

THE COMPARATIVE ANALYSIS OF CLASSICAL RSA WITH MODIFIED RSA IN TERMS OF THROUGHPUT, BITRATE AND DELAY

Author: Jagdeep Kaur¹; Dr. Tanupreet singh²

Research Scholar, CSE Department, Amritsar College of Engineering and Technology, Amritsar¹;

Prof. and HOD, ECE Department, Amritsar College of Engineering and Technology, Amritsar²;

E-mail: er_jagdeep86@yahoo.co.in¹; tanupreet.singh@gmail.com²

ABSTRACT

In this paper, we compare modified RSA with Classical RSA. Modified RSA is a robust algorithm that is combination of (RSA + bit stuffing) for the prevention of DNS server from DNS spoofing. Simulation has been carried out using NS2.35. It is analyzed that modified RSA provides better security than Classical RSA. RSA provides digital signature while bit stuffing is used for digesting the message so that it remains confidential for the intruder, if any changes persists destination can't decipher the message with the help of public key of sender.

Keywords: Modified RSA, Bit Stuffing, Comparison on the basis of Throughput, Bit rate and Delay.

1. INTRODUCTION

RSA system is one of the most common public key password systems. In addition to other domain it successfully provided security to the electronic based commerce. Encryption of plain text in asymmetric key encryption is based on a public key and a corresponding private key. Document authentication and digital signature are other advantages of RSA public key cryptosystems [15].

SSL (Secure Socket Layer) was designed by Netscape to perform a secure communication between a client and a server. SSL uses three interdependent cryptographic algorithms. The first

function is authentication. It is used to allow the client to identify the server and optionally allow the server to identify the client. SSL uses digital certificates to authenticate servers. SSL standard allows the concerned components to negotiate the encryption, authentication and integrity mechanism to use [16].

2. RELATED WORK

The most common cryptographic algorithm used to achieve authentication is the RSA algorithm. The second function is confidentiality that is used to keep the communication confidential. It uses symmetric cryptography. Integrity is the last function used to ensure the integrity of the data against snooping. This is performed using message digests. Checksum of the message is used for message digest [7]. The authentication in SSL did using RSA. Then a modified version of SSL was published using 1024 bits which is measured to be more secure but now it is currently recommended to use 2048 bits key for better secure communication. The RSA used in SSL depends on the integer arithmetic. In order to generate a key with size 512 bits we need two distinct primes each with 256 bits size. 512 bits is equivalent to 155 decimal digits [17].

The standard RSA Algorithm used for authentication is as follows [17]:

1. Firstly find two large primes p and q and find their product $n=p*q$.

2. Secondly find an integer d co-prime to $\phi(n) = (p-1)(q-1)$.
3. Compute e from $e \cdot d \equiv 1 \pmod{(p-1)(q-1)}$.
4. Then public key is broadcasted, which is the pair of numbers (e, n) .
5. And message to be transmitted i.e. m , say as a sequence of integers (m) each in the range 1 to n .
6. Now encrypt each message, m , using the public key by applying the rule $C = m^e \pmod{n}$.
7. The receiver will decrypts the message using the rule $m = C^d \pmod{n}$.

3. PROPOSED WORK

In this section, we will briefly present the modified version of RSA in the BIT STUFFING RSA. The idea behind this paper is to modify the RSA key from 512 bits to 512 bits by applying BIT STUFFING instead of ordinary integers using the same prime numbers used by the 512 bits. In this way we are making DNS more secure by using 512 bits and 512 bits for prime numbers. Confidentiality is the second function used in SSL [2].

For the purpose of making message secure, we are using BIT STUFFING technique that will take care of message confidentiality because with the help of bit stuffing we padded message to maintain a fixed length. An arbitrary length of message is digested into fixed length message; active intruder can't decrypt it until they have knowledge about the padding bit added into arbitrary length input message.

Bit stuffing is used for various purposes, such as for bringing bit streams that do not necessarily have the same or rationally related bit rates up to a common rate, or to fill buffers or frames. The location of the stuffing bits is communicated to the receiving end of the data link, where these extra where these extra bits are removed to return the bit streams to their original bit rates or form. Bit stuffing may be used to synchronize several channels before multiplexing or to rate-match two single channels to each other but we will use it as the bits which are not important for messaging but will be useful for security. Like these extra bits will confused to the attacker and then attacker will not determine the extra bits because only the authorized

person will know which extra bits there are.

