An Efficient and Fast Authentication Technique for E-Education with Special Refernce to Rural Areas

Pradip K. Samanta[1] , Anirban Panja[1 *], Soumen Mondal[1] and Dr. Sunil Karforma[2]
[1]Research Scholar, Department of Computer Science , The University of Burdwan , West Bengal, India
[2]Professor, Department of Computer Science , The University of Burdwan , West Bengal, India

ABSTRACT

E-education is commonly referred to as e-learning or online learning take place using different ICT-enabled tools and media. The backbone of the E-eduction process is the Internet through which educational pieces of information are delivered to the participants in efficient manner. In e-education, academic testimonials of a learner like admit card, mark sheet, registration certificate, etc are transacted through the Internet between learner and administrator of the educational organization. During transmission through the Internet, the valuable documents may be modified by intruders through different active and passive attacks. Digital Signature Algorithm may be applied to counter those attacks by authenticating the vulnerable documents during the transmission process through the Internet. In this paper, we have proposed an efficient and fast authentication technique using the Elgamal public key encryption-based Digital Signature Algorithm to prevent unauthorized access to e-learning documents. In the proposed model HMAC-SHA-256 is used to create the message digest of the input document. Then ElGamal digital signature algorithm is applied to the computed hashed value to generate and verify the signature. Cryptographically Secure Pseudo Random Number Generator (CSPRNG) is used to generate keys of ElGamal DSA to make the proposed model more secure and efficient. The performance and strength of the proposed model have also been analyzed and superiority of the proposed model is shown with the help of supportive tables and charts.

Keywords: E-Education**,** HMAC-SHA-256, Digital Signature Algorithm, ElGamal DSA, PRNG.

1. INTRODUCTION

The E-education system has been used to execute the teaching and learning process efficiently and securely among its users. In the present scenario, the impact of the COVID-19 pandemic has resulted in all educational institutes like Schools, Colleges, Universities, etc to implement online learning process. Consequently, effective learning procedure has also been changed using different ICT-enabled tools and technologies through the Internet. In the online learning process, several transactions are performed among learners, teachers, and administrators of the organaization which may include uploading documents like certificate, identity proof, photograph, signature and downloading documents like admit card, marksheet,

certificate, etc [5]. As the Internet is the backbone of e-learning system so, there exist several challenges caused by hackers in e-education systems by several types of attacks such as eavesdropping, replay attack, man in middle attack, brute force attack, side-channel attack, DOS attack, etc [4]. An appropriate authentication technique is needed to detect whether any modification took place or not on the transmitted data. In an e-education system sometimes a huge amount of data may be transmitted between participants. The process of encryption and decryption of the whole message requires more time and generates a large size of ciphertext. The already stated process is not feasible specifically in rural areas where the bandwidth of the channel of Internet connectivity is a big issue. So to resolve this issue, a digitally signed message may be transmitted instead of transmitting the whole encrypted message. To prevent any unexpected damage Digital Signature technique can be used as an authentication tool to identify a legitimate user. In our proposed model, we have applied HMAC-SHA-256 to create a message digest of the user's input data. The HMAC-SHA-256 uses two steps to compute the hash value of the user data. Based on the provided secret key, two intermediate keys are generated named inner and outer keys respectively, firstly an intermediate hash is computed using the user's input data and the inner key. Finally, in the second step actual hash is derived based on the intermediate hash of the first step and outer key. In this way, HMAC-SHA-256 provides better security compared to the traditional hash algorithm [16]. Then ElGamal-based Digital Signature algorithm [18] has been applied to the generated hash value to compute and verify the signature respectively. ElGamal's digital signature algorithm uses the benefits of discrete logarithmic problems to provide the security of the one-way function of exponentiation in modular rings [1]. We have used Cryptographically Secure Pseudo-Random Number Generator (CSPRNG) to generate all keys of the ElGamal-based DSA. That uses a secure and deterministic PRNG to expand the key into a long and fulfill all probabilistic tests of generation of random numbers polynomial time in the size of the seed [15]. It is known that the RSA [22]algorithm consumes extra time and space overhead due to complex integer factorization calculation because of its large key size. To resolve this problem, we have improved the performance of the DSA algorithm with the help of HMAC-SHA-256 () as the hash function to generate the message digest and ElGamal encryption algorithm to compute Digital Signature from generated message digest obtained from student's e-learning documents instead of RSA and Diffie-Hellman [23] public-key encryption algorithm.

