VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

B.Tech.

In

Computer Science and Engineering

School of Computer Science & Engineering

CSE1011 – Cryptography Fundamentals

Enhanced RSA Algorithm with Additional Security Features

Submitted By:

Rajvi Jasani 19BCE2347

Harsh Rajpal 20BCI0271

Rohan Gupta 20BCI0260

Agniv Bhaumik 20BCI0236

Prakash Kumar 20BCE0080

Submitted To:

Madhu Viswanatham V

Abstract

With the advent of technology, security has become a critical problem. But risk of social engineering attacks and cybercrimes such as phishing, hacking, data theft etc. has increased. This is where cryptography plays a major role. Cryptography provides data security. RSA encryption, or Rivest-Shamir-Adleman encryption, is a kind of public-key cryptography that is commonly used for e-mail and other digital transactions over the Internet. The RSA cryptographic algorithm is an asymmetric technique that is used to send shared keys for symmetric key cryptography, which are subsequently used for encryption and decryption.However, RSA is a slow algorithm. Because of this, it is not commonly used to directly encrypt large files. We looked into many implementations to speed up RSA while not giving up on the secureness of the algorithm. We discovered that many popular crypto libraries (such as OpenSSL, Java and .NET) use the Chinese remainder theorem for efficiency.

Keywords: cryptography, asymmetric key algorithm, RSA, CRT, lesser decryption time, enhanced security

1. Introduction

1.1 Theoretical Background

In RSA algorithm we have two public quantities n (modulus) and e (public key), as well as two private quantities d (private key) and λ(n). λ(n) is defined as the Least Common Multiple (LCM) of all the prime factors of n. The secret exponent d is selected as a smaller and more prime integer than (n). The public key e is the "multiplicative inverse" of d, and may be determined using the formula d = e-1 mod d. (n). In the RSA system, a user secretly selects a pair of prime integers p and q that are sufficiently enormous that factoring the product n = pq would be much beyond expected computer capabilities for the cyphers' lifespan. The quantities n, e are made public and p, q, λ(n), d are kept private in the two-prime RSA cryptosystem.

Using three additional values pre-computed from the prime factors of n, the CRT representation of numbers in Zn may be utilised to conduct modular exponentiation around four times more effectively. We precompute the values d_P, d_Q and q_{inv} using the values of p and q (where p>q).

1.1 Motivation

Due to wide variety of available encryption and decryption algorithms and different techniques to arrive at a particular solution, it is really important to review the entire field prior to selecting a particular approach that satisfies the requirement. This project will provide a better technique than RSA.

1.2 Aim

The application of the Chinese Remainder Theorem (CRT) to speed up the decoding of a modified version of the RSA Algorithm. To prove the benefits of using this modified RSA with CRT over the simple RSA, we will also compare their performances.

1.3 Objectives of the proposed work

Every security system must have a set of security functions that ensure the system's confidentiality. The security system's aims are commonly referred to as these functions. These objectives can be classified into one of five categories:

Authentication: This implies that before sending or receiving a message using that security system, the identity of the recipient and sender should be confirmed.

1.3.1 Confidentiality: It specifies that only those who have been authorised can read or alter the contents of the communication.

1.3.2 Integrity: It means that the substance of the conveyed data is guaranteed to be free of any form of change between the end locations.

1.3.3 Service Reliability and Availability: Intruders may affect their availability and type of service to their users. Such system should provide a way to grant their users the quality of service they expect.

1.3.4 Non-Repudiation: It says both sender and receiver can't deny that they have sent a certain message.

2. Literature Survey

2.1 "RSA cryptosystem design based on the Chinese remainder theorem" by Chung-Hsien Wu, Jin-Hua Hong and Cheng-Wen Wu.

The planning and implementation of a systolic RSA cryptosystem based on a modified Montgomery's algorithmic programme and hence the Chinese Remainder Theorem (CRT) approach are presented in this work. In the best case scenario, the CRT approach raises turnout rate by up to four times. The modular exponentiation algorithm's projected block interleaving mechanism for multiplication and sq. operations ensures that the pulse array's process unit is fully used. The proposed design has a clear advantage over previous approaches of modifying the RSA algorithm to speed it up but it does not tackle the issue of reducing the hardware cost. In fact, the planned style incurs a 50% higher hardware value then alternative designs. The balance between speed and hardware value isn't maintained. This paper conjointly doesn't mention anything regarding the safety facet of the design.

2.2 "Analysis and design of enhanced RSA algorithm to improve the security" by S. Mathur, D. Gupta, V. Goar and M. Kuri

This paper includes n prime numbers, exponential powers, K-NN algorithm and multiple public keys and thus presents an enhanced approach different than the traditional RSA algorithm. The verification at both receiver and sender side added in this modified approach increases the security too.

