A Comparative Analysis of Different Encryption Algorithms: RSA, AES, DSS for Data Security

Shikha Atwal [1], Umesh Kumar [2]

[1] M.Tech., YMCA University of Science and Technology, Faridabad, Haryana, 121006; shikhaatwal780@gmail.com

[2] Assistant Professor, YMCA University of Science and Technology, Faridabad, Haryana, 121006; umesh554@gmail.com

Abstract: With the emerging technology connected with the internet, there is one constant issue related to that is data security. The only solution with which this issue can be resolved at a limit and can be used to protect the data is various algorithms for encryption. Though different approaches were used for the same, Cryptography seems to be efficiently protecting the data while transmitting in network from sender to receiver. Firstly the data is encrypted before sending to receiver using the most secure and reliable encryption algorithm. Secondly, at the receiver end it can be decrypted using the same decryption algorithm. Only receiver will have the key with which the data can be decrypted. In this paper, AES, DSS and RSA algorithms were implemented. These algorithms are encryption algorithms which perform encoding and decoding of data, to be sent from sender to receiver, using the keys. Each have different criteria for encryption and are then compared based on different parameters viz. delay, throughput, PDR is an acronym for packet delivery ratio, PLR represents packet loss ratio and RPC denotes Received Packet Count. The results in the form of graphs are given to analyze the security provided by each algorithm.

Keywords: AES, DSS, RSA, security, cryptography, encryption, decryption

1. Introduction

With the evolution of human intelligence, information security has become more complex. There are different encryption techniques in which some are reliable based on some factors and some on other factors. It has become difficult to decide the kind of cryptographic algorithms to be used for information security [19].

Cryptography is the study of different methods used for the security of information communicated over network. In cryptography, encryption means encoding of information such that only authorized user can access it and read it. It is done at sender's end. Decryption means decoding of that encoded information that has been encrypted in a secret message. Only authorized user can access it by using a secret key through which the information has been encrypted.

Different encryption methods and algorithms are present at hand which are further used for protecting the information. There are two kinds of encryption (cipher) algorithms - Asymmetric (public) key and Symmetric (private) key encryption algorithms.

1.1. Symmetric Key Encryption

In both the processes of cryptography - encryption and decryption, a private key is used at both the ends - sender and receiver [14]. The method of key exchange has to be done before the data exchange starts [2]. Various examples of symmetric encryption algorithms are AES, DES [20]. AES uses different bit size keys whereas DES uses only one.

1.2. Asymmetric Key Encryption

This kind of algorithm solves the problem of key distribution among the users accessing data as there was only one key in symmetric encryption algorithms. In this, two keys are used- public and private. The former one is used for encrypting the data [1] which is known to everyone and only receiver has access to the latter one and is used for decryption of the encrypted data to obtain the original data. There are many examples of asymmetric encryption algorithms like RSA, Digital Signatures [20].

As discussed above, two keys are used which requires more computational calculations than symmetric encryption algorithms which only use one key. That's why asymmetric encryption algorithms are 1000 times slower than the symmetric ones.

2. Encryption Algorithms

Various types of cryptographic algorithms [4] are used for encryption and decryption of information. Encryption is a method used to protect the sensitive data transmitted over network.

The following encryption algorithms are implemented in this paper-

1. Rivest-Shamir-Adleman (RSA)
2. Advanced Encryption Standard (AES)
3. Digital Signature Standard (DSS)

1. Rivest-Shamir-Adleman (RSA)

RSA is an asymmetric cryptographic algorithm [10]. As discussed above, it uses two keys. Using public key, data is encrypted by the sender which is known to everyone but can be decrypted with the private key available to receiver only. This algorithm consists of three steps: key generation, encryption, and decryption.

Step 1: Key is generated using following steps:
a) Choose two prime numbers p and q
b) Compute the value of modulus (n) = p*q
c) Compute the value of totient, $\emptyset(n)=(p-1)(q-1)$
d) Compute 'e' (public key) such that e should be co-prime to $\emptyset(n)$ and $1<e<\emptyset(n)$
e) Compute the value of d (private key) such that $d=e^{-1} \bmod \emptyset(n)$

Step 2: For encryption:
Message is encrypted using (e, n) as the public key using $c=m^{e} \bmod n$, where m is the plaintext message to be sent from sender to receiver and c is the ciphertext [13].

Step 3: For decryption:
Message can be decrypted using d as the private key using $m=c^{d} \bmod n$.