2. STUDY FOR THE PROPOSED MODEL

In the past several authors have exploited many research work to impose security on the transactions associated with e-education or similar services with the help of different cryptographic techniques. In this section, study is made on different suggested research work for securing e-educational transactions with the help of different symmetric and asymmetric key cryptographic algorithms. A new signature scheme has been proposed by T. ElGamal [18], with the help of the Diffie-Hellman key exchange algorithm to achieve a public key cryptosystem. A hybrid approach integrating AES and RSA has been proposed by Hasib et al. [8] where AES has been used for the purpose of encryption of user input data and RSA for key management. A novel signature scheme has been proposed by R. Rivest et al. [19], to achieve a public key cryptosystem based on integer factorization problem. Banerjee et al. [7]

have been suggested how to secure and authenticate academic testimonials using LSB-based steganography in an e-learning system. A comparative study between AES and DES has been drawn by Mandal et al. [9] and shown AES consumes less memory compared to DES as well as AES is faster in execution compared to DES. Pasari et al. [10] explained the process of securing e-learning transactions using a hybrid approach integrating the benefits of IDEA and DSA algorithm and a message digest of input text is generated using a secure hash algorithm. MIHAILESCU et al. [11] proposed a framework in an e-learning system to optimize the quantity of data and to provide security protection for the data using elliptic curve cryptography. A. Mousa. [3] described security strength and performance of ElGamal public key encryption technique considering different parameters and criteria.

In this paper, a hybrid model utilizing the benefits of the ElGamal-based DSA and HMAC-SHA-256 technique is proposed to transmit and authenticate e-educational digital resources in an efficient way which is shown in the next section.

3. PROPOSED METHODOLOGY

In the proposed model learner of the e-education system acts as a client and initiates a transaction by uploading digital documents in a specified format. Before transmitting the uploaded document to the Administrator of the e-education system through the sharable medium several activities have been performed to impose security on the plain text document. We have used HMAC-SHA-256 to generate a message digest of the user's input data to impose additional security compared to the general hash algorithm. We have considered Cryptographically Secure Pseudo-Random Number Generator (CSPRNG) to randomly generate keys of the ElGamal-based DSA. Which generates secure and deterministic pseudo-random keys in polynomial time to execute operations of the ElGamal DSA technique. Finally, we have applied the benefits of the ElGamal-based DSA on the generated hash value to achieve authenticity, integrity, and nonrepudiation of the learner's digital documents. After transmission of the signature and all public parameters of the ElGamal DSA to the server-side (admin), verification of the transmitted signature started. Successful verification of the signature allowed the admin to send an acknowledgment of successful completion of the transaction to the learner otherwise, let the learner start the whole process again. The algorithm of signature generation, verification, and correctness of the algorithm are described below. The process flow of the proposed model is shown in Fig. 1.

