A Comparative Study on the Performance and the Security of RSA and ECC Algorithm

Dr.M.Gobi
Department of Computer Science, Chikkanna Government Arts College, Tirupur, Coimbatore
R.Sridevi
Department of Computer Science, PSG College of Arts & Science, Coimbatore-14
Email: srinashok@gmail.com
R.Rahini priyadharshini
Department of Computer Science, PSG College of Arts & Science, Coimbatore-14
Email: rahinipriyadharshini@gmail.com

----------**ABSTRACT**----------
The information security has become one of the most significant problems in data communication. Information security is the process of protecting the information and it protects its availability, privacy and integrity access to stored information on computer. Security is a broad topic and covers a multitude of sin. In its simplest form, it is concerned with making sure that nosy people cannot read, or worse yet, secretly modify messages intended for other recipients. It is concerned with people trying to access remote services that they are not authorized to use. Most security problems are intentionally caused by malicious people trying to gain some benefit, get attention, or to harm someone. In order to address this problem, cryptographic algorithms are used for the transferring the data securely between sender and the receiver. Cryptography enables all processes, transactions, and communications to be safely performed electronically. There are two types of cryptographic algorithms namely Symmetric key cryptographic algorithms and Asymmetric key cryptographic algorithms. This paper is about analyzing the performance of RSA and Elliptic curve cryptography algorithm while encrypting and decrypting the text. RSA is highly secure algorithm but have high computation time, so many researchers applied various techniques to enhance the speed of an RSA algorithm. ECC helps to establish equivalent security with lower computing power and battery resource usage.

Keywords Cryptography, Decryption, ECC, Encryption, Private key, Public key, RSA.

I. INTRODUCTION

As the technology grows day by day the need of data security over communication channel is increased to high extent, the number of threats a user is supposed to deal with grew exponentially. There are various users and organizations who want to prevent their crucial data from attackers and hackers. Also we need to ensure privacy, integrity and confidentiality about data in the network for it to be a reliable. Thus to achieve security it is very necessary to encode the data before sending it through the various unreliable communication channels available to make it unreadable. This is where the cryptography comes into picture. Cryptography is a technique to hide the data over communication channel. It allows people to communicate or transfer data electronically without worries of deceit and confidentially in addition to ensuring the integrity of the message and authenticity of the sender. For securing the knowledge cryptography is used. By using cryptography we can assist this shaky information by secrete writing on our computer network. Cryptography renders the message unintelligible to outsider by various transformations. Data Cryptography is the scrambling of the content of data like text, image, audio and video to make it unreadable or unintelligible during transmission. Its main goal is to keep the data secure from unauthorized access. Cryptography uses two algorithms symmetric key and asymmetric key cryptography. Symmetric key or secrete-key cryptograph uses the only one key for both encryption and decryption whereas asymmetric key or public key encryption uses two different keys to encryption and decryption of the message.

1. Encryption

The primary purpose of encryption is to protect the confidentiality of digital data stored on computer systems or transmitted via the Internet or other computer networks. Modern encryption algorithms play a vital role in the security assurance of IT systems and communications. Encryption provides confidentiality, authentication, integrity, non repudiation.

2. Decryption

The process of taking encoded or encrypted text or other data and converting it back into text that you or the computer can read and understand. This term could be used to describe a method of un-encrypting the data manually or with un-encrypting the data using the proper codes or keys. f this data needs to be viewable, it may require decryption. If a decryption pass code or key is not available, special software may be needed to decrypt the data using algorithms to crack the decryption and make the data readable.

II. ASYMMETRIC CRYPTOGRAPHIC ALGORITHMS

Asymmetric cryptographic algorithm uses two different keys for encryption and decryption of the message. The public key is made publicly available and can be used to

encrypt messages. The private key is kept secret and can be used to decrypt received messages. By keeping the private key safe, you can assure that the data remain safe. But the disadvantage of asymmetric algorithm is that they are computationally intensive. The below represented table show the comparable key size of both RSA and ECC algorithm. When compared RSA, the key size of ECC is too shorter. Since ECC offers security equivalent to RSA using much smaller key sizes..