The possibility of encrypted text being same as the original text is removed dues to the added randomness. This improves the system's security and efficiency. But the paper lacks any clear proof about reducing the time taken for encryption and decryption.

2.3 "Modified RSA cryptosystem based on offline storage and prime number" by R. Patidar and R. Bhartiya

A new approach to speed up the traditional RSA algorithm is introduced in this paper. This incorporates a third prime number, ensuring that modulus n is difficult to divide by invaders. When compared to the old RSA scheme, this results in faster encryption and decryption.

Although the paper proposes a faster approach, it uses database which can be hacked easily. This method compromises on the security aspect which is actually the reason a cryptosystem is used in the first place.

2.4 "AN IMPROVED RSA CRYPTOSYSTEM BASED ON THREAD AND CRT" by Kayode, S.Y. & Alagbe, G.K.

When files must be encrypted and decrypted, this work provided a parallel implementation solution employing the Chinese Remainder Theorem and thread on encryption and decryption operations in RSA. Also, in this method, the key size is extended from 1024 bits to 2048 bits in length to provide a good level of security, since 1024 bits key size may not be a very secure option now.

This paper discusses a parallel technique that divides RSA power process into separate threads and employs the use of CRT to decrease the encryption and decryption process time. However, it does not list any observations about the case where the thread level is extended which might provide a significant difference the encryption and decryption processes.

2.5 "Modified RSA Algorithm Using Two Public Key and Chinese Remainder Theorem" by Abdeldaym, R.S, Elkader, H.M.A & Hussien, R.

This paper proposed a technique which is increasing the security and improving the speed of RSA decryption using CRT and two public key pairs in place of single public key. It protects RSA algorithm from various attacks. Using the random integer if encrypted the same message more than one time it will look different every time. The general idea towards this technique is to improvement the implementation the algorithm and make it more secure and decrease the decryption time both at the same time.

This modified RSA is more secure but it is slower than the standard RSA and because of this performance of modified RSA decreases. Multiple prime number, public key, private key is the way to provide more security and, in the algorithm, they are using 4 prime numbers and 2 public and private keys. This makes user safer since he is not attacked or robbed by unauthorized people and improving security and efficiency in data sharing over the network, but less speed compares to RSA algorithm.

2.6 "Study on Improvements in RSA Algorithm" by Patel Sarthak R., Prof. Khushbu Shah, Patel Gaurav R

This paper introduces a technique RSA-CRT, to speed up RSA decryption based on the CRT, Quisquater and Couvreur proposed an RSA variation. Then, by transferring the decryption cost to the encryption cost, Rebalanced RSA-CRT decryption was sped up even further.

The public exponent e in Rebalanced RSA-CRT is of the same order of magnitude as (N) in this article, which explains why RSA encryption takes so long. Due to factorization, key creation in this technique is slow (289 seconds on average).

Can be improved by Algorithm optimization.

2.7 "An efficient decryption method for RSA cryptosystem" by Ren-Junn Hwang, Feng-Fu Su, YiShiung Yeh and Chia-Yao Chen

The goal of this research is to accelerate the decryption and signing of RSA keys. The effectiveness of modular exponentiation implementation is directly related to RSA decryption and signature performance. By decreasing modules and private exponents in modular exponentiation, this work presents a variation of the RSA cryptosystem (EAMRSA Encrypt Assistant Multi -Prime RSA). The testing results reveal that the decryption and signature speeds have significantly improved, and the variation may now be implemented in parallel with ease.

This research presents a new RSA version that increases decryption and signature generation performance. The variant can obtain excellent performance by decreasing the modulus and private exponents. Based on existing multicore devices, the upgraded form may be readily implemented in parallel and achieve greater security and speedup. The next research project will look at how to more effectively and efficiently implement the parallel RSA system in a multi-core architecture.

2.8 "Advances in Cryptology — CRYPTO" by Daniel Bleichenbacher

Against some RSA-based protocols, this work proposes a novel adaptive selected cypher text attack. It shows that an RSA private-key operation may be performed if the attacker has access to an oracle that returns just one bit for each supplied cypher text, indicating whether the cypher text matches to some unknown block of data encrypted using PKCS 1. The research demonstrates that when just partial information about the associated message is revealed, a selected cypher text attack may be carried out. It was found that not only is it critical to include a robust integrity check in an RSA encryption, but that this integrity

check should also be executed at the right phase of the protocol, ideally at the beginning. After decryption, immediately.

