International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-9 Issue-3, February 2020

Security Enhancement of RSA Algorithm using Increased Prime Number Set

Nitin Jain, Surendra Singh Chauhan, Alok Raj

Abstract**: *In this era of digital age a lot of secret and non-secret data is transmitted over the internet. Cryptography is one of the many techniques to secure data on network. It is one of the techniques that can be used to ensure information security and data privacy. It is used to secure data in rest as well as data in transit. RSA in the most commonly used cryptographic algorithm and it is also used for the creation on Digital Certificates. RSA algorithm is now not considered to be as secure due to advancement in technology and newer attack vectors. This paper proposed an algorithm for security enhancement of RSA algorithm by increasing prime numbers count. Proposed algorithm has been implemented to encrypt and decrypt the data and execution results for encryption and decryption time have been compared for increased prime numbers count. This proposed algorithm of RSA can be used to replace the existing RSA algorithm in digital signature certificates as well as in all other places where the base RSA algorithm is currently being used. In the proposed technique, as the number of prime number count increases, prime factor calculation becomes difficult. If the attacker has encryption key (e) and Product of prime numbers (N) then it is not easy to find out the prime number combinations and hence decryption key (d) will be more secure by using proposed algorithm. This will be more difficult because given a number n, it is easy to find two numbers whose product is equal to n using Shor's algorithm and Grover's Search Algorithm but it is not very difficult and time taking to exactly determine m numbers whose product is equal to n.***

Keywords:* *Cipher Text, Decryption, Decryption Time, Encryption, Encryption Time, Plain Text, RSA Algorithm.

I. INTRODUCTION

Cryptography is a technique to make a readable data into unreadable data. Modern cryptography is part of mathematics and technology of computer science.[1],[2],[3]

A. Goals of Security (Purpose of Cryptography)

There are some specific security requirements within the context of any application-to-application communication, including these goals.[3],[4],[5]

1) *Confidentiality:* It specifies that only sender and intended recipient should able to access the contents of message. The attack on the availability is called interception. There are two main threats to confidentiality, snooping and traffic analysis.[3],[4],[5]

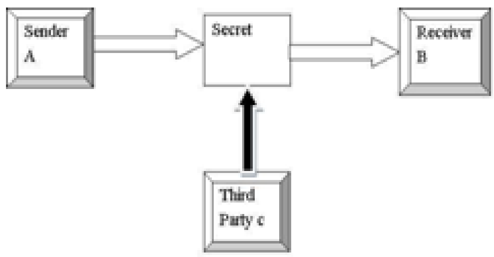


Fig. 1.Loss of Confidentiality [4]

2) *Integrity:* When sender sends a message and ensuring that the receiver receives the message as it was, wholly and error free without any changes. Attack on the integrity is called modification. [3],[4],[5]

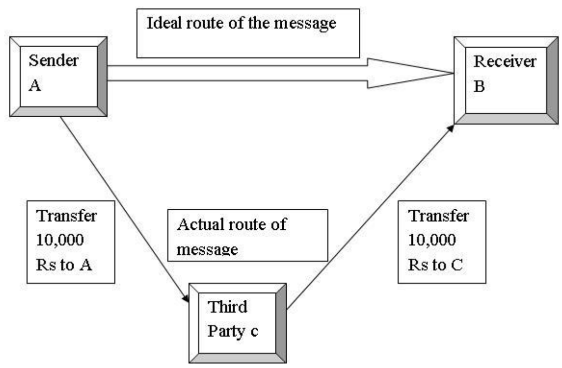


Fig. 2.Loss of Integrity [4]

3) *Availability:* Availability is ensuring that those who have the rights to information or material have always got the access to it or resources should be available to authorized parties at all time. The attack on the availability is called interruption [3],[4],[5].

