Journal of Network Intelligence
Taiwan Ubiquitous Information
 ISSN 2414-8105
Volume **10**, Number **3**, August **2025**

Information Security Design of Student Management System Based on Improved Hybrid Encryption Algorithm

Ya Luo[1,*], Di Luo[2]

[1]Chengdu Textile College, Chengdu 611731, P. R. China
34739142@qq.com

[2]Kmart Australia Ltd., Mulgrave VIC 3170, Australia
luodi820@outlook.com

*Corresponding author: Ya Luo
Received April 18, 2024, revised September 12, 2024, accepted January 9, 2025.

Abstract. Traditional student management systems face data security and privacy protection challenges, especially when dealing with sensitive student information. With the rapid development of education informatisation and cyber security technologies, there is an increasing demand for solutions that can secure data transmission and prevent unauthorised access and data leakage. To address these issues, this study proposes an information security design for student management system based on improved hybrid encryption algorithm. Firstly, the improved Advanced Encryption Standard (AES) algorithm is introduced to optimise the structure of encryption and decryption round function and the forward and inverse column confusion matrices, which reduces the computational complexity and improves the encryption efficiency. Meanwhile, the RSA algorithm is optimised to improve the decryption efficiency by adopting the multi-prime method and ChineseRemainderTheorem (CRT). In addition, the performance of encryption and decryption operations is further enhanced by parallel processing techniques. Secondly, the security of data during transmission and storage is ensured by a hybrid encryption strategy combining AES and RSA algorithms. The experimental results show that compared with the traditional encryption methods, the improved hybrid encryption algorithm significantly improves the processing speed and the overall performance of the system while guaranteeing data security. The designed system enables educational institutions, students and administrators to store and share data securely, and realises efficient management and protection of student information, thus solving the security loopholes and performance bottlenecks existing in traditional systems.
Keywords: Management systems; Information security; AES; RSA; CRT; Multi-prime method; Hybrid encryption

1. **Introduction.** In the digital age, student management systems have become indispensable tools for educational institutions, handling a wide range of sensitive information from personally identifiable information to academic records [1, 2]. The security of these systems is critical as they are increasingly targeted by cyber threats that can lead to data breaches, identity theft, and privacy violations [3]. Traditional security measures, while critical, are often inadequate in the face of sophisticated attacks, highlighting the need for strong and efficient encryption solutions for the integrity and confidentiality of student data [4]. As educational institutions continue to adopt advanced technologies and store increasing amounts of data, it is critical to implement security mechanisms that can withstand current threats and adapt to future challenges [5, 6].

This research aims to address the above problem by proposing an improved hybrid encryption algorithm for designing secure student management systems. By utilising the advantages of symmetric and asymmetric encryption techniques, the proposed solution enhances the overall security posture while maintaining high performance standards. The Advanced Encryption Standard (AES) [7, 8] is optimised for fast and efficient data encryption, while the RSA algorithm [9, 10] is enhanced to ensure secure key management and robust decryption process. Integrating these improved algorithms into the hybrid model not only strengthens the system against unauthorised access, but also facilitates seamless data sharing and collaboration between authorised entities.

1.1. **Related work.** In the last few years, information security technologies have made significant progress in the field of protecting sensitive data, especially in educational management systems [11, 12]. Many researches have focused on the development of encryption algorithms that can secure data transmission and prevent unauthorised access, so that security threats can be detected in a timely manner and appropriate measures can be taken.

Kumar and Mahajan [13] presented a study on a network security technique based on double encrypted data, which enhances the reliability and data security of network communication between terminals and servers by using an improved hybrid AES and MD5 encryption scheme and SSL encrypted transmission channel. The study highlights the need to use secure communication connections and reliable data encryption algorithms in industrial control systems to avoid possible production risks and losses. To address the data security challenges faced by modern healthcare management systems, Abdel-Basset et al. [14] proposed a novel encryption method based on a combination of meta-heuristic optimised security algorithms to reduce privacy leakage and cyber-attacks by unauthorised users and hackers by providing stronger protection for sensitive data. Wang et al. [15] explored the use of blockchain technology in encryption algorithms incorporating blockchain technology in wireless networks with the aim of improving data security and integrity. The study states that blockchain technology-based system stores data through distributed ledger and overcomes data security threats through effective routing protocols and secure hash functions. In the field of cloud computing, Al-Bakri and Mat Kiah [16] proposed a hybrid double encryption-based approach to improve the security of cloud data. This method combines the advantages of NTRU [17] and AES encryption algorithms and introduces quantum adaptive stream encryption to enhance the security of data sharing. By optimising elliptic curve cryptography, Hasan et al. [18] proposed an algorithm that increases the complexity and time consumption of the encryption system to improve the security of IoT healthcare applications. Hussain et al. [19] designed a real-time data encryption system based on the DES algorithm for power systems. By employing a dual encryption system using triple DES and RSA algorithms. improves the security of data network communication. Walle [20] proposed a software-defined network-based security improvement scheme for mobile self-organising networks by combining RSA-AES hybrid encryption algorithm, which improves the security and routing efficiency of the network. Irshad et al. [21] proposed a hybrid approach combining blockchain, generative adversarial network and elliptic curve techniques that for enhancing cloud-based inventory management with improved data security and confidentiality.

