Comparison And Analysis of Asymmetric Key Algorithm

Dr.V. S. Anita Sofia[1], J. Godson[2]
[1]Associate Professor, Dept of MCA
[2]Dept of MCA
[1, 2] PSG College of Arts and Science, Coimbatore, Tamil Nadu, India

Abstract- *To prevent illegal access, data is encrypted using the encryption process. Cybersecurity, which ensures the confidentiality and integrity of data transmission over the internet and provides protection against hostile attacks, is in high demand right now. Technology is advancing every day. Faster and more advanced technology cannot exist without information security. At the execution levels, data authentication is required for this. Cryptography is a useful tool for creating safe data independence. It uses the two fundamental processes of encryption and decryption for safe data exchange. Up till now, several different cryptographic techniques have been proposed and employed. We have covered a few of the suggested Asymmetric key cryptography approaches in this post and performed a quick comparison between them. The basic traits, advantages, drawbacks, and applications of several Asymmetric key cryptography techniques are covered in this work.*

Keywords- Cryptography, Encryption, Decryption, RSA, ECC, DSA, EL Gamal, Diffie-Hellman

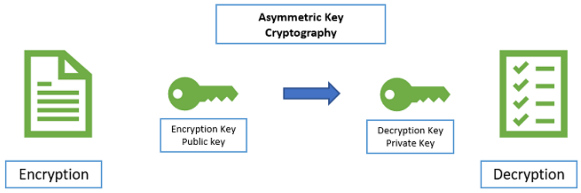
I. INTRODUCTION

The confidentiality and integrity of sensitive information have evolved into top issues for people, corporations, and governments alike in today's digitally connected world. The field of cryptography has made incredible strides in response to the ever expanding demand for secure communication and data security. Asymmetric key algorithms have become a pillar technology in this field, providing strong encryption and authentication procedures that support secure digital communication.

Asymmetric key algorithms, often known as public-key cryptography, are a change from the more conventional symmetric encryption techniques. Asymmetric cryptography uses two different keys: a public key for encryption and a private key for decryption, as opposed to symmetric cryptography, which uses a single key for both encryption and decryption. The secure data landscape has undergone a revolution thanks to our dual-key strategy.This study examines current developments in asymmetric key algorithms with an emphasis on their improved security capabilities and developing applications. We aim to present a thorough overview of the current state of asymmetric cryptography and its applicability in the current cyber-centric environment by exploring the most recent advancements in the area. In addition, we will evaluate critically the difficulties and promising paths for asymmetric key algorithm research.

A. ASYMMETRIC KEY CRYPTOGRAPHY:

Public Key Infrastructure (PKI), a cryptographic technique requiring two separate keys—one to lock or encrypt the plaintext and another to unlock or decrypt the cyphertext—is built on asymmetric keys. No key can perform both tasks. A private key is kept private, while a public key is made public. The technology enables private communication from the general public to the owner of the unlocking key if the lock/encryption key is the one that has been released. The system functions as a signature verification for documents locked by the owner of the private key if the unlock/decryption key is the one that has been released. Another name for this system is asymmetric key cryptography.


Asymmetric Key Cryptography
Encryption Key Public key
Decryption Key Private Key
Encryption
Decryption

II. ASYMMETRIC KEY ALGORITHMS

In this paper, we are discussing the asymmetric key algorithms used for both encryption and decryption. They are

- RSA
- Diffie-Hellman
- ECC
- El Gamal
- DSA

A. RSA ALGORITHM:

In 1978, Ron Rivest, Adi Shamir, and Leonard Adleman developed RSA. It is a well-known public key encoding method for key exchange, digital signatures, and database encryption. The RSA algorithm uses variable-size keys and blocks of encoding data. It uses numeral synthesis and is an asymmetric encoding scheme. It generates the public and private keys using two fundamental numbers. Using the recipient's public key, the sender encrypts the message before delivering it to the recipient. In order to decrypt it, he first uses his unique private key [1]. Key generation, encoding, and decoding are the three processes of RSA. RSA, on the other hand, has a lot of flaws, which is why it is bad for business[2].