2.9 "High speed implementation of RSA algorithm with modified keys exchange" by S. A. Nagar and S. Alshamma

Implementation of the RSA Algorithm at a High Speed with Modified Keys Exchange They boosted the speed of RSA implementation in this study by generating keys offline and storing them in several databases before utilizing the RSA key pair in encryption and decryption procedures. RSA-Key Generations Offline is a new C# software component that we designed to help with the RSA implementation. We also needed a database engine to record the computed values in two tables: table one contains the values of p, q, n, and (n), and table two contains the values of e and d.

To speed up the RSA process, a new generation keys technique called RSA-Key Generations Offline was devised, which generated and saved all key values in database tables. You must first get a Ready Acknowledgment from the RSA Handshake Database protocol, which is responsible for generating or updating the identical gateways database, level choices (Setid), and establishing the algorithm between gateways.

2.10 "RSA with CRT: A New Cost-Effective Solution to Thwart Fault Attacks Cryptographic Hardware and Embedded Systems" by David Vigilant

Fault attacks remain a significant danger to cryptographic solutions that use RSA signatures. We present a novel countermeasure approach for RSA signature generation fault attacks in this paper. Our countermeasure is simple to implement with minimal computational cost in the constrained environment of security devices where execution time, memory consumption, customization management, and code size are major limitations. Any modular exponentiation-based cryptosystem can benefit from our method. This work proposes a novel approach for computing safe exponentiations in any integer ring (Z_N, +,) where N N. Any cryptosystem involving exponentiations in rings or finite fields of integers, such as Diffie Hellman key exchange, El Gamal decryption, RSA in simple mode, Schnorr, DSA, KCDSA, and so on, can benefit from our countermeasure approach. It is particularly important in the case of RSA with CRT, where it serves as an effective defense line against Bell core attacks.

2.11 "Implementation of RSA algorithm with chinese remainder theorem for modulus n 1024 bit and 4096 bit" by Wulansari, D., M. A. Muslim, and E. Sugiharti

The RSA cryptography is a kind of public-key cryptography. It will get progressively harder to factor the value of n as the modulus n grows larger. However, one of the RSA algorithm's drawbacks is that the decryption procedure takes a lengthy time. The Chinese Remainder Theorem was applied in this study (CRT). The purpose is to determine how long it takes RSA-CRT to execute encryption and decryption on modulus n 1024 bits and 4096 bits, as well as its implementation in Java programming. This implementation is intended to validate test findings and create the "RSA and RSA-CRT Text Security" cryptographic system.

According to the findings of the testing algorithm, the decryption speed of RSA-CRT 1024 bits is roughly 3 times quicker. The conclusion that the decryption procedure is likewise effective was reached after evaluating the algorithm RSA-CRT 4096 bits. The fault in the key creation process and the RSA 4096 bits RSA-CRT, however, is that the time required to produce the keys is greater.

2.12 "A study and performance analysis of RSA algorithm" by Preetha, M., and M. Nithya

Create a random integer with a bit length of b/2, where b is the needed bit length of n, to obtain the primes p and q. Set the low bit (to ensure that the number is odd) and the two highest bits (to ensure that the number's high bit is likewise set); Use the Rabin-Miller test to see if it's prime. If not, increase the number by two and check again until a prime number is found.

The encryption is a little tweak to the well-known and widely used RSA-OAEP. Even in the multi-query context, the security of the RSA problem remains significantly tied to the complexity of the RSA issue, according to this scheme. The RSA provides the highest level of security for the business application. Furthermore, without using hybrid or symmetric encryption, this approach may be utilized to encrypt large communications.

2.13 "Design & Implementation of Multi Power RSA–CRT Cryptosystem with N= PmQ" by Sreedevi, E., and M. Padmavathamma

The RSA cryptosystem is a cutting-edge public-key encryption system. RSA variations such as RSA-CRT, multifactor RSA, and rebalanced RSA are all meant to speed up RSA decryption or encryption. We propose the design and implementation of a quicker security method, namely the Multi Power RSA – CRT cryptosystem with N= Pm x Q, in this work. The performance of several RSA variants, as well as the Multi power RSA – CRT with N=P3 Q, has been examined and the findings have been presented.

This work offers a quicker Multi power RSA – CRT with N=Pm Q based on the Chinese Remainder Theorem for a more secure decryption procedure. When compared to existing approaches, the proposed Multi power RSA-CRT with N=PmQ takes less time to execute, provides greater performance, and provides more security. The paper's recommended solution improves performance at the expense of a slight reduction in decryption time. Furthermore, it adds semantic security to the system, while Multi prime RSA and RSA CRT do not. We compared the performance of several RSA variants, as well as the Multi power RSA – CRT with N=P3 Q, in terms of key generation, encryption, and decryption time.

