International Journal of Engineering, Science and Mathematics
Vol. 9, Issue 8, August2020,
ISSN: 2320-0294 Impact Factor: 6.765
Journal Homepage: http://www.ijesm.co.in, Email: ijesmj@gmail.com
Double-Blind Peer Reviewed Refereed Open Access International Journal - Included in the International Serial Directories Indexed & Listed at: Ulrich's Periodicals Directory ©, U.S.A., Open J-Gage as well as in Cabell's Directories of Publishing Opportunities, U.S.A

A STUDY ON INTEGRITY AND AUTHENTICATION USING RSA AND SHA-3 ALGORITHMS FOR SECURED DATA COMMUNICATION

P.M.Pazhani Selvam*

Dr.S.S.Sujatha**

ABSTRACT

RSA is an algorithm suitable for message authentication. It also maintains integrity of the message. The RSA Digital Signature Algorithm applied on the encrypted data and a digital signature is produced. The digital signature maintains authentication as well as integrity. This is one approach. The Secured Hash Algorithm SHA-3 Keccak is applied on the encrypted data and a message digest is produced. The RSA Digital Signature Algorithm is applied on the message digest and a digital signature is produced. This is another one approach to maintain authentication and integrity. This paper compares the performance of these two approaches.

KEYWORDS:

Digital Signature
Message Digest
Authentication
Integrity
Hashing

Author correspondence:

P.M.Pazhani Selvam M.C.A.,M.Phil.,
Research Scholar, Computer Applications
Manonmaniam Sundaranar University, Tirunelveli.

1. INTRODUCTION

1.1.RSA Encryption Algorithm

RSA algorithm was developed in 1977 by Ron Rivest, Adi Shamir and Leonard Adleman for to secure data. It is a public key encryption algorithm. It was recognized by MIT in 1983. This algorithm consists three parts. They are key generation, encryption and decryption. The key generation algorithm is as follows.

1. Select two large prime numbers PN1,PN2.
2. Calculate N by multiplying PN1 and PN2.
3. Calculate I by multiplying PN1-1 and PN2-1
4. Select the value of E which is not the factor of I. E is a public key.
5. Calculate the private key D with the equation (D*E) Mod I = 0

Key generation is the process which consumes more time. Now the public key is (E,N) and the private key is (D,N). RSA key sizes are 512 bits,1024 bits, 2048 bits, 3072 bits and 4096 bits. The data encryption is done by the equation C = (P^E) Mod N. Here P is the plain text and C is cipher text. The data decryption is done by the equation P = (C^D)Mod N. Using RSA algorithm, it is possible to encrypt and decrypt data. It is also possible to exchange the key,which is used to encrypt and decrypt data, between the sender and receiver using RSA algorithm. The data is encrypted using the public key of the receiver. The data is decrypted using the private key of the receiver [1]

.

---

* Research Scholar, Computer Application, Manonmaniam Sundaranar University, Tirunelveli.

**Associate Professor, Department of MCA, S. T. Hindu College, Nagercoil.

1.2.RSA Digital Signature Algorithm

RSA Digital Signature Algorithm is based on RSA encryption algorithm. It also uses the same keys, produced by RSA encryption algorithms. This algorithm creates digital signature, which is used to guarantee authentication and integrity. This algorithm also consists three important parts. They are Key generation, Signing and Verification. Key generation portion uses the key pairs already generated by the RSA encryption algorithm. Sender creates digital signature using private key of the sender. The equation is S = (M^D) Mod N. Here M is the encrypted data, S is the digital signature and D is the private key of the sender. This is called signing work. The (M, S) pair is sent to the receiver. The receiver receives the pair (M, S) and verifies it using the equation M1= (S^E) Mod N. Here E is the

public key of the sender and M1 is the encrypted data. If M1 is same as M, the receiver accepts the data. Otherwise the data is rejected. This is called verification work [2].

The RSA is a secured algorithm. Due to its slowness, it is not generally preferred for the encryption or decryption work of large data. But it is an algorithm, which shows its efficiency in key exchange, message authentication, message integrity and non-repudiation work.

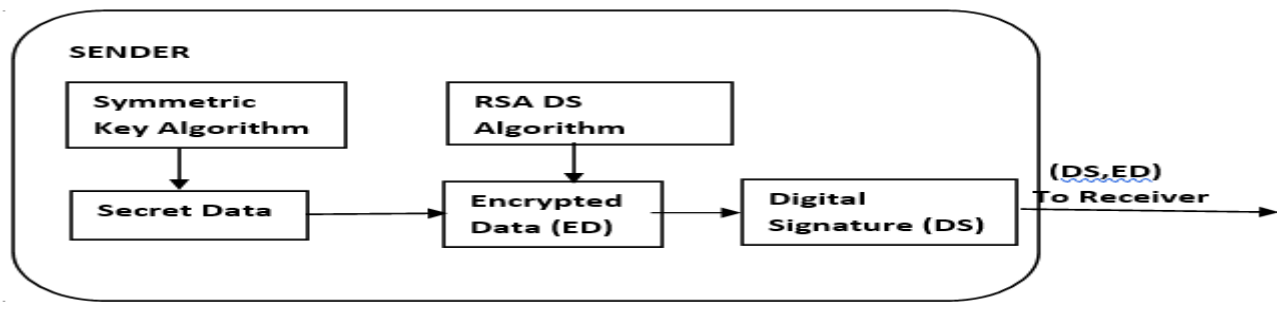
1.3.Secured Hash Algorithm(SHA-3 Keccak)

