International Journal of Innovative Research in Computer Science and Technology (IJIRCST)
ISSN (Online): 2347-5552, Volume-14, Issue-2, March 2026
DOI: https://doi.org/10.55524/ijircst.2026.14.2.6
Article ID IRP-1731, Pages 45-51
www.ijircst.org

Comparative Analysis of Symmetric and Asymmetric Cryptographic Methods for Resource-Limited Devices

Veeresha M M [1], and Vinay S [2]

[1] Research Scholar, Department of Studies in Computer Science, Davanagere University, Davanagere, India
[2] Assistant Professor, Department of Studies in Computer Science, Davanagere University, Davangere, Karnataka, India

Correspondence should be addressed to Veeresha M veereshamm10@gmail.com

Received: 16 February 2026; Revised: 4 March 2026; Accepted: 17 March 2026

ABSTRACT- Currently, the continuous growth of resource-constrained devices used for protecting sensitive information poses a significant challenge. Now a day cryptography plays a vital role. It helps to maintain data privacy, confidentiality, integrity, and authenticity, and also protects data during transmission. In this study, we performed a comparative analysis of Symmetric and Asymmetric algorithms. In particular, the goal is to determine which method is suitable for resource-constrained devices such as Internet of Things(IOT) mobile phones and embedded devices. This study clearly explained symmetric algorithms such as the Advanced Encryption standard(AES), chacha20, and Blowfish and also Asymmetric algorithms such as Rivest–Shamir Adlemen(RSA), Diffie-Hellman, ElGamal, and elliptic curve cryptography(ECC). These algorithms are explained based on computational efficiency, Memory Usage, security length, and suitability for devices with limited resources. It concludes that symmetric cryptographic methods have a high speed and low computational overhead and are therefore suitable for encrypting large volumes of data. On the other hand, asymmetric methods offer a strong platform for ensuring the security of data during transmission. It concludes that the use of both symmetric and asymmetric cryptographic methods can efficiently enable secure communication in modern and resource-constrained computing environments.

KEYWORDS- Cryptographic Algorithms, Performance and Security Analysis, Symmetric and Asymmetric Encryption, Secure Data Transmission.

I. INTRODUCTION

Protecting data plays an important role in today's world. We need to keep our information from unthorized persons. Cryptography is the primary method used for this purpose, as it helps to keep data in secret and ensure that it is not changed or stolen during transmission over the Internet.
There are different types of cryptographic algorithms. Few methods have been used. In, a symmetric algorithm was proposed. In the symmetric algorithm, the same key is used for both encryption and decryption. Its working speed is very high, and it is used for large data. The Advanced Encryption Stadadard(AES), Data Encryption Standard(DES), and Blowfish are common examples. it helps to protect communications, databases, and files. However, in this algorithm, one major challenge is secret key sharing between users. To solve this problem, Asymmetric Encryption is used in this method using two keys: a public key (which can be shared openly) and a private key (which is kept secret by the owner). Asymmetric algorithms, such as RSA and elliptic curve cryptography (ECC), are widely used for secure communication, digital signatures, and authentication. Asymmetric encryption is more secure. It also needs more computer power.This study examines asymmetric encryption methods to determine which one is better. We want to know how well they work, how secure they are, and which one is best for situations. The goal of this study is to determine what is good and bad about each method, and to see how well they can keep our data safe when we communicate online. We discuss asymmetric cryptography algorithms and how they help protect data. Asymmetric and symmetric cryptography methods play a crucial role for safe data in Resource-Limited Devices.

II. CLASSIFICATION OF CRYPTGRAPHIC ALGORITHMS

Cryptographic algorithms are broadly classified into symmetric key algorithms, asymmetric key algorithms.

A. Symmetric Key Cryptography

In symmetric cryptography algorithms, the same key is used for encryption and decryption. These algorithms are divided into stream and block ciphers. In this algorithm, there is a very high processing speed and low computational overhead. They are well suited for encrypt the large data.