2. Advanced Encryption Standard (AES)

AES is a symmetric key encryption algorithm which was originated by Belgian cryptographers Vincent Rijmen and Joan Daemen. In order to protect the data, AES is implemented throughout the world in the form of hardware and software [11]. This algorithm was imparted by the agency called NIST (National Institute of Standards and Technology) [3]. In this, encryption and decryption of data is done in the form of blocks divided into 128 bits each [7][12]. It can be achieved going through 10, 12, or 14 rounds depending on 128-bit, 192-bit, or 256-bit keys. AES-128 has 128-bit Key thus it is referred as AES-128, and so on the other bit Keys. AES will execute 9 processing rounds when block and key are each of 128-bits, 11 processing rounds when they are 192-bits each and it will execute 13 rounds of processing when they are 256-bits each. The process in remaining last round in all three cases are different [15].

The whole process is depicted by fig-1. AES is performed on a 4 x 4 array of bytes, referred to as state array.

There are four steps in each processing rounds:

Step 1: Key Expansion- The set of new round keys are generated from the original secret key as shown in fig-2.

Step 2: Initial round key addition-

1. AddRoundKey: Byte of the round key is combined with each byte of state array using bitwise XOR algorithm as shown in fig-3.

Step 3: For 9, 11, or 13 rounds of state modification-
1. SubBytes: It is a substitution step where each byte of resultant data is replaced using a substitution table depicted in fig-4.
2. ShiftRows: It is a transposition step in which last three rows of the state are shifted periodically a certain number of steps as shown in fig-5.
3. MixColumns: In this, a linear mixing algorithm is performed on the columns by combining the four bytes in each column as depicted in fig-6.
4. AddRoundKey

Step 4: Perform final round (10, 12, or 14)
1. SubBytes
2. ShiftRows
3. AddRoundKey

Step 5: After going through these rounds, the final output is the encrypted data or ciphertext.

3. Digital Signature Standard (DSS)

One of the way for authenticating the genuine data coming from trusted individual is signature. In order to authenticate a digital information coming from a trusted source, digital signatures are used.

DSS which can also be called as Digital Signature Standard which includes specific algorithms as per FIPS (Federal Information Processing Standard). These algorithms use SHA (Secure Hash Algorithm) which further help in generating digital signatures [18], used for the authentication of electronic documents. DSS does not use any encryption or key exchanging algorithms. It only provides us with the digital signature function.

In general, first digital signature is generated at sender side and it is verified and validated at receiver side. The whole process of DSS is shown in fig-7.

From sender side,

As discussed earlier, hash code is generated from the message and passed to the signature function with other inputs, which are-

I. Hash code,
II. Any random number 'k' generated for the signature,
III. Sender A's private key, say PR(a), and
IV. A global public key, say PU(g).

After the processing of these inputs through the function, we get the output signature generated by the signatory including two elements - 's' and 'r'. Only signatory is authorized to use the private key to generate signature. The private key must be kept secret so that other entities couldn't claim public and private key and further use the private key to generate fraud signatures. Finally, The original message combined with the signature is sent to the receiver.

At receiver end,

Here, verification of the sender and authenticity of the signature received is done first by the digital signature verifier. For that, hash code of the message sent is generated. Verification function is used for this purpose which takes following inputs-

I. The hash code generated by receiver,
II. Public key of the sender
III. Global public key, PU(g)
IV. Signature components 's' and 'r' generated at sender's side.

After processing these inputs in verification function, the output is then compared with the signature element 'r'. Signature sent can be valid only if both the values match. It is because only sender can generate a valid signature with the help of its private key.

The process of generation of signature by signatory and verification of signature and sender done by verifier is shown in fig-8.

Example for this algorithm can be taken as, first from sender side, there is a certificate authority mostly referred to as CA. He is the one responsible for signing the all different types of documents like identification papers, warrant, license, ID card, passport, and proof of qualifications which contains an owner's public key and identity. The owner's public key and identity are used to form a certificate after verifying the proof of the owner's identity. Using the generated digital signatures, above said credentials are certified and distributed thereafter. The systems which are used for this purpose are beyond the scope of this standard. There are other methods also which are used for establishing the proof of identity and those are allowed. For example-identity credentials attached with the public key can be provided directly to the potential verifier at the receiver end.

This process is used to verify at receiver side but if it fails, nothing can be deduced as to whether data received is correct or not. In order to validate the verified digital signature, the verifier must have few assurances, which are-

1. Signatory's claimed identity,
2. Validity of the public key, and
3. Assurance that the claimed signatory does actually have the private key that was previously used to generate the digital signature at the time when it was generated.