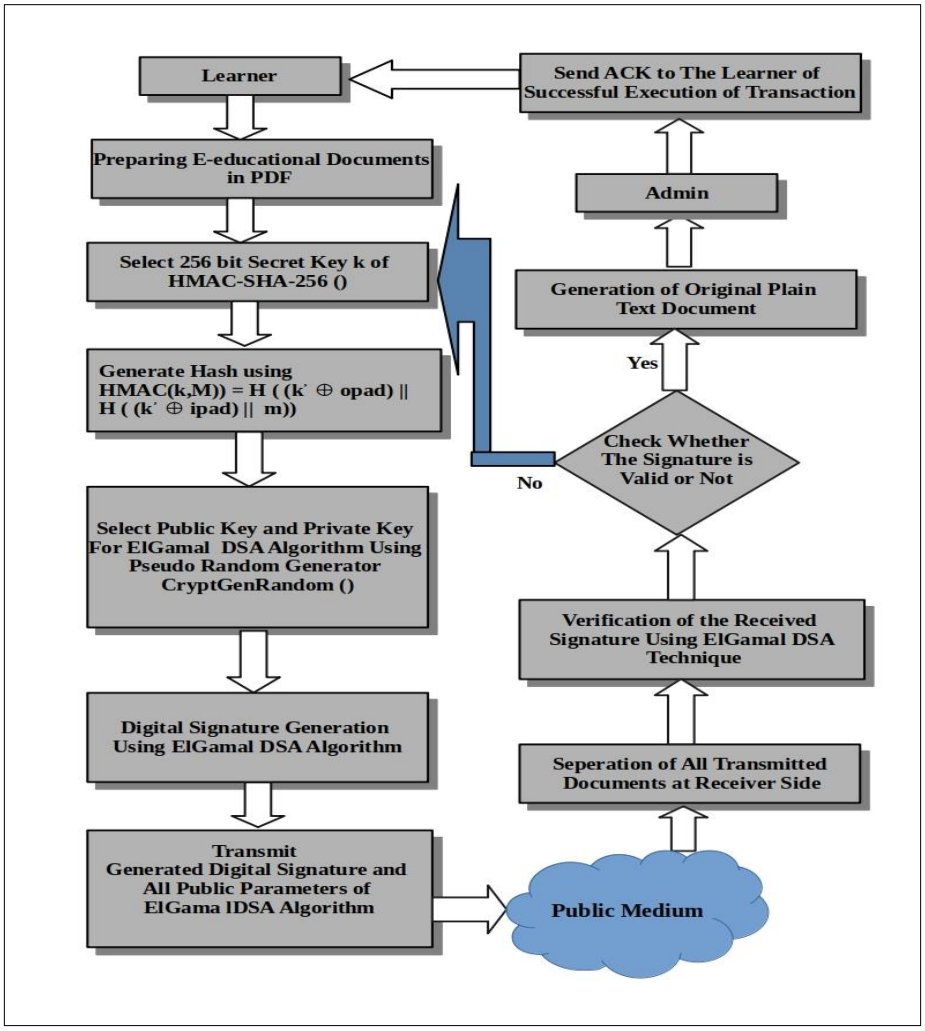

Figure 1. Process flow of the proposed model

3.1. GENERATIONOFTHE ELGAMAL DIGITAL SIGNATURE

Before transmitting the digital documents through the Internet Digital Signature of the input data is created is to achieve authentication, integrity, and nonrepudiation of the sent message. We have improved the performance of the ElGamal DSA using the HMAC-SHA-256 () hash function and Cryptographically Secure Pseudo-Random Number Generator (CSPRNG) to generate keys of the ElGamal DSA randomly. The generating process of a digital signature using the Elgamal DSA technique consists of the following activities.

Step 1: An N bit prime number q and HMAC-SHA-256 hash function with output L bits have been a consideration at the learner side. If L is larger than N, the leftmost N bits of the hash output have been considered.

Step 2: At the learner side, a primitive root g is randomly chosen such that $1 < g < q$ of the multiplicative group of integers modulo q, Z^*_q . The public parameters are (q,g), may be shared among the participants of the system. In addition, the seed value S is randomly taken to generate cryptographically secure pseudo- random value using CryptGenRandom() function such that $1 < S < q$.

Step 3: A secret random integer x is chosen at the learner side using pseudo random number generator function i.e., x = CryptGenRandom (sizeof (S), S) such that $1 < x < q$. Where S is the seed value generated at **step 2**.

Step 4: Now the public key of the system, $y = g^x \mod q$ is calculated at the learner side and published (q, g, y) as public for the different users of the system and kept x as the secret key for signing the digital signature.

Step 5: Users input message M is signed at the learner side using the following steps.

Step 6: Choose a random integer number k using CryptGenRandom() function i.e.,

k = CryptGenRandom (sizeof (S), S) such that $1 < k < q$. The users input message M acts as the input to HMAC-SHA-256() and k as the secret key of the hash function. At the learner side message digest value of the message M is calculated as follows:

HMAC (k , M) = H ((k' $\oplus$ opad) || H ((k'$\oplus$ ipad) || M)))

Where, k is the secret key, k' = H(k), if k is larger than the block size otherwise k' = k. || denotes concatenation operation, $\oplus$ denotes bitwise exclusive OR, ipad is the block-sized inner padding and opad is the block- sized outer padding.

Step 6: The secret random integer k, generated at **step 6** is used to generate the signature. Now compute, $r = g^k \mod q$, and

$s = (HMAC(M) - rx)\, k^{-1} \mod (q-1)$ such that $s \neq 0$, otherwise repeat again with different random value of k.

Step 8: The generated signature pair (r, s) and all public parameters of the system are sent to the admin side through the public channel for its verification and validation.

3.2. VERIFICATIONOFTHE DIGITAL SIGNATURE

On the server-side, the admin can verify the signature (r, s) with the help of all global parameters for the plain text message M using the following steps.

Step 1: To verify the digital signature check whether, $0 < r < q$ and $0 < s < q - 1$ and compute, $u_1 = g^{HMAC(M)} \mod q$, and check whether, $u_1 \equiv y^r \cdot r^s \pmod q$. The signature is valid if and only if the checking criteria holds true otherwise signature is invalid, abort the transaction and repeat all the steps discussed above.

Let us check the correctness of the algorithm so that a computed signature with the help of signing algorithm always be accepted by the verifier.

From the **Step 7** we have,

$$s = (HMAC(M) - r.x)\, k^{-1} \mod (q - 1)$$

$$s.k \equiv HMAC(M) - r.x \mod (q - 1)$$

$$HMAC(M) \equiv s.k + r.x \mod (q - 1)$$

Now as g is the primitive root and gcd(q , g) = 1 such that,

$g^{HMAC(M)} \equiv g^{(s.k + r.x)} \pmod q$

$g^{HMAC(M)} \equiv g^{(s.k)}\, g^{(r.x)} \pmod q$

$g^{HMAC(M)} \equiv g^{(k)s}\, g^{(x)r} \pmod q$

$g^{HMAC(M)} \equiv g^{(k)s} \pmod q .\ g^{(x)r} \pmod q$

$g^{HMAC(M)} \equiv r^{s}.\ y^{r} \pmod q$ [by **Step 4** and **Step 6**]

Therefore, the above algorithm of digital signature computation and verification is correct.