1. Stream ciphers

Streaming ciphers encrypt data bit-by-bit or byte-by-byte using a generated continuous sequence of keys. This sequence is typically generated by a PRNG (pseudo-random number) generator supplied with a secret key. These ciphers are important in situations in which an immediate cipher is required, such as in live video streaming or voice communication. The following algorithms are popular stream ciphers [1].

RC4: RC4 is a symmetric stream cipher that generates a pseudo-layer. In this keystream, cipher text is created by combining plain text and the XOR operation. RC4 is very

popular in one upon time. It has vulnerabilities. Its initial keystream output makes it unsuitable for modern cryptographic applications. RC4 is divided into two parts : Key setup Algorithm(KSA) which initializes the permutation of all byte values using keys of variable length, and the pseudorandom generation algorithm, which constructs a keystream from the permutation. Despite its shortcomings, RC4 has been adapted into innovative methods, such as pixel-shuffling techniques paired with steganography, which have proven effective in securely transmitting sensitive biomedical images while preserving the accuracy of medical diagnostics [2] [3].

In their research titled “Measuring Avalanche Properties on RC4 Stream Cipher Variants,” Madarro-Capo and colleagues examined the avalanche effect, a vital cryptographic feature where a slight change in the input leads to a significant and unpredictable alteration in the output, across various RC4 variants. The team assessed how these changes improved diffusion within the keystream. The findings indicated that while the original RC4 algorithm does not generate a strong avalanche effect, the modified versions show enhanced performance, rendering them more appropriate for secure cryptographic applications [4].

"Vulnerability Assessment of RC4 Cipher using LSTM Networks," Hammami delved deeper into the weaknesses of the RC4 algorithm. He utilized deep learning methods to evaluate the algorithm's deficiencies. By employing long short-term memory (LSTM)-based recurrent neural networks to simulate the RC4 keystream generator, the research achieved an accuracy of up to 85% in forecasting the subsequent byte of the keystream. Consequently, the predictability of RC4 output bytes was notably high. These results illustrate that machine learning can enhance cryptanalytic methods. The study provides a detailed description of the experimental design, performance evaluation metrics, architecture, and other relevant aspects. Finally, the research concludes that encryption techniques need to be strengthened to protect itself from AI-driven threats. This highlights the need to develop quantum-secure ciphers for strong data protection in this era of increasing machine learning. [5].

Salsa20: Salsa20 is the best performance and security . Its developed by Daniel J. Bernstein in 2005. It utilizes a one-time pad to encrypt almost any electronic file by generating a keystream of random numbers. Block cipher processed fixed size blocks. Unlike block ciphers that process data in fixed-size blocks, Salsa20 is a stream cipher that encrypts data sequentially. This characteristic is suitable for secure communications, and it is used in mobile applications, resource-constrained Internet of Things (IoT) environments. A recent study by Dey et al. (2025) indicated that the security of reduced-round versions, Salsa20/8 and Salsa20/8.5, is not as strong as previously thought, as adaptive keys and inputs can lead to successful cryptanalysis. Nevertheless, the full 20-round version of Salsa20 remains highly resistant to attacks and is a reliable, lightweight choice[6].

Chacha20: Chacha20 is a type of encryption algorithm developed by Berntstein et al. in 2008. This helps secure data from unauthorized persons. These people find it difficult to read this. In this method a 256 bit secrete key, 96 bit nonce(unique key) and 32 bit key. These combinations generated a random key.

Encryption data generated by the combination of keystream and original text. ChaCha20 is a famous algorithm because it works fast and secure. This method uses simple mathematical operations (e.g., addition and bit rotation) over 20 rounds. It does not require padding and is protected from timing attacks, thereby making it reliable. For these reasons, Chacha20 is widely used in security protocols such as TL1.3,SSH, and WireGuard. This ensue secure communication Later, an advanced version called ChaCha20-Poly1305 was introduced, which provides authentication along with encryption. This version protects against the wrong key or nonce use without slowing down[7]. Another study by Degabriele et al. examined how ChaCha20-Poly1305 works in systems with many users. They found that it remains secure and can be better than AES-GCM, especially when randomness is low or with large data[6]. Radhakrishnan et al. noted that ChaCha20 is secure and works well in Internet of Things (IoT) settings; however, it requires more memory than lighter ciphers such as SPECK and ASCON, which are better for very low-power devices[8].