1.2. **Motivation and contribution.** These studies have shown that in order to overcome the limitations of a single encryption method, researchers have started to explore hybrid encryption models. In this model, asymmetric algorithms (e.g., RSA) are typically used to securely transmit symmetric keys, while symmetric algorithms (e.g., AES) are used to encrypt the actual data. This approach not only solves the key distribution

problem but also improves the security of the system. However, how to optimise these algorithms for growing data volumes and complex application scenarios is still an open research question. The research work in this paper is based on this background and aims to propose an improved hybrid encryption algorithm to enhance the information security and efficiency of student management systems. The main innovations and contributions of this work include:

(1) The standard AES is improved through the optimisation of the structure of the encryption and decryption round function, the improvement of the forward and inverse column confusion matrices, and the introduction of parallel processing techniques, thus significantly enhancing the execution efficiency of the algorithm and reducing the computational overhead in the encryption and decryption process. This optimisation not only speeds up data processing, enabling the algorithm to respond more quickly to security events, but also reduces the demand for computational resources while maintaining the same level of security.

(2) Decompose the modulo n operation into two smaller modulo p and modulo q power operations, as well as a reversible modulo n operation, which significantly improves the decryption efficiency of RSA.

(3) In order to make full use of the advantages of the improved AES algorithm with high encryption efficiency and the improved RSA algorithm with high attack resistance, the improved AES-RSA hybrid encryption algorithm is proposed.

2. **Analysis of relevant principles.**

2.1. **Security risks faced by the system.**

With the rapid development of information technology, student management systems, as an important part of education informatisation, have been widely used in educational and teaching activities in schools. These systems usually contain sensitive student personal information, performance data, educational assessment reports, and other education-related records. However, with the increase in data volume and the continuous evolution of cyber-attacks, student management systems face many security risks that not only threaten the protection of personal privacy, but also may have a serious impact on the reputation and operations of educational institutions.

Unauthorised access is one of the major security risks facing student management systems. Hackers may illegally gain access to the system and steal or tamper with sensitive data by means of weak password attacks, SQL injection, cross-site scripting attacks, etc. In addition, malicious operation or negligence of internal personnel may also lead to data leakage, e.g. system administrators may cause data leakage by improper operation or over-authorisation of privileges to other users.

Data leakage and data loss are also security issues that student management systems need to focus on. As student management systems usually involve a large amount of personally identifiable information and education records, once these data are leaked, it may cause serious infringement of students' privacy and may even lead to crimes such as identity theft. In addition, data may be accidentally lost due to hardware failure, software defects or catastrophic events, causing incalculable losses to schools and students.

Student management systems may be subject to Denial of Service (DoS) attacks and Distributed Denial of Service (DDoS) attacks [22], which can lead to disruption of system services, affecting the normal operation of the university and the learning activities of students. In addition, with the increase in cyber espionage, student management systems may also become targets of cyber wars between countries or organisations, which requires

that the system must have adequate security measures to protect against potential Advanced Persistent Threats (APTs).

With the popularity of cloud computing and mobile learning, student management systems are increasingly relying on third-party services and mobile devices, which increases the risk of data being intercepted in transit. At the same time, the security of mobile devices themselves has become a new security challenge, as they are more susceptible to loss or theft and may have more security vulnerabilities.

In summary, the security management of student management systems is a complex issue that requires comprehensive consideration of multiple security risks and effective technical and managerial measures to guarantee the privacy, integrity and availability of data. In the next sections, we will explore how to enhance the information security design of student management systems by improving hybrid encryption algorithms to cope with the security risks mentioned above.

2.2. **Principle of AES algorithm.** Advanced Encryption Standard (AES) is a widely used symmetric encryption algorithm that is recognised globally for its high level of security and efficiency. The key length of the AES algorithm can be 128, 192, or 256 bits, which corresponds to the AES-128, AES-192, and AES-256 variants [23]. The number of encryption and decryption rounds corresponding to different key lengths is also different, as shown in Table 1. The main advantage of the AES algorithm is its strong encryption capability, which provides a certain degree of security even in the face of future quantum computer attacks.

Table 1. Different types of AES

Algorithm type	Packet length (Bytes)	Key Length (Bytes)	Number of encryption and decryption rounds
AES-128	4	4	10
AES-192	6	4	2
AES-256	8	4	14

The encryption process of the AES algorithm is shown in Figure 1 and consists of four main steps: SubBytes, ShiftRows, MixColumns, and AddRoundKey. These steps are repeated in each round, except for the last round which does not include the MixColumns step.

(1) SubBytes

The S-box is the non-linear part of the AES algorithm for SubBytes, which maps each byte to another byte. The S-box refers to a substitution table represented as a 16×16 matrix [24]. For example, if we want to find the substitution of the hexadecimal number 54, the sixth row and fifth column of the S-box has the value 20, so the substitution value is S [5][4] = 0x20. The design of the S-box is based on the inverse operation over a finite field, and its mathematical expression can be expressed as follows.

$$S(b) = L^{-1}(b) \mod 256 \tag{1}$$

where b is the input byte; L is a linear transformation over a finite field; and L^{-1} is its inverse transformation.

(2) ShiftRows

The row shift operation circularly shifts each row of the state matrix left by a certain offset. The specific shifting process is shown in Figure 2 below, where the green part indicates the original position of the bytes to be shifted ($S(1,0)$, $S(2,0)$, and $S(3,0)$) and the position of the rows after shifting. The rows are cyclically shifted left by 8, 16 and 24 bits respectively in top-to-bottom order. Mathematically, the offset can be expressed as follows.

$$S(i,j) = S(i,(j+a_i) \mod 4) \tag{2}$$

where i denotes the row index; j denotes the column index; a_i is the offset determined from the row index i.