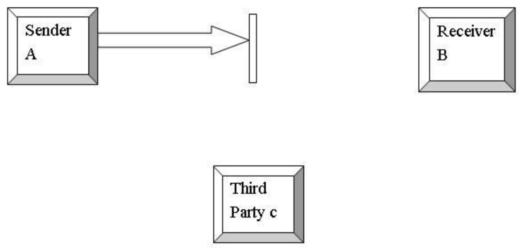


Fig. 3.Attack on Availability [4]

Revised Manuscript Received on February 05, 2020.
* Correspondence Author
Nitin Jain*, Department of AIT-CSE, Chandigarh University, Gharuan, Mohali, India. Email: nitinjain15@rediffmail.com
Surendra Singh Chauhan, Department of Computer Science, Pratap University, Jaipur, India. Email: surendrahitesh1983@gmail.com
Alok Raj, Department of AIT-CSE, Chandigarh University, Mohali, India. Email: alokraj1789@gmail.com

4) *Authentication:* It helps establish proof of identities. It ensures that the origin of a documents or message is correctly identified. Suppose that third party C sends an electronic message over the internet to receiver B. However, the third-party C had posed as Sender A when C sent this document to user B. How would Receiver B know that the message has come from C. Who is posing as Sender A.? This type of attack is called as fabrication [3],[4],[5].

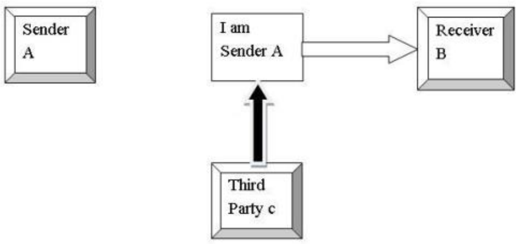


Fig. 4.Absence of Authentication [4]

5) *Non-repudiation:* It is a mechanism to prove that sender really sent this message [3],[4],[5].

B. Types of Cryptosystem

There are two types of cryptosystem:

1) *Symmetric Key Cryptography:* If sender and receiver share the same key for encryption and decryption of message than it is called symmetric key cryptography.[7],[9]

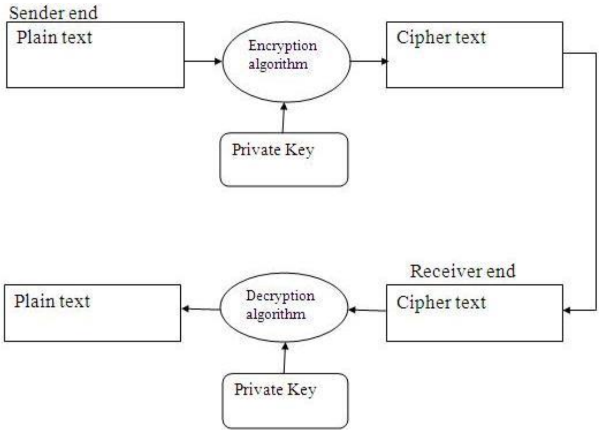


Fig. 5.Private Key Cryptography

2) *Asymmetric Key Cryptography:* If sender and receiver share the one key for encryption and another key decryption of message than it is called asymmetric key cryptography.[6],[7],[9]

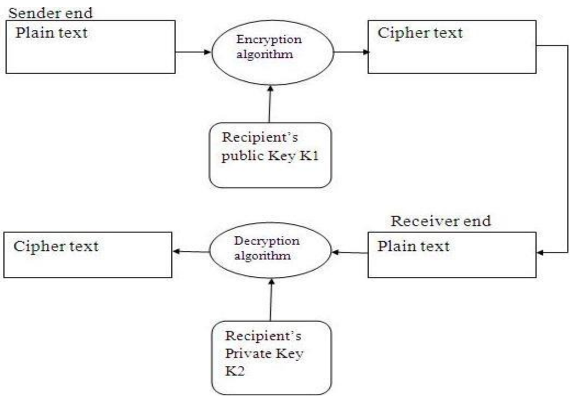


Fig. 6.Public Key Cryptography

- RSA Cryptography is the most commonly implemented Asymmetric Key Cryptography. [7],[8],[9],[10]