Secured hash algorithms are used to generate message digest. Message digest is used to confirm message integrity. Hashing algorithms and Secured hashing algorithms are used to create message digest. MD-2,MD-3,MD-4 and MD-5 are hashing algorithms. SHA-1, SHA-2 and its variants and SHA-3 and its variants are Secured hashing algorithms[9][11][12]. Generally hashing algorithms are one-way algorithms. Secured hashing algorithms are faster and provide more security to data.

Within Secured hashing algorithms, SHA-3 (Keccak) algorithm is better than others.It was developed by Guido Bertoni, Joan Daemen, Michaël Peeters and Gilles Van Assche.It was adopted by National Institute of Standards and Technology (NIST) in August 5, 2015 [10].It isa one-way function. The algorithm converts the data of any size to a fixed length message digest. The SHA-3 Keccak algorithm uses sponge technology. Absorbtion and Squeeze are the two important operations of the sponge. During the absorbtion, the message blocks are XORed with 1600 bits size state and transformed into one element. During the squeezing, output blocks are read from the element [3][11][12]. The length of message digest may be 224 bits, 256 bits ,384 bit and 512 bits. The message digest should always be unchanged.This algorithm can process data up to the length 2^{128}-1. The block size may be 1024.The word size is 64 bits. It requires 24 rounds to complete its work [6] [7][8].

2.PROPOSED WORK

In this paper, two works are proposed. In the first proposed work, create digital signature



directly using RSA algorithm. Using one symmetric key algorithm like AES the secret data is encrypted. On the encrypted data, apply RSA Digital Signature Algorithm using the private key of the sender and a digital signature is created.

Fig.2.1.Creation of Digital Signature from encrypted data using RSA DS Algorithm

In this first proposed work, the encrypted data and digital signature are sent to the receiver as a pair. The receiver receives both encrypted data and digital signature. The receiver applies RSA Digital Signature Algorithm on the received digital signature and verifies it. For the purpose of verification, RSA DS Algorithm uses public key of the sender. If the output of the verification process is same as the encrypted data, the authentication and integrity are confirmed. The encrypted data is accepted by the receiver.Then using the symmetric key algorithm,the encrypted data is decrypted and receiver gets the original plain text. If the verification result is wrong, the encrypted data and digital signature are rejected by the receiver. The receiver asks the sender to resend the data. To improve the security, the digital signature is created from the encrypted data. The time taken to create digital signature and verification of digital signature is measured.

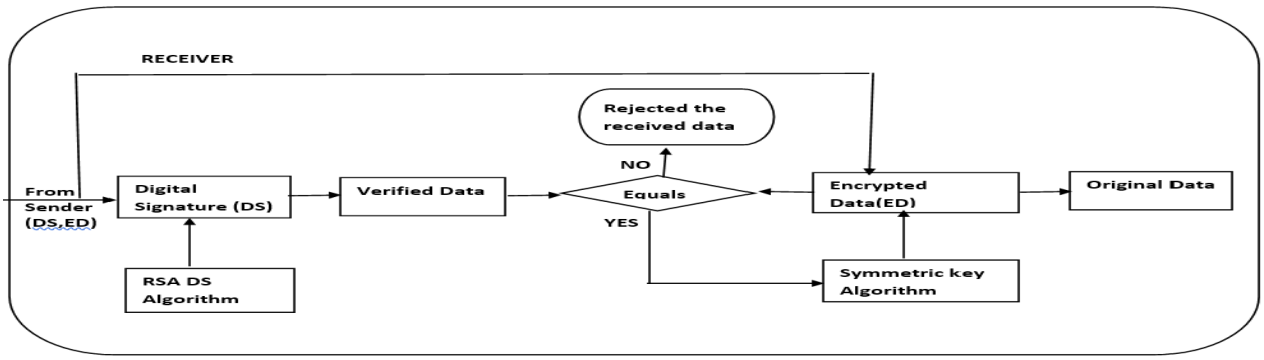
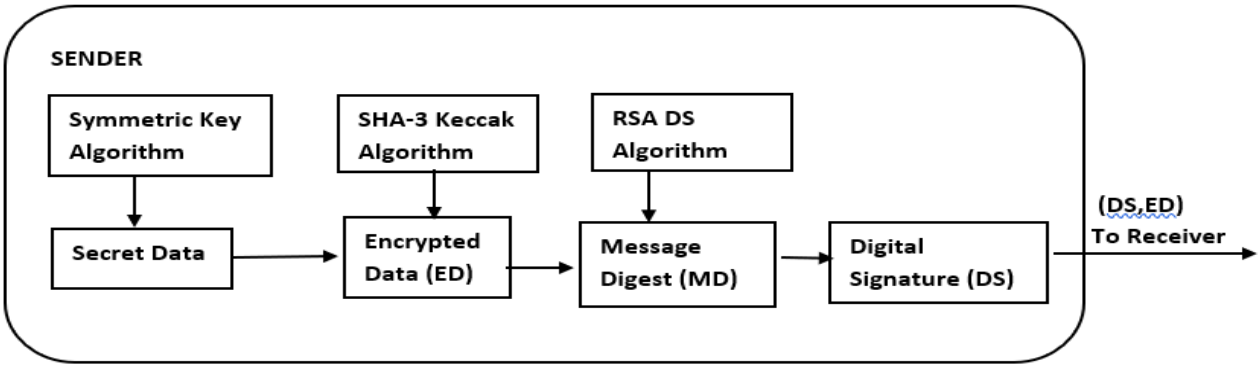