In this way the whole process of the proposed model is accomplished to achieve secure and fast delivery of the user's document to the intended receipent through the Internet.

4. RESULTAND PERFORMANCE ANALYSISOF THE PROPOSED MODEL

To get the most accurate experimental result of the proposed model we have executed the program of the proposed model several times for different sizes of input files. We have taken the average of the times thereby obtained after several times executing the program. Simulation of the ElGamal public-key encryption technique for large prime number and Digital Signature algorithm with HMAC-SHA-256() has been written in C++ programming language with support of the FLINT 2.5 library to achieve modular exponentiation for 512-bit prime number. All other supporting programs have also been designed in the same environment. System configuration from which experimental data have been extracted is Intel core i5 processor with a speed of 2.5 GHz, 4 GB DDR3 of RAM, and Ubuntu 20.04 LTS Operating System. The performance of the proposed model is measured considering the following experimental data given in different tables and charts for different sizes of input files.

Table 1. Duration of time needed to execute different steps of the DSA with Elgamal technique.

Size of the file (in Bytes)	**Amount of time needed to execute client side operations (in miliseconds)**		**Amount of time needed to execute server side operations (in miliseconds)**	
	Encryption time	**Signature computation time**	**Signature verification time**	**Deccryption time**
60416	15.47	4.18	4.25	15.53
105472	18.13	4.61	4.59	17.65
257027	21.91	5.76	5.79	21.31
592408	38.29	9.43	8.92	39.31
1055744	74.57	19.53	19.73	79.81
2040832	139.46	36.57	35.97	141.23

The experimental result of the amount of time needed to execute each step of the proposed technique has shown in Table 1. From the experimental result given inTable 1**,** it is seen that the amount of time needed to encrypt and decrypt a user's plain text input using ElGamal public-key encryption is approximately the same. It is also observed that amount of time needed to compute and verify the signature using ElGamal DSA with HMAC-SHA-256() is also near about the same for different sizes of input files.

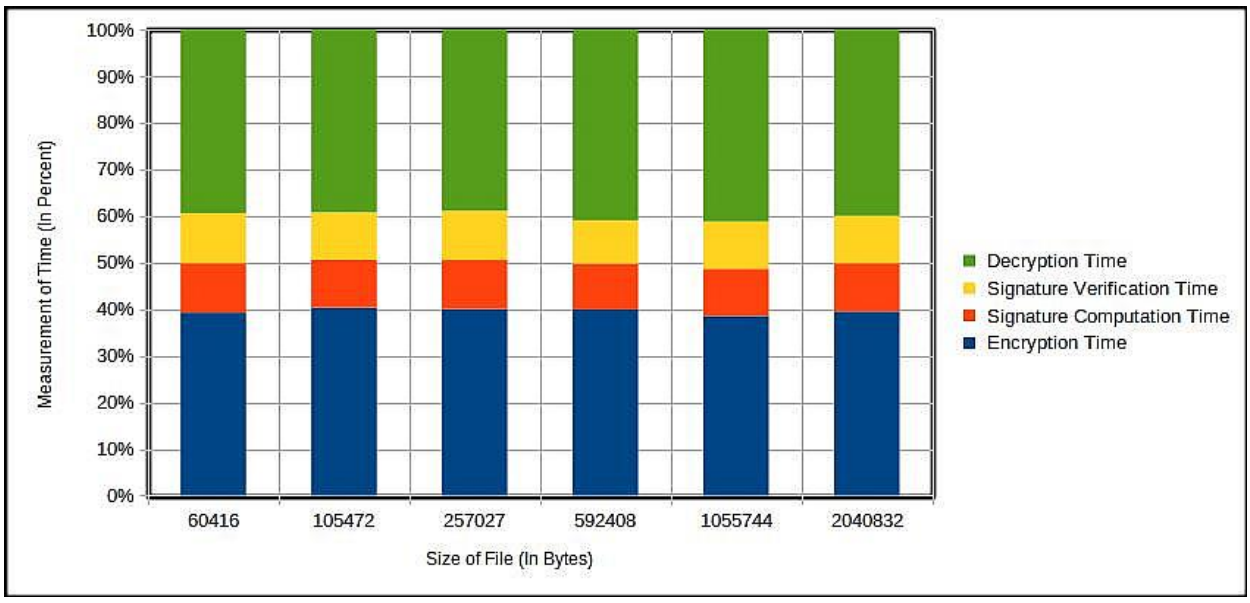

Figure 2. Percent of time needed to execute different steps of the Elgmal based DSA technique

Figure 2 shows the graphical representation of the percent of the time needed to complete each operation of the proposed model for different sizes of input files. From Figure 2, it is observed that near about 40% of the total processing time is spent on the encryption process, 40% is needed for the decryption process, 10% is required for signature generation and approximately 10% of the total time is required to verify the signature using DSA algorithm.