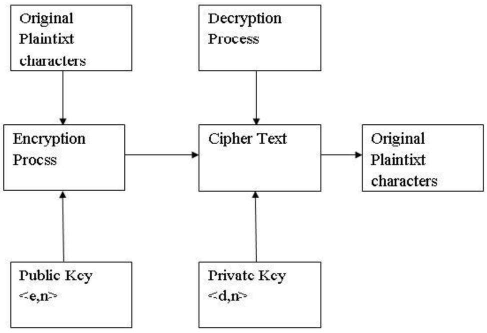


Fig. 7.RSA Model [4]

II. PROPOSED ALGORITHM

Step 1 – Take the n prime numbers (P_1, P_2, P_3, ……….,P_n) instead of two prime numbers that is used in RSA Algorithm.
Step 2 – Calculate the product of these prime numbers
($N = P_1 \times P_2 \times P_3 \times \ldots\ldots\ldots P_n$)
Step 3 – Now, select the encryption key e, such that it is not a factor of numbers ((P_1-1), (P_2-1), (P_3-1) ……… (P_n-1))
Step 4 – Calculate the decryption key d, such that
($d \times e$) mod ((P_1-1), (P_2-1), (P_3-1) ……… (P_n-1)) = 1
Step 5 – Calculate cipher text (CP) from plain text (PT) as
$CT = PT^e \bmod n$
Step 6 – At the receiver's end, calculate plain text (PT) as
$PT = CT^d \bmod n$

III. FLOW CHART

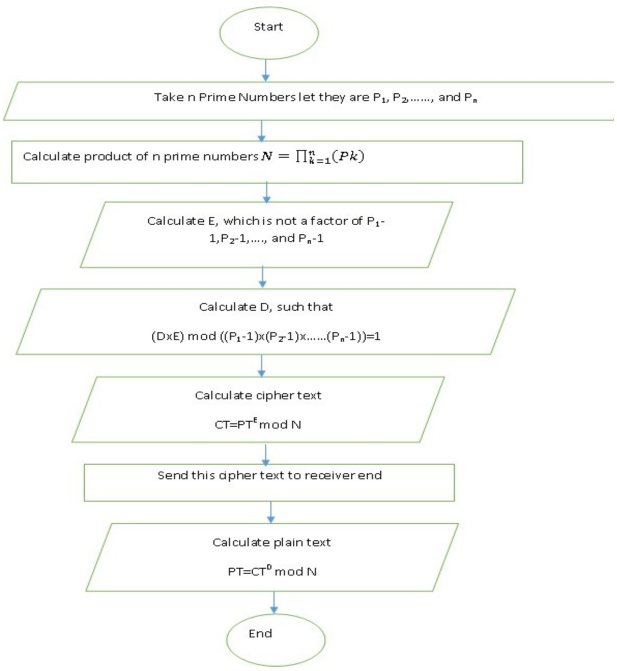


Fig. 8.Flowchart of Proposed Algorithm

IV. PROPOSED ALGORITHM IMPLEMENTATION RESULT

The proposed algorithm is implemented in C and python. We take here 4 types of examples.

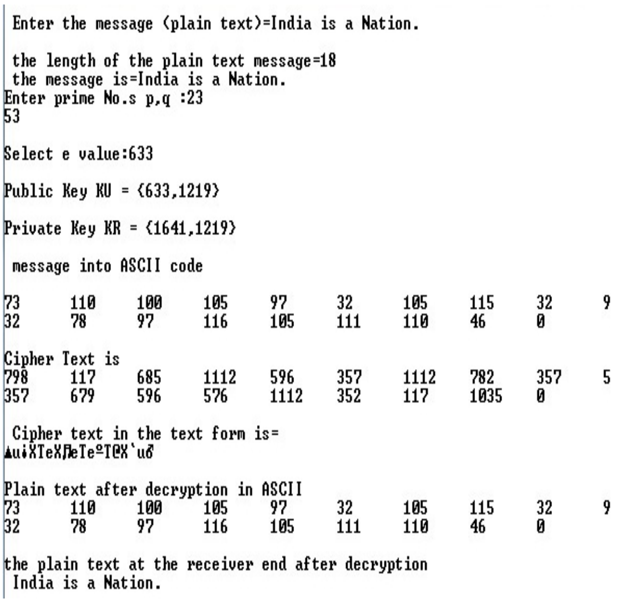
In these example there is plain text is same for all example that is "India is a Nation." and there are 2, 3, 4 and 5 different prime numbers are used and we calculate the value of N, encryption key value, decryption key value, Encryption time, Decryption time, cipher text and again plain text from cipher text for different combinations. We check variations in encryption and decryption time according to the prime numbers count and check the behavior of time graphs. Here the prime numbers are small for making calculations easy, but we can take large prime numbers in our practical life.