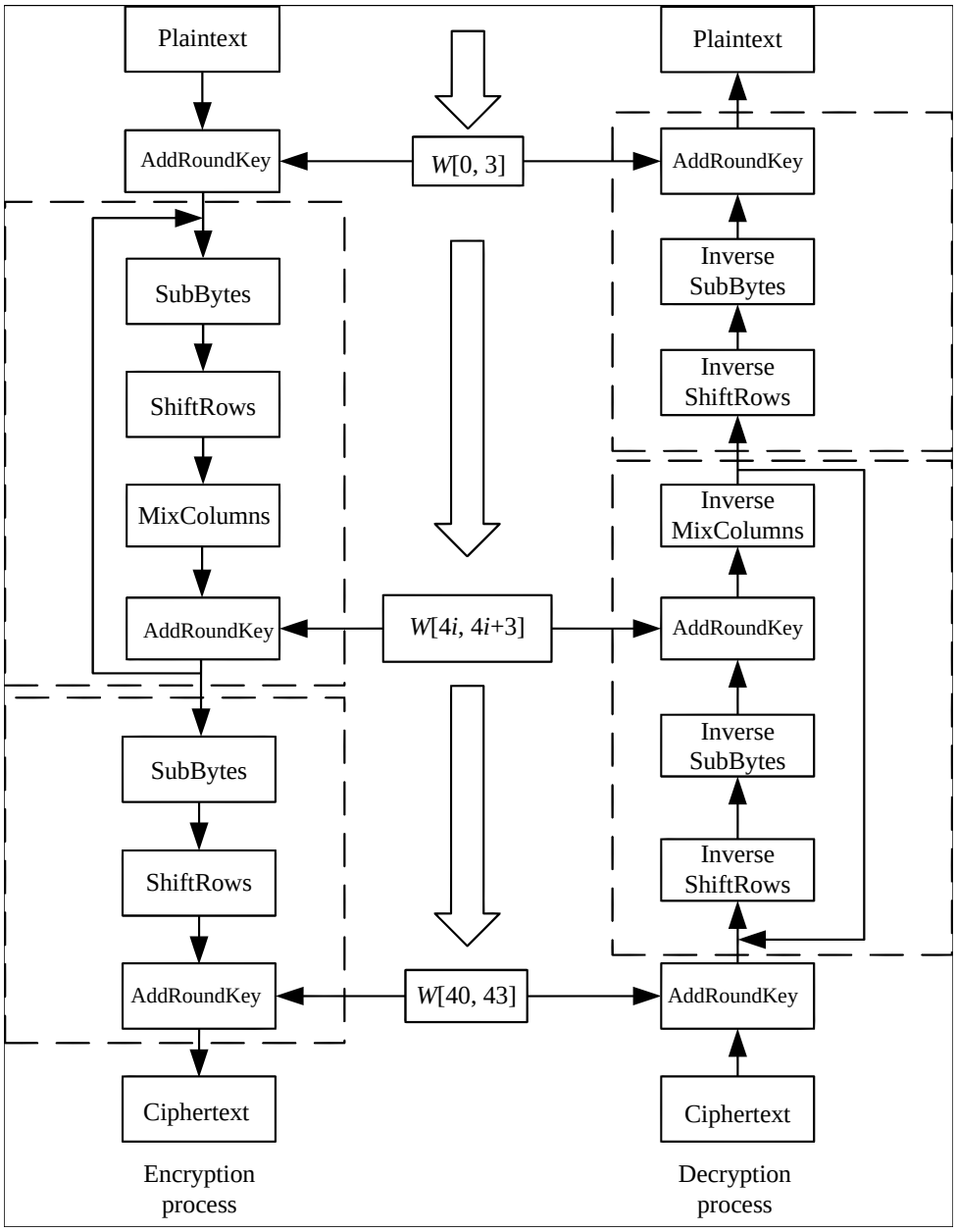

Figure 1. The encryption process of the AES algorithm

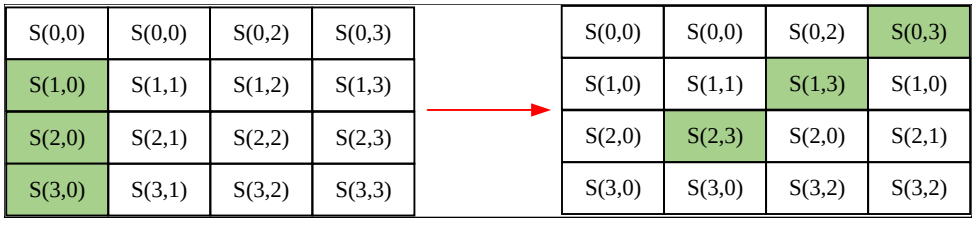

Figure 2. Row Shift Schematic

MixColumns obfuscates each byte in a column by multiplying it with a fixed obfuscation matrix M_C. Obfuscation is the left-multiplication of a standard matrix by a transformation matrix such that each element of the matrix is a weighted sum of all the elements in the original column of the element. The mathematical expression for the confusion matrix is shown as follow.

$$M_C = \begin{bmatrix} 03 & 01 & 01 & 02 \\ 01 & 02 & 03 & 01 \\ 02 & 03 & 01 & 01 \\ 01 & 01 & 02 & 03 \end{bmatrix} \tag{3}$$

The obfuscation operation can be expressed as follows.

$$S' = M_C \times S \mod 256 \tag{4}$$

(4) Round Key Add (AddRoundKey)

The wheel key addition operation is an operation that differentiates each byte of the state matrix from the corresponding byte of the wheel key. The so-called AddRoundKey is simply the bitwise dissimilarity of the content to be encrypted with the round key. Mathematically it is expressed as.

$$S'(i,j) = S(i,j) \oplus R(i,j) \tag{5}$$

where R is the round key matrix; $\oplus$ denotes the hetero-or operation.