2. Block Ciphers

The plaintext is partitioned into blocks using block ciphers, and each block has the same size as the others. Subsequently, each block of the plaintext is encrypted. This can be performed independently for each block. A method that links the blocks together can also be used. The resulting ciphertext blocks the same size as the plaintext.

Block ciphers use modes of operation to process plaintexts of varying lengths and enhance security. These methods are known as modes. The modes of operation include the electronic codebook, cipher block chaining, and counter modes. Different modes specify the method by which a block cipher works on the output of each block and how the input of one block can affect the input of the previous blocks. Block ciphers and their modes are important for ensuring security. To ensure the security of the plaintext, the counter, cipher block chaining, and electronic codebook modes work together.

Several block cipher algorithms are commonly used. These include the Advanced Encryption Standard (AES), Data Encryption Standard (DES), and blowfish. This varies depending on how fast work is, which key is used, and how secure[1]. In 2023, Xiang et al. aimed to improve the system performance of small devices. For example, STM32 Cortex-M4 has limited memory(64 KB RAM). Uses a simple ferromagnetic guidance system without complex controls or sensors. Therefore, it can be utilized in embedded applications such as small maglev trains, small autonomous vehicles, and transportation systems based on IoT, which have limited power and hardware[9].

DES & 3DES: DES is a symmetric key block cipher and is now considered an obsolete algorithm because it has only 56 bits. A small key is used to easily break the key using the brute force technique. DES has been secured, but it is currently not safe and not suitable for use. DES encrypts data in 64-bit blocks using a 64-bit key, although only 56 bits are employed for encryption, with the remaining bits designated for parity [1].

To address the security limitations of DES, a more secure variant known as triple DES (3DES) was developed. The 3DES encryption process entails three iterations using two or three distinct secret keys to enhance security. Despite providing improved security compared to standard DES, 3DES is significantly slower and less efficient than modern encryption algorithms due to its computational requirements[10].

Blowish and Twofish: The Blowfish encryption algorithm is a faster alternative to data encryption standard (DES). It encrypts data through 16 rounds using a Feistel network, which involves both permutation and substitution, depending on the key and data. The process primarily uses XOR operations and the addition of 32-bit words. Each round incorporated four table lookups from a predetermined substitution box (S-box). The 64-bit input block is divided into two 32-bit halves, denoted as xL and xR, which are processed alternately in encryption rounds [1].

A research paper on the research Gate in January 2025 explained how well three encryption methods work: blowfish, AES, and 3DES. The researchers tested these methods based on the types and sizes of the files. They found that blowfish was the fastest algorithm. Moreover, the researchers concluded that Blowfish was the fastest among the three. The offered algorithm offered better speed and throughput than those of AES and 3DES during the encryption/decryption of different plaintext types, such as text, audio, and video. Blowfish working is the fastest algorithm but utilizes more memory. AES allows for a good balance between security, speed, and memory use. 3DES is the slowest and uses at least memory, which is helpful to older systems. The best encryption method depends on what is the application needs, like speed, security, or memory efficiency[11].

RC5 and RC6: In 1994, Rivest developed RC5, which is a simple and fast method. and operates both hardware and software. R5 is the key feature in improving security by rotating data in a certain pattern. RC5 can be changed by different block sizes, key lengths, and rounds, and is useful for mobile phones, which have the best efficiency. Later, RC6 was developed, and Faragallah et al. made an improved version of RC6. This version is faster, with encryption and decryption speeds that are 70% and 64% faster than the original, while maintaining the same security level. The improved RC6 is useful for devices with limited resources, such as smartphones and Internet of Things (IoT) devices[12].