Table- I: List of Prime Numbers Used in this Experiment

	P1	P2	P3	P4	P5
1	23	53	11	37	17
2	29	59	13	41	19
3	31	61	17	43	23
4	37	67	19	47	29
5	41	71	23	53	31
6	43	73	29	59	37
7	47	79	31	61	41
8	53	83	37	67	43
9	59	89	41	71	47
10	61	97	43	73	53

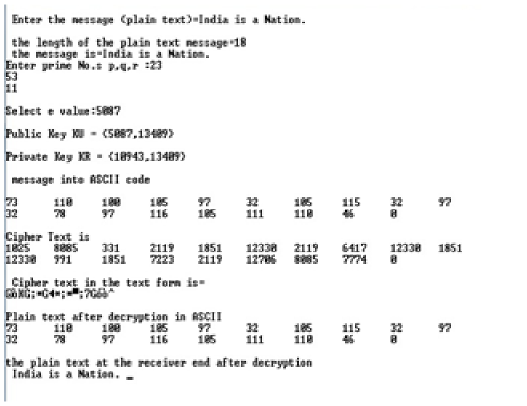
If we are using 2 prime numbers, then P1 and P2 are used. If we are using 3 prime numbers then P1, P2 and P3 are used. If we are using 4 prime numbers then P1, P2, P3and P4 are used.

If we are using 5 prime numbers then P1, P2, P3, P4 and P5 are used.



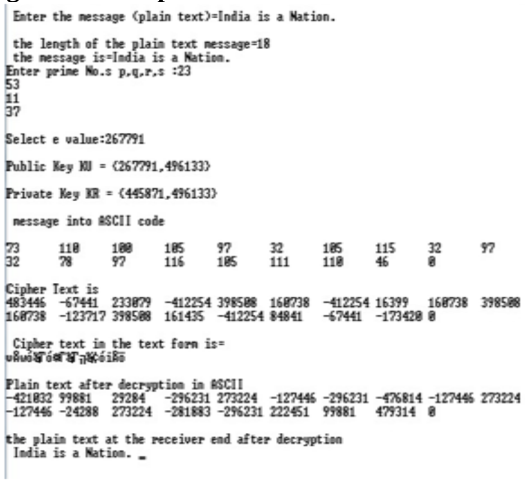
```
 Enter the message (plain text)=India is a Nation.

 the length of the plain text message=18
 the message is=India is a Nation.
Enter prime No.s p,q :23
53

Select e value:633

Public Key KU = (633,1219)

Private Key KR = (1641,1219)

 message into ASCII code

73      110     100     105     97      32      105     115     32      9
32      78      97      116     105     111     110     46      0

Cipher Text is
798     117     685     1112    596     357     1112    782     357     5
357     679     596     576     1112    352     117     1035    0

 Cipher text in the text form is=
▲uiXTeX₧eTeºI@X'uð

Plain text after decryption in ASCII
73      110     100     105     97      32      105     115     32      9
32      78      97      116     105     111     110     46      0

the plain text at the receiver end after decryption
 India is a Nation.
```

