

# Variant of RSA – Multi prime RSA

Vaishali Kalaria<sup>1</sup>, Bhavesh Kataria<sup>2</sup>

<sup>1</sup>Department of Computer Engineering and Information Technology, RK University, Rajkot

<sup>2</sup>Department of Computer Engineering, LDRP Institute of Technology and Research, Gandhinagar

## ABSTRACT

Variant of RSA – Multi prime RSA that is backwards compatible that is a system using multi prime RSA can interoperate with systems using standard RSA and this variant is used to speed up RSA decryption.

## Keywords

RSA, Modulo arithmetic, prime numbers, CRT, ECM

## I. INTRODUCTION

Multi-prime RSA is a generalization of the standard RSA cryptosystem in which the modulus contains more than two primes. When decryption operations are done modulo each prime and then combined using the Chinese Remainder Theorem, the cost of decryption is reduced with each additional prime added to the modulus (for a fixed modulus size). Thus, multi-prime RSA might be a practical alternative to RSA when decryption costs need to be lowered.

The benefit of multi-prime RSA is lower computational cost for the decryption and signature primitives, provided that the CRT (Chinese Remainder Theorem) is used. Better performance can be achieved on single processor platforms, but to a greater extent on multiprocessor platforms, where the modular exponentiations involved can be done in parallel.

The security of these variants is an open problem. We cannot show that an attack on any of these variants would imply an attack on the standardized version of RSA (as described, e.g., in ANSI X9.31). Therefore, when using these variants, one can only rely on the fact that so far none of them has been shown to be weak. In other words, Use at your own risk.

## II. METHODS AND MATERIAL

### Fast Variant of RSA

Multi prime RSA:  $N = pqr$ . It is referred in PKCS#1v2.0.

### Key generation:

The key generation algorithm takes as input a security parameter  $n$  and an additional parameter  $b$ . It generates an RSA public/private keypair as follows:

Step 1: Generate  $b$  distinct primes  $p_1, \dots, p_b$  each  $\lfloor n/b \rfloor$  bits long. Set  $N \leftarrow \prod_{i=1}^b p_i$ . For a 1024 bit modulus,  $b=3$  at most.

Step 2: Pick the same  $e$  used in standard RSA public keys. Then compute  $d = e^{-1} \pmod{\Phi(N)}$ .  $e$  is relatively prime to  $\Phi(N) = \prod_{i=1}^b (p_i - 1)$ . The public key is  $(N, e)$ , the private key is  $d$ .

### Encryption:

Given a public key  $(N, e)$ , the encrypter encrypts exactly as in standard RSA.

### Decryption:

Decryption is done using the Chinese Remainder Theorem (CRT). Let  $r_i = d \pmod{p_i - 1}$ . To decrypt a ciphertext  $C$ , one first computes  $M_i = C^{r_i} \pmod{p_i}$  for each  $i$ ,  $1 \leq i \leq b$ . One then combines the  $M_i$ 's using the CRT to obtain  $M = c^d \pmod{N}$ . The CRT step takes negligible time compared to the  $b$  exponentiations.

### Performance:

Standard RSA decryption using CRT requires two full exponentiations modulo  $n/2$  bit numbers. In multi prime RSA decryption requires  $b$  full exponentiations modulo  $n/b$  bit numbers. Using basic algorithms computing  $x^d \pmod{p}$  takes time  $O(\log d \log^2 p)$ . When  $d$  is on the order of  $p$  the running time is  $O(\log^3 p)$ . Therefore the

asymptotic speed up of multi-prime RSA over standard RSA is:

$$2.(n/2)^3 / b.(n/b)^3 = b^2/4$$

For 1024 bit RSA, b=3 at most is used, which gives a theoretical speedup of 2.25 over standard RSA decryption.

**Security:**

The security of multi factor RSA depends on the difficulty of factoring integers of the form  $N = p_1 \dots p_b$  for  $b > 2$ . The fastest known factoring algorithm (the number field sieve) cannot take advantage of this special structure of N. Prime factors of N should not fall within the range of the Elliptical Curve Method (ECM). Currently, 256 bit prime factors are considered within the bounds of ECM, since the work to find such factors is within range of the work needed for the RSA-512 factoring project. For 1024 bit moduli, more than three factors should not be used.

In this paper we survey four simple variants of RSA that are designed to speed up RSA decryption in software. Throughout the paper we focus on a 1024-bit RSA modulus. We emphasize backwards compatibility: A system using one of these variants for fast RSA decryption should be able to interoperate with systems that are built for standard RSA; moreover, existing Certificate Authorities must be able to respond to a certificate request for a variant-RSA public key.

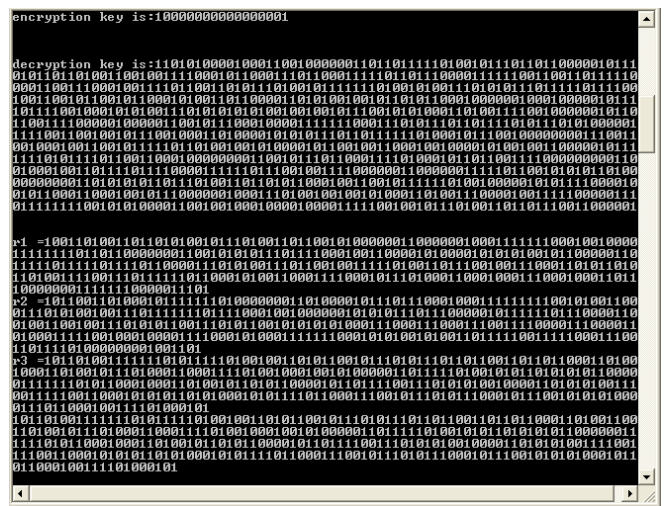
The security of these variants is an open problem. We cannot show that an attack on any of these variants would imply an attack on the standardized version of RSA (as described, e.g., in ANSI X9.31). Therefore, when using these variants, one can only rely on the fact that so far none of them has been shown to be weak. In other words, Use at your own risk.

