International Journal of Safety and Security Engineering
Vol. 16, No. 10, October, 2026, pp. 2299-2306
Journal homepage: http://iieta.org/journals/ijsse

Digital Signature Based Secure Aggregation of Privacy-Preserving Student Feedback in Higher Education

Ramachandra H. N.[1], K. S. Shivaprakasha[2*]

[1] Department of Electronics and Communication (Advanced Communication Technology), NMAM Institute of Technology (NMAMIT), Nitte (Deemed to be University), Nitte 574110, India
[2] Department of Electronics and Communication Engineering, NMAM Institute of Technology (NMAMIT), Nitte (Deemed to be University), Nitte 574110, India

Corresponding Author Email: shivaprakasha.ks@nitte.edu.in

Copyright: ©2026 The authors. This article is published by IIETA and is licensed under the CC BY 4.0 license (http://creativecommons.org/licenses/by/4.0/).

https://doi.org/10.18280/ijsse.161009

Received: 25 August 2026
Revised: 29 September 2026
Accepted: 9 October 2026
Available online: 31 October 2026

Keywords:
verifiable aggregation, multi-signature schemes, Boneh Lynn Shacham signatures, Bresson–Catalano–Pointcheval cryptosystem, ciphertext computation, additive Homomorphic Encryption

ABSTRACT

In today's educational institutions, student feedback is considered an important tool for assessing the teaching effectiveness and improving academic quality. Conventional systems usually save student feedback ratings in plaintext format, making them susceptible to modifications, unauthorized access, and privacy violations. One possible solution to this problem is to use Homomorphic Encryption (HE). HE allows computations to be performed on ciphertexts by eliminating the necessity for decryption and re-encryption during processing. This paper proposes authentication and aggregation (AUT-AGG), a privacy-preserving student feedback technique for institutional management systems. The AUT-AGG protocol uses the Boneh-Lynn-Shacham multi-signature aggregation algorithm to authenticate student identities. Furthermore, the Bresson–Catalano–Pointcheval technique is used to encrypt student ratings so as to guarantee efficient computations on ciphertexts. Only a central authority is able to decrypt the aggregated encrypted feedback, ensuring privacy. The experimental evaluation of the scheme shows its effectiveness in efficient aggregation of encrypted ratings and signatures with reduced computational and communication overheads. The AUT-AGG protocol offers a promising solution for trustworthy faculty evaluation systems in resource-constrained educational environments through efficient and secure aggregation of feedback with privacy preservation, thus contributing to the advancement of cryptographic applications in secure data management in institutions of learning.

1. INTRODUCTION

Outcome-Based Education (OBE) has become a key requirement in accreditation processes for higher education institutions, shifting the focus from teacher-centric instruction to student-centric learning and measurable outcomes [1]. This transition is further supported by the National Education Policy (NEP) 2020, which emphasizes quality teaching, skill-based learning, infrastructure, and student expectations to improve global competitiveness [2, 3]. In this context, faculty evaluation through student feedback plays an important role in assessing teaching effectiveness, communication skills, class engagement, and curriculum relevance. However, conventional plaintext feedback systems expose student opinions to unauthorized access, potentially revealing the identity of critics or supporters and discouraging honest feedback due to fear of reprisal.

Many digital feedback systems generally do not have secure mechanisms of transmitting and storing responses; hence it becomes possible for an unauthorized party to access and manipulate rating data, which may have a negative effect on the users' confidence levels in case they know that their evaluation is going to be considered by management staff members [4]. Cryptographic methods like AES encryption [5] could be used to enhance security by making sure that there is no compromise in the confidentiality and integrity of rating data during transmission and storage processes in digital feedback systems [6]. Most traditional feedback systems have the issue that ratings are sent and stored in plaintext, revealing the identities and opinions of voters.

In feedback and rating systems, the primary objective is to obtain the aggregated rating while preserving the privacy of individual responses. Homomorphic Encryption (HE) schemes are used for analyzing sensitive data like medical records, ratings, election polling data etc. A federated learning scheme with HE for Electronic Healthcare Records was presented in [7]. Also, schemes like Paillier and other schemes in their family [8-10] are based on an additive form of homomorphism, thus allowing for secure aggregation of ratings in the ciphertext space. The end result is finally decrypted by the receiver alone to avoid any privacy issues. In addition to the privacy requirement, another crucial feature that must be guaranteed in a rating system is that of authenticity and integrity of the rating data. While it may be easy to achieve the former through signing each rating individually, this could lead to an unnecessarily high amount

of communication and validation overhead, particularly in large groups.

To address these challenges, this work proposes authentication and aggregation (AUT-AGG), a privacy-preserving protocol for secure student feedback systems. Student ratings are encrypted using a Bresson–Catalano–Pointcheval (BCP) additive HE scheme, enabling secure aggregation without intermediate decryption, while only a central authority decrypts the final result. Authentication and integrity are ensured through Boneh Lynn Shacham (BLS) signature aggregation [11], which verifies a single compact signature instead of multiple individual signatures. By combining HE with aggregated authentication, AUT-AGG ensures confidentiality, integrity, and scalability to secure and efficient feedback collection.