Fig. 9.Example for Two Prime Numbers



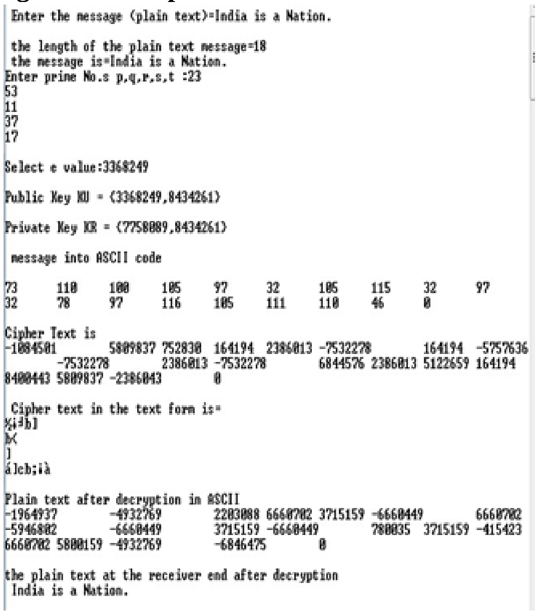
```
 Enter the message (plain text)=India is a Nation.

 the length of the plain text message=18
 the message is=India is a Nation.
Enter prime No.s p,q,r :23
53
11

Select e value:5087

Public Key KU = (5087,13409)

Private Key KR = (10943,13409)

 message into ASCII code

73      110     100     105     97      32      105     115     32      97
32      78      97      116     105     111     110     46      0

Cipher Text is
1025    8085    331     2119    1851    12330   2119    6417    12330   1851
12330   991     1851    7223    2119    12706   8085    7774    0

 Cipher text in the text form is=
GbKG;=G4=;=■;7Gbb^

Plain text after decryption in ASCII
73      110     100     105     97      32      105     115     32      97
32      78      97      116     105     111     110     46      0

the plain text at the receiver end after decryption
 India is a Nation. _
```

Fig. 10. Example for Three Prime Numbers



```
 Enter the message (plain text)=India is a Nation.

 the length of the plain text message=18
 the message is=India is a Nation.
Enter prime No.s p,q,r,s :23
53
11
37

Select e value:267791

Public Key KU = (267791,496133)

Private Key KR = (445871,496133)

 message into ASCII code

73      110     100     105     97      32      105     115     32      97
32      78      97      116     105     111     110     46      0

Cipher Text is
483446  -67441  233079  -412254 398508  160738  -412254 16399   160738  398508
160738  -123717 398508  161435  -412254 84841   -67441  -173420 0

 Cipher text in the text form is=
uÑuòŠóæŠ╥ÑçiÑõ

Plain text after decryption in ASCII
-421032 99881   29284   -296231 273224  -127446 -296231 -476814 -127446 273224
-127446 -24288  273224  -281083 -296231 222451  99881   479314  0

the plain text at the receiver end after decryption
 India is a Nation. _
```

Fig. 11. Example for Four Prime Numbers



```
 Enter the message (plain text)=India is a Nation.

 the length of the plain text message=18
 the message is=India is a Nation.
Enter prime No.s p,q,r,s,t :23
53
11
37
17

Select e value:3368249

Public Key KU = (3368249,8434261)

Private Key KR = (7758089,8434261)

 message into ASCII code

73      110     100     105     97      32      105     115     32      97
32      78      97      116     105     111     110     46      0

Cipher Text is
-1084501        5809837 752830  164194  2386013 -7532278        164194  -5757636
        -7532278        2386013 -7532278        6844576 2386013 5122659 164194
8400443 5809837 -2386043        0

 Cipher text in the text form is=
%i³b]
K
]
álcb;ià

Plain text after decryption in ASCII
-1964937        -4932769        2203088 6660702 3715159 -6660449        6660702
-5946802        -6660449        3715159 -6660449        780035  3715159 -415423
6660702 5800159 -4932769        -6846475        0

the plain text at the receiver end after decryption
 India is a Nation.
```

Fig. 12. Example for Five Prime Numbers

V. PERFORMANCE ANALYSIS OF PROPOSED ALGORITHM

10 sets for 2, 3, 4 and 5 prime numbers have been taken. Tables II,III,IV and V shows the values obtained during the use of two, three, four and five prime numbers.