Fig.2.2. Verification of Digital signature

In the second proposed work, create message digest from the encrypted data using SHA-3 Keccak algorithm. The sender

applies RSA Digital Signature Algorithm on the message digest and create digital signature. To create digital signature the

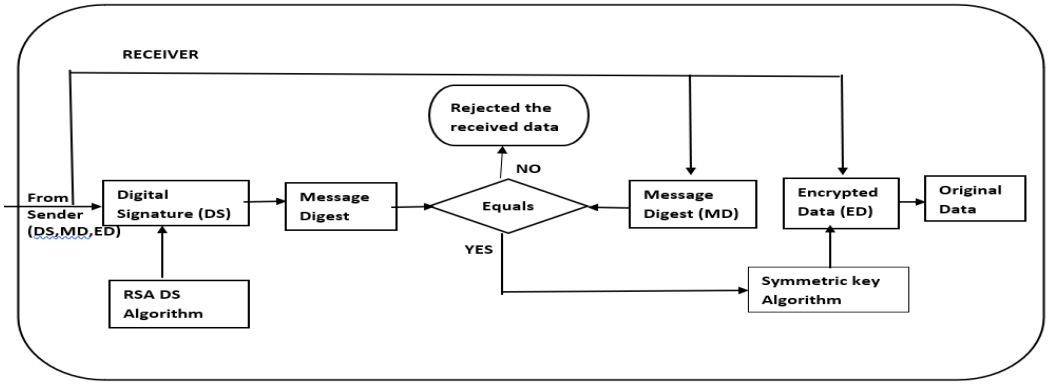


private key of the sender is used.

Fig.2.3. Creation of digital signature from the message digest

.

The encrypted data, message digest and digital signature are sent to the receiver. The receiver receives the triple. The receiver applies RSA DS Algorithm using the public key of the sender on the received digital signature and produce a message digest. Then this message digest is compared with the received message digest. If they are equal, the



encrypted data is accepted by the receiver. Using the symmetric key algorithm, the encrypted data is decrypted and receiver receives the original plaintext. If the two message digests are not equal, the receiver rejects the received triple and requests the sender to resend the data. In the second proposed work, for the security purpose, the message digest is created from the encrypted data [4]. The time taken to createdigital signature, message digest and verification of message digestis measured. These two times are compared to find the timing difference between these two proposed works.

Fig.2.4. Verification of Digital Signature of Message Digest

3.RESULTS AND DISCUSSION

3.1.Sample Outputs

Output.3.1.1. Create a digital signature from the encrypted data and verified it using RSA-DS Algorithm

RSA-DSA algorithm
Calculate the key pairs of the sender
Enter a prime number p1: 101
Enter a prime number q1: 103
Elapsed time is 0.015116 seconds.
Public key of sender is (6451,10403)
Private key of sender is (5251,10403)
Calculate the key pairs of the receiver
Enter a prime number p: 107
Enter a prime number q: 109
Elapsed time is 0.002690 seconds.
Public key of receiver is (1117,11663)
Private key of receiver is (6877,11663)

Original message
This is the text which is going to be encrypted.
Using RSA method this text is going to be encrypted.
Then using RSA digital signature algorithm a digital signature is created.
Then it is verified. if verification is over, then the text is decrypted.

Encryption
Elapsed time is 0.012858 seconds.
The encrypted message is

Signing
Elapsed time is 0.009738 seconds.
The Signed text is

Verification
Elapsed time is 0.006735 seconds.
The verified text is
Verified

Decryption
Elapsed time is 0.007863 seconds.
The Decrypted text is
Thisisthetextwhichisgoingtobeencrypted.UsingRSAmethodthistextisgoingtobeencrypted.ThenusingRSAdigitalsignaturealgorithmadigitalsignatureiscreated.Thenitisverified.ifverificationisover,thenthetextisdecrypted.

