java
$java -Xmx128M -Xms16M RPrimeART
Modulo:
13709785052568414469339834375695642458688050984589889786031847615382007409938413271134780939884586220274836765844551147921805948070641368780198962945
71716957411761691927709382090534425171953561869737019856671377345441787616161313617868889140311725082579794582233269295656522474488323619777677414526
56468937679
Modulo Size:512
Encrypting String: Improving the decryption speed of RPrime RSA using Aryabhatta Remainder Theorem
String in Bytes:
73109112114111118105110103321161041013210010199114121112116105111110321151121011011003211110232828011410510910132828365321171151051101033265114121979
81049711611697328210110997105110100101114328410410111111414101109
Decrypted String: Improving the decryption speed of RPrime RSA using Aryabhatta Remainder Theorem
```

Figure 5: Code and Result Snippet of RPrimeART with n=512



	4-Prime	5-Prime	6-Prime	7-Prime
RPrimeCRT	634	641	653	667
RPrimeART	632	638	649	663

Figure 6: Decryption Time with 4, 5, 6, 7 Primes

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

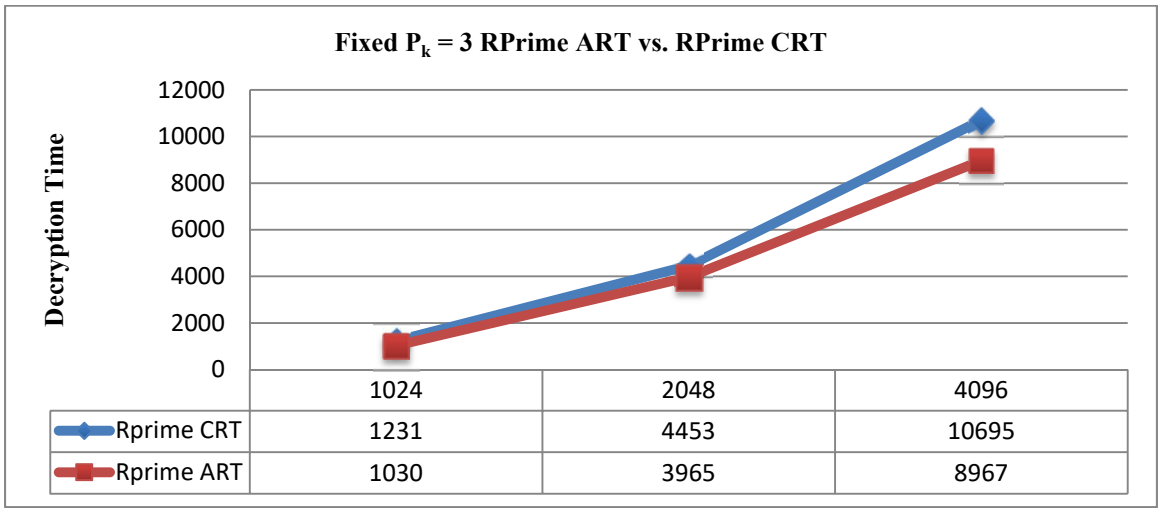


Fig. 7: Decryption Times with variable moduli length.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Appendix

S.No	Type of Attack	Short Code
1	Pollard's p-1 Factorization Method [1, 20, 23]	A1
2	General Number field Sleeve [24, 25]	A2
3	Special Number field Sleeve [26]	A3
4	Attack on Common Modulo (N) [23]	A4
5	Attack on Multiplicative structure [11]	A5
6	Cyclic Attack [11]	A6
7	Attack on Decryption Exponent [1]	A7
8	Attack based on the Low Decryption Exponent [14]	A8
9	Attack based on the Low Decryption CRT Exponent [11]	A9
10	Attack based on the Fraction of Decryption exponent [27]	A10
11	Attacks on the Encryption Exponent [17]	A11
12	Hastad's Attack [15]	A12
13	Related Message Attack [18]	A13
14	Short Padding Attack [19]	A14
15	Attacks based on Power Consumption [20, 21]	A15
16	Timing Attacks [2, 3, 16, 22]	A16
17	Side Channel Attacks [28]	A17
18	Hardware Bugs [12]	A18
19	Fault Based Attacks [13]	A19
20	Genetic Algorithms [9]	A20
21	Quantum Computing [10]	A21
22	Boneh's Factoring Attack [5]	A22

S.No	Mitigation Strategy	Short Code
1	Optimum Asymmetric Encryption Padding (OAEP) []	S1
2	Length of Private Exponent [14]	S2
3	Use of Strong Primes [1, 20, 23, 27]	S3
4	Value of e = 65,537 [27]	S4
5	Adding Delays [27]	S5
6	Random Padding [17, 19]	S6
7	Padding Length [11]	S7
8	G.C.D of N >1	S8
9	K=160	S9