(5) KeyExpansion

Key expansion is the process of expanding the initial key into a round key. The expanded round key is used for each round of encryption. The mathematical expression for key expansion can be expressed as.

$$W[i] = g(W[i-1]) \oplus Rcon[i] \tag{6}$$

where W is an extended key array; g is a nonlinear transformation function; and $Rcon$ is a round constant.

(6) Inverse AES algorithm

The decryption process is the inverse of the encryption process, including InvSubBytes, InvShiftRows, InvMixColumns and InvAddRoundKey.

2.3. **Principle of RSA algorithm.** The RSA algorithm is constructed based on the difficulty of factorisation of large integers and is widely used for data encryption and digital signatures [25, 26]. The creation of key pairs, or public and private keys, that are used to encrypt and decode data, is essential to the security of the RSA algorithm.

The key generation process involves choosing two large prime numbers p and q, computing their product $n = p \times q$, and computing the Euler function $\phi(n) = (p-1) \times (q-1)$.

Then, choose an integer e less than $\phi(n)$, usually e can be 65537, and compute the modulo inverse element d of e with respect to $\phi(n)$.

$$d \times e \equiv 1 \mod \phi(n),\ 1 < e < \phi(n) \tag{7}$$

The public key is (n, e); the private key is (n, d).

During the encryption process, for a plaintext message M, the encrypted ciphertext C can be calculated by the following method:

$$C \equiv M^e \mod n \tag{8}$$

For a ciphertext C, the decrypted plaintext message M can be computed by the following method:

$$M \equiv C^d \mod n \tag{9}$$

The RSA algorithm provides a method for encrypting and decrypting data that is both secure and reliable. However, the computational efficiency of the RSA algorithm is relatively low, particularly when working with big data sets. Therefore, in practice, RSA is typically combined utilizing symmetric encryption techniques, i.e., RSA is used to encrypt the key of the symmetric algorithm, while the symmetric algorithm is used to encrypt the actual data. This hybrid encryption is suitable for student management systems that require high security protection.

2.4. **Comparative analysis of encryption algorithms.** In the field of information security, AES and RSA algorithms are two very critical encryption techniques, each of which has unique advantages and limitations.

AES algorithm makes key management relatively simple, as only one key needs to be securely distributed and stored. However, it also means that in a multi-user environment, a unique key is required between each pair of users to secure communications, which can lead to a dramatic increase in the number of keys. The RSA algorithm is an asymmetric encryption algorithm that uses a pair of keys, a public key and a private key. The public key can be shared publicly and used to encrypt data, while the private key must be kept secret and used to decrypt data. This mechanism is more flexible in terms of key management, especially in distributed systems, because the public key can be distributed without limit, while the private key is always kept private.

The security of the AES algorithm is very high, especially when longer keys (e.g., 256 bits) are used. The AES algorithm has withstood extensive cryptanalysis and practical applications, and there is no known effective attack method that can break the AES encryption in a short period of time [27, 28]. The security of the RSA algorithm is based on the difficulty of the large integer factorisation. As the key length increases (e.g., 2048 bits or higher), the RSA algorithm is able to provide a high level of security. However, with the development of quantum computing, the RSA algorithm may face potential security threats because quantum computers are able to theoretically factorise large integers quickly.

The AES algorithm performs efficiently in both software and hardware. The symmetric encryption algorithm is relatively simple to execute and does not require complex mathematical operations, so both encryption and decryption are fast. The RSA algorithm involves multiplication and division of large integers during encryption and decryption, which is computationally very expensive. As a result, the RSA algorithm performs relatively slowly, especially when dealing with large data volumes. In practice, the RSA algorithm is usually used to encrypt only small amounts of data, such as keys for symmetric encryption algorithms.

In summary, the main differences between the AES and RSA algorithms in terms of key management, security and performance speed are shown in Table 2.

Table 2. Key Differences between the AES and RSA Algorithms

Characterisation	**AES**	**RSA**
Key management	More complex	Simpler
Safety	Security is average	Higher security
Encryption speed	Fast and suitable for encryption of large amounts of data	Relatively slow and suitable for small data volume encryption

3. **Improved AES-RSA hybrid encryption algorithm.**

3.1. **Optimisation of the AES algorithm.** In order to improve the performance of AES algorithm in student management system, we have optimised the standard AES algorithm as follows:

(1) Optimisation of encryption and decryption wheel function structure

The standard AES algorithm performs 10, 12 or 14 rounds of operations depending on the key length. To improve the efficiency of encryption and decryption, we can reduce the number of rounds by security analysis while maintaining sufficient security. For example, for AES-128, we can reduce the number of rounds to 8 without affecting its security. The formula for the reduction of the number of rounds can be expressed as:

$$Rounds = \frac{KeyLength}{32} - 6 \tag{10}$$

where *KeyLength* is the key length. For a 128-bit key, this is reduced to 8 rounds.

Among the four operations of the AES wheel function, the row shift operation only affects the position of the byte element in the state matrix, and does not affect the value of the byte element; while the SubBytes operation only replaces the byte element with its corresponding value in the S-box, and does not change the position of the replaced byte element value in the state matrix [29]. Therefore, the ShiftRows and SubBytes operations in the encryption wheel function can be switched in order without affecting the implementation of the wheel function.

In order to make the encryption and decryption processes have the same wheel function structure, the SubBytes and row shift operations in the encryption wheel function can be switched in the order of the encryption and decryption wheel function structure after the switch is shown in Figure 3.