Output.3.1.2. Create a digital signature from a message digest and verified it using RSA DS Algorithm

```
SHA-3 KECCAK Algorithm and RSA DS Algorithm
Enter a prime number p1:  101
Enter a prime number q1: 103
Elapsed time is 0.019445 seconds.
Public key of sender is (9341,10403)
Private key of sender is (4061,10403)
Calculate the key pairs of receiver\n
Enter a prime number p:  107
Enter a prime number q: 109
Elapsed time is 0.007400 seconds.
Public key of receiver is (7507,11663)
Private key of receiver is (4915,11663)

The original Text is:
Thisisthetextwhichisgoingtobeencrypted.UsingRSAmethodthistextisgoingtobeencrypted.ThenusingRSAdigit
alsignaturealgorithmadigitalsignatureiscreated.Thenitisverified.ifverificationisover,thenthetextisdecrypted.

The encrypted text is
Elapsed time is 0.015242 seconds.
[illegible]

Message Digest of the encrypted text is:
Elapsed time is 1.553940 seconds.
3ab01aa4ad6f7969ac3e9a78e23650c07030282a4911f0645cdda5b04479d2e78fafb54204e43cedec21251948ce53
dbea220302928e34575dca18312b190c15

The signed message on message digest is:
Elapsed time is 0.008068 seconds.
[illegible]

The verified text is:
Elapsed time is 0.007520 seconds.
3ab01aa4ad6f7969ac3e9a78e23650c07030282a4911f0645cdda5b04479d2e78fafb54204e43cedec21251948ce53
dbea220302928e34575dca18312b190c15
verified

The decrypted text is:
Elapsed time is 0.027159 seconds.
Thisisthetextwhichisgoingtobeencrypted.UsingRSAmethodthistextisgoingtobeencrypted.ThenusingRSAdigit
alsignaturealgorithmadigitalsignatureiscreated.Thenitisverified.ifverificationisover,thenthetextisdecrypted
```

3.2.Tables

Table.3.2.1. Time taken to generate RSA Receiver keys and Digital Signature

Prime number 1	Prime number 2	Time taken togenerate Receiver keys(seconds)	Data size(bytes)	Time taken to generateDigitalSignature(seconds)	Total Time(seconds)
107	109	0.0129	254	0.0097	0.0226
521	523	0.0214	254	0.0070	0.0284
1019	1021	0.0252	254	0.0096	0.0348
2017	2027	0.2056	254	0.0114	0.2170
3019	3023	0.5088	254	0.0103	0.5191
107	109	0.0016	1423	0.0215	0.0231
521	523	0.0061	1423	0.0183	0.0244
1019	1021	0.0176	1423	0.0278	0.0454
2017	2027	0.2454	1423	0.0254	0.2708
3019	3023	0.5128	1423	0.0248	0.5376
107	109	0.0049	15834	0.1050	0.1099
521	523	0.0105	15834	0.1373	0.1478
1019	1021	0.0246	15834	0.1320	0.1566
2017	2027	0.1547	15834	0.1697	0.3244
3019	3023	0.5353	15834	0.1889	0.7242
107	109	0.0049	41425	0.2378	0.2427
521	523	0.0110	41425	0.3293	0.3403
1019	1021	0.0142	41425	0.3554	0.3696
2017	2027	0.1529	41425	0.4779	0.6308
3019	3023	0.9878	41425	0.4625	1.4503

As the key size increases, the time to generate keys also increases. As the data size increases, the time to generate digital signature also increases. Random number generation by the program also the causes the time variation of key generation process.

Table.3.2.2. Time taken to generate RSA Receiver keys and Verification work

Prime number1	Prime number2	Time taken to generate Receiver keys(seconds)	Data size(bytes)	Verification Time(seconds)	Total Time(seconds)
107	109	0.0129	254	0.0067	0.0196
521	523	0.0214	254	0.0083	0.0297
1019	1021	0.0252	254	0.0100	0.0352
2017	2027	0.2056	254	0.0111	0.2167
3019	3023	0.5088	254	0.0162	0.5250
107	109	0.0016	1423	0.0175	0.0181
521	523	0.0081	1423	0.0222	0.0303
1019	1021	0.0176	1423	0.0217	0.0393
2017	2027	0.2454	1423	0.0230	0.2684
3019	3023	0.3728	1423	0.0264	0.3992
107	109	0.0049	15834	0.1047	0.1096
521	523	0.0105	15834	0.1340	0.1445
1019	1021	0.0246	15834	0.1474	0.1720
2017	2027	0.1547	15834	0.1765	0.3312
3019	3023	0.5353	15834	0.1820	0.7173
107	109	0.0049	41425	0.2707	0.2756
521	523	0.0110	41425	0.3366	0.3476
1019	1021	0.0142	41425	0.3435	0.3577
2017	2027	0.1529	41425	0.4680	0.6209
3019	3023	0.9878	41425	0.4478	1.4356

As the key size increases, the time to generate keys also increases. As the data size increases, the time to verify digital signature also increases. Random number generation by the program also the causes the time variation of key generation process. Generally, the time taken to verify digital signature is less than the time taken to generate digital signature in the first proposed work.