The rest of the paper is organised as follows: Section 2 presents the literature review in the area of digital signature and HE. Section 3 gives an insight into the methodology used in the proposed architecture. Section 4 presents the performance evaluations. Finally, Section 5 concludes the paper.

2. RELATED WORKS

The HE enables computation over ciphertexts while preserving decryption correctness. Early HE schemes were primarily Partially Homomorphic Encryption (PHE) systems supporting either additive or multiplicative operations. The studies [12, 13] provided multiplicative homomorphism based on discrete logarithm problem. A probabilistic encryption with XOR homomorphism [14] provided a foundation for privacy-preserving computation. A scheme supporting the extension of additive homomorphism to larger message spaces was proposed [15]. Paillier cryptosystem became widely adopted for efficient ciphertext aggregation [8]. Further improvements include the Cramer–Shoup scheme [10] for resistance against adaptive chosen-ciphertext attacks and the BCP scheme [9], which supports double-trapdoor additive HE for threshold and distributed environments. Digital signature schemes ensure authentication, integrity, and non-repudiation in secure systems. RSA supports digital signature and is widely used in certification systems [12]. ElGamal [13] formed the basis for DSA, and ECDSA improved security efficiency through elliptic curve cryptography. Table 1 shows existing basic AUT-AGG schemes proposed in the literature.

Table 1. Cryptographic primitives

Paper / Scheme	Digital Signature	Aggregated Digital Signature	Ciphertext Aggregation
RSA [12]	√	×	×
Paillier [8]	Not the primary focus	×	√

AUT-AGG techniques have been extensively studied for privacy-preserving applications. Secure authentication and integrity verification mechanisms were presented in [16, 17], while compact and aggregation-capable signature schemes were proposed in [18, 19]. A trusted third-party authentication framework was introduced in [20], and secure aggregation techniques have been applied to federated learning [21], contact tracing [22], Byzantine fault-tolerant consensus [23], blockchain-based credential verification [24], and decentralized IoT authentication [25].

Several authentication and privacy-preserving data aggregation schemes have been proposed for secure data collection. The Decentralized, Efficient and Selective Aggregation (DESA) scheme [26] provides efficient and selective aggregation with authentication. The Fog-Enabled Secure Data Aggregation (FESDA) scheme [27] uses a modified Paillier cryptosystem for privacy-preserving aggregation in fog-enabled smart-grid environments. The Verifiable Privacy-Preserving Data Aggregation (LVPDA) scheme [28] combines Paillier-based HE with an online/offline signature mechanism to provide lightweight and verifiable data aggregation. However, the existing schemes mainly focus on specific IoT, fog, edge, or smart-grid environments and do not provide the complete combination of digital signature based authentication, cloud-side verification, receiver-side verification, and double-trapdoor decryption in a single aggregation framework. In particular, the existing Paillier-based approaches do not provide the double-trapdoor decryption capability used in the proposed framework.

Therefore, there is a need for an aggregation scheme that can simultaneously support privacy-preserving homomorphic aggregation, efficient digital signature based authentication, verification at both the cloud and receiver sides, and flexible decryption through a double-trapdoor mechanism. To address this gap, the proposed AUT-AGG scheme integrates BCP-based homomorphic aggregation with BLS-based authentication and verification of digital signature at the cloud and receiver, together with double-trapdoor decryption. Furthermore, the proposed framework applies these security and aggregation mechanisms to an educational data-collection environment, which is the application context considered in this work. The application of AUT-AGG schemes in educational systems remains largely unexplored, motivating the development of scalable and privacy-preserving frameworks for such environments.

Table 2 shows a comparative analysis of the existing protocols/schemes with the proposed AUT-AGG.

Table 2. Comparative analysis

Scheme	HE Aggn	BLS Authn	Cloud Verifn	Receiver Verifn	Double Trapdoor
DESA [26]	√	√	×	√	×
FESDA [27]	√	×	×	×	×
LVPDA [28]	√	×	√	×	×
AUT-AGG	√	√	√	√	√

3. METHODOLOGY

3.1 System model

Proposed authenticated, secure privacy-preserving AUT-AGG protocol has three entities: (i) Students Site (SS) (ii) Cloud Computer (CC) (iii) Data Requester (DR). Figure 1 shows the cloud architecture of AUT-AGG protocol which comprises of the following entities.

SS consists of registered students who provide faculty feedback ratings (in a scale of 1–10 through a feedback interface). The ratings are encrypted into ciphertexts, while the student count represents the number of submissions. Using the BLS signature scheme, individual signatures are generated and

aggregated into a single signature by the Signature subblock of SS. The ciphertexts, student count, and aggregated signature are then transmitted to the CC.

CC collects, processes, and stores encrypted feedback and student counts. It verifies aggregated BLS signatures using Key Generation and Management Centre (KGMC) generated public keys, performs ciphertext-domain aggregation to preserve privacy, and generates a digital signature before forwarding the aggregated ciphertext to the DR.