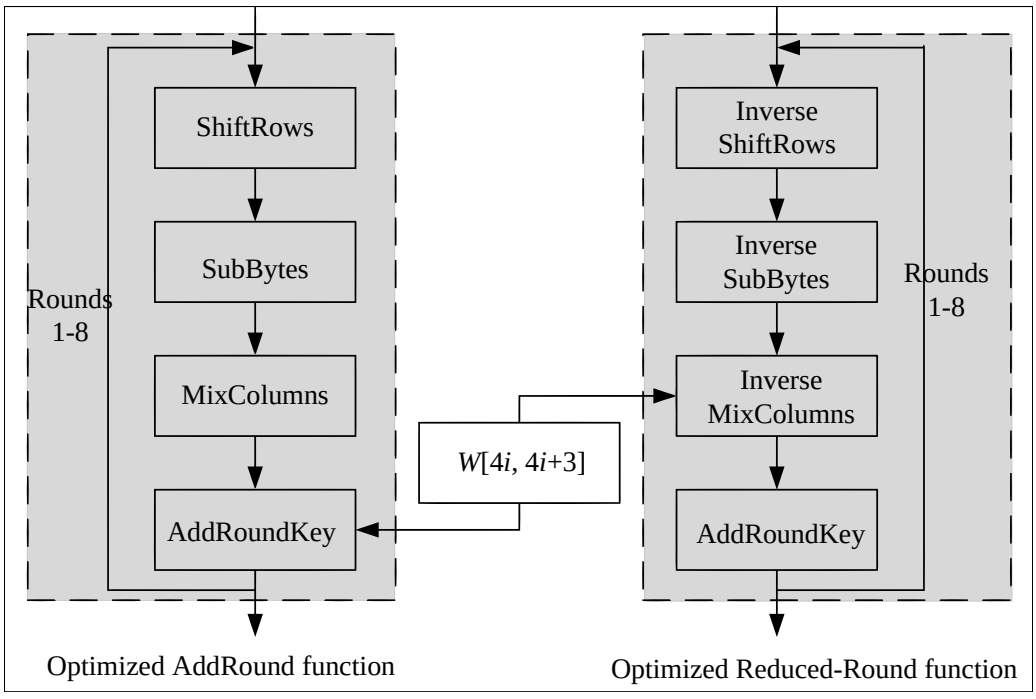

Figure 3. Optimised encryption/decryption wheel function structure

(2) Optimisation of positive inverse column confusion matrices

In the AES algorithm, MixColumns and inverse MixColumns operations involve fixed mathematical transformations. The operations of inverse MixColumns operations for decryption wheel functions are much more complex than the MixColumns operations for

encryption wheel functions. To reduce the computational complexity of these operations, we can optimise the confusion matrix. We use the simplest form of the positive inverse column confusion matrix M over a finite field, which reduces the number of multiplication and different-or operations.

$$M = \begin{pmatrix} 02 & 01 & 03 & 01 \\ 01 & 02 & 01 & 03 \\ 03 & 01 & 02 & 01 \\ 01 & 03 & 01 & 02 \end{pmatrix} = M^{-1} \tag{11}$$

The optimised MixColumns operation can be represented as

$$C' = M \times C \bmod 256 \tag{12}$$

where C is a column of the state matrix; M is the optimised confusion matrix.

Since the inverse matrix of the matrix M over a finite field is itself, this makes the arithmetic complexity of the forward-inverse column confusion operation the same. In addition, because the element values of the matrix are the same as those of the column confusion matrix in AES algorithm, but the corresponding positions of the element values are different, the computational complexity of using the matrix M as the positive and negative column mixed matrix is the same as that of the column confusion operation in the unoptimized AES algorithm.

(3) Parallel processing

Some parts of the AES algorithm, such as SubBytes and AddRoundKey, can be processed in parallel. By implementing parallel mechanisms in hardware or software, we can significantly increase the execution speed of the algorithm. For example, we can partition the state matrix into four parts and perform SubBytes operations simultaneously on multiple processors or cores. The efficiency of parallel processing can be expressed as:

$$Speedup = \frac{1}{1 - \frac{1}{P}} \tag{13}$$

where P is the number of parallel processors. Ideally, as the number of parallel processors increases, the speed of encryption and decryption will increase exponentially.

With the above optimisation measures, we can improve the encryption and decryption efficiency of the AES algorithm in the student management system while maintaining its high security.

3.2. **RSA algorithm optimisation.** In order to improve the efficiency of the RSA in the student management system while ensuring security, we use the following two different methods to improve the encryption and decryption operations respectively:

(1) Improvement of cryptographic operations

In order to reduce the number of power operations in the RSA encryption process, we use a chunked encryption method, which divides the plaintext message M into multiple chunks and then encrypts each chunk independently. This method can reduce the number of power operations and improve the encryption efficiency. The improved encryption process can be expressed as

$$C_i = M_i^e \bmod n \tag{14}$$

where C_i is the ith encrypted block; M_i is the ith block of the plaintext message; e is the public key index; and n is the modulus. This method transforms the original single-block encryption into multi-block encryption, reducing the complexity of each curtain operation.

(2) Decryption operation improvement

In the RSA decryption process, we can use Chinese Remainder Theorem (CRT) to optimise the computation. CRT can decompose the power operation of modulus n into the power operation of modulus p and modulus q, where p and q are the prime factors of modulus n. The improved decryption process is as follows.