Table- II: List of Values obtained using Two Prime Numbers

P_1	P_2	n	e	d	encryption time (in sec)	decryption time (in sec)
23	53	1219	633	1641	0.000062900000003197	0.000061999999997844
29	59	1711	429	1677	0.000057400000002872	0.000063700000054400
31	61	1891	799	1399	0.000069099999997491	0.000073200000027710
37	67	2479	779	2315	0.000058899999999085	0.000065299999957160
41	71	2911	277	4013	0.000055599999996048	0.000074499999966467
43	73	3139	319	2095	0.000058299999999178	0.000066600000038025
47	79	3713	511	2935	0.000060699999998803	0.000119300000014880
53	83	4399	1263	2711	0.000069000000003427	0.000068200000011840
59	89	5251	1095	3687	0.000074800000000153	0.000078700000030957
61	97	5917	5563	3187	0.000091199999999958	0.000078500000007585

Table- III: List of Values obtained using Three Prime Numbers

P_1	P_2	P_3	N	e	d	encryption time (in sec)	decryption time (in sec)
23	53	11	13409	5087	10943	0.0000921000000033700	0.0000945000000029950
29	59	13	22243	11957	13469	0.0000849999999985585	0.0000775999999973465
31	61	17	32147	27403	21667	0.0000900999999942087	0.0000864000000007081
37	67	19	47101	8689	25585	0.0000847000000021580	0.0000895000000014079
41	71	23	66953	45811	72091	0.0001067999999975200	0.0001084000000020070
43	73	29	91031	18749	123989	0.0000974000000013575	0.0001036000000027570
47	79	31	115103	52093	61237	0.0001096999999958820	0.0001018999999899960
53	83	37	162763	104861	126005	0.0001141999999987320	0.0001058000000000450
59	89	41	215291	133999	205839	0.0001196999999990570	0.0001091999999971450
61	97	43	254431	4283	186227	0.0000848999999902844	0.0001457999999985300

Table- IV: List of Values obtained using Four Prime Numbers

P_1	P_2	P_3	P_4	N	e	d	encryption time (in sec)	decryption time (in sec)
23	53	11	37	496133	257791	432511	0.0001244000000042430	0.0001199999999954570
29	59	13	41	911963	617993	1126457	0.0001297000000022310	0.0001265999999873200
31	61	17	43	1382321	675587	1271723	0.0001284000000083550	0.0001371999999975060
37	67	19	47	2213747	1403035	1429843	0.0001374999999939060	0.0001302000000009680
41	71	23	53	3548509	1670597	3166733	0.0001431000000025050	0.0001338000000004060
43	73	29	59	5370829	4226941	5490901	0.0002096000000051390	0.0001463999999913310
47	79	31	61	7021283	2868137	7946873	0.0001446999999927810	0.0001470999999924060
53	83	37	67	10905121	1542943	11135071	0.0001385000000198030	0.0001543000000197030
59	89	41	71	15285661	12370703	9366767	0.0001847000000054780	0.0001547999999900190
61	97	43	73	18573463	6919681	23585281	0.0001405000000431760	0.0001477999999792700

Table- V: List of Values obtained using Five Prime Numbers

P_1	P_2	P_3	P_4	P_5	N	e	d	encryption time (in sec)	decryption time (in sec)
23	53	11	37	17	8434261	3368249	7758089	0.0001458000000056360	0.0001405000000005430
29	59	13	41	19	17327297	5298563	12062507	0.0001500000000049800	0.0001534000000020800
31	61	17	43	23	31793383	21087041	18400961	0.0001553999999828190	0.0001479999999958180
37	67	19	47	29	64198663	36667549	28792117	0.0001725999999848680	0.0001637999999957170
41	71	23	53	31	1.1E+08	51873697	87112033	0.0001759999998967030	0.0001730000000179640
43	73	29	59	37	1.99E+08	1.49E+08	227267713	0.0001886000000013150	0.0001746000000366620
47	79	31	61	41	2.88E+08	64729411	184227691	0.0001738000000841570	0.0001877999998214360
53	83	37	67	43	4.69E+08	3.05E+08	394023173	0.0001965999999811170	0.0001861999999164250
59	89	41	71	47	7.18E+08	2.64E+08	513683237	0.0002696999999898250	0.0002264999993712990
61	97	43	73	53	9.84E+08	1.03E+08	465483839	0.0003010999989783160	0.0001972999998542950

Retrieval Number: C6278029320/2020©BEIESP
DOI: 10.35940/ijeat.C6278.029320
Journal Website: www.ijeat.org