DR acts as the central entity of the system and possesses the master key. It verifies the digital signature received from CC and decrypts the aggregated feedback ciphertext after successful authentication. The DR also receives the count of students from CC from which the average rating is calculated.

KGMC is a trusted entity responsible for generating and distributing the public and private/secret keys required by different entities in the system.

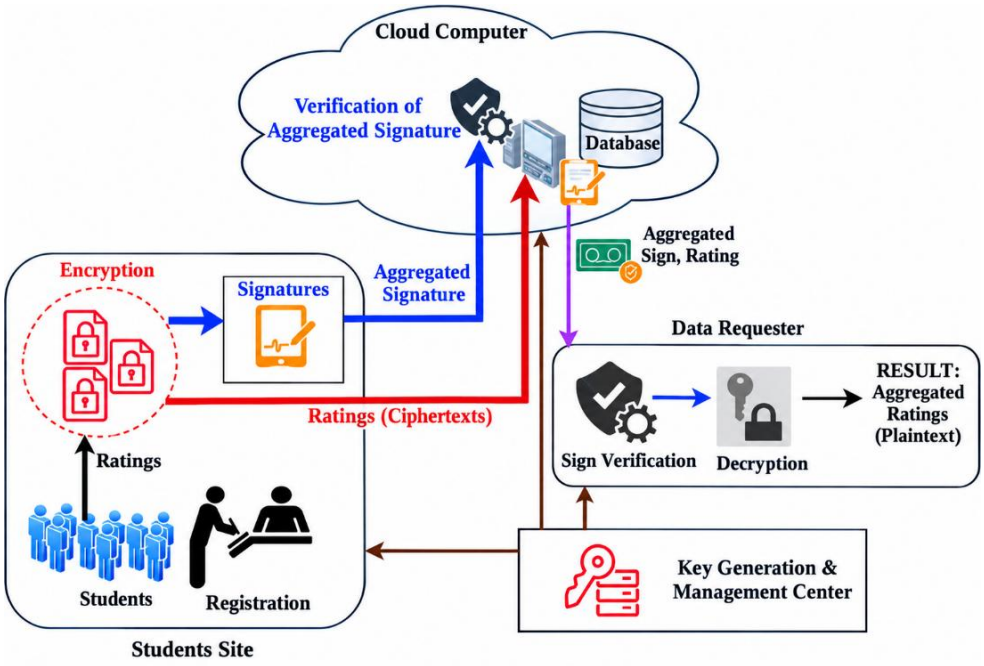

Figure 1. System model for proposed authentication and aggregation (AUT-AGG) protocol

3.2 Analytical model

In this section we provide an insight into AUT-AGG protocol. AUT-AGG protocol relies on the BCP cryptosystem [9] and BLS digital signature scheme [11].

3.2.1 Bresson–Catalano–Pointcheval building blocks

AUT-AGG protocol incorporates BCP cryptosystem which is used to generate the ciphertext from the student's feedback ratings. The double trapdoor feature of BCP cryptosystem is incorporated for the final decryption at the DR.

Key generation. The AUT-AGG protocol employs the BCP cryptosystem to encrypt students' feedback ratings. Its double trapdoor feature enables final decryption at the DR. The cryptosystem generates secret keys sk and corresponding public keys pk using two-prime numbers p and q to produce n.

$$n = p.q \tag{1}$$

A master key λ is generated from the product of two-prime numbers p and q

$$\lambda = lcm(p-1, q-1) \tag{2}$$

where, λ is selected such that,

$$gcd(\lambda, n) = 1$$

The generator key g is of order λ(n) and can be found by selecting $\mu \in Z^*_{n^2}$ and

$$g = \mu^{2n} \tag{3}$$

The public key h is generated from the secret key z as follows.

$$h = g^z mod\ n^2 \tag{4}$$

Here, $pk = (n, g, h)\ and\ sk = (z, \lambda)$

Encryption and decryption. The encryption algorithm generates two ciphertexts (A, B). Choose a random number $r \in [0, \frac{n}{4}]$.

$$A = g^r mod\ n^2 \tag{5}$$

$$B = h^r(1 + m.n) mod\ n^2 \tag{6}$$

The decryption using the master key λ:

$$m = \frac{B^\lambda - 1 mod n^2}{n}.\lambda^{-1} mod\ n \tag{7}$$

3.2.2 Boneh Lynn Shacham building blocks

BLS is a cryptographic digital signature scheme that uses bilinear pairings on elliptic curves for efficient verification and unique properties like signature aggregation.

Key generation in BLS scheme. Let G_1, G_2 are pairing-friendly elliptic curve groups of prime order q. Select a random secret key $x \in [0, q-1]$ where q is the group order. Generate g_2 in group G_2, and compute public key y, where $e: G_1 \times G_2 \rightarrow G_T$ is the bilinear map.

For our application, let the student S_i has the secret key x_i,

$$x_i \leftarrow Z_q \tag{8}$$

where, x_i is the secret BLS key of student S_i.