The digital signature and signed data will be considered valid if the process at receiver end, which is verification with these assurances, are successful.

On the opposite, signature and signed data will be considered invalid if this process fails. For this, the organization, according to their standards and policy, will take action on invalid digital signature.

3. Implementation

For the implementation, multiple nodes are connected through network and data is send from sender to receiver through node to node. This work was utilized and implemented in network simulator used for network research and the version in which it was implemented was ns-2.35.

Following are the cases for the implementation of three algorithms [16][2] i.e., AES algorithm, DSS algorithm and RSA algorithm.

3.1. RSA algorithm

In this, firstly user has to enter two prime numbers and then going through the steps of the algorithm, user has to enter the message to be transmitted to the destination node [17]. Then possible values of d and e are calculated. Using these values, resultant encrypted message is shown. The decryption process also works immediately after encryption showing the resultant original decrypted message as depicted through figure 13.

3.2. AES algorithm

For the implementation of this algorithm, first user has to select which key size he wants to use for encryption. If the user chooses 128-bit key, then Case 1 will be executed and if he chooses 192-bit key, the Case 2 will be executed and if 256-bit key is chosen, then Case 3 will be executed.

Case 1: 128-bit AES

Figure 9 shows implementation of 128-bit AES. In this, 128-bit key and 16-bit data is taken as input which results in 16-bit encrypted data. In order to check, same 128-bit key and 16-bit encrypted data is taken as input for decryption which results in the original 16-bit data.

Case 2: 192-bit AES

Figure 10 shows implementation of 192-bit AES. In this, 192-bit key and 16-bit data is taken as input which results in 16- bit encrypted data. In order to check, same 192-bit key and 16-bit encrypted data is taken as input for decryption which results in the original 16-bit data.

Case 3: 256-bit AES

Figure 11 shows implementation of 256-bit AES. In this, 256-bit key and 16-bit data is taken as input which results in 16- bit encrypted data. In order to check, same 256-bit key and 16-bit encrypted data is taken as input for decryption which results in the original 16-bit data.

3.3. DSS algorithm

When data is transmitted from source to destination, source node generates a session key which is then validated by all the neighboring nodes. The digital signature is first created at sender's side. If session key is validated by any adjacent node, then data is transmitted to that node till it reaches to the destination node as shown in figure 12.

4. Comparison

4.1. Parameters

1. Delay: In context of networking, delay is referred to as the propagation delay which is the total time taken by the head of signal to be carried from source towards its destination.

2. Throughput: In context of networking, it is the rate of production of messages and its successful delivery over a network. It indicates how much data or the messages are delivered from source to destination over a physical or logical link through a network node. Throughput is usually measured in bits per second (bps).

3. Packet delivery ratio (pdr): It is the ratio which depicts how many packets are delivered while transmitting information from sender to receiver.

4. Packet loss ratio (plr): It is the ratio which depicts how many packets are lost while transmitting information from sender to receiver.

5. Received packet count (rpc): It is the count which depicts how many packets are sent and how many are received while transmitting information from sender to receiver.

4.2. Graphs based on parameters

The algorithms are compared and analyzed [3] based on the above parameters.

4.2.1. Based on Delay

Figure 14 shows the graph for delay in AES algorithm. In this, for some values of x, y is constant and then it continuously increases after one point of x. After this point of time, it varies irregularly towards x-direction.

Figure 15 shows the graph for delay in DSS algorithm. In this, the graph is linearly increasing in between x and y-direction.

Figure 16 shows the graph for delay in RSA algorithm. In this, the graph is linearly increasing in between x and y-direction.

4.2.2. Based on Throughput

Figure 17 shows the graph for throughput in AES algorithm. In this, throughput increases invariantly.

Figure 18 shows the graph for throughput in DSS algorithm. In this, firstly throughput increases but after sometime, it goes towards saturation.

Figure 19 shows the graph for throughput in RSA algorithm. In this, firstly throughput increases but after sometime, it goes towards saturation.

4.2.3. Based on Packet Delivery Ratio (PDR)

Figure 20 shows the graph for PDR in AES algorithm. In this, packet delivery ratio increases in the form of step function.

Figure 21 shows the graph for PDR in DSS algorithm. In this, packet delivery ratio increases in the form of step function.

Figure 22 shows the graph for PDR in RSA algorithm. In this, packet delivery ratio increases in the form of step function.