In cryptography, RSA (Rivest-Shamir-Adleman) is a popular asymmetric encryption technique. It makes use of digital signatures as well as a pair of keys, a public key for encryption and a private key for decryption. Large composite numbers are challenging to factor into their prime factors, which is the foundation of RSA security.ensuring integrity and confidentiality in secure communications.

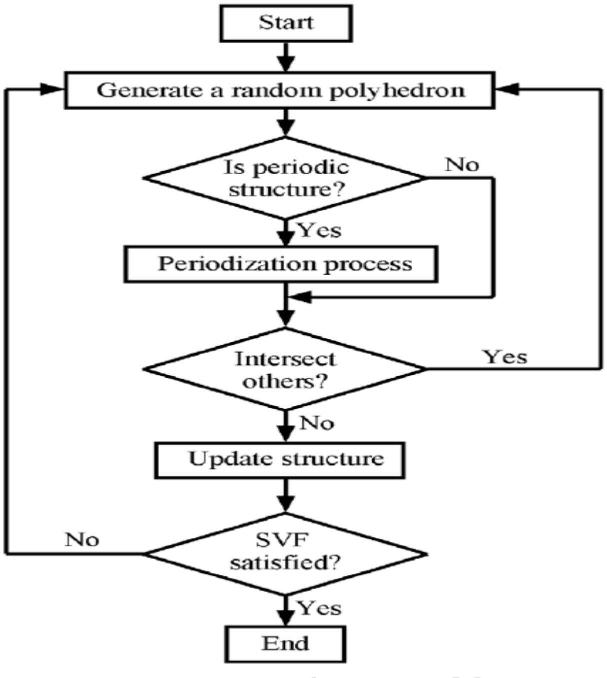


Figure 2.1 RSA Flow Diagram[3]

B. DIFFIE-HELLMAN ALGORITHM:

Diffie-Hellman developed this algorithm in 1976. Each group creates a key pair and disperses the public key in this algorithm. The Diffie-Hellman algorithm enables two users to connect across an unsecured communication channel by locating a common secret key. The fact that this method is used for communication, however, makes it vulnerable to violation during an attack and is one of its key drawbacks [4].

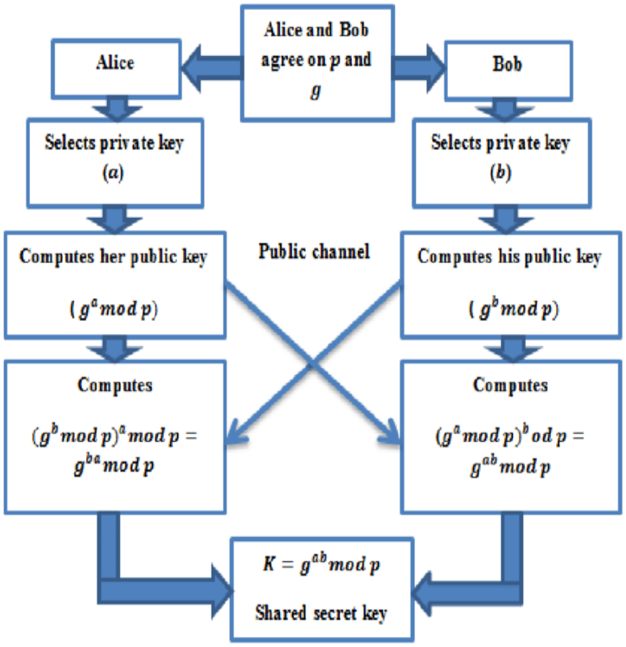

Alice and Bob agree on p and g
Alice
Bob
Selects private key (a)
Selects private key (b)
Computes her public key ($g^a mod\, p$)
Public channel
Computes his public key ($g^b mod\, p$)
Computes $(g^b mod\, p)^a mod\, p = g^{ba} mod\, p$
Computes $(g^a mod\, p)^b od\, p = g^{ab} mod\, p$
$K = g^{ab} mod\, p$
Shared secret key