Then, y_i is the corresponding public key is given as,

$$y_i = {g_2}^{x_i} \in G_2 \tag{9}$$

Signing and verification. Digital signature is produced by exponentiation of hashed message as given below:

$$\sigma_i = \big(H(m_i)\big)^{x_i} \in G_1 \tag{10}$$

Aggregated digital signature is generated as follows:

$$\sigma_{agg} = \prod_{i=1}^{N} \sigma_i \tag{11}$$

The verification is carried out for pair of aggregated signature and generator against hashed messages and aggregated public key y.

$$e\big(\sigma_{agg}, g_2\big) = \prod_{i=1}^{N} e(H(m_i), y_i) \tag{12}$$

3.3 Authentication and aggregation protocol

The proposed AUT-AGG protocol performs authentication using BLS digital signature scheme and aggregation operation using BCP algorithm in the cloud architecture. The protocol uses multiple SS's and single DR.

The sequence of operations involved in AUT-AGG protocol is presented below:

Generate the private/secret key (z, λ)

Public key (n, g, h) and random number r from BCP algorithm

Generate the secret key $(x_i)$and public key (y_i) aggregared public key (y) from BLS digital signature

Step 1: Encryption using BCP algorithm in SS.

$$A_i = g^{r_i} mod n^2 \tag{13}$$

$$B_i = h^{r_i}(1 + m_i.n) mod n^2 \tag{14}$$

Step 2: Generate the digital signature σ_i, aggregate digital signature σ_{agg}.

$$\sigma_i = \big(H(B_i)\big)^{x_i} \tag{15}$$

$$\{\sigma_{agg}\} = \sigma_1 \times \sigma_2 \times \dots\dots\dots \sigma_N \tag{16}$$

$$\mathrm{CC} \leftarrow (B_i, \sigma_{agg}, N)$$

Step 3: Verify the aggregate digital signature using the public keys of individual users y_i, and perform the aggregation of B_i.

$$e\big(\{\sigma_{agg}\}, g_2\big) = \prod_{i=1}^{N} e(H(m_i), y_i) \tag{17}$$

If the aggregate signature is validated, all students are authenticated together. If the signature validation fails, one or more students are not legitimate. Only under these circumstances, individual signatures are validated to identfy the non-legititmate student(s).

$$\{B\} = B_1 \times B_2 \times \dots\dots\dots B_j \tag{18}$$

Step 4: Generate the digital signature σ_c for B along with the count N, using HASH function and secret key x_c in cloud.

$$\sigma_c = \big(H(B||N)\big)^{x_c} \tag{19}$$

DR receives $\{B, \sigma_c, N\}$

Step 5: Verify the digital signature along with the value N, at DR using cloud BLS public key y_c and then decrypt the value of $\{B\}$

$$e(\{\sigma_c\}, g_2) = e(H(\{B||N\}), y_c) \tag{20}$$

$$m = \frac{\{B\}^{\lambda} - 1 mod n^2}{n}.\lambda^{-1} mod n \tag{21}$$

3.3.1 Encryption at Students Site

During this phase, messages from SS are encrypted using the pk of the same SS as in Eqs. (13) and (14). It generates pair of ciphertexts (A_i, B_i) for each student ratings. The ciphertext (B) depends upon the ratings of students. Hence only ciphertext (B_i) is used to compute the aggregated values of students' ratings. Using the B_i ciphertext, a digital signature is generated for each student i and is represented as σ_i. The digital signature is generated with the help of HASH function and secret key as shown in Eq. (15).

The digital signature for individual student is unique, since it uses separate secret keys for each student. All these digital signatures are combined together to get the aggregated digital signature $\{\sigma_{agg}\}$ as given in Eq. (16). The SS sends the ciphertext B_i and digital signature $\{\sigma_{agg}\}$ along with the count of students (N) to the CC.

3.3.2 Ciphertext computation at Cloud Computer

During this phase, the received aggregated digital signature $\{\sigma_{agg}\}$ is compared with the HASH function as given in Eq. (17). After the successful verification of aggregated digital signature in CC, all B_i ciphertext of students' ratings are taken and its ciphertext multiplication is carried out to get the aggregated value ciphertext $\{B\}$, which is given in Eq. (18). Along with $\{B\}$ and N, a digital signature σ_c for $\{B\}$ is also obtained as given in Eq. (19) and is sent to DR. σ_c verified at the DR before decryption.

3.3.3 Retrieval of result at Data Requester

During this phase, the received digital signature σ_c is verified with the HASH function value of $\{B\}$ as given in Eq. (20). After the verification of digital signature, the decryption of aggregated $\{B\}$ is carried out to get the aggregated ratings as shown in Eq. (21). The student count N is directly taken from CC and it is used to calculate the average rating.

3.4 Threat model

The proposed AUT-AGG protocol assumes an honest-but-curious CC that faithfully executes protocol operations while attempting to learn additional information from encrypted data. User ratings are assumed to be valid values in the range of 1–10. Ensuring input correctness against malicious users who intentionally submit out-of-range encrypted values is outside the scope of this work. DR is trusted for authorized decryption of aggregated ciphertexts and is restricted from accessing individual ciphertext pairs (A_i, B_i). The protocol assumes that CC and DR do not collude. Registered students are considered legitimate users submitting valid feedback through authorized procedures.