3. Overview of the Proposed System

RSA Cryptosystem

RSA is an asymmetric cryptographic algorithm. Public and Private keys concept is used here. This means that there we can have any number of pairs of encryption - decryption algorithms (E, D).

E is a public encryption method, and D is a private decryption algorithm, both of which are based on the same set of values (E, D).These satisfy:

- Encryption: c= Encrypt(m), here c is cipher text corresponding to some plaintext m, Decryption: m=Decrypt(c), here m is plaintext corresponding to ciphertext c.
- So, m=Decrypt(Encrypt(m))
- Public and private keys stay that way: There is no effective technique to discover D from knowledge of E.

Algorithm: RSA cryptosystem construction

Step 1. Select two random large prime integers p and q.

Step 2. Calculate the product n = pq.

Step 3. Choose a random encryption exponent e, where e should be 0<e<(p-1) *(q-1).

Step 4. Calculate: ed mod (p − 1) (q − 1) = 1.

Step 5. The encryption function is encrypt(m) or C = m^e mod n.

Step 6. The decryption function is Decrypt(c) or m = c^d mod n.

Step 7. The public key is the pair of integers (n, e).

Step 8. The private key is the triple of integers (p, q, d).

Chinese Remainder Theorem (CRT)

It states that there is always exists an "x" (positive integer value) that satisfies the scenario when x which is divided by 2, 3, and 5 gives remainder 1 and is divisible by 7. Does a solution necessarily exist? If yes, is there more than one solution? Such questions are solved by Chinese Reminder Theorem.

Given a system of congruence to different moduli:

x == rem [0] mod (read_num[0]),

x == rem [1] mod (read_num[1]),

...

x == rem[j] mod (read_num[j]),

Here (num [0], num[1], num[2], ….., num[j]) all must be coprime.

Chinese Remainder Theorem in RSA-CRT

During decryption only we employ CRT in RSA-CRT because it gives faster results in a decryption than modular exponentiation. For key generation and decryption RSA-CRT and RSA have different methods.

We can't made short value of d and the secret exponent, d should be < N 0.292 otherwise attacker can attack RSA system.

RSA-CRT Decryption

Let M = plaintext and C = cipher text. If C is not dividable by p and d*p == d mod (p – 1), then C*dp == C*d mod p.

For decryption,

Mp= C*dp (mod p) = C*d (mod p)

and.

Mq= C*dq (mod q) = C*d (mod q).

Then using CRT, we find the solution.

M=Mp =Cdp (mod p) = C*d (mod p),

M=Mq= Cdq (mod q) = C*d (mod q).

Security

In the CRT form of RSA, decryption needs prime values p, q, and the decryption exponent d, which may appear to be a further source of weakness. However, because factoring the modulus n given the decryption exponent d is trivial, this approach does not compromise security.

4. Proposed System Analysis and Design

Simple RSA

```
def main(p, q, e, pt):
    # Encryption
    start = timeit.default_timer()
    n = p*q
    c = encrypt(pt, n, e)
    stop = timeit.default_timer()
    enc_time = stop-start
    # Decryption
    start = timeit.default_timer()
    m = decryptionRSA(p, q, e, c)
    stop = timeit.default_timer()
    dec_time = stop-start
    return enc_time*1000, dec_time*1000
```