Table 2. Duration of total processing time required for different ecncryption technique with DSA algorithm

Size of File (in milisecond)	**Total Execution Time of RSA & DSA (in milisecond)**	**Total Execution Time of Deffile Hellman & DSA (in milisecond)**	**Total Execution Time of DES & DSA (in milisecond)**	**Total Execution Time of Elgamal & DSA (in milisecond)**
60416	28.31	31.26	32.51	39.43
105472	63.19	64.78	61.51	71.37
257027	108.27	111.33	119.18	118.68
592408	198.89	195.47	189.29	185.67
1055744	409.17	399.35	401.83	388.43
2040832	923.41	891.37	814.42	783.19

Table 2 shows the experimental result of the amount of time needed to accomplish encryption, decryption, signature generation, and signature verification for different encryption techniques with the DSA algorithm. From the experimental result, it is observed that the proposed ElGamal-based DSA model is a little bit faster than the other given models when the size of the input file is large.

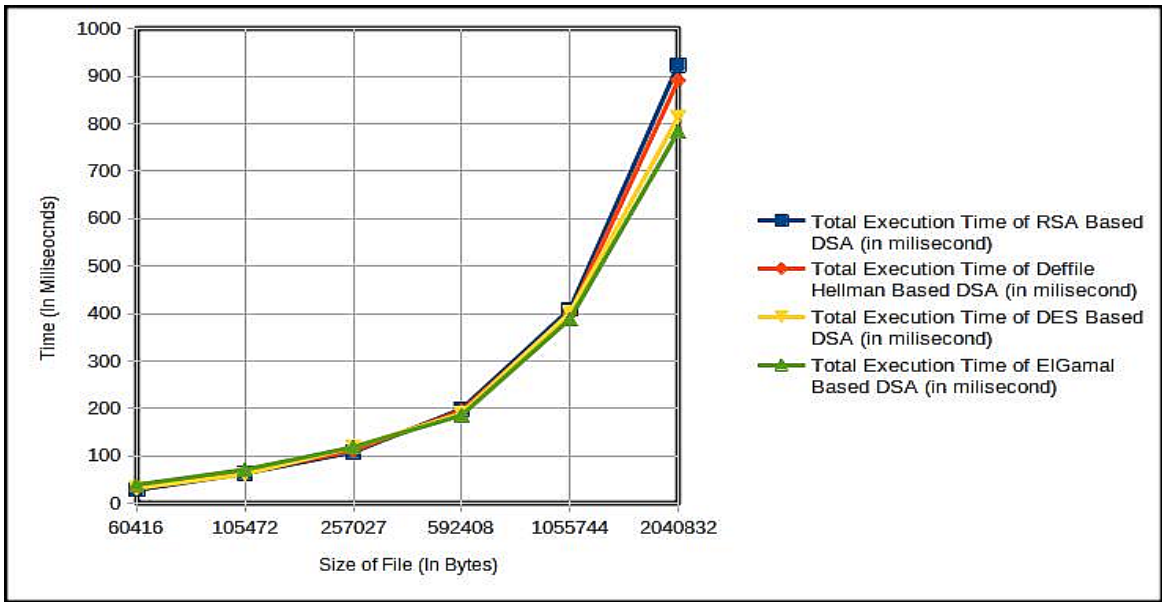

Figure 3. Comparison of total processing time for different technique with DSA algorithm

Figure 3 shows the graphical representation of the experimental result given in Table 2. From Figure 3, it is seen that the proposed model is faster in execution than the other given approaches.