The current system utilizes timestamps to record data transmission events and verify successful data reception at the CC. However, a dedicated SID-based freshness verification and replay-protection mechanism is beyond the scope of the present implementation and will be incorporated in future work.

The student count (N) is assumed to be available at the CC and is transmitted to the DR for the computation of average ratings. Any deviation in the value of 'N' would result in signature mismatch at the DR due to the cryptographic binding in the hash function. However, the inclusion of the identifiers such as the course, instructor, and survey session in the signature is not considered in the present work. Enhancing security through the inclusion of these metadata will be considered in future work.

KGMC is trusted for system initialization, key generation, and key distribution; however, its compromise may affect system security. Under these assumptions, the AUT-AGG protocol ensures confidentiality, authentication, integrity, and privacy-preserving aggregation of student feedback.

4. PERFORMANCE EVALUATION

In this section, we first analyze the computational complexity and communication overheads of the proposed AUT-AGG protocol theoretically. Later the system performance in terms of computational complexity overheads is validated through experimental simulations.

4.1 Computation complexity

The proposed scheme uses the BCP technique [9] to generate ciphertexts (A, B). For N students, the encryption complexity is $(2N)ModExp, (2N)ModMul$ and $(N)ModAdd$ operations, while decryption at DR using the master key has similar complexity. For authentication, the BLS signature scheme requires $(N)ModExp$ and $(N)ModMul$ operations at SS, whereas CC performs one $ModExp$ operation for aggregated signature generation. Since students can only select predefined ratings between 1 and 10, arbitrary large integer inputs are prevented.

The computational cost at SS increases with the number of students due to $ModExp$ operations in the BCP and BLS schemes. In contrast, CC performs only ciphertext multiplication, resulting in lower overhead, while DR processes only the final aggregated result, keeping its computation cost minimal. The overall complexity is summarized in Table 3.

Since BLS is an elliptic-curve pairing-based signature scheme, its computational cost is characterized using BLS-specific cryptographic primitives rather than only generic modular arithmetic operations. Accordingly, Table 4 reports the number of Hash-to-Curve (H2C), Elliptic-Curve Scalar Multiplication (EC-SM), Elliptic-Curve Point Addition (EC-Add), and pairing-product operations. At the SS, M distinct messages require N BLS signing operations followed by aggregation of the N signatures. At the CC, verification of the aggregated signature requires N H2C operations and a pairing-product computation involving $N + 1$ pairing terms. The CC also generates one BLS signature for a single message, which is subsequently verified by the DR. This separate accounting provides a more accurate representation of the computational requirements of BLS.

Table 3. Computational complexity in Bresson–Catalano–Pointcheval (BCP) scheme

Entity	ModExp	ModMul	ModAdd
SS	$2N + N$	$2N + N$	N
CC	1	N	-
DR	1	2	1

Note: Students Site (SS); Cloud Computer (CC); Data Requester (DR).

Table 4. Computational complexity in Boneh Lynn Shacham (BLS) scheme

	Operation	H2C	EC-SM	EC-Add	Pairing Product
SS	N BLS signatures + aggregation	N	N	$N - 1$	0
CC	Verification of aggregated signature	N	0	0	1 product involving $N + 1$ pairing terms
CC	BLS signing of one message	1	1	0	0
DR	Verification of one BLS signature	1	0	0	1 product involving 2 pairing terms

Note: Hash-to-Curve (H2C); Elliptic-Curve Scalar Multiplication (EC-SM); Elliptic-Curve Point Addition (EC-Add); Students Site (SS); Cloud Computer (CC); Data Requester (DR).

4.2 Communication complexity

The encrypted feedback generated at SS contains two ciphertexts of size $2L$ bits each, where $L = 2048$ bits considered for the simulation. Since only one ciphertext (B_i) is transmitted from SS to CC, the communication overhead per message is $2L$ bits. The BLS individual signature size is 48 bytes (384 bits), denoted as S bits. Although the total signature size at SS is $N \times S$ bits, only one aggregated signature of size S bits is transmitted to CC. Similarly, CC forwards one aggregated ciphertext and one signature to DR, resulting in an overhead of $2L + S$ bits. The communication overheads between entities are summarized in Table 5.

Table 5. Communication Complexity

Entity	Communication Overheads in Bits
SS to CC	$N \times 2L + 1S$
CC to DR	$2L + 1S$

Note: Students Site (SS); Cloud Computer (CC); Data Requester (DR).

4.3 Experimental results

AUT-AGG protocol was implemented and evaluated using Python 3.9 on a Windows 11 platform with Spyder IDE, Intel Core i5-12400 processor (2.5 GHz), and 16 GB RAM. A 2048-bit modulus is used in the proposed AUT-AGG protocol to ensure strong security against factorization attacks and unauthorized ciphertext recovery. The selected key size follows standard cryptographic recommendations and provides secure operation of the BCP HE scheme while enhancing long-term confidentiality and privacy protection. The security properties of AUT-AGG are directly inherited from the underlying BCP and BLS schemes. Results are obtained by averaging ten simulation runs. Initially, the encryption and decryption performance of the BCP algorithm was analyzed, as shown in Figure 2.