RSA using CRT

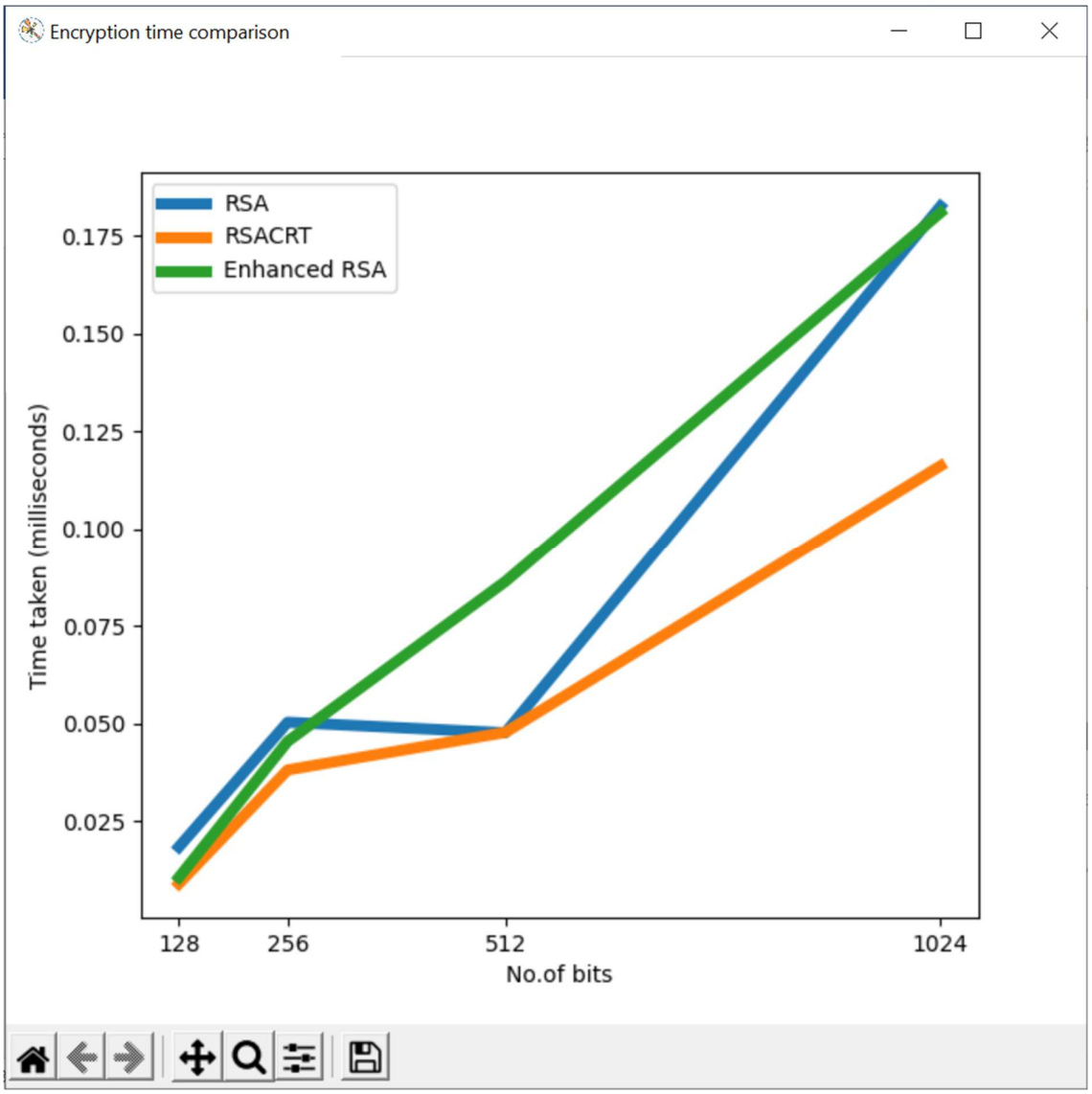
```
def main(p, q, e, pt):
    # Encryption
    start = timeit.default_timer()
```

```
    n = p*q
    c = encrypt(pt, n, e)
    stop = timeit.default_timer()
    enc_time = stop-start
    # Decryption
    start = timeit.default_timer()
    m = decryptionCRT(p, q, e, c)
    stop = timeit.default_timer()
    dec_time = stop-start
    return enc_time*1000, dec_time*1000
```