Compute the inverse powers of the C-module p and the C-module q.

$$C_p = C^{d_p} \bmod p \qquad C_q = C^{d_q} \bmod q \tag{15}$$

Use the CRT to reconstruct the value of C modulo n :

$$M = (C_p - C_q) \times p \times q^{-1} \bmod n \tag{16}$$

where d_p and d_q are the inverse elements of d modulo $p-1$ and d modulo $q-1$, respectively; p and q are prime factors of n.

In this way, we can significantly improve the decryption efficiency by decomposing the expensive modulo n Shufu operation into two smaller modulo p and modulo q power operations, as well as a reversible modulo n operation. With the above improvements, we can significantly improve the efficiency of encryption and decryption of the RSA algorithm while maintaining its security. These optimisations are especially important for scenarios such as student management systems that need to deal with large amounts of data to ensure the security of data transmission without overly affecting the system performance.

3.3. **AES-RSA based encryption scheme.**

3.3.1. *Encryption process.* The hybrid encryption scheme based on AES-RSA combines the high efficiency of the AES algorithm and the security of the RSA algorithm, which is especially suitable for scenarios that need to guarantee the efficiency and security of data transmission at the same time, such as student management systems. The following is the detailed encryption process of this hybrid encryption scheme:

Step 1: Generate RSA key pair including public key (n, e) and private key (n, d). Generate AES key K which is used for encryption and decryption of AES algorithm.

Step 2: Encrypt plaintext data M with AES key K to get ciphertext C_{AES}.

Step 3: Encrypt the AES key K using the RSA public key (n, e) to get the cipher key C_K.

$$C_K = K^e \bmod n \tag{17}$$

Step 4: Send the AES ciphertext C_{AES} and the encrypted AES key C_K together to the receiver.

Step 5: The receiver decrypts the encrypted AES key C_K using the RSA private key (n, d) to recover the AES key K.

$$K = C_K^d \bmod n \tag{18}$$

Step 6: Use the recovered AES key K to decrypt the received AES ciphertext C_{AES} to get the original plaintext data M.

Through the above process, the hybrid AES-RSA based encryption scheme ensures the security and efficiency of the data during transmission. The RSA algorithm is used to encrypt the symmetric key while the AES algorithm is used to encrypt the real information. The advantage of this approach is that even in the case where the RSA key is cracked, the attacker still has to deal with the security of the AES algorithm, which makes it significantly more difficult to crack the entire encryption scheme. Also, due to the high efficiency of the AES algorithm, the process of encrypting and decrypting data does not significantly affect the system performance. Applying this encryption scheme in

student management system can effectively protect students' personal information and educational data from unauthorised access and data leakage.

3.3.2. *Efficiency Analysis.* In order to verify the advantage of hybrid encryption mode in terms of encryption and decryption operation efficiency, this paper focuses on testing the speed of encryption and decryption of the same encrypted data to be treated by RSA-1024, AES-128 and improved AES-RSA encryption algorithms. The test results are shown in Table 3 below.

Table 3. Algorithmic Encryption Speed Test

Key length (bit)	**30kb file (ms)**	**100kb file (ms)**	**200kb file (ms)**
RSA 1024	806	2105	6528
AES 128	3.62	11.65	18.34
Improved AES-RSA 1024	4.75	13.88	22.09

The above comparison can be more intuitively seen, first of all, the RSA encrypts large amounts of data when the performance is significantly worse than the AES and the improved AES-RSA. In contrast, the AES encryption and the improved AES-RSA encryption are much faster in encrypting and decrypting both small data and large data. Moreover, the improved AES-RSA encryption has higher encryption efficiency, and the difference in encryption time is not much compared to the single AES, and the encryption time is in the same order of magnitude.

3.3.3. *Security analysis.* In order to test and compare the security of the encryption algorithms, we use the brute force attack method to carry out experiments to test the three algorithms' resistance to attack, and the results are shown in Figure 4. Comparing the cracking time of the three algorithms in the face of brute force attack, the time consumed by the improved AES-RSA algorithm to resist the attack is much higher than that of the traditional RSA algorithm and AES algorithm. This indicates that the improved AES-RSA algorithm can effectively improve the security while enhancing the encryption efficiency, which proves the superiority of the algorithm.

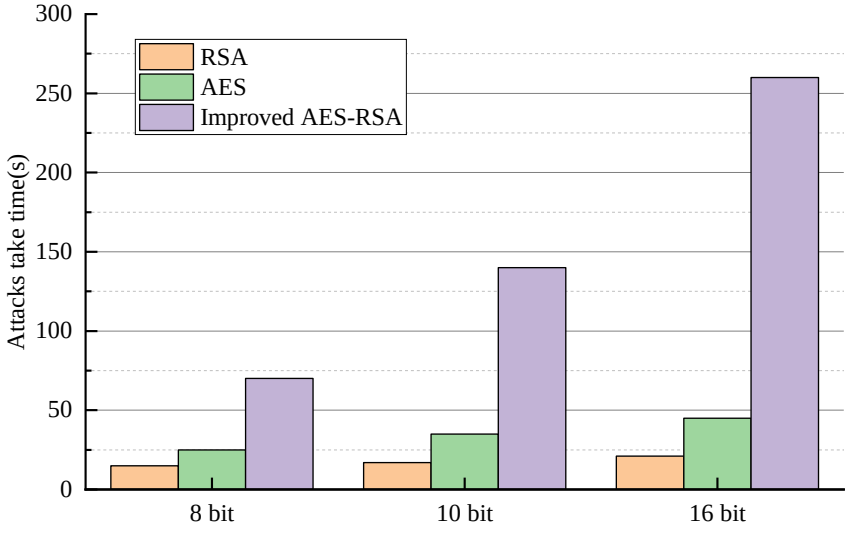

Figure 4. Comparison of violent attack times

4. System testing.

4.1. **Test environment.** In order to comprehensively evaluate the performance and security of the AES-RSA hybrid encryption algorithm based on AES-RSA in the student management system, a series of experiments were conducted in the WEB test environment. The hardware configuration and software parameters in the test environment are shown in Table 4. Network packet capturing is performed using the Wireshark tool to simulate interception and tampering attacks that may be encountered during data transmission.