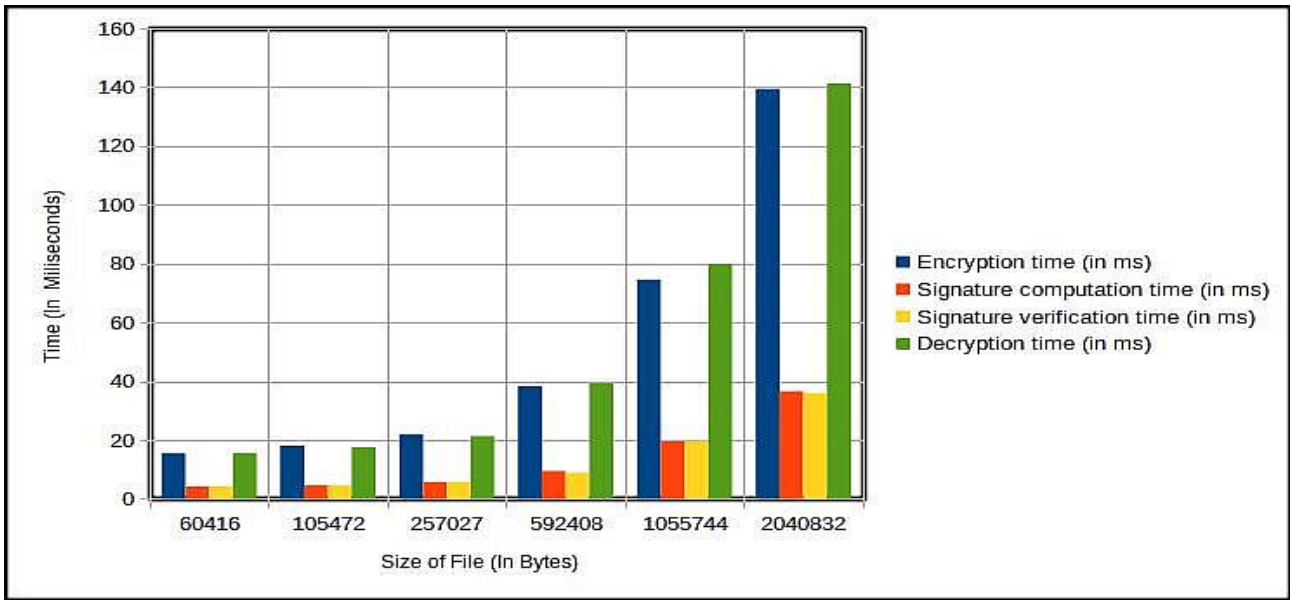

Figure 4. Contrast of processing times for different steps in Elgamal based DSA technique

Figure 4 represents comparison among various processing times such as encryption time, decryption time, signature generation and verification time which are required to complete an e-educational transaction in efficiently.

Table 3. Experimental result of signature generation and verification using different hash technique

Size of File (in bytes)	Signatre generation and verification time in MD 5 (in milisecond)	Signatre generation and verification time in SHA-1 (in milisecond)	Signatre generation and verification time in SHA-256 (in milisecond)	Signatre generation and verification time in HMAC- SHA-256()
60416	1.1	1.8	2.59	2.61
105472	2.9	3.19	4.21	4.36
257027	3.88	4.49	6.21	6.51
592408	6.37	7.89	11.38	11.46
1055744	16.38	17.38	20.39	20.49
2040832	29.78	31.84	39.17	39.43

Table 3 shows the amount of time needed to generate the message digest of the user's input file using the different message-digest algorithms. From the experimental result, it has been observed that HMAC-SHA-256() consumed more time to compute the digest compared to other given hashing techniques. But it is proven that security associated with HMAC-SHA-256 is relatively high than other hashing techniques.

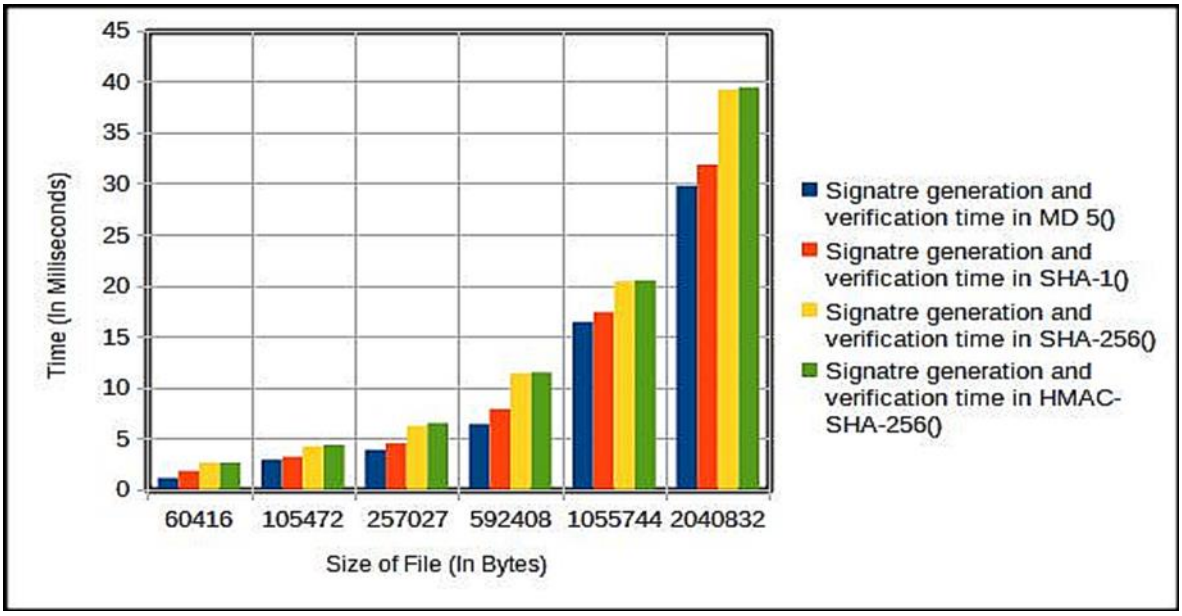

Figure 5. Comparison of signature generation and verification time using different hash technique

Figure 5 shows the graphical representation of the duration of time needed by different hash techniques to generate and verify a digital signature. From the above figure, it has been observed that complex calculations of HMAC-SHA-256() required maximum time to compute message digest value compared to other techniques. One more noticeable thing is that there is a negligible difference between the processing time of SHA-256() and HMAC-SHA-256().