4.2.4. Based on Packet Loss Ratio (PLR)

Figure 23 shows the graph for PLR in AES algorithm. In this, the graph increases towards x-direction constantly then it increases towards y-direction constantly and then it will be constant for some time in x-direction and again it will increase towards y-direction and then variably towards x and y-direction.

Figure 24 shows the graph for PLR in DSS algorithm. In this, the graph increases towards x-direction constantly then it increases towards y-direction constantly and then it will be constant for some time in x-direction and again it will increase towards y-direction and then variably towards x and y-direction.

Figure 25 shows the graph for PLR in RSA algorithm. In this, the graph increases towards x-direction constantly then it increases towards y-direction constantly and then it will be constant for some time in x-direction and again it will increase towards y-direction and then variably towards x and y-direction.

4.2.5. Based on Received Packet Count (RPC)

Figure 26 shows the graph for RPC in AES algorithm. In this, the graph firstly increases towards x-direction constantly then after some values of y, it increases gradually. Then the same pattern is followed till graph increases towards x and y-direction.

Figure 27 shows the graph for RPC in DSS algorithm. In this, the graph firstly increases towards x-direction constantly then after some values of y, it increases gradually. Then the same pattern is followed till graph increases towards x and y-direction.

Figure 28 shows the graph for RPC in RSA algorithm. In this, the graph firstly increases towards x-direction constantly then after some values of y, it increases gradually. Then the same pattern is followed till graph increases towards x and y-direction.

After analyzing all the graphs taking them individually with each parameter, we get the result that AES algorithm is best algorithm out of three because after integrating all algorithms in one graph with one parameter each, we get mostly a straight line which is overlapping of three algorithms.

AES encryption algorithm is an efficient and better algorithm as compared to RSA and DSS algorithm. It is simply because it provides different key lengths which are 128-bit, 192-bit, and 256-bit for encryption and decryption.

5. Conclusion and Future Scope

In order to prevent the confidential information from hackers, cryptography is used. Different cryptographic algorithms are implemented successfully and then compared to get what algorithm will provide better security. The algorithms compared and analyzed are AES (Advanced Encryption Standard), RSA (Rivest-Shamir-Adleman), and DSS (Digital Signature Standard) algorithms.

After analyzing and comparing all three, we get to know where the actual strength of the algorithm lies. It depends upon the key length in case of AES specifically. Based on that, it can be said that as the length of key increases, security of the data through algorithm also increases but vice-versa is the case with performance as the time taken by algorithm to encrypt the confidential information and then forward it to receiver side becomes more.

After critically analyzing all three algorithms through their graphs based on different parameters; it is found that there are some flaws in these algorithms. There can be different attacks on the data, for example, man-in-the-middle attack or Denial of Service

(DoS) attack etc. and in order to know how to overcome these, firstly a full comparison is done between these algorithms based on some parameters such as PDR (Packet Delivery Ratio), Propagation delay, throughput, PLR (Packet Loss Ratio), and RPC (Received Packet Count) and then we find out which one is best through the comparisons in graphs.

The best cryptographic algorithm found, among these three through the analysis, still has many complexities and flaws in it which we can further try to remove or reduce in the future. This work can be extended in the future to reducing the complexity.

Data Availability Statement: The data presented in this study are available in supplementary pdf file with this named images.pdf.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