Figure 2.2 Diffie-Hellman Flow Diagram[5]

Bob and Alice settle on the generator q and prime number p.
They each select a secret key (for Alice it's a, for Bob it's b).
They use the formulas a* = (q^a) mod p and b* = (q^b) mod p to calculate and share their public keys (a* and b*).
Bob and Alice settle on the generator q and prime number p.
They each select a secret key (for Alice it's a, for Bob it's b).
They use the formulas a* = (q^a) mod p and b* = (q^b) mod p to calculate and share their public keys (a* and b*).
Both can calculate the shared secret (x) independently using either x = (a*^b) mod p or x = (b*^a) mod p.
They communicate securely using the shared secret x.
The security is based on how challenging it is to determine x from the public keys a* and b*, which are sent through an unsafe channel.

C. ECC ALGORITHM:

An asymmetric algorithm called elliptic curve cryptography uses different keys for encoding and decoding. In 1985, V. Miller from IBM and N. Koblitz from the University of Washington created it. On the algebraic structures of left-shaped carvings in constrained domains, ECC was built. It works well enough to provide security with a 164-bit key. For security, that system needs a 1024-bit key. ECC offers the highest level of security while using the same bit widths. Given that it uses less energy, it is also advantageous for battery backup [6]. The fundamental benefit of ECC is that it uses short keys, which allows for speedy encoding and low energy consumption. Contrarily, one of its drawbacks is that it increases the size of the ciphered text and requires a lot of resources.

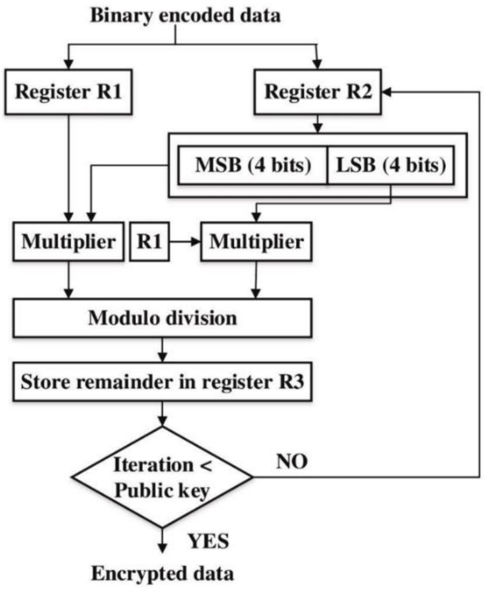


Figure 2.3 ECC Flow Diagram[7]

D. EL GAMAL ALGORITHM:

The El Gamal encryption system is a public-key cryptography asymmetric key encryption scheme that is based on the Diffie-Hellman key exchange. TaherEl Gamal first described it in 1985.[8] Recent iterations of PGP, the free GNU Privacy Guard programme, and other cryptosystems all employ El Gamal encryption. El Gamal is not to be confused with El Gamal encryption; the Digital Signature Algorithm (DSA) is a variation of the El Gamal signature system.

El Gamal encryption is a cyclic group G that can be defined over any multiplicative group of integers modulo nThe security depends on the difficulty of a particular discrete logarithm computing issue in G.

The El Gamal Encryption System depends on the difficulty of a certain algorithm problem, in which it is simple to increase the number of powerful entities. The distinctive logarithm's opposite computation is more challenging to do, though. El Gamal Encryption is based on particular parameters that affect the algorithm's flow, speed, and security. One of the many encoding schemes that employs the Adhoc system in the encoding process is this one.

