ISSN 2350-1022

International Journal of Recent Research in Mathematics Computer Science and Information Technology
Vol. 4, Issue 1, pp: (1-4), Month: April 2017 – September 2017, Available at: www.paperpublications.org

Improving Network Security by Modifying RSA Algorithm

KANNIKA PARAMESHWARI B[1], KRITHIKA M[2], KARTHI P[3]

[1,2] Computer Science And Engineering, Jeppiaar Engineering College, Semmanchcheri, Chennai, India
[3] Computer Science And Engineering, Rajalakshmi Engineering College, Thandalam, Chennai, India

Abstract: **Security is playing an important and crucial role in the field of network communication system and internet. Here, lot of encryption algorithms were developed and so far .Though many algorithms are used now a days, there is a lack of security in message transformation. Security can be improved by making some modifications in traditional algorithms. Algorithms are DES, RSA, ECC algorithm etc. Among this it is preferred to do some modifications in RSA Algorithm. So, the changes applied in these algorithms, security will be better than the previous.**

Keywords: **Encryption, Decryption, DES, RSA, ECC, Plain Text, Cipher Text.**

I. INTRODUCTION

The process of encoding the plain text into cipher text is called Encryption and the reverse process is called decryption. It can be done by two techniques, symmetric and asymmetric key cryptography. Symmetric key uses same public key for both encryption and decryption but the asymmetric key uses public key for encryption and private key for decryption. The algorithm comes under the symmetric is DES Algorithm and ECC ALGORITHM. The algorithm comes under asymmetric is RSA ALGORITHM. For each algorithm there are two key aspects are involved. They are algorithm type (size of plaintext should be encrypted is defined) and algorithm mode (cryptographic algorithm mode is defined).Algorithm mode is a combination of series of basic algorithm and block cipher and some feedback from above steps. So, that we are going to discuss the comparison of DES, RSA, ECC ALGORTIHMS and preferred some modifications in these algorithms.

II. DES-DATA ENCRYPTION STANDARD

The Data Encryption Standard (DES) is a symmetric-key block cipher. DES is an implementation of a Feistel Cipher. It uses 16 round Feistel cipher structure. The block size is 64-bit. Though, key length is 64-bit, DES has an effective key length of 56 bits, since 8 of the 64 bits of the key are not used by the encryption algorithm (function as check bits only).

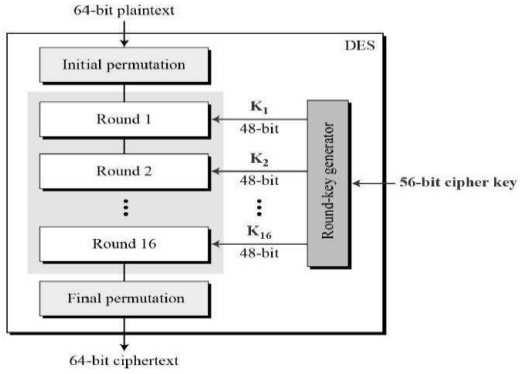

Fig: General Structure of DES

ISSN 2350-1022

International Journal of Recent Research in Mathematics Computer Science and Information Technology
Vol. 4, Issue 1, pp: (1-4), Month: April 2017 – September 2017, Available at: www.paperpublications.org

Since DES is based on the Feistel Cipher, all that is required to specify DES is

1) Round function

2) Key schedule

Any additional processing − Initial and final permutation

A. Round function:

The heart of this cipher is the DES function, *f*. The DES function applies a 48-bit key to the rightmost 32 bits to produce a 32-bit output.

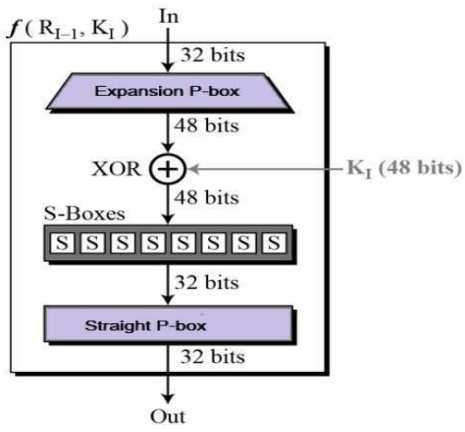

Fig: Round function.

B. Advantage:

It's better than XOR, and probably better than some crypto scheme you thought up yourself.

III. NEED

It's possible to brute-force in finite time on modern processors, so no-one uses it for anything serious anymore.

Also, some password systems secured with DES were limited to 8 characters and would silently truncate otherwise-secure passwords (match only the first 8 characters).

IV. ALGORITHM

RSA is an algorithm used by modern computers to encrypt and decrypt messages. It is an asymmetric cryptographic algorithm. Asymmetric means that there are two different keys. This is also called public key cryptography, because one of them can be given to everyone. The other key must be kept private. It is based on the fact that finding the factors of an integer is hard (the factoring problem). RSA stands for Ron Rivest, Adi Shamir and Leonard Adleman, who first publicly described it in 1978. A user of RSA creates and then publishes the product of two large prime numbers, along with an auxiliary value, as their public key. The prime factors must be kept secret. Anyone can use the public key to encrypt a message, but with currently published methods, if the public key is large enough, only someone with knowledge of the prime factors can feasibly decode the message.