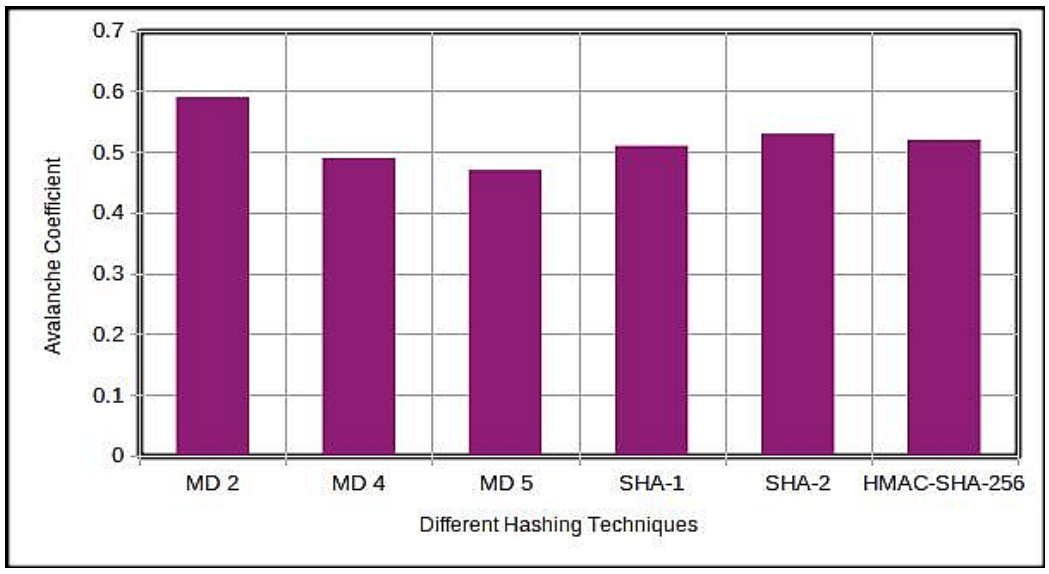

Figure 6. Avalanche criteria for different hashing techniques

Avalanche effect measures the significant changes in the output, which is 50% of the output bits changed even if for a single bit change in the input. Therefore, the avalanche test is a necessary condition for cryptographic algorithms to check whether the algorithm has strong randomization or not. Otherwise, there may be a probability that the input may be predicted being only the output available. Fig. 6 described that the avalanche coefficient for the HMAC-SHA-256() is above 0.5. So, it has secure randomization and is hard to break for an attacker.

5. SECURITY ANALYSISOFTHE PROPOSED MODEL

Major parts of the proposed model are the ElGamal public-key encryption-based Digital Signature algorithm with HMAC-SHA-256() hash function used to achieve authenticity, integrity, and non-repudiation of the user's plain text input. It is well-known that the ElGamal encryption system is a public-key encryption model, based on the difficulty of discrete logarithm problem where it is directly forward to raise numbers to large powers but it is much harder to do the inverse computation of the discrete logarithm [2]. In addition keys of the proposed model are generated using a cryptographically secure pseudo-random function which adds an extra layer of security to the proposed model. The security strength of (L, N) pair and HMAC-SHA-256() have been imposed in the ElGamal DSA algorithm which makes the algorithm hard to break. In the proposed model, we have used the HMAC-SHA-256() hash algorithm which has equal or greater security strength compared to the (L, N) pair. From the result section, it is observed that the avalanche coefficient of HMAC-SHA-256() is above

0.5, it is difficult for a cryptanalyst to predict the input being the output available. Therefore, security in our proposed model is not compromised at all.

6. CONCLUSION

To prevent unauthorized access of educational digital documents such as certificates, mark sheets, admit cards, photos, signatures, etc a secure and efficient technique is proposed integrating ElGamal public-key encryption and Digital Signature algorithm with SHA-256. The proposed technique provides strong cryptographic security to the e-educational documents without any extra computational overhead in terms of time and space. Result and performance analysis of the proposed model has been made using different experimental data and charts given in the performance analysis section. This exhibits that the proposed model is faster in execution compared to other given models. Security analysis of the proposed technique reveals that the strength of the security is properly maintained and not compromised. We can successfully apply the advantages of the proposed scheme is similar kinds of e-services such as e-governance, e-commerce, m-learning, etc. In the future, we may improve the strength of the security of the proposed model by imposing the features of the Elliptic Curve Digital Signature Algorithm (ECDSA) with the HMAC SHA-256 hash function.