The simulation considers student feedback ratings in the range of 1–10, and analyzes the impact of key size and number of students on computation time. Figure 3 presents the computation time for key sizes from 512 to 2048 bits using feedback from two-students. The results show that encryption and decryption times increase with key size, while decryption at DR remains lower than encryption at SS. Additionally, signature verification time at CC is higher than signing time at SS.

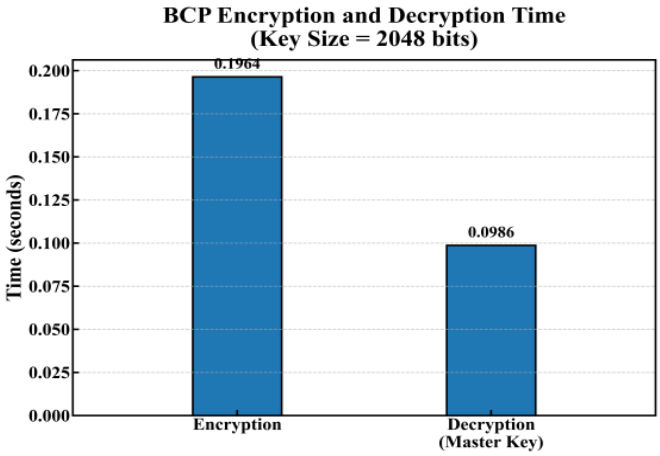

Figure 2. Encryption and decryption time in Bresson–Catalano–Pointcheval (BCP) crypto system for one feedback rating

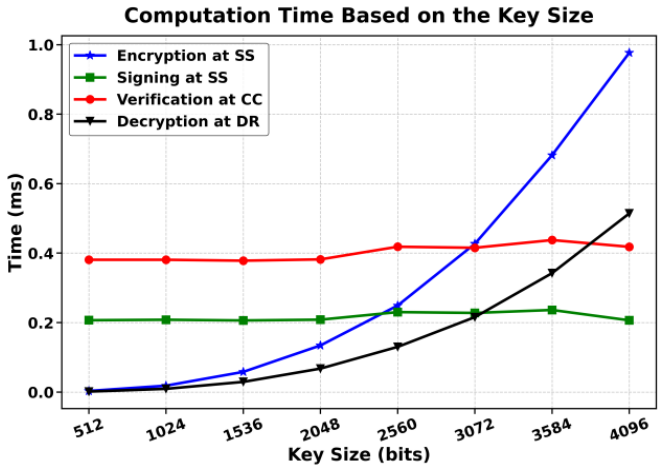

Figure 3 Computation time based on key size for two-student feedback ratings

Figure 4 shows the large-scale student participation experiment where the number of students is varied from 100 to 1000. As the number of feedback ratings increases, the encryption and signing times at SS also increase. Since CC receives only one aggregated signature, the communication overhead is significantly reduced. However, verification at CC requires checking all public keys along with the aggregated ciphertext, making the verification time slightly higher than the signing time at SS. As AUT-AGG produces only one aggregated result, the decryption time at DR remains very small and nearly constant, as shown in Figure 4.

The BLS digital signature verification time is higher than the signing time. The aggregated feedback rating ciphertext is digitally signed at CC, and is verified at DR before the decryption process. The computation time for this signing and verification is shown in Figure 5.