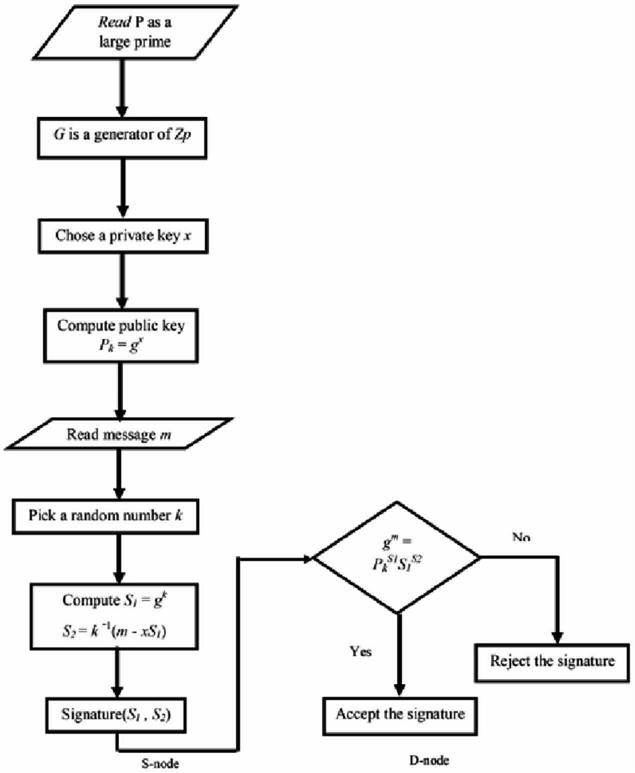


Figure 2.4 Diffie-Hellman Flow Diagram[9]

El Gamal is a digital signature and asymmetric encryption algorithm. A public key is used for encryption, whereas a private key is used for decryption or signature. It is secure when used with suitably large key sizes because it is based on the mathematical difficulty of solving the discrete logarithm problem in a limited field. El Gamal is used to safeguard information transfer and confirm the validity of digital signatures, ensuring secrecy and integrity in secure interactions.El Gamal's application in secure data transfer and digital signatures, which offer message integrity and confidentiality, are two of its standout advantages. When using El Gamal, the sender encrypts the message using the recipient's public key. To ensure that only they may access the authentic material, the recipient uses their private key to decrypt the ciphertext. El Gamal is adaptable enough to support digital signatures, in which the sender signs a message with their private key and the recipient uses the sender's public key to confirm the signature's validity.

E. DSA ALGORITHM:

A public key encoding system called a digital signature algorithm (DSA) was developed to protect the confidentiality of numerical text.NIST laid the groundwork for the DSA. A secret key is used to create a signature for a text, and a public key is used to compare the signature to the text. Similarly, any group can verify the authenticity of the signatures, but only the party in possession of the secret key is able to sign the messages. A recipient has reason to believe that the message was created by a recognised sender who possesses the secret key and that it was not modulated during transit if the message's numeric signature is readily available [10].

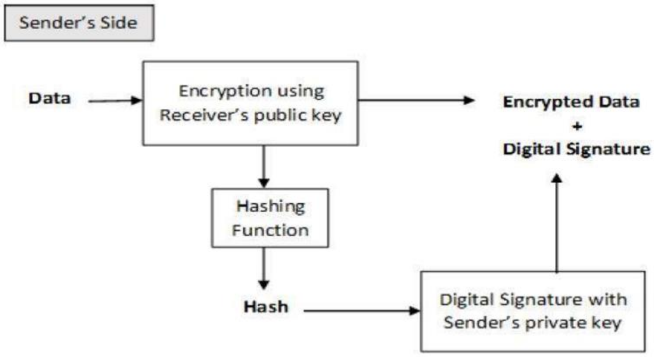

Sender's Side
Data
Encryption using Receiver's public key
Encrypted Data
+
Digital Signature
Hashing Function
Hash
Digital Signature with Sender's private key

Figure 2.5 Diffie-Hellman Flow Diagram

DSA is best known for its application in digital signatures, in which the sender signs a message or file with a private key and the recipient verifies the signature's validity with the associated public key. This technique has broad applications in email authentication, data transfer security, and other situations where it's crucial to confirm the legitimacy of digital content. DSA relies heavily on the choice of key length, with longer keys—typically ranging from 1024 to 3072 bits—providing more security but demanding more processing power. DSA is a useful technique for assuring the reliability of digital information because of how well it can provide secure digital signatures without the requirement for encryption.