Advanced Encryption Standard (AES): AES is a reliable method to maintain data safety. It is used for secure websites, as it helps to store files, and in disk encryption systems such as BitLocker. AES encrypts data in 128-bit pieces and can use keys of 128,192 or 256 bits, making it hard to break. Later, AES-RVs were made for RISC-V processors, which improved the AES performance. This is up to 453 time more energy efficient and 256 times faster, making it best for low power devices, such as those in the Internet of Things(IOT))[13].

B. Asymmetric Key Cryptography

Asymmetric key algorithms, such as RSA, Diffie-Hellman, ElGamal, and elliptic curve cryptography, use two keys: a public key and a private key. These algorithms are mainly used to exchange keys and create digital signatures. They are effective for these tasks because they use key pairs to ensure security.

Diffie–Hellman key exchange (DH) : Diffi-hellman is the method used key exchange between two peoples over a internet. even if the others were listening. This method working based on a difficult mathematical problem called discrete logarithm problem, it is difficulty to solve so that attacker cannot figureout the key. it was developed in 1970, In modern form ephemeral Diffie-Hellman(DHE), provides forward security, meaning that even if long-term keys are compromised later, previous messages remain secure.

Another version of DH is Elliptic curve Diffi-Hellman(ECDH), which works more efficiently while consuming less power. Therefore, it is suitable for mobile devices and IoT systems. According to Xu(2025), DH and ECDH both play a crucial role in security protocols such as TLS/SSL. which protect internet communication. [14].

Rivest Shamir Adlemen (RSA): RSA, introduced in 1978, is a well-known method for keeping communication and digital signatures safe because break in requires factoring large numbers, which is difficult for normal computers. RSA has been trusted for many years; however, it has some drawbacks. It requires significant computing power and has a complicated key management system, making it less efficient than newer methods, such as elliptic curve cryptography (ECC). Recent study, in 2023, In this a study Explained that RSA becomes slower with large keys. With a 2048-bit key, the RSA can perform approximately 1500 signatures per second and check up to 34000 signatures per second. However, the 4096-bit key can only perform 150signatures per second. A similar slowdown occurs in Intel Xeon Phi Hardware, and decryption also slows down with larger keys, causing delays. Currently, RSA still uses many computers and phones; however, it is not as fast as ECC. ECC is faster and uses less power on devices such as smartphones and Internet of Things(IOT) devices. RSA is still used on many computers and phones; however, it is not as fast as ECC. ECC is quicker and uses less power on devices such as smartphones and Internet of Things (IoT) gadgets [15].

ElGamal : ElGamal is a method used to keep information private on computers. It has two main functions: encrypting messages and creating digital signatures. It is good at keeping things secret because it uses a difficult mathematical problem called the discrete logarithm. A special feature of ElGamal is that it uses random numbers to make each signature different and difficult to copy. Other versions, such as Schnorr or DSA, make it work faster and verify whether signatures are real. ElGamal is popular because it is very secure. However, it usually generates larger signatures and requires more computer power than other methods, such as RSA or ECDSA [14].

In 2021, Tea et al. introduced a security-mediated ElGamal approach. This innovative method employs a trusted third party, known as a security mediator, to aid in message decryption. The security mediator enhances the system by refining key management and access control, while safeguarding against chosen ciphertext attacks. A notable benefit of this approach is its avoidance of complex pairing operations, resulting in a simpler and faster system than many other secure systems. The security-mediated ElGamal variant offers a lightweight and efficient alternative to the traditional ElGamal variant. Although it incurs slightly

higher computational costs than standard mediated RSA, it exhibits superior scalability in high-demand server environments. The authors have also suggested potential improvements, such as integrating escrow-free mechanisms and certificateless architectures, to further enhance its applicability in modern secure communication systems [16].

Elliptic Curve Cryptography (ECC): ECC is a method used for keeping data secure by using elliptic curves. ECC protects data with smaller keys(uses less memory and power) than other methods, Such as RSA. It is very useful for resource-constrained devices such as mobile phones and Internet of Things (IoT) devices. The ECC algorithm uses much less power and memory. ECC is used in many areas, such as exchanging keys, creating digital signatures such as ECDSA, and maintaining private communications. It is becoming more common in online services and mobile identity solutions [14].