Table.3.2.3. Time taken to generate RSA receiver keys, Message Digest and Digital Signature

Prime Number 1	Prime Number 2	Time taken to generate RSA receiver Keys(seconds)	Data size (Bytes)	Time taken to generate Message Digest (seconds)	Time taken to generate Digital Signature (seconds)	Total Time (seconds)
101	103	0.0249	254	1.4495	0.0096	1.4840
563	567	0.0453	254	1.4532	0.0099	1.5081
1009	1013	0.0395	254	1.4587	0.0108	1.5090
2003	2011	0.1447	254	1.4698	0.0115	1.6260
3001	3011	0.4586	254	1.4699	0.0097	1.9382
101	103	0.0212	1423	5.2345	0.0098	5.2655
563	567	0.0410	1423	5.4563	0.0103	5.5076
1009	1013	0.0513	1423	5.6754	0.0094	5.7361
2003	2011	0.2085	1423	5.7834	0.0095	6.0014
3001	3011	0.4623	1423	5.8860	0.0098	6.3581
101	103	0.0202	15834	65.0974	0.0087	65.1263
563	567	0.0343	15834	65.1567	0.0118	65.2028
1009	1013	0.0303	15834	66.4534	0.0110	66.4947
2003	2011	0.2378	15834	66.5103	0.0099	66.7580
3001	3011	0.4164	15834	67.9254	0.0111	68.3529
101	103	0.0284	41425	163.3423	0.0073	163.3780
563	567	0.0308	41425	163.5410	0.0053	163.5771
1009	1013	0.0685	41425	164.4563	0.0057	164.5305
2003	2011	0.2831	41425	164.6544	0.0062	164.9437
3001	3011	0.4057	41425	165.2501	0.0060	165.6618

As the data size increases, the time to generate message digest also increases. The time to generate keys also increases as the key size increases. Message digest size is fixed even if the file sizes may vary. To generate message digest, huge amount of time is needed.

Table.3.2.4. Time taken to generate RSA receiver keys and to verify Digital Signature

Prime Number 1	Prime Number 2	Time taken to generate RSA receiver Keys(seconds)	Data size (Bytes)	Time taken to verify Message Digest (seconds)	Total Time (seconds)
101	103	0.0249	254	0.0074	0.0323
563	567	0.0453	254	0.0069	0.0522
1009	1013	0.0495	254	0.0076	0.0571
2003	2011	0.1447	254	0.0085	0.1532
3001	3011	0.4586	254	0.0075	0.4661
101	103	0.0212	1423	0.0076	0.0288
563	567	0.0410	1423	0.0081	0.0491
1009	1013	0.0513	1423	0.0083	0.0596
2003	2011	0.2085	1423	0.0079	0.2164
3001	3011	0.4623	1423	0.0084	0.4707
101	103	0.0202	15834	0.0072	0.0274
563	567	0.0343	15834	0.0070	0.0413
1009	1013	0.0373	15834	0.0080	0.0453
2003	2011	0.2378	15834	0.0074	0.2452
3001	3011	0.4164	15834	0.0092	0.4256
101	103	0.0284	41425	0.0092	0.0376
563	567	0.0308	41425	0.0074	0.0382
1009	1013	0.0685	41425	0.0077	0.0762
2003	2011	0.2831	41425	0.0089	0.2920
3001	3011	0.4057	41425	0.0085	0.4142

As the key size increases, the time taken to generate keys also increases. Generally, the time taken to verify the digital signature of message digest is more than the time taken to sign the message digest in the second proposed work.

signature directly from encrypted data. Work.2. Create digital signature from message Table.3.2.5. Comparison between two proposed works (Work.1. Create digital digest and message digest is created from encrypted data**)**

Prime number 1	Prime number2	Data size(bytes)	Time taken by Work1	Time taken by Work2	Time taken by Verification Work1	Time taken by verification work2
101	103	254	0.0226	1.4840	0.0196	0.0323
563	567	254	0.0284	1.5081	0.0297	0.0522
1009	1013	254	0.0348	1.5090	0.0352	0.0571
2003	2011	254	0.2170	1.6260	0.2167	0.1532
3001	3011	254	0.5191	1.9382	0.5250	0.4661
101	103	1423	0.0231	5.2655	0.0181	0.0288
563	567	1423	0.0244	5.5076	0.0303	0.0491
1009	1013	1423	0.0454	5.7361	0.0393	0.0596
2003	2011	1423	0.2708	6.0014	0.2684	0.2164
3001	3011	1423	0.5376	6.3581	0.3992	0.4707
101	103	15834	0.1099	65.1263	0.1096	0.0274
563	567	15834	0.1478	65.2028	0.1445	0.0413
1009	1013	15834	0.1566	66.4947	0.1720	0.0453
2003	2011	15834	0.3244	66.7580	0.3312	0.2452
3001	3011	15834	0.7242	68.3529	0.7173	0.4256
101	103	41425	0.2427	163.3780	0.2756	0.0376
563	567	41425	0.3403	163.5771	0.3476	0.0382
1009	1013	41425	0.3696	164.5305	0.3577	0.0762
2003	2011	41425	0.6308	164.9437	0.6209	0.2920
3001	3011	41425	1.4503	165.6618	1.4356	0.4142