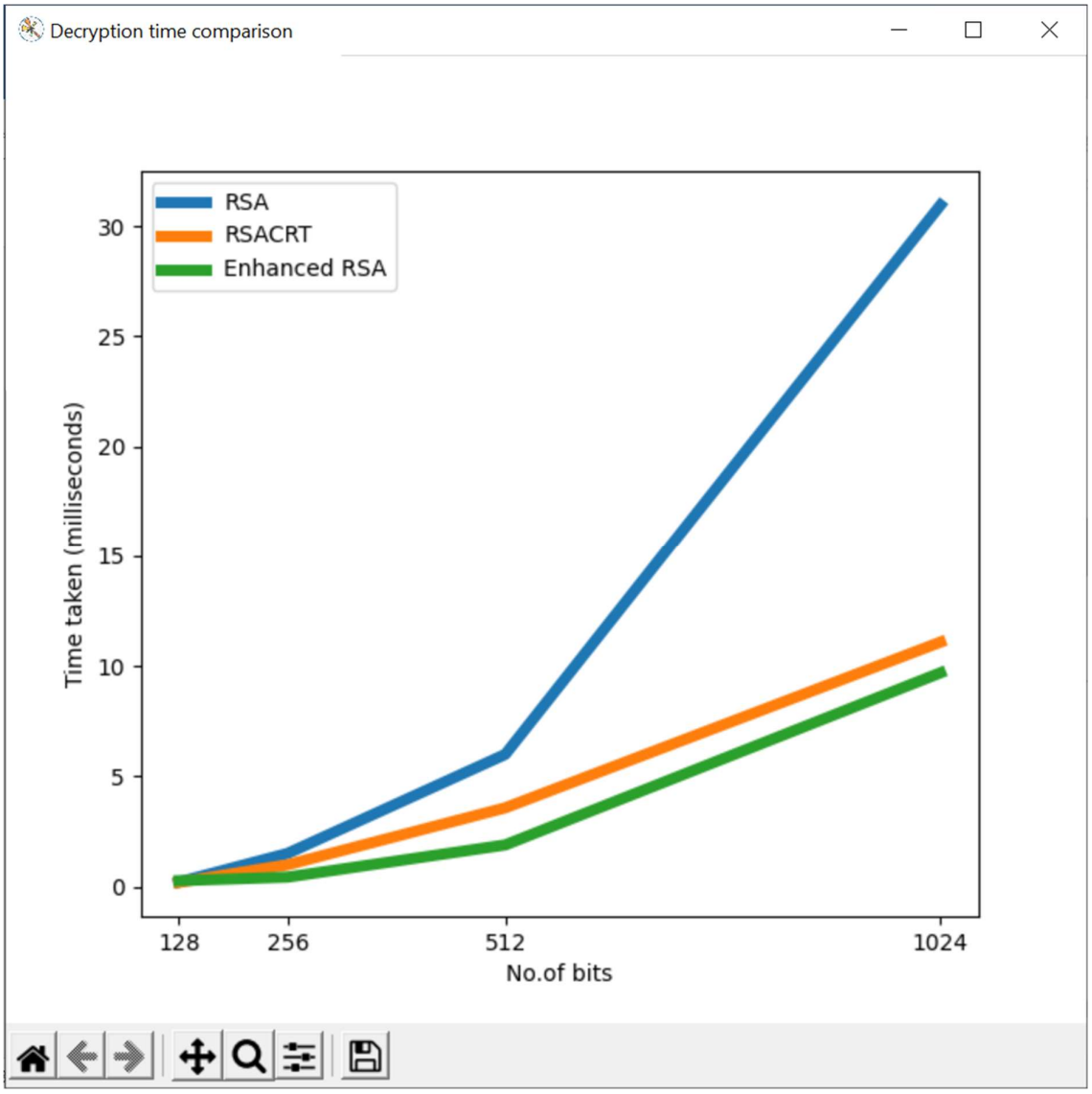
Enhanced RSA

```
def main(a, b, p, q, e, pt):
    # Encryption
    start = timeit.default_timer()
    n = p*q
    # Added security
    c1 = a*pt+b
    c = encrypt(c1, n, e)
    stop = timeit.default_timer()
    enc_time = stop-start
    print('Encrypted message using modified RSA-CRT: ', c)
    # Decryption
    start = timeit.default_timer()
    m = decryption(a, b, p, q, e, c)
    stop = timeit.default_timer()
    dec_time = stop-start
    print('Decrypted message using modified RSA-CRT: ', m)
    return enc_time*1000, dec_time*1000
```

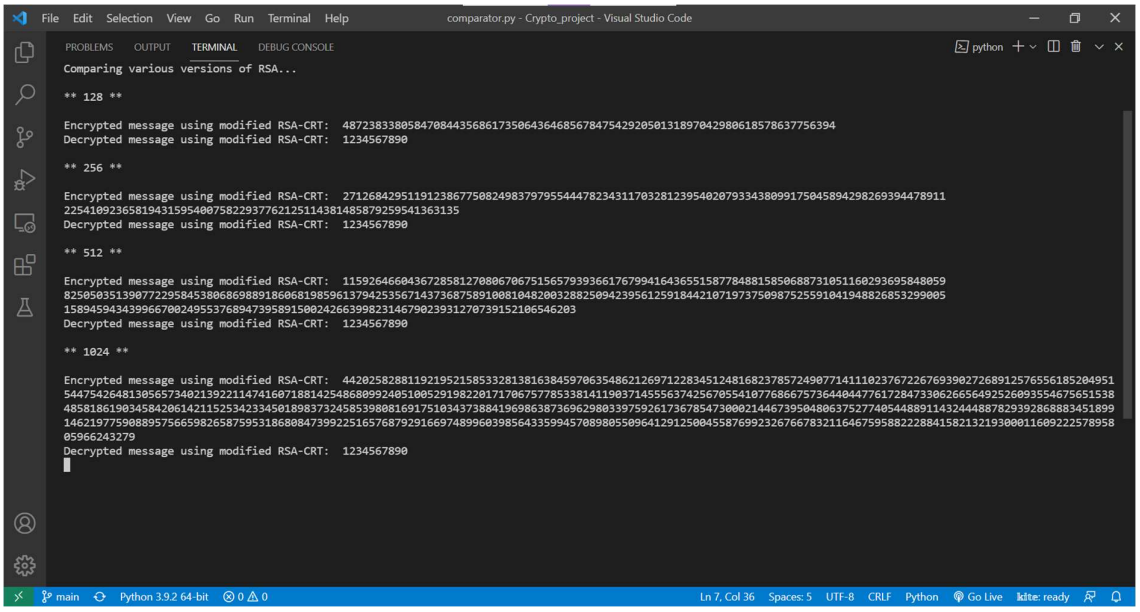
5. Results and Discussion



This figure shows the encryption time comparison of three encryption algorithm i.e., Simple RSA, RSA using CRT and Enhanced RSA. From this graph we can conclude that time taken for encryption in Simple RSA and RSA using CRT is nearly same while the Enhanced RSA has more encryption time than other two algorithms.