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

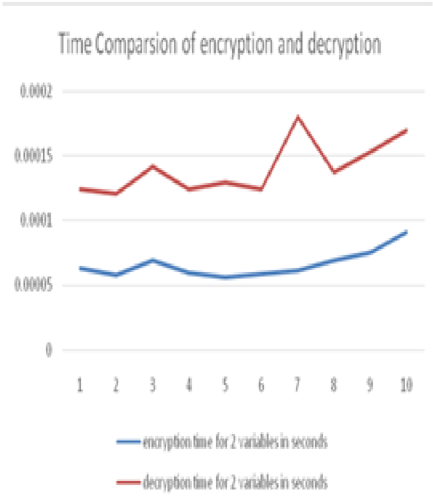


Fig. 13. Encryption-Decryption Time Graph for Two Prime Numbers (based on Table II)

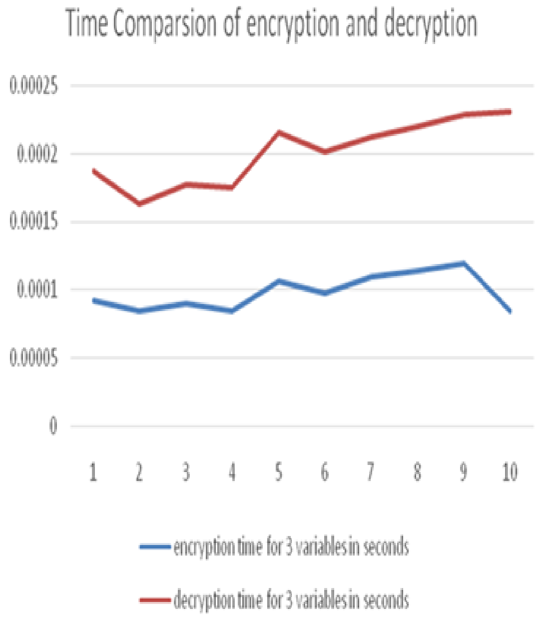


Fig. 14. Encryption-Decryption Time Graph for Three Prime Numbers (based on Table III)

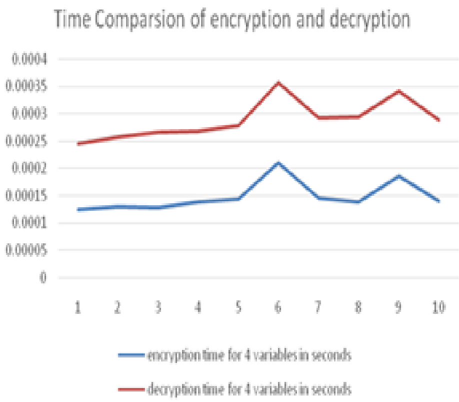


Fig. 15. Encryption-Decryption Time Graph for Four Prime Numbers (based on Table IV)

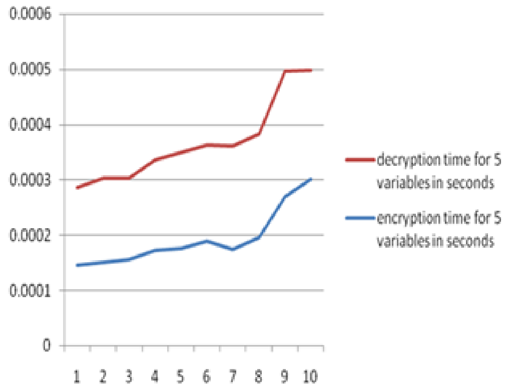


Fig. 16. Encryption-Decryption Time Graph for Five Prime Numbers (based on Table V)

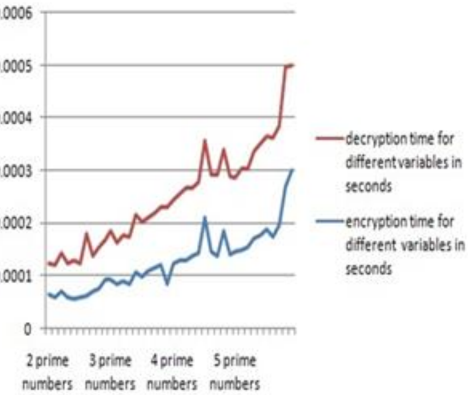


Fig. 17. Behavior of Encryption-Decryption Time Graph for all combinations from 2, 3, 4 and 5 Prime Numbers

By analysis of these graphs we can say that if we increase the prime number then encryption and decryption time will be increased in terms of e^x.