Modified RSA

The modified RSA is reliable and more secure than the Classical RSA. The modified algorithm is as follows:

1. Find two large prime numbers p and q and compute their product $n = p \times q$ with bit stuffing mechanism.
2. Find an integer d that is co-prime to $\phi(n) = (p-1)(q-1)$.
3. Compute e from $e \cdot d \equiv 1 \pmod{(p-1)(q-1)}$.
4. Broadcast the public key, that is, the pair of numbers (e, n) .
5. Represent the message to be transmitted, m , say as a sequence of integers $\{M\}$ each in the range 1 to n .
6. Encrypt each message, m , using the public key by applying the rule $C = M^e \pmod{n}$.
7. Add the random into C using $C' = C + BS$.
8. Now remove random number at receiver side as $C = C' - BS$.
9. The receiver decrypts the message using the rule $m = C^d \pmod{n}$.

This way we can easily encrypt and decrypt the message and can secure the communication.

We have proposed a framework for securing the communication between the client and the server in DNS. The idea has been come from RSA when it was hacked. This paper also discussed about the bit stuffing used in the modified RSA algorithm, the number which will be generated will not repeat. Lastly we have designed an algorithm which will be providing a secure communication. So In this memorandum we are trying to provide a robust algorithm that is combination of (RSA + bit stuffing), i.e. RSA accounts for legacy of user by providing the Digital signature, while bit stuffing is used digesting the message so that it remain confidential for the intruders, if any changes persists (violation of integrity) destination can't decipher the message with the help of public key of sender. So this way we can provide more robustness for DNS [1].

3.1 GRAPHICAL ANALYSIS

We have done comparative analysis of the work

done. The following are the graphs showing Throughput, Bit rate, Delay, Packets Loss Rate etc comparison of classical RSA with modified RSA.

3.1.1 Throughput

Throughput is the number of packet that is passing through the channel in a particular unit of time. In fig.12 we see an increase in throughput upto a certain point where hacker enables. When the hacker enables Throughput remains constant. There is again an increase in Throughput after disabling hacker by using our proposed technique that we have implemented in this paper.



Fig.1 Throughput (Y-axis shows Bandwidth and X-axis shows simulation time)

Table 1: Throughput

Time	Bandwidth(bps)
0	0
2 (60 sec)	32000
3 (90 sec)	48000
5 (150 sec)	80000

As shown in the Table1, Throughput was minimum at time interval 60 sec the time when hacker got enabled. After applying modified RSA the hacker is disabled and throughput increases upto a certain limit. The comparison of Throughput for classical RSA and modified RSA is shown in Fig.2.

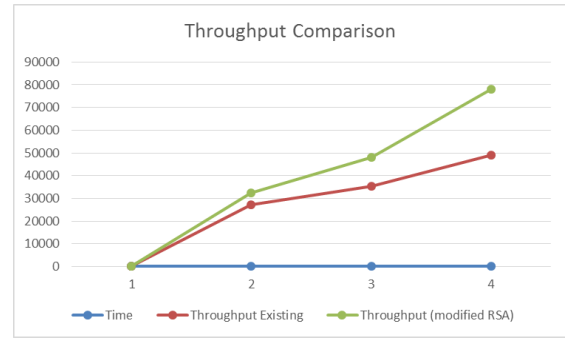


Fig.2. Throughput Comparison

3.1.2 Bit rate

Bit rate is number of bits that passes through network in the given time.



Fig.3.Bit rate(Y-axis shows Bitrate and X-axis shows Simulation Time)

Table 2: Bit rate

Time	Bit Rate(bps)
0	0
2 (60 sec)	80000
4 (120sec)	12000
5 (150 sec)	19000

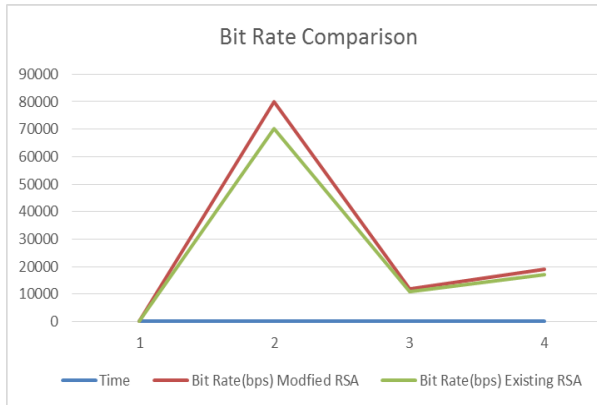


Fig.4. Bit rate comparison

As shown in Table 2, Bit rate during the time when hacker is enabled is low and after applying security mechanism there is increase in bit rate. The bit rate comparison graph is shown in fig.4.