In 2024, Xiao et al. developed a system for smart transportation and small devices using elliptic curve creyptography(ECC). their study shows that Ecc is fastest and secure. also concluded that ECC can encrypt and decrypt data within 15 ms, even on a regular laptop. The system checks who is communicating, keeps talking private, and protects information from being overheard later. This system can be easily integrated into current setups. So, ECC is a good and safe way for real-time communication on small devices[17].

III. COMPARATIVE ANALYSIS

Table 1 lists various Symmetric and Asymmetric cryptography algorithms. And a comparison algorithm based on security, speed, memory usage, and optimal usage. AES is a popular symmetric algorithm. it supports multiple key sizes(128,192,256 bits), but it can be difficult to set up. ChaCha20 works well in software and is suitable for mobile devices, VPNs, and real-time communication. However, it can be risky to reuse the keys or starting numbers. Blowfish encrypts very quickly and can use different key sizes. However, utilizing a large amount of memory limited by a 64 -bit block size 3DES is outdated because It is very slow and less efficient. It is mostly used in older systems. RSA-2048 is often used for digital signatures and public-key systems. Elliptic curve cryptography (ECC), such as ECDSA and ECDH, provides the same security as RSA but has smaller keys. This makes ECC more efficient and better for the Internet of Things and mobile devices, and the ElGamal variant provides secure public key encryption and can handle large communications, but it is difficult to set up. Diffie–Hellman (DH) is a basic method for secure key exchange, but it requires larger keys, making it less efficient compared to ECC. Generally, AES and ChaCha20 are suitable for encrypting data, whereas ECC methods are suitable for secure key exchange and authentication, especially in modern settings with limited resources.

Table 1: Symmetric and Asymmetric Cryptographic Algorithms

Algorithms	Category	Memory consumption	Speed / Efficiency	Advantages	Drawbacks	Platform compatibility
AES [1]	Symmeric Block Cipher	Moderate	Moderate	1.NIST standard, high security, 2.widely adopted, supports multiple key sizes (128, 192, 256 bits)	Moderate implementation complexity	1.File encryption, 2.secure communication
ChaCha20 [1]	Symmetric Stream Cipher	Moderate memory footprint	Very fast execution; faster and more efficient than RC4	1.Provides quick cryptography thanks to its simple scheme, High security - more secure compared to RC4, 2.Enhanced diffusion per round, Implemented in Android systems and TLS protocols , 3.Suitable for resource-constrained environments	1.Susceptible to key reuse attacks, Vulnerable to weak initial vector issues, 2.May lack the confusion introduced by substitution-permutation networks compared to block ciphers like AES	1.Appropriate for instantaneous applications (stream services and VPNs) , 2.Excellent for live video streaming and real-time communication 3.Safe communication protocols (TLS/SSL), File encryption
Blowfish [1] [11]	Symmetric Block Cipher	Highest	High	1.High-Speed Performance compared to AES and 3DES, 2..Key Sizes range from 32-448 bits, 3.Flexible Implementation-Flexible due to its maintenance for various modes of operations, 4.Low Implementation Complexity: Characterized as having low complexity	1.High Memory Consumption, 2.High Memory Consumption, 3.64-Bit Block Size Vulnerability: Its 64-bit block leads it to be susceptible to specific attacks	Disk encryption, VPNs, high-speed operations