El Gamal's security depends on the key size selection, just like with other cryptographic methods, with bigger keys providing more security.

III.COMPARISION OF ASYMMETRIC KEYALGORITHM

CRYPTOGRAPHY ALGORITHM	KEY SIZE (in bits)	STRENGTH	WEAKNESS	POSSIBLE ATTACKS
RSA	• 1024(can be breakable in near future) • 2048 • 3072 • 4096	• Less computation time	• Small encryption exponent and small message. • Same key for encryption and signing. • Using a common modulus for different users.	• Adaptive chosen cipher text attack. • Side-channel analysis attack. • Power fault attack.
DFFIE-HELLMAN	• 1024 or 3072 for p (the modulus)	• Solves challenging discrete logarithm. • Creating and sharing key, no information.	• Expensive exponential operation. • Lack of authentication.	• Man in the middle attack.
ECC	• Smaller key sizes, i.e. • 160 • 224 • 256	• Smaller key size. • Reduce storage • Reduce transmission time	• Increases the size of encrypted text. • Dependent on extremely intricate equations, which makes algorithms more complex.	• Side-channel attacks • Backdoors • Quantum computing attacks
EL-GAMAL	• 1024 • 2048	• based on the hardness of solving discrete logarithms.	• Cryptosystem for Enciphering Large Messages	• Brute force attack. • Discrete logarithm problem
DSA	• Multiple of 64 between 512 and 1024 (inclusive)	• Authentication • Data Integrity • Non-repudiation	• Entropy,Secrecy,and Uniqueness of the random signature valueare critical.	• Key-recovery attack • Lattice attacks. • Side-channel attacks.

IV. CONCLUSION

Modern cryptography relies heavily on asymmetric key algorithms, which provide a strong framework for secure communication, data security, and digital authentication. The complicated workings of these algorithms have been thoroughly examined in this study paper, which also emphasises their importance in preserving the privacy, integrity, and validity of digital transactions and information exchange.

It is clear from an examination of well-known asymmetric algorithms like RSA, El Gamal, and DSA that they are based on intricate mathematical ideas and take advantage of the computational complexity of issues like factorization and discrete logarithms. These algorithms support the security of our digital world by enabling safe data transmission, email encryption, digital signatures, and more.Asymmetric key cryptography's difficulties and potential change along with technology. A growing threat from quantum computing is felt by many.

REFERENCES

[1] Zhou, X., & Tang, X.: Research and Implementation of RSA Algorithm for Encryption and Decryption. In

Strategic Technology (IFOST), 2011 6thInternational Forum on IEEE, Vol. 2, 1118-1121, (August 2011).

[2] Somani, U., Lakhani, K., &Mundra, M.: Implementing Digital Signature with RSA Encryption Algorithm to Enhance the Data Security of Cloud in Cloud Computing. In Parallel Distributed and Grid Computing (PDGC), 2010 1st International Conference on IEEE, 211-216, (October 2010).

[3] https://www.researchgate.net/figure/Flow-chart-of-RSA-algorithm_fig3_260801145

[4] Omar G. A., Elsadd, M. A., &Guirguis, S. K.: Investigation of Cryptography Algorithms used for Security and Privacy Protection in Smart Grid. In Power Systems Conference (MEPCON), 2017

[5] https://www.researchgate.net/figure/Block-diagram-of-the-Diffie-Hellman-algorithm_fig1_349609600

[6] Singh, G.: A Study of Encryption Algorithms (RSA, DES, 3DES and AES) for Information Security. International Journal of Computer Applications, Vol. 67, No. 19, (2013).

[7] https://www.researchgate.net/figure/Flow-diagram-of-optimized-ECC_fig1_322803036

[8] TaherEl Gamal (1985). "A Public-Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms"

[9] https://www.researchgate.net/figure/A-graphical-representation-of-El Gamal-digital-signature-scheme_fig1_308271411

[10] Sridevi, C.: A Survey on Network Security. Global Journal of Computer Science and Technology (2018)