Table 4. Test environment

Form	**Descriptions**
Processing unit	IntelCorei7-9700K@3.6GHz, 8 cores and 16 threads
Random access memory (RAM)	32GB DDR4 RAM
Stockpile	512GB NVMe SSD
Reticulation	1Gbps Ethernet connectivity
Operating system	Windows 10 Professional, 64-bit version
Development environment	Python 3.8, PyCryptodome cryptographic library
Comprehensive database	MySQL 8.0 for storing student administrative data

4.2. **Performance testing.** In order to verify the performance of the system under multi-user usage, this paper uses concurrency testing to examine the reliability and robustness of the system. The purpose of the performance test is to evaluate the response time and system resource consumption of the student management system based on the improved AES-RSA hybrid encryption algorithm under different conditions, and the results are shown in Table 5 below.

Table 5. Performance test results

Number of users	Intervals	Average response time	Successes	Success rate (%)	WEB server CPU usage (%)		Database server CPU usage (%)	
					Average	Greatest	Average	Greatest
20	0	6.302	11	100.00	25.07	35.4	27.16	58.09
60	0	6.432	23	100.00	28.51	37.09	37.76	52.16
100	0	10.582	43	100.00	29.95	42.21	55.03	61.23
140	0	10.523	53	100.00	39.09	59.47	71.19	82.16
180	0	11.623	96	100.00	40.84	54.13	84.12	92.93
220	0	13.612	103	96.37	40.47	56.8	90.87	92.73
260	0	15.292	117	91.59	42.49	55.73	90.9	93.07

The average response time gradually increases as the number of users increases, indicating that the system takes more time to handle more concurrent requests. However, the average response time remains under 15 seconds even when the number of users increases to 260, which could mean that the system is able to handle higher concurrent loads with the current configuration. The success rate stays above 90% and remains close to 100% even when the number of users reaches 220, which indicates that the system is still very reliable under high loads. The CPU occupancy of the WEB server rises as the number of users increases, but the growth rate is relatively low. This indicates that the WEB server has not reached the limit of its processing power in the current test. The success rate is

close to or reaches 100 per cent when the number of users is low (20, 60, 100, 140, 180), which indicates that the system is able to process requests consistently.

In summary, the student management system based on the improved AES-RSA hybrid encryption algorithm shows good reliability and robustness in the case of multi-user concurrent use. The system is able to handle a large number of concurrent requests in a short period of time, and most of the requests can be responded to within 15 seconds. Although some aspects of the system may experience performance degradation under high loads, overall the performance of the system is satisfactory.

5. **Conclusion.** In this work, an improved hybrid AES-RSA encryption algorithm is proposed to improve the information security and efficiency of student management system. The standard AES is improved by optimising the structure of the encryption and decryption round function, the improvement of the forward and inverse column confusion matrices, and the introduction of parallel processing techniques, so as to significantly enhance the execution efficiency of the algorithm and reduce the computational overhead in the encryption and decryption process. The modulo n surrogate operation is decomposed into two smaller modulo p and modulo q power operations, as well as an invertible modulo n operation, which significantly improves the decryption efficiency of RSA. The experimental results show that the improved AES-RSA algorithm consumes much more time than the traditional RSA algorithm and AES algorithm to defend against attacks. The student management system based on the improved AES-RSA hybrid encryption algorithm shows good reliability and robustness in the case of concurrent multi-user usage. The system is able to handle a large number of concurrent requests in a short period of time and most of the requests can be responded within 15 seconds. The follow-up research plan will be to research and design lightweight but secure encryption algorithms for resource-constrained environments, such as mobile devices, in order to meet the needs of low energy consumption and high performance.

REFERENCES

[1] S. Bharamagoudar, R. Geeta, and S. Totad, "Web based student information management system," *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 2, no. 6, pp. 2342-2348, 2013.

[2] E. S. Walia and E. S. K. Gill, "A framework for web-based student record management system using PHP," *International Journal of Computer Science and Mobile Computing*, vol. 3, no. 8, pp. 24-33, 2014.

[3] S. Natek and M. Zwilling, "Student data mining solution–knowledge management system related to higher education institutions," *Expert Systems with Applications*, vol. 41, no. 14, pp. 6400-6407, 2014.

[4] T.-Y. Wu, C.-M. Chen, K.-H. Wang, C. Meng, and E. K. Wang, "A provably secure certificateless public key encryption with keyword search," *Journal of the Chinese Institute of Engineers*, vol. 42, no. 1, pp. 20-28, 2019.

[5] T.-Y. Wu, X. Fan, K.-H. Wang, C.-F. Lai, N. Xiong, and J. M.-T. Wu, "A DNA Computation-Based Image Encryption Scheme for Cloud CCTV Systems," *IEEE Access*, vol. 7, pp. 181434-181443, 2019.

[6] T.-Y. Wu, C.-M. Chen, K.-H. Wang, and J. M.-T. Wu, "Security Analysis and Enhancement of a Certificateless Searchable Public Key Encryption Scheme for IIoT Environments," *IEEE Access*, vol. 7, pp. 49232-49239, 2019.