Asymmetric Algorithms	**ECC**	**RSA**
80	163	1024
112	233	2240
128	283	3072
192	409	7680
256	571	15360

Fig : 1 Comparable Key Size (in bits)

1. RSA

The first, and still most commonly used asymmetric algorithm. RSA is named for the three mathematicians who developed it, Rivest, Shamir, and Adleman. RSA today is used in hundreds of software products and can be used for key exchange, digital signatures, or encryption of small blocks of data. RSA uses a variable size encryption block and a variable size key. The key pair is derived from a very large number, n, that is the product of two prime numbers chosen according to special rules. Since it was introduced in 1977, RSA has been widely used for establishing secure communication channels and for authentication the identity of service provider over insecure communication medium. In the authentication scheme, the server implements public key authentication with client by signing a unique message from the client with its private key, thus creating what is called a digital signature. The signature is then returned to the client, which verifies it using the server's known public key encryption algorithms consume a significant amount of computing resources such as CPU time, memory, and battery power. This paper examines a method for evaluation performance of various algorithms. A performance characteristic typically depends on both the encryption key and the input data.

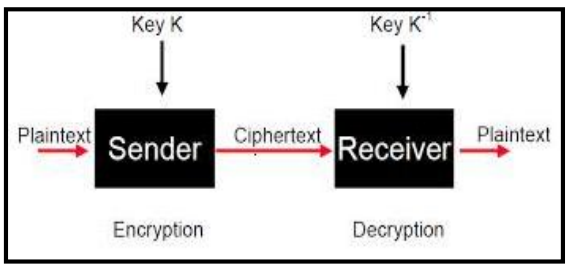

Fig 2: Encryption and Decryption Using Secure RSA

The above diagram which is representing that how the txt is encrypted and decrypted using RSA algorithm.

1.1. RSA key generation

- Randomly choose two prime numbers p and q. We choose p = 11 and q = 13.
- Compute n = pq. We compute n = pq = 11 · 13 = 143.
- Randomly choose an odd number e in the range $1 < e < \varphi(n)$ which is co prime to $\varphi(n)$ (i.e., $e \in Z^* \varphi(n)$).
- $\varphi(n) = \varphi(p) \cdot \varphi(q) = 10 \cdot 12 = 120$. Thus, we choose e = 7 ($e \in Z^*120$).
- Compute $d \equiv e{-}1 \pmod{\varphi(n)}$ by Euclid's algorithm. Thus, $de \equiv 1 \pmod{\varphi(n)}$.
- We compute $d \equiv e{-}1 \equiv 7{-}1 \equiv 103 \pmod{\varphi(143) = 120}$.
- Check that $120|7 * 103 - 1 = 721 - 1 = 720$.
- Publish (n, e) as the public key, and keep d secret as the secret key. We publish (n, e) = (143, 7) as t he public key, and keeps d = 103 secret as the secret key.

1.2. RSA encryption algorithm

Encryption Algorithm E:

- Everybody can encrypt messages m ($0 \leq m < n$ A) to user A by
- $c = EA(m) = meA \bmod nA$.
- The cipher text c ($0 \leq c < nA$) can be sent to A, and only A can decrypt.
- Encrypt m = 3:
- $EA(m) \equiv meA \equiv 37 \equiv 42 \pmod{143}$

1.3. RSA decryption algorithm

Decryption algorithm D:

- Only A knows his secrete key dA and can decrypt.
- $m = DA(c) = cdA \bmod nA$.
- Decrypt c = 42:
- $DA(c) \equiv cdA \equiv 42103 \equiv 3 \pmod{143}$
- Decrypt c = 2:
- $DA(c) \equiv cdA \equiv 2103 \equiv 63 \pmod{143}$

Advantages of RSA Algorithm are

1. They are easy to understand.
2. They are easy to implement.
3. They are easy to modify.

Disadvantages of RSA Algorithm are

1. Slower than secret key method, but can be used in conjunction with the secret key to make it more efficient.
2. Can be vulnerable to impersonation if hacked.

2. Elliptic curve cryptographic algorithm

ECC was developed by Certicom, a mobile e-business security provider, and was recently licensed by Hifn, a manufacturer of integrated circuitry and network security products. Elliptical curve cryptography is a public key encryption technique based on elliptic curve theory that can be used to create faster, smaller, and more efficient cryptographic keys. The small key size in ECC provides greater security. For faster cryptographic operations and reliability, ECC can be implemented in hardware chips. 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. 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 and to establish equivalent security with lower computing power and battery resource usage; it is becoming widely used for mobile applications. Many manufacturers, including 3COM, Cylink, Motorola, Pitney Bowes, Siemens, TRW, and VeriFone have included support for ECC in their products.