3DES [11]	Symmetric Block Cipher	Moderate	Slowest	1.Legacy support, 2.simple implementation, Adequate Brute Force Resistance, 3.Smallest Memory Footprint, making it the most memory-efficient among AES, 3DES, and Blowfish	1.Outdated, 2.very slow	Resource-constrained, Legacy systems
RSA-2048[14] [15]	Asymmetric (Public Key) Algorithm	High (substantial)	Slower	1.Industry Standard: Widely recognized and trusted for digital signatures and secure communications, 2.High Compatibility, Flexible Key Sizes,	1.Slow Performance, High Resource Requirements, 2.Quantum Computing Vulnerability	1.Digital Signature Applications, PKI Infrastructure: X.509 certificate systems and public key infrastructure deployments, 2. Legacy Systems: Well-established in existing infrastructure
ECC (ECDSA) [1] [14] [17]	Asymmetric Digital Signature	1.Moderate to Low - Requires significantly smaller key sizes compared to RSA, making it suitable for resource-constrained devices	1.Moderate Performance - Performance can vary widely depending on the selected curve and available hardware support, 2.making performance unpredictable in some cases	1.Delivers identical security criteria to RSA utilizing minor (smaller) keys, creating it more effective 2. More adaptable than RSA to post-quantum cryptography developments 3.Ideal for secure digital signatures 4.Provides robust security with smaller computational overhead	1.Performance can vary widely depending on the selected curve and available hardware support, making performance unpredictable in some cases, While not fully quantum-resistant, 2.ECDSA remains subject to quantum computing threats	Excellent for: 1. Mobile phones or smartphones 2. IoT (Internet of Things) systems 3. Blockchain and cryptocurrency applications 4.Resource-constrained environments 5.Autonomous vehicles and vehicular networks 6. Not Suitable for: Systems with very limited cryptographic capability or no hardware acceleration support
ECC (ECDH) [1] [14] [17]	Asymmetric Key Exchange Protocol	Moderate to Low - More memory-efficient than RSA for equivalent security levels	Good Efficiency - Efficient key exchange with security similar to RSA while utilizing smaller keys	1.Offers a protected approach for key exchange with security similar to RSA while utilizing smaller keys 2. Efficient for symmetric key establishment in secure communication protocols 3. Suitable for resource-constrained environments requiring key exchange capability	1.Requires careful management of transaction size and key 2.Vulnerability to quantum computing attacks like classical discrete logarithm-based systems 3.Implementation complexity similar to ECDSA	1.Excellent for: Lightweight key agreement protocols 3 IoT device communication 2. Autonomous systems requiring secure key negotiation 3. Resource-constrained networks
ElGamal Variant [16]	Asymmetric (Public-Key) Encryption	Moderate - More memory-efficient than RSA-based alternatives for equivalent security	High Volume Communication: More efficient via pairing-free scheme for server-side operations than mRSA	1.Pairing-Free Design: Does not involve pairing computations, making it more efficient 2. Ciphertext Integrity: Applies Fujisaki-Okamoto transformation for ciphertext integrity checks on both SEM and receiver sides 3. High Volume Efficiency: Notably better suited for	1.Key Escrow Problem: Currently does not address key escrow; CA has absolute control of user's secret key 2. Not Certificateless: Users' public keys must be submitted to CA for authentication	1.High-Volume Communication Systems: Excellent for scenarios requiring multiple transactions and communications 2. Server-Centric Architectures: Suitable for

				high volume communication than pairing-based mRSA schemes 4. Server-Side Efficiency: High volume of operations at server sites is much more efficient via mediated ElGamal scheme 5. Security Proof: Proven secure against IND-CCA (Indistinguishable against Chosen-Ciphertext Attack) in random oracle model 6. Extensible Framework: Can be easily transformed into elliptic curve and pairing-based settings	3. Operational Overhead: SEM (Security Mediator) has extensive operational overhead at central server 4. Higher Complexity than mRSA: Performance somewhat undesirable compared to mRSA due to Fujisaki-Okamoto transformation	systems with centralized security mediators 3. Enterprise PKI Systems: Good for organizations requiring mediated encryption
Diffie–Hellman (DH) Algorithm [1] [14] [16]	Asymmetric (Public-Key) Key Exchange Protocol	Moderate to High - Requires larger key sizes for adequate security	Moderate to High - Requires larger key sizes for adequate security	1.Minimalist Design: Known for its minimalism and still the common protocol after many years 2. Proven Foundation: Foundational key exchange protocol for e-trading and different applications 3.Well-Established: Long history of standardization and deployment 3. Symmetric Key Establishment: Excellent for establishing temporary session keys 4. Flexible Application: Can be combined with other cryptographic schemes	1.Computational Overhead: Larger key sizes lead to increased computational and memory overhead 2.Transaction Management: Involves cautious managing of transaction size and key 3. Not Suitable for IoT: Less suitable for severely resource-constrained devices compared to ECC variants	1.E-Commerce & Banking: Common protocol for e-trading and secure key exchange 2.Legacy Systems: Well-established in existing cryptographic infrastructure 3. Mobile Platforms: ECDH preferred for mobile and embedded systems