3.1.3 Packet loss rate

It is the number of packets lost per unit of time.

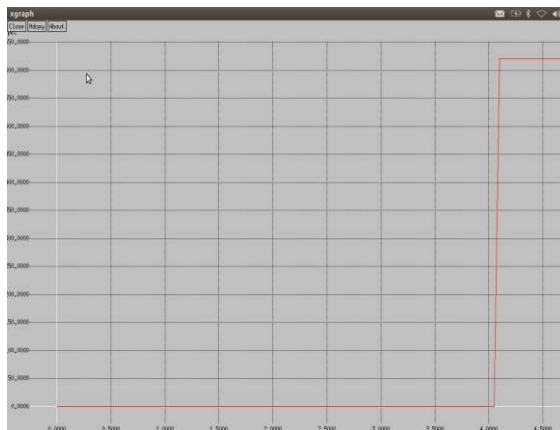


Fig.5. Packet loss rate versus Time

Table 3: Packet loss rate

Time	Packets loss rate
0	0
2 (60 sec)	0
4 (120 sec)	0
5 (150 sec)	625

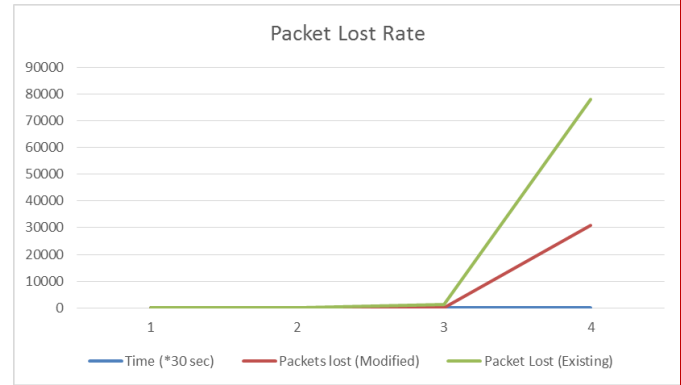


Fig.6. Packet lost rate comparison of existing RSA with modified RSA

As shown in the fig.6. , packet lost rate of modified RSA is less than the that of existing RSA. This is because of using modified algorithm security increases and hacker is unable to harm the user data.

3.1.4 Packets Delay

It is the average time taken by the data packets to reach the intended destinations. To find out the packet delay the difference of packet sent and received time was stored and then dividing the total time difference over the total number of packet received gave the average end to-end delay for the received packets. The performance is better when packet delay is low.