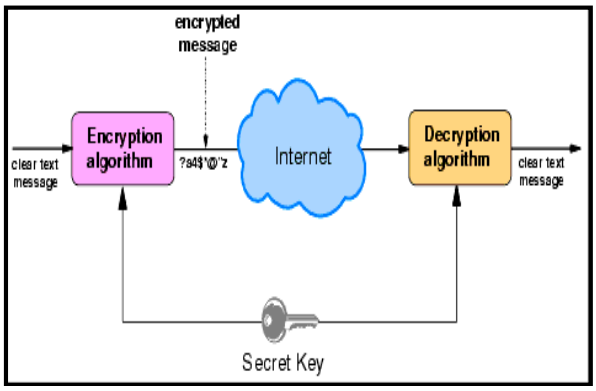

Fig 3: Encryption and decryption in ECC

The above figure represents that how the text is encrypted and decrypted in ECC with secrete key.

2.1. ECC key generation

Procedure KEY_PAIR_GEN

- Select a number‘d’ within the range of ‘n’.
- Generate the public key using the equation
- Q = d * G
- Where d is the random number that we have selected.
- Within the range of 1 to (n-1).
- G is the point on the curve.

Q is the public key and‘d’ is the private.

2.2. ECC encryption algorithm

Procedure ENCRYPT_MSG

- Consider ‘m’ has the point ‘M’ on the curve ‘E’.
- Randomly select ‘k’ in the interval 1 to (n-1).
- ‘q’ is the prime number in the interval 1 to (n-1).
- Cipher text C will be generated using the equation
- C = m Q mod q
- Cipher text C will be sent.

2.3. ECC decryption algorithm

Procedure DECRYPT_MSG

- Plaintext M will be generated using the equation
- M = Cd mod q
- Where M is the original message.
- ‘q’ is the prime number in the interval 1 to (n-1).
- ‘d’ is the private key.

Advantages of ECC Algorithm

1. This short key is faster and requires less computing power.

2. Provides greater security.

Disadvantages of RSA Algorithm

1. Increase the size of the encrypted message.

2. More complex and more difficult to implement.

3. Reduce the security of the algorithm.

III. PERFORMANCE AND THE COMPARISON OF RSA AND ECC ALGORITHM

The performance of RSA and ECC cryptographic algorithms has been analysed based on their key generation and processing time. At the 163-bit ECC/1024-bit RSA security level, an elliptic curve exponentiation for general curves over arbitrary prime fields is roughly 5 to 15 times as fast as an RSA private key operation, depending on the platform and optimizations. At the 256-bit ECC/3072-bit RSA security level the ratio has already increased to between 20 and 60, depending on optimizations. To secure a 256-bit AES key, ECC-521 can be expected to be on average 400 times faster than 15,360-bit RSA.

1. Compared key generation performance of RSA and ECC algorithm

The performance of the two algorithms does not differ until the larger key sizes, where ECC outperforms RSA Key generation for ECC out performs RSA at all key lengths, and is especially apparent as the key length increases. Since ECC does not have to devote resources to the computationally intensive generation of prime numbers, ECC can create the private/public key pair in superior speed to RSA comparable lengths. ECC key generation time grows linearly with key size, while RSA grows exponentially.

Key Length		Time (s)	
RSA	ECC	RSA	ECC
1024	163	0.16	0.08
2240	233	7.47	0.18
3072	283	9.80	0.27
7680	409	133.90	0.64
15360	571	679.06	1.44

Fig 3: Table 2: Key generation performance of RSA and ECC.

IV. CONCLUSION

Cryptographic algorithms major deals the encryption and decryption process for protecting the text files and images using some of the cryptographic algorithms like RSA,DSA, Diffie Hellman, ElGamal, ECC. In this paper a detailed performance comparison of RSA and ECC algorithms where compared based on their key size and the time taken to encrypt and decrypt the text. During this analysis it was observed that ECC was the best when compared to the RSA algorithm in terms of Authentication, based on execution time, speed, scalability, flexibility, reliability, security and Limitation that are essential for secure communication. Although the RSA algorithms were also competent but it takes more time when compared to ECC and also have a trade off between memory usage and encryption performance is better by using ECC.