IV. CONCLUSION

Cryptography is important for keeping information secure and for enabling people to securely communicate when using the internet. In this study, asymmetric cryptography was examined to learn more about it. In cryptography, symmetric cryptography and asymmetric cryptography are important for keeping information secure. However, symmetric cryptography is different from asymmetric cryptography. In most cases, symmetric cryptography is faster and more efficient. It is best for encrypting large data. However, it has some weaknesses related to sharing keys among users. In contrast, asymmetric cryptography is best for sharing keys and identifying users by means of private keys. However, it is slower and consumes more power. In cryptography, symmetric cryptography is best for encrypting data due to efficiency. In addition, asymmetric cryptography is best for sharing keys. In cryptography, it is important to consider various factors while choosing a particular cryptography method. The factors to consider are related to the level of security that is desired as well as how it is supposed to function. In conclusion, cryptography, including asymmetric cryptography, is important for keeping information secure.

V. FUTURE WORK

Although the present study presents a comparative analysis of symmetric and asymmetric cryptography algorithms, there are various scopes for further research and improvement. The scope of the research can be extended to analyze more cryptographic algorithms and their performances in various environments. This will increase knowledge about the various techniques of data encryption and their use in the field of computer science. Another scope of the research can be the experimental implementation of various algorithms in the field of computer science. This will be more effective if the experiments are conducted on various systems and environments, such as cloud computing, mobile computing, and Internet of Things (IoT). The scope of the research can also be extended to the implementation of a hybrid cryptography system. This will be more effective as it will allow the use of the benefits of both symmetric and asymmetric cryptography. The use of a hybrid system will allow the use of asymmetric cryptography for secure key exchange and symmetric cryptography for efficient data encryption. The research can also be extended to the use of better techniques for improving the performance of the cryptography system.

CONFLICTS OF INTEREST

The authors declare that they have no conflicts of interest.

REFERENCES

[1] D. F. Chalob, R. H. Hasan, and S. M. Saber, "A comprehensive review on cryptography algorithms: Methods and comparative analysis," *Int. J. Sci. Res. Sci. Eng. Technol.*, vol. 12, no. 1, pp. 275–282, Jan. 2025, Available from: https://doi.org/10.32628/ijsrset25121171

[2] D. Ramakrishna and M. A. Shaik, "A comprehensive analysis of cryptographic algorithms: Evaluating security, efficiency, and future challenges," *IEEE Access*, vol. 13, pp. 11576–11593, 2025, Available from: https://doi.org/10.1109/ACCESS.2024.3518533