VI. ADVANTAGES OF PROPOSED ALGORITHM

1) It is very hard to find out the factors of N. In this case ((P_1-1), (P_2-1), (P_3-1) ……… (P_n-1)) because when we increase number of prime numbers then its product is also a big number.

2) The security aspects are not compromised here like confidentiality, availability, integrity, Authentication.

VII. CONCLUSION

At the end by comparing and checking all the parameters of proposed algorithm with existing algorithm, we can say that when we increase the number of prime numbers in RSA algorithm then its security also improves because it's hard to find the factor of N, while there are more than two prime numbers.

Encryption and Decryption time is depends on the value of e (encryption key) and d (decryption key) and here value of e is smaller because we are using more than 2 prime numbers so due to this the value of d is also not so big and by this process the encryption and decryption time is less.

REFERENCES

1. RSA algorithm using modified subset sum cryptosystem, Sonal Sharma, Computer and Communication Technology (ICCCT), pp-457-461, IEEE 2011
2. The large prime numbers based on genetic algorithm, hang Qing, (ICISIE) pp-434-437, IEEE 2011. .
3. An advanced secure (t, n) threshold proxy signature scheme based on RSA cryptosystem for known signers, Kumar, R, Dept. of Compute. Sci. and Eng, pp 293-298, IEEE 2010.
4. An efficient decryption method for RSA cryptosystem, Ren-Junn Hwang, Dept. of Compute. Sci. and Inf. Eng, pp-585-590, IEEE 2005.
5. A new RSA cryptosystem hardware design based on Montgomery's algorithm, Ching Chao Yang, Dept. of Electron. Eng, pp- 908-913, IEEE 1998.
6. A systolic RSA public key cryptosystem, Po – Song Chen, Dept. of Electron. Eng, pp 408-411, IEEE 1996.
7. "Secure Key Exchange using RSA in Extended Playfair Cipher Technique" Surendra Singh Chauhan, International Journal of Computer Applications (0975 – 8887) Volume 104 – No 15, October 2014.
8. Blocking method for RSA cryptosystem without expanding cipher length, NEC Corp, Kanagawa, Japan, pp 773-774, IEEE 1989.
9. A method for obtaining digital signatures and public key cryptosystems, R.Rivest, A.Shamir and L.Adleman "communication of the association for computing machinery " 1978, pp 120-126.
10. "A modified RSA cryptosystem based on 'n' prime numbers", B.Persis Urbana Ivy, Purshotam Mandiwa. Mukesh Kumar, International Journal Of Engineering And Computer Science ISSN:2319-7242 Volume1 Issue 2 Nov 2012 Page No. 63-66

AUTHOR'S PROFILES

Nitin Jain is working as Professor in AIT-CSE, Chandigarh University, Gharaun, India. He has more than 17 years of vast experience and depth knowledge of teaching at undergraduate and postgraduate level. His areas of research include Ubiquitous Computing, Network Security, and Information Security. He has published more than 10 research papers in National and International Journals and Conferences.

Surendra Singh Chauhan is working as a Research Scholar at Pratap University, Jaipur. He has 10 years of teaching experience and 2 years of corporate experience with Nokia Siemens Networks. He is working in the field cryptography and security from the past 8 years. He has published 8 research papers in International and National Conferences and Journals.

Alok Raj is an IT Enthusiast, who is passionate about exploring all the latest technologies from research perspective. He has deep interest and understanding of information security and data privacy, especially network security and cryptography. He is IRCA Certified ISO/IEC 27001:2013 Lead Auditor as well as C|EH, CPISI and HCNA certified. He is constantly transforming himself to improve his performance and evolving constantly and responding quickly and intuitively to the changing market dynamics.