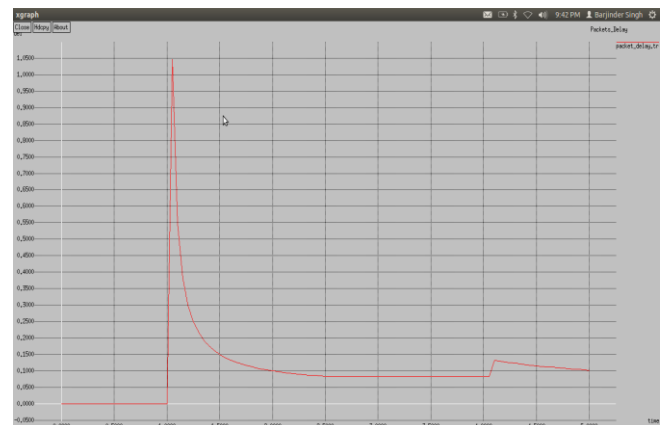


Fig.7. Packet Delay

Table 4: Packet Delay

Time	Packets delay
0	0
1	1.050
2 (60 sec)	.100

4 (120 sec)	.080
5 (150 sec)	.100

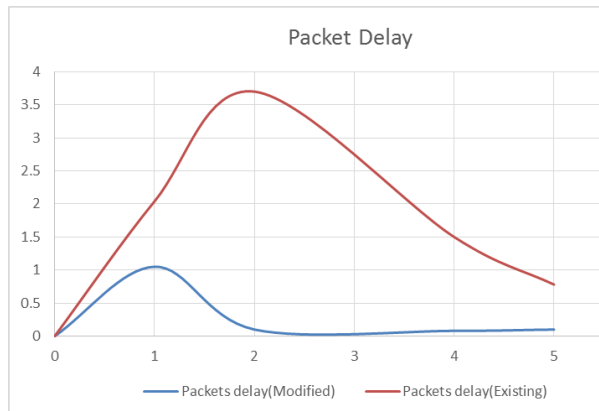


Fig.8 Packet delay Comparison

As shown in the above fig.8 Packet delay in case of modified RSA is much less than that of packet delay in existing RSA.

4. CONCLUSION

In this paper comparison of two algorithms classical RSA and Modified RSA is performed. The results have been carried out by evaluating the value of Throughput, Bit rate, Packet Loss Rate and Packet Delay. We find that modified RSA is much better than Classical RSA. The modified RSA provides more security. Proposed algorithm describes a generalized approach that works on public cryptosystem along with bit stuffing, public cryptosystem that accounts for authenticity and bit stuffing that deals with message confidentiality by digesting message from arbitrary length size to fixed length. This process overcomes the limitation of RFC 2535.

6. REFERENCES

- [1] Jagdeep Kaur, Tanupreet Singh, Barjinder Singh, "A New Security Mechanism for the Prevention of DNS Spoofing to Isolate Phishing Attacks", JCT Journals, Vol.3, Issue 12, December 2014.
- [2] Bit-Stuffing
http://en.wikipedia.org/wiki/Bit_stuffing
- [3] Cisco Systems, Introduction to Secure Sockets Layer, <http://www.ehacking.net/2011/05/secure-socketslayer-ssl-introduction.html> University, June 2003.
- [4] Fanglu Guo Jiawu Chen Tzi-cker Chiueh "Spoof Detection for Preventing DoS Attacks against DNS Servers" published in 26th IEEE International Conference on Distributed Computing Systems, , Stony Brook University, NY 11794,2006.
- [5] Gaurav, Madhuresh Mishra ,Anurag Jain, "Anti-phishing Techniques: A Review", International Journal of Engineering Research and Application, Vol.2, Issue2, April 2012.
- [6] Giuseppe Ateniese and Stefan Mangard, "A new approach to DNS security (DNSSEC)" Philadelphia, Pennsylvania, USA. Copyright 2001 ACM 1-58113-385-5/01/0011
- [7] H. Otrok, Security testing and evaluation of Cryptographic Algorithms, M.S. Thesis, Lebaneseerican, June 2003.
- [8] Mayur Bhati, Rashid Khan, "Prevention Approach of Phishing on different Websites ", International Journal of Engineering and Technology, Vol. 2, No.7, July 2012.
- [9] Maziar Janbeglou, Mazdak Zamani, Suhaimi Ibrahim "Redirecting Network Traffic toward a Fake DNS Server on a LAN", IEEE 978-1-4244-5539 3/10/\$26.00©2010
- [10] P. Ferguson and D. Senie "Network ingress filtering: Defeating denial of service attacks which employ IP source address spoofing," RFC 2827, 2000
- [11] Thawatchai Chomsiri "HTTPS Hacking Protection", 21st International Conference on Advanced Information Networking and Applications Workshops (AINAW'07) 0-7695-2847-3/07, 200
- [12] Thawatchai Chomsiri "Sniffing packets on LAN without ARP spoofing" IEEE International Conference on Convergence and Hybrid Information Technology, ,2008.
- [13] U.Steinho , A.Wiesmaier, and R.Araújo "The state of the art in DNS spoofing", Department of Cryptography and Computer algebra Technische Universität Darmstadt Hochschulstr. 10; D-64283 Darmstadt, Germany
- [14] Weider D. Yu Shruti Nargundkar Nagapriya Tiruthani, "PhishCatch - A phishing detection tool", 33rd Annual IEEE International Computer Software and Applications Conference, Computer Engineering Department, San Jose State University San Jose (Silicon Valley), California, USA 95192-0180

[16] Sushma Pradhan, Birendra Kumar Sharma, “A modified variant of RSA Algorithm for Gaussian Integers” published in IJINS, Vol.2, No.4, August 2014.

[17] H. Otok “Improving Secure Socket Layer by modifying its Authentication functions”, 2006

IJournals