References

1. Idrizi, Florim, Dalipi, Fisnik & Rustemi, Ejup. “Analyzing the speed of combined cryptographic algorithms with secret and public key”. International Journal of Engineering Research and Development, e-ISSN: 2278-067X, p-ISSN: 2278-800X, www.ijerd.com Volume 8, Issue 2 (August 2013), pp. 45
2. Himani Agrawal and Monisha Sharma, “Implementation and analysis of various Symmetric Cryptosystems”, Indian Journal of science and Technology Vol.3, No.12, 2012
3. Vishal R. Pancholi and Dr. Bhadresh P. Patel, “Cryptography: Comparative Studies of Different Symmetric Algorithms”, International Journal of Technology and Science, ISSN (Online) 2350-1111, (Print) 2350-1103 Volume VI, Issue I, 2015 pp. 4-7
4. Mini Malhotra and Aman Singh, ”Study of Various Cryptographic Algorithms”, International Journal of Scientific Engineering and Research, www.ijser.in, ISSN (Online): 2347-3878, Volume 1 Issue 3, November 2013
5. Atul Kahate, “Cryptography and Network Security”, Tata McGraw-Hill publishing company, New Delhi, 2008.
6. Gurjeevan Singh, Ashwani Kumar Singla, K.S.Sandha, “Through Put Analysis of Various Encryption Algorithms”, IJCST Vol.2, Issue3, September 2011.
7. William Stallings, “Cryptography and Network Security”, Pearson prentice hall, 2006, 4th edition.
8. Bruce Schneier, “Applied Cryptography, Protocols, Algorithms and Source Code in C”.
9. Arash Habibi Lashkari, Mir Mohammad Seyed Danesh, Behrang Samadi, “A Survey on Wireless Security Protocols (WEP, WPA and WPA2/802.11i)”, International Conference on Computer Design and Applications 2009
10. Saranya, Vinothini, Vasumathi, “A Study on RSA Algorithm for Cryptography”, International Journal of Computer Science and Information Technologies 2014.
11. Vedkiran Saini, Parvinder Bangar, Harjeet Singh Chauhan, “Study and Literature Survey of Advanced Encryption Algorithm for Wireless Application”, International Journal of Emerging Science and Engineering 2014
12. Hua Li and J. Li, “A New Compact Dual-Core Architecture for AES Encryption and Decryption”, IEEE Canadian Journal of Electrical and Computer Engineering, Vol. 33, No. 3, pp. 209-213, 2008.
13. H. C. Williams, “A Modification of the RSA Public-Key Encryption Procedure”, IEEE Transactions on Information Theory, Vol. 26, No. 6, pp. 726-729, 1980.
14. T. Elgamal, “A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms”, IEEE Transactions on Information Theory, Vol. 31, No. 4, pp. 469-472, 1985.
15. S. Mangard, M. Aigner and S. Dominikus, “A Highly Regular and Scalable AES Hardware Architecture”, IEEE Transactions on Computers, Vol. 52, No. 4, pp. 483-491, 2003.
16. H. W. Kim and S. Lee, “Design and Implementation of a Private and Public Key Crypto Processor and Its Application to a Security System”, IEEE Transactions on Consumer Electronics, Vol. 50, No. 1, pp. 214-224, 2004.
17. H. M. Sun, M. E. Wu, W. C. Ting, and M. J. Hinek, “Dual RSA and Its Security Analysis”, IEEE Transactions on Information Theory, Vol. 53, No. 8, pp. 2922-2933, 2007.
18. J. Ren and L. Harn, “Generalized Ring Signatures”, IEEE Transactions on Dependable and Secure Computing, Vol. 5, No. 3, pp. 155-163, 2008.
19. A. Khalique, K. Singh and S. Sood, “A Password-Authenticated Key Agreement Scheme Based on ECC Using Smart Cards”, International Journal of Computer Applications, Vol. 2, No.3, pp. 26-30, 2010.
20. S. F. Mare, M. Vladutiu and L. Prodan, “Secret data communication system using Steganography, AES and RSA”, IEEE 17th International Symposium for Design and Technology in Electronic Packaging, pp. 339-344, 2011.
21. Gurpreet Singh, Supriya, “A Study of Encryption Algorithms (RSA, DES, 3DES and AES) for Information Security”, International Journal of Computer Applications (0975 – 8887) Volume 67– No.19, April 2013
22. Dr. Prerna Mahajan, Abhishek Sachdeva, “A Study of Encryption Algorithms AES, DES and RSA for Security”. Global Journal of Computer Science and Technology, [S.l.], Dec. 2013. ISSN 0975-4172

doi:10.20944/preprints202104.0673.v1

23. E. Thambiraja, G. Ramesh, Dr. R. Umarani, "A survey on various most common Encryption Techniques", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 2, Issue 7, July 2012 ISSN: 2277 128X
24. Kyaw Myo Thu, Kyaw Swar Hlaing, Nay Aung Aung, "Performance Analysis of RSA and Elgamal Public-Key Cryptosystems", International Journal of Trend in Scientific Research and Development (IJTSRD) Volume 3 Issue 6, October 2019 e-ISSN: 2456 – 6470
25. Srinivasan Nagaraj, Dr. G.S.V.P. Raju, V.Srinadth, "Data Encryption and Authentication Using Public Key Approach", International Conference on Intelligent Computing, Communication & Convergence (ICCC-2014)
26. Alongbar Daimary, Prof. (Dr.) L. P. Saikia, "A Study of Different Data Encryption Algorithms at Security Level: A Literature Review", (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 6 (4) , 2015, 3507-3509 ISSN: 0975-9646