Creating message digest using SHA-3 Keccak algorithm consumes more amount of time. So, the second proposed work takes much time than the first proposed work. Creating digital signature directly from the encrypted data using RSA Digital Signature Algorithm is a work which takes less time and also secured.

3.3.Charts

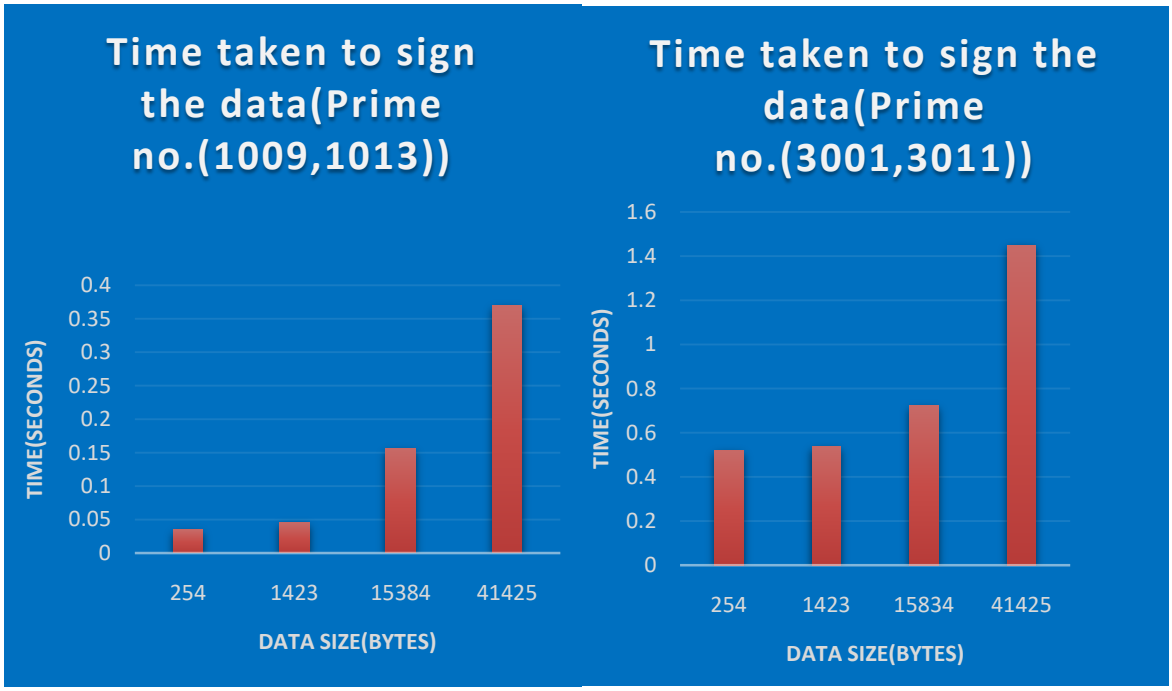


Fig.3.3.1. Time taken to sign the data using RSA-DS Algorithm. (Signing time = Keys generation time + Signing time)