[3] S. Jiang and E. Kumar, "Investigation of biomedical cell image cryptography based on RC4 technique," *Cold Spring Harbor Laboratory*, Dec. 30, 2023, Available from: https://doi.org/10.1101/2023.12.29.573618
[4] E. J. Madarro-Capó, C. M. Legón-Pérez, O. Rojas, and G. Sosa-Gómez, "Measuring avalanche properties on RC4 stream cipher variants," *Appl. Sci.*, vol. 11, no. 20, p. 9646, Oct. 2021, Available from: https://doi.org/10.3390/app11209646
[5] A. Hammami, "Vulnerability assessment of the RC4 cipher using LSTM networks," *Int. J. Eng. Res.*, vol. 13, no. 11, 2024. Available from: https://doi.org/10.5281/zenodo.18130545
[6] S. Dey, S. Maitra, S. Sarkar, and N. K. Sharma, "Significantly improved cryptanalysis of Salsa20 with two-round criteria," *IACR Trans. Symmetric Cryptol.*, vol. 2025, no. 1, pp. 420–443, Mar. 2025, Available from: https://doi.org/10.46586/tosc.v2025.i1.420-443
[7] T. Beyne, Y. L. Chen, and M. Verbauwhede, "A robust variant of ChaCha20-Poly1305."
[8] I. Radhakrishnan, S. Jadon, and P. B. Honnavalli, "Efficiency and security evaluation of lightweight cryptographic algorithms for resource-constrained IoT devices," *Sensors*, vol. 24, no. 12, p. 4008, Jun. 2024, Available from: https://doi.org/10.3390/s24124008
[9] Y. Xiang et al., "Design and analysis of guidance function of permanent magnet electrodynamic suspension," *Technologies*, vol. 11, no. 1, p. 3, Dec. 2022, Available from: https://doi.org/10.3390/technologies11010003
[10] I. Peter, "Comparative analysis of symmetric and asymmetric cryptographic algorithms in secure data transmission." ResearchGate, 2025 Available from: https://tinyurl.com/w3zyn5bf
[11] B. A. Buhari et al., "Performance and security analysis of symmetric data encryption algorithms: AES, 3DES and Blowfish," *Int. J. Adv. Netw. Appl.*, vol. 16, no. 04, pp. 6473–6486, 2025, Available from: https://doi.org/10.35444/ijana.2025.16404
[12] O. S. Faragallah et al., "Improved RC6 block cipher based on data dependent rotations," *Comput. Mater. Contin.*, vol. 70, no. 1, pp. 1921–1934, 2022, Available from: https://doi.org/10.32604/cmc.2022.019798
[13] V. T. Nguyen, P. H. Pham, V. T. D. Le, H. L. Pham, T. H. Vu, and T. D. Tran, "AES-RV: Hardware-efficient RISC-V accelerator with low-latency AES instruction extension for IoT security," *arXiv*, 2025, Available from: https://doi.org/10.48550/ARXIV.2505.11880
[14] J. Xu, "A comprehensive study of digital signatures: Algorithms, challenges and future prospects," *ITM Web Conf.*, vol. 73, p. 03009, 2025, Available from: https://doi.org/10.1051/itmconf/20257303009
[15] K. Assa-Agyei and F. Olajide, "A comprehensive evaluation of the Rivest-Shamir-Adleman (RSA) algorithm performance on operating systems using different key bit sizes," *Int. J. Comput. Appl.*, vol. 185, no. 19, pp. 14–20, Jun. 2023, Available from: https://doi.org/10.5120/ijca2023922884
[16] B. C. Tea, M. R. K. Ariffin, A. H. A. Ghafar, and M. A. Asbullah, "A security-mediated encryption scheme based on ElGamal variant," *Mathematics*, vol. 9, no. 21, p. 2642, Oct. 2021, Available from: https://doi.org/10.3390/math9212642
[17] J. Xiao, Y. Liu, Y. Zou, D. Li, and T. Leng, "An efficient elliptic curve cryptography-based secure communication with privacy preserving for autonomous vehicle," *J. Adv. Transp.*, vol. 2024, no. 1, p. 5808088, Jan. 2024, Available from: https://doi.org/10.1155/2024/5808088

ABOUT THE AUTHORS

VEERESHA M M. is a Research Scholar at Davanagere University. He completed his Master of Science (M.Sc.) at Davanagere University and has actively participated in several conferences in the field of Network Security.

Dr.Vinay S. is an Academician, having 15 years of teaching experience in Computer Science . Currently working in Davangere, Karnataka.

He completed is MCA from VTU, Belgaum, Karnataka. in the year 2011 and Ph.D. from RNTU, Bhopal in the year 2022. have published several research papers in the field of computer networks and network security