**III. RESULT AND DISCUSSION**

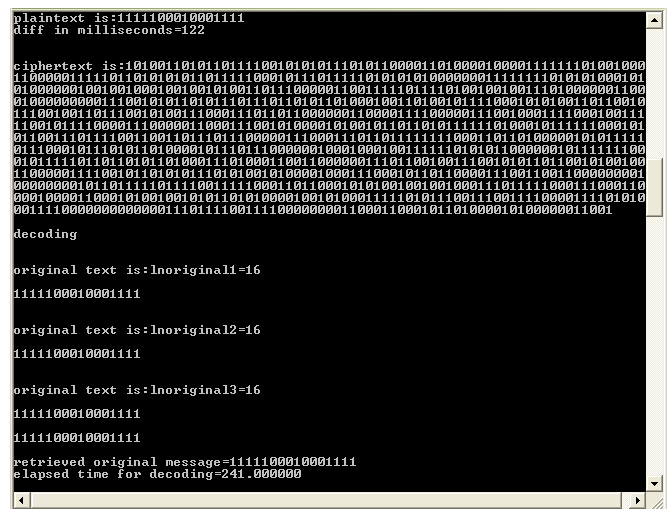
**A. Prime Numbers of Length 341 is generated**



**B. Decryption key is generated of length equal to the length of prime number**



**C. Cipher Text Generation**



## D. Time Analysis Of Basic Rsa And Multi Prime Rsa

### 1. Basic RSA

```
1. encrypt data
2. decrypt data
enter 1 for encrypt enter 2 for decrypt : 1

plaintext is:111100010001111

ciphertext of 208 length is:101011100111010110100001000111001000110111011001101
011001000101110110100111011101101011000001110111000001001001010111000000
101010101001000101100101011101010000001000011111001101101000010001111
Plus milliseconds= 50
elapsed time= 0 seconds
encryption successful
Press any key to continue_
```

```
1. encrypt data
2. decrypt data
enter 1 for encrypt enter 2 for decrypt : 2
1010110011101010100001000110010001101110100110101100100011011101101001110
11101010110000011101110000010010010010101110000000101010100100010111001010
111101010000011000011111001101101000101001111
tot_cipher=208

original text is:111100010001111
elapsed time= 900
decryption successful
Press any key to continue_
```

### 2. Multiprime RSA

```
plaintext is:111100010001111
diff in milliseconds=122

ciphertext is:10100110101011100101011010110101101000110100001000011111101001000
110000111101101010110111100010110111101010101000000111111010101000101
010000001001000100010010010010110000010011110111010010010011010000001100
010000000011001010101101101101010100010010100101110001010100101010010
110010011011100101001110001101101000000100001110000011001000111000100111
110010111000011000011000110010000010100101010111101000101111000101111000101
011001101110011001101101100000110001101101111100010110100000101011111
011100010110101010000101101100000010001000100111110101010000001011111000
0101111010101010100011010001001100000011011001011001010101010101010100
111000011110010101010101000010000100010101000011001100110011001100000001
000000001010111101110011110001011000101010010010001000110111100011000110
0001000011000101001010101010000100100011110101100111000111101010
00111000000000000110111100111000000010001100010110100001010000001001
```

```
decoding

original text is:lnoriginal1=16
111100010001111

original text is:lnoriginal2=16
111100010001111

original text is:lnoriginal3=16
111100010001111
111100010001111

retrieved original message=111100010001111
elapsed time for decoding=241.000000
```

## IV. CONCLUSION

Variants of RSA multi prime RSA gives theoretical speedup of approximately 2.25 over standard RSA decryption. In this implementation slower algorithms are used to implement basic RSA so speed up with multi prime RSA is 3.73 could be achieved over basic RSA

## V. REFERENCES

- [1] RSA Labs. Public Key Cryptography Standards (PKCS)
- [2] 2. Fast Variants of RSA By Dan Boneh and Hovav Shacham
- [3] 3. A Practical Public Key Cryptosystem Provably Secure against Adaptive Chosen Ciphertext Attack By Ranald Cramer and Victor Shoup
- [4] 4. Evaluation of Security Level of Cryptography : RSA-OAEP,RSA-PSS, RSA Signature By Alfred Menezes
- [5] 5. Network and Internetwork security By William Stallings.
- [6] 6. R. Rivest, A. Shamir, and L. Adleman. "A Method for Obtaining Digital Signatures and Public Key Cryptosystems."
- [7] 7. M. Wiener. "Cryptanalysis of Short RSA Secret Exponents." IEEE Trans. Information Theory 36(3):553–558. May 1990.Zd
- [8] 8. A. Fiat. "Batch RSA." In G. Brassard, ed., Proceedings of Crypto 1989, vol. 435 of LNCS, pp. 175–185. Springer-Verlag, Aug. 1989.