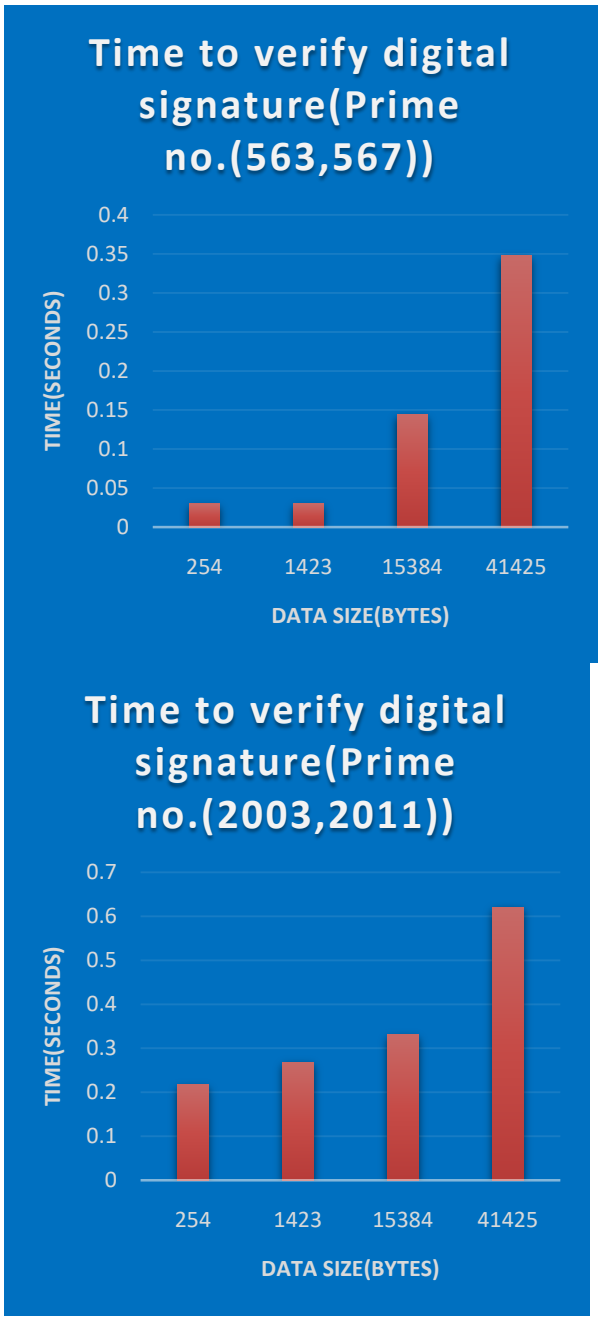


Fig.3.3.2. Time taken to verify the digital signature using RSA-DS Algorithm. (Verification time = Keys generation time + Verification time)

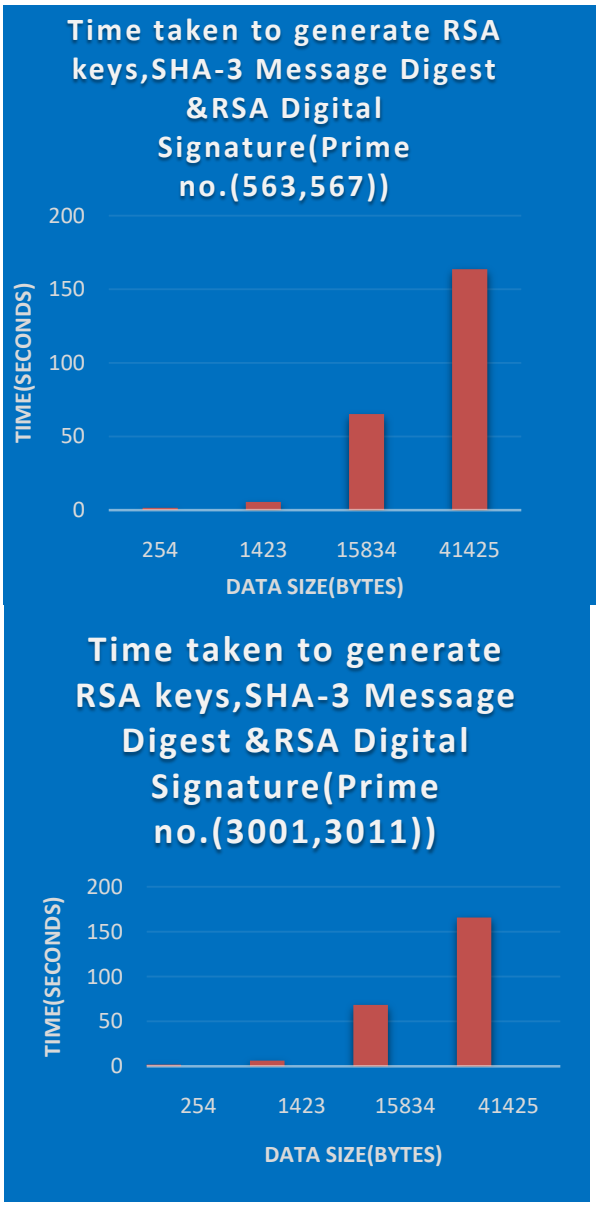


Fig.3.3.3. Time taken to sign the message digest(RSA Key generation time+ SHA-3 Message Digest generation time+ time taken to sign the message digest)