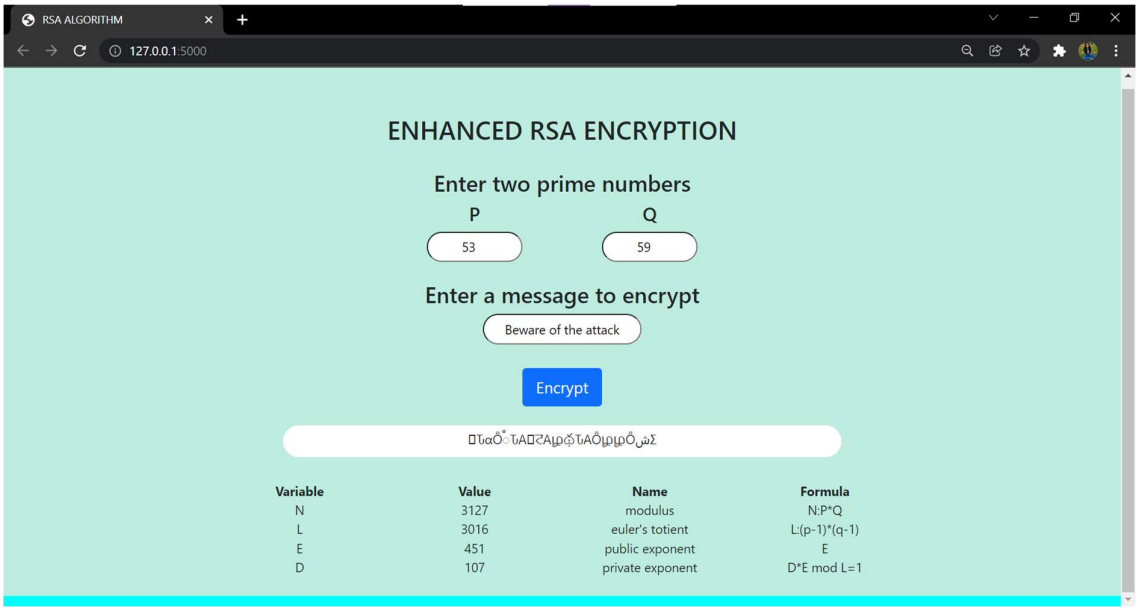


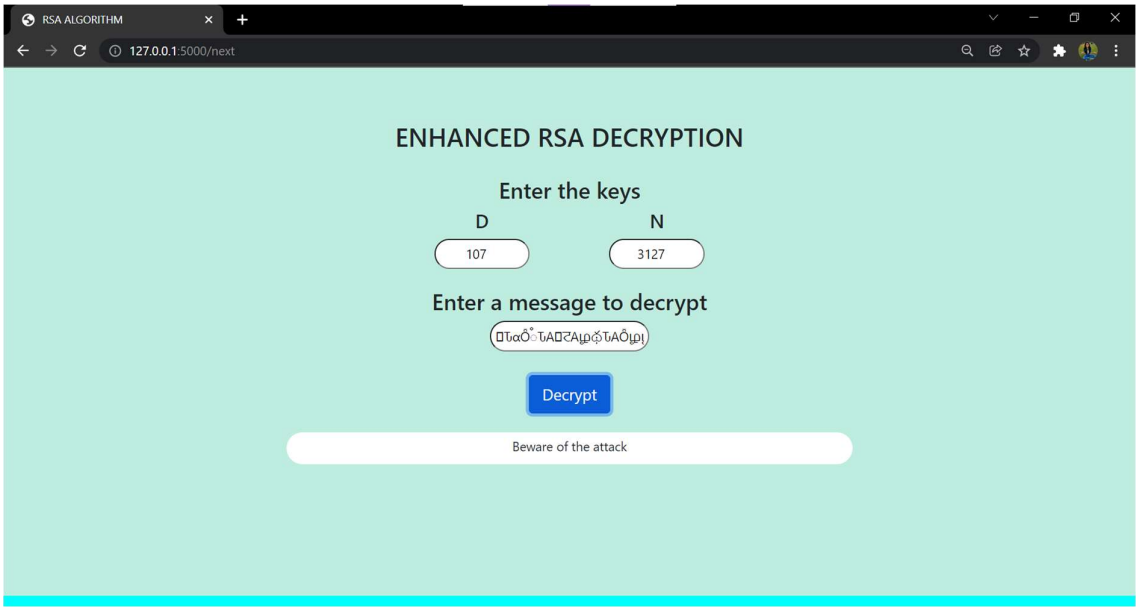
This figure shows the decryption time comparison of three encryption algorithm i.e., Simple RSA, RSA using CRT and Enhanced RSA. From this graph we can conclude that time taken for decryption in RSA using CRT and Enhanced RSA is nearly same while the simple RSA has much more decryption time than other two algorithm.