A simple, worked example: Alice generates her RSA keys by selecting two primes: p=11 and q=13. The modulus n=p*q=143. The quotient of n $\phi(n)=(p-1)x(q-1)=120$. She chooses 7 for her RSA public key e and

Bob wants to send Alice an encrypted message M so he obtains her RSA public key (n,e) which in this example is (143, 7). His plain text message is just the number 9 and is encrypted into ciphertext C as follows: $M^e \bmod n = 9^7 \bmod 143 = 48 = C$

When Alice receives Bob's message she decrypts it by using her RSA private key (d, n) as follows:

$C^d \bmod n = 48^{103} \bmod 143 = 9 = M$

As discussed, the security of RSA relies on the computational difficulty of factoring large integers. As computing power increases and more efficient factoring algorithms are discovered, the ability to factor larger and larger numbers also increases. Encryption strength is directly tied to key size, and doubling key length delivers an exponential increase in strength, although it does impair performance. RSA keys are typically 1024- or 2048-bits long, but experts believe that 1024-bit keys could be broken in the near future, which is why government and industry are moving to a minimum key length of 2048-bits. Barring an unforeseen breakthrough in quantum computing, it should be many years before longer keys are required, but elliptic curve cryptography is gaining favor with many security experts as an alternative to RSA for implementing public-key cryptography. It can create faster, smaller and more efficient cryptographic keys. Much of today's hardware and software is ECC-ready and its popularity is likely to grow as it can deliver equivalent security with lower computing power and battery resource usage, making it more suitable for mobile apps than RSA. Finally, a team of researchers which included Adi Shamir, a co-inventor of RSA, has successfully determined a 4096-bit RSA key using acoustic cryptanalysis, however any encryption algorithm is vulnerable to this type of attack.

A. Merits:

1) Very fast, very simple encryption and verification.
2) Easier to implement than ECC.
3) Easier to understand.
4) Signing and decryption are similar; encryption and verification are similar.
5) Widely deployed, better industry support.

B. Demerits:

1) Very slow key generation.
2) Slow signing and decryption, which are slightly tricky to implement securely.
3) Two-part key is vulnerable to GCD attack if poorly implemented.

V. ECC

Elliptical curve cryptography (ECC) is a public key encryption technique based on elliptic curve theory that can be used to create faster, smaller, and more efficient cryptographic keys. ECC generates keys through the properties of the elliptic curve equation instead of the traditional method of generation as the product of very large prime numbers. The technology can be used in conjunction with most public key encryption methods, such as RSA, and Diffie-Hellman. According to some researchers, ECC can yield a level of security with a 164-bit key that other systems require a 1,024-bit key to achieve. Because ECC helps to establish equivalent security with lower computing power and battery resource usage, it is becoming widely used for mobile applications. ECC was developed by Certicom, a mobile e-business security provider, and was recently licensed by Hifn, a manufacturer of integrated circuitry (IC) and network security products. RSA has been developing its own version of ECC. Many manufacturers, including 3COM, Cylink, Motorola, Pitney Bowes, Siemens, TRW, and VeriFone have included support for ECC in their products.

The properties and functions of elliptic curves have been studied in mathematics for 150 years. Their use within cryptography was first proposed in 1985, (separately) by Neal Koblitz from the University of Washington, and Victor Miller at IBM. An elliptic curve is not an ellipse (oval shape), but is represented as a looping line intersecting two axes (lines on a graph used to indicate the position of a point). ECC is based on properties of a particular type of equation created from the mathematical group (a set of values for which operations can be performed on any two members of the group to produce a third member) derived from points where the line intersects the axes. Multiplying a point on the curve by a number will produce another point on the curve, but it is very difficult to find what number was used, even if you know the original point and the result. Equations based on elliptic curves have a characteristic that is very valuable for cryptography purposes: they are relatively easy to perform, and extremely difficult to reverse.

The industry still has some reservations about the use of elliptic curves. Nigel Smart, a Hewlett Packard researcher, discovered a flaw in which certain curves are extremely vulnerable. However, Philip Deck of Certicom says that, while

there are curves that are vulnerable, those implementing ECC would have to know which curves could not be used. He believes that ECC offers a unique potential as a technology that could be implemented worldwide and across all devices. According to Deck (quoted in Wired), "the only way you can achieve that is with elliptic curve."

VI. SOLUTION TO THIS PROBLEM

1. Important information can be sent by using only private key for both encryption and decryption.
2. If we reduce the key size to 32 bits, the efficiency will be more.
3. Creation of new algorithm has to be done, to control the interference of third party.
4. Encryption algorithm is always known by third party, that's why we want to create new algorithm which (i.e. the logic of algorithm) was not known to third party.
5. The third party should get the liscence from both encrypter and decrypter for knowing the secret logic.

VII. CONCLUSION

Here we analyzed the different algorithm, from that we have some ideas to make the network more secure. RSA is evolved from DES and ECC is evolved from RSA.so that we analyzed three algorithms and give ideas to create a new secured algorithm. our need for new algorithm is to the control the interference of third party.

ACKNOWLEDGEMENT

We express our deep sense of gratitude to BHUVANESWARAN B, Software Engineer of Rajalakshmi Engineering College for his valuable guidance, keen interest and encouragement throughout our work.

REFERENCES

[1] Xianmin Weiand Peng Zhang,School of Computer Engineering, Weifang University,5147 Eastern Dongfeng Street, Weifang 261061, China

[2] Sombir Singh, BRCM CET, Bahal, India.

[3] Sunil K. Maakar , Asstt. Prof. in CSE Dept, BRCM CET, Bahal, India.

[4] Dr.Sudesh Kumar, Asstt. Prof. in CSE Dept, BRCM CET, Bahal, India.