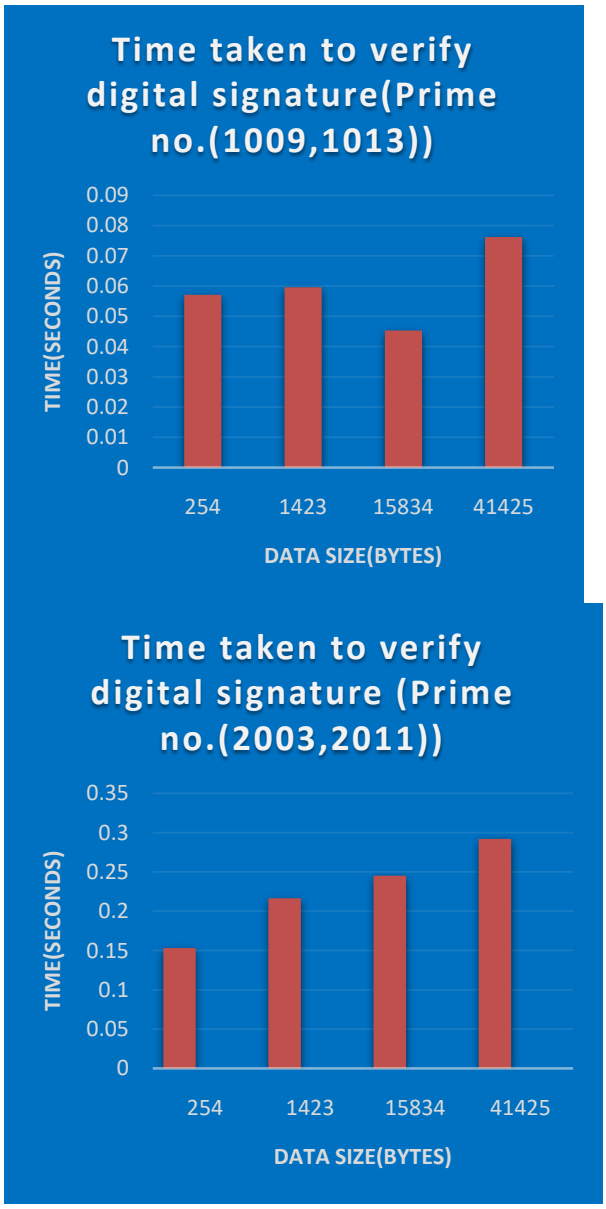


Fig.3.3.4. Time taken to verify the digital signature(RSA Key generation time+ time taken to verify the digital signature)

4.CONCLUSION

The RSA algorithm ensures message authentication by creating digital signature. The digital signature also ensures message integrity. If there are any changes in the message, during data transmission, then it is not possible to produce the same digital signature again. The verification process will also be failed. Thus, the digital signature is used to guarantee the message integrity. In this paper, the RSA algorithm is directly applied on encrypted data and produces digital signature. This work consumes less time than the other work, which creates message digest from encrypted data using SHA-3(Keccak) algorithm and creates digital signature from the message digest using RSA Digital Signature Algorithm.

It is concluded that RSA algorithm is a secured algorithm for message integrity and message authentication.

REFERENCES

[1] "What is RSA encryption and how does it work?", JOSH LAKE, December 10, 2018, "https://www.comparitech.com/blog/information-security/rsa-encryption/"

[2]"RSA Digital Signature Scheme using Python", Ami Munshi, "https://www.geeksforgeeks.org/rsa-digital-signature-scheme-using python/#:~:text=string%20in%20Python,RSA%20Digital%20Signature%20Scheme%20using%20Python,Private%20key%20is%20kept%20private."

[3]Hashing Algorithms, Jscrambler, October 18, 2016, "https://blog.jscrambler.com/hashing-algorithms".

[4]"SHA-3HASH",DavidHill, "https://in.mathworks.com/matlabcentral/fileexchange/71760-sha-3-hash".

[5]."SHA-3Cryptographic Hash Algorithms", Movable Type Scripts, "https://www.movable-type.co.uk/Scripts/sha3.html".

[6]Behrouz A. Forouzan, "Cryptography and Network Security", McGrawHill,2008.

[7]Imad Fakhri Alshaikhli, Mohammad A. Alahmad, "COMPARISON AND ANALYSIS STUDY OF SHA-3 FINALISTS", 2012 International Conference on Advanced Computer Science Applications and Technologies.

[8]Imad Fakhri Alshaikhli, Mohammad A. Alahmad, Khanssaa Munthir, " Hash Function of Finalist SHA-3: Analysis Study", International Journal of Advanced Computer Science and Information Technology (IJACSIT) Vol. 2, No. 2, April 2013, Page: 1-12, ISSN: 2296-1739.

[9]Nithin R. Chandrana ,Ebin M. Manuelb," Performance Analysis of Modified SHA-3 ", International Conference on Emerging Trends in Engineering, Science and Technology (ICETEST- 2015).

[10]Jeethu Jamesa , Karthika Rb , Nandakumar Rc, " Design & Characterization of SHA 3- 256 bit IP core", International Conference on Emerging Trends in Engineering, Science and Technology (ICETEST - 2015)

[11] S.Neelima, R.Brindha, "A Low Power FGPA Implementation of SHA3 Design", International Journal of Advanced Science and Technology,2019.

[12] S.Neelima, R.Brindha, "512 bit-SHA3 design approach and implementation of field programmable gate arrays", International Journal of Reconfigurable and Embedded Systems, November 2019.