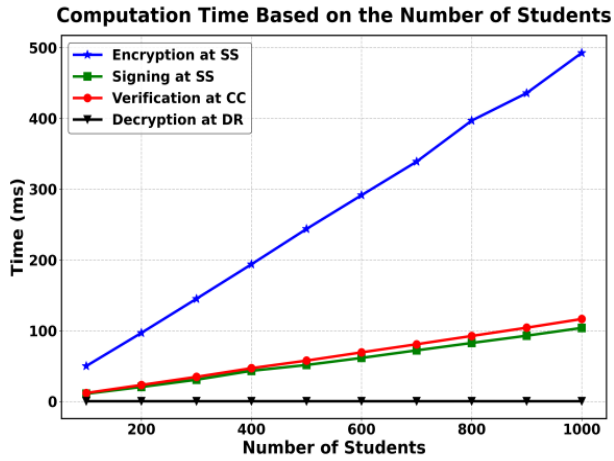

Figure 4. Computation time based on the number of students

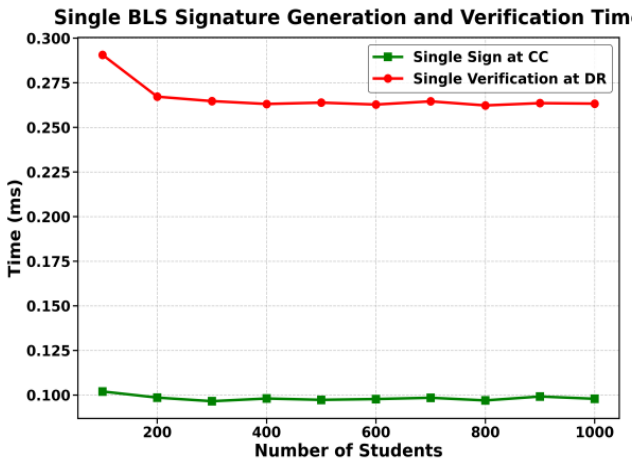

Figure 5. Signing and verification time as a function of number of students

5. CONCLUSION

In this paper, we proposed the AUT-AGG protocol, designed to support secure and privacy-preserving student feedback in higher education institutions. The proposed protocol enables each student's rating to be encrypted and digitally signed, ensuring both confidentiality and authenticity of the submitted data. Ciphertext aggregation is performed in the cloud, which eliminates the need for high computational

capability at the student side and improves overall system scalability. The degree of security supported by AUT-AGG is directly inherited from the underlying BCP and BLS schemes. Furthermore, transmitting only the aggregated digital signature to the central authority significantly reduces communication and storage overhead associated with individual signatures. Signature verification at the decision-making or review entity is simple and efficient, and the decryption process is fast due to the use of a master key. Overall, AUT-AGG provides strong security, privacy, and efficiency across all participating entities, making it suitable for large-scale feedback systems in educational environments. Future work will focus on integrating privacy-preserving range verification and SID-based replay-attack protection to further strengthen the security and robustness of the proposed protocol.

ACKNOWLEDGMENT

The authors thank Nitte (Deemed to be University), Mangaluru for providing the technical support during the experimental work.

REFERENCES

[1] Williams, A. (2024). Delivering effective student feedback in higher education: An evaluation of the challenges and best practice. International Journal of Research in Education and Science, 10(2): 473-501. https://doi.org/10.46328/ijres.3404

[2] Khan, Z., Sarkar, I. (2025). Efficacy of outcome-based education in enhancing English proficiency among engineering students in Hyderabad. Journal of Engineering Education Transformations, 38(2): 445-450. https://doi.org/10.16920/jeet/2025/v38is2/25054

[3] Aithal, P.S., Aithal, S. (2025). Student-centered approach in higher education to transform learning in India – A new ISL model. Poornaprajna International Journal of Management, Education & Social Science, 2(1): 81-103. https://doi.org/10.5281/zenodo.15349007

[4] Kanwar, A., Sanjeeva, M. (2022). Student satisfaction survey: A key for quality improvement in the higher education institution. Journal of Innovation and Entrepreneurship, 11: 27. https://doi.org/10.1186/s13731-022-00196-6

[5] Ramachandra, H.N., Joshi, D.N., Shivaprakasha, K.S. (2024). Data security paradigms: Paillier homomorphic encryption and AES in the context of privacy-preserving computation. In Proceedings of the International Conference on VLSI, Signal Processing, Power Electronics, IoT, Communication and Embedded Systems, Karkala, India, pp. 139-154. https://doi.org/10.1007/978-981-97-4657-6_11

[6] Abhijith, H.V. (2024). Efficient and secure data aggregation for resource-constrained IoT environments. International Journal of Safety and Security Engineering, 14(3): 999-1006. https://doi.org/10.18280/ijsse.140329

[7] Hegde, M.G., Ruthvika, B., Jain, R.B., et al. (2025). A privacy-preserving federated learning method with homomorphic encryption for chronic kidney disease stage prediction. Engineering, Technology & Applied Science Research, 15(4): 26019-26026. https://doi.org/10.48084/etasr.11928

[8] Paillier, P. (1999). Public-key cryptosystems based on composite degree residuosity classes. In Advances in Cryptology – EUROCRYPT '99, Lecture Notes in Computer Science, 1592, Berlin, Heidelberg: Springer, pp. 223-238. https://doi.org/10.1007/3-540-48910-X_16

[9] Bresson, E., Catalano, D., Pointcheval, D. (2003). A simple public-key cryptosystem with a double trapdoor decryption mechanism and its applications. In Advances in Cryptology – ASIACRYPT 2003, Lecture Notes in Computer Science, 2894, Berlin, Heidelberg: Springer, pp. 37-54. https://doi.org/10.1007/978-3-540-40061-5_3

[10] Cramer, R., Shoup, V. (1998). A practical public key cryptosystem provably secure against adaptive chosen ciphertext attack. In Advances in Cryptology – CRYPTO '98, Lecture Notes in Computer Science, 1462, Santa Barbara, CA, USA, pp. 13-25. https://doi.org/10.1007/BFb0055717

[11] Boneh, D., Lynn, B., Shacham, H. (2001). Short signatures from the Weil pairing. In Advances in Cryptology – ASIACRYPT 2001, Lecture Notes in Computer Science, 2248, Gold Coast, Australia, pp. 514-532. https://doi.org/10.1007/3-540-45682-1_30

[12] Rivest, R.L., Shamir, A., Adleman, L. (1978). A method for obtaining digital signatures and public-key cryptosystems. Communications of the ACM, 21(2): 120-126. https://doi.org/10.1145/359340.359342

[13] ElGamal, T. (1985). A public key cryptosystem and a signature scheme based on discrete logarithms. IEEE Transactions on Information Theory, 31(4): 469-472. https://doi.org/10.1109/TIT.1985.1057074

[14] Goldwasser, S., Micali, S. (1984). Probabilistic encryption & how to play mental poker keeping secret all partial information. Journal of Computer and System Sciences, 28(2): 270-299. https://doi.org/10.1016/0022-0000(84)90070-9

[15] Benaloh, J. (1994). Dense probabilistic encryption. In Proceedings of the Workshop on Selected Areas in Cryptography (SAC), pp. 120-128.