REFERENCES

[1] W. Lee a, C. Wu a, W. Tsaur(2006), "A novel deniable authentication protocol using generalized ElGamal signature scheme" Information Sciences Elsevier Inc, Vol. 177 pp, 1376–1381.

[2] Shparlinski, I.E., (2004), "On the uniformity of distribution of the decryption exponent in fixed encryption exponent RSA" Inform. Proc. Lett., 92: 143-147.

[3] A. Mousa (2005) , "Security and Performance of ElGamal Encryption Parameters" Journal of Applied Sciences, Vol. 5, pp. 883-886.

[4] P.K. Samanta, K.A. Islm, A. Panja, S. Mondal and S. Karforma(2021), "AN IMPROVEDAUTHENTICATION SCHEME FOR E-LEARNING DOCUMENTS USING ELLIPTIC CURVE DIGITAL SIGNATURE ALGORITHM", Vidyabharati International Interdisciplinary Research Journal, ISSN 2319-4979.

[5] Anup Pasari, Kh. Amirul Islam, Sunil Karforma, Sripati Mukhopadhyay (2019), “Securing e-Learning Transactions using Digital Signature”, International Journal of Computer Sciences and Engineering, Vol.7, Special Issue.1.

[6] Stallings William(2014), "Cryptography and Network Security: Principles and Practice", Pearson Education, Sixth Edition.

[7] Banerjee, Soumendu, Sunil Karforma, and Akash Nag. (2017), "Applying LSB steganography for disseminating academic testimonials in e-learning and its authentication aspects". International Journal of Computer Trends and Technology, Vol. 47,Pp. 170-175.

[8] A. A. Hasib and A. A. M. MahmudulHaque.(2008), “A comparative study of the performance and security issues of AES and RSA cryptography”. Convergence and Hybrid information technology, Vol. 2, Pp. 505-510.

[9] A. K. Mandal, C. Parakash, and A.Tiwari. (2012), “Performance evaluation of cryptographic algorithms: DES and AES”. Electrical, Electronics and Computer Science (SCEECS).Pp. 1- 5.

[10] AnupPasari, KhAmirul Islam, Sunil Karforma, SripatiMukhopadhyay. (2019), “Securing e-Learning Transactions using Digital Signature”. International Journal of Computer Sciences and Engineering, Vol. 7, Pp. 249-256.

[11] Marius Iulian MIHAILESCU, StefaniaLoredana NITA, Pau Valentin Corneliu.(2020),"E-Learning System Framework Using Elliptic Curve Cryptography And Searchable Encryption". The International Scientific Conference eLearning and Software for Education; Bucharest, Vol. 1.
[12] Tanenbaum, Wetherall(2011), "Computer Networks" Pearson Education, Fifth Edition.
[13] Forouzan Behrouz A (20017), "Data Communications and Networking", Tata McGraw-Hill Publishing Company Ltd, Fourth Edition,2007.
[14] Fadia Ankit (2006), "Network Security", Macmillan Publishers India Ltd., Second Edition.
[15] https://en.wikipedia.org/wiki/Cryptographically secure_pseudorandom_number_generator
[16] https://en.wikipedia.org/wiki/HMAC
[17] R. Kasodhan and N. Gupta, "A New Approach of Digital Signature Verification based on BioGamal Algorithm," 2019 3rd International Conference on Computing Methodologies and Communication (ICCMC), 2019, pp. 10-15,doi: 10.1109/ICCMC.2019.8819710.
[18] T. ElGamal (1985), "A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms", IEEE TRANSACTIONS ON INFORMATION THEORY, VOL. IT-31, NO. 4, JULY.
[19] R. Rivest, A. Shamir, and L. Adleman (1978), "A method for obtaining digital signatures and public key cryptosystems," Commum. ACM, vol. 21, no. 2, pp. 120-126.
[20] L. Adleman, "A subexponential algorithm for the discrete logarithm problem with applications to cryptography," in Proc. 20th IEEE symp. Foundations of Computer Science 1979, pp. 55-60.
[21] https://en.wikipedia.org/wiki/ElGamal_signature_scheme
[22] R. L. Rivest, A. Shamir, L. Adleman, "A method for obtaining digital s/ignatures and public key cryptosystems", Commun. of the ACM, 21:120{126}. (1978).
[23] Diffie W, Hellman M, "New directions in cryptography", IEEE Trans Inf Theory Vol. 22 Issue: 6. (1976). pp. 644–654.