[7] J. Nechvatal et al., "Report on the development of the Advanced Encryption Standard (AES)," *Journal of research of the National Institute of Standards and Technology*, vol. 106, no. 3, 511, 2001.

[8] M. A. Wright, "The advanced encryption standard," *Network Security*, vol. 2001, no. 10, pp. 11-13, 2001.

[9] R. Gennaro, T. Rabin, S. Jarecki, and H. Krawczyk, "Robust and efficient sharing of RSA functions," *Journal of Cryptology*, vol. 13, pp. 273-300, 2000.

[10] 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.
[11] H. Venter and J. H. Eloff, "A taxonomy for information security technologies," *Computers & Security*, vol. 22, no. 4, pp. 299-307, 2003.
[12] H. Cavusoglu, B. Mishra, and S. Raghunathan, "The value of intrusion detection systems in information technology security architecture," *Information Systems Research*, vol. 16, no. 1, pp. 28-46, 2005.
[13] R. Kumar and G. Mahajan, "A novel framework for secure file transmission using modified AES and MD5 algorithms," *International Journal of Information and Computer Security*, vol. 7, no. 2-4, pp. 91-112, 2015.
[14] M. Abdel-Basset, D. El-Shahat, I. El-Henawy, A. K. Sangaiah, and S. H. Ahmed, "A novel whale optimization algorithm for cryptanalysis in Merkle-Hellman cryptosystem," *Mobile Networks and Applications*, vol. 23, pp. 723-733, 2018.
[15] S.-Y. Wang, Y.-J. Hsu, and S.-J. Hsiao, "Integrating blockchain technology for data collection and analysis in wireless sensor networks with an innovative implementation," in *2018 International Symposium on Computer, Consumer and Control (IS3C)*, IEEE, 2018, pp. 149-152.
[16] S. H. Al-Bakri and M. Mat Kiah, "A novel peer-to-peer SMS security solution using a hybrid technique of NTRU and AES-Rijndael," *Scientific Research and Essays*, vol. 5, no. 22, pp. 3455-3466, 2010.
[17] Y. Agus, M. A. Murti, F. Kurniawan, N. D. Cahyani, and G. B. Satrya, "An efficient implementation of ntru encryption in post-quantum internet of things," in *2020 27th International Conference on Telecommunications (ICT)*, IEEE, 2020, pp. 1-5.
[18] M. K. Hasan, S. Islam, R. Sulaiman, S. Khan, A.-H. A. Hashim, S. Habib, M. Islam, S. Alyahya, M. M. Ahmed, and S. Kamil, "Lightweight encryption technique to enhance medical image security on internet of medical things applications," *IEEE Access*, vol. 9, pp. 47731-47742, 2021.
[19] I. Hussain, M. C. Negi, and N. Pandey, "A secure IoT-based power plant control using RSA and DES encryption techniques in data link layer," in *2017 International Conference on Infocom Technologies and Unmanned Systems*, IEEE, 2017, pp. 464-470.
[20] Y. M. Walle, "Hybrid RSA–AES-Based Software-Defined Network to Improve the Security of MANET," *Open Information Science*, vol. 8, no. 1, 20240001, 2024.
[21] R. R. Irshad, Z. Hussain, I. Hussain, S. Hussain, E. Asghar, I. M. Alwayle, K. M. Alalayah, A. Yousif, and A. Ali, "Enhancing Cloud-Based Inventory Management: A Hybrid Blockchain Approach with Generative Adversarial Network and Elliptic Curve Diffie Helman Techniques," *IEEE Access*, vol. 12, pp. 25917-25932, 2024.
[22] O. Osanaiye, K.-K. R. Choo, and M. Dlodlo, "Distributed denial of service (DDoS) resilience in cloud: Review and conceptual cloud DDoS mitigation framework," *Journal of Network and Computer Applications*, vol. 67, pp. 147-165, 2016.
[23] N. Z. Bawany, J. A. Shamsi, and K. Salah, "DDoS attack detection and mitigation using SDN: methods, practices, and solutions," *Arabian Journal for Science and Engineering*, vol. 42, pp. 425-441, 2017.
[24] K. Muttaqin and J. Rahmadoni, "Analysis and design of file security system AES (advanced encryption standard) cryptography based," *Journal of Applied Engineering and Technological Science (JAETS)*, vol. 1, no. 2, pp. 113-123, 2020.
[25] A. Altigani, S. Hasan, B. Barry, S. Naserelden, M. A. Elsadig, and H. T. Elshoush, "A polymorphic advanced encryption standard–a novel approach," *IEEE Access*, vol. 9, pp. 20191-20207, 2021.
[26] T. S. Obaid, "Study a public key in RSA algorithm," *European Journal of Engineering and Technology Research*, vol. 5, no. 4, pp. 395-398, 2020.
[27] M. S. A. Mohamad, R. Din, and J. I. Ahmad, "Research trends review on RSA scheme of asymmetric cryptography techniques," *Bulletin of Electrical Engineering and Informatics*, vol. 10, no. 1, pp. 487-492, 2021.
[28] L. Teng, H. Li, S. Yin, and Y. Sun, "A Modified Advanced Encryption Standard for Data Security," *International Journal of Network Security*, vol. 22, no. 1, pp. 112-117, 2020.
[29] S. M. Kareem and A. M. S. Rahma, "New method for improving add round key in the advanced encryption standard algorithm," *Information Security Journal: A Global Perspective*, vol. 30, no. 6, pp. 371-383, 2021.
[30] N. Ahmad and S. R. Hasan, "A new ASIC implementation of an advanced encryption standard (AES) crypto-hardware accelerator," *Microelectronics Journal*, vol. 117, 105255, 2021.