[16] Zhu, H., Li, D. (2008). Research on digital signature in electronic commerce. In Proceedings of the International Multiconference of Engineers and Computer Scientists (IMECS), Hong Kong, China, pp. 807-809.

[17] Natho, P., Boonmee, S., Khongkrairoek, N. (2025). Evaluation of digital signatures using RSA and HMAC algorithms. Engineering, Technology & Applied Science Research, 15(6): 29509-29514. https://doi.org/10.48084/etasr.14149

[18] Na, J., Kim, H.Y., Park, N., Seo, B. (2022). Comparative analysis of Schnorr digital signature and ECDSA for efficiency using private Ethereum network. IEIE Transactions on Smart Processing & Computing, 11(3): 231-239. https://doi.org/10.5573/IEIESPC.2022.11.3.231

[19] Schnorr, C.P. (1991). Efficient signature generation by smart cards. Journal of Cryptology, 4(3): 161-174. https://doi.org/10.1007/BF00196725

[20] Neuman, B.C., Ts'o, T. (1994). Kerberos: An authentication service for computer networks. IEEE Communications Magazine, 32(9): 33-38. https://doi.org/10.1109/35.312841

[21] Bonawitz, K., Ivanov, V., Kreuter, B., et al. (2017). Practical secure aggregation for privacy-preserving machine learning. In Proceedings of the 2017 ACM

SIGSAC Conference on Computer and Communications Security (CCS), Dallas, TX, USA, pp. 1175-1191. https://doi.org/10.1145/3133956.3133982

[22] Wang, P., Su, X., Jourenko, M., Jiang, Z., Larangeira, M., Tanaka, K. (2022). Environmental adaptive privacy preserving contact tracing system: A construction from public key rerandomizable BLS signatures. IEEE Access, 10: 37181-37199. https://doi.org/10.1109/ACCESS.2022.3164186

[23] Tong, S., Li, J., Fu, W. (2024). An efficient and scalable byzantine fault-tolerant consensus mechanism based on credit scoring and aggregated signatures. IEEE Access, 12: 10393-10410. https://doi.org/10.1109/ACCESS.2024.3352605

[24] Li, Z. (2022). A verifiable credentials system with privacy-preserving based on blockchain. Journal of Information Security, 13(2): 43-65. https://doi.org/10.4236/jis.2022.132003

[25] Bojič Burgos, J., Pustišek, M. (2024). Decentralized IoT data authentication with signature aggregation. Sensors, 24(3): 1037. https://doi.org/10.3390/s24031037

[26] Mustafa, M.A., Zhang, N., Kalogridis, G., Fan, Z. (2014). DESA: A decentralized, efficient and selective aggregation scheme in AMI. In Proceedings of the IEEE Innovative Smart Grid Technologies (ISGT), Washington, DC, USA, pp. 1-5. https://doi.org/10.1109/ISGT.2014.6816376

[27] Saleem, A., Khan, A., Malik, S.U.R., et al. (2020). FESDA: Fog-enabled secure data aggregation in smart grid IoT network. IEEE Internet of Things Journal, 7(7): 6132-6142. https://doi.org/10.1109/JIOT.2019.2957314

[28] Zhang, J., Zhao, Y., Wu, J., Chen, B. (2020). LVPDA: A lightweight and verifiable privacy-preserving data aggregation scheme for edge-enabled IoT. IEEE Internet of Things Journal, 7(5): 4016-4027. https://doi.org/10.1109/JIOT.2020.2978286

NOMENCLATURE

A, B	Ciphertext generated in BCP Algorithm
g	Generator key (Public key)
h	Public key generated from secret key z
$H(m)$	HASH function for the message m
m	Feedback rating in integers
n	Product of two-prime numbers (Public key)
N	Number of Students
p, q	Two large prime-numbers
pk	Public Key
r	Random number used in encryption process
sk	Secret Key
S_i	i^{th} student
x_i	Secret BLS key of student S_i
y_i	Public BLS key of student S_i
y	Aggregated public key
x_c	Secret BLS key of cloud
y_c	Public BLS key of cloud
z	Secret key used in BCP algorithm
$\{B\}$	Aggregated value of B ciphertext

Greek symbols

λ	Master key (Super-Secret key)
μ	Special key used to generate g
σ_i	Individual student BLS Signature
σ_{aqg}	Aggregated student BLS Signature
σ_c	BLS Signature of cloud