This figure shows the length of encrypted message for different sizes of prime number used in encryption in Enhanced RSA.

Enhanced RSA Encryption and Decryption Webapp





Conclusion

From this paper we can conclude that we can use Enhanced RSA algorithm instead of Simple RSA and RSA using CRT because the Enhanced RSA has same decryption time as of RSA using CRT with better security feature. Enhanced RSA is more secure than Simple RSA and RSA using CRT.

6. References

[1] Abdeldaym, R.S, Elkader, H.M.A & Hussien, R. (2019). “Modified RSA Algorithm Using Two Public Key and Chinese Remainder Theorem”. Vol.10, No.1, PP.51-64, (DOI: 10.6636/IJEIE.201903 10(1).06).

[2] Chung-Hsien Wu, Jin-Hua Hong and Cheng-Wen Wu, "RSA cryptosystem design based on the Chinese remainder theorem," Proceedings of the ASP-DAC 2001. Asia and South Pacific Design Automation Conference 2001 (Cat. No.01EX455), 2001, pp. 391-395, doi: 10.1109/ASPDAC.2001.913338.

[3] David Vigilant, ”RSA with CRT: A New Cost-Effective Solution to thwart Fault Attacks Cryptographic Hardware and Embedded Systems” – CHES 2008, 2008, Volume 5154 ISBN : 978-3-540-85052-6.

[4] Kayode, S.Y. & Alagbe, G.K. (2017). “AN IMPROVED RSA CRYPTOSYSTEM BASED ON THREAD AND CRT”. e-ISSN: 2289-6589. Volume 6. pp. 71-79.

[5] Patel Sarthak R., Prof. Khushbu Shah, Patel Gaurav R., "Study on Improvements in RSA Algorithm", International Journal of Engineering Development and Research (IJEDR), ISSN:2321-9939, Volume.1, Issue 3, pp.142 - 145, Dec 2014.

[6] Preetha, M., and M. Nithya. "A study and performance analysis of RSA algorithm." International Journal of Computer Science and Mobile Computing 2, no. 6 (2013): 126-139.

[7] Ren-Junn Hwang, Feng-Fu Su, Yi-Shiung Yeh and Chia-Yao Chen, "An efficient decryption method for RSA cryptosystem," 19th International Conference on Advanced Information Networking and Applications (AINA'05) Volume 1 (AINA papers), 2005, pp. 585-590 vol.1, doi: 10.1109/AINA.2005.97.

[8] R. Patidar and R. Bhartiya, "Modified RSA cryptosystem based on offline storage and prime number," 2013 IEEE International Conference on Computational Intelligence and Computing Research, 2013, pp. 1-6, doi: 10.1109/ICCIC.2013.6724176.

[9] S. A. Nagar and S. Alshamma, "High speed implementation of RSA algorithm with modified keys exchange," 2012 6th International Conference on Sciences of Electronics, Technologies of Information and Telecommunications (SETIT), 2012, pp. 639-642, doi: 10.1109/SETIT.2012.6481987.

[10] Sreedevi, E., and M. Padmavathamma. "Design & Implementation of Multi Power RSA–CRT Cryptosystem with N= PmQ." International Journal of Innovative Research in Computerand Communication Engineering 5, no. 3 (2017): 5466-5472

[11] “Advances in Cryptology — CRYPTO” by Daniel Bleichenbacher

[12] S. Mathur, D. Gupta, V. Goar and M. Kuri, "Analysis and design of enhanced RSA algorithm to improve the security," 2017 3rd International Conference on Computational Intelligence & Communication Technology (CICT), 2017, pp. 1- 5, doi: 10.1109/CIACT.2017.7977330.

[13] Wulansari, D., M. A. Muslim, and E. Sugiharti. "Implementation of RSA algorithm with chinese remainder theorem for modulus n 1024 bit and 4096 bit." International Journal of Computer Science and Security 10, no. 5 (2016): 186-194.