V. FUTURE SCOPE

We have been witnessing a tremendous growth in the use of Internet nowadays. The world has been digitized. It is very much normal to expect a scenario in near future where almost all the work will become dependent on Internet. This would require a lot of software's. And for every software generator there are hundreds of hackers. That means, in case of an attack, it will take seconds to destroy world's largest economies through Internet. Hence we require strong algorithms to secure the networks. Our findings suggest that RSA key generation is significantly slower than ECC key generation for RSA key of sizes 1024 bits and greater. Considering there are affordable devices that can break RSA keys smaller than 1024 bits in a matter of days, the cost of key generation can be considered as a factor in the choice of public key systems to use when using digital signatures, especially for smaller devices with lesser computational resources. The difference in their key sizes grows exponentially to maintain the same relative power as compared to the average computing power available.

In fact, RSA Security on their own has admitted on their website that ECC is the technique to be in demand in the future. However, the fact remains that ECC was discovered during the process of trying to find out new ways to attack on systems using RSA techniques. That means, ECC is still in the process of being researched upon and a vast area remains unexplored. RSA, on the other hand, is well researched and trusted. That can be a hindrance. Since RSA offers better security features and with stand attacks when compared to other cryptosystems it is feasible to use ECC in Distributed sensor networks with an additional consumption of very few units of energy. In future, the research should be on the usage of ECC for wireless devices.

REFERENCES

[1]. V.B. Kute et al."A software comparison of RSA & ECC"*IJCSA Vol 2, No.1*, April/May 2009.

[2]. Zhiyong Peng; Xiaojuan Li." The improvement and simulation of software engineering and Service Science"(ICSESS),2010 *IEEE international conference on Vol,pp.500-503,16-18 July 2011*9.

[3]. Zhang; Heys, H.M.; Cheng Li; , "An Analysis of Link Layer Encryption Schemes in Wireless Sensor Networks," Communications (ICC), 2010 *IEEE International Conference on , vol., no., pp.1-6,* 23-27 May 2010.

[4]. Bahadori, M.; Mali, M.R.; Sarbishei, O.; Atarodi, M.;Sharifkhani, M.;"A novel approach for secure and fast generation of RSA public and private Smart ard, "NEWCAS Conference (NEWCAS), V*ol., no., pp.265-268,* 20-23 June 2010.

[5]. http://www.dkrypt.com/home/pkcs.

[6]. Manikandan.G, Rajendiran.P, Chakarapani.K, Krishnan.G, Sundarganesh.G, "A Modified Crypto Scheme for Enhancing Data Security", Journal of Theoretical and Advanced Information Technology, Jan 2012 *Conference on, vol., no., pp.1-6,* 23-27 May 2010.

[7]. Çakiro lu," Software implementation and performance comparison of popular block ciphers on 8- bit low-cost micro controller", *International Journal of the Physical Science Vol. 5(9), pp. 1338-1343*, 18 August, 2010.

[8]. Massey, J.L, "An Introduction to Contemporary Cryptology" Proceedings of the *IEEE, Special Section on Cryptography, 533-549*, May 2011.

[9]. M. Ciet *et al.*, "Trading Inversions for Multiplications in Elliptic Curve Cryptography" pre print,2010,

[10]. http://eprint.iacr.org/

[11]. Gururaja, H.S., Seetha, M., Koundinya, A.K. (2012), Covertness analysis of Subliminal channels in legitimate communication, P.S.Thilagam et al. (Eds.): *ADCONS 2011, LNCS 7135, pp. 582–591*, Springer, Heidelberg.

[12]. Wikipedia.orghttp://en.wikipedia.org/wiki/Elliptic_ curve.

[13]. W. Stallings," *Cryptography and Network Security*", 2006.

[14]. R. Zuccherato,"*Elliptic Curve Cryptography Support in Entrust"*, 2000.

[15]. Gururaja, H.S., Seetha, M., Koundinya, A.K. (2010), A Practical Password based authentication using elliptic curve cryptography, Proceedings of *International Conference on Convergence of Science and Engineering*, DSI Campus, Bangalore, India.